

# Chapter 6: LMM Based Quantitative Genetic Analyses

Jixiang Wu

Tuesday, November 08, 2018

## Introduction

In this manual, we will concentrate on two R packages and their applications in quantitative genetics data analyses. Both R packages are based on linear mixed model (LMM) approaches and resampling techniques. The R package minque focuses on general linear mixed model system while the qgtools package focuses on several specific genetic models.

## Part I: Applications with the minque package

In the other R document for this Chapter (6), we focused on the various applications of the two LMM approaches. However, in this one, we will focus on applications of two specific cases: Case I (nested genetic model) and Case II (GGE model) ## Case I: Nested genetic model

If data are balanced, ANOVA based methods can be applied to obtain SS and MS, then variance components or mean comparisons for some factors. ANOVA based methods require the knowledge of expected mean squares (EMS) for appropriate F-tests and/or t-tests. However, we may also use linear mixed model analysis for this nested factor experimental design. By doing so, we will need to determine which effects are fixed or random. You need install an R package, minque, if you haven't done yet before conducting some of the following data analyses. The vertical bar "|" will be used to define or separate fixed effect(s) and random effects.

First we will need to load a data set built in the R package minque. This data set was from a NCI genetic mating design, which is also called nested genetic mating design. We will use this data set for various demonstrations depending random or fixed male and female effects.

```
require(mq0)

## Loading required package: mq0

data(ncii)
dat=ncii
```

And we will factorize all categorical variables in this data set.

```
dat=transform(dat,Female=factor(Female),Male=factor(Male),Rep=factor(Rep))
#dat$Female=factor(dat$Female)
#dat$Male=factor(dat$Male)
#dat$Rep=factor(dat$Rep)
```

The first case is for that both male and female are fixed while block is random.

```
#require(minque)
res=lmm.jack(Yld~Male+Male:Female|Rep,data=dat)

## Estimated variance components
data.frame(res$Var[[1]])

##      Estimate      SE  PValue X2.5.LL X97.5.UL
## V(Rep)    0.0399 0.126 9.78e-01  -0.406    0.486
## V(e)      92.5743 4.774 4.77e-08   75.711   109.438

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##      Estimate      SE PValue  X2.5.LL X97.5.UL
## V(Rep)/VP 0.000496 0.00157  0.978 -0.00504  0.00603
## V(e)/VP   0.999504 0.00157  0.000  0.99397  1.00504

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE  PValue X2.5.LL X97.5.UL
## mu          86.0685 0.445 0.00e+00   84.50   87.64
## Male(6)       0.0139 0.503 9.95e-01   -1.76    1.79
## Male(7)       5.4502 0.609 3.57e-05    3.30    7.60
## Male(8)      -5.4641 0.520 9.46e-06   -7.30   -3.63
## Male:Female(6:1) -17.3964 1.592 6.81e-06  -23.02  -11.77
## Male:Female(6:2) -15.3152 1.442 8.67e-06  -20.41  -10.22
## Male:Female(6:3)  14.9751 1.601 2.48e-05    9.32   20.63
## Male:Female(6:4)  20.0045 3.560 1.30e-03    7.43   32.58
## Male:Female(6:5)  -2.2540 3.300 9.06e-01  -13.91    9.40
## Male:Female(7:1)  16.1213 2.498 4.70e-04    7.30   24.94
## Male:Female(7:2)   6.1213 1.388 6.78e-03    1.22   11.03
## Male:Female(7:3)  20.0213 1.387 6.30e-07   15.12   24.92
## Male:Female(7:4) -27.0687 1.092 5.41e-09  -30.92  -23.21
## Male:Female(7:5)  -9.7452 2.117 5.12e-03  -17.22   -2.27
## Male:Female(8:1) -27.9867 1.588 1.10e-07  -33.60  -22.38
## Male:Female(8:2)   5.0457 1.810 8.14e-02   -1.35   11.44
## Male:Female(8:3)  37.1307 1.328 1.86e-09   32.44   41.82
## Male:Female(8:4) -33.8694 0.976 2.70e-10  -37.32  -30.42
## Male:Female(8:5)  14.2157 1.718 6.76e-05    8.15   20.28

## Predicted random effects: block effects only
data.frame(res$RandomEffect[[1]])
```

```
##           Pre      SE PValue X2.5.LL X97.5.UL
## Rep(1) -0.0682 0.2155  0.978  -0.829   0.693
## Rep(2) -0.0314 0.0993  0.978  -0.382   0.319
## Rep(3)  0.0762 0.2409  0.978  -0.775   0.927
## Rep(4)  0.0234 0.0739  0.978  -0.238   0.285
```

The second case is that both male and female are random.

```
res=lmm.jack(Yld~1|Male+Male:Female+Rep,data=dat)

## Estimated variance components
data.frame(res$Var[[1]])

##           Estimate      SE    PValue X2.5.LL X97.5.UL
## V(Male)           0.000  0.000  1.00e+00   0.00   0.00
## V(Rep)             0.111  0.351  9.78e-01  -1.13   1.35
## V(Male:Female)    491.497 18.736  3.28e-09  425.32  557.68
## V(e)              94.123  9.090  1.07e-05   62.02  126.23

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##           Estimate      SE    PValue X2.5.LL X97.5.UL
## V(Male)/VP        0.000000 0.000000  1.00e+00  0.000000  0.000000
## V(Rep)/VP          0.000199 0.00063  9.78e-01 -0.00203  0.00243
## V(Male:Female)/VP  0.839159 0.01387  1.85e-12  0.79017  0.88814
## V(e)/VP            0.160642 0.01402  4.57e-06  0.11111  0.21017

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## mu      86.1 0.262      0    85.2      87

## Predicted random effects: block effects only
data.frame(res$RandomEffect[[1]])

##           Pre      SE    PValue X2.5.LL X97.5.UL
## Male(6)      0.0000 0.000  1.00e+00   0.000   0.000
## Male(7)      0.0000 0.000  1.00e+00   0.000   0.000
## Male(8)      0.0000 0.000  1.00e+00   0.000   0.000
## Rep(1)     -0.0896 0.283  9.78e-01  -1.090   0.911
## Rep(2)     -0.0847 0.268  9.78e-01  -1.031   0.861
## Rep(3)       0.1252 0.396  9.78e-01  -1.274   1.524
## Rep(4)       0.0490 0.155  9.78e-01  -0.499   0.597
## Male:Female(6:1) -18.4487 1.392  1.31e-06 -23.365 -13.532
## Male:Female(6:2) -16.0714 1.551  1.06e-05 -21.548 -10.595
## Male:Female(6:3)  15.5896 1.126  9.01e-07  11.614  19.566
## Male:Female(6:4)  20.8091 1.520  9.96e-07  15.439  26.179
## Male:Female(6:5)  -1.8937 2.227  8.46e-01  -9.760   5.973
## Male:Female(7:1)  17.0433 2.666  5.05e-04   7.627  26.459
## Male:Female(7:2)   6.6738 1.945  2.96e-02  -0.197  13.545
```

```
## Male:Female(7:3) 21.1126 1.854 4.81e-06 14.563 27.663
## Male:Female(7:4) -27.9285 2.911 2.02e-05 -38.213 -17.644
## Male:Female(7:5) -9.8664 1.572 5.80e-04 -15.420 -4.313
## Male:Female(8:1) -29.7233 1.662 9.74e-08 -35.595 -23.851
## Male:Female(8:2) 4.9477 1.927 1.15e-01 -1.859 11.754
## Male:Female(8:3) 38.7519 1.417 2.27e-09 33.745 43.759
## Male:Female(8:4) -35.6372 2.092 1.49e-07 -43.028 -28.246
## Male:Female(8:5) 14.6413 2.264 4.63e-04 6.645 22.638
```

Based on the Chapter 5, with estimated male and female variance components, we can estimate genetic variance components such as additive and dominance variance components and then narrow- and broad-sense heritability.

