Genetic analysis using the sommer package

Giovanny Covarrubias-Pazaran

2017-03-01

The sommer package has been developed to provide R users with a powerful multivariate mixed model solver for different genetic and non-genetic analysis in diploid and polyploid organisms. This package allows the user to estimate variance components for a mixed model with the advantage of specifying the variance-covariance structure of the random effects and obtain other parameters such as BLUPs, BLUEs, residuals, fitted values, variances for fixed and random effects, etc.

The package is focused on genomic prediction (or genomic selection) and GWAS analysis, although general mixed models can be fitted as well. The package provides kernels to estimate additive (`A.mat`), dominance (`D.mat`), and epistatic (`E.mat`) relationship matrices that have been shown to increase prediction accuracy under certain scenarios. The package provides flexibility to fit other genetic models such as full and half diallel models as well.

Vignettes aim to provide several examples in how to use the sommer package under different scenarios in breeding and genetics. We will spend the rest of the space providing examples for:

1) Heritability (h^2) calculation
2) Half and full diallel designs
3) Genome wide association analysis (GWAS) in diploids and tetraploids
4) Genomic selection
5) Single cross prediction
6) Multivariate genetic models and genetic correlations
7) Multivariate GWAS

Background

The core of the package is the `mmer` function and solves the mixed model equations. An user friendly version named `mmer2` has been added as well, using the ASReml sintax. The functions are an interface to call one of the 4 ML/REML methods supported in the package; `EMMA` efficient mixed model association (Kang et al. 2008), `AI` average information (Gilmour et al. 1995; Lee et al. 2015), `EM` expectation maximization (Searle 1993; Bernardo 2010), and the default `NR` Newton-Raphson (Tunnicliffe 1989). All methods can estimate multiple random effect models specifying covariance structures.

The mixed model solved by the algorithms has the form:

$y = X\beta + Zu + \epsilon$

or

$y = X\beta + Zu_1 + ... + Zu_i + \epsilon$

where:

X is an incidence matrix for fixed effects

Z is an incidence matrix for random effects

β is the vector for BLUEs of fixed effects

u is the vector for BLUPs of random effects

ϵ are the residuals

The variance of the response is known to be the random part of the model:

$\text{Var(y)} = \text{Var}(Zu + \epsilon) = \text{ZGZ} + \text{R} = \text{V}$

and with

$\text{u} \sim \text{MVN(u, G)}$

$\epsilon \sim \text{MVN(u, R)}$

When multiple random effects are present the Z matrix becomes the column binding of each of the Z_i matrices for the i random effects. And the G matrix becomes the diagonal binding of each of the variance covariance structures (K matrices) for the random effects:

$$\mathbf{Z} = \begin{bmatrix} Z_1 & ... & Z_i \end{bmatrix}$$

$$\mathbf{G} = \begin{bmatrix} K_1\sigma^2_u 1 & 0 & 0 \\ 0 & ... & 0 \\ 0 & 0 & K_i\sigma^2_u i \end{bmatrix}$$

The program takes the Zs and Ks for each random effect and construct the neccesary structure inside and estimates the variance components by ML/REML using any of the 4 methods available in sommer; Average Information, Expectation-Maximization, Newton-Raphson, and Efficient Mixed Model Association. Please refer to the canonical papers listed in the Literature section to check how the methods work. We have tested widely the methods to make sure they provide the same solution when the likelihood behaves well but for complex problems they might lead to slightly different answers. If you have any concer please contact me at cova_ruber@live.com.mx or covarrubiasp@wisc.edu.

1) Marker and non-marker based heritability calculation

The heritability is one of the most famous parameters in breeding and genetics theory. The heritability is usually estimated as narrow sense (h^2; only additive variance in the numerator σ^2_A), and broad sense (H^2; all genetic variance in the numerator σ^2_G).

In a classical experiment with no molecular markers, special designs are performed to estimate and disect the additive (σ^2_A) and dominance (σ^2_D) variance along with environmental variability. Designs such as generation analysis, North Carolina designs are used to disect σ^2_A and σ^2_D to estimate the narrow sense heritability (h^2). When no special design is available we can still disect the genetic variance (σ^2_G) and estimate the broad sense heritability. In this example we will show the broad sense estimation which doesn't use covariance structures for random effects. For big models with no covariance structures, sommer's direct inversion is a bad idea to use but we will show anyways how to do it, for very sparse models we recommend using the lmer function from the lme4 package from Douglas Bates.

The dataset has 41 potato lines evaluated in 5 locations across 3 years in an RCBD design. We show how to fit the model and extract the variance components to calculate the h^2.

```
library(sommer)
data(h2)
head(h2)
```

```
##                 Name     Env Loc Year     Block y
## 1           W8822-3 FL.2012  FL 2012 FL.2012.1 2
## 2           W8867-7 FL.2012  FL 2012 FL.2012.2 2
## 3          MSL007-B MO.2011  MO 2011 MO.2011.1 3
## 4        CO00270-7W FL.2012  FL 2012 FL.2012.2 3
## 5 Manistee(MSL292-A) FL.2013  FL 2013 FL.2013.2 3
## 6          MSM246-B FL.2012  FL 2012 FL.2012.2 3
```

```
ans1 <- mmer2(y~1, random=~Name + Env + Name:Env + Block,data=h2, silent = TRUE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
```

```
vc <- ans1$var.comp
V_E <- vc[2,1];V_GE <- vc[3,1];V_G <- vc[1,1];Ve <- vc[5,1]

n.env <- length(levels(h2$Env))
h2 <- V_G/(V_G + V_GE/n.env + Ve/(2*n.env)) #the 2 is a reference for block
h2
```

```
## [1] 0.8594805
```

Recently with markers becoming cheaper, thousand of markers can be run in the breeding materials. When markers are available, an special design is not neccesary to disect the additive genetic variance. The estimation of the additive, dominance and epistatic relationship matrices allow us to estimate σ^2_A, σ^2_D and σ^2_I.

