دانشگاه صنعتی اصفهان

DMU

A package for analysing multivariate mixed models

$$\begin{bmatrix} X_1'\varepsilon^{11}X_1 & X_1'\varepsilon^{12}X_2 & X_1'\varepsilon^{11}Z_1 & X_1'\varepsilon^{12}Z_2 \\ X_2'\varepsilon^{21}X_1 & X_2'\varepsilon^{22}X_2 & X_2'\varepsilon^{21}Z_1 & X_2'\varepsilon^{22}Z_2 \\ Z_1'\varepsilon^{11}X_1 & Z_1'\varepsilon^{12}X_2 & Z_1'\varepsilon^{11}Z_1+c^{11}A^{-1} & Z_1'\varepsilon^{12}Z_2+c^{12}A^{-1} \\ Z_2'\varepsilon^{21}X_1 & Z_2'\varepsilon^{22}X_2 & Z_2'\varepsilon^{21}Z_1+c^{21}A^{-1} & Z_2'\varepsilon^{22}Z_2+c^{22}A^{-1} \end{bmatrix} \cdot \begin{bmatrix} b_1 \\ b_2 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} X_1'(\varepsilon^{11}y_1+\varepsilon^{12}y_2) \\ X_2'(\varepsilon^{21}y_1+\varepsilon^{22}y_2) \\ Z_1'(\varepsilon^{11}y_1+\varepsilon^{12}y_2) \\ Z_2'(\varepsilon^{21}y_1+\varepsilon^{22}y_2) \end{bmatrix}$$

Out line

- DMU in the past
- Current status
- Work in progress

DMU history (1)

Development of the DMU-package started in the late 80's

The first set of programs was for Derivative Free REML estimation of (co)-variance components.

The name DMU originates from this first set of programs

D (derivative free)

MU (multivariate)

DMU history (2)

In the beginning of the 90's new modules were added:

BLUP:
- dmu4: in core solver
- dmu5: iteration on data

Average information REML
- dmuai

DMU history (3)

Code written in FORTRAN 77

Drawbacks:

Dimensions of arrays are static
i.e. must be set at compilation
⟶ need for several compiled versions of each module

User interface: Complicated - long learning curve

DMU history (4)

In the late 90's large part of the original code were converted to FORTRAN 90

Advantages: Array dimensions can be dynamic

- The dimensions can be set/changed during execution

⟶ only one set of executables

Many new facilities were introduced

DMU version 6

In 2000 DMU version 6 was released.

All "own" code now in FORTRAN 90/95

Major new facility:

- Random regressions models

- User interface: Improved

DMUv6 Status

Modules:

- dmu1 : Prepare program
- dmuai : AI-REML estimation of (co)-variance components
- dmu4 : BLUE and BLUP in core
- dmu5 : BLUE and BLUP iteration on data
- rjmc : Bayesian analysis of linear and binary traits

User interface

- Flexible "directive file"
- Data and pedigree in separate files
- Data and pedigree files can be in ASCII or BINARY format
- Names for variables can be specified
- Variable names used in the listing file
- "Simple statistics" for depended variables and co-variables

DMUAI

REML estimation of (co)-variance components

Using either:
- Average Information (AI)
- Expectation Maximization (EM)
- Combined AI-EM

Can handle single and multi trait models with:
- Random regressions
- Direct and maternal effects
- Sire-Dam/MGS models
- Reduced rank models

Mixed linear model

$$\mathbf{y} = \mathbf{X}\beta + \sum_{i=1}^{r} \mathbf{Z}_i \mathbf{u}_i + \mathbf{Z}_a \mathbf{a} + \mathbf{e}$$

where:

y	= vector of observations
β	= vector of fixed effects
u_i and a	= vectors of random effects
e	= vector of random residuals
X, Z_1,, Z_r and Z_a	= known design matrices

Assumptions

$E[u_i] = 0\ (\forall\ i)$, $V[u_i] = G_i = G_{0i} \otimes I\ (\forall\ i)$

$E[a] = 0$, $V[a] = G_a = G_{0a} \otimes A$,

$E[e] = 0$, $V[e] = R = R_0 \otimes I$,

$Cov[u_i, u'_j] = 0\ (i \neq j)$, $Cov[a, u'_i] = 0\ (\forall\ i)$,

$Cov[u_i, e'] = 0\ (\forall\ i)$, $Cov[a, e'] = 0$

Notation

Let: $\mathbf{G} = \bigoplus_{i=1}^{r} \mathbf{G}_i \oplus \mathbf{G}_a$

$$\mathbf{Z} = [\mathbf{Z}_1 \;\vdots\; \mathbf{Z}_2 \;\vdots\; \cdots \;\vdots\; \mathbf{Z}_r \;\vdots\; \mathbf{Z}_a]$$

$$\mathbf{u}' = [\mathbf{u}_1' \;\vdots\; \mathbf{u}_2' \;\vdots\; \cdots \;\vdots\; \mathbf{u}_r' \;\vdots\; \mathbf{a}']$$

then: $Var[\mathbf{y}] = \mathbf{V} = \mathbf{ZGZ}' + \mathbf{R}$

Henderson's MME

$$\begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Z} + \mathbf{G}^{-1} \end{bmatrix} \begin{bmatrix} \hat{\beta} \\ \hat{\mathbf{u}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{R}^{-1}\mathbf{y} \\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{y} \end{bmatrix}$$

