Using BLUPF90

UGA 05-2018

BLUPF90 family programs

- All programs are controled by the SAME paramenter file.
- Extra options could be used to set non-default behaviour of each program
- Understanding parameter file usually solve most of problems

BLUPF90 parameter file

```
# Each keyword can be preceded by comments, each starting with
#
DATAFILE
name of data file
NUMBER_OF_TRAITS
number of traits
NUMBER_OF_EFFECTS
number of effects
OBSERVATION(S)
position of observations in data file (one per trait)
WEIGHT(S)
position of weight(s) in data file (one per trait); blank if
all weights equal
EFFECTS: POSITIONS_IN_DATAFILE NUMBER_OF_LEVELS TYPE_OF_EFFECT
[EFFECT NESTED]
one line per effect:
     position of effect (one per trait)
     number of levels
     type of effect (one of cross for crossclassified or cov
for covariable)
     position of effect where nested (one per trait, optional)

RANDOM_RESIDUAL VALUES
residual variance covariance matrix (full stored)
RANDOM_GROUP
number of one random effect or list of correlated effects
(order as in EFFECTS above); correlated effects must be
consecutive
RANDOM_TYPE
type of random effect:   one of diagonal, add_animal,add_sire,
                         add_an_upg, add_an_upginb, user_file,
                         user_file_i or par_domin
FILE
relationship file; blank line if file unnecessary
(CO)VARIANCES
variance covariance matrix for given correlated effects and
traits(full stored)
```

Repeat for each Random effect

Data file

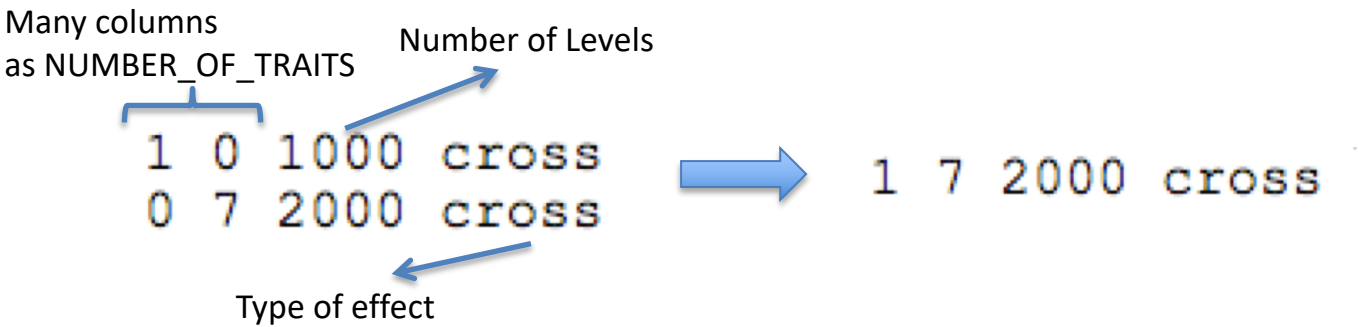
- Free format, i.e. at least one space to separate columns
- TABs are not valid to separate columns
- Some programs (MS Excel) export flat files with TAB separators !!
- Only numbers: integer or reals
- If reals decimal separators “.” not “,”
- One “.” is not a missing value
- All effect need to be renumber from 1 consecutively (see later RENUMF90)

Number of traits / effects

- No restriction for number of traits or effects
- But memory requirements and computing time increase exponentially with them

Effects section

- Many rows as the NUMBER_OF_EFFECTS
- In this section de model for each trait is defined
- Different models per trait are supported
- If an effect is missing for one trait use 0



RANDOM_RESIDUAL VALUES

- This matrix should 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

43.1	0.0	0.0
0.0	5.1	3.2
0.0	3.2	10.3

Random effect definition

- RANDOM_GROUP
  - Number(s) of effect from list of effects
  - Correlated effects should be consecutive e.g. Maternal effects, Random Regression models
- 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 number_of_traits*number_of_correlated_effects

(CO)VARIANCES structure

- 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 enviroment effects, assume 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 maximun number of animals
    - Parent code = 3 – nb 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 maximun 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*
  - a matrix is read from file. Matrix is stored only upper- or lower-triang
  - Matrix file:
    - `row, col, value`
- *user_file_i*
  - As before but the matrix will be inverted
- *par_domin*
  - A parental dominance file created by program RENDOM
  - File format
    - `s-d s-sd s-dd ss-d ds-d ss-sd ss-dd ds-sd ds-dd code`

Pedigree files

- As with data files pedigree files are separated by at least one SPACE!!
- TABs are not supported !!
- Order of columns depends on the type of the random effect
- Duplicates pedigree are not checked!!
- Identification number need to be coded sequentially from 1 to the maximum number of animals
- No order is required !!!!

