



UNIVERSITY OF
GEORGIA

College of Agricultural &
Environmental Sciences

Introduction to **BLUPF90** software suite

BLUPF90 TEAM – 05/2022

BLUPF90 software suite

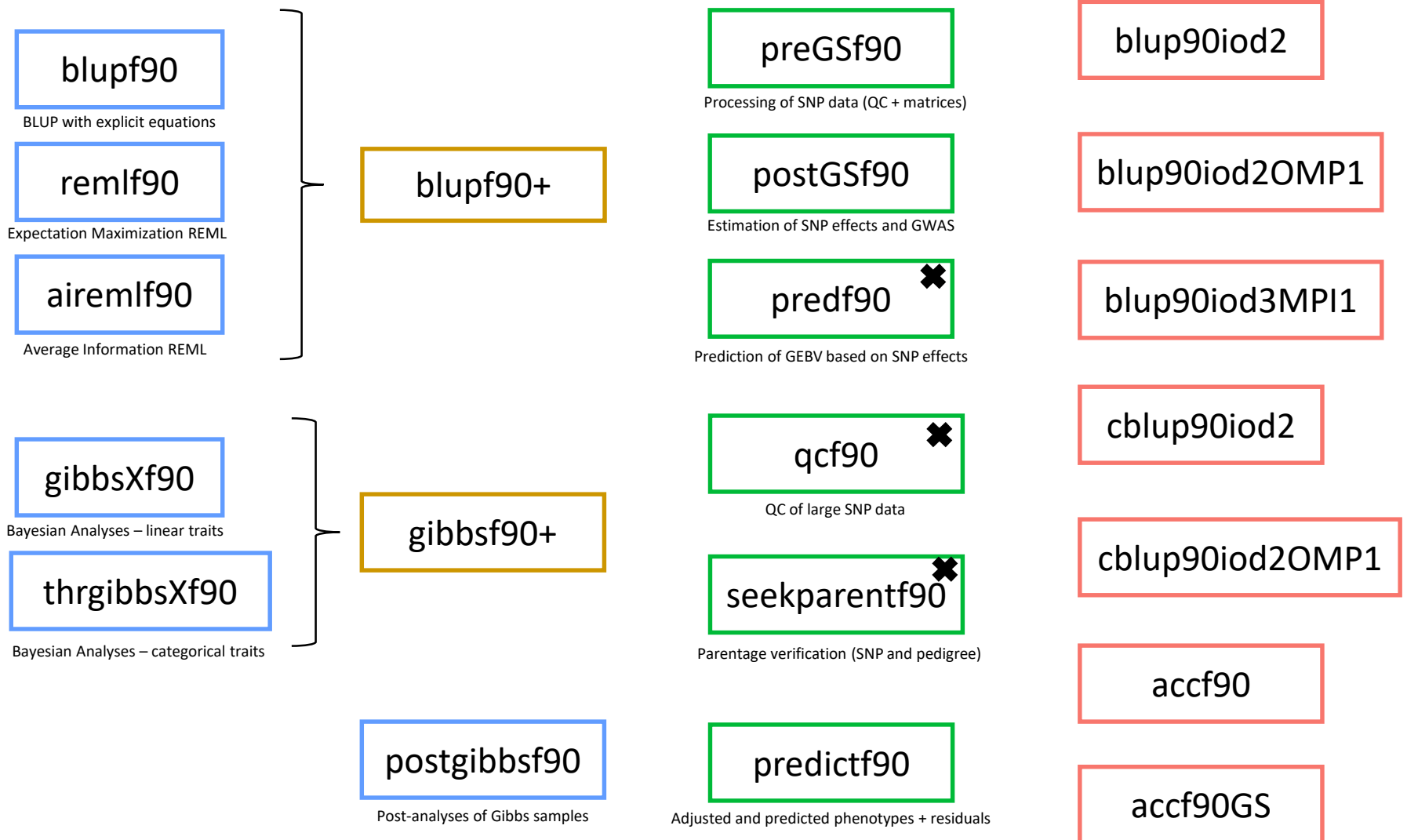
- Collection of software for computations
 - Focus on applications in Breeding and Genetics
- Fortran \geq 90
- Since 1998/1999 by Ignacy Misztal
- Several developers + collaborators
- Simple, efficient, and comprehensive
- No GUI (graphical user interface)!!!

- First idea: to solve the MME

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

- First software: blupf90
- Second idea: variance components estimation
- Software: remlf90, gibbsf90, airemlf90

BLUPF90 software suite



blupf90+

blupf90: MME solver

airemlf90: variance components using Average Information REML

remlf90: variance components using Expectation Maximization REML

Mixed Model Equations Solver
Variance Components Estimation

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

blupf90+



MME Solver



VC Estimation

Additional OPTION in the parameter file

blupf90+

- Supports virtually any model used in AB&G:
 - animal model
 - models with maternal effect
 - MPE
 - PE
 - Random Regression
 - Social interaction
 - Multiple traits
 - up to 70 if no correlated effects
 - up to $\lfloor 70/\text{number of correlated effects} \rfloor$

blupf90+

- How to

```
[dani@dodo5 examples]$ blupf90+  
name of parameter file?
```

```
[dani@dodo5 examples]$ blupf90+ --help
```

```
*****  
*   BLUPF90+   *  
*****
```

Computation of variance components, solutions, and s.e.
Default behavior avoids variance components estimation
For help about genomics, use blupf90+ --help-genomic

- * OPTION SNP_file snp
Specify the SNP file name to use genotype data.
- * OPTION method VCE (default BLUP with blupf90 options)
Run airemlf90 for variance component estimation (default running blupf90)
- * OPTION conv_crit 1d-12
Convergence criterion (default 1d-10)

blupf90+

- Input files
 - Free format (minimum one space to separate columns)
 - TAB is not a valid separator
 - Only numbers: integer or real
 - Decimal separators “.” not “,”
 - One “.” is not a missing value as in SAS
 - All effects need to be renumbered from 1 (consecutively)

blupf90+

- Computes generalized solutions by several methods:
 - Preconditioner Conjugate Gradient (PCG)
 - Default Iterative method (fast)
 - Successive over-relaxation (SOR)
 - an iterative method based on Gauss-Seidel
 - Direct solution using sparse Cholesky factorization
 - FSPAK or YAMS (greater memory requirements)
 - Can provide PEV
- The solution values change among methods
 - estimable functions should be the same