The third case is that male is fixed while female is random.

```
res=lmm.jack(Yld~Male|Male:Female+Rep,data=dat)
## Estimated variance components
data.frame(res$Var[[1]])

##           Estimate      SE      PValue X2.5.LL X97.5.UL
## V(Rep)           0.0    0.0 1.00e+00      0      0
## V(Male:Female) 492.7 25.3 4.57e-08    403    582
## V(e)           94.4   7.2 1.44e-06     69    120

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##           Estimate      SE      PValue X2.5.LL X97.5.UL
## V(Rep)/VP      0.000 0.0000 1.00e+00   0.000   0.000
## V(Male:Female)/VP 0.839 0.0152 4.34e-12   0.785   0.893
## V(e)/VP        0.161 0.0152 8.99e-06   0.107   0.215

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##           Estimate      SE      PValue X2.5.LL X97.5.UL
## mu          86.0636 0.455 0.00e+00   84.46   87.67
## Male(6)    -0.0548 0.559 9.93e-01   -2.03    1.92
## Male(7)     6.5837 0.511 1.68e-06    4.78    8.39
## Male(8)    -6.5289 0.496 1.40e-06   -8.28   -4.78

## Predicted random effects: block effects only
data.frame(res$RandomEffect[[1]])

##           Pre      SE      PValue X2.5.LL X97.5.UL
## Rep(1)      0.00 0.00 1.00e+00    0.00    0.00
## Rep(2)      0.00 0.00 1.00e+00    0.00    0.00
## Rep(3)      0.00 0.00 1.00e+00    0.00    0.00
## Rep(4)      0.00 0.00 1.00e+00    0.00    0.00
## Male:Female(6:1) -18.38 1.41 1.48e-06  -23.35  -13.41
## Male:Female(6:2) -16.05 1.85 4.67e-05  -22.60   -9.51
## Male:Female(6:3) 15.77 1.69 2.50e-05    9.81   21.73
```

```
## Male:Female(6:4) 20.67 2.56 8.19e-05 11.63 29.71
## Male:Female(6:5) -2.01 1.70 6.84e-01 -8.03 4.01
## Male:Female(7:1) 15.74 1.43 6.23e-06 10.70 20.77
## Male:Female(7:2) 5.23 1.98 1.03e-01 -1.77 12.23
## Male:Female(7:3) 19.79 1.51 1.45e-06 14.46 25.12
## Male:Female(7:4) -29.63 1.79 1.90e-07 -35.95 -23.32
## Male:Female(7:5) -11.13 1.07 1.04e-05 -14.91 -7.35
## Male:Female(8:1) -28.38 1.94 5.61e-07 -35.23 -21.53
## Male:Female(8:2) 6.50 1.35 3.81e-03 1.73 11.27
## Male:Female(8:3) 40.26 1.27 6.18e-10 35.77 44.76
## Male:Female(8:4) -34.47 1.58 1.72e-08 -40.07 -28.88
## Male:Female(8:5) 16.09 1.41 4.79e-06 11.10 21.08
```

## Case 2: GGE model

It is very often that some experiments are conducted in different environments. For such experiments, randomized complete blocks in each environment are independent of those in another one. That means block effects are nested within each environment. It is very common that genotypes are grown in multiple environments, each with a randomized complete block design. The model used for such a data structure is typically called genotype and genotype-by-environment interaction (GGE) model. The GGE model has been analyzed throughoutly in the other document for this Chapter. Here, we will focus on analysis for a data set, `plrv`, from the R package `agricolae`. We will do factorization for block (Rep) and rename two variables genotype and locality with `Geno` and `Loc`, respectively.

```
library(agricolae)
data(plrv)
names(plrv)

## [1] "Genotype"      "Locality"      "Rep"           "WeightPlant"  "WeightPlot"
## [6] "Yield"

dat=plrv
dat=transform(dat,Geno=Genotype,Loc=Locality,Rep=factor(Rep))
names(dat)

## [1] "Genotype"      "Locality"      "Rep"           "WeightPlant"  "WeightPlot"
## [6] "Yield"         "Geno"          "Loc"
```

First we treat both genotype and environment as fixed. The following data analyses are focused on the MINQUE approach with a jackknife resampling technique. The results include four components: (1) estimated variance components, (2) estimated proportional variance components, (3) estimated fixed effects, and (4) predicted random effects. The results for these four components will be displayed accordingly.

```
#require(minque)

res=lmm.jack(Yield~Loc*Geno|Loc:Rep,data=dat)
## Estimated variance components
data.frame(res$Var[[1]])
```

```
##           Estimate      SE    PValue X2.5.LL X97.5.UL
## V(Loc:Rep)      2.08 0.695 5.95e-02  -0.379    4.53
## V(e)           37.00 2.181 1.54e-07  29.300   44.71
```

## Estiamted proportional variance components

```
data.frame(res$PVar[[1]])
```

```
##           Estimate      SE    PValue  X2.5.LL X97.5.UL
## V(Loc:Rep)/VP    0.053 0.0171 5.04e-02 -0.00754    0.114
## V(e)/VP         0.947 0.0171 4.19e-12  0.88646    1.008
```

## Estiamted fixed effects

```
data.frame(res$FixedEffect[[1]])[1:20,]
```

```
##           Estimate      SE    PValue X2.5.LL X97.5.UL
## mu          30.66029 0.158 0.00e+00  30.104   31.217
## Loc(Ayac)    -6.66457 0.301 1.48e-08  -7.728   -5.601
## Loc(Hyo-02)  14.53585 0.230 1.27e-12  13.722   15.349
## Loc(LM-02)   3.81865 0.190 3.45e-08   3.148    4.489
## Loc(LM-03)  22.33195 0.321 5.32e-13  21.198   23.466
## Loc(SR-02)  -15.16937 0.130 5.33e-15 -15.630  -14.708
## Loc(SR-03)  -18.85252 0.176 1.07e-14 -19.475  -18.230
## Geno(102.18) -3.72423 0.544 3.00e-04  -5.646   -1.803
## Geno(104.22)  0.48490 0.400 6.65e-01  -0.926    1.896
## Geno(121.31) -0.49618 0.547 8.22e-01  -2.427    1.435
## Geno(141.28)  7.71862 0.615 2.10e-06   5.546    9.892
## Geno(157.26)  5.30539 0.765 2.71e-04   2.605    8.006
## Geno(163.9)  -7.93666 0.295 2.64e-09  -8.979   -6.894
## Geno(221.19) -6.64929 0.614 7.39e-06  -8.819   -4.479
## Geno(233.11) -1.67817 0.425 1.34e-02  -3.180   -0.177
## Geno(235.6)   6.90386 0.428 2.38e-07   5.393    8.415
## Geno(241.2)  -3.71073 0.275 1.14e-06  -4.684   -2.738
## Geno(255.7)  -0.00165 0.611 9.96e-01  -2.160    2.157
## Geno(314.12) -2.26561 0.700 4.02e-02  -4.740    0.208
## Geno(317.6)   3.94842 0.419 2.36e-05   2.467    5.430
```