Note the resembles with the normal equations for a LS-problem

(co)variance matrices

$$\mathbf{G} = \bigoplus_{i}^{r} \mathbf{G}_i \oplus \mathbf{G}_a \Rightarrow \mathbf{G}^{-1} = \bigoplus_{i}^{r} \mathbf{G}_i^{-1} \oplus \mathbf{G}_a^{-1}$$

$$\mathbf{G}_i = \mathbf{G}_{0_i} \otimes \mathbf{I} \qquad \Rightarrow \mathbf{G}_i^{-1} = \mathbf{G}_{0_i}^{-1} \otimes \mathbf{I}$$

$$\mathbf{G}_a = \mathbf{G}_{0a} \otimes \mathbf{A} \qquad \Rightarrow \mathbf{G}_a^{-1} = \mathbf{G}_{0a}^{-1} \otimes \mathbf{A}^{-1}$$

$$\mathbf{R} = \mathbf{R}_0 \otimes \mathbf{I} \qquad \Rightarrow \mathbf{R}^{-1} = \mathbf{R}_0^{-1} \otimes \mathbf{I}$$

Useful relations

Let: $\mathbf{P} = \mathbf{V}^{-1} - \mathbf{V}^{-1}\mathbf{X}(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-1}\mathbf{X}'\mathbf{V}^{-1}$

Then:
$$\begin{aligned} \mathbf{Py} &= \mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}\hat{\beta}) \\ &= \mathbf{R}^{-1}(\mathbf{y} - \mathbf{X}\hat{\beta} - \mathbf{Z}\hat{\mathbf{u}}) \\ &= \mathbf{R}^{-1}\hat{\mathbf{e}} \end{aligned}$$

Restricted log-likelihood

Minimize Likelihood functions

Method:

- Derivative-free method
- EM algorithm (based on 1. derivatives)
- Algorithm based on 1. and 2. derivatives (Newton-Raphson, Fisher-scoring, AI-REML)

The i'th element in the vector of 1. derivatives

$$\frac{\partial L(\Theta)}{\partial \Theta_i} = \mathrm{tr}\left[\frac{\partial \mathbf{V}}{\partial \Theta_i}\mathbf{P}\right] - \mathbf{y}'\mathbf{P}\left[\frac{\partial \mathbf{V}}{\partial \Theta_i}\right]\mathbf{Py}$$

Can be calculated as a function of:
û, ê, C (the coefficient matrix),
a spares inverse of C,
and the residual (co)variance matrix R

The ij'th element of the observed information matrix I_O

$$\frac{\partial^2 L(\Theta)}{\partial\Theta_i \partial\Theta_j} = -tr\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right] + 2\mathbf{y}'\mathbf{P}\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right]\mathbf{P}\mathbf{y}$$

The observed information matrix is used in the Newton-Raphson algorithm

The ij'th element of the expected information matrix I_E

$$E\left[\frac{\partial^2 L(\Theta)}{\partial\Theta_i \partial\Theta_j}\right] = tr\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right]$$

The expected information matrix is used in the Fisher-scoring algorithm

Computation problems

$$tr\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right]$$

Computation of this trace is most practical applications very time consuming

AI-REML

Asymptotic $I_o = I_E$

$$\mathbf{I}_A = \frac{1}{2}\left(\frac{\partial^2 L(\Theta)}{\partial\Theta_i \partial\Theta_j} + E\left[\frac{\partial^2 L(\Theta)}{\partial\Theta_i \partial\Theta_j}\right]\right)$$

$$= \frac{1}{2}\left[-tr\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right] + 2\mathbf{y}'\mathbf{P}\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right]\mathbf{P}\mathbf{y} + tr\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right]\right]$$

$$= \mathbf{y}'\mathbf{P}\left[\frac{\partial \mathbf{V}}{\partial\Theta_i}\mathbf{P}\frac{\partial \mathbf{V}}{\partial\Theta_j}\mathbf{P}\right]\mathbf{P}\mathbf{y}$$

Can be calculated as a function of:
û , **ê**, **C**, and **R**

Updating Θ

$$\hat{\Theta}^{i+1} = \hat{\Theta}^{i} - \mathbf{I}_\mathbf{A}^{-1}\frac{\partial L(\Theta)}{\partial\Theta}$$

