UNIVERSITY OF GEORGIA
College of Agricultural & Environmental Sciences

# Introduction to BLUPF90 software suite

Daniela Lourenco
UGA USA

Ignacio Aguilar
INIA Uruguay

UGA TEAM – 09/2019

# BLUPF90 software suite

- Collection of software for computations
  - Focus on applications in Breeding and Genetics
- Fortran 90/95
- No GUI (graphical user interface) programs !!!
- Since 1998 by Ignacy Misztal
- First idea: to solve the MME

$$\begin{bmatrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{Z} \\ \mathbf{Z}'\mathbf{X} & \mathbf{Z}'\mathbf{Z}+\mathbf{A}^{-1}\dfrac{\boldsymbol{\sigma}_e^2}{\boldsymbol{\sigma}_a^2} \end{bmatrix} \begin{bmatrix} \hat{\beta} \\ \hat{u} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathrm{y} \\ \mathbf{Z}'\mathrm{y} \end{bmatrix}$$

- First software: blupf90
- Second idea: variance components estimation
- Developers?

# BLUPF90 software developers

Ignacy Misztal

Shogo Tsuruta

Andres Legarra

Ignacio Aguilar

Yutaka Masuda

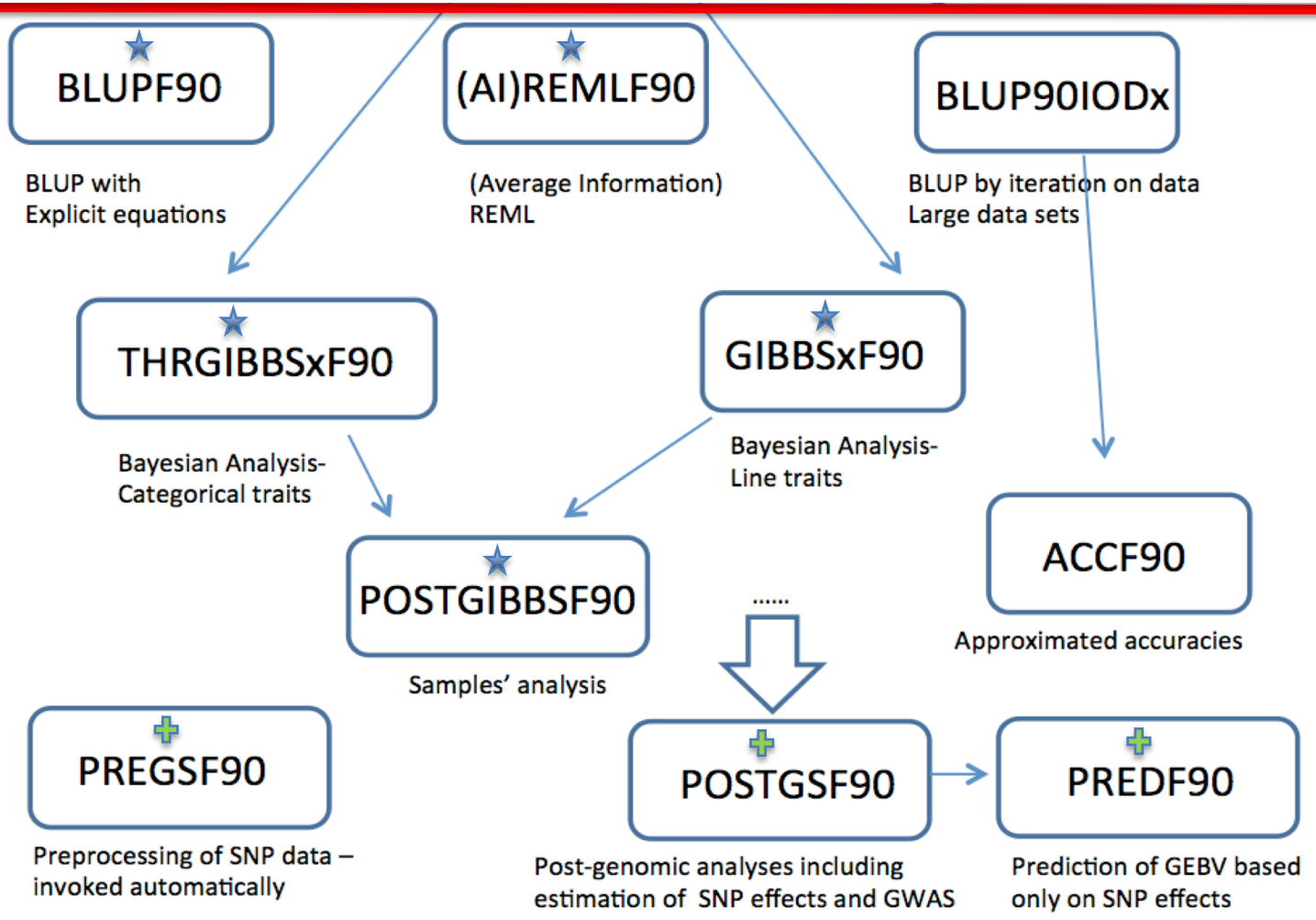
- + Several contributors
- Research turns into code