## Predicted random effects

```
data.frame(res$RandomEffect[[1]])
```

```
##           Pre      SE    PValue X2.5.LL X97.5.UL
## Loc:Rep(Ayac:1)  0.1749 0.337 0.94753  -1.016    1.366
## Loc:Rep(Ayac:2)  0.6161 0.452 0.58014  -0.979    2.211
## Loc:Rep(Ayac:3) -0.7910 0.448 0.36772  -2.374    0.792
## Loc:Rep(Hyo-02:1) -1.8835 0.717 0.10487  -4.416    0.649
## Loc:Rep(Hyo-02:2)  1.8639 0.708 0.10382  -0.636    4.364
## Loc:Rep(Hyo-02:3)  0.0195 0.199 0.99297  -0.682    0.721
## Loc:Rep(LM-02:1)  2.1893 0.406 0.00175   0.755    3.624
## Loc:Rep(LM-02:2)  0.6370 0.275 0.16882  -0.334    1.608
## Loc:Rep(LM-02:3) -2.8263 0.572 0.00321  -4.848   -0.805
## Loc:Rep(LM-03:1) -2.5866 0.560 0.00500  -4.564   -0.609
## Loc:Rep(LM-03:2)  1.5069 0.642 0.16122  -0.760    3.774
```

```
## Loc:Rep(LM-03:3)    1.0797 0.442 0.13889 -0.480 2.639
## Loc:Rep(SR-02:1)   -0.7537 0.157 0.00391 -1.309 -0.199
## Loc:Rep(SR-02:2)    0.2251 0.208 0.73791 -0.511 0.961
## Loc:Rep(SR-02:3)    0.5286 0.197 0.09643 -0.168 1.225
## Loc:Rep(SR-03:1)   -0.2670 0.141 0.30849 -0.764 0.230
## Loc:Rep(SR-03:2)    0.2444 0.149 0.43190 -0.284 0.772
## Loc:Rep(SR-03:3)    0.0226 0.118 0.98821 -0.393 0.439
```

Second we treat genotype as random and environment as fixed. Again, the same four components of results are presented.

```
#require(minque)
```

```
res=lmm.jack(Yield~Loc|Loc*Geno+Loc:Rep,data=dat)
```

```
## Estimated variance components
```

```
data.frame(res$Var[[1]])
```

```
##           Estimate      SE    PValue X2.5.LL X97.5.UL
## V(Geno)         26.35 1.221 1.86e-08  22.037  30.66
## V(Loc:Geno)      46.19 3.133 5.25e-07  35.123  57.26
## V(Loc:Rep)        2.09 0.422 3.08e-03   0.606   3.58
## V(e)            37.08 1.816 3.03e-08  30.661  43.49
```

```
## Estiamted proportional variance components
```

```
data.frame(res$PVar[[1]])
```

```
##           Estimate      SE    PValue X2.5.LL X97.5.UL
## V(Geno)/VP       0.2359 0.00972 6.52e-09  0.20161  0.2703
## V(Loc:Geno)/VP   0.4132 0.01798 1.06e-08  0.34966  0.4767
## V(Loc:Rep)/VP    0.0188 0.00371 2.71e-03  0.00567  0.0318
## V(e)/VP          0.3322 0.01863 9.98e-08  0.26634  0.3980
```

```
## Estiamted fixed effects
```

```
data.frame(res$FixedEffect[[1]])
```

```
##           Estimate      SE    PValue X2.5.LL X97.5.UL
## mu          30.67 0.111 0.00e+00  30.27  31.06
## Loc(Ayac)    -6.96 0.295 8.45e-09  -8.00  -5.91
## Loc(Hyo-02)  15.05 0.315 1.54e-11  13.94  16.16
## Loc(LM-02)    3.97 0.254 3.09e-07   3.08   4.87
## Loc(LM-03)   23.16 0.471 1.18e-11  21.50  24.83
## Loc(SR-02)   -15.73 0.173 4.88e-14 -16.34 -15.11
## Loc(SR-03)   -19.51 0.220 5.95e-14 -20.28 -18.73
```

```
## Predicted random effects
```

```
data.frame(res$RandomEffect[[1]])[1:20,]
```

```
##           Pre      SE    PValue X2.5.LL X97.5.UL
## Geno(102.18) -3.6970 0.479 1.17e-04  -5.39  -2.006
## Geno(104.22)  0.5305 0.439 6.68e-01  -1.02   2.081
## Geno(121.31) -0.5156 0.435 6.81e-01  -2.05   1.020
## Geno(141.28)  7.7402 0.710 6.94e-06   5.23  10.248
```



```
## Geno(157.26)  5.4003 0.672 8.52e-05    3.03    7.773
## Geno(163.9)  -7.8613 0.237 4.07e-10   -8.70   -7.024
## Geno(221.19) -6.5686 0.386 1.51e-07   -7.93   -5.204
## Geno(233.11) -1.6947 0.769 2.00e-01   -4.41    1.023
## Geno(235.6)   6.7505 0.423 2.61e-07    5.26    8.243
## Geno(241.2)  -3.6692 0.197 6.67e-08   -4.36   -2.975
## Geno(255.7)  -0.0247 0.486 9.95e-01   -1.74    1.690
## Geno(314.12) -2.1220 0.561 1.72e-02   -4.10   -0.140
## Geno(317.6)   3.9263 0.454 4.73e-05    2.32    5.530
## Geno(319.20)  6.8744 0.697 1.60e-05    4.41    9.335
## Geno(320.16) -3.6731 0.394 2.58e-05   -5.07   -2.280
## Geno(342.15) -3.9187 0.462 5.49e-05   -5.55   -2.288
## Geno(346.2)  -5.8100 0.375 3.39e-07   -7.13   -4.486
## Geno(351.26)  4.6155 0.609 1.36e-04    2.46    6.766
## Geno(364.21)  2.8848 0.512 1.29e-03    1.07    4.695
## Geno(402.7)  -2.6353 0.617 8.29e-03   -4.82   -0.455
```

Third we treat both genotype and environment as random. In this case, no single factor is fixed, so we need put 1 before the vertical '|' and all effects after the bar are random.

```
#require(minique)
```

```
res=lmm.jack(Yield~1|Loc*Geno+Loc:Rep,data=dat)
```

```
## Estimated variance components
```

```
data.frame(res$Var[[1]])
```

```
##           Estimate      SE   PValue  X2.5.LL X97.5.UL
## V(Loc)          288.2 5.125 3.58e-12 270.0757 306.28
## V(Geno)          26.4 1.183 1.40e-08 22.1728 30.53
## V(Loc:Geno)       46.6 2.526 7.44e-08 37.6480 55.49
## V(Loc:Rep)        2.1 0.583 2.25e-02  0.0442  4.16
## V(e)             36.9 1.303 1.65e-09 32.3316 41.54
```

```
## Estiamted proportional variance components
```

```
data.frame(res$PVar[[1]])
```

```
##           Estimate      SE   PValue  X2.5.LL X97.5.UL
## V(Loc)/VP        0.72017 0.00388 0.00e+00 0.70647 0.7339
## V(Geno)/VP        0.06586 0.00296 1.40e-08 0.05542 0.0763
## V(Loc:Geno)/VP    0.11639 0.00607 5.27e-08 0.09494 0.1378
## V(Loc:Rep)/VP     0.00526 0.00145 2.18e-02 0.00014 0.0104
## V(e)/VP           0.09232 0.00346 2.85e-09 0.08009 0.1046
```

```
## Estiamted fixed effects
```

```
data.frame(res$FixedEffect[[1]])
```

```
##      Estimate      SE PValue  X2.5.LL X97.5.UL
## mu      30.7 0.115      0    30.3    31.1
```

```
## Predicted random effects
```

```
data.frame(res$RandomEffect[[1]])[1:20,]
```