Parameter file for blupf90+

```
DATAFILE
  ../renf90.dat
NUMBER_OF_TRAITS      2
NUMBER_OF_EFFECTS     5
OBSERVATION(S)
  1      2
WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT[EFFECT NESTED]
  3  4      40593 cross
  5  5      2 cross
  6  0      4 cross
  7  0      8 cross
  8  8     918111 cross
RANDOM_RESIDUAL VALUES
  2.5300      1.3425
  1.3425      29.714
RANDOM_GROUP
  5
RANDOM_TYPE
  add_an_upginb
FILE
  ../renadd05.ped
(CO)VARIANCES
  0.7600      2.2391
  2.2391      30.609
```

} Unlimited number of traits and effects

Parameter file for blupf90+

```
DATAFILE
  ../renf90.dat
NUMBER_OF_TRAITS
  2
NUMBER_OF_EFFECTS
  5
OBSERVATION(S)
  1 2
WEIGHT(S)
  1 2
EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT[EFFECT NESTED]
  3 4 40593 cross
  5 5 2 cross
  6 0 4 cross
  7 0 8 cross
  8 8 918111 cross
RANDOM_RESIDUAL VALUES
  2.5300 1.3425
  1.3425 29.714
RANDOM_GROUP
  5
RANDOM_TYPE
  add_an_upginb
FILE
  ../renadd05.ped
(CO) VARIANCES
  0.7600 2.2391
  2.2391 30.609
```

As many columns as the number of traits

Number of levels

Type of effect

- As many rows as the NUMBER_OF_EFFECTS
- Model definition for each trait
- Different models per trait are supported
- If an effect is missing for one trait use 0

Parameter file for blupf90+

```
DATAFILE
  ../renf90.dat
NUMBER_OF_TRAITS
  2
NUMBER_OF_EFFECTS
  5
OBSERVATION(S)
  1 2
WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT[EFFECT NESTED]
  3 4 40593 cross
  5 5 2 cross
  6 0 4 cross
  7 0 8 cross
  8 8 918111 cross
RANDOM_RESIDUAL VALUES
  2.5300 1.3425
  1.3425 29.714
RANDOM_GROUP
  5
RANDOM_TYPE
  add_an_upginb
FILE
  ../renadd05.ped
(CO) VARIANCES
  0.7600 2.2391
  2.2391 30.609
```


} Should be a square matrix with dimension equal to the number of traits

- Use 0.0 to indicate uncorrelated effects between traits
- e.g., RANDOM_RESIDUAL VALUES for a 2-trait model
2.53 0.0
0.0 29.714

Parameter file for blupf90+

```
DATAFILE
  ../renf90.dat
NUMBER_OF_TRAITS
  2
NUMBER_OF_EFFECTS
  5
OBSERVATION(S)
  1    2
WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT[EFFECT NESTED]
  3  4      40593 cross
  5  5      2 cross
  6  0      4 cross
  7  0      8 cross
  8  8     918111 cross
RANDOM_RESIDUAL VALUES
  2.5300      1.3425
  1.3425      29.714
RANDOM_GROUP
  5
RANDOM_TYPE
  add_an_upginb
FILE
  ../renadd05.ped
(CO) VARIANCES
  0.7600      2.2391
  2.2391      30.609
```



Definition of random effects

RANDOM_GROUP
RANDOM_TYPE
FILE
(CO) VARIANCES

Definition of random effects

- RANDOM_GROUP
 - Number of the effect(s) from the list of effects
 - Correlated effects should be consecutive e.g. Maternal effects, Random Regression
- RANDOM_TYPE
 - diagonal, add_animal, add_sire, add_an_upg, add_an_upginb, add_an_self, user_file, user_file_i, or par_domin
- FILE
 - Pedigree file, parental dominance, or user file
- (CO)VARIANCES
 - Square matrix with dimension equal to the number_of_traits*number_of_correlated_effects

(CO)VARIANCES

- Assuming a 3 trait (T1-T3) and 2 correlated effects (E1-E2)

		E1			E2		
		T1	T2	T3	T1	T2	T3
E1	T1						
	T2						
	T3						
E2	T1						
	T2						
	T3						

RANDOM_TYPE

- *diagonal*
 - For permanent environment effects
 - Assumes no correlation between levels of the effect
- *add_sire*
 - To create a relationship matrix using sire and maternal grandsire
 - Pedigree file:
 - animal number, sire number, maternal grandsire number
- *add_animal*
 - To create a relationship matrix using sire and dam information
 - Pedigree file:
 - animal number, sire number, dam number

RANDOM_TYPE

- *add_an_upg*
 - As before but using rules for unknown parent group
 - Pedigree file:
 - animal number, sire number, dam number, parent code
 - missing sire/dam can be replaced by upg number
 - usually greater than the maximum number of animals
 - Parent code = 3 – # of known parents
 - 1 both parents known
 - 2 one parent known
 - 3 both parents unknown
- *add_an_upginb*
 - As before but using rules for unknown parent group and inbreeding
 - Pedigree file:
 - animal number, sire number, dam number, inb/upg code
 - missing sire/dam can be replaced by upg number
 - usually greater than the maximum number of animals
 - $\text{inb/upg code} = 4000 / [(1+ms)(1-Fs) + (1+md)(1-Fd)]$
 - ms (md) is 0 if sire (dam) is known and 1 otherwise
 - Fs(Fd) inbreeding coefficient of the sire (dam)

RANDOM_TYPE

- *Add_an_self*
 - *To create a relationship matrix when there is selfing*
 - Pedigree file:
 - individual number, parent 1 number, parent 2, number of selfing generations
- *user_file*
 - An inverted matrix is read from file
 - Matrix is stored only upper- or lower-triangular
 - Matrix file:
 - row, col, value
- *user_file_i*
 - As before but the matrix will be inverted by the program
- *par_domin*
 - A parental dominance file created by program RENDOM

OPTIONS for blupf90+

- Program behavior can be modified by adding extra options
 - end of the parameter file
- `OPTION option_name x1 x2 ...`
- `option_name`: each program has its own options
- The number of optional parameters (`x1, x2...`) to control the behavior depends on the option

Options for blupf90+

Options

```
OPTION conv_crit 1e-12
```

Set convergence criteria (default 1e-12).

```
OPTION maxrounds 10000
```

Set maximum number of rounds (default 5000).

```
OPTION solv_method FSPAK
```

Selection solutions by FSPAK, SOR or PCG (default PCG).

```
OPTION r_factor 1.6
```

Set relaxation factor for SOR (default 1.4).

```
OPTION sol se
```

Store solutions and standard errors.

```
OPTION store_pev_pec 6
```

Store triangular matrices of standard errors and its covariances for correlated random effects such as direct-maternal effects and random-regression effects in "pev_pec_bf90".

