GBLUP AND QTL

IN ASREML AND ECHIDNA

ARTHUR R GILMOUR PHD
ECHIDNAMMS.ORG
24 FEBRUARY 2023

GBLUP and QTL in ASReml and Echidna

Arthur R Gilmour

This report is adapted from a document prepared by the author in 2015.

GBLUP is the prediction of breeding values from a Genomic Relationship Matrix (GRM), typically formed from marker data pertaining to many more SNPs (single nucleotide polymorphisms) than genotypes defining the matrix since each SNP represents a contrasts among the genotypes.

QTL (Quantitative trait loci) are identified from markers of apparent large effect assuming the markers are in linkage disequilibrium with a gene of large effect.

Contents

Preface

This is one of a series of technical reports describing facilities in ASReml-SA 4.3 and/or Echidna 1.67. Some of these facilities are not included in the official software documentation.

Those interested in trialling these facilities are welcome to do so and give feedback to the author.

ASReml is commercial software owned and distributed by VSN International. (www.vsni.co.uk). **ASReml** is available as a standalone program (**ASReml-SA**) and through an R implementation (**ASREML-R**). These notes specifically relate to the stand-alone program although most of the models can also be fitted in **ASReml-R**. **ASReml** was developed by the author with strong support from Dr Brian Cullis and Dr Robin Thompson. It was sold to VSN. In 2012.

Echidna is owned by the author. By agreement with VSNi and to the extent that it duplicates many facilities of ASReml, potential users are requested not to use it for commercial purposes unless they have a current ASReml license from VSN. To access Echidna, register at www.EchidnaMMS.org.

These notes cover a variety of statistical methods related to marker based genetic variation that have been implemented in **ASReml** and/or **Echidna** and it is hoped will provide some help to students in learning what has been tried. The viewpoint of the author is that the glory and power of Almighty God are clearly scene in the things he has made, and are revealed to us as we investigate the nature of His creation. We are therefore without excuse (Romans 1:18-32)

> IT IS THE GLORY OF GOD TO CONCEAL A MATTER,
> BUT THE GLORY OF KINGS IS TO SEARCH OUT A MATTER.
> SOLOMON, PROVERBS 25:2

Introduction

The aim of this document is to show ways in which **ASReml** can be used to fit models common in genomic analysis. It assumes readers have a basic familiarity with the models and does not attempt to describe or justify use of any particular model. References to ASReml also apply to **Echidna** except where a difference is noted.

We consider the common genetic model for diploid species where the genome is divided into chromosomes and along the chromosomes are snippets of genetic material which are unique to a particular location and come in 2 or more forms which can be read and are distinct, known as markers.

There are several types of markers, which do not concern us here except to the extent that some have just 2 forms while others have more than two forms. The forms can be called 'alleles'.

Some marker methods depend on mapping the markers on the chromosomes and a common distance measure is called the Morgan. This is a distance measure based on the probability of crossover during meiosis, rather than being based on number of base pairs. **ASReml** does not include any procedures to map markers on chromosomes, although if maps are available, they may be used.

There has been a huge investment in genomics especially since 2000. The initial expectation was that by reading the genome, we would have all we needed for genetic improvement. But everything is not controlled by Quantitative trait Loci (QTL) although a few important QTL have been identified. Many QTL though affect multiple traits but more often, we cannot identify what drives genetic variation.

These notes cover a variety of statistical methods related to marker based genetic variation that have been implemented in ASReml and/or Echidna and it is hoped will provide some help to students in learning what has been tried. Several datasets are used to demonstrate the methods.

Mouse Data

Horvat and Medrano (1995).

Nassau Tree data

This tree data was shared with the author by Patricio Munoz (Resende et al 2012). This dataset has 6795 records providing growth data (HT and DBH) on clones of 860 genotypes representing 71 families derived 50 parents. Genotypic information is available for 923 trees (4854 SNP markers); 3 with data do not have marker information.

The data is available in the R library ASRgenomics:
? ASRgenomics::pheno.pine
? ASRgenomics::geno.pine926
? ASRgenomics::ped.pine

Simulated QTLMAS data

The QTLMAS dataset was simulated by Szydlowski, M. & Paczynska, P. (2011) (\http://www.biomedcentral.com/1753-6561/5/S3/S3) came as four files:

- SNP genotypes are in file genotype.mkr which has 3,227 lines and 10,032 columns (fields). The header (first) line specifies names of 10,031 SNPs and each line below includes number of major alleles (0/1/2) for each SNP for that given individual.
- Phenotypes are in file phenotype.txt, including Identity and phenotypic values for 2,326 animals, comprising 5 generations. Note that 900 genotypes do not have phenotype data and the method is evaluated by comparing the BLUPs for these individuals with their 'true' values.
- True breeding values are in file trueBreedingValue.txt, including Identity and true breeding values for 3,226 individuals.
- Pedigree and gender (M indicates male and F female) for 3,226 individuals are in the file pedigree.txt .

Single Marker QTL analysis

One of the older technologies is 'Interval Mapping'. The standard technique basically involved successively regressing the response against each of the markers and noting the largest effect. If a map was available, they would be placed in map order. Technically, interval mapping allows imputing a pseudo-marker from flanking markers, assuming knowledge of which markers are neighbours and how far apart they are, and regressing many such markers to create a profile.

Consider some mouse data published by Horvat and Medrano (1995). There is marker data on 190 mice and liveweight gain on 189 of them. The data file looks like

```
Mouse,D10MIT31,D10MIT42,IGF1,D10MIT9,D10MIT10,D10MIT41,D10MIT12,D10NDS2,D10MIT14,gain
1,1,1,1,1,1,1,1,1,1,12.1
2,1,0,0,0,0,0,0,0,0,15.6
3,1,1,1,1,1,1,1,1,2,14
4,1,1,1,1,1,1,1,1,2,14.6
5,0,0,0,0,0,0,0,0,0,13.5
6,2,1,1,1,1,1,1,1,1,13.2
7,1,1,1,1,1,1,1,1,1,17.3
...
184,1,1,0,0,0,0,0,0,0,10.3
185,2,2,2,2,2,2,2,2,2,11.2
186,1,1,1,1,1,1,1,1,1,16
187,2,2,2,2,2,2,2,2,2,19.2
188,2,2,2,2,2,2,2,2,2,20.8
189,0,1,1,1,1,1,1,1,1,13.3
190,0,0,0,0,0,0,0,0,0,11.8
```

The marker positions are given as 0. 9.1 13.3 14.6 15.9 17.5 20.8 23 31.3.

The **ASReml** input code (apart from the comments which show the original form of the data) to read this data would be:

```
Single QTL search: Data from Horvat and Medrano, 1995. Genetics 139:1737-1748
Mouse
D10MIT31 !-1 # 9.1 Aa Aa Aa Aa aa AA Aa Aa aa Aa Aa Aa Aa AA aa Aa aa Aa Aa
D10MIT42 !-1 # 4.2 Aa aa Aa Aa aa Aa Aa Aa aa Aa Aa Aa aa AA aa Aa Aa aa Aa
IGF1     !-1 # 1.3 Aa aa Aa Aa aa Aa Aa Aa aa Aa Aa Aa aa AA aa Aa Aa aa Aa
D10MIT9  !-1 # 1.3 Aa aa Aa Aa aa Aa Aa Aa aa Aa AA Aa aa AA Aa Aa Aa aa Aa
D10MIT10 !-1 # 1.6 Aa aa Aa Aa aa Aa Aa Aa aa Aa AA Aa aa Aa Aa Aa Aa aa Aa
D10MIT41 !-1 # 3.3 Aa aa Aa Aa aa Aa Aa Aa aa Aa AA Aa aa aa Aa Aa Aa aa Aa
D10MIT12 !-1 # 2.2 Aa aa Aa Aa aa Aa Aa Aa aa Aa AA aa aa aa Aa Aa Aa aa Aa
D10NDS2  !-1 # 8.3 Aa aa Aa Aa aa Aa Aa Aa aa Aa AA aa aa aa Aa Aa Aa Aa aa
D10MIT14 !-1 # 0.  Aa aa AA AA aa Aa Aa Aa aa Aa Aa aa aa aa Aa aa Aa Aa aa
```

```
 GAIN
HM.dat !skip 1
```

and we could fit the markers 1 by 1 as fixed effects by appending

```
!CYCLE !SAMEDATA D10MIT31 D10MIT42 IGF1 D10MIT9 D10MIT10 D10MIT41,
          D10MIT12 D10NDS2 D10MIT14
GAIN ~ mu  I
```

although I prefer to fit them as random effects and compare LogL values
by using

```
!CYCLE !SAMEDATA D10MIT31 D10MIT42 IGF1 D10MIT9 D10MIT10 D10MIT41,
       D10MIT12 D10NDS2 D10MIT14
GAIN ~ mu !R  I
```

N.B. The !SAMEDATA qualifier, in **ASReml** 4 but not yet in **Echidna**, speeds processing by only reading the data in ONCE rather than repeatedly for each cycle.

The use of !CYCLE results in a single output file containing the 9 model fits.
Extracting the ``LogL:'' lines from the former (FIXED) model gives a comparison:

```
LogL:    LogL  Residual  NEDF  NIT Cycle Text
LogL: -325.72   11.3797   187    2 D10MIT31 "LogL Converged"
LogL: -316.22   10.2855   187    2 D10MIT42 "LogL Converged"
LogL: -315.39   10.1947   187    2 IGF1 "LogL Converged"
LogL: -316.21   10.2847   187    2 D10MIT9 "LogL Converged"
LogL: -315.16   10.1688   187    2 D10MIT10 "LogL Converged"
LogL: -311.34   9.75879   187    2 D10MIT41 "LogL Converged"
LogL: -310.87   9.71001   187    2 D10MIT12 "LogL Converged"
LogL: -315.70   10.2269   187    2 D10NDS2 "LogL Converged"
LogL: -324.02   11.1714   187    2 D10MIT14 "LogL Converged"
Local Peak at CYCLE    7 D10MIT12    LogL:  -310.87 Deviance   29.71
```

As this fits the model as a fixed effect, the best model will have the smallest residual (but not necessarily the highest LogL), and is indeed D10MIT12, but its neighbour D10MIT41 is almost as good!

Fitting the markers separately as random effects gives similar results.

```
LogL:    LogL  Residual  NEDF  NIT Cycle Text
LogL: -326.47   11.3797   188    5 D10MIT31 "LogL Converged"
LogL: -317.47   10.2855   188    7 D10MIT42 "LogL Converged"
LogL: -316.66   10.1947   188    7 IGF1 "LogL Converged"
LogL: -317.46   10.2847   188    7 D10MIT9 "LogL Converged"
LogL: -316.43   10.1688   188    7 D10MIT10 "LogL Converged"
LogL: -312.68   9.75882   188    7 D10MIT41 "LogL Converged"
LogL: -312.22   9.71006   188    7 D10MIT12 "LogL Converged"
LogL: -316.95   10.2269   188    7 D10NDS2 "LogL Converged"
LogL: -324.87   11.1714   188    6 D10MIT14 "LogL Converged"
Local Peak at CYCLE    7 D10MIT12    LogL:  -312.22 Deviance   28.51
```

The value known as Deviance is twice the difference in LogL between the
best and worst fits (326.47-312.22)*2.

An equivalent way of coding this job (but losing the marker names) is

```
!RENAME !NOGRAPH !ARG 1 2 !CONTINUE
Single QTL search: Data from Horvat and Medrano, 1995. Genetics 139:1737-1748
 Mouse
 Marker !G 9
 GAIN

HM.dat !skip 1  !DOPART  $1
!CYCLE 1:9
!PART 1//GAIN ~ mu Marker[ I]
!PART 2//GAIN ~ mu !R Marker[ I]
```

Following are the marker effects (allele substitution effects) from the two models.

```
  Marker       Fixed model (SE)          Random model (SE)
  D10MIT31     1.291    ( 0.3657 )       1.188    ( 0.3507 )
  D10MIT42     2.112    ( 0.3638 )       2.049    ( 0.3584 )
  IGF1         2.166    ( 0.3628 )       2.106    ( 0.3577 )
  D10MIT9      2.120    ( 0.3653 )       2.058    ( 0.3598 )
  D10MIT10     2.169    ( 0.3604 )       2.109    ( 0.3554 )
  D10MIT41     2.320    ( 0.3436 )       2.269    ( 0.3398 )
  D10MIT12     2.349    ( 0.3435 )       2.299    ( 0.3398 )
  D10NDS2      2.110    ( 0.3569 )       2.050    ( 0.3518 )
  D10MIT14     1.422    ( 0.3534 )       1.334    ( 0.3423 )
```

If we plotted the profile of the marker effects (or the residual variances),
we would see a peak between markers 6 and 7 and postulate a QTL in that vicinity.

Whole Genome QTL analysis: Mixed Model Regression Mapping

A criticism of Simple interval mapping is that there could be several influential markers and we need to be able to test for markers adjusting for other significant markers. The usual approach is to fit the identified influential markers in the model and then scan the others for influence.

Another way of doing this is to fit the markers simultaneously, as random effects. One approach is implemented in the R wgaim software (Verbyla et al 2012) based on ASReml-R. Another approach (Gilmour, 2007) described here is implemented in **ASReml-SA**.