##		Pre	SE	PValue	X2.5.LL	X97.5.UL
##	Loc(Ayac)	-6.9449	0.230	9.27e-10	-7.756	-6.133
##	Loc(Hyo-02)	14.9917	0.317	1.71e-11	13.871	16.113
##	Loc(LM-02)	3.9758	0.191	2.51e-08	3.302	4.649
##	Loc(LM-03)	23.0290	0.367	1.33e-12	21.734	24.324
##	Loc(SR-02)	-15.6425	0.128	3.55e-15	-16.093	-15.192
##	Loc(SR-03)	-19.4092	0.228	8.62e-14	-20.214	-18.604
##	Geno(102.18)	-3.6928	0.524	2.39e-04	-5.543	-1.843
##	Geno(104.22)	0.5431	0.278	2.87e-01	-0.440	1.527
##	Geno(121.31)	-0.4763	0.480	7.82e-01	-2.170	1.218
##	Geno(141.28)	7.7111	0.701	6.46e-06	5.234	10.188
##	Geno(157.26)	5.3385	0.485	6.44e-06	3.624	7.053
##	Geno(163.9)	-7.8466	0.400	4.32e-08	-9.260	-6.433
##	Geno(221.19)	-6.4957	0.357	8.45e-08	-7.758	-5.233
##	Geno(233.11)	-1.6985	0.557	5.38e-02	-3.665	0.268
##	Geno(235.6)	6.7656	0.257	3.16e-09	5.859	7.673
##	Geno(241.2)	-3.6796	0.441	6.28e-05	-5.236	-2.123
##	Geno(255.7)	-0.0979	0.246	9.68e-01	-0.968	0.772
##	Geno(314.12)	-2.1698	0.643	3.23e-02	-4.441	0.102
##	Geno(317.6)	3.9710	0.520	1.29e-04	2.133	5.809
##	Geno(319.20)	6.9472	0.591	3.65e-06	4.861	9.033

## Application of the R Package: qgtools

### Features for Package qgtools

This R package includes two major components for quantitative genetics analyses: (1) genetic variance components estimation and genetic effects prediction and (2) crop trial data analysis.

Two linear mixed model approaches: restricted maximum likelihood (REML) and minimum norm quadratic unbiased estimation (MINQUE), are currently available in this R package. In addition, two resampling techniques are included. The package includes the function for model simulation. Several built-in genetic models for bi-parental mating designs are additive-dominance (AD) model, AD model with additive-by-additive interaction effects (ADAA), AD model with maternal effects (ADM), and AD model with cytoplasmic effects (ADC). At least four columns will be needed: environment, female, male, generation (columns 1 to 4). Block in column 5 is optional. Two built-in genetic models for multi-parental mating designs are AD and ADC models. At least six columns will be needed: environment, female 1, male 1, female 2, male 2, generation (columns 1 to 6), where female 1 and male 1 are maternal-grand parents and female 2 and male 2 are paternal grand parents. Block in column 7 is optional.

Functions for Finley and Wilkson regression, variation across environments, mean across environments, and rank across environments are available. In addition, bootstrapping and permutation tests are included as options. Three key variables: genotype, environment, and yield (or any other trait) will be needed. This part data analysis will be detailed in Chapter 7.

## Example 1: Analyses for Bi-parental Breeding Data

In this example, we will demonstrate the use of different approaches for an AD model analysis with MINQUE and REML with and without Jackknife.

### AD model with MINQUE approach

```
library(qgtools)

## Loading required package: Matrix

## Loading required package: MASS

data(cotf12)
head(cotf12)

##   Year Entry Female Male Gen Blk CtYld TlnY LintY Bolls Bsize
## 1     1     1     1     1   0   1  0.87 0.32  28.5  12.1  4.49
## 2     1     2     2     2   0   1  1.14 0.38  34.2  18.6  4.53
## 3     1     3     3     3   0   1  1.14 0.42  37.5   8.0  5.17
## 4     1     4     4     4   0   1  1.28 0.53  47.0  15.2  4.88
## 5     1     5     5     5   0   1  1.33 0.52  46.9  15.1  4.72
## 6     1     6     6     6   0   1  1.72 0.66  59.5  20.0  4.47

cotf12=cotf12[, -2]
Ped=cotf12[, c(1:5)]
Y=as.matrix(cotf12[, 10])
colnames(Y)=colnames(cotf12)[10]

## MINQUE approach without Jackknife
res=ad.mq(Y, Ped)

## Estimated variance components
data.frame(res$Var[[1]])

##           Est      SE  Chi_sq P_value
## V(E)    6.56e-02 0.10117 4.21e-01 0.25829
## V(A)    1.04e-02 0.01101 9.00e-01 0.17142
## V(D)    4.96e-02 0.02049 5.85e+00 0.00777
## V(AE)   6.80e-03 0.00477 2.03e+00 0.07698
## V(DE)  -1.88e-06 0.00902 4.37e-08 0.49992
## V(B)    7.55e-03 0.00668 1.28e+00 0.12916
## V(e)    9.46e-02 0.00871 1.18e+02 0.00000

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Est      SE z_value P_value
## mu 4.78 0.194    24.6      0

## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects
```

```

##           Pre      SE z_value P_value
## E(1)  0.1811 0.1735  1.044  0.2965
## E(2) -0.1811 0.1735 -1.044  0.2965
## A(1) -0.0682 0.0666 -1.025  0.3056
## A(2) -0.0470 0.0676 -0.695  0.4869
## A(3)  0.1184 0.0653  1.813  0.0698
## A(4) -0.0608 0.0665 -0.915  0.3602

## MINQUE approach with Jackknife
res=ad.mq.jack(Y,Ped)

## Estimated Variance components
data.frame(res$Var[[1]])

##           Estimate      SE  PValue  X2.5.LL X97.5.UL
## V(E)   0.06598 0.00487 1.09e-06  0.04878  0.0832
## V(A)   0.01018 0.00211 3.71e-03  0.00274  0.0176
## V(D)   0.05030 0.00449 5.57e-06  0.03443  0.0662
## V(AE)  0.00700 0.00167 9.25e-03  0.00111  0.0129
## V(DE)  0.00126 0.00282 9.61e-01 -0.00871  0.0112
## V(B)   0.00741 0.00168 6.65e-03  0.00149  0.0133
## V(e)   0.09484 0.00371 4.09e-09  0.08175  0.1079

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##           Estimate      SE  PValue  X2.5.LL X97.5.UL
## V(E)/VP  0.27832 0.01827 3.95e-07  0.21379  0.3429
## V(A)/VP  0.04292 0.00849 2.73e-03  0.01295  0.0729
## V(D)/VP  0.21223 0.01806 3.69e-06  0.14843  0.2760
## V(AE)/VP 0.02958 0.00721 1.06e-02  0.00412  0.0550
## V(DE)/VP 0.00531 0.01189 9.61e-01 -0.03671  0.0473
## V(B)/VP  0.03132 0.00732 8.19e-03  0.00546  0.0572
## V(e)/VP  0.40032 0.01745 1.08e-08  0.33868  0.4620

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##           Estimate      SE  PValue  X2.5.LL X97.5.UL
## mu          4.78 0.00788      0    4.75    4.81

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##           Pre      SE  PValue  X2.5.LL X97.5.UL
## E(1)  0.1815 0.00663 2.26e-09  0.1581  0.2050
## E(2) -0.1815 0.00663 2.26e-09 -0.2050 -0.1581
## A(1) -0.0671 0.01342 2.96e-03 -0.1145 -0.0197
## A(2) -0.0454 0.00955 4.15e-03 -0.0791 -0.0117
## A(3)  0.1158 0.01288 3.43e-05  0.0704  0.1613
## A(4) -0.0598 0.00683 4.27e-05 -0.0840 -0.0357