- Which programs?

Data File for blupf90 family:

a) Only numbers – Integer or real

a) All effects need to be renumbered from 1 to N




**Controlled by the same parameter file!**

# Downloading BLUPF90

**http://nce.ads.uga.edu**

**Set a path for the programs to work in whatever directory**

settings -> control panel-> system -> advanced ->

environment variables -> system variables

- Edit Patch
- ; D:\programs\bin

# blupf90

Mixed Model Equations Solver

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

# 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)

- The solution values change among methods but estimable function should be the same
- Prediction error variances can be obtained using sparse inverse (FSPAK or YAMS)

# Parameter file for blupf90

**Model:** $y = sex + animal + e$

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

EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT[EFFECT NESTED]
 2          2 cross
 3      12010 cross
RANDOM_RESIDUAL VALUES
  0.60000
 RANDOM_GROUP
     2
 RANDOM_TYPE
 add_animal
 FILE
renadd02.ped
(CO)VARIANCES
  0.40000
```

Unlimited number of traits and effects

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

Residual variance

Definition of random effects

# 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)

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 zero (0.0) to indicate uncorrelated residual effects between traits
- e.g. For a 3-trait model

  43.1 0.0 0.0
  0.0 5.1 3.2
  0.0 3.2 10.3

# 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(s) of effect from 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, 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 3 correlated effects (E1-E3)

| | | E1 | | | E2 | | | E3 | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | T1 | T2 | T3 | T1 | T2 | T3 | T1 | T2 | T3 |
| E1 | 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
  - Pedigre file:
    - `individual number, sire number, maternal grandsire number`

- *add_animal*
  - To create a relationship matrix using sire and dam information
  - Pedigre file:
    - `animal number, sire number, dam number`

# RANDOM_TYPE

- *add_an_upg*
  - As before but using rules for unknown parent group
  - Pedigre file:
    - `animal number, sire number, dam number, parent code`
    - missing sire/dam can be replaced by upg number, usually greater than 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
  - Pedigre file:
    - `animal number, sire number, dam number, inb/upg code`
    - missing sire/dam can be replaced by upg number, usually greater than maximum number of animals
    - inb/upg code = 4000 / [(1+md )(1-Fs ) + (1+ms )(1-Fd )]
    - ms (md) is 0 if sire (dam) is known and 1 otherwise
    - Fs(Fs) inbreeding coefficient of the sire (dam)

# RANDOM_TYPE

- *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 at the end of the parameter file
- `OPTION option_name x1 x2 …`
- option_name: each program has its own definition of 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 (deault 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.

# Example of parameter file for blupf90

Single trait "USDA-type" animal model

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

where

$y_{ijkl}$ - production yield
$hys_i$ - fixed herd year season
$hs_{ij}$ - random herd x sire interaction
$p_k$ - random permanent environment
$a_k$ - random animal

and

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

From blupf90.pdf documentation: http://nce.ads.uga.edu/wiki/doku.php

```
# 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

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

where

$y_{ijkl}$ - production yield

$hys_i$ - fixed herd year season

$hs_{ij}$ - random herd x sire interaction

$p_k$ - random permanent environment

$a_k$ - random animal

and

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

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


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
```

Parameter file

data file

Number of levels for each effect