Assume you have a population, and a similar model like the one displayed previously has been fitted. Now we have BLUPs for the genotypes but in addition we have genetic markers. NOTICE WE WILL USE THE `mmer` FUNCTION THIS TIME.

```
data(CPdata)
CPpheno <- CPdata$pheno
CPgeno <- CPdata$geno
### look at the data
head(CPpheno)
```

```
##        id Row Col Year      color  Yield FruitAver Firmness
## P003 P003   3   1 2014 0.10075269 154.67     41.93  588.917
## P004 P004   4   1 2014 0.13891940 186.77     58.79  640.031
## P005 P005   5   1 2014 0.08681502  80.21     48.16  671.523
## P006 P006   6   1 2014 0.13408561 202.96     48.24  687.172
## P007 P007   7   1 2014 0.13519278 174.74     45.83  601.322
## P008 P008   8   1 2014 0.17406685 194.16     44.63  656.379
```

```
CPgeno[1:5,1:4]
```

```
##      scaffold_50439_2381 scaffold_39344_153 uneak_3436043 uneak_2632033
## P003                   0                  0            0            1
## P004                   0                  0            0            1
## P005                   0                 -1            0            1
## P006                  -1                 -1           -1            0
## P007                   0                  0            0            1
```

```
## fit a model including additive and dominance effects
y <- CPpheno$color
Za <- diag(length(y)); Zd <- diag(length(y)); Ze <- diag(length(y))
A <- A.mat(CPgeno) # additive relationship matrix
D <- D.mat(CPgeno) # dominance relationship matrix
E <- E.mat(CPgeno) # epistatic relationship matrix
```

```
ETA.ADE <- list(add=list(Z=Za,K=A),dom=list(Z=Zd,K=D),epi=list(Z=Ze,K=E))
ans.ADE <- mmer(Y=y, Z=ETA.ADE,silent = TRUE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
```

```
(H2 <- sum(ans.ADE$var.comp[1:3,1])/sum(ans.ADE$var.comp[,1]))
```

```
## [1] 0.7503613
```

```
(h2 <- sum(ans.ADE$var.comp[1,1])/sum(ans.ADE$var.comp[,1]))
```

```
## [1] 0.620319
```

In the previous example we showed how to estimate the additive (σ^2_A), dominance (σ^2_D), and epistatic (σ^2_I) variance components based on markers and estimate broad (H^2) and narrow sense heritability (h^2).

2) Half and full diallel designs

When breeders are looking for the best single cross combinations, diallel designs have been by far the most used design in crops like maize. There are 4 tipes of diallel designs depending if reciprocate and self cross (omission of parents) are performed (full diallel with parents n^2; full diallel without parents n(n-1); half diallel with parents 1/2 * n(n+1); half diallel without parents 1/2 * n(n-1)). In this example we will show a full dialle design (reciprocate crosses are performed) and half diallel designs (only one of the directions is performed).

In the first data set we show a full diallel among 40 lines from 2 heterotic groups, 20 in each. Therefore 400 possible hybrids are possible. We have pehnotypic data for 100 of them across 4 locations. We use the data available to fit a model of the form:

$y = X\beta + Zu_1 + Zu_2 + Zu_S + \epsilon$

We estimate variance components for GCA_1, GCA_2 and SCA and use them to estimate heritability. Additionally BLUPs for GCA and SCA effects can be used to predict crosses.

```
data(cornHybrid)
hybrid2 <- cornHybrid$hybrid # extract cross data
head(hybrid2)
```

```
##   Location GCA1   GCA2         SCA Yield PlantHeight
## 1        1 A258 AS5707 A258:AS5707    NA          NA
## 2        1 A258     B2     A258:B2    NA          NA
## 3        1 A258    B99    A258:B99    NA          NA
## 4        1 A258   LH51   A258:LH51    NA          NA
## 5        1 A258   Mo44   A258:Mo44    NA          NA
## 6        1 A258  NC320  A258:NC320    NA          NA
```

```
modFD <- mmer2(Yield~Location, random=~GCA1+GCA2+SCA, data=hybrid2,silent = TRUE, draw=FALSE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
```

```
summary(modFD)
```

```
##
## Information contained in this fitted model:
## * Variance components, Residuals, Fitted values
## * BLUEs and BLUPs, Inverse phenotypic variance(V)
## * Variance-covariance matrix for fixed & random effects
## * Predicted error variance (PEV), LogLikelihood
## Use the '$' symbol to access such information
## ========================================================
## Linear mixed model fit by restricted maximum likelihood
## ********************  sommer 2.5  **********************
## ========================================================
## Method:[1] "NR"
##
## logLik    AIC    BIC
##  -1340   2688   2703
## ========================================================
## Random effects:
##               VarComp VarCompSE  Zratio
## Var(GCA1)       0.000     8.109  0.0000
## Var(GCA2)       7.283    18.226  0.3996
## Var(SCA)      187.671    43.709  4.2937
## Var(Residual) 221.142    18.147 12.1860
## Number of obs: 400  Groups: 20 20 400
## ========================================================
## Fixed effects:
##                    Value   Std.Error t.value
## (Intercept)  1.3793e+02  2.1212e+00 65.0255
## Location2    2.7001e-13  2.1030e+00  0.0000
## Location3    7.8353e+00  2.1030e+00  3.7258
## Location4   -9.0975e+00  2.1030e+00 -4.3259
## ========================================================
## Use the '$' symbol to access all information
```

```
Vgca <- sum(modFD$var.comp[1:2,1])
Vsca <- modFD$var.comp[3,1]
Ve <- modFD$var.comp[4,1]
Va = 4*Vgca
Vd = 4*Vsca
Vg <- Va + Vd
(H2 <- Vg / (Vg + (Ve)) )
```

```
## [1] 0.7790698
```

```
(h2 <- Va / (Vg + (Ve)) )
```

```
## [1] 0.02910431
```

Don't worry too much about the small h2 value, the data was simulated to be mainly dominance variance, therefore the Va was simulated extremely small leading to such value of narrow sense h2.

In this second data set we show a small half diallel with 7 parents crossed in one direction. n(n-1)/2 crosses are possible $7(6)/2 = 21$ unique crosses. Parents appear as males or females indistictly. Each with two replications in a CRD. For a half diallel design a single GCA variance component can be estimated and an SCA as well ($\sigma^2_G CA$ and $\sigma^2_S CA$ respectively). And BLUPs for GCA and SCA of the parents can be extracted. We would create the design matrices in sommer using the `hdm` and `model.matrix` functions for the GCA and SCA matrices respectively.