The code

```
Single QTL search: Data from Horvat and Medrano, 1995. Genetics 139:1737-1748
 Mouse
 Marker !G 9
 GAIN
HM.dat !skip 1  !DOPART  1 !CONTINUE

!PART 1 # Base model gives LogL = -330.778
GAIN ~ mu
!PART 2 # Marker model gives LogL = -313.304
GAIN ~ mu !r Marker
```

The logic here is that under the null hypothesis there are no QTL linked with the markers and therefore the marker covariables are just error contrasts, and a variance component based on them should therefore be zero since there is no extra variation.

```
    6 LogL=-313.304     S2=  9.4332        188 df   0.1311
 Final parameter values                        0.1314

          - - - Results from analysis of GAIN - - -
 Akaike Information Criterion      630.61 (assuming 2 parameters).
 Bayesian Information Criterion    637.08
```

```
         Approximate stratum variance decomposition
 Stratum     Degrees-Freedom   Variance      Component Coefficients
 Marker                 8.86    21.5558          9.8     1.0
 Residual Variance    179.14    9.43318          0.0     1.0

 Source                Model  terms     Gamma     Component    Comp/SE   % C
 Marker                    9      9  0.131402       1.23954       1.18   0 P
 Variance                189    188   1.00000       9.43318       9.46   0 P
```

The change in LogL is highly significant so we conclude that a QTL is present.

The BLUPs of the simple marker effects are

```
  Marker                        1              -0.6351      0.4816
  Marker                        2                1.098      0.6503
  Marker                        3               0.2942      0.8226
  Marker                        4              -0.4423      0.8789
  Marker                        5              -0.3952      0.9080
  Marker                        6                1.146      0.8003
  Marker                        7                1.527      0.7927
  Marker                        8               0.2513      0.7442
  Marker                        9              -0.6388      0.5007
```

and we see Markers 6 and 7 are again dominant but Marker 2 is also relatively large.
Now, since we know marker positions, we could plot the marker effects against marker position.

But marker covariables are not independent; neighbouring markers are correlated. Gilmour (1997) describes a QTL detection method based on the simultaneous fitting of all markers as random effects, and recreating a marker profile assuming the QTL effect has been taken up by several markers in the vicinity of the QTL according to the distance from the QTL. The theory holds for Backcross and F2 data, where the correlation between marker variables is directly related to the map distance. The mouse data was derived from an F2 cross of two inbred lines so the distance between markers provides a basis for calculating the expected correlation between the effects of neighbouring markers.

The method is invoked in **ASReml** by appending the map distances to the definition of the marker variables, and appending a
`PREDICT Marker !PLOT`
statement produces a plot of the predicted marker effects.

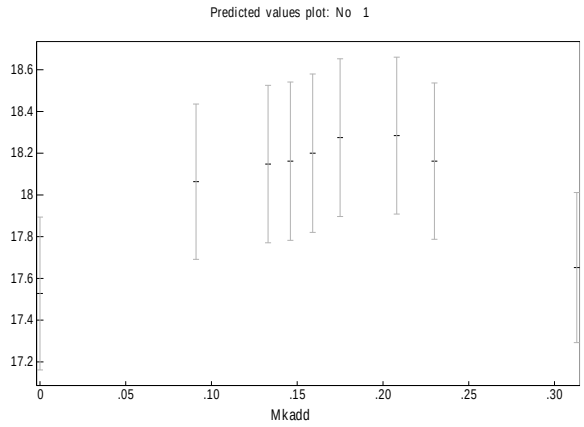
```
Single QTL search
!  Data from: Horvat and Medrano, 1995.  Genetics 139:1737-1748
!  Data coded 0=aa, 1=aA, 2=AA
 SEQ
! D10MIT31 !-1 # 9.1 Aa Aa Aa Aa aa AA Aa Aa aa Aa Aa Aa Aa AA aa Aa aa Aa Aa
 Mkadd !G 9 !rescale -1 1. !MM 0. 9.1 13.3 14.6 15.9 17.5 20.8 23 31.3 35
 GAIN
 Mkdom !G 9 !dom Mkadd # takes (MkAdd^2-0.5)*2
HM.dat !skip 1 !READ 11 !EXTRA 2
GAIN ~ mu !r Mka
predict Mka !PLOT
```

The BLUPs of the predicted QTL effect if a single QTL was at the respective marker positions are

```
  Marker          Predicted_Value Standard_Error Ecode
       0.0000             15.3225         0.2977 E
       0.0910             15.8584         0.2807 E
       0.1330             15.9428         0.2871 E
       0.1460             15.9567         0.2906 E
```

```
      0.1590            15.9950         0.2936 E
      0.1750            16.0695         0.2954 E
      0.2080            16.0792         0.2968 E
      0.2300            15.9568         0.2966 E
      0.3130            15.4469         0.2970 E
 Notice: A turning point is at  0.1894 M; value is   16.1024 for Marker
 SED: Overall Standard Error of Difference   0.1634
```

ASReml fits a local quadratic to the marker positions to identify the turning points, which are potential QTL positions. (MMRM.wmf)



The next step would be to fit a covariable at the nominated position, and see if it removes the variance of the random effects. This is achieved by adding the model term qtl(Marker,0.189).

```
  6 LogL=-310.134     S2=  9.4412        187 df   0.3872E-01
 Final parameter values                     0.3741E-01

         - - - Results from analysis of GAIN - - -
 Akaike Information Criterion      624.27 (assuming 2 parameters).
 Bayesian Information Criterion    630.73

         Approximate stratum variance decomposition
 Stratum     Degrees-Freedom   Variance      Component Coefficients
 Marker                7.63    13.9428        12.7     1.0
 Residual Variance    179.37    9.44123         0.0     1.0

 Source                Model  terms     Gamma     Component    Comp/SE   % C
 Marker                    9      9  0.374133E-01  0.353227       0.62  -2 P
 Variance                189    187   1.00000       9.44123       9.47   0 P

                                Wald F statistics
    Source of Variation         NumDF     DenDF    F-inc           P-inc
  4 mu                              1     136.1   837.62           <.001
  5 qtl(Marker,0.189)               1       1.8     7.11           0.117
 Notice: The DenDF values are calculated ignoring fixed/boundary/singular
           variance parameters using algebraic derivatives.

                   Solution       Standard Error    T-value     T-prev
  5 qtl(Marker,0.189)
                  1    3.43464        1.28831          2.67
```

The QTL effect appears NS but that is because it is tested against the residual marker variance. Dropping the residual marker effect gives the proper test (and shows there is no more marker variance to explain).

```
  2 LogL=-310.269     S2=  9.6499        187 df

         - - - Results from analysis of GAIN - - -
 Akaike Information Criterion      622.54 (assuming 1 parameters).
 Bayesian Information Criterion    625.77

 Source                Model  terms     Gamma     Component    Comp/SE   % C
 Variance                189    187   1.00000       9.64993       9.67   0 P

                                  Wald F statistics
    Source of Variation           NumDF     DenDF    F-inc            P-inc
  4 mu                                1     187.0  5161.23            <.001
  5 qtl(Marker,0.189)                 1     187.0    48.22            <.001
 Notice: The DenDF values are calculated ignoring fixed/boundary/singular
            variance parameters using algebraic derivatives.

                    Solution       Standard Error    T-value     T-prev
  5 qtl(Marker,0.189)
                   1    2.41959       0.348439          6.94
```

Genomic Selection

Given we now have access to genomic data, it can be used in two main ways. The first is to try and identify QTL (regions in the genome with large phenotypic variation dependent on which alleles are present). The second is to use the markers to define overage genomic relationships between individuals to facilitate selecting superior genotypes.

The basic GBLUP model for Genomic selection is

$$\mathbf{y} = \mathbf{X\beta} + \mathbf{Mg} + e$$

where **M** is a matrix (*n* x *m*) of marker (*m*) scores (values 0/1/2 being counts of minor allele) for each genotype (*n*)
and **g** are the BLUPs of the marker effects assumed to have common variance as proposed by Meuwissen *et al.* (2001). More recently, technology has improved so that it is common to have many more markers than the number of individuals genotyped (*m*>>*n*). Fitting a model with a large number of markers becomes infeasible as that number increases. This has lead to the marker model being reformulated as a genotype model using a genomic relationship matrix we write as
$\mathbf{G_A = MDM'}$ where **D** = diag(1/*s*), $s = \Sigma_{(i=1,m)} H_i = \Sigma\ 2p_i(1\text{-}pzi)$, H_i is the heterozygosity associated with the *i*th marker and p_i is the probability of the minor allele. We then fit the (more general) model

$$\mathbf{y} = \mathbf{X\beta} + \mathbf{Zu} + e$$

where var(**u**) = $\sigma^2\mathbf{G_A}$ that is, using $\mathbf{G_A}$ instead of (or in addition to) the usual numerator relationship matrix (**A**) based on a pedigree in an animal model. The link between the models is that **u** = **Mg** and **g** = $\mathbf{M'G^{-1}u}$.

This is one of several ways of defining $\mathbf{G_A}$ in the literature. Given the user has created $\mathbf{G_A}$ it (or its inverse) can be supplied to **ASReml** as a `.grm` (`.giv`) file as in the following example.

```
!WORK 12  !ARG 1
QTL ANALYSIS
 id !P # pedigree factor
 SEX    2  !A
 AGE   73  !A
 HEIGHT 1  !M-9999

ibdgrm.ped  !MAKE !ALPHA # Pedigree file
ibdgrm.grm !ND !DENSE    # GRM matrix: dense format lower triangle rowwise
ibdgrm.dat               # Data file

HEIGHT ~ mu SEX !R nrm(id) grm1(id)
```

It has been observed that user supplied GRM matrices are often not positive definite, and are sometimes singular. Singularity can arise because it is rank deficient (fewer markers than individuals), some individuals are clones, or because there is no effect associated with a particular individual (such as no dominance effect for a fully inbred individual). Also, the matrix may be negative definite (some negative eigen values) possibly because of unbalance in the data from which the matrix was formed, or because of insufficient precision in the values obtained from the `.grm` file. **ASReml** 4 allows this matrix to be supplied as a binary file, which retains precision and is faster to input; indicated by using filename extensions `.dgrm, .dgiv, sgrm, .sgiv`.

ASReml has three qualifiers `!ND, !NSD, !PSD` which can be used to instruct **ASReml** to proceed with the analysis even though the supplied GRM matrix may be Negative Definite, Negative Semi-Definite or Positive Semi-Definite respectively. For the singular case, **ASReml** applies Lagrangian constraints in forming the $\mathbf{G}_A^{-1}$ matrix it needs to undertake the analysis (See the companion report on Non-Singular Matrices in **ASReml** for more detail).

An option was added in 2012 for **ASReml** to form $\mathbf{G}_A = \mathbf{MDM'}$ if the user supplies a `.grr` file containing the marker data. The markers are assumed coded 0/1/2 being the incidence of the minor allele. Missing marker scores are replaced by the mean marker score in this process. This was generalised and revamped in 2014 allowing more general 'regression variables' . It was further extended in 2022 following Vitezica et al (2017) to form

$\mathbf{G}_A=\mathbf{MM}'/(2\ \Sigma(j=1:m)\ p_j(1-p_j))$ where **M** has elements $(0-2p_j)$, $(1-2p_j)$ and $(2-2p_j)$ for genotypes AA, AB and BB respectively, $2p_j$ being the mean for SNP *j*.

$\mathbf{G}_D=\mathbf{KK}'/(4\ \Sigma\ (j=1:m)\ (p_j^2(1-p_j)^2))$ where

K has elements $-2p_j^2$, $2p_j(1-p_j)$ and $-2(1-p_j)^2$ for genotypes AA, AB and BB respectively

$\mathbf{G}_{AA} = \mathbf{G}_A * \mathbf{G}_A/(\mathrm{tr}(\mathbf{G}_A * \mathbf{G}_A)/n)$,

$\mathbf{G}_{AD} = \mathbf{G}_A * \mathbf{G}_D/(\mathrm{tr}(\mathbf{G}_A * \mathbf{G}_D)/n)$ and

$\mathbf{G}_{DD} = \mathbf{G}_D * \mathbf{G}_D/(\mathrm{tr}(\mathbf{G}_D * \mathbf{G}_D)/n)$ where * represents the Hadamard product and tr() is the trace function.

The `.grr` file typically has a heading (first line) and a data row for each genotype, each data row beginning with a genotype ID. The .grr file is specified where the .grm matrix would otherwise be specified, and is incorporated into the model using the `grmi(ID)` model term where *I* selects which GRM matrix to use. When the GRM matrix is computed from the markers, and fitted as a single factor (i.e. not in an interaction), **ASReml** backsolves to report the marker effects in a file with filename extension `.mef` (`.eme` in Echidna).

A related procedure has been implemented in the R wgaim package (Verbyla et al, 2012 . This is further discussed in the next chapter where it is extended to a 'Fast Bayes A like' method.