Programs Options

- Programs behavior can be modified by adding lines with OPTION at the end of the parameter file

- `OPTION option_name x1 x2 …`

- option_name, each program has it own definition of options
- The number of optional parameters (x1, x2…) to control the behavior depends on the option.

BLUPF90

- Blupf90 computes generalized solutions by several methods:
  - Preconditioner Conjugate Gradient (PCG). Default Iterative method, fast.
  - Succesive over-relaxation (SOR), a iterative method based on Gauss-Seidel
  - Direct solution using sparse Cholesky factorization (FSPAK) Greater memory requirements

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

BLUPF90 options

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

Set convergence criteria (deault 1e-12).

```
OPTION maxrounds 10000
```

Set maximum number of rounds (default 1000).

```
OPTION solv_method FSPAK
```

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

Example of parameter file BLUPF90

Single trait "USDA-type" animal model. This example is from the documentation of program JAA20.

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

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

Parameter File

```
DATAFILE
ic
NUMBER_OF_TRAITS
1
NUMBER_OF_EFFECTS
4
OBSERVATION(S)
5
WEIGHT(S)

EFFECTS: POSITIONS_IN_DATA
2 3 cross
3 6 cross
4 4 cross
1 14 cross
RANDOM_RESIDUAL VALUES
1
RANDOM_GROUP
2
RANDOM_TYPE
diagonal
FILE

(CO)VARIANCES
.1
RANDOM_GROUP
3
RANDOM_TYPE
diagonal
FILE

(CO)VARIANCES
.05
RANDOM_GROUP
4
RANDOM_TYPE
add_an_upg
FILE
is
(CO)VARIANCES
.5
```

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

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

Data file ic

Format: animal/hys/p/hs/y

```
1  1  1  1  10
2  1  2  1  11
3  2  3  2  15
4  2  4  3  13
5  3  5  4  14
6  3  6  3  12
```

Relationship file (is)

Format: animal/dam/sire/code

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

How to run BLUPF90 program

```
$blupf90
 name of parameter file?
exiap
```

- Output that is printed to the terminal is not SAVED in any file !!!
- Use redirection or pipes to store outputs in log files:

```
echo exiap | blupf90 | tee blup.log
```

Output from BLUPf90

```
     BLUPF90 1.51

 Parameter file:                exiap
 Data file:                     ic
 Number of Traits                1
 Number of Effects               4
 Position of Observations        5
 Position of Weight (1)          0
 Value of Missing Trait/Observation          0

EFFECTS
 #  type                  position (2)        levels
    1  cross-classified         2                3
    2  cross-classified         3                6
    3  cross-classified         4                4
    4  cross-classified         1               14
```

Parameter file

data file

Number of levels for Each effect

```
 read            6  records in   6.6707999E-02  s,
  nonzeroes
  read           11  additive pedigrees
 finished peds in   6.6791996E-02  s,
round =     1  convergence =  0.1475
round =     2  convergence =  0.2884E-01
round =     3  convergence =  0.1361E-02
round =     4  convergence =  0.4133E-03
round =     5  convergence =  0.2570E-03
round =     6  convergence =  0.1100E-03
round =     7  convergence =  0.6376E-04
round =     8  convergence =  0.3953E-04
round =     9  convergence =  0.1128E-03
round =    10  convergence =  0.1812E-03
round =    11  convergence =  0.3401E-04
round =    12  convergence =  0.1759E-04
round =    13  convergence =  0.4981E-05
round =    14  convergence =  0.8271E-05
round =    15  convergence =  0.3786E-04
round =    16  convergence =  0.2849E-03
round =    17  convergence =  0.2078E-04
round =    18  convergence =  0.6162E-06
round =    19  convergence =  0.5674E-08
round =    20  convergence =  0.5511E-11
round =    21  convergence =  0.4846E-14
   21 iterations,   convergence criterion= 0.4846E-14
 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
ic
NUMBER_OF_TRAITS
1
NUMBER_OF_EFFECTS
4
OBSERVATION(S)
5
WEIGHT(S)

EFFECTS: POSITIONS_IN_DATA
2 3 cross
3 6 cross
4 4 cross
1 14 cross
```

