PS R functions

V Segura

27 septembre 2018

Contents

R functions used in Rincent *et al.* for evaluating the predictive abilities of SNP and NIRS through cross-validations: example of use with the poplar dataset.

Setup

PS functions

Sourcing PS functions from github:

```r
source("https://raw.githubusercontent.com/visegura/PS/master/rfuncPS.r")
```

Dependencies

Making use of the very cool `anyLib` package to check and install all dependencies **except** `emma`:

```r
install.packages("anyLib")
```

```
## Installing package into '/home/visegura/R/x86_64-pc-linux-gnu-library/3.5'
## (as 'lib' is unspecified)
```

```r
library(anyLib)
anyLib(c("doParallel", "MASS", "corpcor", "BGLR"))
```

```
## Loading required package: doParallel
```

```
## Loading required package: foreach
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```
## Loading required package: MASS
## Loading required package: corpcor
```

```
## Loading required package: BGLR
```

```
## # Package Bayesian Generalized Regression (BGLR), 1.0.5.
```

```
## # Gustavo de los Campos & Paulino Perez-Rodriguez
```

```
## # Support provided by the U.S., National Institutes of Health (NIH)
```

```
## # (Grant: R01GM101219, NIGMS)
```

```
## # and by the International Maize and Wheat Improvement Center (CIMMyT).
```

```
## # Type 'help(BGLR)' for summary information
```

```
## doParallel       MASS    corpcor       BGLR 
##       TRUE       TRUE       TRUE       TRUE
```

The `emma` package (Kang *et al.*, 2008) required here is different from the one available on CRAN. It can be installed from github as follows:

```
install.packages("https://github.com/Gregor-Mendel-Institute/mlmm/files/1356516/emma_1.1.2.tar.gz",
                 repos = NULL)
```

```
## Installing package into '/home/visegura/R/x86_64-pc-linux-gnu-library/3.5'
## (as 'lib' is unspecified)
```

```
anyLib("emma")
```

```
## Loading required package: emma
```

```
## emma 
## TRUE
```

Example dataset

The present example is carried out on the poplar dataset for the trait “Bud Set” evaluated in the Orléans trial. The dataset is publicly available in the INRA datapartage repository and can be accessed with the following link: http://dx.doi.org/10.15454/MB4G3T

```
Sys.setenv("DATAVERSE_SERVER" = "data.inra.fr")
anyLib(c("dataverse", "data.table", "apercu"))
```

```
## Loading required package: dataverse
```

```
## Loading required package: data.table
```

```
## data.table 1.11.4  Latest news: http://r-datatable.com
```

```
## Loading required package: apercu
```

```
##  dataverse data.table     apercu 
##       TRUE       TRUE       TRUE
```

Phenotype

```
writeBin(get_file("Phenotyping_Poplar.txt", "doi:10.15454/MB4G3T"), "phenot.txt")
phenot <- fread("phenot.txt", header = TRUE, data.table = FALSE)
ap(phenot)
```

```
##   Accession   HT-ORL CIRC-ORL CIRC-SAV   BF-ORL
## 1     1-A01 200.0467 5.048141 10.60961 2.166667
## 2     1-A02 326.2046 8.072041 11.24455 2.833333
## 3     1-A03 284.2197 7.111861 13.51628 3.000000
## 4     1-A06 211.4417 7.592933 11.19221 4.000000
## 5     1-A07 211.2536 5.170886 11.03278 3.333333
```

```
phen <- phenot[, "BS-ORL"]
names(phen) <- phenot[, "Accession"]
ap(phen)
```

```
##    1-A01    1-A02    1-A03    1-A06    1-A07
## 1.583333 2.000000 1.833333 1.833333 1.333333
```

NIRS

Orléans design

```
writeBin(get_file("NIRS_NormDer_Wood_Poplar_ORL.txt", "doi:10.15454/MB4G3T"), "NIRS_Orl.txt")
NIRS_Orl <- fread("NIRS_Orl.txt", header = TRUE, data.table = FALSE)
ap(NIRS_Orl)
```

```
##   Accession         4000         4002          4004          4006
## 1     1-A01 0.0008952447 0.0005262395  1.572343e-04 -0.0002117709
## 2     1-A02 0.0012715782 0.0008863018  5.010253e-04  0.0001157489
## 3     1-A03 0.0007606739 0.0003887431  1.681220e-05 -0.0003551187
## 4     1-A06 0.0006994197 0.0003399435 -1.953264e-05 -0.0003790088
## 5     1-A07 0.0008290945 0.0004676450  1.061955e-04 -0.0002552541
```

```
NIRSOrl <- as.matrix(NIRS_Orl[, -1])
rownames(NIRSOrl) <- NIRS_Orl[, "Accession"]
ap(NIRSOrl)
```

```
##               4000         4002          4004          4006          4008
## 1-A01 0.0008952447 0.0005262395  1.572343e-04 -0.0002117709 -0.0005807761
## 1-A02 0.0012715782 0.0008863018  5.010253e-04  0.0001157489 -0.0002695276
## 1-A03 0.0007606739 0.0003887431  1.681220e-05 -0.0003551187 -0.0007270495
## 1-A06 0.0006994197 0.0003399435 -1.953264e-05 -0.0003790088 -0.0007384850
## 1-A07 0.0008290945 0.0004676450  1.061955e-04 -0.0002552541 -0.0006167036
```

