Variance component estimation (VCE) using MiX99

Anna-Kaisa Ylitalo and Martin Lidauer

MiX99 course on genomic prediction

COURSE DAY 2, March 10th, 2026

Introduction to REML

Introduction to REML

- Restricted maximum likelihood (REML) is a widely used method for variance component estimation (VCE) in linear mixed effect models
  - REML estimation maximizes only the part of the likelihood that is independent of the fixed effects
  - Produces less biased variance estimates than ordinary maximum likelihood (ML)

- Two most widely used algorithms for REML are expectation-maximization REML (EM-REML) and average information REML (AI-REML)
  - EM-REML uses the first derivatives of the REML log-likelihood (numerically stable but slow)
  - AI-REML uses both the first and second derivatives of the REML log-likelihood (fast but sensitive to starting values)

- EM-REML computes REML estimates by iterating two steps:
  1. E-step computes expectation of the complete-data log-likelihood given the current parameter estimates
  2. M-step maximizes the likelihood -> updated parameter estimates

Luke

EM-REML

Consider single-trait model

$$\begin{aligned}\mathbf{y} &= \mathbf{X\beta} + \mathbf{Zu} + \mathbf{e}\\ \mathrm{Var}(\mathbf{u}) &= \mathbf{G} = \sigma_u^2\mathbf{A}\\ \mathrm{Var}(\mathbf{e}) &= \mathbf{R} = \sigma_e^2\mathbf{I}\\ \mathrm{Var}(\mathbf{y}) &= \mathbf{ZGZ}' + \mathbf{R}\end{aligned}$$

Mixed model equations:

$$\begin{bmatrix}\mathbf{X}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{R}^{-1}\mathbf{Z}\\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{R}^{-1}\mathbf{Z} + \frac{1}{\sigma_u^2}\mathbf{A}^{-1}\end{bmatrix}\begin{bmatrix}\hat{\boldsymbol{\beta}}\\ \hat{\mathbf{u}}\end{bmatrix} = \begin{bmatrix}\mathbf{X}'\mathbf{R}^{-1}\mathbf{y}\\ \mathbf{Z}'\mathbf{R}^{-1}\mathbf{y}\end{bmatrix}$$

The EM-REML updates of the VC are:

$$\hat{\sigma}_u^2 = \frac{1}{q}\left[\hat{\mathbf{u}}'\mathbf{A}^{-1}\hat{\mathbf{u}} + \mathrm{tr}(\mathbf{A}^{-1}\mathbf{C}^{uu})\right]$$

$$\hat{\sigma}_e^2 = \frac{1}{n}\left[\hat{\mathbf{e}}'\hat{\mathbf{e}} + \mathrm{tr}(\mathbf{WCW}')\right]$$

where $\mathbf{C}$ is the inverse of the coefficient matrix, $\mathbf{C}^{uu}$ its submatrix corresponding to $\mathbf{u}$ and $\mathbf{W} = [\mathbf{X} \quad \mathbf{Z}]$

Monte Carlo EM-REML

- MiX99 uses Monte Carlo expectation maximization restricted maximum likelihood (MC EM-REML)
- WHY?
  - The analytical REML-based methods needs elements from the inverse coefficient matrix of the MME, i.e. for prediction error variances (PEV)
  - Inverting a dense and large coefficient matrix is computationally challenging in genomic models

- For large data, Monte Carlo algorithm can approximate PEV without inverting the coefficient matrix (García-Cortés et al. 1992)
  - Idea is to estimate the elements of the inverse coefficient matrix by generating samples from the same distribution as the original data and then solving the MME using the sampled data as observations
  - By generating the samples, the true random effect and residual values are known and the mean of PEVs over the samples can be calculated

Luke

Monte Carlo EM-REML

- MC EM-REML algorithm (Matilainen et al. 2012) makes REML feasible for large data sets and complex models for which the inversion of the coefficient matrix would be too memory and time consuming
  - Is faster than EM-REML and can handle larger models
- Simulated data (MC samples) is generated according to:

$$\tilde{\mathbf{y}}^{\mathrm{h}} = \mathbf{X}\tilde{\boldsymbol{\beta}}^{\mathrm{h}} + \mathbf{Z}\tilde{\mathbf{u}}^{\mathrm{h}} + \tilde{\mathbf{e}}^{\mathrm{h}} \qquad \text{OBS! } \tilde{\boldsymbol{\beta}}^{\mathbf{h}} = \mathbf{0}$$
$$\tilde{\mathbf{u}}^{\mathrm{h}} \sim \mathrm{N}(\mathbf{0}, \hat{\sigma}_{\mathrm{u}}^2 \mathbf{A})$$
$$\tilde{\mathbf{e}}^{\mathrm{h}} \sim \mathrm{N}(\mathbf{0}, \hat{\sigma}_{\mathrm{e}}^2 \mathbf{I})$$
$$\mathrm{h} = 1, \ldots, \mathrm{s}$$