```

## AD model with REML approach

```
library(qgtools)
data(cotf12)
head(cotf12)

##   Year Entry Female Male Gen Blk CtYld TlnY LintY Bolls Bsize
## 1     1     1     1     1   0   1  0.87 0.32  28.5  12.1  4.49
## 2     1     2     2     2   0   1  1.14 0.38  34.2  18.6  4.53
## 3     1     3     3     3   0   1  1.14 0.42  37.5   8.0  5.17
## 4     1     4     4     4   0   1  1.28 0.53  47.0  15.2  4.88
## 5     1     5     5     5   0   1  1.33 0.52  46.9  15.1  4.72
## 6     1     6     6     6   0   1  1.72 0.66  59.5  20.0  4.47

cotf12=cotf12[,-2]
Ped=cotf12[,c(1:5)]
Y=as.matrix(cotf12[,10])
colnames(Y)=colnames(cotf12)[10]

## REML approach without Jackknife
res=ad.reml(Y,Ped)
## Estimated variance components
data.frame(res$Var[[1]])

##           Est          SE    Chi_sq P_value
## V(E)    6.24e-02 0.09699 4.14e-01  0.2599
## V(A)    1.19e-02 0.01171 1.02e+00  0.1558
## V(D)    4.81e-02 0.02002 5.76e+00  0.0082
## V(AE)    7.45e-03 0.00507 2.16e+00  0.0708
## V(DE)   -1.30e-06 0.00901 2.09e-08  0.4999
## V(B)    7.55e-03 0.00668 1.28e+00  0.1291
## V(e)    9.46e-02 0.00870 1.18e+02  0.0000

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Est      SE z_value P_value
## mu 4.78 0.194    24.7      0

## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects

##           Pre          SE z_value P_value
## E(1)  0.1767 0.1686    1.048  0.2946
## E(2) -0.1767 0.1686   -1.048  0.2946
## A(1) -0.0701 0.0731   -0.960  0.3373
## A(2) -0.0490 0.0741   -0.662  0.5082
## A(3)  0.1293 0.0718    1.800  0.0719
## A(4) -0.0628 0.0730   -0.860  0.3899

## MINQUE approach with Jackknife
res=ad.reml.jack(Y,Ped)
```

```
## Estimated Variance components
data.frame(res$Var[[1]])

##      Estimate      SE  PValue  X2.5.LL X97.5.UL
## V(E)  0.06332 0.002975 2.10e-08  0.052812  0.0738
## V(A)  0.01183 0.002708 7.18e-03  0.002265  0.0214
## V(D)  0.04818 0.004166 4.22e-06  0.033464  0.0629
## V(AE) 0.00742 0.001900 1.43e-02  0.000703  0.0141
## V(DE) 0.00186 0.003372 9.40e-01 -0.010048  0.0138
## V(B)  0.00742 0.000952 1.10e-04  0.004054  0.0108
## V(e)  0.09473 0.002202 3.95e-11  0.086956  0.1025

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##      Estimate      SE  PValue X2.5.LL X97.5.UL
## V(E)/VP  0.26972 0.01169 1.03e-08  0.2284  0.3110
## V(A)/VP  0.05039 0.01143 6.76e-03  0.0100  0.0907
## V(D)/VP  0.20518 0.01676 2.60e-06  0.1460  0.2644
## V(AE)/VP 0.03162 0.00822 1.56e-02  0.0026  0.0606
## V(DE)/VP 0.00793 0.01437 9.41e-01 -0.0428  0.0587
## V(B)/VP  0.03159 0.00393 8.57e-05  0.0177  0.0455
## V(e)/VP  0.40359 0.00979 5.82e-11  0.3690  0.4382

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## mu      4.78 0.00685      0    4.75    4.8

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##      Pre      SE  PValue X2.5.LL X97.5.UL
## E(1)  0.1779 0.00421 4.66e-11  0.1630  0.1928
## E(2) -0.1779 0.00421 4.66e-11 -0.1928 -0.1630
## A(1) -0.0689 0.00922 1.52e-04 -0.1014 -0.0363
## A(2) -0.0483 0.00952 2.68e-03 -0.0819 -0.0146
## A(3)  0.1274 0.01590 8.79e-05  0.0712  0.1835
## A(4) -0.0622 0.01049 8.86e-04 -0.0992 -0.0251
```

The above R scripts can be used for other genetic models such as ADC, ADAA, and ADM with slight changes. For example, we will run similar data analysis for the ADAA model with the above data.

### ADAA model with MINQUE approach

```
library(qgtools)
data(cotf12)
head(cotf12)
```

```
##   Year Entry Female Male Gen Blk CtYld TlnY LintY Bolls Bsize
## 1    1     1      1    1   0    1  0.87 0.32  28.5  12.1  4.49
## 2    1     2      2    2   0    1  1.14 0.38  34.2  18.6  4.53
## 3    1     3      3    3   0    1  1.14 0.42  37.5   8.0  5.17
## 4    1     4      4    4   0    1  1.28 0.53  47.0  15.2  4.88
## 5    1     5      5    5   0    1  1.33 0.52  46.9  15.1  4.72
## 6    1     6      6    6   0    1  1.72 0.66  59.5  20.0  4.47
```

```
cotf12=cotf12[, -2]
Ped=cotf12[, c(1:5)]
Y=as.matrix(cotf12[, 10])
colnames(Y)=colnames(cotf12)[10]
```

```
## MINQUE approach without Jackknife
res=adaa.mq(Y, Ped)
```

```
## Estimated variance components
data.frame(res$Var[[1]])
```

```
##           Est      SE   Chi_sq P_value
## V(E)      1.35e-01 0.21533 3.93e-01 0.26535
## V(A)     -4.54e-05 0.03558 1.63e-06 0.49949
## V(D)      1.39e-01 0.06188 5.06e+00 0.01221
## V(AA)     4.81e-03 0.00966 2.47e-01 0.30952
## V(AE)    -4.54e-05 0.03174 2.05e-06 0.49943
## V(DE)    -2.27e-05 0.02887 6.18e-07 0.49969
## V(AAE)    2.45e-02 0.00977 6.30e+00 0.00605
## V(B)      7.78e-03 0.00668 1.36e+00 0.12217
## V(e)      8.34e-02 0.00806 1.07e+02 0.00000
```

```
## Estimated fixed effects
data.frame(res$FixedEffect[[1]])
```

```
##      Est      SE z_value P_value
## mu 4.76 0.264    18.1      0
```

```
## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects
```

```
##      Pre      SE z_value P_value
## E(1) 0.26 0.244800    1.06    0.289
## E(2) -0.26 0.244800   -1.06    0.289
## A(1) 0.00 0.000180    0.00    1.000
## A(2) 0.00 0.000187    0.00    1.000
## A(3) 0.00 0.000172    0.00    1.000
## A(4) 0.00 0.000180    0.00    1.000
```

```
## MINQUE approach with Jackknife
res=adaa.mq.jack(Y, Ped)
```

```
## Estimated Variance components
```

```
data.frame(res$Var[[1]])
```