Options for blupf90+

Missing data
Not pedigree!



```
OPTION missing -999
```

Specify missing observations (default 0) in integer.

```
OPTION residual
```

y-hat and residual will be included in "yhat_residual".

```
OPTION blksize 3
```

Set block size for preconditioner (default 1).

```
OPTION use_yams
```

Run the program with YAMS (modified FSPAK).

```
OPTION SNP_file snp
```


Specify the SNP file name to use genotype data.

Options for blupf90+

Storing accuracy based on PEV

```
OPTION store_accuracy eff
```

Stores reliabilities based on PEV, where *eff* is the number of the animal effect.

By default, it uses inbreeding (F) in the denominator of the reliability formula: $\text{reliability} = 1 - \text{PEV} / (\sigma_u^2(1 + F))$  [Aguilar et al. \(2020\)](#).

It uses inbreeding based on **A** or **H** from the direct inversion of **A**⁻¹ or **H**⁻¹, whichever is being used.

```
OPTION type 1.0
```

Select 1.0 for dairy cattle (Reliability) or 0.5 for beef cattle (BIF accuracy) (default 1.0).

```
OPTION correct_accuracy_by_inbreeding filename
```

filename is the name of the inbreeding file if other than renf90.inb

```
OPTION correct_accuracy_by_inbreeding_direct 0
```

This option turns off the inbreeding correction in the reliability formula.

Options for blupf90+

<http://nce.ads.uga.edu/wiki/doku.php?id=readme.blupf90plus>

Example of parameter file for blupf90+

Single trait “USDA-type” animal model

$$y_{ijkl} = hys_i + p_k + hs_{ij} + a_k + e_{ijkl}$$

where

y_{ijkl} - production yield

hys_i - fixed herd year season

p_k - random permanent environment

hs_{ij} - random herd x sire interaction

a_k - random animal

and

$$\text{var}(hs_{ij}) = .05, \text{var}(p_k) = .1, \text{var}(a_k) = .5, \text{var}(e_{ijkl}) = 1$$

BLUPF90 parameter file created by RENUMF90

DATAFILE

renf90.dat

NUMBER_OF_TRAITS

1

NUMBER_OF_EFFECTS

4

OBSERVATION(S)

1

WEIGHT(S)

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS

2 3 cross

3 6 cross

4 4 cross

5 14 cross

RANDOM_RESIDUAL VALUES

1.0000

RANDOM_GROUP

2

RANDOM_TYPE

diagonal

FILE

(CO)VARIANCES

0.10000

RANDOM_GROUP

3

RANDOM_TYPE

diagonal

FILE

(CO)VARIANCES

0.50000E-01

RANDOM_GROUP

4

RANDOM_TYPE

add_an_upg

FILE

renadd04.ped

(CO)VARIANCES

0.50000

Model

$$y_{ijkl} = hys_i + p_k + hs_{ij} + a_k + e_{ijkl}$$

where

y_{ijkl} - production yield

hys_i - fixed herd year season

p_k - random permanent environment

hs_{ij} - random herd x sire interaction

a_k - random animal

and

$$\text{var}(hs_{ij}) = .05, \text{var}(p_k) = .1, \text{var}(a_k) = .5, \text{var}(e_{ijkl}) = 1$$

renf90.dat

Format: phen/hys/pe/hs/ani

1 1 1 1 3
1 1 2 1 6
2 2 3 2 2
3 2 4 3 5
4 3 5 4 1
3 3 6 3 4

renadd04.ped

Format: ani/sire/dam/code/.....

1 12 11 2 0 1 1 1 0 5
2 6 8 1 0 2 1 0 0 3
7 1 11 1 0 2 0 0 3 8
9 10 14 2 0 1 0 0 2 10
3 12 7 2 0 1 1 2 0 1
4 3 9 1 0 2 1 0 0 6
11 13 14 3 0 0 0 0 2 11
5 10 9 1 0 2 1 0 0 4
8 13 7 2 0 1 0 0 1 9
6 3 7 1 0 2 1 1 0 2
10 13 14 3 0 0 0 2 0 7

Output from blupf90+

name of parameter file?

renf90.par

BLUPF90 ver. 1.66

Parameter file: renf90.par

Data file: renf90.dat

Number of Traits 1

Number of Effects 4

Position of Observations 1

Position of Weight (1) 0

Value of Missing Trait/Observation 0

Parameter file

data file

EFFECTS

#	type	position (2)	levels [positions for nested]
1	cross-classified	2	3
2	cross-classified	3	6
3	cross-classified	4	4
4	cross-classified	5	14

Number of levels for
each effect

read 6 records in 6.1703999E-02 s,
read 11 additive pedigrees

54 nonzeros

Records read from data file

finished peds in 6.1760999E-02 s,

90 nonzeros

Records read from pedigree file

round = 1 convergence = 0.1435
round = 2 convergence = 0.3000E-01
round = 3 convergence = 0.1714E-02
round = 4 convergence = 0.2914E-03
round = 5 convergence = 0.1207E-03
round = 6 convergence = 0.1278E-03
round = 7 convergence = 0.1085E-03
round = 8 convergence = 0.1405E-03
round = 9 convergence = 0.1904E-03
round = 10 convergence = 0.1119E-03
round = 11 convergence = 0.1562E-04
round = 12 convergence = 0.6141E-05
round = 13 convergence = 0.4609E-05
round = 14 convergence = 0.1750E-04
round = 15 convergence = 0.8708E-04
round = 16 convergence = 0.2800E-03
round = 17 convergence = 0.1286E-04
round = 18 convergence = 0.2309E-06
round = 19 convergence = 0.2566E-08
round = 20 convergence = 0.1131E-09
round = 21 convergence = 0.2413E-12