This is an iterative process based on:

- Building and solving MME
- Computing the vector of 1. derivatives and I_A
- Updating the parameter vector Θ

The process is stopped when:

Comparison of methods

Rank of MME: 46581
NZ in MME: 564634
Parameters i Θ: 12

	DF	EM	AI
Rounds	1435	1000	6
Time (sec)	54896	219127	2068
-2ln(L)	16.850	15.158	15.156

Limitations of AI-REML

Memory for: The coefficient matrix (C)

Work vector for factorization of C

The factor of C

A vector for the solution to MME

Vectors for solutions for each parameter

Numerical: Sufficient accuracy for calculations of the factor and the sparse inverse of C

DMU4

BLUE and BLUP in core

Using either:
- Iterative solver (ITPACK)
- Direct solver (FSPAK)
- Parallel direct solver on SMP workstations (Intel® Math Kernel Library)

Can handle single and multi trait models with:
- Random regressions
- Direct and maternal effects
- Sire-Dam/MGS models
- Reduced rank models
- Experimental GLMM

Iterative solvers in dmu4

ITPACK solvers:

Jacobi Conjugate Gradient (JCG)
Jacobi Semi-Iteration (JSI)
Successive Overrelaxation (SOR)
Symmetric SOR Conjugate Gradient (SSORCG)
Symmetric SOR Semi-Iteration (SSORSI)
Reduced System Conjugate Gradient (RSCG)

(ref. ACM Transactions on Mathematical Software Vol 8, no. 3: 302-322)

Direct solvers in dmu4

Direct solvers based on FSPAK:

Prediction error calculated one by one element at the time

Prediction error calculated from a sparse inverse of the MME calculated speed optimised

Prediction error calculated from a sparse inverse of the MME calculated memory optimised

ref. Misztal, I. & M. Perez-Enciso 1993. J. Dairy Sci. 76, 1479-1483)

Which solver to use?

Direct solvers based on FSPAK are preferable because s.e. are also calculated

The iterative solvers based on ITPACK can handle larger problems then the direct solvers

If iterative solvers are used, Jacobi Conjugate Gradient is good chose

S.E. for selected equations

Iterative solver:
Standard errors for solutions are not calculated

Standard errors for selected solutions (max. 100) can be computed

Direct solver:
Standard errors for all solutions are calculated

Iterative and direct solvers:
Standard errors for functions of solutions can be estimated

DMU5

BLUE and BLUP using "iteration on data"

- Solving strategy preconditioned conjugate gradient (PCG)
- Data and pedigree file in core or on disk

Can handle single and multi trait models with:
- Random regressions
- Direct and maternal effects
- Sire-Dam/MGS models
- Reduced rank models

Current implementation uses a block diagonal preconditioner, where all but one fixed effect are treated as a block

DMU5

"ITERATION ON DATA"

MME is not build
Data and pedigree file
read/processed in each round
Data and pedigree file in core if possible otherwise on disk

RJMC

MCMC estimation of location and dispersion parameters for Gaussian, Binary and two component mixture traits

Can handle single and multi trait models with:

- Random regressions
- Reduced rank
- Heterogeneous residual co-variance structure
- Reaction norm models with unknown environmental gradient

RJMC

RJMC is based on iteration on data techniques

Sampling is performed block-diagonal

Can handle large models, but can be very time consuming

The progress of the program is monitored in a file named **status** in the directory where the analysis is running

DMUv6 - Future

Continuous development and implementation of new facilities

- QTL detection
- MA-BLUP
- Categorical traits in RJMC
- Model with direct and maternal/paternal effects in RJMC
- Generalized Linear Mixed Models
- R interface

Work in progress (1)

AI-REML approach for QTL detection (Sørensen et. al 2003)

- Multivariat analyses
- Multi QTL analyses
- Estimates variance components for each QTL based on user specified inverse IBD matrices
- Types of IBD matrices supported:
  - Gametic
  - Genotypic
  - Cluster

Work in progress (2)

Generalized Linear Mixed Models in DMUAI

Implementation of more predefined Link and variance functions and on facilities for user specified functions

Tools for model validation and tests

Work in progress (3)

BLUP – DMU4 and DMU5

MA-BLUP:

- Approach of Fernando & Grossman in DMU4
  Based on user specified inverse IBD matrices
- Approach of Jafarikia et. al in DMU4 (and DMU5)
  Based on user specified IBD

Generalized Linear mixed models (DMU4 and DMU5)

Speed and dimensional improvement (DMU5):

- Alternative structure of the preconditioner

Availability

The DMU-packages is distributed as executables for a variety of platforms and operation systems

Can be downloaded from**: http://dmu.agrsci.dk**

- It is free of charge for research purpose
- For commercial use (i.e. routine genetic evaluation) contact DIAS for terms of conditions