- Obtain estimate $\hat{\mathbf{u}}^{h}$ by solving MME using simulated data $\tilde{\mathbf{y}}^{\mathrm{h}}$
- Calculate $\hat{\mathbf{y}}^{\mathrm{h}} = \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{h}} + \mathbf{Z}\hat{\mathbf{u}}^{\mathrm{h}}$
  - Calculate residuals $\hat{\mathbf{e}}^{\mathrm{h}} = \tilde{\mathbf{y}}^{\mathrm{h}} - \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{h}} - \mathbf{Z}\hat{\mathbf{u}}^{\mathrm{h}}$

Monte Carlo EM-REML

- Traces can be approximated by e.g. (García-Cortés method 1):

$$\mathrm{tr}(\mathbf{A}^{-1}\mathbf{C}^{uu}) \approx \frac{1}{s}\sum_{h=1}^{s}\left(q\hat{\sigma}_u^2 - \hat{\mathbf{u}}^{h\prime}\mathbf{A}^{-1}\hat{\mathbf{u}}^{h}\right)$$

$$\mathrm{tr}(\mathbf{WCW}') \approx \frac{1}{s}\sum_{h=1}^{s}\left(n\hat{\sigma}_e^2 - \hat{\mathbf{e}}^{h\prime}\hat{\mathbf{e}}^{h}\right)$$

- Update variance components by plugging in the trace approximations:

$$\hat{\sigma}_u^2 = \frac{1}{q}\left[\hat{\mathbf{u}}'\mathbf{A}^{-1}\hat{\mathbf{u}} + \mathrm{tr}(\mathbf{A}^{-1}\mathbf{C}^{uu})\right]$$

$$\hat{\sigma}_e^2 = \frac{1}{n}\left[\hat{\mathbf{e}}'\hat{\mathbf{e}} + \mathrm{tr}(\mathbf{WCW}')\right]$$

- An increase in MC sample size *s* will give more accurate estimates but costs more time!
  - Sample size 5-25 gives already good approximation, but 1-5 is also used

Luke

VCE in MiX99

Variance component estimation in MiX99

Variance component estimation in MiX99 requires execution of

1. mix99i with initial values for the (co)variance components
2. mix99s with specific instructions in the solver option file (.slv)

```
~> mix99i model.clm > mix99i.log
~> mix99s < reml.slv > mix99s.log
```

Luke

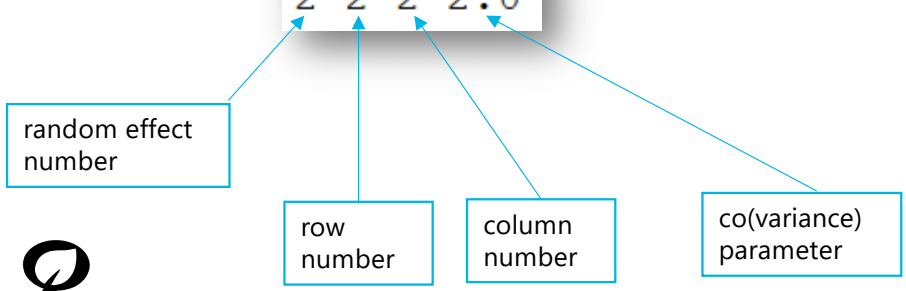
Starting values of (co)variance components

Options to specify the starting values for the (co)variance components in CLIM

- PARFILE parfile.var
  - parameters are read from a file in sparse matrix format
  - The traditional sparse matrix format has one line for each non-zero (co)variance.

parfile.var

```
1 1 1 1.0
1 2 2 1.0
2 1 1 2.0
2 2 2 2.0
```



- PARFILE CLIM
  - parameters are specified in CLIM

```
PARFILE CLIM
1 1 1 1.0
1 2 2 1.0
2 1 1 2.0
2 2 2 2.0
```

Starting values of (co)variance components

Options to specify the starting values for the (co)variance components in CLIM

- PARFILE MIXED parfile.mix
  - mixed matrix format
  - IDENTITY, DIAGONAL, LOWER, SPARSE

parfile.mix

random effect number

```
1 IDENTITY
2 DIAGONAL
  2.0
  2.0
```

or

parfile.mix

```
1 LOWER
  1.0
  0.0 1.0
2 SPARSE
  1 1 2.0
  2 2 2.0
```

- PARFILE IDENTITY
  - identity matrix as (co)variance matrix for all random effects
  - easy option, but can be very slow

Good starting values can save a lot of time!