21 iterations, convergence criterion= 0.2413E-12
solutions stored in file: "solutions"

$$\mathbf{Cx} = \mathbf{b}$$

$$\varepsilon = \frac{\|\mathbf{b} - \mathbf{Cx}\|^2}{\|\mathbf{b}\|^2}$$

Solutions file

File “solutions”

$$y_{ijkl} = hys_i + p_k + hs_{ij} + a_k + e_{ijkl}$$

where

y_{ijkl} - production yield

hys_i - fixed herd year season

p_k - random permanent environment

hs_{ij} - random herd x sire interaction

a_k - random animal

Parameter File

```
DATAFILE
renf90.dat
NUMBER_OF_TRAITS
1
NUMBER_OF_EFFECTS
4
OBSERVATION(S)
1
WEIGHT(S)
```

EFFECTS: POSITIONS IN DATAFILE

2 3 cross

3 6 cross

4 4 cross

5 14 cross

trait/effect level solution

1	1	1	0.49585171
1	1	2	2.52240030
1	1	3	2.91017217
1	2	1	-0.00965409
1	2	2	0.00965406
1	2	3	-0.01930794
1	2	4	0.01930787
1	2	5	0.03861622
1	2	6	-0.03861599
1	3	1	-0.00000001
1	3	2	-0.00965384
1	3	3	-0.00965406
1	3	4	0.01930810
1	4	1	0.64574095
1	4	2	-0.30035705
1	4	3	0.61034316
1	4	4	0.52426082
1	4	5	0.27486415
1	4	6	0.39795337
1	4	7	0.18556405
1	4	8	-0.90212661
1	4	9	0.63126003
1	4	10	-0.17807241
1	4	11	-0.17807202
1	4	12	1.25233833
1	4	13	-1.91741245
1	4	14	1.51299821

Common problem in blupf90+

- Wrong data file and pedigree name
 - Program may not stop if file name does not exist
 - Check outputs for data file name and number of records and pedigree read

```
round = 4995  convergence =      NaN
round = 4996  convergence =      NaN
round = 4997  convergence =      NaN
round = 4998  convergence =      NaN
round = 4999  convergence =      NaN
round = 5000  convergence =      NaN
5001 iterations,  convergence criterion=      NaN
solutions stored in file: "solutions"
```

blupf90+



MME Solver



VC Estimation

Additional OPTION in the parameter file

blupf90+



VC Estimation

REML

- REML = restricted/residual maximum likelihood
 - Patterson and Thompson (1971)
- Most used method for VCE in AB&G
- blupf90+ has 2 REML algorithms
 - EM-REML: expectation-maximization (EM) algorithm
 - AI-REML: average information (AI) algorithm

EM-REML

- This method requires iterations:

$$y = \mathbf{X}\beta + \mathbf{Z}u + e \quad \begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} \\ \mathbf{Z}'\mathbf{X} & \mathbf{Z}'\mathbf{Z} + \mathbf{A}^{-1} \frac{\sigma_e^2}{\sigma_a^2} \end{bmatrix} \begin{bmatrix} \hat{\beta} \\ \hat{u} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{y} \\ \mathbf{Z}'\mathbf{y} \end{bmatrix}$$

- 1) set initial variance components
- 2) compute $\hat{\beta}$ and \hat{u} via mixed model equations
- 3) update variance components with the following equations

$$\hat{\sigma}_a^2 = \frac{\hat{u}'\mathbf{A}^{-1}\hat{u} + \text{tr}(\mathbf{A}^{-1}\mathbf{C}^{uu})}{N_a}$$

Inverse of LHS for animal effect

$$\hat{\sigma}_e^2 = \frac{\mathbf{y}'(\mathbf{y} - \mathbf{X}\hat{\beta} - \mathbf{Z}\hat{u})}{N - \text{rank}(\mathbf{X})}$$

animals (rank of A)

- 4) go to 1 or stop if the parameters do not change anymore

EM-REML

- Simpler equations
 - More complicated equations in multiple-trait models
- Easier to understand
- Very slow convergence (looks stable but may not converge)
- Computationally demanding especially for \mathbf{C}^{uu}

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

AI-REML

Vector of variance components

$$\theta_{n+1} = \theta_n - \mathbf{H}^{-1}(\theta_n) \mathbf{d}(\theta_n)$$

Hessian Matrix

Gradient (score vector)

Average-information algorithm uses this matrix as Hessian,

$$\mathbf{H}(\theta) = \mathcal{I}_A(\theta) = \begin{bmatrix} -\frac{1}{2} \mathbf{y}' \mathbf{P} \mathbf{Z} \mathbf{A} \mathbf{Z}' \mathbf{P} \mathbf{Z} \mathbf{A} \mathbf{Z}' \mathbf{P} \mathbf{y} & -\frac{1}{2} \mathbf{y}' \mathbf{P} \mathbf{Z} \mathbf{A} \mathbf{Z}' \mathbf{P} \mathbf{P} \mathbf{y} \\ -\frac{1}{2} \mathbf{y}' \mathbf{P} \mathbf{P} \mathbf{Z} \mathbf{A} \mathbf{Z}' \mathbf{P} \mathbf{y} & -\frac{1}{2} \mathbf{y}' \mathbf{P} \mathbf{P} \mathbf{P} \mathbf{y} \end{bmatrix}$$

