Package 'BIGf90'

June 26, 2024

Title R face front for running K-
fold crossvalidation and estimating ebvs and variance components with Blupf90 modules

Version 0.1.0

Authors Josue Chinchilla-Vargas, Alexander M. Sandercock and Breeding Insight Team

Maintainer Josue Chinchilla-Vargas <jc3635@cornell.edu>

Description
This package provides R functions to run several BLUPf90 modules. Along with an R function to run K-fold crossvalidation for univariate analyses through blupf90+.
The output table from the K-fold crossvalidation function calculates accuracy as cor(y,ebv^) and cor(y*,ebv^) along with bias of ebvs calculated as reg(y,ebv.)
You will need to create the .par file to feed into Renumf90 manually. Once this is done, there are functions to run Renumf90, Blupf90+, Predictf90, Gibbsf90+ and Postgibbsf90.
If you need to learn how to use the blupf90 suite of programs refer to nce.ads.uga.edu/wiki/doku.php?id=start. Please remember to cite Blupf90 appropriately along with this package.

License Apache 2.0

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Imports base (>= 4.3.1),
dplyr (>= 1.1.4),
utils (>= 4.3.1)

R topics documented:

bf90_cv *Run K-fold cross-validation analysis (CVA)*

Description

This function runs a K-fold cross-validation analysis (CVA) using blupf90 modules.

Usage

```
bf90_cv(
  path_2_execs,
  missing_value_code,
  random_effect_col,
  h2,
  num_runs,
  num_folds,
  output_table_name,
  renf90_ped_name
)
```

Arguments

path_2_execs	path to a folder that holds all blupf90 executables that ill be used (blupf90+,predictf90). This field should be in quotes "".
missing_value_code	code used in the .par file after OPTION MISSING to indicate missing phenotype, if this option is no use, this value must be 0.
random_effect_col	Column where random effects are located, found under RANDOM_GROUP in the renf90.par file.
h2	estimate of narow-sense heritabilty.This value is use to calculate accuracy of ebvs
num_runs	Number of independent cross-valdation runs to be performed.
num_folds	Number of folds to be generated within each independent run.
output_table_name	Name of the final tab-separated out-up file. This field should be in quotes "".
renf90_ped_name	Name of pedigree file generated by renumf90. This field should be in quotes "".

Details

This function sets up and runs a K-fold cross-validation analysis (CVA) using blupf90+ and predictf90. The function run_renumf90 needs to be used beforehand to process a .par file created by the user. This function calculates 2 accuracy estimates: correlation between raw phenotypes and ebvs divided by the square-root of narrow sense heritabilty and correlation between corrected phenotypes and ebvs along with bias estimations calculated as the regression of the phenotypes on the ebvs.

Value

a tab-separated file that includes accuracy and bias estimates of ebvs.

Examples

```
## Example for a CVA with 5 independent runs dividing the data in 10 folds.


# bf90_cv(path_2_execs = "/Users/johndoe/Desktop/bf90_execs/",
#      missing_value_code = -999,
#      random_effect_col= 3,
#      h2 = 0.5,
#      num_runs = 5,
#      num_folds = 10,
#      output_table_name = "example_run",
 #     renf90_ped_name = "renadd03.ped")

#The function will output the file "example_run" with a layout:
# Metric Run Value
# y-ebv_correlation run 1 0.038
# y-ebv_correlation run 2 0.053
# y-ebv_correlation run 3 0.112
# y-ebv_correlation run 4 0.075
# y-ebv_correlation run 5 0.089
# y*-ebv_correlation run 1 0.947
# y*-ebv_correlation run 2 0.964
# y*-ebv_correlation run 3 0.893
# y*-ebv_correlation run 4 0.883
# y*-ebv_correlation run 5 0.939
# bias run 1 0.085
# bias run 2 0.117
# bias run 3 0.262
# bias run 4 0.164
# bias run 5 0.189
# y-ebv_average_corr  0.074
# y*-ebv_accuracy  0.925
# y_corrected_accuracy (yraw_average_corr/sqrt(h2))  0.105
# average_bias  0.163
```

---

run_blup *Run blupf90+*

Description

This function runs blupf90+ using a pre-processed parameter file called renf90.par.