Convergence of VCE in MiX99

- Monte Carlo noise in parameter estimates are accounted with a special convergence indicator
  - Predicted variance component estimates are calculated using linear regression on estimated variance components of the *x* latest REML rounds
- Convergence indicator at REML round k*:*

$$\mathrm{cc}_{\mathrm{E}}^{(\mathrm{k})} = \frac{(\hat{\mathbf{s}}^{(\mathrm{k})}(\mathrm{x}) - \hat{\mathbf{s}}^{(\mathrm{k}-1)}(\mathrm{x}))^{\mathrm{T}}(\hat{\mathbf{s}}^{(\mathrm{k})}(\mathrm{x}) - \hat{\mathbf{s}}^{(\mathrm{k}-1)}(\mathrm{x}))}{(\hat{\mathbf{s}}^{(\mathrm{k})}(\mathrm{x}))^{\mathrm{T}}(\hat{\mathbf{s}}^{(\mathrm{k})}(\mathrm{x}))},$$

where $\hat{\mathbf{s}}^{(\mathrm{k})}(\mathrm{x})$ is the vector of predicted VC estimates based on x previous estimates. ($\mathrm{x} = \frac{\mathrm{k}+1}{2}$ in MiX99)

- After $\mathrm{cc}_{\mathrm{E}}^{(\mathrm{k})}$ has reached a value smaller than the specified convergence criterion, the REML analysis will perform a sequence of 30 additional MC EM REML rounds
  - Reduces the Monte-Carlo error from the parameter estimates by using weighted average with decreasing weights for latest solutions

Luke

VCE in MiX99 using solver option file (.slv)

- For VCE in MiX99 one can define:
  - Maximum number of REML iterations
  - Number of Monte Carlo samples per iteration
  - Convergence value
- In solver option file this is option E on the VAROPT line followed by three additional lines:
  - STOPE max. number of REML rounds, number of MC samples, convergence value for REML
    - Default values: 1000, 5 and 1.0e-9
  - SEED Type of the seed used by the random number generator
    - D=default initialization of seeds
    - R=seeds initialized based on the system clock
    - G=user specified seeds
  - MIXPATH
    - Directory path for MiX99 preprocessor

Luke

Example of solver option file (.slv)

Options for PCG (CD ≤ 10^{-5}, max 5000 iterations)

Options for VC estimation (criterion ≤ 10^{-9}, 2 REML samples, max 1000 REML iterations)

reml.slv

```
# RAM: RAM demand: H=high, M=medium, L=low
   H   nt 10
# STOP: Max.iter, Conv.value BLUP, Criterion (A/R/D), Enforce
   5000              1.0e-5            d               f
# RESID: Calculate residuals? (Y/N)
   N
# VALID: Model validation. N=no, P=prediction, S=sum of effects, Y=YD, D=DYD, I=IDD, G=generate
   N
# VAROPT: Variance options. (N)o HV, (S)tart HV, (C)ontinue HV, (F)inale, (E)stimation of VC by EM
   E
   # STOPE Max.REML rounds, Samples/step, Conv.value VCE, [Conv.value BLUP, sampled data]
          1000                 2             1.0e-9             1.0e-4
   # SEED Type of the seed used by the random number generator
           R
   # MIXPATH directory path of mix99i preprocessor
     /home/L1677/bin
# SOLTYP: Solution files? (N)o, (Y)es, (A)itken, (H)alf-Chebychev
   Y
```

~> mix99s < reml.slv > mix99s.log

Output files and convergence

Output files related to VCE

- **parfile** contains the latest solutions of variance component estimates
  - The structure of the file is the same as in the file defined by PARFILE
  - Columns: random effect, row, column, value
- **REMLlog** contains the VC estimates at each REML round
  - Columns: REML iteration number, convergence criterion value, variance component estimates
  - The first three lines in the REMLlog file describe the order of the parameter columns
  - The fourth line contains the initial parameter values used
  - Can be used to plot traces of each component (for checking convergence)
- **.log** files. It is a good practice to save and check the log files (both from the preprocessor and the solver)!