P = Projection
or hat matrix

Gradient

$$-2\mathbf{d}(\theta) = \begin{bmatrix} \text{tr}(\mathbf{P} \mathbf{Z} \mathbf{A} \mathbf{Z}') - \mathbf{y}' \mathbf{P} \mathbf{Z} \mathbf{A} \mathbf{Z}' \mathbf{P} \mathbf{y} \\ \text{tr}(\mathbf{P}) - \mathbf{y}' \mathbf{P} \mathbf{P} \mathbf{y} \end{bmatrix} = \begin{bmatrix} \frac{N_a}{\sigma_a^2} - \frac{\text{tr}(\mathbf{A}^{-1} \mathbf{C}^{uu})}{(\sigma_a^2)^2} - \frac{\hat{\mathbf{u}}' \mathbf{A}^{-1} \hat{\mathbf{u}}}{(\sigma_a^2)^2} \\ \frac{N - \text{rank}(\mathbf{X})}{\sigma_e^2} - \frac{1}{\sigma_e^2} \left[N_a - \frac{\text{tr}(\mathbf{A}^{-1} \mathbf{C}^{uu})}{\sigma_a^2} \right] - \frac{\hat{\mathbf{e}}' \hat{\mathbf{e}}}{(\sigma_e^2)^2} \end{bmatrix}$$

expensive

AI-REML

- Computationally demanding
- Much faster than EM-REML
 - Fewer iterations
- Provides estimation of standard errors
- BUT
 - For complex models and poor starting values
 - Slow convergence
 - Parameter estimates out of the parameter space
 - In some cases, initial rounds with EM-REML may help

blupf90+

Options for VCE

- AI-REML:

OPTION method VCE

- EM-REML:

OPTION method VCE

OPTION EM-REML xx



Number of EM rounds

Options for VCE in blupf90+

```
OPTION conv_crit 1d-12
```

Convergence criterion (default 1d-10).

```
OPTION maxrounds 1000
```

Maximum rounds (default 5000). When the number = 0, the program calculates BLUP without iterating REML and some statistics (-2logL, AIC, SE for (co)variances, ...).

```
OPTION sol se
```

Store solutions and se.

```
OPTION residual
```

y-hat and residuals will be included in "yhat_residual".

```
OPTION missing -999
```

Specify missing observations (default 0) in integer.

```
OPTION use_yams
```

Run the program with YAMS (modified FSPAK). The computing time can be dramatically improved.

```
OPTION constant_var 5 1 2
```

5: effect number

1: first trait number

2: second trait number

implying the covariance between traits 1 and 2 for effect 5.

Options for VCE in blupf90+

```
OPTION store_pec_pec 6
```

Store triangular matrices of standard errors and its covariances for correlated random effects such as direct-maternal effects and random-regression effects in "pec_pec_bf90".

Heterogeneous residual variances for a single trait

```
OPTION hetres_pos 10 11
```

Specify the column positions of (two) covariables in the data file.

```
OPTION hetres_pol 4.0 0.1 0.1
```

Initial values of coefficients for heterogeneous residual variances using $\ln(a_0, a_1, a_2, \dots)$ to make these values.

To transform back to the original scale, use $\exp(a_0 + a_1 \cdot X_1 + a_2 \cdot X_2)$

log-residual function (Foulley and Quaas, 1995)

Options for airemlf90

```
OPTION se_covar_function <label> <function>
```

<label>

A name for a particular function (e.g., `P1` for phenotypic variance of trait 1, `H2_1` for heritability for trait 1, `rg12` for genetic correlation between traits 1 and 2, ...).

<function>

A formula to calculate a function of (co)variances to estimate SD. All terms of the function should be written with no spaces.

Each term of the function corresponds to (co)variance elements and could include any random effects (G) and residual (R) (co)variances.

`G_eff1_eff2_trt1_trt2`

`R_trt1_trt1`

Examples:

```
OPTION se_covar_function P G_2_2_1_1+G_2_3_1_1+G_3_3_1_1+G_4_4_1_1+R_1_1
```

```
OPTION se_covar_function H2d G_2_2_1_1/(G_2_2_1_1+G_2_3_1_1+G_3_3_1_1+G_4_4_1_1+R_1_1)
```

```
OPTION se_covar_function rg12 G_2_2_1_2/(G_2_2_1_1*G_2_2_2_2)**0.5
```

```
OPTION out_se_covar_function
```

Indicate to store in file samples of (co)variances function for postprocessing (histogram, etc.)

Does REML always converge?

- When the expected variance is very small or the covariance matrix is close to non-positive definite, try the following starting values:
 - much smaller = 0.00001
 - much bigger = 1000
- If `blupf90+` does not converge with AI-REML but converges with EM-REML options using the same data and model:
 - run with EM-REML options again but with a smaller starting value to check the estimate because it could be an artifact

gibbsf90+

gibbs1f90: stores single trait matrices once – fast for multi-trait models

gibbs2f90: gibbs1f90 with joint sampling of correlated effects – Maternal effects and RRM

gibbs3f90: gibbs2f90 with heterogeneous residual variance

thrgibbs1f90: for linear-threshold models

thrgibbs3f90: thrgibbs1f90 with heterogeneous residual variance

Variance Components Estimation

Mixed Model Equations Solver

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

Gibbs Sampling

Bayes Theorem

$$p(\theta|y) = p(y|\theta) p(\theta)$$

Likelihood function
indicates how likely the observations are from a distribution
(with particular parameters)

prior probability of unknown θ

posterior probability of unknown θ with known y

- Basic idea of Gibbs sampling:
- Numerical method to draw samples from a posterior distribution
 - Distribution may not be always explicitly available
 - Draw samples = generate random numbers following a distribution
- The results are random numbers (not theoretical formulas)
- The posterior distribution will be drawn based on the numerical values
 - e.g., a histogram

Gibbs Sampling

Ingredients for Gibbs sampling:

- 1) Theoretical derivation: conditional posterior distribution for each unknown parameter
- 2) Software: a random number generator for a particular distribution

```
# Basic Gibbs sampling for mu (normal) and sigma2 (inverted chi-square)
```

```
y <- c(14,16,18)
```

```
N <- length(y)
```

```
n.samples <- 100
```

```
mu <- rep(0,n.samples)
```

```
sigma2 <- rep(0,n.samples)
```

```
# initial value
```

```
mu[1] <- 0
```

```
sigma2[1] <- 10
```

```
# sampling
```

```
for(i in 2:n.samples){
```

```
  mu[i] <- rnorm(1, mean=mean(y), sd=sqrt(sigma2[i-1]/N)) # using the most recent sigma2
```

```
  df <- N-2
```

```
  S <- sum((y-mu[i])^2)
```

```
  sigma2[i] <- rinvchisq(1, df=df, scale=S) # using the most recent mu
```

```
}
```

gibbsf90+

- Name of parameter file?