GRR Syntax

The expected structure of the marker data file is

```
Genotype M1 M2 M3 …
G1   0 0 1 …
G2   2 1 0 …
…
```

where the first line is names for all the columns which begin with a Genotype label followed by the SNP values coded 0, 1 or 2 (* or NA if value is unknown). The syntax is a single line beginning with a

file name and then qualifiers. The line is recognized by the file name which must have a .grr file extension.

The main qualifiers are:

!IDS *g*	The number of genotypes (or slightly more)
!ALPHA *g*	declares the *g* genotype labels are alphanumeric
!NOID	indicates the first column of genotype labels is missing
!MARKERS *m*	The number of markers (or slightly more)
!NOHEAD	indicates the first line of column labels is missing
!CSKIP *c*	is used to skip fields before the marker data
!SKIP s	is used to skip data lines before the marker data.
!DOM	requests $\mathbf{G}_D$ (see below) be formed as well as $\mathbf{G}_A$
!EPI	requests $\mathbf{G}_{AA}$ ($\mathbf{G}_{DAD}$ and $\mathbf{G}_{DD}$) be formed.
!FBA	requests the 'Fast Bayes A' approximation be used (request details from author).
!SAVEGIV	writes the GRM (inverse) to file
!PSD *s*	declares that the derived variance matrix may have up to s singularities,
!PEV	requests calculation of Prediction Error Variance of marker effects which are reported in the .mef/.eme file. The recommendation is to always use !PEV
!CENTRE [c]	requests the regressors be centered at c if c is specified else at the individual regressor means

The .grr file may in fact of regressors that are not SNP codes in which case the following qualifiers may be needed.

!SMODE b	sets the storage mode for the regressor data, b = 2 sets 2bit storage for strictly 0/1/2 marker data, b = 8 (the default) sets 8bit storage useful for marker data with imputed values having 2 digits after the decimal, b = 16 sets 16bit storage useful for marker data with imputation with more than 2 digits and b = 32 sets 32bit real storage and should be used for non-marker data
!RANGE l h	indicates the marker scores range l : h and are to be transformed to have a range 0:2,
!GSCALE s	controls the scaling of the GRM matrix. If unspecified s = $\Sigma 2p(1-p)$ is used for marker data, s = 1 for non-marker data (!SMODE 32). Scaling is often used with centred marker data to scale the MM' matrix so that it is a genomic matrix.

Alternatively, the .grr file name may be followed by a Name for the Genotype Factor, the number of genotypes, a name for the Regressors and the number of variables, and other qualifiers as required.
If the genotype identifiers are not present (!NOID), **ASReml** assumes that the order of the factor classes in the data file matches the order in the .grr file.
If the factor identifiers are present and named (as in the alternative just mentioned) and the same name appears in the data file specification, **ASReml** (but not Echidna yet) uses the identifiers obtained from the .grr file to define the order of the factor classes when the data is read;

any extra identifiers in the data not in the .grr file are appended at the end of the factor level name list.
If !NOID is set, identifiers in the .grr file are not needed and if present should be skipped using !CSKIP.
Values are typically TAB, COMMA or SPACE separated but may be packed (no separator) when all values are integers 0/1/2. Missing values in the regression variables may be represented by *, NA. Invalid data is also treated as missing. Missing values are replaced by the mean of the respective regressor. Alternative missing data methods that involve imputation from neighbouring markers have not been implemented.

Example

This dataset has 860 genotypes with replicated data, 923 with genotype (4854 SNP markers); 3 with data do not have genotype.

```
!WORK 1 !RENAME 2 !out !NOGRAPHS  !arg HT6 2
Testing Pedigree Matrices vs Marker Matrices with Nassau Data   !DOPART  2
 Nfam 71 !A
 Nfemale 26 !A
 Nmale 37 !A
 Clone  !A 926  # !L  snpData.grr !LSKIP 1
 MatOrder 914 !A
 rep 8 !A
 iblk 80 !A
 tree row col
 prop 1 !A
 culture 2 !A
 treat 2 !A
 measure 1 !A
 SURV DBH6 HT6 HT8 CWAC6 !M-9

#nass_ped_v22.txt !SKIP 1 !ALPHA !MAKE                  # Pedigree
snpData.grr Clone 923 !DOM !EPI !PEV
nassau_cut_v3.csv !MAXIT 30 !SKIP 1 !DDF -1             # DataNassau Clone Data
!PART 1
 1 ~  mu culture culture.rep  !r Nfam Nfem Nmal Clone rep.iblk
!PART 2
 1 ~  mu culture culture.rep  !r grm1(Clone) Clone  rep.iblk

!PART 3
 1 ~  mu culture culture.rep  !r grm1(Clone) 0.28 grm2(Clone) 0.28 Clone 0.15 rep.iblk 0.31
!PART 4
 1 ~  mu culture culture.rep  !r grm1(Clone) 0.28 grm3(Clone) 0.28 Clone 0.15 rep.iblk 0.31
!PART 5
 1 ~  mu culture culture.rep  !r grm1(Clone) 0.28 grm2(Clone) 0.28 grm3(Clone) +
 Clone 0.15 rep.iblk 0.31
!PART 6
 1 ~  mu culture culture.rep  !r grm1(Clone) 0.28 grm2(Clone) 0.28 grm4(Clone) +
Clone 0.15 rep.iblk 0.31
!PART 7
 1 ~  mu culture culture.rep  !r grm1(Clone) 0.28 grm2(Clone) 0.28 grm5(Clone) +
Clone 0.15 rep.iblk 0.31
```

where snpData.grr is first (in **ASReml**)used to declare Clone identifiers (taken from the first field) in the correct order, and then contains the marker scores; it looks like

```
Genotype,0-10024-01-114,0-10037-01-257,0-10040-02-394,...
140099,2,2,1,2,2,2,2,2,1,2,1,2,1,1,2,1,2,2,2,2,2,1,2...
141099,2,2,0,0,2,2,1,2,2,1,2,1,2,2,0,2,2,2,2,1,2,2,1,1...
...
547853,2,2,1,2,2,2,1,2,2,0,2,1,2,2,2,2,2,2,2,1,2,...
547966,2,2,1,1,1,2,0,2,2,1,2,2,2,2,2,2,2,2,2,1,2,...
548082,2,2,1,2,2,2,1,2,1,2,2,1,2,2,1,2,2,2,2,1,2,...
```

The primary output follows.

```
ASReml 4.3nh [10 Feb 2023] Testing Pedigree Matrices against Marker Matrices for Variance
Partition with Na
 Windows x64    1.0 Gbyte  CloneHT6_4.a/Clone  22 Feb 2023 13:26:43.930
 * Licensed to: Arthur Gilmour    31-dec-2023

Folder: C:\MMX\Ex\GRM\PatricioNassau
 Nfam 71 !A
 Nfemale 26 !A
 Nmale 37 !A
 Clone  !A 926
 MatOrder 914 !A
 rep 8 !A
 iblk 80 !A
 prop 1 !A
 culture 2 !A
 treat 2 !A
 measure 1 !A
 CWAC6 !M-9

 Parsing: snpData.grr Clone !DOM !EPI
 Class names for factor "Clone" are initialized from the .grr file.
 GRR Header line begins: Genotype,0-10024-01-114,0-10037-01-257,0
        4854 Marker labels found
Marker labels 0-10024-01-114 ... UMN-CL98Contig1-02-
 Notice: The header line indicates there are 4854 regressors in the file.
 Notice: SNP data line begins: 140099,2,2,1,2,2,2,2,2,2,1,2,1,2,1,1,
 Notice: Markers coded -9 treated as missing.
          Use !RANGE min max  if this value is to be included.
 Marker data [0/1/2] for 923 genotypes and 4854 markers read from snpData.grr
      160414 missing Regressor values (  3.6%) replaced by column average!
        Regressor values ranged 0.00 to 2.00
        Regressor Means ranged  1.00 to 2.00
 Regressors centered at their respective means
          Sigma (2p(1-p)) is     1057.12558
          Sigma (2p(1-p))^2 is    372.81831

 GIV  Identifier      Rows    Type      LogDet  GroupsDF
   1  snpData.grr      923       9     -959.99         0
   2  snpData_DOM      923       9     -551.40         0
   3  snpData_AxA      923       9     -164.33         0
   4  snpData_AxD      923       9      -78.68         0
   5  snpData_DxD      923       9      -46.82         0

 QUALIFIERS: !MAXIT 30 !SKIP 1 !DDF -1
 QUALIFIERS: !SLN
 QUALIFIER: !DOPART 2 is active
 Reading nassau_cut_v3.csv  FREE FORMAT skipping     1 lines

 Univariate analysis of HT6
 Summary of 6399 records retained of 6795 read

  Model term          Size #miss #zero   MinNon0    Mean      MaxNon0  StndDevn
   1 Nfam               71     0     0      1    36.3379         71
   2 Nfemale            26     0     0      1    12.8823         26
   3 Nmale              37     0     0      1    15.2285         37
   4 Clone             926     0     0      1   464.6765        926
  Warning: Fewer levels found in MatOrder  than specified
   5 MatOrder          914     0     0      1   432.5760        860
   6 rep                 8     0     0      1     4.4837          8
   7 iblk               80     0     0      1    40.1164         80
   8 tree                      0     0  1.000      7.473      14.00      4.018
   9 row                       0     0  1.000      28.52      56.00      16.09
  10 col                       0     0  1.000      10.50      20.00      5.760
```

```
 Warning: Fewer levels found in prop  than specified
 11 prop                2     0     0      1     1.0000          1
 12 culture             2     0     0      1     1.4945          2
 13 treat               2     0     0      1     1.4945          2
 Warning: Fewer levels found in measure  than specified
 14 measure             2     0     0      1     1.0000          1
 15 SURV                      0     6  1.000     0.9991      1.000     0.3061E-01
 16 DBH6                      4     0 0.3000E-01  11.29      18.80      2.400
 17 HT6            Variate    0     0  76.20      838.6      1286.      163.6
 18 HT8                      83     0  91.44      1148.      1576.      170.6
 19 CWAC6                  3167     0  97.54      301.3      542.5      52.26
 20 mu                        1
 21 culture.rep              16 12 culture   :   2   6 rep           :    8
Note: The GRM matrix specified in grm1(Clone) is smaller ( 923) than Clone ( 926)
       and is extended with an Identity to cover the extra levels.
 22 grm1(Clone)       926
 23 rep.iblk                640  6 rep       :   8   7 iblk          :   80
Forming     2511 equations:    19 dense.
Initial updates will be shrunk by factor    0.316
* This job uses all of the 12 processor threads. *
Notice: LogL values are reported relative to a base of -30000.000
Notice: 11 singularities detected in design matrix.
  1 LogL= -2738.47     S2=  7697.0       6391 df
  2 LogL= -2738.47     S2=  7697.7       6391 df
  3 LogL= -2738.46     S2=  7699.2       6391 df

         - - - Results from analysis of HT6 - - -
Akaike Information Criterion    65484.93 (assuming 4 parameters).
Bayesian Information Criterion  65511.98

Model_Term                            Gamma         Sigma   Sigma/SE   % C
rep.iblk               IDV_V  640  0.307845       2370.17      13.00   0 P
grm1(Clone)            GRM_V  926  0.284090       2187.27       5.87   0 P
Clone                  IDV_V  926  0.148880       1146.26       5.91   0 P
Residual               SCA_V 6399   1.00000       7699.23      49.64   0 P

                                Wald F statistics
    Source of Variation         NumDF              F-inc
 20 mu                              1           0.11E+06
 12 culture                         1            2616.61
 21 culture.rep                     6              30.45
 23 rep.iblk                      640 effects fitted
 22 grm1(Clone)                   926 effects fitted
  4 Clone                         926 effects fitted (      66 are zero)
* This job used at least .3 of the 1.0 Gbyte of primary workspace. *
     78 possible outliers in Section 11: see .res file
Finished: 24 Feb 2023 11:52:16.274   LogL Converged
```

Notes:

- 926 clones identified, 860 have data and 923 have genomic data. The GRM is therefore initially of order 923 but is expanded to 926 when used in the model by appending diagonal elements of 1 since the relationship with other genotypes is unknown for these extra genptypes with data.
- The .res file contains additional details about the analysis including a listing of the larger marker effects. All marker effects are reported in the .mef file.
- Particular columns of the .grr data can be included in the model using the grr(Clone,*i*) model term where and *I* specifies which (number) regressor variable to include.

```
Listing of the larger marker/regressor effects
       368  0-12761-01-121     1.43392       1.35248
       617  0-14383-01-111     1.28761       1.38239
       777  0-15417-01-138    -1.29011       1.34868
      1246  0-18644-02-210     1.25224       1.36135
```

```
1903  0-6963-01-202     -1.28258       1.35400
2445  2-1563-02-244     -1.38473       1.35886
2497  2-2167-01-413     -1.24139       1.36389
3180  2-8668-03-42      -1.24819       1.36847
3521  CL1577Contig1-03  -1.19473       1.35261
3802  CL2573Contig1-03   1.19015       1.35333
4195  CL595Contig1-01-  -1.22746       1.37156
4351  UMN-1397-01-416   -1.38282       1.34570
```

Genomic model vs Pedigree model

The title line in the preceding job indicates the researchers who shared this data with me were interested to compare a pedigree based analysis with the Genomic (GBLUP) analysis.