+ **vceSE** and **vceI** including standard errors and information matrix (introduced later)

Luke

Checking the output files

- ✓ OK_mix99s file appeared
- ✓ mix99s.log file
  - ✓ REML has converged:
    - reached the convergence value and not the maximum number of iterations

mix99s.log (end)

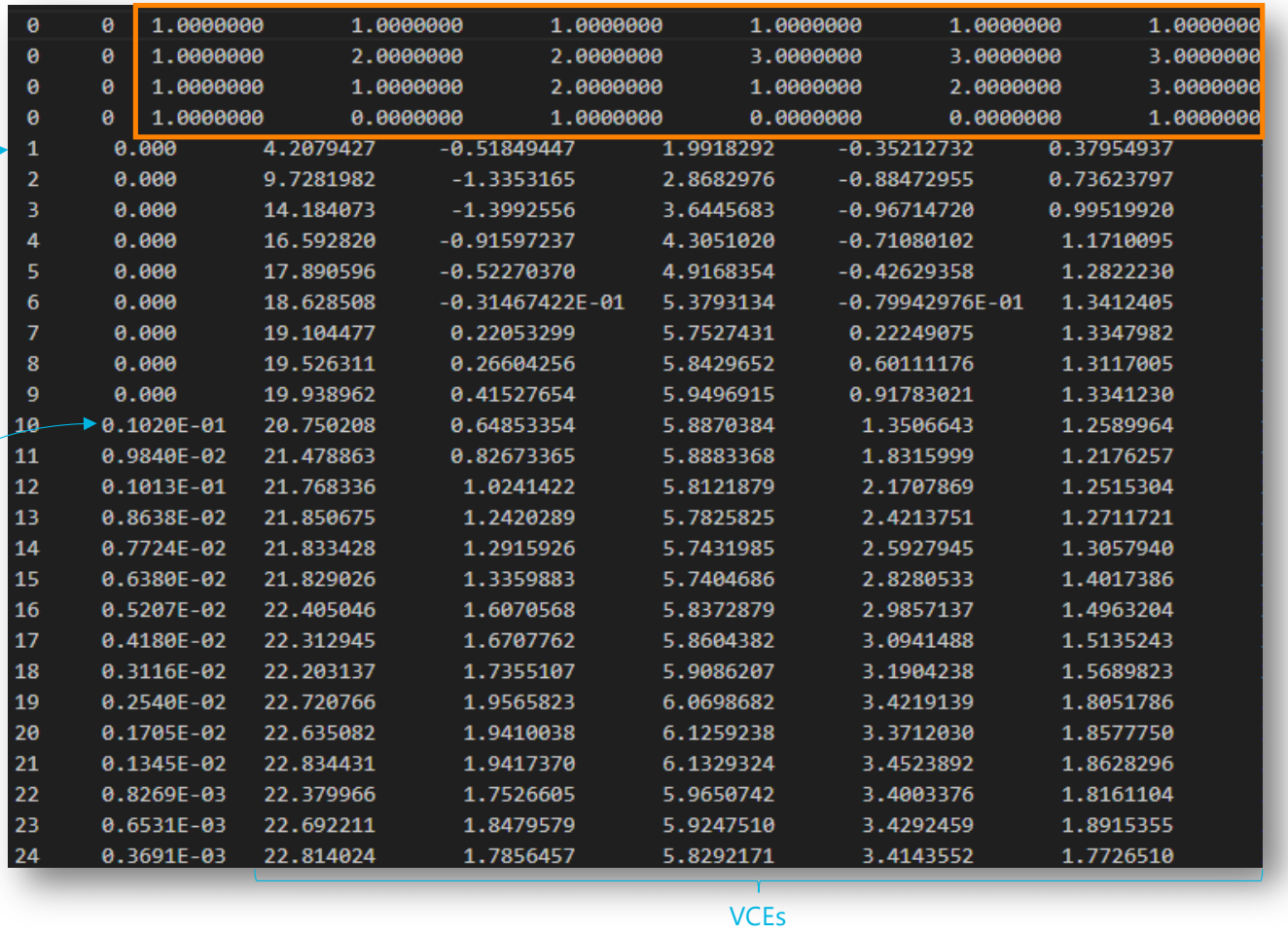
```
MiX99_SOLVE:      --- D O N E ---                Time: 15:21:47.8  24.02.2026

      Copyright(C) 2024 Natural Resources Institute Finland (Luke)
```

mix99s.log

```
     87    0.2735E-06    0.1405E-06    0.2307E-06    0.9500E-05   -0.9148E-04
     88    0.2963E-06    0.1529E-06    0.2387E-06    0.6956E-05    0.7216E-04
   Stopping criterion CD < 0.1E-4 achieved in 88 iterations.
   NOTE: CD criterion must be met by two consecutive iterations!

   Solutions have converged according to CD criterion of the last iteration.

 MiX99_SOLVE: End of PCG Iteration              Time: 15:21:44.6  24.02.2026

    250    0.1345E-05    0.3531E-05    0.1616E-05    0.8755E-05   -0.1140E-03
    258    0.1627E-05    0.3132E-05    0.1864E-05    0.8676E-05   -0.8867E-04

REML ROUND  1000   CONV Cd    0.7891E-08

 MiX99_SOLVE: End of MC EM REML iteration       Time: 15:21:47.3  24.02.2026

REML convergence criterion 0.1E-8 was _not_ achieved in 1000 rounds!
```

```
REML convergence criterion (0.1E-8) was achieved in 5533 rounds
and additional 30 rounds were performed to reduce variation in variance
component estimates.
```