$y = X\beta + Zu_g + Zu_s + \epsilon$

```
data(HDdata)
head(HDdata)
```

```
##   rep geno male female     sugar
## 1   1   12    1      2 13.950509
## 2   2   12    1      2  9.756918
## 3   1   13    1      3 13.906355
## 4   2   13    1      3  9.119455
## 5   1   14    1      4  5.174483
## 6   2   14    1      4  8.452221
```

```
# GCA matrix for half diallel using male and female columns
Z1 <- hdm(HDdata[,c(3:4)])
# SCA matrix
Z2 <- model.matrix(~as.factor(geno)-1, data=HDdata)
# Fit the model
y <- HDdata$sugar
ETA <- list(GCA=list(Z=Z1), SCA=list(Z=Z2)) # Zu component
modHD <- mmer(Y=y, Z=ETA,silent = TRUE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
```

```
summary(modHD)
```

```
##
## Information contained in this fitted model:
## * Variance components, Residuals, Fitted values
## * BLUEs and BLUPs, Inverse phenotypic variance(V)
## * Variance-covariance matrix for fixed & random effects
## * Predicted error variance (PEV), LogLikelihood
## Use the '$' symbol to access such information
## ============================================================
## Linear mixed model fit by restricted maximum likelihood
## ********************  sommer 2.5  **********************
## ============================================================
## Method:[1] "NR"
##
## logLik    AIC    BIC
## -58.18 118.36 120.09
## ============================================================
## Random effects:
##                VarComp VarCompSE Zratio
## Var(GCA)         5.508    3.5771  1.540
```

```
## Var(SCA)        1.816    1.3637  1.332
## Var(Residual)   3.117    0.9619  3.241
## Number of obs: 42  Groups: 7 21
## ========================================================
## Fixed effects:
##             Value Std.Error t.value
## Intercept 10.3332    1.8183  5.6828
## ========================================================
## Use the '$' symbol to access all information
```

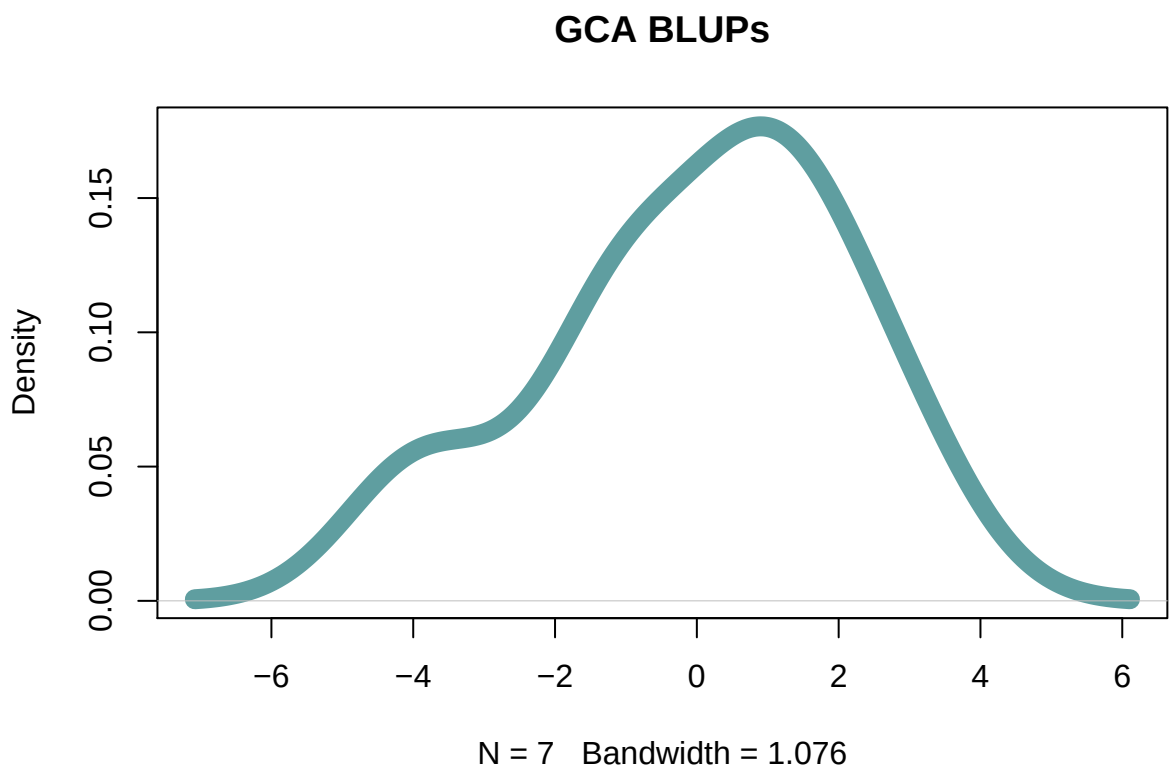
```
Vgca <- modHD$var.comp[1,1]
Vsca <- modHD$var.comp[2,1]
Ve <- modHD$var.comp[3,1]
Va = 4*Vgca
Vd = 4*Vsca
Vg <- Va + Vd
(H2 <- Vg / (Vg + (Ve/2)) ) # 2 technical reps
```

```
## [1] 0.9494872
```

```
(h2 <- Va / (Vg + (Ve/2)) )
```

```
## [1] 0.7140583
```

```
plot(density(randef(modHD)$GCA[,1]), col="cadetblue",
     lwd=10, main="GCA BLUPs")
```



3) Genome wide association analysis (GWAS) in diploids and tetraploids

With the development of modern statistical machinery the detection of markers associated to phenotypic traits have become quite straight forward. The days of QTL mapping using biparental populations exclusively are in the past. In this section we will show how to perform QTL mapping for diploid and polyploid organisms with complex genetic relationships. In addition we will show QTL mapping in biparental populations to clarify that the fact that is not required anymore doesn't limit the capabilities of modern mixed model machinery.

First we will start doing the GWAS in a biparental population with 363 individuals genotyped with 2889 SNP markers. This is easily done by creating the variance covariance among individuals and using it in the random effect for genotypes. The markers are added in the `W` argument to fit the model of the form:

$y = X\beta + Zu + Wg + \epsilon$

In this case $X\beta$ is the fixed part only for the intercept, Zu is the random effect for genotypes with the additive relationship matrix (A) as the variance-covariance of the random effect, Wg is the marker matrix and the effects of each marker. This is done in this way:

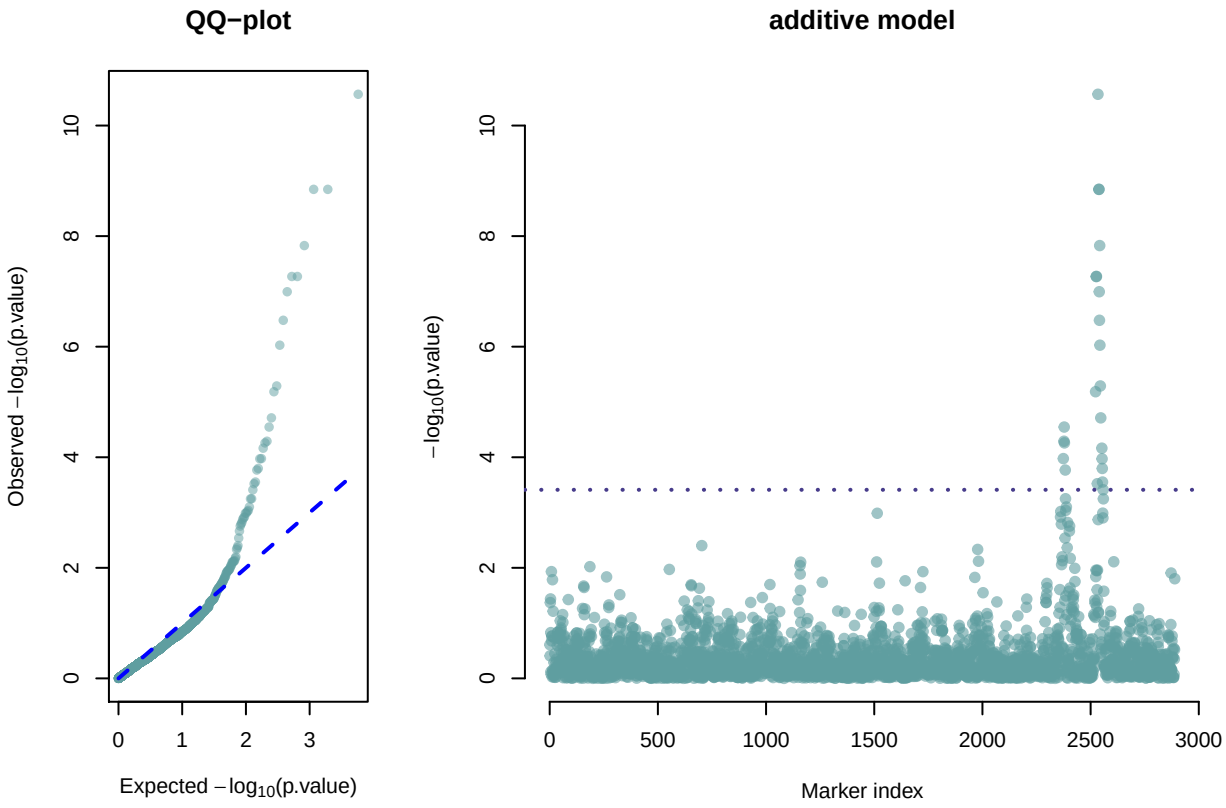
```
data(CPdata)
CPpheno <- CPdata$pheno
CPgeno <- CPdata$geno
my.map <- CPdata$map
### look at the data
head(CPpheno); CPgeno[1:5,1:4]
```

```
##        id Row Col Year      color  Yield FruitAver Firmness
## P003 P003   3   1 2014 0.10075269 154.67     41.93  588.917
## P004 P004   4   1 2014 0.13891940 186.77     58.79  640.031
## P005 P005   5   1 2014 0.08681502  80.21     48.16  671.523
## P006 P006   6   1 2014 0.13408561 202.96     48.24  687.172
## P007 P007   7   1 2014 0.13519278 174.74     45.83  601.322
## P008 P008   8   1 2014 0.17406685 194.16     44.63  656.379
```

```
##      scaffold_50439_2381 scaffold_39344_153 uneak_3436043 uneak_2632033
## P003                   0                  0            0            1
## P004                   0                  0            0            1
## P005                   0                 -1            0            1
## P006                  -1                 -1           -1            0
## P007                   0                  0            0            1
```

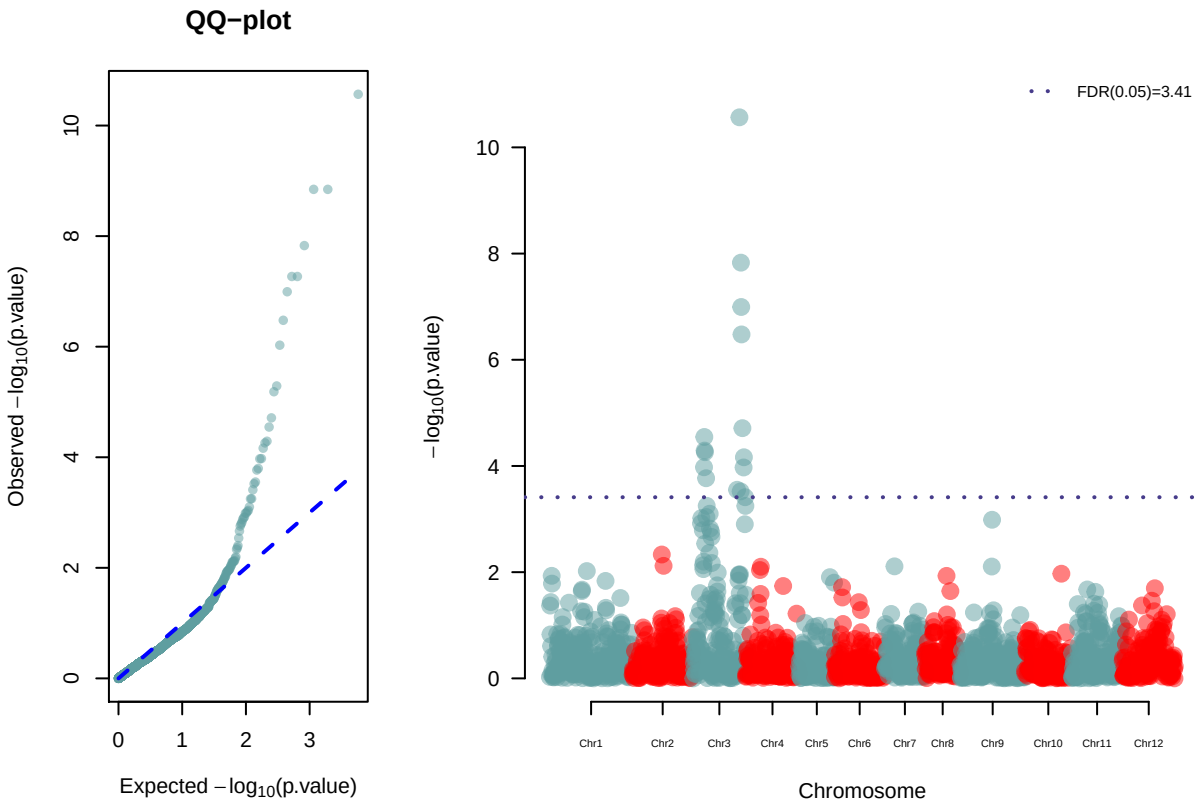
```
y <- CPpheno$color # response
Za <- diag(length(y)) # inidence matrix for random effect
A <- A.mat(CPgeno) # additive relationship matrix
ETA.A <- list(add=list(Z=Za,K=A)) # create random component
ans.A <- mmer(Y=y, Z=ETA.A, W=CPgeno, silent=TRUE) # fit the model
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
## Response is imputed for estimation of variance components in GWAS models.
```




```
### if you have a genetic map you can use it
ans.B <- mmer(Y=y, Z=ETA.A, W=CPgeno, silent=TRUE, map=my.map) # fit the model
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
## Response is imputed for estimation of variance components in GWAS models.
```