`gibbs1.par`

- Number of samples and length of burn-in?

`10000 0`

- Give n to store every n-th sample?

`10`

```
gibbsf90+ gibbs1.par --rounds 10000 --burnin 0 --thin 10
```

gibbsf90+

```
[dani@dodo5 examples]$ gibbsf90+ --help
```

```
*****  
*   GIBBSF90+   *  
*****
```

Gibbs sampler for mixed threshold-linear models involving multiple categorical and linear variables.

Thresholds and variances can be estimated or assumed.

For help about genomics, use `gibbsf90+ --help-genomic`

* OPTION SNP_file snp
Specify the SNP file name to use genotype data.

* OPTION cat 0 0 2 5
"0" indicate that the first and second traits are linear.
"2" and "5" indicate that the third and fourth traits are categorical with 2 (binary) and 5 categories.

* OPTION fixed var all
Store all samples for solutions in `all_solutions` and posterior means and SD for all effects in `final_solutions`
This assumes that (co)variances in the parameter file are known.

gibbsf90+

- Procedure
 - Run `gibbsf90+` to estimate variance components
 - Run `postgibbsf90` to process the samples and check convergence
 - Run `gibbsf90+` with new variance components to estimate breeding values (2k to 10k samples)

```
OPTION fixed_var mean X
```



Number of the
animal effect

gibbsf90+

```
OPTION cat 0 0 2 5
```

"0" indicate that the first and second traits are linear. "2" and "5" indicate that the third and fourth traits are categorical with 2 (binary) and 5 categories.

```
OPTION fixed_var all
```

Store all samples for solutions in "all_solutions" and posterior means and SD for all effects in "final_solutions", assuming that (co)variances in the parameter file are known.

```
OPTION fixed_var all 1 2 3
```

Store all samples for solutions in "all_solutions" and posterior means and SD for 1, 2, and 3 effects in "final_solutions", assuming that (co)variances in the parameter file are known.

```
OPTION fixed_var mean
```

Only posterior means and SD for solutions are calculated for all effects in "final_solutions", assuming that (co)variances in the parameter file are known.

```
OPTION fixed_var mean 1 2 3
```

Only posterior means and SD for solutions are calculated for effects 1, 2, and 3 in "final_solutions", assuming that (co)variances in the parameter file are known.

gibbsf90+

```
OPTION save_halfway_samples n
```

This option can help the 'cold start' (to continue the sampling when the program accidentally stops before completing the run). An integer value n is needed. In every n rounds, the program saves intermediate samples to 2 files (`last_solutions` and `binary_final_solutions`). The program can restart the sampling from the last round where the intermediate files were saved. The program also writes a log file `save_halfway_samples.txt` with useful information for the next run.

To restart, add `OPTION cont 1` to your parameter file and run `gibbsf90+` again. Input 3 numbers (samples, burn-in, and interval) according to `save_halfway_samples.txt`. Gibbsf90+ can take care of all restarting process by itself, so no other tools are needed.

Tips

- Small n will make the program slow because of frequent file writing. The n should be a multiple of the interval (the 3rd number you will input in the beginning of the program).
- If the program stops during burn-in, the restart will fail because `gibbs_samples` is not created. Recommendation is burn-in=0 (but it doesn't provide posterior mean and SD for solutions).
- The cold start may add tiny numerical errors to the samples. Samples from the cold start wouldn't be identical to samples from a non-stop analysis.
- If, unfortunately, the program is killed during its saving the intermediate samples, the cold start will fail. To avoid this, you can manually make a backup for `gibbs_samples`, `fort.99`, `last_solutions`, and `binary_final_solutions` at some point and write them back if needed.

gibbsf90+

```
OPTION cont 10000
```

"10000" is the number of samples run previously. The user can restart the program from the last run. This option requires "binary_final_solutions", "gibbs_samples", and "fort.99" files. When using "OPTION cont", all output files will be replaced by new ones. Before running with this option, all files should be backed up.

```
OPTION prior 5 2 -1 5
```

The (co)variance priors are specified in the parameter file.

Degree of belief for all random effects should be specified using the following structure:

```
OPTION prior eff1 db1 eff2 db2 ... effn dbn -1 dbres
```

effx correspond to the effect number and dbx to the degree of belief for this random effect, -1 corresponds to the degree of belief of the residual variance.

In this example 2 is the degree of belief for the 5th effect, and 5 is the degree of belief for the residual.

```
OPTION seed 123 -432
```

Two seeds for a random number generator can be specified.

```
OPTION thresholds 0.0 1.0 2.0
```

Set the fixed thresholds. No need to set 0 for binary traits.

```
OPTION residual 1
```

Set the residual variance = 1 for categorical traits. For binary traits, the residual variance is automatically set to 1, so no need to use this option.

gibbsf90+

```
OPTION censored 1 0
```

Negative values for the categorical trait in the data set indicate censored records. "1 0" determines that the first categorical trait is censored and the second uncensored.

```
OPTION hetres_int col nlev
```

where col is column in the data file that selects which residual (co)variance to select, and nlev is the maximum number of levels. Different residual (co)variances need to be numbered consecutively starting from 1.

```
OPTION hetres_int 5 10
```

The position "5" to identify the interval in the data file and the number of intervals "10" for heterogeneous residual variances.

gibbsf90+

Parameter file (ex5)