Convergence statistics in REMLlog



0	0	1.0000000	1.0000000	1.0000000	1.0000000	1.0000000	1.0000000
0	0	1.0000000	2.0000000	2.0000000	3.0000000	3.0000000	3.0000000
0	0	1.0000000	1.0000000	2.0000000	1.0000000	2.0000000	3.0000000
0	0	1.0000000	0.0000000	1.0000000	0.0000000	0.0000000	1.0000000
1	0.000	4.2079427	-0.51849447	1.9918292	-0.35212732	0.37954937	
2	0.000	9.7281982	-1.3353165	2.8682976	-0.88472955	0.73623797	
3	0.000	14.184073	-1.3992556	3.6445683	-0.96714720	0.99519920	
4	0.000	16.592820	-0.91597237	4.3051020	-0.71080102	1.1710095	
5	0.000	17.890596	-0.52270370	4.9168354	-0.42629358	1.2822230	
6	0.000	18.628508	-0.31467422E-01	5.3793134	-0.79942976E-01	1.3412405	
7	0.000	19.104477	0.22053299	5.7527431	0.22249075	1.3347982	
8	0.000	19.526311	0.26604256	5.8429652	0.60111176	1.3117005	
9	0.000	19.938962	0.41527654	5.9496915	0.91783021	1.3341230	
10	0.1020E-01	20.750208	0.64853354	5.8870384	1.3506643	1.2589964	
11	0.9840E-02	21.478863	0.82673365	5.8883368	1.8315999	1.2176257	
12	0.1013E-01	21.768336	1.0241422	5.8121879	2.1707869	1.2515304	
13	0.8638E-02	21.850675	1.2420289	5.7825825	2.4213751	1.2711721	
14	0.7724E-02	21.833428	1.2915926	5.7431985	2.5927945	1.3057940	
15	0.6380E-02	21.829026	1.3359883	5.7404686	2.8280533	1.4017386	
16	0.5207E-02	22.405046	1.6070568	5.8372879	2.9857137	1.4963204	
17	0.4180E-02	22.312945	1.6707762	5.8604382	3.0941488	1.5135243	
18	0.3116E-02	22.203137	1.7355107	5.9086207	3.1904238	1.5689823	
19	0.2540E-02	22.720766	1.9565823	6.0698682	3.4219139	1.8051786	
20	0.1705E-02	22.635082	1.9410038	6.1259238	3.3712030	1.8577750	
21	0.1345E-02	22.834431	1.9417370	6.1329324	3.4523892	1.8628296	
22	0.8269E-03	22.379966	1.7526605	5.9650742	3.4003376	1.8161104	
23	0.6531E-03	22.692211	1.8479579	5.9247510	3.4292459	1.8915355	
24	0.3691E-03	22.814024	1.7856457	5.8292171	3.4143552	1.7726510	