```
 read            6  records in   6.1703999E-02  s,
nonzeroes
  read           11  additive pedigrees
 finished peds in   6.1760999E-02  s,
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"
```

Records read from data file

Records read from pedigree file

Solutions file

# File *"solutions"*

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

where

$y_{ijkl}$ - production yield

$hys_i$ - fixed herd year season

$hs_{ij}$ - random herd x sire interaction

$p_k$ - random permanent environment

$a_k$ - 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"
```

# remlf90
# and
# airemlf90

Variance components estimation

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

# remlf90 and airemlf90

- REML = restricted/residual maximum likelihood
  - Patterson and Thompson (1971)

- Most used method to estimate variance components in breeding and genetics

- BLUPF90 family has 2 REML programs
  - remlf90: expectation-maximization (EM) algorithm
  - airemlf90: average information (AI) algorithm

- remlf90 and airemlf90 use the same parameter file as blupf90

# remlf90

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

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

Inverse of LHS for animal effect

# animals (rank of A)

- The equations contain BLUE and BLUP but those values are calculated with known variance components

- This method requires iterations:

1. set initial variance components
2. compute $\hat{\beta}$ and $\hat{u}$ via mixed model equations
3. update and variance components with above equations
4. go to 1 or stop if the parameters do not change any more

Instructions from Masuda (2018)

# remlf90

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

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

# airemlf90

Vector of variance components

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

Gradient (score vector)

Hessian Matrix

Average-information algorithm uses this matrix as Hessian,

$$\mathbf{H}(\theta) = \mathcal{I}_A(\theta) = \begin{bmatrix} -\frac{1}{2}\mathbf{y}'\mathbf{PZAZ}'\mathbf{PZAZ}'\mathbf{Py} & -\frac{1}{2}\mathbf{y}'\mathbf{PZAZ}'\mathbf{PPy} \\ -\frac{1}{2}\mathbf{y}'\mathbf{PPZAZ}'\mathbf{Py} & -\frac{1}{2}\mathbf{y}'\mathbf{PPPy} \end{bmatrix}$$

**Gradient**

expensive

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

Instructions from Masuda (2018)

# airemlf90

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

# Options for remlf90 and airemlf90

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

Convergence criterion (default 1d-10).

```
OPTION maxrounds 10000
```

Maximum rounds (default 5000).

```
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 airemlf90

```
OPTION EM-REML 10
```

Run EM-REML (REMLF90) for first 10 rounds to get initial variances within the parameter space (default 0).

```
OPTION tol 1d-12
```

Tolerance (or precision) (default 1d-14) for positive definite matrix and g-inverse subroutines.
Convergence may be much faster by changing this value.

```
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".

**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*(a0, a1, a2, …) to make these values.

To transform back to the original scale, use exp(a0+a1*X1+a2*X2)

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
```

# 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 AIREMLF90 does not converge but REMLF90 converges with the same data set and the same model:
  - run REMLF90 again but with a small starting value to check the estimate because it could be artifact

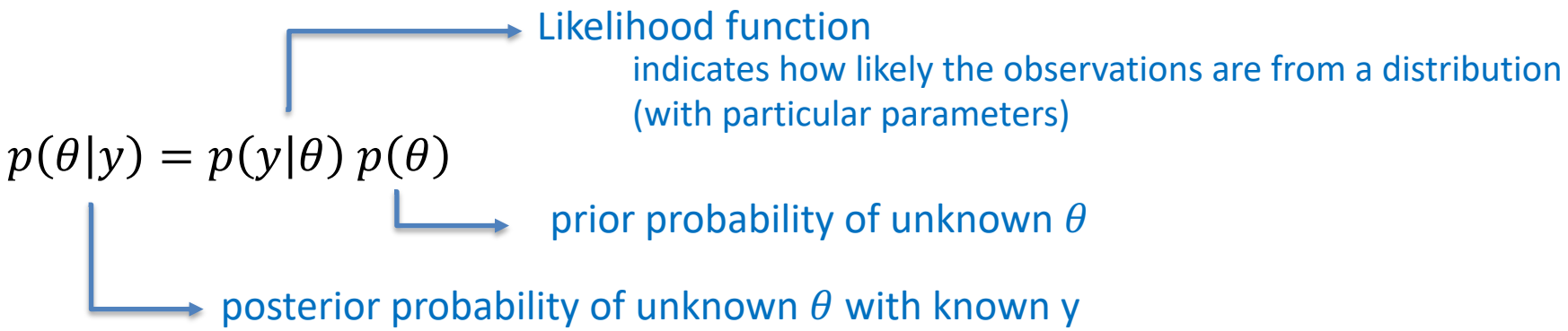
  - use an option to use EM-REML inside AI-REML:

    `OPTION EM-REML xx`

    where xx is the number of rounds of EM

# gibbsf90

Bayes Theorem




- Basic idea of Gibbs sampling:
- Gibbs sampling is a numerical method to draw samples from a posterior distribution (not 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 (like a histogram)

Instructions from Masuda (2018)

# gibbsf90

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
}
```