Now we will show how to do GWAS in a tetraploid using potato data. Is not very different from diploids. We only need to pay attention to the `ploidy` argument in the `atcg1234` and `A.mat` functions. In addition, when running the `mmer` model there is more models that can be implemented according to Rosyara et al. (2016).

```
data(PolyData)
genotypes <- PolyData$PGeno
phenotypes <- PolyData$PPheno
## convert markers to numeric format
numo <- atcg1234(data=genotypes, ploidy=4, silent = TRUE); numo[1:5,1:4]; dim(numo)
```

```
## Obtaining reference alleles
## Checking for markers with more than 2 alleles. If found will be removed.
## Converting to numeric format
## Calculating minor allele frequency (MAF)
## Imputing missing data with mode
```

```
##             c2_41437 c2_24258 c2_21332 c2_21320
## A96104-2           1        2        2        4
## A97066-42          2        3        2        4
## ACBrador           2        4        2        4
## ACLPI175395        0        4        0        4
## ADGPI195204        0        4        0        4
```

```
## [1]  221 3521
```

```r
# get only plants with both genotypes and phenotypes
common <- intersect(phenotypes$Name,rownames(numo))
marks <- numo[common,]; marks[1:5,1:4]
```

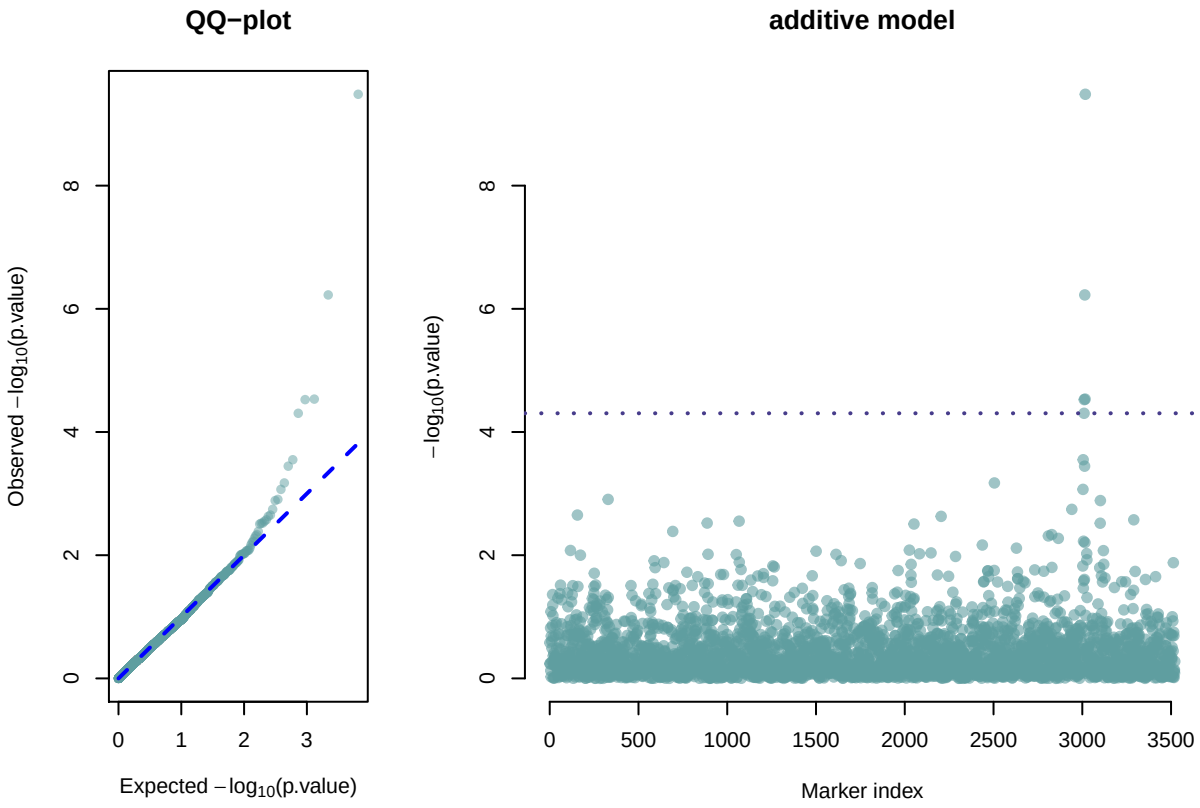
```
##                c2_41437 c2_24258 c2_21332 c2_21320
## A97066-42             2        3        2        4
## ACBrador              2        4        2        4
## AdirondackBlue        2        2        2        4
## AF2291-10             0        4        2        4
## AF2376-5              1        3        2        4
```

```r
phenotypes2 <- phenotypes[match(common,phenotypes$Name),];
phenotypes2[1:5,1:4]
```

```
##             Name total_yield chip_color tuber_eye_depth
## 1      A97066-42       13.10       2.35            3.03
## 2       ACBrador       15.56       2.63            4.37
## 3 AdirondackBlue       11.77       2.82            3.76
## 4      AF2291-10       13.43       1.50            4.50
## 5       AF2376-5       12.58       1.83            4.50
```

```r
# Additive relationship matrix, specify ploidy
yy <- phenotypes2$tuber_shape
K1 <- A.mat(marks, ploidy=4)
Z1 <- diag(length(yy))
ETA <- list( list(Z=Z1, K=K1)) # random effects for genotypes
# run the model
models <- c("additive","1-dom-alt","1-dom-ref","2-dom-alt","2-dom-ref")
ans2 <- mmer(Y=yy, Z=ETA, W=marks, method="EMMA",
             ploidy=4, models=models[1], silent = TRUE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
## Response is imputed for estimation of variance components in GWAS models.
```




```
summary(ans2)
```

```
##
## Information contained in this fitted model:
## * Variance components, Residuals, Fitted values
## * BLUEs and BLUPs, Inverse phenotypic variance(V)
## * Variance-covariance matrix for fixed & random effects
## * Predicted error variance (PEV), LogLikelihood
## Use the '$' symbol to access such information
## ============================================================
## Linear mixed model fit by restricted maximum likelihood
## ********************  sommer 2.5  **********************
## ============================================================
## Method:[1] "EMMA"
##
## logLik    AIC    BIC
## -192.5  387.0  390.3
## ============================================================
## Random effects:
##          VarComp
## V(u.1)   0.60778
## V(Error) 0.03807
## Number of obs: 187  Groups: 187
## ============================================================
## Fixed effects:
```

```
##              Value Std.Error t.value
## Intercept 3.307861  0.018302  180.74
## ============================================================
## Use the '$' symbol to access all information
```