Since the pedigree file genotype order is different from the order of genotypes in the .grr file, I duplicated the 'Clone' field, declared the inserted field `PClone !P`, and activated the pedigree line `nass_ped_v22.txt !SKIP 1 !ALPHA`

And fitted the model

```
!PART 11 # Pedigree
 1 ~  mu culture culture.rep  !r nrm(PClone) 0.28 Clone 0.15  rep.iblk 0.31
```

Getting

```
      22 individuals appear as both male and female parent
     914 identities in the pedigree,
         generations on   ma side range 2 to 3
         generations on   pa side range 2 to 3
       ma     ma_of_ma     pa_of_ma       pa     ma_of_pa     pa_of_pa
       34            7            5       42            9            5
…
   8 LogL= -2737.28     S2=  7696.1       6391 df  :   1 components restrained

          - - - Results from analysis of HT6 - - -
 Akaike Information Criterion    65482.56 (assuming 4 parameters).
 Bayesian Information Criterion  65509.61

 Model_Term                            Gamma         Sigma   Sigma/SE   % C
 rep.iblk               IDV_V  640  0.309628       2382.93      13.02   0 P
 nrm(PClone)            NRM_V  914  0.522151       4018.54       3.64   0 P
 Clone                  IDV_V  926  0.133354E-06  0.102631E-02   0.00 -90 B
 Residual               SCA_V 6399   1.00000       7696.12      49.65   0 P

                                 Wald F statistics
    Source of Variation          NumDF              F-inc
  21 mu                              1            4855.63
  13 culture                         1            2608.23
  22 culture.rep                     6              30.38
  24 rep.iblk                      640 effects fitted
  23 nrm(PClone)                   914 effects fitted
   5 Clone                         926 effects fitted (      66 are zero)
```

This model is basically equivalent to the Genomic analysis, the LogL being 1.18 higher.

Is there dominance or epistatic variance?

Parts 3 to 7 of the GBLUP code fit various models which look for structure in the non-additive (Clone) variance which is not significant in the pedigree based analysis but quite large in the GBLUP (A) analysis. The results are summarized in the following table.

Comparison of model fits to Ht6

	A(ped)+I	A+ I	A+D+ I	A+AA+ I	A+D+ AA+I	A+D+ AD+I	A+D+ DD+I
LogL	-32737.28	-32738.46	-32736.06	-32728.45	-32728.45	-32732.45	-32733.99
$\mathbf{G}_A$ (ped)	4018.54						
$\mathbf{G}_A$		2187.38	1973.16	1184.35	1184.35	1645.64	1838.91
$\mathbf{G}_D$			470.305		0.01	80.40	94.93
$\mathbf{G}_A$*$\mathbf{G}_A$				1908.67	1908.66		
$\mathbf{G}_D$*$\mathbf{G}_A$						1456.01	
$\mathbf{G}_D$*$\mathbf{G}_D$							1294.14
$\mathbf{G}_I$	0.00	1146.21	871.995	0.00	0.00	0.00	0.00
Residual	7696.12	7699.22	7698.66	7687.67	7687.67	7691.80	7693.37
Total Genetic	4018.54	3333.59	3325.46	3093.02	3093.02	3182.05	3227.98

Adding the $\mathbf{G}_A$*$\mathbf{G}_A$ term to the pedigree based model did not improve its LogL significantly (LogL -32736.72) so there is no definitive answer in this data as to whether the marker based model is superior.

High Dimensional Whole Genome Analysis: Fast Bayes A

When there was a major focus on looking for QTL, several Bayesian approaches were taken which attempted to identify SNP markers explaining large amounts of genetic variation. This chapter describes a mixed model variation on the Fast Bayes A approach. It is not commonly used but is reported for those interested in exploring methods which search for markers of great influence.

Sun et al. (2012) developed a mixed model form of the popular 'Bayes A' model which has been implemented in {\sf ASReml-SA for predicting genetic merit based on a genomic relationship matrix, and of identifying markers of relatively large influence in an Association mapping context.
The implementation here is not identical to the paper but is effectively equivalent.

The method is described in overview in the next section. This is followed by the particular syntax required for fitting these models. We then present examples: a GBLUP analysis followed by a 'Fast Bayes A' (FBA) analysis. Next we discuss the differences between what **ASReml** does and what Sun et al, 2012 proposed. We conclude the chapter discussing issues that have been raised.

Outline of Fast Bayes A like method

The statistical context is that we have a matrix **M** of scores for *m* markers evaluated for *n* genotypes and have phenotypic data on the *N* (< *n*) genotypes. Typically *n* is small (<5000) and *m* is large (*m*>>*n*). The usual coding of this data is in terms of allele substitution effects (i.e. 0,1,2).

The GBLUP approach discussed in the previous chapter is based on forming $\mathbf{G}_A$ = **MDM'** where **D** = diag(1/*s*), *s* = Σ $2p_i(1-pzi)$. In this case. The diagonal elements of **D** are constant and just scale the cross product so that the resulting variance components are comparable to those obtained using a NRM. Here, a large number of markers provide an empirical measure of genetic relationship.
Each marker is given equal weight assuming they are uniformly distributed across the genome.

Bayes A is a Monte Carlo Markov Chain method proposed by Meuwissen, et al., (2001). It is assumed that there are some alleles (QTL) of large effect influencing the phenotype, and these will be in linkage disequalibrium with nearby markers. Rather than giving equal weight to each marker, it

seeks to give increased weight to markers of large effect. Under the equal variance model, the effect of any QTL will by smeared over several nearby (and therefore correlated) markers. But if the markers are weighted according to their magnitude, one is emphasized and the others deprecated. Therefore, rather than assuming marker effects, γ_j, have a common variance, Bayes A assumes a scaled inverse Chi-square distribution for the individual marker variances, σ^2_j, resulting in a t distribution prior for the marker effects. The t distribution, with low degrees of freedom (k), is a peaked distribution with long tails. Using this distribution results in most markers having small effect but some possibly having very large effects. Algebraically

$\gamma_j \mid \sigma^2_j \sim N(0, \sigma^2_j)$, $\sigma^2_j \sim kS_j^2 \chi_k^{-2}$, $j = 1,\ldots,m$

leading to a marginal distribution $\gamma_j \sim t(0, S_j^2, k)$, that is, $E[Var(\gamma_j)]=E[S_j^2] = kS_j^2/(k-2) = \sigma^2_\gamma$ where k is the degrees of freedom of the Chi-squared distribution (and therefore controls the skewness of the distribution) and S_j^2 is a scaling parameter.

The Bayes-A like algorithm described by Sun et al. (2012) involves forming $\mathbf{G_A = MDM'}$ where $\mathbf{D} = diag(d_j)$ weights the markers according to the effect variances and $d_j = (\gamma_j^2 + kS_\gamma^2)/(k+1)$ where $\boldsymbol{\gamma}$ is the vector of estimated marker effects from the latest fit of the mixed model, and S_γ^2 is a prior for the scale parameter. Given $kS_\gamma^2 = (k-2)\sigma^2_\gamma$, he used the marker variance estimated from the GBLUP model ($\sigma^2_\gamma = \sigma^2_G/s$) and so calculated $d_j = (\gamma_j^2 + (k-2)\sigma^2_G/s)/(k+1)$. In **ASReml** we supply the variance parameter (σ^2_G) as a scaling factor to $\mathbf{G_A}$ and so use $d_j = (\gamma_j^2/\sigma^2_G + (k-2)/s)/(k+1)$. This expression is a combination of the weight used in GBLUP ($1/s$) and the squared marker effect suitably scaled.

This expression holds when σ^2_G is known and held fixed across iterations. However if σ^2_G is not fixed, the REML machinery does not give a valid estimate of σ^2_G under the model; the weight expression regresses the relative effect variances toward (k-2)/(k+1) and so the REML machinery, assuming Normality, compensates by increasing the estimate of σ^2_G. Consequently, when estimating \sigma^2_G , **ASReml** uses d_j = (γ^2_j/σ^2_G +k/s) / (k+1) . This applies less shrinkage of the marker effects (for a given value of k) but removes the bias from the variance parameter estimate.
The value of s which relates the marker variance to the genetic variance is updated to reflect the changing weights applied to the marker covariates in forming $\mathbf{G_A}$. The weighted value is
$s = m\Sigma\, 2d_j p_j (1-p_j)/(\Sigma\, d_j)$ although this effect is minor.

Bayes B is a variation on the Bayes A model also proposed by Meuwissen, et al., (2001) in which a proportion of the smallest marker effects are fixed at zero. Analogously, an option exists in **ASReml** whereby the smallest marker effects are progressively set to zero as iteration proceeds until a nominated proportion are zero.

Syntax for Fast Bayes options

The following qualifiers apply to the .grr marker file line. Bayes options may not be used with the !DOM and/or !EPI qualifiers.

<filename> grr !MARKERS <m> !IDS <n >
or <filename .grr <Genotype> <n> <Markers> <n>
with !FBA k
!FBB p
as needed.

Although listed here over 7 lines, qualifiers must be specified on a single line after the filename.

The first two qualifiers specify the size of the marker matrix (n rows/genotypes, m columns/ markers). They are required if the marker/genotype identifier labels are not present or if **ASReml** fails to count them properly. The name <Genotype> should match the Factor name given previously in relation to the data when the genotype class names in the .grr file match those given in the data file.

The !FBA qualifier controls whether **ASReml** performs a simple GBLUP (equal marker variances (!FBA omitted, or *k*=0) or 'Fast Bayes A like' analysis with *k degrees of freedom* specified for the inverse Chi-square distribution of the marker variances. The default value (if !FBA is specified without an argument is *k*=4. If specified, *k*, the *degrees of freedom*, must be at least 3.
If a value greater than 20 is specified, **ASReml** will divide it by 10 (so for example, *k*= 45 uses a value of 4.5 as the degrees of freedom). When !FBA is set, **ASReml** also sets the general qualifier !EXTRA 5 and will attempt to read initial relative marker variances from the .mef file if it exists.

The !FBB *p* qualifier instructs **ASReml** to utilize the Fast Bayes A process extended to progressively set the smallest p% of marker variances (and hence associated marker effects) to zero.

Qualifiers !PENALTY *d* !DFOFFSET *t* !MSCALE *s* were used to test a range of marker variance weights (**D**) and are not intended for general use.

!DFOFFSET *t* defines the degrees of freedom offset used for calculation of **D** affecting the skewness of the distribution of marker variances as discussed later. If unspecified, a value of *t*=2 is used when the variance parameter (or variance ratio) is known and fixed, a value of *t*=0 is used when the variance parameter (or variance ratio) is unknown and being estimated.

!PENALTY *d* qualifier instructs **ASReml** at add d*0.00001 to the diagonal of the G matrix formed. This might be needed with the !FBB qualifier if very many marker effects are fixed at zero or the number of markers is less than the number of individuals genotyped.

The !MSCALE qualifier tells **ASReml** that the variance parameter is the marker variance (ratio), not the Genotype variance (ratio).

Effect of using !FBA on the Nassau data discussed above.

Using

```
snpData.grr Clone !FBA
```

and fitting

```
$1 ~  mu culture culture.rep  !r grm1(Clone) 0.28 Clone 0.15  rep.iblk 0.31
```

ASReml uses the default of 4 degrees of freedom for the t distribution and converges to

```
 16 LogL= -2734.15     S2=  7699.6       6391 df
        - - - Results from analysis of HT6 - - -
Akaike Information Criterion    65476.31 (assuming 4 parameters).
Bayesian Information Criterion  65503.36

Model_Term                              Gamma         Sigma   Sigma/SE   % C
rep.iblk               IDV_V  640  0.307707       2369.21      13.00   0 P
grm1(Clone)            GRM_V  926  0.366914       2825.08       6.06   0 P
Clone                  IDV_V  926  0.138756       1068.36       5.58   0 P
Residual               SCA_V 6399   1.00000       7699.55      49.64   0 P
The model was fitted using [u^2_j/S^2 + (k-t)/s]/(k+1) as effect variance
where S^2=  0.0000E+00; k=  4.00; t=0 and s=  1063.52.
The ratio of genetic:marker variance is    1063.524
```

```
                                Wald F statistics
    Source of Variation         NumDF              F-inc
 20 mu                              1           0.11E+06
 12 culture                         1            2617.35
 21 culture.rep                     6              30.46
```

Comparing with GBLUP model,

	LogL	$\mathbf{G}_A$	$\mathbf{G}_I$	Residual	
FBA 4	-32734.15	2825.08	1068.36	7699.55	
GBLUP	-32738.4	2187.34	1146.21	7699.22	

The estimate of $\mathbf{G}_A$ is inflated. It is recommended that the genetic variance should be held at its GBLUP value.