Instructions from Masuda (2018)

# gibbsXf90

- gibbs1f90: faster for multiple trait models
- gibbs2f90: better for correlated random effects
- gibbs3f90: for heterogeneous residual variance

- Name of parameter file?

  gibbs1.par
- Number of samples and length of burn-in?

  samples=10,000 to 100,000; burn-in=0
- Give n to store every n-th sample?

  10

```
gibbs1f90 gibbs1.par --rounds 10000 --burnin 0 --thin 10
```

# gibbsXf90

- Procedure
  - Run gibbsXf90 to estimate variance components
  - Run postgibbsf90 to process the samples and verify convergence
  - Run gibbsXf90 with new variance components to estimate breeding values (2k to 10k rounds)

# gibbsXf90

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

All solutions and posterior means and SD for effects for effects1, 2, and 3 are stored in "all_solutions" and in "final_solutions" every round using fixed variances. Without numbers, all solutions for all effects are stored.

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

Posterior means and SD for effects1, 2, and 3 in "final_solutions".

```
OPTION solution all 1 2 3
```

Caution: this option will create a huge output solution file when you run many rounds and/or use a large model. All solutions and posterior means and SD for effects1, 2, and 3 are stored in "all_solutions" and in "final_solutions" every round. Without numbers, all solutions for all effects are stored.

```
OPTION solution mean 1 2 3
```

Caution: this option will create a huge output solution file when you run many rounds and/or use a large model. Posterior means and SD for effects1, 2, and 3 in "final_solutions".

```
OPTION cont 10000
```

10000 is the number of samples run previously when restarting the program from the last run.

# gibbsXf90

```
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 123 321
```

Two seeds for a random number generator can be specified.

```
OPTION SNP_file snp
```

Specify the SNP file name to use genotype data.

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

# thrgibbsXf90

- thrgibbs1f90: Gibbs sampler for mixed threshold-linear models
  Thresholds and variances can be estimated or assumed
- thrgibbs3f90: for heterogeneous residual variance

- Name of parameter file?
  gibbs1.par
- Number of samples and length of burn-in?
  samples=10,000 to 100,000; burn-in=0
- Give n to store every n-th sample?
  10

```
thrgibbs1f90 gibbs1.par --rounds 10000 --burnin 0 --thin 10
```

# thrgibbsXf90

## Options

```
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 save_halfway_samples 5000
```

The program saves every "5000" samples to restart or recover the job right after the last saved samples. It is useful when the program accidentally stopped.

To restart, add `OPTION cont 1` to your parameter file and run `thrgibbs1f90` again

```
OPTION thresholds 0.0 1.0 2.0
```

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

```
OPTION residual 1
```

The residual variance can be set to 1 but not necessary for categorical traits more than 2 categories. For binary traits, the residual variance is automatically set to 1, so no need to use this option.

# postgibbsf90

- Basic idea of post Gibbs analysis:
- Summarize and visualize the samples drawn by gibbsXf90
- 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”

Instructions from Masuda (2018)

# 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 Interval (95%) | | Effective sample size | Median | Mode | Independent chain 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 |