4) Genomic selection

In this section we will use wheat data from CIMMYT to show how is genomic selection performed. This is the case of prediction of specific individuals within a population. It basically uses a similar model of the form:

$y = X\beta + Zu + \epsilon$

and takes advantage of the variance covariance matrix for the genotype effect known as the additive relationship matrix (A) and calculated using the `A.mat` function to establish connections among all individuals and predict the BLUPs for individuals that were not measured. The prediction accuracy depends on several factors such as the heritability (h^2), training population used (TP), size of TP, etc.

```
data(wheatLines)
X <- wheatLines$wheatGeno; X[1:5,1:4]; dim(X)
```

```
##      wPt.0538 wPt.8463 wPt.6348 wPt.9992
## [1,]       -1        1        1        1
## [2,]        1        1        1        1
## [3,]        1        1        1        1
## [4,]       -1        1        1        1
## [5,]       -1        1        1        1
```

```
## [1]  599 1279
```

```
Y <- wheatLines$wheatPheno
rownames(X) <- rownames(Y)
# select environment 1
y <- Y[,1] # response grain yield
Z1 <- diag(length(y)) # incidence matrix
K <- A.mat(X) # additive relationship matrix
# GBLUP pedigree-based approach
set.seed(12345)
y.trn <- y
vv <- sample(1:length(y),round(length(y)/5))
y.trn[vv] <- NA
ETA <- list(g=list(Z=Z1, K=K))
ans <- mmer(Y=y.trn, Z=ETA, method="EMMA", silent = TRUE) # kinship based
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
```

```
cor(ans$u.hat$g[vv],y[vv])
```

```
## [1] 0.4885687
```

```
## maximum prediction value that can be achieved
sqrt(ans$var.comp[1,1]/sum(ans$var.comp[,1]))
```

```
##    Var(g)
## 0.5771923
```

5) Single cross prediction

When doing prediction of single cross performance the phenotype can be dissected in three main components, the general combining abilities (GCA) and specific combining abilities (SCA). This can be expressed with the same model analyzed in the diallel experiment mentioned before:

$y = X\beta + Zu_1 + Zu_2 + Zu_S + \epsilon$

with:

$u_1 \sim \mathrm{N}(0,\ K_1\sigma_u^2 1)$

$u_2 \sim \mathrm{N}(0,\ K_2\sigma_u^2 2)$

$u_s \sim \mathrm{N}(0,\ K_3\sigma_u^2 s)$

And we can specify the K matrices. The main difference between this model and the full and half diallel designs is the fact that this model will include variance covariance structures in each of the three random effects (GCA1, GCA2 and SCA) to be able to predict the crosses that have not ocurred yet. We will use the data published by Technow et al. (2015) to show how to do prediction of single crosses.

```
data(Technow_data)

A.flint <- Technow_data$AF # Additive relationship matrix Flint
A.dent <- Technow_data$AD # Additive relationship matrix Dent
M.flint <- Technow_data$MF # Marker matrix Flint
M.dent <- Technow_data$MD # Marker matrix Dent

pheno <- Technow_data$pheno # phenotypes for 1254 single cross hybrids
pheno$hy <- paste(pheno$dent, pheno$flint, sep=":");head(pheno);dim(pheno)
```

```
##    hybrid dent flint     GY    GM      hy
## 1 518.298  518   298  -8.04 -0.85 518:298
## 2 518.305  518   305 -11.10  1.70 518:305
## 3 518.306  518   306 -16.85  2.24 518:306
## 4 518.316  518   316   2.08 -1.33 518:316
## 5 518.323  518   323   5.65 -2.71 518:323
## 6 518.327  518   327 -16.95 -0.52 518:327

## [1] 1254    6
```

```
# CREATE A DATA FRAME WITH ALL POSSIBLE HYBRIDS
DD <- kronecker(A.dent,A.flint,make.dimnames=TRUE)

hybs <- data.frame(sca=rownames(DD),yield=NA,matter=NA,gcad=NA, gcaf=NA)
hybs$yield[match(pheno$hy, hybs$sca)] <- pheno$GY
hybs$matter[match(pheno$hy, hybs$sca)] <- pheno$GM
hybs$gcad <- as.factor(gsub(":.*","",hybs$sca))
hybs$gcaf <- as.factor(gsub(".*:","",hybs$sca))
head(hybs)
```

```
##       sca yield matter gcad gcaf
## 1 513:316 10.02  -2.05  513  316
## 2 513:323  6.97  -3.78  513  323
## 3 513:330    NA     NA  513  330
## 4 513:336    NA     NA  513  336
## 5 513:340    NA     NA  513  340
## 6 513:341    NA     NA  513  341
```

```
# CREATE INCIDENCE MATRICES
Z1 <- model.matrix(~gcad-1, data=hybs)
Z2 <- model.matrix(~gcaf-1, data=hybs)
# SORT INCIDENCE MATRICES ACCORDING TO RELATIONSHIP MATRICES, REAL ORDERS
real1 <- match(  colnames(A.dent), gsub("gcad","",colnames(Z1)))
real2 <- match(  colnames(A.flint), gsub("gcaf","",colnames(Z2)))
Z1 <- Z1[,real1]
Z2 <- Z2[,real2]
# RUN THE PREDICTION MODEL
y.trn <- hybs$yield
vv1 <- which(!is.na(hybs$yield))
vv2 <- sample(vv1, 100)
y.trn[vv2] <- NA
ETA2 <- list(GCA1=list(Z=Z1, K=A.dent), GCA2=list(Z=Z2, K=A.flint))
anss2 <- mmer(Y=y.trn, Z=ETA2, method="EM", silent=TRUE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
## [1] "Names of columns in matrices Z and K for the 1 th random effect do not match."
## [1] "This can lead to incorrect estimation of variance components. Double check."
## [1] "Names of columns in matrices Z and K for the 2 th random effect do not match."
## [1] "This can lead to incorrect estimation of variance components. Double check."
## Column names of Z and K for random effect are not the same. Make sure they are in the correct order.
## Column names of Z and K for random effect are not the same. Make sure they are in the correct order.
```

```
summary(anss2)
```

```
##
## Information contained in this fitted model:
## * Variance components, Residuals, Fitted values
## * BLUEs and BLUPs, Inverse phenotypic variance(V)
## * Variance-covariance matrix for fixed & random effects
## * Predicted error variance (PEV), LogLikelihood
## Use the '$' symbol to access such information
## ============================================================
## Linear mixed model fit by restricted maximum likelihood
## ********************  sommer 2.5  **********************
## ============================================================
## Method:[1] "EM"
##
## logLik    AIC    BIC
##  -4998   9999  10004
## ============================================================
## Random effects:
##               VarComp
```

```
## Var(GCA1)       16.01
## Var(GCA2)       11.16
## Var(Residual)   17.71
## Number of obs: 1154  Groups: 123 86
## ========================================================
## Fixed effects:
##             Value Std.Error t.value
## Intercept 0.24778   0.20360  1.2169
## ========================================================
## Use the '$' symbol to access all information
```

```
cor(anss2$fitted.y[vv2], hybs$yield[vv2])
```

```
## [1] 0.8778897
```