Savigliano design

```
writeBin(get_file("NIRS_NormDer_Wood_Poplar_SAV.txt", "doi:10.15454/MB4G3T"), "NIRS_Sav.txt")
NIRS_Sav <- fread("NIRS_Sav.txt", header = TRUE, data.table = FALSE)
ap(NIRS_Sav)
```

```
##   Accession        4000         4002         4004         4006
## 1     1-A01 0.001286716 0.0008886659 0.0004906154 9.256484e-05
## 2     1-A02 0.001887066 0.0014602839 0.0010335014 6.067190e-04
## 3     1-A03 0.001550515 0.0011453745 0.0007402341 3.350938e-04
## 4     1-A06 0.001675321 0.0012682111 0.0008611011 4.539911e-04
## 5     1-A07 0.001347113 0.0009435133 0.0005399138 1.363144e-04
```

```
NIRSSav <- as.matrix(NIRS_Sav[, -1])
rownames(NIRSSav) <- NIRS_Sav[, "Accession"]
ap(NIRSSav)
```

```
##              4000         4002         4004         4006          4008
## 1-A01 0.001286716 0.0008886659 0.0004906154 9.256484e-05 -3.054857e-04
## 1-A02 0.001887066 0.0014602839 0.0010335014 6.067190e-04  1.799365e-04
## 1-A03 0.001550515 0.0011453745 0.0007402341 3.350938e-04 -7.004653e-05
## 1-A06 0.001675321 0.0012682111 0.0008611011 4.539911e-04  4.688116e-05
## 1-A07 0.001347113 0.0009435133 0.0005399138 1.363144e-04 -2.672850e-04
```

SNP

```
writeBin(get_file("Genotyping_Poplar.txt", "doi:10.15454/MB4G3T"), "Genot.txt")
Genot <- fread("Genot.txt", header = TRUE, data.table = FALSE)
ap(Genot)
```

```
##   Accession SNP_IGA_1_3090809 SNP_IGA_1_3127827 SNP_IGA_1_3437307
## 1     1-A01                 1               1.0               0.0
## 2     1-A02                 1               1.0               0.5
## 3     1-A03                 1               0.5               0.0
## 4     1-A06                 1               0.5               0.0
## 5     1-A07                 1               1.0               0.0
##   SNP_IGA_1_3478569
## 1               0.5
## 2               0.0
## 3               0.0
## 4               0.0
## 5               0.0
```

```
SNP <- 2*as.matrix(Genot[, -1])
rownames(SNP) <- Genot[, "Accession"]
ap(SNP)
```

```
##       SNP_IGA_1_3090809 SNP_IGA_1_3127827 SNP_IGA_1_3437307
## 1-A01                 2                 2                 0
## 1-A02                 2                 2                 1
## 1-A03                 2                 1                 0
## 1-A06                 2                 1                 0
## 1-A07                 2                 2                 0
##       SNP_IGA_1_3478569 SNP_IGA_1_3478647
## 1-A01                 1                 1
## 1-A02                 0                 0
## 1-A03                 0                 1
## 1-A06                 0                 1
## 1-A07                 0                 1
```

Calibrations

Calibration are made with 3 different predictor matrices : NIRS at Orléans, NIRS at Savigliano and SNP. For the first 2 matrices (NIRS) a ridge regression model is carried out while for the thrid one (SNP) both a GBLUP and a Bayesian LASSO models are used.

In all cases we use a repeated cross-validation scheme with 5 folds and 20 repetitions.

NIRS Orléans

```
NIRSOrl_RidgeBLUP <- RidgeBLUP(Y = phen, X = NIRSOrl, fold = 5, iter = 20, cores = 10,
                               lambda = seq(from = 1, to = 2001, by = 20))
```

Cross-validation statistics:

```
colMeans(NIRSOrl_RidgeBLUP[["CVstats"]])
```

```
##        R2     RMSEP       Rho
## 0.2721463 0.4567273 0.5279883
```

NIRS Savigliano

```
NIRSSav_RidgeBLUP <- RidgeBLUP(Y = phen, X = NIRSSav, fold = 5, iter = 20, cores = 10,
                               lambda = seq(from = 1, to = 2001, by = 20))
```

Cross-validation statistics:

```
colMeans(NIRSSav_RidgeBLUP[["CVstats"]])
```

```
##        R2     RMSEP       Rho
## 0.1913721 0.4814235 0.4408810
```

SNP

GBLUP

```
SNP_GBLUP <- GBLUP(Y = phen, X = SNP, fold = 5, iter = 20, cores = 10)
```

Cross-validation statistics:

```
colMeans(SNP_GBLUP[["CVstats"]])
```

```
##        R2     RMSEP       Rho
## 0.5387859 0.3635815 0.7340976
```

Bayesian LASSO

Note that we use the genomic heritability estimates previously computed with GBLUP as a prior for the trait heritability.

```
SNP_GBLUP$h2
```

```
## [1] 0.8346909
```

```
SNP_BL <- BL(Y = phen, X = SNP, fold = 5, iter = 20, cores = 10,
             nIterBGLR = 30000, burnInBGLR = 5000, thinBGLR = 1,
             h2 = SNP_GBLUP$h2)
```

Cross-validation statistics:

```r
colMeans(SNP_BL[["CVstats"]])
```

```
##        R2     RMSEP       Rho 
## 0.5482714 0.3598188 0.7408250
```