##		Estimate	SE	PValue	X2.5.LL	X97.5.UL
##	V(E)	0.134188	0.01044	1.71e-06	0.09731	0.17107
##	V(A)	0.000365	0.00077	9.56e-01	-0.00235	0.00308
##	V(D)	0.144197	0.02134	3.32e-04	0.06882	0.21957
##	V(AA)	0.001860	0.00292	9.19e-01	-0.00845	0.01217
##	V(AE)	0.000000	0.00000	1.00e+00	0.00000	0.00000
##	V(DE)	0.002683	0.00681	9.69e-01	-0.02139	0.02675
##	V(AAE)	0.025424	0.00970	1.06e-01	-0.00885	0.05970
##	V(B)	0.007845	0.00170	5.03e-03	0.00184	0.01385
##	V(e)	0.083241	0.00307	2.43e-09	0.07240	0.09408

```
## Estimated proportional variance components
```

```
data.frame(res$PVar[[1]])
```

##		Estimate	SE	PValue	X2.5.LL	X97.5.UL
##	V(E)/VP	0.335820	0.02109	2.69e-07	0.261311	0.41033
##	V(A)/VP	0.000915	0.00193	9.56e-01	-0.005909	0.00774
##	V(D)/VP	0.359906	0.03889	2.72e-05	0.222522	0.49729
##	V(AA)/VP	0.004879	0.00758	9.18e-01	-0.021909	0.03167
##	V(AE)/VP	0.000000	0.00000	1.00e+00	0.000000	0.00000
##	V(DE)/VP	0.006601	0.01685	9.69e-01	-0.052911	0.06611
##	V(AAE)/VP	0.063123	0.02225	7.55e-02	-0.015485	0.14173
##	V(B)/VP	0.019872	0.00537	1.95e-02	0.000898	0.03885
##	V(e)/VP	0.208884	0.01480	7.64e-07	0.156612	0.26116

```
## Estimated fixed effects
```

```
data.frame(res$FixedEffect[[1]])
```

##		Estimate	SE	PValue	X2.5.LL	X97.5.UL
##	mu	4.76	0.00696	0	4.74	4.79

```
## Predicted random effects
```

```
head(data.frame(res$RandomEffect[[1]]))
```

##		Pre	SE	PValue	X2.5.LL	X97.5.UL
##	E(1)	0.25884	0.01023	4.51e-09	0.2227	0.2950
##	E(2)	-0.25884	0.01023	4.51e-09	-0.2950	-0.2227
##	A(1)	-0.00367	0.00788	9.58e-01	-0.0315	0.0242
##	A(2)	-0.00055	0.00149	9.72e-01	-0.0058	0.0047
##	A(3)	0.00955	0.02014	9.56e-01	-0.0616	0.0807
##	A(4)	-0.00646	0.01371	9.57e-01	-0.0549	0.0420

## ADAA model with REML approach

```
library(qgtools)
```

```
data(cotf12)
```

```
head(cotf12)
```



```
##   Year Entry Female Male Gen Blk CtYld TlnY LintY Bolls Bsize
## 1    1     1      1    1   0    1  0.87 0.32  28.5  12.1  4.49
## 2    1     2      2    2   0    1  1.14 0.38  34.2  18.6  4.53
## 3    1     3      3    3   0    1  1.14 0.42  37.5   8.0  5.17
## 4    1     4      4    4   0    1  1.28 0.53  47.0  15.2  4.88
## 5    1     5      5    5   0    1  1.33 0.52  46.9  15.1  4.72
## 6    1     6      6    6   0    1  1.72 0.66  59.5  20.0  4.47
```

```
cotf12=cotf12[, -2]
Ped=cotf12[, c(1:5)]
Y=as.matrix(cotf12[, 10])
colnames(Y)=colnames(cotf12)[10]
```

```
## REML approach without Jackknife
res=adaa.reml(Y, Ped)
## Estimated variance components
data.frame(res$Var[[1]])
```

```
##           Est      SE   Chi_sq P_value
## V(E)      1.20e-01 0.19011 3.96e-01 0.26450
## V(A)      2.78e-03 0.02748 1.03e-02 0.45964
## V(D)      7.96e-02 0.04795 2.76e+00 0.04837
## V(AA)     5.43e-03 0.00731 5.52e-01 0.22868
## V(AE)    -4.54e-05 0.02657 2.92e-06 0.49932
## V(DE)    -2.27e-05 0.02812 6.51e-07 0.49968
## V(AAE)    2.01e-02 0.00847 5.60e+00 0.00897
## V(B)      7.70e-03 0.00668 1.33e+00 0.12439
## V(e)      8.70e-02 0.00830 1.10e+02 0.00000
```

```
## Estimated fixed effects
data.frame(res$FixedEffect[[1]])
```

```
##      Est      SE z_value P_value
## mu 4.77 0.259    18.4      0
```

```
## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects
```

```
##           Pre      SE z_value P_value
## E(1)  0.24462 0.2310   1.059 2.90e-01
## E(2) -0.24462 0.2310  -1.059 2.90e-01
## A(1) -0.02941 0.0126  -2.336 1.95e-02
## A(2) -0.00968 0.0130  -0.745 4.56e-01
## A(3)  0.06236 0.0121   5.152 2.58e-07
## A(4) -0.03561 0.0126  -2.831 4.64e-03
```

```
## MINQUE approach with Jackknife
res=adaa.reml.jack(Y, Ped)
```

```
## Estimated Variance components
data.frame(res$Var[[1]])
```

```
##           Estimate      SE   PValue   X2.5.LL X97.5.UL
## V(E)      0.11766 0.01283 2.92e-05  0.072350  0.1630
## V(A)      0.00776 0.00742 7.56e-01 -0.018459  0.0340
## V(D)      0.08083 0.01660 3.54e-03  0.022177  0.1395
## V(AA)     0.00550 0.00925 9.30e-01 -0.027183  0.0382
## V(AE)     0.00000 0.00000 1.00e+00  0.000000  0.0000
## V(DE)     0.00000 0.00000 1.00e+00  0.000000  0.0000
## V(AAE)    0.02040 0.00758 9.46e-02 -0.006360  0.0472
## V(B)      0.00768 0.00201 1.64e-02  0.000565  0.0148
## V(e)      0.08730 0.00416 2.37e-08  0.072606  0.1020

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##           Estimate      SE   PValue   X2.5.LL X97.5.UL
## V(E)/VP    0.3595 0.02219 2.31e-07  0.281152  0.4379
## V(A)/VP    0.0234 0.02262 7.61e-01 -0.056501  0.1033
## V(D)/VP    0.2454 0.03688 3.74e-04  0.115079  0.3756
## V(AA)/VP   0.0179 0.03088 9.34e-01 -0.091171  0.1270
## V(AE)/VP   0.0000 0.00000 1.00e+00  0.000000  0.0000
## V(DE)/VP   0.0000 0.00000 1.00e+00  0.000000  0.0000
## V(AAE)/VP  0.0613 0.01796 3.03e-02 -0.002099  0.1248
## V(B)/VP    0.0237 0.00683 2.80e-02 -0.000451  0.0478
## V(e)/VP    0.2688 0.02754 1.75e-05  0.171524  0.3661

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## mu      4.77 0.0114      0    4.73    4.81

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##      Pre      SE   PValue X2.5.LL X97.5.UL
## E(1) 0.2422 0.0132 7.88e-08  0.1955  0.2889
## E(2) -0.2422 0.0132 7.88e-08 -0.2889 -0.1955
## A(1) -0.0401 0.0317 6.37e-01 -0.1519  0.0718
## A(2) -0.0135 0.0108 6.45e-01 -0.0517  0.0247
## A(3)  0.0848 0.0648 6.12e-01 -0.1442  0.3139
## A(4) -0.0466 0.0362 6.23e-01 -0.1744  0.0812
```