Usage

```
run_blup(path_2_execs)
```

Arguments

path_2_execs	path to a folder that holds the renumf90 executable. This field should be in quotes "".

Details

This function runs blupf90+ using a parameter file named renf90.par. Since this function depends only on the renf90.par parameter file, the only input needed from the user is a path where the blupf90+ executable is located.

Examples

```
## Example

# run_blup(path_2_execs = "/Users/johndoe/Desktop/bf90_execs/")
```

`run_gibbs` *Run gibbsf90+*

Description

This function runs gibbsf90+.

Usage

```
run_gibbs(path_2_execs, gibbs_iter, gibbs_burn, gibbs_keep)
```

Arguments

`path_2_execs`	path to a folder that holds the renumf90 executable. This field should be in quotes "".
`gibbs_iter`	number of samples in the Gibbs sampler.
`gibbs_burn`	number of samples to be discarded at the begining of the Gibbs sampler
`gibbs_keep`	the interval to save samples (thinning). Entering a 1 means all samples are kept.

Details

This function runs gibbsf90+ using input from the user and a renf90.par file. This function is written to use a renf90.par paramererfile, therefore the function run_renumf90 needs to be used beforehand to process a .par file created by the user. The user will have to enter the number of samples in the MCMC chain, the number of samples to burn and the number used to thin samples.

Examples

```
## Example

# run_gibbs( path_2_execs = "/Users/johndoe/Desktop/bf90_execs/",
# gibbs_iter = 250000,
# gibbs_burn = 20000
# gibbs_keep = 1)
```

run_postgibbs *Run postgibbsf90*

Description

This function runs postgibbsf90 using the renf90.par file used to run gibbsf90+

Usage

```
run_postgibbs(path_2_execs, postgibbs_burn, postgibbs_keep)
```

Arguments

path_2_execs	path to a folder that holds the renumf90 executable. This field should be in quotes "".
postgibbs_burn	number of samples to be discarded at the begining of the Gibbs sampler
postgibbs_keep	the interval to save samples (thinning). Entering a 1 means all samples are kept.#'

Details

This function runs postgibbsf90 to provide diagnostic statistics,posterior means and standard deviations for analyses performed through gibbsf90+. This function is written to run using the same renf90.par file used with run_gibbs and its output files are be the standard output files produced by postgibbsf90. The user will have to enter the number of samples to burn and the number used to thin samples.

Examples

```
## Example

# run_postgibbs( path_2_execs = "/Users/johndoe/Desktop/bf90_execs/",
# postgibbs_burn =1,
# postgibbs_keep = 100)
```

run_predict *Run predictf90*

Description

This function runs predict90.

Usage

```
run_predict(path_2_execs)
```

Arguments

path_2_execs	path to a folder that holds the renumf90 executable. This field should be in quotes "".

Details

This function runs predictf90 using a pre-processed parameter file called renf90.par to calculate adjusted phenotypes. This function is written to use a renf90.par file and a solutions file. Therefore run_renum and run_blup should be ran before using this function. The output files are be the standard output files produced by predictf90

Examples

```
## Example

# run_predict(path_2_execs = "/Users/johndoe/Desktop/bf90_execs/")
```

`run_renum` *Run renumf90*

Description

This function runs renumf90.

Usage

```
run_renum(path_2_execs, raw_par_file)
```

Arguments

`path_2_execs`	path to a folder that holds the renumf90 executable. This field should be in quotes "".
`raw_par_file`	name of the .par file that will be processed. This field should be in quotes "".

Details

This function runs renumf90 to process the raw parameter(.par) file to be used with the blupf90 suite of programs. The outputs will be the standard output files produced by renumf90.

Examples

```
## Example

# run_renum(path_2_execs = "/Users/johndoe/Desktop/bf90_execs/",
 raw_par_file = "weight_2022_no_cov_cv.par")
```