The same markers are reported as large except CL2573Contig1-03 (MARKER 3802), but with greater magnitude (partly due to the inflation of $\mathbf{G}_A$).

```
Listing of the larger marker/regressor effects
       368  0-12761-01-121     1.81999       1.53858
       617  0-14383-01-111     1.58185       1.54118
       777  0-15417-01-138    -1.58957       1.49523
      1246  0-18644-02-210     1.49556       1.49929
      1903  0-6963-01-202     -1.56000       1.49919
      2445  2-1563-02-244     -1.73621       1.53325
      2497  2-2167-01-413     -1.47977       1.49945
      3180  2-8668-03-42      -1.51762       1.51163
      3521  CL1577Contig1-03  -1.46025       1.48187
      4195  CL595Contig1-01-  -1.48474       1.51069
      4351  UMN-1397-01-416   -1.74417       1.51618
```

CL2573Contig1-03 1.385961 1.472018 0.8882049E-03
is however still large.

Effect of using !FBA on the QTLMAS data

The simulated test dataset provided by Szydlowski, M. & Paczynska, P. (2011) (\http://www.biomedcentral.com/1753-6561/5/S3/S3) came as four files:

- SNP genotypes are in file genotype.mkr which has 3,227 lines and 10,032 columns (fields). The header (first) line specifies names of 10,031 SNPs and each line below includes number of major alleles (0/1/2) for each SNP for that given individual.
- Phenotypes are in file phenotype.txt, including Identity and phenotypic values for 2,326 animals, comprising 5 generations. Note that 900 genotypes do not have phenotype data and the method is evaluated by comparing the BLUPs for these individuals with their 'true' values.
- True breeding values are in file trueBreedingValue.txt, including Identity and true breeding values for 3,226 individuals.
- Pedigree and gender (M indicates male and F female) for 3,226 individuals are in the file pedigree.txt .

The marker file genotype.mkr looks like

```
ID 1 2 3 ... 10031
1 2 2 0 1 ... 2
...
3226 0 1 2 ... 1
```

The marker values, typically 0, 1 or 2 copies of the minor allele, are stored in **ASReml** as 8bit integers 0, 100, 200. The usual missing value codes (NA and *) are recognised. Data values outside the range [-2,2] are treated as missing values. Any missing values are replaced with the average of the marker values present for that marker.
Let this be represented by the matrix M_I.

For this GBLUP case, a weighting **matrix** **D** = diag(1/s) is formed where $s = \Sigma\, 2p_j(1-p_j)$ and p_j is the proportion of SNP *j* with the minor allele. Then the GRM matrix is formed as **G** = **M D** M ' where M is a column centered version and scaled of $\mathbf{M}_I$. This is a Genomic relationship matrix for the individuals with marker data, and is used to estimate a genomic variance for the individuals (assuming there are more markers than individuals so that **G** is positive definite).

In this model, the weight, s is a matrix scaling factor which relates the marker model variance component (σ^2_γ) to the additive genetic variance (animal model). That is, it scales **G** = **M D M** ' so that it is an empirical measure of the usual pedigree based Numerator Relationship Matrix Additive variance so that the estimated variance component σ^2_γ is on a scale where it can be interpreted as genetic variance analogously to the additive variance estimated under the common animal model.

The basic code then to fit the GBLUP model is

```
!WORKSPACE 1
Analysis of marker data using the standard GBLUP model
 ID *
 phenotype
genotype.mkr !MARKERS 10031 !IDS 3226
phenotype.txt  !SKIP 1   !MAXIT 50   !GDENSE
phenotype  ~ mu       !r grm1(ID)
residual units
```

This code assumes an implicit link between rows of the matrix M and genotype levels coded in data variable ID. That is, the data variable ID contains the numbers 1:3226 (in whatever order) indexing the rows of M .

The output from such a run was

```
 Marker data [0/1/2] for 3226 genotypes and 10031 markers read from ..\genotype.mkr
        Marker values ranged 0 to 2
        Marker Means ranged  0.00 to 2.00
          Sigma2p(1-p) is 3741.94803
 GIV1  ..\genotype.    3226       9   -11901.61
 QUALIFIERS: !SKIP 1   !MAXIT 50  !GDENSE
 QUALIFIER: !DOPART    3 is active
 Reading ..\phenotype.txt  FREE FORMAT skipping     1 lines

 Univariate analysis of phenotype
 Summary of 2326 records retained of 2326 read

  Model term          Size #miss #zero   MinNon0    Mean      MaxNon0  StndDevn
   1 ID               2326     0     0      1  1163.5000       2326
   2 phenotype      Variate    0     0  35.00      68.66      100.8      10.03
   3 mu                          1
   4 grm1(ID)         3226

 Forming     3227 equations:3227 dense.
 Initial updates will be shrunk by factor    0.316
   1 LogL=-6194.12     S2=  71.314       2325 df   0.1000E+00
   2 LogL=-6156.37     S2=  67.648       2325 df   0.1520
   3 LogL=-6114.01     S2=  62.852       2325 df   0.2694
```

```
  4 LogL=-6083.24     S2=  57.549       2325 df   0.5308
  5 LogL=-6077.71     S2=  55.022       2325 df   0.7485
  6 LogL=-6077.52     S2=  54.477       2325 df   0.8076
  7 LogL=-6077.52     S2=  54.470       2325 df   0.8084
Final parameter values                        0.8084

         - - - Results from analysis of phenotype - - -
Akaike Information Criterion    12159.03 (assuming 2 parameters).
Bayesian Information Criterion  12170.53

Source                Model  terms     Gamma     Component    Comp/SE   % C
grm1(ID)               3226   3226  0.808352       44.0308       8.93   0 P
Variance               2326   2325  1.000000       54.4699      29.80   0 P

                                  Wald F statistics
    Source of Variation           NumDF              F-inc
  3 mu                                1             200.15

                     Solution       Standard Error    T-value     T-prev
  3 mu
                   1    70.5679        4.98807         14.15
  4 grm1(ID)                          3226 effects fitted
Finished: 27 May 2013 13:22:35.547   LogL Converged
```

This gives genotype effects directly in the .sln file based on the GRM matrix used. Note that there is data only for ID values 1:2326 but there is marker data for an extra 900 individuals. Thus ID has 2326 levels but grm1(ID) has 3226 levels.

Marker effects are simply derived and are reported in the .mef file. There is a direct link between the marker model

$\mathbf{y} = \mu + \mathbf{M}\ \gamma + \varepsilon$

if var $(\gamma) = I\,\sigma^2_\gamma = I\,\sigma^2_G/s = D\,\sigma^2_G$ where $s = \Sigma\, 2p_j(1-p_j)$ and the genotype model

$y = \mu + u + \varepsilon$

with var$(u) = \mathbf{M}\ \mathbf{D}\ \mathbf{M}'\backslash\, \sigma^2_G$ giving $u = M\ \gamma$ and $\gamma = D\ M'\,(M\ D\ M')^{-1}\ u_g$.

That is, the relationship between the variance of marker effects (assumed equal variance) and the genotype effects is given by $\sigma^2_\gamma = \sigma^2_G/s$.

PEV (γ) is then $G_{mm} + G_{\{mg}\ G^{-1}_{gg}\ \text{PEV}(u\_g)\ G^{-1}_{gg}\ G_{gm}$

$= \sigma^2_G\,(D - D\ M'(M\ D\ M')^{-1}\ M\ D) + D\ M'\,(M\ D\ M')^{-1}\ C^{GG}\,(M\ D\ M')^{-1}\ M\ D)$

where C^{GG} is the block of the C^{-1} corresponding to **u**.

In the data described above, the genomic relationship matrix includes 900 individuals for which there is not data in the phenotype file. The .sln reports genomic breeding values for these individuals predicted on the basis of the marker based genetic correlation with individuals having phenotype.

The following lines extracted from the .mef file report the larger (magnitude) marker effects. The first field is the marker number, the second field is the marker effect, the third field is zero here but would contain the predicted standard error of the marker effect if the !PEV qualifier had been selected and the fourth field is diag(d_j)=1/s=1/3741.948.

```
145  0.8776342E-01  0.00000        0.26724E-03
929  0.103685       0.00000        0.26724E-03
932  0.981438E-01   0.00000        0.26724E-03
937 -0.978502E-01   0.00000        0.26724E-03
939 -0.948803E-01   0.00000        0.26724E-03
952  0.132753       0.00000        0.26724E-03
```

```
 954 -0.136886       0.00000        0.26724E-03
 956 -0.126098       0.00000        0.26724E-03
 957 -0.869884E-01   0.00000        0.26724E-03
 959 -0.130554       0.00000        0.26724E-03
2719  0.7654950E-01  0.000000       0.26724E-03
4480  0.165363       0.00000        0.26724E-03
4481  0.812835E-01   0.00000        0.26724E-03
4485  0.136997       0.00000        0.26724E-03
4491  0.121994       0.00000        0.26724E-03
4496 -0.118146       0.00000        0.26724E-03
5482 -0.991316E-01   0.00000        0.26724E-03
5483  0.963315E-01   0.00000        0.26724E-03
5484  0.743808E-01   0.00000        0.26724E-03
5485  0.997677E-01   0.00000        0.26724E-03
5488 -0.123439       0.00000        0.26724E-03
5492  0.786835E-01   0.00000        0.26724E-03
5494  0.889696E-01   0.00000        0.26724E-03
5495 -0.893046E-01   0.00000        0.26724E-03
5496  0.893046E-01   0.00000        0.26724E-03
```

Note that the large effects are clustered, particularly around markers 954, 4480 and 5488.

Rerunning the model in 2023 in Echidna, the slow steps (given 10031 markers) are forming the GRM matrix (83s), forming the marker PEV matrix (which is now the default, 121s) and writing the .eme matrix (400s). The criterion for identifying large effects is different when PEV is known and Echidna reported more (59) large marker effects. PEV values ranged 0.15 to 0.92 (but 8 were NaN).

It is of interest to compare the GBLUP analysis with the traditional pedigree based animal model. It reports

```
  14 LogL=-6259.64     S2=  47.910       2325 df    1.178
  15 LogL=-6259.64     S2=  47.910       2325 df    1.178
 Final parameter values                        1.178

          - - - Results from analysis of phenotype - - -
 Akaike Information Criterion    12523.29 (assuming 2 parameters).
 Bayesian Information Criterion  12534.79

 Source            Model  terms     Gamma         Sigma   Sigma/SE   % C
 nrm(ID)            3226   3226   1.17797       56.4364       6.08   0 P
 Residual           2326   2325  1.000000       47.9099       9.82   0 P
```

Fast Bayes A like analysis, variable marker variances

It is sometimes of interest to try and identify markers of apparent large effect, assuming such markers are linked with alleles of large effect, commonly referred to as QTL. However, the marker effects from the GBLUP analysis are shrunken estimators, and any QTL effect is likely to be smeared over several near markers such that none stand out. Bayes A is one method of identifying markers of large effect and the method implemented here, called a fast Bayes-A like algorithm, has the same aim. Bayes A algorithms have been shown to generally predict true breeding value better than the simpler GBLUP method just described.

The Fast Bayes A like EM algorithm of Sun et al. (2012) used the genomic variance σ^2_G estimated under the equal variance assumption as a known prior to calculate individual marker variances. Given σ^2_G, the marker variance is σ^2_G/s and the expected value of the scaled inverse Chi-squared distribution is $k\,\sigma^2_G/(k-2)/s$. Therefore, following Meuwissen et al. (2001), Sun used the expression $d_j = (\gamma^{j\wedge 2} + (k-2)\,\sigma^2_G/s)/(k+1)$ where γ_j is the marker effect from the last iteration, σ^2_G/s is the prior estimate of marker variance, σ^2_G is the genetic variance, k is the degrees of freedom (4 default) of the Chi-square distribution assumed for the variances of the marker effects, and s is $\Sigma\, 2\, p_j(1-p_j)$.

The cycle is essentially,

reform the marker weights (D)
reform G = M D M ' ,
solve the mixed model equations to get new estimates of the random effects
repeat to convergence.

The matrix D used by Sun et al. (2012) incorporates the genetic variance \ σ^2_G and so he applies no scaling factor to G when solving the mixed model equations. However, in **ASReml**, the genetic variance is used as a scaling factor for G and factored out of D.
Thus the calculation used in **ASReml** is $d_j = (\gamma^{j\wedge 2} / \sigma^2_G + (k-t) / s)/(k+1)$ having divided through by σ^2_G. In this expression, the offset 2 has been parameterised to t ; k and σ^2_G are parameters the user can control. The offset t is taken as 2 when σ^2_G is known (fixed) but 0 if σ^2_G is being estimated. If the variance parameter σ^2_G is estimated without setting t=0 , REML (assuming normality) attempts to reverse the shrinkage applied to the random effects by increasing the estimate of σ^2_G in a compounding manner, making the weights wrong. However, using $t \neq 2$ means the distribution of marker effects is no longer strictly a t distribution with k degrees of freedom.

The four parameters in this expression have distinct roles.

σ^2_G is the estimated variance of the genotype effects under the GBLUP model and provides the overall scaling for the magnitude of the marker effects,
s is a scaling parameter relating the genetic variance to the marker variance
k (degrees of freedom) controls the shape of the inverse Chi-squared distribution of the marker variances, and
t (=2) causes the distribution of marker variances to conform to the scaled inverse Chi-squared distribution with k degrees of freedom.

In **ASReml**, there is the choice of updating the estimate of the genetic variance, or holding it, or the ratio σ^2_G / σ^2_E constant. However, the updated estimate is obtained under a modified distributional assumption.