## Example 2: Analyses for multi-parental Breeding Data

In this example, we will demonstrate the use of different approaches for an AD model analysis with MINQUE and REML with and without Jackknife.

### AD model with MINQUE approach

```
library(qgtools)
data(wheat)
head(wheat)
```

```

##   Env P1 P2 P3 P4 Gen REP Average
## 1   1  1  2  3  3   2   1   4.89
## 2   1  1  2  3  3   2   1   5.80
## 3   1  1  2  3  3   2   1   4.11
## 4   1  1  2  3  3   2   1   5.44
## 5   1  1  2  3  3   2   1   4.70
## 6   1  1  2  4  4   2   1   5.60

Ped=wheat[,c(1:6)]
Y=as.matrix(wheat[,8])
colnames(Y)=colnames(wheat)[8]

## MINQUE approach without Jackknife
res=ad4.mq(Y,Ped)

## Estimated variance components
data.frame(res$Var[[1]])      ## Variance components

##      Est      SE Chi_sq P_value
## V(A) 0.660 0.3040   4.71 0.0150
## V(D) 0.349 0.2462   2.01 0.0781
## V(e) 0.756 0.0388 380.04 0.0000

## Estimated fixed effects
data.frame(res$FixedEffect[[1]]) ## Fixed effects

##      Est      SE z_value P_value
## mu 6.46 0.316   20.5      0

## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects

##      Pre      SE z_value P_value
## A(1) -0.7231 0.738 -0.9801 0.327
## A(2) -0.8765 0.747 -1.1726 0.241
## A(3) -0.7605 0.752 -1.0114 0.312
## A(4) -0.0258 0.748 -0.0346 0.972
## A(5) -0.0500 0.753 -0.0664 0.947
## A(6) -0.0985 0.747 -0.1318 0.895

## MINQUE approach with Jackknife
res=ad4.mq.jack(Y,Ped)

## Estimated Variance components
data.frame(res$Var[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## V(A)    0.633 0.03927 2.41e-07 0.4944 0.772
## V(D)    0.354 0.10969 4.09e-02 -0.0338 0.741
## V(e)    0.756 0.00999 2.50e-13 0.7207 0.791

```

```
## Estimated proportional variance components
data.frame(res$PVar[[1]])

##           Estimate      SE   PValue X2.5.LL X97.5.UL
## V(A)/VP      0.364 0.0300 2.81e-06  0.2583   0.470
## V(D)/VP      0.200 0.0524 1.61e-02  0.0153   0.385
## V(e)/VP      0.435 0.0273 2.68e-07  0.3388   0.532

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## mu        6.46 0.0186      0      6.4   6.53

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##           Pre      SE   PValue X2.5.LL X97.5.UL
## A(1) -0.7110 0.0481 5.11e-07  -0.881  -0.5412
## A(2) -0.8603 0.0381 1.23e-08  -0.995  -0.7258
## A(3) -0.7368 0.0548 1.16e-06  -0.930  -0.5433
## A(4) -0.0385 0.0452 8.46e-01  -0.198   0.1212
## A(5) -0.0318 0.0128 1.31e-01  -0.077   0.0134
## A(6) -0.0982 0.0337 6.66e-02  -0.217   0.0208
```

## AD model with REML approach

```
library(qgtools)
data(wheat)
head(wheat)

##      Env P1 P2 P3 P4 Gen REP Average
## 1      1  1  2  3  3  2   1    4.89
## 2      1  1  2  3  3  2   1    5.80
## 3      1  1  2  3  3  2   1    4.11
## 4      1  1  2  3  3  2   1    5.44
## 5      1  1  2  3  3  2   1    4.70
## 6      1  1  2  4  4  2   1    5.60

Ped=wheat[,c(1:6)]
Y=as.matrix(wheat[,8])
colnames(Y)=colnames(wheat)[8]

## REML approach without Jackknife
res=ad4.mq(Y,Ped)

## Estimated variance components
data.frame(res$Var[[1]])      ## Variance components

##           Est      SE Chi_sq P_value
## V(A) 0.660 0.3040   4.71  0.0150
```

```

## V(D) 0.349 0.2462 2.01 0.0781
## V(e) 0.756 0.0388 380.04 0.0000

## Estimated fixed effects
data.frame(res$FixedEffect[[1]]) ## Fixed effects

##      Est      SE z_value P_value
## mu 6.46 0.316 20.5 0

## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects

##      Pre      SE z_value P_value
## A(1) -0.7231 0.738 -0.9801 0.327
## A(2) -0.8765 0.747 -1.1726 0.241
## A(3) -0.7605 0.752 -1.0114 0.312
## A(4) -0.0258 0.748 -0.0346 0.972
## A(5) -0.0500 0.753 -0.0664 0.947
## A(6) -0.0985 0.747 -0.1318 0.895

## REML approach with Jackknife
res=ad4.mq.jack(Y,Ped)

## Estimated Variance components
data.frame(res$Var[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## V(A) 0.638 0.0348 7.92e-08 0.5146 0.761
## V(D) 0.343 0.1063 4.08e-02 -0.0325 0.719
## V(e) 0.756 0.0197 1.08e-10 0.6865 0.825

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## V(A)/VP 0.369 0.0362 1.22e-05 0.240991 0.497
## V(D)/VP 0.195 0.0550 2.51e-02 0.000319 0.389
## V(e)/VP 0.437 0.0236 7.20e-08 0.353239 0.520

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## mu 6.46 0.0273 0 6.37 6.56

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##      Pre      SE PValue X2.5.LL X97.5.UL
## A(1) -0.7106 0.0436 2.20e-07 -0.865 -0.5565
## A(2) -0.8607 0.0429 3.53e-08 -1.012 -0.7092
## A(3) -0.7389 0.0495 4.70e-07 -0.914 -0.5641