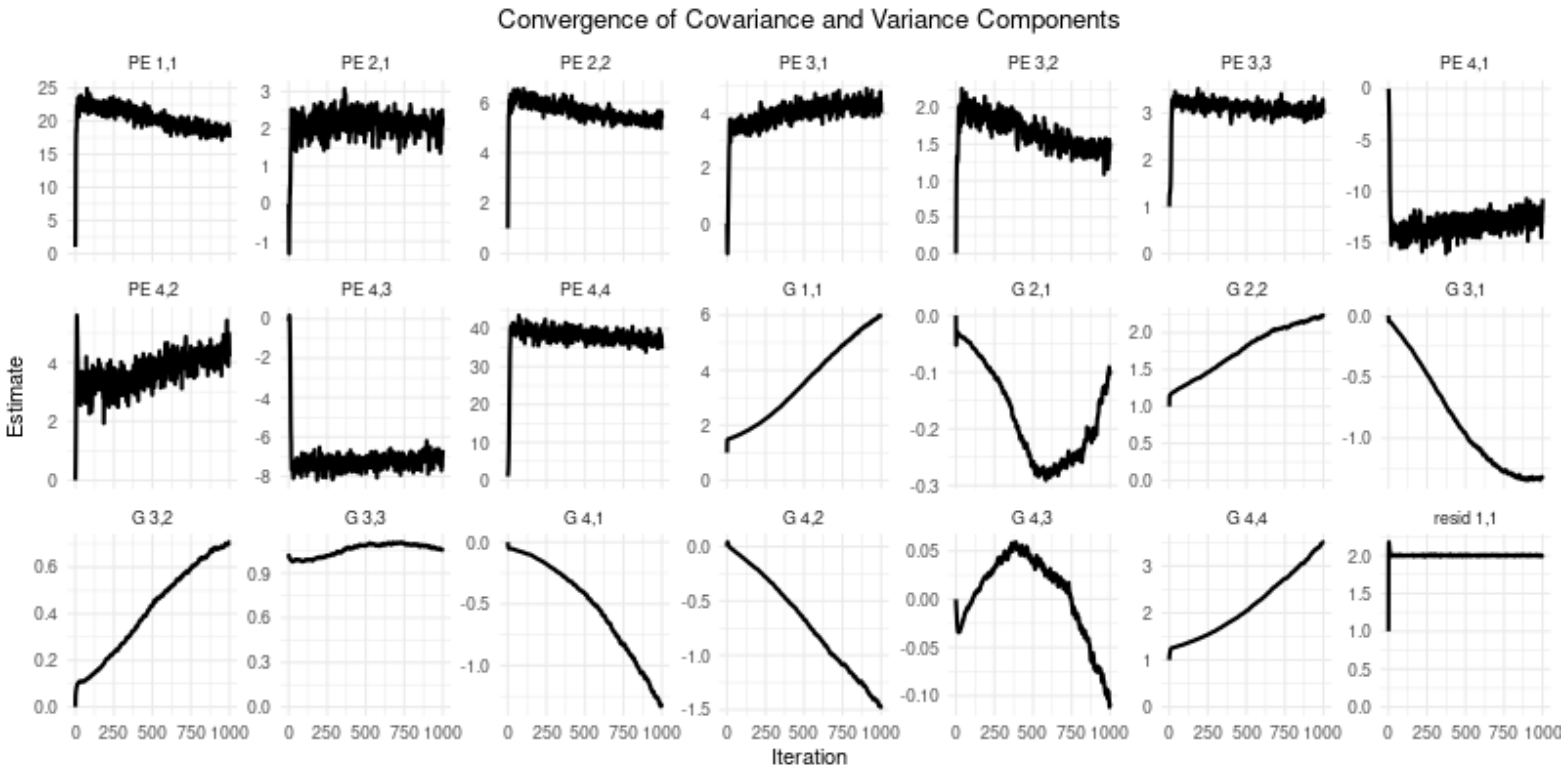
random effect,
row,
column,
initial value