trait	effect	level	solution
1	1	1	10.23612694
1	1	2	12.09158350
1	1	3	13.08139319
1	2	1	-0.00804515
1	2	2	0.00804485
1	2	3	-0.01608983
1	2	4	0.01609003
1	2	5	0.03218022
1	2	6	-0.03218023
1	3	1	-0.00000015
1	3	2	-0.00804504
1	3	3	-0.00804510
1	3	4	0.01609012
1	4	1	-0.14763082
1	4	2	0.67537821
1	4	3	3.09345225
1	4	4	0.73947039
1	4	5	0.54853421
1	4	6	-0.71936576
1	4	7	2.52868986
1	4	8	1.49838702
1	4	9	5.59197718
1	4	10	-1.13020008
1	4	11	2.52868981
1	4	12	-1.61263546
1	4	13	9.74590627
1	4	14	-4.72875256

Different Models with BLUPF90

faq [BLUPF90]

nce.ads.uga.edu/wiki/doku.php?id=faq

[[faq]]

BLUPF90

Search

Trace: • tricks • faq.parameter • faq.renum_example • readme.blupf90 • distribution • readme.gibbs1 • application_programs • start • faq.example • faq

FAQ: examples and options for BLUPF90 family programs

Table of Contents

- FAQ: examples and options for BLUPF90 family programs
  - Frequent mistakes in data, pedigree, and parameter files for BLUPF90 programs
  - Data preparation program
  - REML
  - Gibbs sampler
  - Post-Gibbs Analysis
  - BLUP
  - Accuracy
  - Genomic Analysis

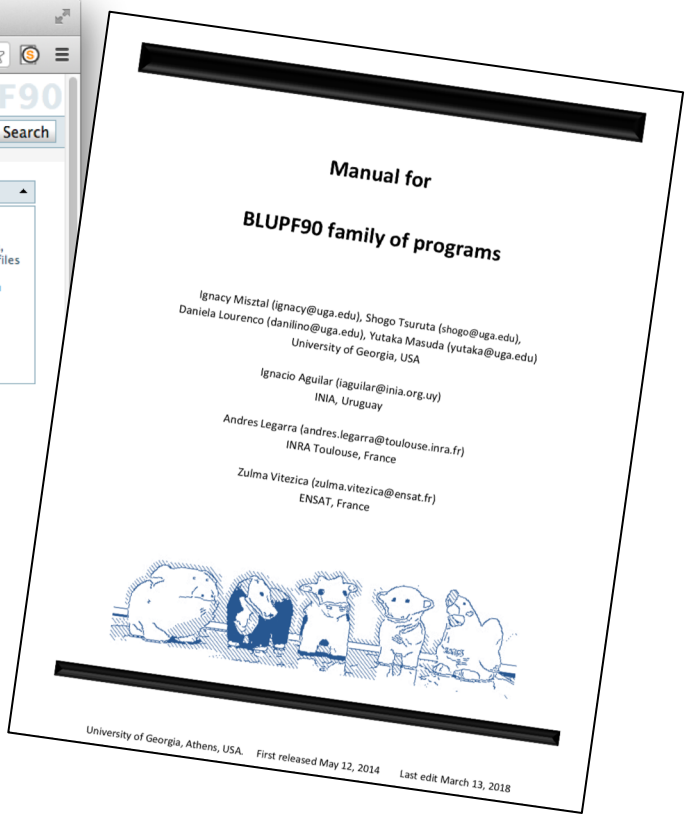
Frequent mistakes in data, pedigree, and parameter files for BLUPF90 programs

- data
- pedigree
- parameter
- examples

Data preparation program

- renumf90
- examples

http://nce.ads.uga.edu/wiki/doku.php?id=faq



Manual for

BLUPF90 family of programs

Ignacy Misztal (ignacy@uga.edu), Shogo Tsuruta (shogo@uga.edu),
Daniela Lourenco (danilino@uga.edu), Yutaka Masuda (yutaka@uga.edu)
University of Georgia, USA

Ignacio Aguilar (iaguilar@inia.org.uy)
INIA, Uruguay

Andres Legarra (andres.legarra@toulouse.inra.fr)
INRA Toulouse, France

Zulma Vitezica (zulma.vitezica@ensat.fr)
ENSAT, France

University of Georgia, Athens, USA. First released May 12, 2014 Last edit March 13, 2018