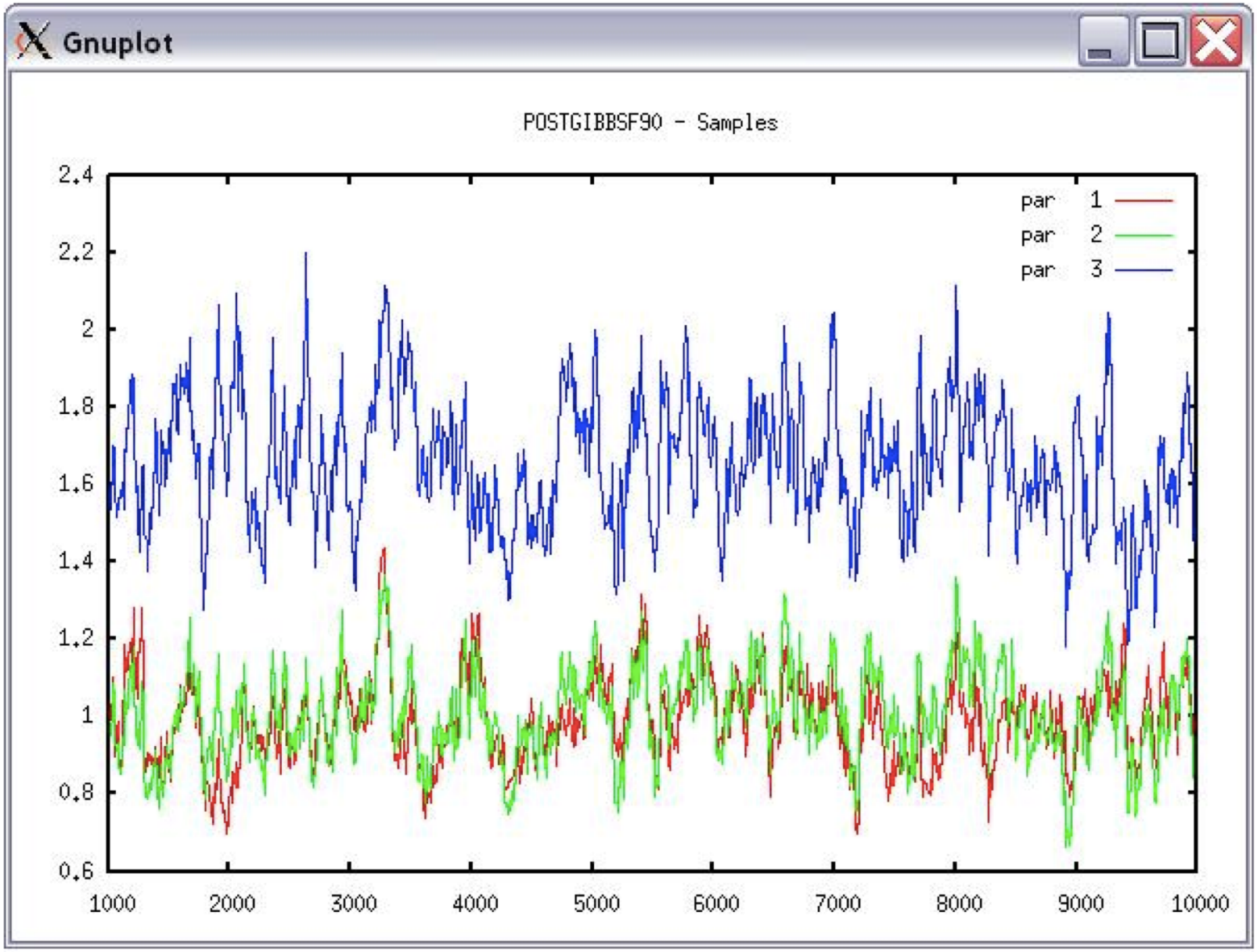
******** P Lower and upper bounds of Mean ± 1.96PSD :io ratio between first half and second half of the samples ; should be < 1.0

| Pos. | eff1 | eff2 | trt1 | trt2 | PSD | Mean | PSD Interval (95%) | | Geweke diagnostic | Autocorrelations lag: 1 | 10 | 50 | Independent # batches |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4 | 4 | 1 | 1 | 0.1144 | 0.9889 | 0.7648 | 1.213 | -0.02 | 0.853 | 0.188 | 0.049 | 50 |
| 2 | 4 | 4 | 1 | 2 | 0.1182 | 1.006 | 0.7742 | 1.237 | -0.11 | 0.828 | 0.111 | -0.066 | 50 |
| 3 | 4 | 4 | 2 | 2 | 0.1656 | 1.66 | 1.335 | 1.984 | 0.06 | 0.828 | 0.108 | -0.021 | 36 |
| 4 | 0 | 0 | 1 | 1 | 0.1967 | 24.47 | 24.09 | 24.86 | -0.01 | 0.034 | 0.029 | -0.062 | 450 |
| 5 | 0 | 0 | 1 | 2 | 0.1643 | 11.84 | 11.51 | 12.16 | 0.03 | 0.032 | -0.006 | -0.016 | 450 |
| 6 | 0 | 0 | 2 | 2 | 0.2429 | 30.1 | 29.62 | 30.57 | -0.02 | 0.07 | -0.014 | 0.037 | 180 |