The code for running the large test data set follows. In this code, k=4.2 is set using !FBA 42 , and the genetic is fixed at 0.808 times the residual variance, the same ratio as estimated in the GBLUP run. In this code, a pedigree file is used to define the full list of 3226 genotypes.

```
!WORKSPACE 1
Title: phenotype.
 ID  !P           # 4
 phenotype        # 68.51

pedigree.txt !SKIP 1
genotype.mkr !MARKERS 10031  !FBA 42
phenotype.txt  !SKIP 1   !MAXIT 50
phenotype  ~ mu !r grm1(ID)  0.808  !GF
residual units
```

The output produced is

```
...
 ID  !P
 pedigree.txt !SKIP 1
 Reading pedigree file pedigree.txt: skipping            1  lines
 Pedigree Header Line: ID Sire Dam Gender
    3226 identities in the pedigree over 4 generations.
       Assuming first parent is Sire,
       Sires SiresofSire  DamsofSire        Dams  SiresofDam   DamsofDam
          92          42          54         104          46          59
 Using an adapted version of  Meuwissen & Luo GSE 1992 305-313:
```

```
PEDIGREE [pedigree.txt ] has     3226 identities,    9742 Non zero elements

Marker data [0/1/2] for 3226 genotypes and 10031 markers read from ..\genotype.mkr
       Marker values ranged 0 to 2
       Marker Means ranged  0.00 to 2.00
         Sigma2p(1-p) is 3741.94803 !FBA 42 ChiDF   4.20
GIV0   NRM            3226       7    -2228.91
GIV1  ..\genotype.    3226       9    -6894.59
QUALIFIERS: !SKIP 1   !MAXIT 50
QUALIFIER: !DOPART    1 is active
Reading ..\phenotype.txt  FREE FORMAT skipping     1 lines

Univariate analysis of phenotype
Summary of 2326 records retained of 2326 read

 Model term          Size #miss #zero   MinNon0    Mean      MaxNon0  StndDevn
  1 ID          !P   3226     0     0      1       1164       2326
  2 phenotype      Variate    0     0  35.00      68.66      100.8      10.03
  3 mu                          1
  4 grm1(ID)         3226
Forming     3227 equations:   1 dense.
Initial updates will be shrunk by factor    0.316
  1 LogL=-6166.16     S2=  43.998       2325 df   0.8080
  2 LogL=-6064.09     S2=  58.652       2325 df   0.8080
  3 LogL=-6082.05     S2=  59.964       2325 df   0.8080
  4 LogL=-6081.10     S2=  59.892       2325 df   0.8080
  5 LogL=-6074.81     S2=  59.558       2325 df   0.8080
  6 LogL=-6068.91     S2=  59.243       2325 df   0.8080
  7 LogL=-6066.82     S2=  59.139       2325 df   0.8080
  8 LogL=-6066.21     S2=  59.123       2325 df   0.8080
  9 LogL=-6065.51     S2=  59.094       2325 df   0.8080
 10 LogL=-6063.98     S2=  59.017       2325 df   0.8080
 11 LogL=-6060.42     S2=  58.834       2325 df   0.8080
 12 LogL=-6053.31     S2=  58.465       2325 df   0.8080
 13 LogL=-6045.93     S2=  58.080       2325 df   0.8080
 14 LogL=-6043.12     S2=  57.931       2325 df   0.8080
 15 LogL=-6042.61     S2=  57.913       2325 df   0.8080
 16 LogL=-6042.55     S2=  57.916       2325 df   0.8080
 17 LogL=-6042.56     S2=  57.917       2325 df   0.8080
 18 LogL=-6042.56     S2=  57.918       2325 df   0.8080
 19 LogL=-6042.57     S2=  57.918       2325 df   0.8080
 20 LogL=-6042.57     S2=  57.918       2325 df   0.8080
 21 LogL=-6042.58     S2=  57.919       2325 df   0.8080
 22 LogL=-6042.58     S2=  57.919       2325 df   0.8080
 23 LogL=-6042.58     S2=  57.919       2325 df   0.8080
Final parameter values                       0.8080

         - - - Results from analysis of phenotype - - -
Akaike Information Criterion    12089.16 (assuming 2 parameters).
Bayesian Information Criterion  12100.67

Source                Model  terms     Gamma     Component    Comp/SE   % C
grm1(ID)               3226   3226  0.808000       46.7983      34.10   0 F
Variance               2326   2325  1.000000       57.9187      34.10   0 P
Current Sigma 2 Wi Pi (1-Pi) is     3809.780

                                Wald F statistics
    Source of Variation         NumDF              F-inc
  3 mu                              1          0.14E+06

                    Solution       Standard Error    T-value     T-prev
  3 mu
                  1    68.6702       0.181952        377.41
  4 grm1(ID)                          3226 effects fitted
          1  possible outliers: see .res file
Finished: 24 May 2013 18:09:00.473   LogL Converged
```

As well as reporting all the marker effects in the .mef file along with the individual marker weights), **ASReml** reports the dominant markers in the .res file.

```
 954   3.38553       0.00000     ******************
4480  -3.07967       0.00000     ***************
5482  0.854442E-01   0.00000
5483  0.750411E-01   0.00000
5485  0.736952E-01   0.00000
5488  0.125913       0.00000     *
```

These values agree closely with values that Sun calculated (*pers. comm.*).

It is of interest to compare them with the GBLUP values given earlier. We note that the former larger marker effects associated with markers 929-959 are concentrated into one large effect for marker 954, and those associated with markers 4480-4496 are concentrated into one large effect for marker effect for 4480, but there are still a cluster of large marker effects 5482-5488,

Table 5.1 Comparison of models run in 2013 using a range of values for the inverse Chi-square distribution degrees of freedom parameter, k , and fixing or estimating the variance parameter. 'Iter' is the number of iterations performed, Accuracy is the correlation between the BLUP values and true breeding values for the 900 individuals without phenotype. The large markers are those identified as having large effects.

	GBLUP: Genetic variance ratio from marker effects					
k	LogL	Iter	σ^2_G	σ^2_E	Accuracy	Large markers
	-6077.52	7	44.0	54.5	0.611	
	Fast Bayes A: Genetic variance ratio fixed at 0.808					
k	LogL	Iter	σ^2_G	σ^2_E	Accuracy	Large markers
4.2	-6042.58	23	46.8	57.9	0.635	954/4480
3.8	-6008.08	22	42.4	52.5	0.656	952/954/4480/5488
3.5	-5995.95	39	42.0	52.0	0.659	145/952/954/2719/4480/5488
2023	-6032.19	32	44.0	53.9		954/4480
3.2	-6030.88	19	47.8	59.1	0.631	954/4480/5488
2.7	-6048.61	27	49.8	61.6	0.619	954/4480/5488
2.2	-6110.38	21	54.8	67.8	0.587	952/954/4480/5488
	Fast Bayes A: Genetic variance ratio estimated					
k	LogL	Iter	σ^2_G	σ^2_E	Accuracy	Large markers
4.2	-6050.34	30	49.7	53.7	0.636	4480
3.8	-6033.00	31	43.4	53.9	0.645	954/4480
2023	-6046.02	26	47.4	58.7		954/4480
3.5	-6004.31	20	44.6	53.8	0.655	952/954/4480/5488
3.2	-6015.16	44	39.7	53.9	0.648	954/4480/5488
2.7	-6014.11	23	41.0	53.9	0.648	954/4480/5488
2.2	-6013.00	20	43.1	53.9	0.648	954/4480/5488

Table 5.1 shows a summary of a series of runs with *k* set at various values, and either holding the variance ratio fixed as proposed by Sun, or estimating it. When estimating the variance ratio, the initial value was 0.5. The results could be slightly different if the iteration started with a different initial variance ratio because the early marker variances will be calculated slightly differently.

From these and other runs in 2013, first we note that the reported REML Loglikelihood jumps as more large markers are identified (see typical values in table). The number of iterations required for the Fast bayes algorithm varied from 19 to 44 in these runs. In the longer runs, examination of the LogL value changing over iterations shows that the model had almost converged after about 20 iterations, but then identified another marker of large effect and quickly made another substantial increase in LogL. For these models, **ASReml** sets !EXTRA 5 which facilitates this extra gain in LogL. That is, the model which identified 6 large markers nearly stopped after identifying just 4 with a LogL of -6006.02. But this issue may be resolved in the 2023 re-runs, with centering automated and giving results more consistent with the 4.2 results.

Large markers	0	1	2	3	4	5	6
LogL	-6077	-6049	-6033	-6012	-6006		-5996
Accuracy	0.611	0.635	0.645	0.648	0.655	.	0.659

The Accuracy, that is the correlation between true breeding value and BLUP for the 900 individuals without data, also increases with the number of large markers identified.

The 3 models fitted with the *variance ratio* fixed (assumed known at 0.808) and k <= 3.2 are evidently a poor fit as indicated by the high residual and lower Accuracy (correlation). This strongly argues against using values of k<3.5 under this model. This is because the marker variance expression for d_j involves a numerator k-2 . Looking at the 3 larger values of k , we note an increase in the number of markers with 'large' effects: 2 for 4.2, 4 for 3.8 and 6 for 3.5. However the gain in accuracy from 3.8 to 3.5 is quite small. Obviously, other data sets will have different characteristics but using a value for k which is too small will be counterproductive.

When estimating the variance parameter, the accuracy does not fall off as seriously when k<3.5; using k=3.5 is very similar to using k=3.8 when the variance parameter is fixed.

The following table compares marker effects for the dominant markers from four models run in 2013, 2 rerun in 2023. The last three models use the marker variance expression assuming the variance is known. The FBA results were both obtained with the variance ratio set to 0.808, the value obtained from then GBLUP analysis. Sun held the genetic variance fixed at 44.03 (rather than the variance ratio) and used a different (slightly lower) residual variance and so the effects he calculated are not identical to FBA 4.2 even though he used k=4.2. In interesting result here is that markers 952 and 954 are evidently very close neighbours (highly correlated covariates) yet seem to complement each other in the FBA 3.8 model. Note that the sign changes in the following table are due to the !CENTRE qualifier which reverses the marker covariable if its mean is greater than 1.0. The last 2 columns relate to the 2023 reruns.

Marker	GBLUP	FBA 4.2	FBA 3.8	Sun	FBA 4.2 C	FB 3.8 C
952	0.133	-0.040	-2.155	0.0649	0.040	0.036
954	-0.137	3.385	1.890	-3.3872	-3.3855	-3.449
4480	0.165	-3.080	-3.407	3.1400	3.0800	3.089
5488	-0.123	0.126	3.171	-0.1625	-0.1253	-0.035

Limitations of ASReml implementation:

- Only one marker GRM is permitted. If two are needed, one would need to be saved from an earlier run and used directly as a GIV matrix rather than being formed each iteration from the marker variables.
- A marker GRM can only be fitted as a simple term e.g. grm1(ID) in Fast Bayes mode. To incorporate it in an interaction, it would need to be saved and used directly as a known GIV matrix.
- ASReml does not check that the ID in first field of the marker file matches any ID in data unless a label for the ID variable is given on the .grr file line and it matches the name of a factor in the data. The appropriate order can be set by supplying an appropriate list of level names when defining the ID factor.

The genotype effects are reported as usual in the .sln file.

The marker effects (with SE if !PEV specified) and their individual weights (d_j) are reported in the .mef file. The larger marker effects are also reported in the .res file.

Having identified markers of special interest, you may wish to include them in the model as separate covariates. This is done by specifying grr(G,m) (or snp(G,m)) in the model where G is the factor used to associate markers with the data, and m is the marker (position/number) to be fitted. So, for example we could include `grr(ID,954) grr(ID,4480)` as fixed or random terms in the model and re-estimate the remaining marker based genetic variance.

Differences with ASReml implementation

The first difference is that **ASReml** allows the genetic variance σ^2_G to be updated under the REML machinery as the marker effect variances are updated, but using a modified expression for the variances of the marker effects which is less skewed. Thus the expression used for calculating **D** has been rearranged so that the genetic variance is factored out of **D**.

Second, Sun (pers. comm.) reports that the residual variance he obtained was 50.667 (cf **ASReml** value of 57.92) and a genetic variance of 49.993 (calculated as total variance less residual variance). It appears he used a faulty expression, the sum of squares of the residuals **y - Xβ - Zu** rather than (**y** - **Xβ**)'(y - **Xβ** - **Zu**) as the residual Sum of Squares to calculate the residual variance and consequently the genetic variance. However, he always used 44.03 as the genetic variance when calculating **D** although he used his estimate of the residual variance when solving the mixed model equations. It is possible to fit the model on the variance scale in **ASReml** and to obtain the same results if his algorithm is modified to calculate the residual conformably with **ASReml**.

Third, Sun (pers. comm.) added 0.0001 to the diagonal of G to 'avoid singularity'. ASReml (2023) adds 0.00001 to the whole matrix when using CENTERED marker variables.