In the previous model we only used the GCA effects (GCA1 and GCA2) for practicity, altough it's been shown that the SCA effect doesn't actually help that much in increasing prediction accuracy and increase a lot the computation intensity required since the variance covariance matrix for SCA is the kronecker product of the variance covariance matrices for the GCA effects, resulting in a 10578x10578 matrix that increases in a very intensive manner the computation required.

A model without covariance structures would show that the SCA variance component is insignificant compared to the GCA effects. This is why including the third random effect doesn't increase the prediction accuracy.

6) Multivariate genetic models and genetic correlations

Sometimes is important to estimate genetic variance-covariance among traits, multi-reponse models are very useful for such task. Currently sommer can deal with multivariate models for multiple random effects of the form:

$Y = X\beta + Zu + \epsilon$

with:

$$\mathbf{V} = \begin{bmatrix} Z_1 G_1 Z_1' + ... + Z_1 R_1 Z_1' & ... & Z_1 H_1 Z_t' + ... + Z_1 S_1 Z_t' \\ ... & ... & ... \\ Z_t H_1 Z_1' + ... + Z_t S_1 Z_1' & ... & Z_t G_1 Z_t' + ... + Z_t R_1 Z_t' \end{bmatrix}$$

where G are H are variance and covariance matrices among random effects for the "t" trait, and R and S are variance and covariance matrices among residuals. Here R=S=Iσ_ϵ , where I is an identity matrix. We can specify the covariance matrices. BLUPs will also be corrected for such covariances usually leading to more accurate predictions.

```
data(CPdata)
CPpheno <- CPdata$pheno[,-c(1:4)]
CPgeno <- CPdata$geno
### look at the data
head(CPpheno)
```

```
##           color  Yield FruitAver Firmness
## P003 0.10075269 154.67     41.93  588.917
## P004 0.13891940 186.77     58.79  640.031
## P005 0.08681502  80.21     48.16  671.523
```

```
## P006 0.13408561 202.96     48.24  687.172
## P007 0.13519278 174.74     45.83  601.322
## P008 0.17406685 194.16     44.63  656.379
```

```
CPgeno[1:5,1:4]
```

```
##      scaffold_50439_2381 scaffold_39344_153 uneak_3436043 uneak_2632033
## P003                   0                  0             0             1
## P004                   0                  0             0             1
## P005                   0                 -1             0             1
## P006                  -1                 -1            -1             0
## P007                   0                  0             0             1
```

```
## fit a model including additive and dominance effects
Y <- CPpheno
Za <- diag(dim(Y)[1])
A <- A.mat(CPgeno) # additive relationship matrix
####================####
#### ADDITIVE MODEL ####
####================####
ETA.A <- list(add=list(Z=Za,K=A))
ans.A <- mmer(Y=Y, Z=ETA.A, MVM=TRUE, method="EMMA")
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
## Running multivariate model
```

```
summary(ans.A)
```

```
## Information contained in this structure:
## * Results for a multi response model
## Displayed:
## * Variance-covariance component summaries
## Use the '$' sign to access parameters
## =========================================================
##     Multivariate Linear Mixed Model fit by REML
## ********************  sommer 2.5  ********************
## =========================================================
## Method:[1] "EMMAM"
##       logLik       AIC       BIC
## MVM 1849.151 -3696.302 -3692.407
## =========================================================
## Variance-Covariance components:
##
## Var-Covar(add)
##              color    Yield FruitAver Firmness
## color     0.005275   0.3241   0.06042    0.577
## Yield     0.324111 644.3235  26.73624  -81.254
## FruitAver 0.060416  26.7362  10.02137  -20.154
## Firmness  0.577043 -81.2542 -20.15401 1269.856
##
## Var-Covar(Residual)
```

```
##               color     Yield FruitAver   Firmness
## color      0.002692    0.2106   0.06136   -0.06473
## Yield      0.210586 4010.0260  29.18299  168.24899
## FruitAver  0.061364   29.1830  40.17632   72.37994
## Firmness  -0.064727  168.2490  72.37994 1201.09209
## ============================================================
## Fixed effects:
##            color Yield FruitAver Firmness
## Intercept 0.1825   133     45.39    638.8
## ============================================================
## Groups and observations:
##     Observ Groups
## add    363    363
## ============================================================
## Use the '$' sign to access parameters
```

Now you can extract the BLUPs using the 'randef' function or simple accesing with the '$' sign and pick 'u.hat'. Calculate genetic correlations and heritabilities easily.

```
## genetic variance covariance
gvc <- ans.A$var.comp$Vu
## extract variances (diagonals) and get standard deviations
sd.gvc <- as.matrix(sqrt(diag(gvc)))
## get possible products sd(Vgi) * sd(Vgi')
prod.sd <- sd.gvc %*% t(sd.gvc)
## genetic correlations cov(gi,gi')/[sd(Vgi) * sd(Vgi')]
(gen.cor <- gvc/prod.sd)
```

```
##               color       Yield  FruitAver    Firmness
## color     1.0000000  0.17580938  0.2627761  0.22296201
## Yield     0.1758094  1.00000000  0.3327245 -0.08982895
## FruitAver 0.2627761  0.33272445  1.0000000 -0.17865726
## Firmness  0.2229620 -0.08982895 -0.1786573  1.00000000
```

```
## heritabilities
(h2 <- diag(gvc) / diag(cov(Y, use = "complete.obs")))
```

```
##     color     Yield FruitAver  Firmness
## 0.8479870 0.1415729 0.2315060 0.4955782
```