```

```
## A(4) -0.0398 0.0628 9.20e-01 -0.262 0.1820
## A(5) -0.0313 0.0605 9.48e-01 -0.245 0.1825
## A(6) -0.0979 0.0422 1.68e-01 -0.247 0.0512
```

The above R scripts can be used for the ADC model with slight changes. For example,

```
library(qgtools)
data(wheat)
head(wheat)

##   Env P1 P2 P3 P4 Gen REP Average
## 1   1  1  2  3  3  2   1   4.89
## 2   1  1  2  3  3  2   1   5.80
## 3   1  1  2  3  3  2   1   4.11
## 4   1  1  2  3  3  2   1   5.44
## 5   1  1  2  3  3  2   1   4.70
## 6   1  1  2  4  4  2   1   5.60

Ped=wheat[,c(1:6)]
Y=as.matrix(wheat[,8])
colnames(Y)=colnames(wheat)[8]

## REML approach without Jackknife
res=adc4.mq(Y,Ped)

## Estimated variance components
data.frame(res$Var[[1]]) ## Variance components

##           Est      SE   Chi_sq P_value
## V(C)  0.541308 0.3011 3.23e+00 0.0361
## V(A)  0.601508 0.2638 5.20e+00 0.0113
## V(D) -0.000164 0.2335 4.95e-07 0.4997
## V(e)  0.747257 0.0384 3.79e+02 0.0000

## Estimated fixed effects
data.frame(res$FixedEffect[[1]]) ## Fixed effects

##      Est   SE z_value P_value
## mu 6.78 0.35   19.4      0

## Predicted random effects
head(data.frame(res$RandomEffect[[1]])) ## Random Effects

##      Pre   SE z_value P_value
## C1 -0.556 0.694 -0.802 0.4225
## C2  0.728 0.595  1.222 0.2216
## C3  1.052 0.605  1.738 0.0822
## C4  0.371 0.626  0.592 0.5537
## C5  1.195 0.610  1.960 0.0500
## C6  0.363 0.618  0.588 0.5568
```

```
## REML approach with Jackknife
res=adc4.mq.jack(Y,Ped)

## Estimated Variance components
data.frame(res$Var[[1]])

##      Estimate      SE   PValue X2.5.LL X97.5.UL
## V(C)    0.535 0.1255 8.40e-03  0.0915  0.978
## V(A)    0.593 0.0414 6.84e-07  0.4462  0.739
## V(D)    0.000 0.0000 1.00e+00  0.0000  0.000
## V(e)    0.747 0.0169 3.15e-11  0.6870  0.807

## Estimated proportional variance components
data.frame(res$PVar[[1]])

##      Estimate      SE   PValue X2.5.LL X97.5.UL
## V(C)/VP    0.283 0.0446 5.42e-04  0.125  0.440
## V(A)/VP    0.317 0.0219 6.09e-07  0.240  0.394
## V(D)/VP    0.000 0.0000 1.00e+00  0.000  0.000
## V(e)/VP    0.400 0.0280 6.77e-07  0.302  0.499

## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate      SE PValue X2.5.LL X97.5.UL
## mu        6.78 0.0425      0   6.62   6.93

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##      Pre      SE   PValue X2.5.LL X97.5.UL
## C1 -0.548 0.0664 6.86e-05 -0.7828 -0.314
## C2  0.708 0.0618 4.55e-06  0.4901  0.927
## C3  1.043 0.0945 6.29e-06  0.7088  1.377
## C4  0.322 0.1394 1.71e-01 -0.1705  0.814
## C5  1.186 0.3574 3.53e-02 -0.0765  2.448
## C6  0.321 0.0970 3.55e-02 -0.0211  0.664
```

## AD model with Rectangular Row-Column Blocking

This R package also provide a function integrating AD genetic model and rectangular row-column design to improve genetic data analysis.

```
library(qgtools)
data(adrcdat)
dat=adrcdat[which(adrcdat$Env==1),]
Ped=dat[,c(1,4,5,6)]
Y=as.matrix(dat[,8])
colnames(Y)=colnames(dat)[8]

Row=dat$Row
Col=dat$Column
```



```
##run AD model without jackknifing with row and column blocking
res=adrc.mq(Y,Ped,Row=Row,Col=Col)
```

```
## Estimated variance components
```

```
data.frame(res$Var[[1]]) ## Variance components
```

```
##      Est      SE   Chi_sq P_value
## V(A)   2740.693  2141 1.64e+00 1.00e-01
## V(D)  28102.200 15621 3.24e+00 3.60e-02
## V(Row)   -0.907   655 1.92e-06 4.99e-01
## V(Col)   877.445  1119 6.15e-01 2.16e-01
## V(e)   12210.456  3276 1.39e+01 9.69e-05
```

```
## Estimated fixed effects
```

```
data.frame(res$FixedEffect[[1]]) ## Fixed effects
```

```
##      Est      SE z_value P_value
## mu 989 23.9    41.3      0
```

```
## Predicted random effects
```

```
head(data.frame(res$RandomEffect[[1]])) ## Random Effects
```

```
##      Pre      SE z_value P_value
## A(3)   21.9 33.8    0.647 0.5178
## A(4)  -75.0 26.9   -2.788 0.0053
## A(9)   57.4 33.4    1.719 0.0856
## A(10) -23.0 33.6   -0.685 0.4937
## A(19) -38.5 26.7   -1.441 0.1495
## A(33) -28.6 29.0   -0.985 0.3244
```

```
## REML approach with Jackknife
```

```
res=adrc.mq.jack(Y,Ped,Row=Row,Col=Col)
```

```
## Estimated Variance components
```

```
data.frame(res$Var[[1]])
```

```
##      Estimate      SE PValue X2.5.LL X97.5.UL
## V(A)       2886   1743 0.42121  -3270    9041
## V(D)      29024  11048 0.10479  -9999   68047
## V(Row)        0      0 1.00000      0      0
## V(Col)      1026    564 0.34278   -966    3019
## V(e)      12056   2856 0.00891   1967   22144
```

```
## Estimated proportional variance components
```

```
data.frame(res$PVar[[1]])
```

```
##      Estimate      SE PValue X2.5.LL X97.5.UL
## V(A)/VP    0.0696 0.0452 0.48340 -0.0902  0.2294
## V(D)/VP    0.6274 0.1280 0.00338  0.1752  1.0797
## V(Row)/VP   0.0000 0.0000 1.00000  0.0000  0.0000
## V(Col)/VP   0.0224 0.0119 0.31203 -0.0195  0.0644
## V(e)/VP    0.2805 0.0940 0.05964 -0.0514  0.6125
```

```
## Estimated fixed effects
data.frame(res$FixedEffect[[1]])

##      Estimate    SE PValue X2.5.LL X97.5.UL
## mu          989 3.79      0      975     1002

## Predicted random effects
head(data.frame(res$RandomEffect[[1]]))

##      Pre    SE PValue X2.5.LL X97.5.UL
## A(3)   21.5 12.5  0.389   -22.7     65.8
## A(4)  -69.1 31.6  0.205  -180.8     42.5
## A(9)   52.1 25.5  0.252   -37.9    142.2
## A(10) -21.4 13.8  0.477   -70.2     27.4
## A(19) -35.6 15.9  0.187   -91.7     20.4
## A(33) -27.4 21.7  0.636  -104.0     49.1
```