REML iteration

convergence
criterion

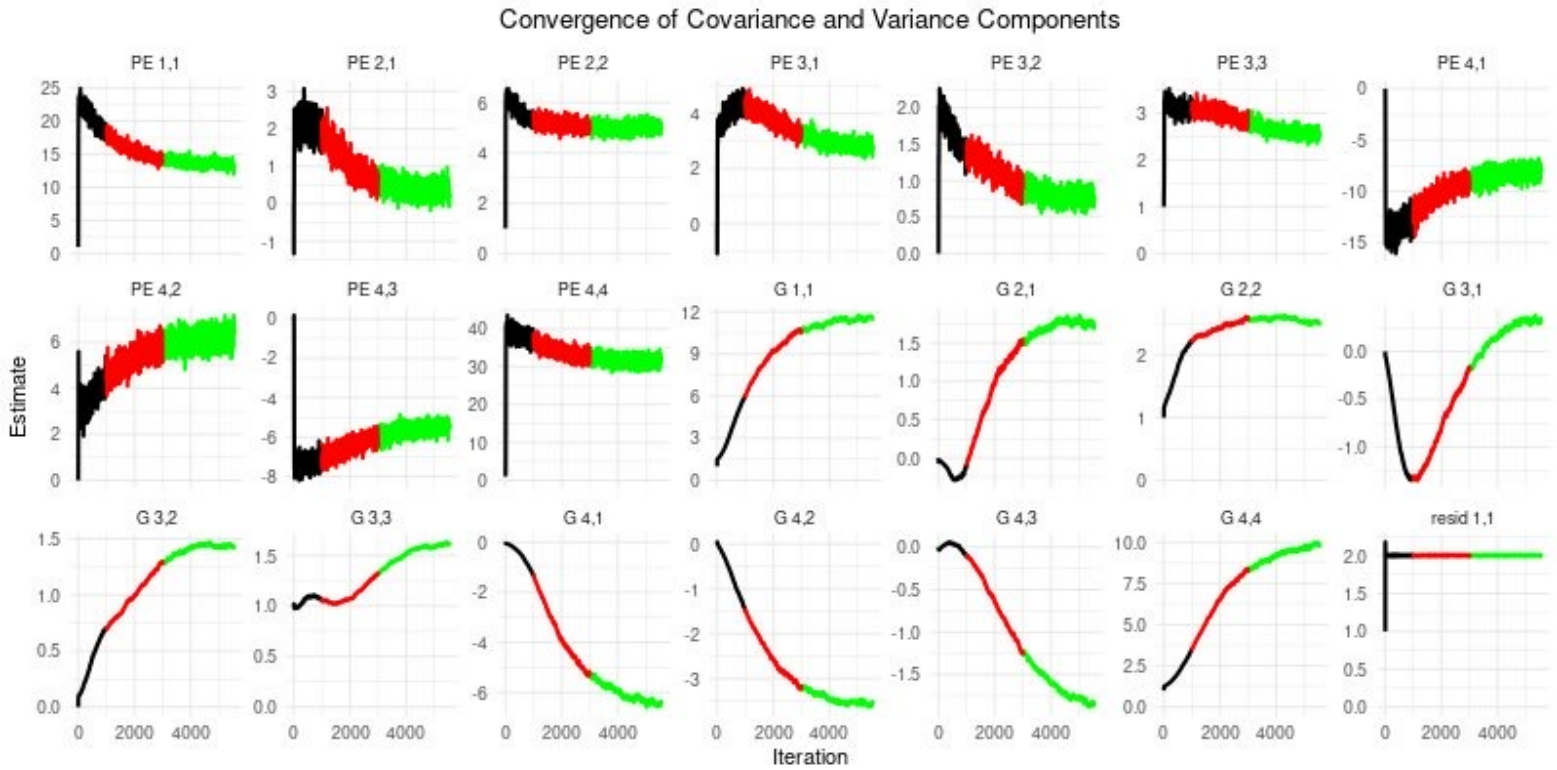
VCEs

Convergence statistics in REMLlog



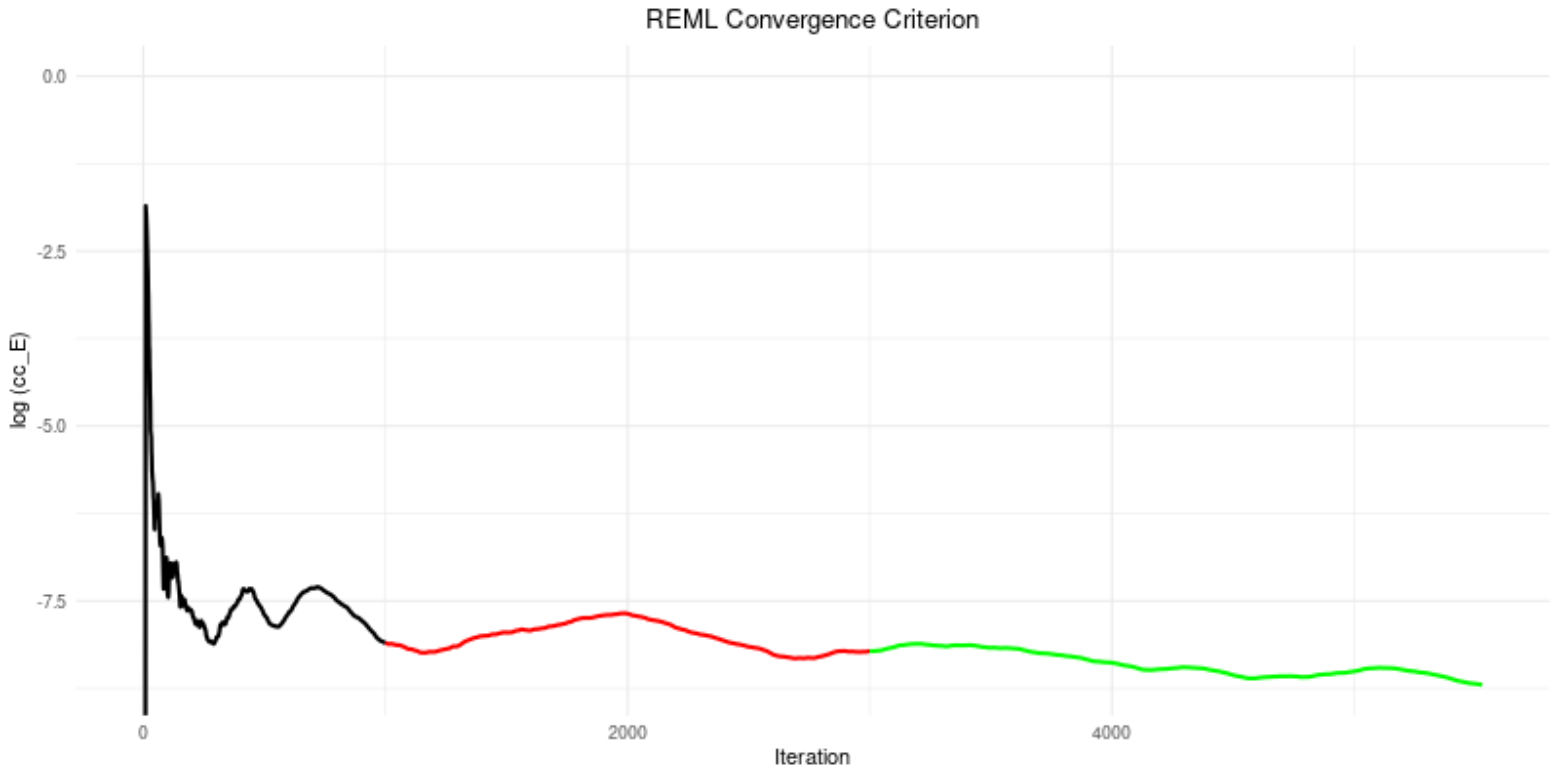
✕ REML convergence

Convergence statistics in REMLlog



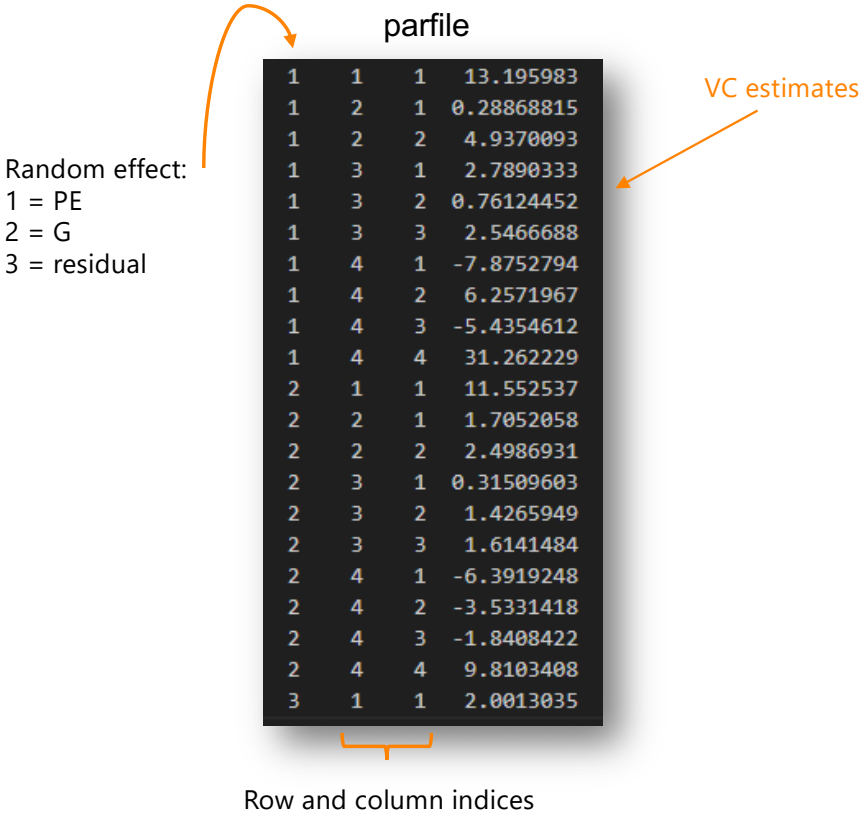

```
REML convergence criterion (0.1E-8) was achieved in 5533 rounds
and additional 30 rounds were performed to reduce variation in variance
component estimates.
```

Convergence statistics in REMLlog




```
REML convergence criterion (0.1E-8) was achieved in 5533 rounds
and additional 30 rounds were performed to reduce variation in variance
component estimates.
```

Variance component estimates



To sum up

- VCE in MiX99 is implemented with iterative MC EM-REML algorithm
- Choices have to be made before the VCE:
  - Initial values for the (co)variance parameters
  - Value for the convergence criterion (REML and BLUP)
  - Number of Monte Carlo samples
- Check the outputs and plot convergence plots
  - Make sure that REML algorithm has converged

Variance component estimation is often balancing between computing resources and estimation accuracy

Luke