7) Multivariate GWAS

Following the same theory of multivariate methods, theorically the marker effects can take advantage of the information contained in the correlation among traits besides exploiting the correlations between individuals. We have extended the GWAS framework to multivariate GWAS. Here, we will show a multivariate GWAS in a biparental population using the information of 4 traits.

```
data(CPdata)
CPpheno <- CPdata$pheno[,-c(1:4)]
CPgeno <- CPdata$geno
### look at the data
head(CPpheno)
```

```
##           color  Yield FruitAver Firmness
## P003 0.10075269 154.67     41.93  588.917
## P004 0.13891940 186.77     58.79  640.031
## P005 0.08681502  80.21     48.16  671.523
## P006 0.13408561 202.96     48.24  687.172
## P007 0.13519278 174.74     45.83  601.322
## P008 0.17406685 194.16     44.63  656.379
```

```
CPgeno[1:5,1:4]
```

```
##      scaffold_50439_2381 scaffold_39344_153 uneak_3436043 uneak_2632033
## P003                   0                  0            0             1
## P004                   0                  0            0             1
## P005                   0                 -1            0             1
## P006                  -1                 -1           -1             0
## P007                   0                  0            0             1
```

```
## fit a model including additive and dominance effects
Y <- CPpheno
Za <- diag(dim(Y)[1])
A <- A.mat(CPgeno) # additive relationship matrix
####================####
#### ADDITIVE MODEL ####
####================####
ETA.A <- list(add=list(Z=Za,K=A))
ans.A <- mmer(Y=Y, Z=ETA.A, W=CPgeno, MVM=TRUE, EIGEND=TRUE, IMP=TRUE,
              map=CPdata$map, silent=TRUE,gwas.plots = FALSE)
```

```
## Version out of date. Please update sommer to the newest version using:
## install.packages('sommer') in a new session
## Response is imputed for estimation of variance components in GWAS models.
## EIGEND feature activated. Eigen decomposition of K will be performed
```

```
summary(ans.A)
```

```
## Information contained in this structure:
## * Results for a multi response model
## Displayed:
## * Variance-covariance component summaries
## Use the '$' sign to access parameters
## ========================================================
##     Multivariate Linear Mixed Model fit by REML
## ******************  sommer 2.5  ********************
## ========================================================
## Method:[1] "MNR"
##      logLik     AIC      BIC
## MVM -589.47 1186.94 1208.063
## ========================================================
## Variance-Covariance components:
##
## Var-Covar(add)
##             color    Yield FruitAver Firmness
```

```
## color     0.77379  0.05767   0.10615  0.13693
## Yield     0.05767  0.14065   0.05644 -0.02426
## FruitAver 0.10615  0.05644   0.20929 -0.05616
## Firmness  0.13693 -0.02426  -0.05616  0.49266
##
## Var-Covar(Residual)
##              color   Yield FruitAver Firmness
## color      0.39916 0.03824   0.10787 -0.01527
## Yield      0.03824 0.88411   0.06308  0.04965
## FruitAver  0.10787 0.06308   0.83982  0.20656
## Firmness  -0.01527 0.04965   0.20656  0.47107
## ========================================================
## Standard errors for variance components:
##                                VarComp VarCompSE  Zratio
## u.1.color-color                0.77379   0.15483  4.9977
## u.1.color-Yield                0.05767   0.07745  0.7446
## u.1.color-FruitAver            0.10615   0.08481  1.2516
## u.1.color-Firmness             0.13693   0.09519  1.4385
## u.1.Yield-Yield                0.14065   0.07054  1.9940
## u.1.Yield-FruitAver            0.05644   0.05456  1.0345
## u.1.Yield-Firmness            -0.02426   0.06598 -0.3677
## u.1.FruitAver-FruitAver        0.20929   0.08355  2.5049
## u.1.FruitAver-Firmness        -0.05616   0.06882 -0.8160
## u.1.Firmness-Firmness          0.49266   0.11428  4.3111
## Residual.color-color           0.39916   0.04384  9.1057
## Residual.color-Yield           0.03824   0.04073  0.9390
## Residual.color-FruitAver       0.10787   0.04060  2.6567
## Residual.color-Firmness       -0.01527   0.03192 -0.4784
## Residual.Yield-Yield           0.88411   0.07531 11.7399
## Residual.Yield-FruitAver       0.06308   0.05288  1.1929
## Residual.Yield-Firmness        0.04965   0.04201  1.1819
## Residual.FruitAver-FruitAver   0.83982   0.07293 11.5155
## Residual.FruitAver-Firmness    0.20656   0.04267  4.8412
## Residual.Firmness-Firmness     0.47107   0.04643 10.1460
## ========================================================
## Fixed effects:
##              color     Yield FruitAver Firmness
## Intercept -0.02072 -0.002936   0.02698 -0.01001
## ========================================================
## Groups and observations:
##     Observ Groups
## add    363    363
## ========================================================
## Use the '$' sign to access parameters
```

Good luck with your analysis.

Literature

Covarrubias-Pazaran G. 2016. Genome assisted prediction of quantitative traits using the R package sommer. PLoS ONE 11(6):1-15.

Bernardo Rex. 2010. Breeding for quantitative traits in plants. Second edition. Stemma Press. 390 pp.

Gilmour et al. 1995. Average Information REML: An efficient algorithm for variance parameter estimation in linear mixed models. Biometrics 51(4):1440-1450.

Henderson C.R. 1975. Best Linear Unbiased Estimation and Prediction under a Selection Model. Biometrics vol. 31(2):423-447.

Kang et al. 2008. Efficient control of population structure in model organism association mapping. Genetics 178:1709-1723.

Lee et al. 2015. MTG2: An efficient algorithm for multivariate linear mixed model analysis based on genomic information. Cold Spring Harbor. doi: http://dx.doi.org/10.1101/027201.

Searle. 1993. Applying the EM algorithm to calculating ML and REML estimates of variance components. Paper invited for the 1993 American Statistical Association Meeting, San Francisco.

Yu et al. 2006. A unified mixed-model method for association mapping that accounts for multiple levels of relatedness. Genetics 38:203-208.

Abdollahi Arpanahi R, Morota G, Valente BD, Kranis A, Rosa GJM, Gianola D. 2015. Assessment of bagging GBLUP for whole genome prediction of broiler chicken traits. Journal of Animal Breeding and Genetics 132:218-228.

Tunnicliffe W. 1989. On the use of marginal likelihood in time series model estimation. JRSS 51(1):15-27.