Data (datasire)

```
1 - HYS
2 - sire
3 - y1
4 - heterogeneous clas
5 - y2
```

cat datasire

```
6 13 317.55 1 644.26
3 10 280.44 1 563.05
.....
37 1 270.52 5 543.63
53 10 286.43 5 579.84
```

```
DATAFILE
datasire
NUMBER_OF_TRAITS
NUMBER_OF_EFFECTS
OBSERVATION(S)
WEIGHT(S)
EFFECTS: POSITIONS_IN_DATAFILE
1 1 100 cross
2 2 50 cross
RANDOM_RESIDUAL VALUES
500 100
100 1000
RANDOM_GROUP
RANDOM_TYPE
diagonal
FILE
(CO)VARIANCES
75 10
10 150
OPTION hetres_int 4 5
```

```
round 98
209. 416.
416. 828.
Residual variance, interval 1
df_r 1997 ee/n 99.4738134864675
101. 202.
202. 412.
Residual variance, interval 2
df_r 1997 ee/n 146.518188769043
148. 296.
296. 602.
Residual variance, interval 3
df_r 1997 ee/n 198.183671561078
198. 397.
397. 806.
Residual variance, interval 4
df_r 1997 ee/n 232.307903786663
228. 455.
455. 917.
Residual variance, interval 5
df_r 1997 ee/n 301.189371418363
311. 622.
622. 0.126E+04
```

postgibbsf90

- Basic idea of post Gibbs analysis:
- Summarize and visualize the samples drawn by `gibbsf90+`
- Confirm if the chain converged
- Find the most probable value
 - posterior mode as a “point estimate”
- Find the reliability of the estimates
 - the highest posterior density as a “confidence interval”

postgibbsf90

- Name of parameter file?

`gibbs1.par`

- Burn-in?

`0`

- Give n to store every n-th sample? (1 means read all samples)

`10`

- input files

`gibbs_samples, fort.99`

- output files

`"postgibbs_samples"`

all Gibbs samples for additional post analyses

`"postmean"`

posterior means

`"postsd"`

posterior standard deviations

`"postout"`

postgibbsf90

at least > 10 is recommended
> 30 may be better

number of independent
cycles of Gibbs samples

*****					Monte	Carlo	Error by	Time Series	*****			
Pos.	eff1	eff2	trt1	trt2	MCE	Mean	HPD	Effective	Median	Mode	Independent	
							Interval (95%)	sample size				
1	4	4	1	1	1.362E-02	0.9889	0.7788	1.215	70.4	0.9844	0.9861	18
2	4	4	1	2	1.288E-02	1.006	0.777	1.219	84.1	1.006	0.952	18
3	4	4	2	2	1.847E-02	1.66	1.347	1.987	80.3	1.652	1.579	25
4	0	0	1	1	9.530E-03	24.47	24.07	24.84	425.6	24.47	24.53	2
5	0	0	1	2	8.253E-03	11.84	11.54	12.18	395.8	11.83	11.82	2
6	0	0	2	2	1.233E-02	30.1	29.65	30.58	387.8	30.09	29.97	5

postgibbsf90

```
Choose a graph for samples (= 1) or histogram (= 2); or exit (= 0)
```

```
1
```

```
positions
```

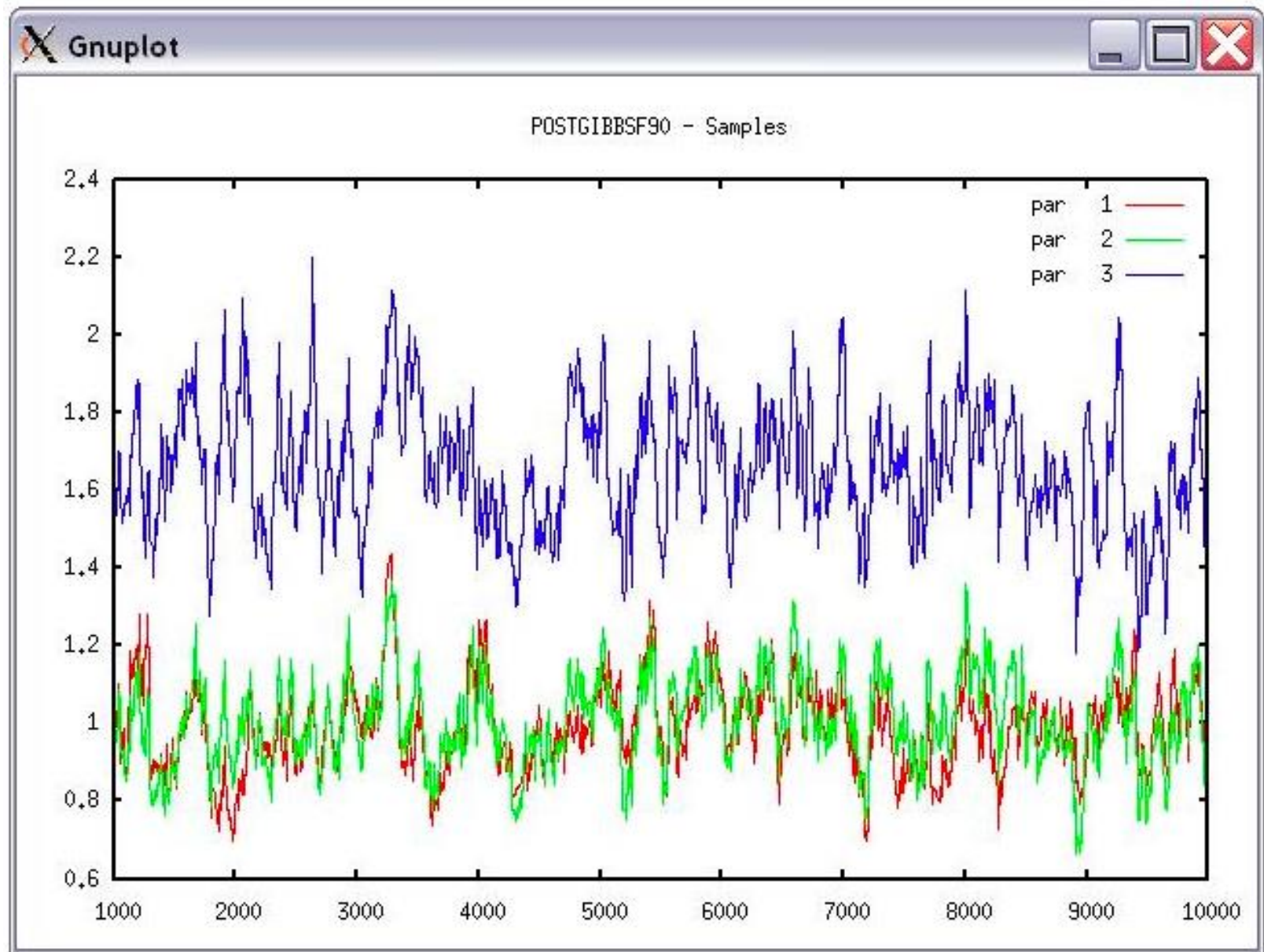
```
1 2 3 # choose from the position numbers 1 through 6
```