Fourth, we have modified the calculation of **D** when the variance parameter is not fixed, so that the estimated genetic variance will be consistent, that is it will be estimated at a value in the vicinity of the value estimated under the equal marker variance (GBLUP) model. Having done some runs where the variance is estimated and k-2 is used in calculating **D** , I note that the model fit is almost identical although the genetic variance parameter is increased by k/(k-2) . For example, with k=4.2 and using k-2 , the estimate of the genetic variance becomes 94.7 but the LogL is -6050.1, the residual variance is 53.7 and the marker effects are very close to those using k rather than k-2 in calculating **D** . Note that the ratio 94.7/49.7 is very close to k/(k-2) = 4.2/2.2 so the effect of using

k-2 is simply to rescale the estimated value of the variance parameter. Following are the larger marker effects from the two runs:

```
    Marker       4.2          2.2
       929  0.110551     0.109084
       952 -0.201308     0.195562        *
       954  0.265858    -0.317992        *
       956 -0.163249    -0.161011        *
       959 -0.190636    -0.186029        *
      4480  -3.15816      3.15813        ************
      5482  0.113984    -0.113907
      5485  0.107698     0.107581
      5488  0.173630    -0.173696        *
```

Sun ({pers. comm.) provided the solutions (BLUPS) he had obtained for the large test example; they had a mean of zero. For the GBLUP model (equal weights), the **ASReml** BLUPs were the same except for a mean offset; they were 1.8 lower than his when the markers scores used were -1,0,1; and 1.2 lower when the marker scores were 0,1,2. I do not expect the BLUPs to have a mean of zero when they are correlated as these are. Nevertheless, this offset will modify the calculation of variances slightly.

Marker Prediction error

A comparison of fitting markers directly and indirectly in a smaller example resulted in the same LogL and marker BLUPs. I have attempted to calculate SE of marker effects.

```
   6 LogL= 14.2984     S2= 0.43999E-01    199 df   0.5525E-01

 Source                Model  terms     Gamma     Component    Comp/SE   % C
 Markers                 260    260  0.552491E-01  0.243091E-02   6.99   0 P
 Variance                200    199   1.00000      0.439991E-01   3.20   0 P

  Model_Term                Level               Effect    seEffect
  mu                            1                7.769      0.2802E-01
  Markers                       1               -0.3006E-01  0.3072E-01
  Markers                       2               -0.3812E-01  0.3203E-01
  Markers                       3                0.6608E-02  0.3355E-01
```

cf

```
   2 LogL= 14.2984     S2= 0.43997E-01    199 df    7.157

 Source                Model  terms     Gamma     Component    Comp/SE   % C
 grm1(entry)             200    200   7.15720      0.314897       6.98   0 P
 Variance                200    199   1.00000      0.439972E-01   3.20   0 P

Marker_name             Effect        seEffect  Weighting_in_G
         1        -0.3005719E-01   0.3072172E-01   0.7719340E-02
         2        -0.3811851E-01   0.3203082E-01   0.7719340E-02
         3         0.6607372E-02   0.3354752E-01   0.7719340E-02
```

The difference in the reported marker variance Component is a factor of 129.5, the weight $\Sigma\, 2p_j(1-p_j)$ used in the latter method.

Discussion

This chapter demonstrates the implementation of a Bayes A like method in ASReml. The implementation involves estimating genetic effects directly in the genotype space using a **G** matrix formed from the marker covariables, and predicting the marker effects. The predicted marker

effects are used to predict individual marker variances which are used as weights when reforming the **G** matrix. The method is essentially that of Sun et al. (2012). With consistent parameter settings, **ASReml** produces the same genotype and marker effects as the algorithm used by Sun.

ASReml extends the method of Sun et al. (2012) by allowing the genetic variance to be estimated subject to an adjustment to the calculation of **D**.

ASReml does not provide any mechanism for estimating the degrees of freedom parameter k yet this is critical to the shape of the distribution of relative marker effect variances and hence the number of markers of large effect detected. For the simulated data used here, a value in the range 3.5 to 3.8 seems optimum. A value less than 3.5 is likely to provide too much shrinkage, whether or not the variance parameter is known. It seems k = 4 is a reasonable value (Sun used 4.2). Larger values of k quickly move to results similar to GBLUP (Normal).

The number of markers with large effects identified varied between runs depending critically on the value of k. In the test data set, it is evident that contiguous marker covariables are correlated so that under the GBLUP model, any nearby QTL effect will be smeared over those markers. The FBA model attempts to focus the effect on a particular marker. Thus the six markers identified in the k=3.5 model all appeared with relatively large effect in the GBLUP model, the dominant 3 surrounded by other markers of large effect in that model.

For example, one run with k=4.2 and estimating σ^2_G at 49.67 showed large marker effects:

```
       929  0.110551       0.00000
       952 -0.201308       0.00000     *
       954  0.265858       0.00000     *
       956 -0.163249       0.00000     *
       959 -0.190636       0.00000     *
      4480  -3.15816       0.00000     ************
      5482  0.113984       0.00000
      5485  0.107698       0.00000
      5488  0.173630       0.00000     *
```

whereas a run with k=3.2 and estimating σ^2_G at 39.68 showed large marker effects:

```
       954   3.23386       0.00000     **************
      4480  -3.23281       0.00000     **************
      5488   3.00184       0.00000     *************
```

It has been observed that each marker identified as having a large effect produces a jump in the reported Log-Likelihood value. However, there may be several iterations where the Log-Likelihood hardly changes before the jump is made.

It is further observed that fitting markers 954, 4480 and 5488 as fixed effects, and rerunning the FBA model with k=4 did not identify any more markers of large effect, but reduced the genetic variance component to 40.82.

The number of large effects detected does not appear particularly sensitive to the genetic variance being over or understated. The following table compares the large effects identified with k=4.2 and the variance ratio set at 1.2, 0.8 and 0.4. Obviously the general effects are larger when the variance is larger, and evidently the residual is smaller in this example.

```
 Variance ratio   1.2          0.8          0.4
LogLikelihood  -6034.40     -6042.6      -6075.7
Residual        55.4         57.9         62.7
Large marker effects
       954      3.23788      3.38553      3.59661      ******************
      4480     -3.12854     -3.07967     -3.00805      ***************
      5482     0.098363     0.085444     0.063958
```

```
      5483     0.087718      0.075041       0.055536
      5485     0.087668      0.073695       0.052891
      5488     0.147609      0.125913       0.086991       *
      5512    -0.083052     -0.073114      -0.054998
```

There is the question of objective. One possible objective is simply to form a genetic relationship matrix that is efficient and based on SNP markers. Another is to identify putative QTL.

FBL part 15; Gamma scale GBLUP

```
  11 LogL=-6002.86     S2=  54.166       2321 df
grm1(ID)               3226   3226  0.486422       26.3474       7.64   0 P

    Source                NDF      F-inc    F-con
  3 mu                      1     322.19   308.96 .
  4 snp(ID,952)             1      55.84    12.80 A
  5 snp(ID,954)             1      14.42    12.90 A
  6 snp(ID,4480)            1      60.69    61.51 A
  7 snp(ID,5488)            1      46.57    46.57 A
```

So there is still a large amount of genetic variance, but these 4 markers explain 17.68 (40%) of the original 44.03 variance component. Note that 952 and 954 largely substitute for each other, but both are significant.

Timing issues

The run time for these jobs is about 180 s per iteration on an HP EliteBook 8540w with 16 Gb RAM and 8 processors; processor speed 1.87 Ghz. The major components of this are 70s to form G = MDM ' , 25s to invert G and 60s to invert C as part of the REML iteration. Consequently, the GBLUP model runs much faster because G is constructed and inverted only once in the run rather than in each iteration as is required for the Fast Bayes like method. ASReml 4.2 runs the last 2 steps much faster.

ASReml uses link list matrix methods for processing the sparse equations and has always included random model terms in these sparse equations. However, in this instance, the GRM matrix is dense, and it would be more efficient generally to process them as such. The !GDENSE qualifier (set just before the model line) facilitates this. If !GDENSE is set and the first random term is a GRM term, its equations will be processed as DENSE. For this example with 3226 rows on the GRM matrix, this reduced the iteration time from 196 to 175 seconds. In ASReml 4.2 and Echidna, the matrix processing has been greatly speeded up and the iteration time is now 50s, mainly in forming the GRM inverse.

Conclusion concerning Fast Bayes A

As shown by Sun et al. (2012) , the Bayes A like procedure described here is effective at producing a better representation of the Genomic relationship matrix than the standard GBLUP method which gives equal weight to all markers. Further, it is able to identify markers of large effect.

Based on the experience with the data set used here, the number of large marker effects (putative QTL locations) detected is related to the degrees of freedom parameter. Using a value less than 3.5 was unhelpful in this data, and is not encouraged.

If the variance parameter is unknown, **ASReml** can estimate it under a slightly different model and at a greatly increased computational cost per iteration compared to obtaining the estimate from the standard GBLUP method.

Multiple Relationship Matrices

Echidna has an mrm() model function which facilitates fitting multiple conformable GRM matrices in a univariate model.

mrmk(.) specifies the relationship matrix which is a sum of other relationship matrices. The matrices must be conformable. *k* selects the components. For example '12i' would indicate the sum of GRM1, GRM2 and an Identity, and so would fit 3 components. The test job fitted equivalent models:

```
!PART 6
Ablue !WT Ywt !DISP 1 ~ mu Env !r giv1(Hyb) giv2(Hyb) ide(Hyb) +
   idv(Env).giv1(Hyb) idv(Env).giv2(Hyb) idv(Env).ide(Hyb)
!PART 66
Ablue !WT Ywt !DISP 1 ~ mu Env !r mrm12i(Hyb)
                        id(Env).mrm12i(Hyb)
```

with 65 levels of Env and 1919 Hybrids; Part 6 takes 30m per iteration, part 66 takes 7.

!MRM should be specified if the GRM matrix is to be used as part of an mrm() variance function model. Normally, if you supply a GRM matrix, Echidna will invert the matrix and in the process calculate the LogDeterminant. In the mrm() case this inverse and LogDet are not required.

!MRM should be specified if the GRM matrix is to be used as part of an mrm() variance function model. Normally, if you supply a GRM matrix, Echidna will invert the matrix and in the process calculate the LogDeterminant. In the mrm() case this inverse and LogDet are not required.

SVD transformation of GRM model

GTDATA introduction

The GTDATA directive, described in this chapter, is designed to obtain the eigen values (D) and eigen vectors (U) of a dense GRM file (XXX.[b]grm),
write them to files (XXX_D.bgrm and XXX_U.bgrm) respectively
and transform a data file (IFY.csv) by premultiplying factors and variates by U.
A common genomic model can be run faster on the transformed scale. For example, the common genomic animal model

```
!WORK 6 !REN 2 !ARG imf A22
10K bivariate data set $1 $2
ID !A !LL20 !L data.csv !LSKIP 1
CG *  CGs *
imf  sf5
$2.bgiv
```

data.csv !skip 1 !GDENSE
$1 ~ mu CG !r grm1(ID)
is fitted with the equivalent model

!WORK 6 !REN 2 !ARG imf A22
10K bivariate data set $1 $2 Transformed data BGRM matrices

GTdata A22 data.csv !SKIP 1 IF-VV

ID !A !LL20 !L data.csv !LSKIP 1
imf sf5
CG !G 376

$2_D.bgrm
data_IF-VV_$2.bin
$1 ~ tCG !r grm1(ID)

GTDATA syntax

The GTDATA statement must be placed after the TITLE line and before the first variable definition line.
It is processed immediately and so it may be the last line in the job file.
If the 3 files it seeks to create already exist, ASReml assumes that the statement has already been processed.
The statement has 3 arguments and 2 qualifiers.
The full command is:
GTDATA *GRM_basename* [!ADD *offset*] *datafile*] [!SKIP *lines*] *IFYcodestring*
GRM_basename is the name (without file extension) of the GRM file. GRM files are discussed here. The expected (allowed) file extenstions are .grm or .bgrm. The .grm file is an ASCII file with a line for each cell of the lower triangle matrix in the form *row column value*. This form is slow to read and not well suited to a large dense matrix. The .bgrm form is a REAL BINARY form with a record for each line; row *i* contains the values of cells *1:i* . If both forms exist, the .bgrm file is read. If only the .grm file is present, a .bgrm file is created from it.
!ADD *offset* is optional. If specified, it adds *offset* to the diagonal elements of the GRM in the expectation that this will make the matrix positive definite. It is not necessary the matrix be positive definite.
ASReml uses the MKL SVPED routine to factorize the (symmetric) GRM matrix as UDU'. Equivalently, D = U'GU.
datafile] [!SKIP *lines*]
supplies the name of the data file and whether it has a header line to be skipped. The usual form is a .csv file in which case parsing of the data file is simplistic. It may be a binary (.bin) file (created by ASReml with the !SAVE qualifier).
IFYcodestring is a string made up of the characters I, F, V, - with a letter for each field in the data. I is identifier and is recoded 1:N, F is a factor coded 1:t which is expanded to a design matrix (t columns) and premultiplied by U, - is a variable which is ignored, V is a variate which is premultiplied by U. The output file is binary. It contains I, UV and UF and its name is built from the 3 primary arguments: e.g. data_IF-VV_A22.bin. Since the factors in the file are not necessarily in the order if the original data, a template command file is written with the same name as the binary data file to help.
Discussion
This approach only works when the data file has the same (or fewer) rows as the GRM matrix and the number of other effects in the model is substantially less than the number of genotypes.