# 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
```

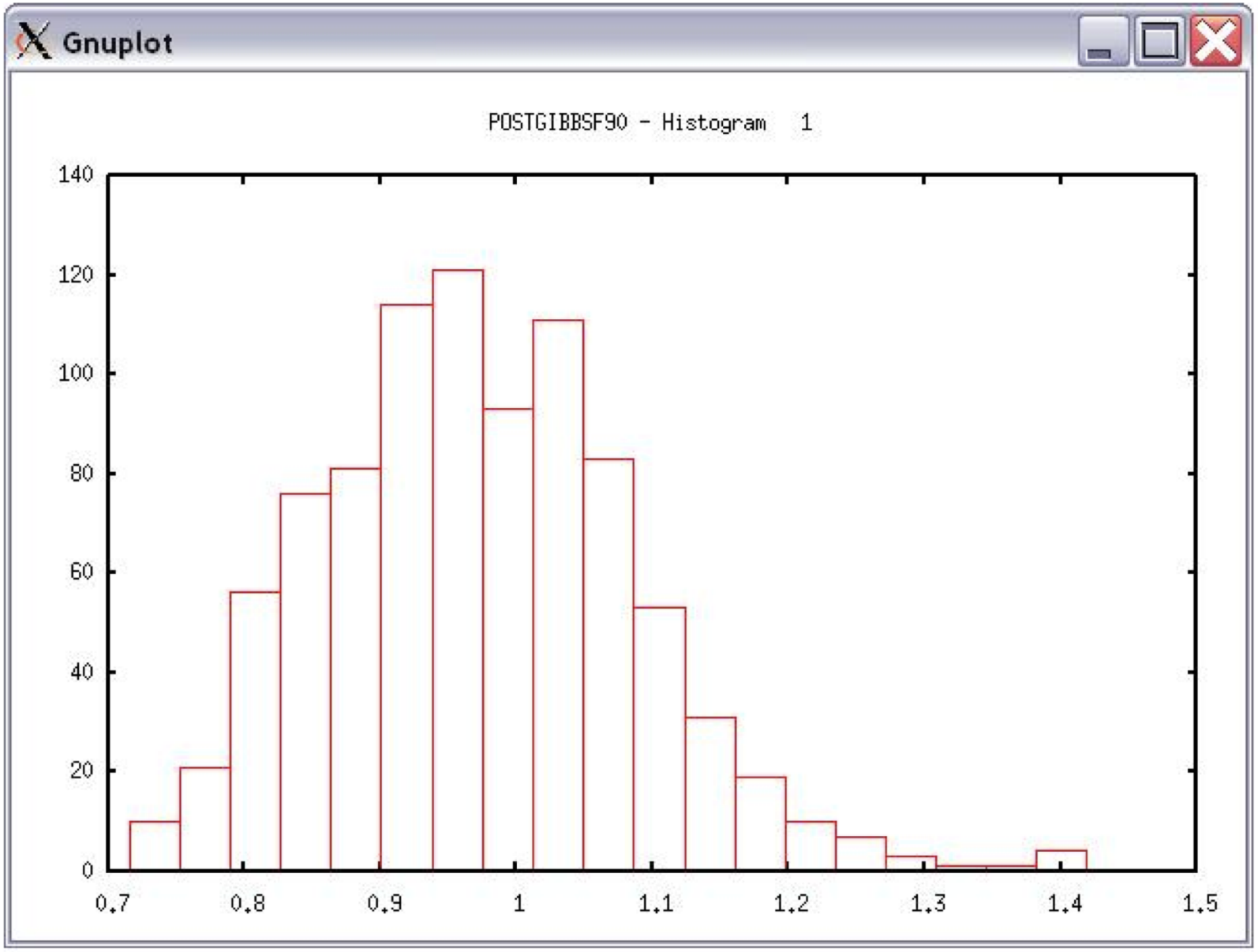
# postgibbsf90



# postgibbsf90

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

Type position and # bins
1 20
```

# postgibbsf90



# 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