If the graph is stable (not increasing or decreasing), the convergence is met.
All samples before that point should be discarded as burn-in.

```
print = 1; other graphs = 2; or stop = 0
```

```
2
```


postgibbsf90



postgibbsf90

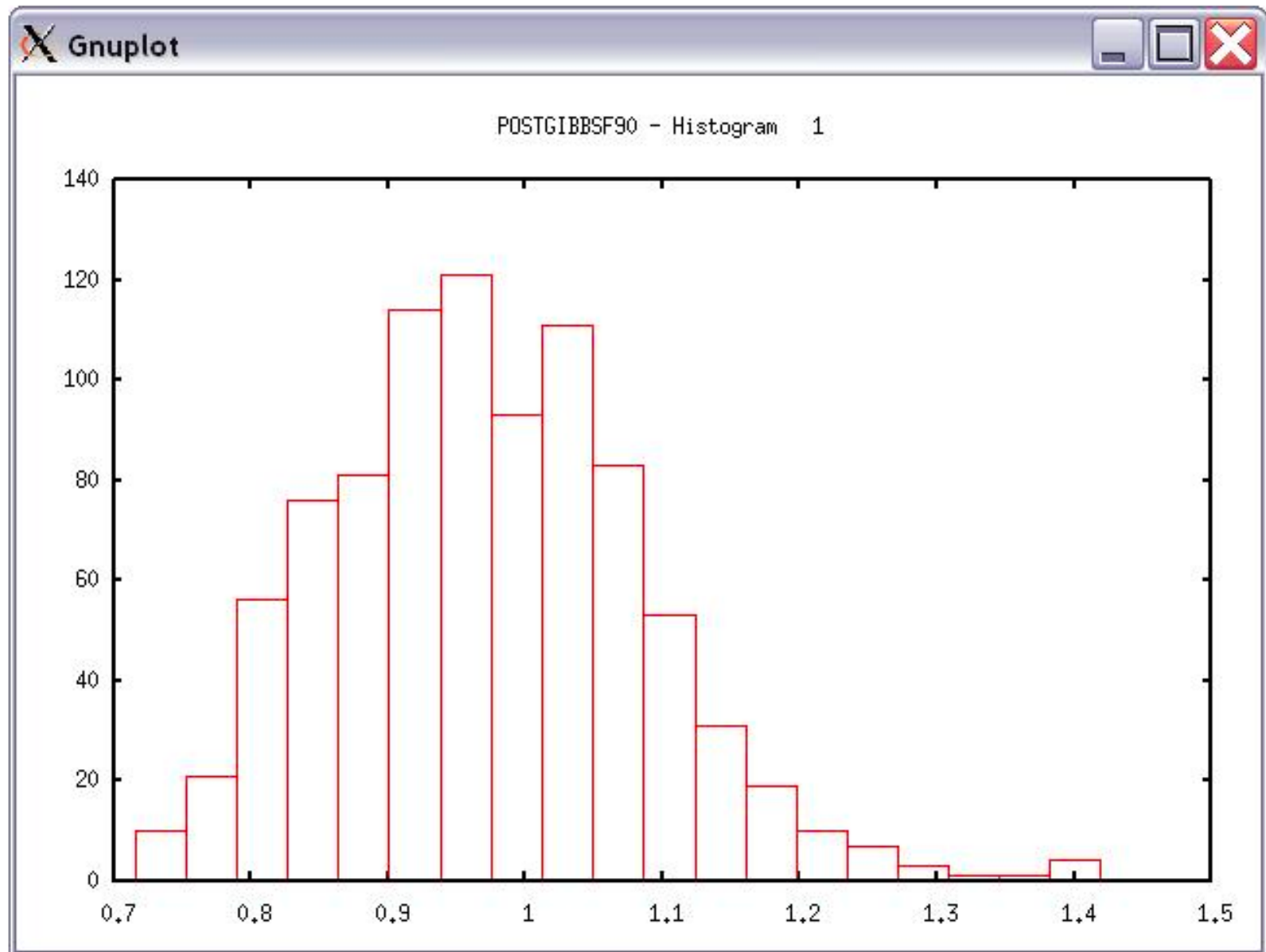
```
Choose a graph for samples (= 1) or histogram (= 2); or exit (= 0)
```

```
2
```

```
Type position and # bins
```

```
1 20
```

postgibbsf90



General output from BLUPF90 family

- Output printed on the screen is not saved to any file!
- Should use redirection or pipes to store output (Linux / MAC)

blupf90+

```
blupf90+ renf90.par | tee blup.log
```

gibbsf90+

```
gibbsf90+ renf90.par | tee gibbs.log
```

Run in background + Save output

```
$vi bp.sh
```

```
#type the following commands inside bp.sh
```

```
blupf90+ <<AA > blup.log
```

```
renf90.par
```

```
AA
```

```
#save and exit
```

```
$bash bp.sh & #can replace bash by sh
```

```
$vi gibbs.sh
```

```
#type the following commands inside gibbs.sh
```

```
gibbsf90+ <<AA > gibbs.log
```

```
renf90.par
```

```
1000 0
```

```
10
```

```
AA
```

```
#save and exit
```

```
$bash gibbs.sh & #can replace bash by sh
```

Common problems for BLUPF90 family

- Wrong position or formats for observation and effects
- Misspelling of Keywords
 - Program may stop
- (Co)variance matrices not symmetric, not positive definite
 - Program may not stop
- Large numbers (e.g. 305-day milk yield 10,000 kg)
 - Scale down i.e. $10,000 / 1,000 = 10$