It will save considerable time when there are many response variates to analyse, especially for bivariate analyses.
Comparing the two models given above, with 9688 genotypes,

the convertional analysis took 9s (invert GRM) + 8*43s (for 8 iterations)
the overhead of the SVD factorization was 160 secs and 8s (for transforming the data)
the transformed analysis took 8*2 sec for 8 iterations.
A conventialal bivariate analysis took 8*313 sec
The transformed bivariate analysis took 8*13 sec

The fitted effects between the two models agree except for the genomic BLUPs. The BLUPs from the conventional BLUP (u) are calculated as u=U'a where a are the BLUPs from the transformed analysis. ASReml does not calculate them at present.

H inverse

The **H** matrix is a particular form of a **G** matrix obtained by merging an **A** matrix (numerator relationship matrix) with a G (Genomic relationship matrix) pertaining to a subset of the genotypes in **A**.

If the **H** matrix has been formed outside of ASReml, then it can be used as a GRM matrix is used.

From August 2022, ASReml 4.2 can form the H matrix. The process is to first specify the pedigree file from which the A matrix is formed, then to specify the G matrix with the !HINV qualifier.

!HINV <GRM_ID_file.txt> [!Hskip h] which creates a **H** inverse (see equation below) from the pedigree based **A** inverse and the **G** inverse defined on this line.
< GRM_ID_file.txt> is a file containing the list of genotype identifiers for the G matrix which must be a subset of the pedigree file identifiers.

Use !Hskip h to skip header lines in < GRM_ID_file.txt>

!OMEGA ω and !TAU τ specify coefficients used in computing the H inverse:

$$\mathbf{H}^{-1}_{\tau,\omega} := \mathbf{A}^{-1} + \begin{pmatrix} 0 & 0 \\ 0 & (\tau \mathbf{G}^{-1} - \omega \mathbf{A}^{-1}_{22}) \end{pmatrix}. \qquad (3)$$

ASReml saves the H matrix as a binary file (filename given in the output) which can subsequently be used directly (saving the setup time in the subsequent runs).

Binary sparse G inverse (.sgiv) layout (October 2022)

In developing a sparse inverse GRM matrix for Mila, I desired a simple binary layout to read the inverse into ASReml. The usual way of reading a sparse G inverse matrix is through a .giv file which presents the non-zero values in lower triangle row-wise order. A .giv file has file extension .giv and has a line for each non-zero cell. Each line contains the row rumber (1:N), the column number (indexed 1:R) and the cell value.

For a large matrix, this is slow to read in. A binary .sgiv form has now been defined in which there is 1 record for each row of the matrix. The first record contains

G11, Ldet, NGroups,Nrow, 77

where G11 is the (1,1) cell value as a real (32bit) value, Ldet is the LogDeterminant of the matrix as a real (32bit) value, NGroups is a 32bit integer specification of the number of fixed DF associated with the matrix,Nrow is a 32bit integer specification of the rows in the matrix, and 77 is a 32bit integer specification of the 77 which is a code indicating this particular binary layout.

Subsequent records (one per row of the matrix) are written and read with the fortran statements

READ/WRITE() Ni, (COL(I),VAL(I),I=1,Ni) where Ni is the number of values to be read, COL is the column number for each non-zero value in the row. COL(N1) will be the Row number and so the corresponding VAL(Ni) is the diagonal element. The diagonal element must be present for each row.

Having developed this as an input form, I have added an !SGIV qualifier to the Pedigree line which writes the inverted A matrix in this binary form. The !SGIV qualifier should be associated with a !DIAG qualifier which writes and .aif file needed to specify the order of identifiers in the .sgiv file.

Note that, to read back the Ainverse as a Ginverse, several changes will be required. For example, if the original job (say PED.as) included lines

```
Animal !P
…
Pedigree.csv   !SGIV !DIAG
…
Y ~ … !r nrm(Animal)
```

Copy it as say GIV.as and change it to say

```
Animal !A !L  PED.aif
…
#Pedigree.csv   !SGIV !DIAG
PED_A.sgiv
…
#Y ~ … !r nrm(Animal)
Y ~ … !r grm1(Animal)
```

Very Large GRM

The marker based GRM matrix is dense and is difficult to work with when large. Its maximum size is 66,000 genotypes in ASReml for a simple univariate model using the !DENSE qualifier, which indexes is cells using 32bit integer addressing. The maximum is 55,000 without the !DENSE qualifier.

Two clients wanted to analyze UK BioBank data. One has data on 62 traits for 276470 subjects with Genomic data. He proposed using a sparse G inverse but the inverse he supplied, while very spare, was not positive definite. Working with such a huge initial matrix is not easy.

This is human data and it turned out that there are few genetically close individuals. Analyses using the initial G inverse found genetic variance in 16 traits. Reforming the inverse from scratch, there was genetic variance in 25 traits.

I then attempted to form an inverse. The premise is that small genetic covariances make negligible contribution. The latest runs ignored correlations less then 0.02 and then identified groups of related individuals. Many individuals were not related at all to others in the data set using a 0.02 threshold and the others were in in very small groups.

This a sparse inverse is formed by ignoring all correlations between members of different groups.

A further idea is built on the observation that a pedigree based inverse is very sparse having main connections between offspring and parents. These then propagate through the population in the relationship matrix. I therefore proposed using an ANTE-DEPENDENCE 1 inverse after ordering members of groups putting the highest covariances on the first off diagonal.

Using these inverses, we obtained genetic variance in 35 of the 62 traits, and then were able to quickly perform pairwise analyses among these traits to estimate genetic correlations.

Contact the author for further details. The steps involved in forming the sparse inverse have not been incorporated into ASReml/Echidna although could be in the fashion of the GTdata procedure discussed above.

Summarising interactions so far:

- He supplied an initial sparce inverse which was not positive definite. ASReml was able to fit univariate and some bivariate models using this matrix. ASReml struggled to fit models with this inverse because the AI matrix was not positive definite! The distribution of values in the Ginverse matrix did not seem reasonable. Univariate analyses showed genetic variance in 16? Traits.

Whole block inversion (January 2023)

My program ANTE7,f90 which formed the Antedepence 1 inverse was adapted to form the normal inverse for each block. Running this using the 0.03 covariance data gave a slightly larger G inverse file and a very similar set of traits with genetic variance. (16, 17 dropped out, 36, 46, and 53 added) using Z-ratio>0.8 as a criterion). I then reran the analyses using covariances greater than 0.02. I was not able to repeat it using covariance greater than 0.01 because that file was too big.

```
01/09/2022  03:14 PM       502,906,776 GRM01.bin
18/01/2023  11:45 AM        24,531,464 GRM02.bin
05/09/2022  02:30 PM        11,434,136 GRM03.bin

The inverses used were
19/01/2023  08:46 AM         6,321,584 grm03_A.sgiv
19/01/2023  08:45 AM         8,084,400 grm03_B.sgiv
19/01/2023  08:46 AM         6,576,384 grm02_A.sgiv
19/01/2023  08:45 AM         9,820,232 grm02_B.sgiv
where A is the Antedependence 1 inverse and B is the Full block inverse.
```

For a given input matrix, the A and B inverse have the same blocking structure. The A1 inverse just has 1 off-diagonal connection with others in the block. The B inverse as all connections within the block.

The results from these for inverses are not materially different. Full details are in FBGI.rtf.

Although grm02_B.sgiv is 50% bigger than grm02_A.sgiv, bivariate analyses using B took only 33% more time (15.8s vs 12 s for 7 iterations).

Using Antedependence inverse and covariances > 0.02

Heritability values where variance component Zratio > 0.8

A020	0.024	0.026	0.025	0.018	0.027	0.038	NS	0.032	NS
0.047	0.065	NS	0.043	0.052	NS	NS	NS	NS	NS
0.121	NS	NS	0.082	0.166	0.031	0.054	NS	0.133	0.113
0.058	NS	0.031	0.029	NS	0.038	NS	NS	0.022	NS
NS	NS	0.040	0.031	NS	0.016	NS	NS	NS	0.030
0.088	0.030	NS	NS	0.041	0.034	0.069	NS	0.063	NS
0.118	NS	0.031							

List of 35 traits: (5 and 39 added to the B03 list, 19 and 53 dropped)

2 3 4 5 6 7 9 11 12 14 15 21 24 25 : 27 29 : 31 33 34 36 39 43 44 46 50 : 52 55 56 57 59 61 63

Using the B inverse, 39 dropped out.

Future work

A correlation model approach

I understand markers are not necessarily ordered, and are arbitrary in orientation. If it were assumed they had been ordered and aligned, one could calculate the lag 1 correlation and get an average absolute value. This could then be used to predict a putative QTL for each marker position as done by Gilmour (2007) calculating the weights assuming equal spacing and spanning say 7-15 markers, aligning the marker covariables to have positive lag 1 correlation, and using the average correlation as the basis for the weights. This would generate a profile from which peaks could be identified, but not assuming an F2/Backcross context. The implicit assumption is however that some QTL effect has been smeared over nearby markers because of the shrinkage and that the information could be recovered.

Distance matrix

ASReml can presently handle a distance based covariance structure in one or two dimensions. Typical syntax is gau(fac(X)) where X is a variate, fac(X) identifies the unique levels of X and codes them in the design matrix as a factor, and stores the values of X for each level of the factor

for use by the gau() function to create distances between points and thence a correlation matrix bassed on the distances, and the current parameter value.

To extend this, we could add a !DISTANCE qualifier to the .grr file line, requesting **ASReml** create and store a distance matrix for the genotypes based on the marker variables. There is an issue of scaling the distance matrix so the correlation parameter is moderate in size. This would be stored as a dense matrix, in the same fashion as giv matrices. Then the say gaussian model could be fitted based on the genotypes with syntax like gauv(dis(Geno,2)) . Assuming the distance matrix was held in the same structure as GRM matrices, the first structure would hold the GIV matrix and the second the DIS matrix, hence the 2 in gauv(dis(Geno,2)) .

Bayes B like approach
I suppose this approach can be extended to modifying the smallest marker variances assigning them a value of zero. It may then be necessary to add a small constant to G before inverting to ensure it is positive definite (when the number of markers included in it drops below the number of individuals present).

A qualifier was added to specify the percentage of markers permitted to go to zero: (!FBB *p*). The default for *p* is 50 and the maximum is 80, the minimum is 1. Use !FBA to set the parameter k .

The procedure is to initialize δ at a very small number, and double delta each iteration until the percentage of markers fixed at zero exceeds *p* . Relative individual marker variances less than the minimum value ± δ are fixed to zero.

The reported LogL increased to `LogL=-5998.43` compared to `LogL=-6042.58 S2=57.919` for the corresponding Bayes A model. The large marker effects were very similar but the correlation of the animal BLUPs for the Validation set reduced from 0.0635 to 0.0633.

Acknowledgements

I thank Julian Taylor for reviewing an early version of this document, Robin Thompson for helful and enlightening discussion helping to clarify what Sun was actually doing, and Xiaochen Sun for making his work available prepublication.

References

Gilmour, A. R. (2007). Mixed model regression mapping for qtl detection in experimental crosses. Computational Statistics and Data Analysis 51, 3749–3764.

Gilmour, A. R., Gogel, B. J., Cullis, B. R., & Thompson, R. (2006). Asreml user guide release 2.0. Reference manual, VSN International, Hemel Hempstead, HP1 1ES, UK, www.vsni.co.uk.

Gilmour, A. R., Thompson, R., & Cullis, B. R. (1995). AI, an efficient algorithm for REML estimation in linear mixed models. Biometrics 51, 1440–1450.

Horvat & Medrano (1995). Genetics 139, 1737–1748.

Meuwissen, T., Hayes, B., & Goddard, M. (2001). Prediction of total genetic value using genome-wide dense marker maps. Genetics 157, 1819–1929.

Resende, M.F.R., Munoz, P. Resende, M.D.V., Garrick, D.J., Fernando, R.L., Davis, J.M., Jokela, E.J., Martin, T.A., Peter, G.F., and Kirst, M. 2012. Accuracy of genomic selection methods in a standard data set of loblolly pine (*Pinus taeda* L.). Genetics 190:1503-1510.

Sun, X., Qu, L., Garrick, D. J., Dekkers, J. C. M., & Fernando, R. L. (2012). A fast em algorithm for bayesa-like prediction of genomic breeding values. PLoS ONE 7(11), e49157.

Szydlowski, M. & Paczynska, P. (2011). Qtlmas 2010: simulated dataset. BMC Proceedings http://biomedcentral.com/1753-6561/5/S3/S3 5, S3.

Verbyla, A. P., Taylor, J. D., & Verbyla, K. L. (2012). Rwgaim: An efficient high dimensional random whole genome average (qtl) interval mapping approach. Genetic Research

Vitezica, Z. G., A. Legarra, M. A. Toro, and L. Varona. 2017. Orthogonal estimates of variances for additive, dominance, and epistatic effects in populations. Genetics 206(3):1297-1307. doi: 10.1534/genetics.116.199406