BayesRCO User Guide

Version 0.0.1

Fanny Mollandin[1], Pascal Croiseau[1], Andrea Rau[1,2]

[1] Université Paris-Saclay, INRAE, AgroParisTech, GABI
[2] BioEcoAgro Joint Research Unit, INRAE, Université de Liège, Université de Lille, Université de Picardie Jules Verne

May 25, 2021

Contents

1 Overview

The BayesRCO software includes five different Bayesian genomic prediction models, including three state-of-the art approaches and two novel algorithms:

- BayesCπ (Habier et al., 2011)
- BayesR (Erbe et al., 2012)
- BayesRC (MacLeod et al., 2016)
- BayesRCπ
- BayesRC+

All five models are Bayesian Gaussian mixture models for the genomic prediction of complex traits using genetic variation such as single nucleotide polymorphisms (SNPs), with parameters estimated using a Markov Chain Monte Carlo (MCMC) algorithm. These prediction methods also facilitate a study of the underlying genomic architecture of these traits, in particular by enabling QTL mapping. The two new methods implemented in BayesRCO, BayesRC+ and Bayesπ, both aim to integrate prior categorizations of SNPs arising from multiple, potentially overlapping annotations.

This document is intended to describe the underlying models of BayesRCO, provide help for download and compilation of the software, and describe the various inputs, outputs, and options provided by the software.

> *Note*: The core of the BayesRCO software is based on version 0.75 of the BayesR software described and implemented by Moser et al. (2015), although further functionalities and outputs have been added (including options for the BayesRCπ and BayesRC+ algorithms). As many of the input arguments in BayesRCO are the same as those of BayesR, there are many similarities between this document and the BayesR User Manual.

2 Bayesian genomic prediction models

2.1 SNP effect prior distributions

All five Bayesian genomic prediction models included in BayesRCO exploit the same underlying linear model, which aims to obtain an accurate prediction of a vector of phenotypes **y** by best estimating a vector of SNP effects β:

$$\mathbf{y} = \mu\mathbb{1}_n + \mathbf{X}\boldsymbol{\beta} + \mathbf{e},$$
$$\mathbf{e} \sim N(0, \mathbf{I}_n\sigma_e^2)$$

The five Bayesian models included in BayesRCO can be differentiated by the prior distribution attributed to $\boldsymbol{\beta}$, as indicated in the table below. In each model, SNP effects are assumed to follow a Gaussian mixture distribution with varying numbers of components: 2 (null and non-null) for BayesCπ, or 4 (null, low, medium and high) for all of the other methods. In addition, three of the models (BayesRC, BayesRCπ and BayesRC+) additionally incorporate a prior known categorization of SNPs (e.g., according to functional information, or lists of candidate or causal mutations).

Method	SNP effect prior distribution	# effect classes	Prior Annotations
BayesCπ	$\beta_i \sim \pi\mathcal{N}(0,0) + (1-\pi)\mathcal{N}(0,\sigma_\beta^2)$	2	No
BayesR	$\beta_i \sim \sum_{K=1}^{4}\pi_K\mathcal{N}(0,k\sigma_g^2)$	4	No
BayesRC	$\beta_i \vert a = A(i) \sim \sum_{K=1}^{4}\pi_{K,a}\mathcal{N}(0,k\sigma_g^2)$	4	Yes, disjointed
BayesRC+	$\beta_i \vert a \in \mathbf{A}(i) \sim \sum_{a\in\mathbf{A}(i)}\sum_{K=1}^{4}\pi_{K,a}N(0,k\sigma_g^2)$	4	Yes, overlapping
BayesRCπ	$\beta_i \vert a \in \mathbf{A}(i) \sim \sum_{a\in\mathbf{A}(i)}p_{i,a}\sum_{K=1}^{4}\pi_{K,a}N(0,k\sigma_g^2)$	4	Yes, overlapping

where σ_g^2 is the total additive genetic variance, $\pi = (\pi_1, \pi_2, \pi_3, \pi_4)$ the mixing proportions such that $\sum_{i=1}^{4}\pi_i = 1$, $p_{i,a}$ the mixing proportions of SNP i in its set of annotations $\mathbf{A}(i)$ such that $\sum_{a\in\mathbf{A}(i)}p_{a,i} = 1$, and $k = \{0, 10^{-5}, 10^{-4}, 10^{-3}\}$.

2.2 Gibbs sampler algorithm

As an exact computation of the posterior distribution is intractable for this set of models, Bayesian inference is performed in all cases by obtaining draws from the posterior distribution using a Gibbs sampler. Model parameters are subsequently estimated using the posterior mean across iterations, after excluding a burn-in phase and thinning draws. By default, the Gibbs sampler runs for a total of 50,000 iterations, including 20,000 as a burn-in and a thinning rate of 10.

2.3 Novelty of BayesRCπ and BayesRC+

We developed BayesRCπ and BayesRC+ as an extension of BayesRC to handle cases where prior categorizations of SNPs are overlapping rather than disjointed (i.e., where SNPs can potentially be assigned to multiple categories).

In the case of BayesRCπ, SNP effects are assumed to follow a mixture of mixtures distribution; that we assume that SNPs follow a mixture distribution over their corresponding annotation categories, and within a given annotation in turn, SNP effects are modeled with a 4-component Gaussian mixture distribution as in the BayesR model. Concretely, within a given iteration of the Gibbs sampler used for estimation, SNPs are assigned to the annotation category which maximizes its likelihood given the current estimates of the other model parameters. Note that this step is analogous to that in the standard BayesR algorithm of assigning SNPs to one of the four SNP effect classes based on a likelihood calculation and the current estimates of model parameters.

In the case of BayesRC+, we assume that multiple annotation categories cumulatively impact the estimate of SNP effects; that is, we assume that multiple annotation categories should have an additive effect of estimated SNP effects.

3 Download & Compilation

The core of the BayesRCO software is based on version 0.75 of the BayesR software by Moser et al. (2015). As such, a very similar file structure is used:

- *RandomDistributions.f90*: auxiliary file containing various random generator
- *baymodsRCO.f90*: support module for BayesRCO containing common variables and routines (note: unchanged from version 0.75 of BayesR)
- *bayesRCO.f90*: main program

BayesRCO can be compiled with a FORTRAN95 compiler on a Unix operating system using the following command:

```
gfortran RandomDistributions.f90 baymodsRCO.f90 bayesRCO.f90 -o bayesRCO
```

4 Inputs

4.1 Data

BayesRCO requires PLINK binary ped file format. It requires *.bim and *.fam files to determine the number of SNPs and the number of individuals, and a *.bed file for the genotype information.

Genotype data: BayesRCO requires genotypes in PLINK binary format in default-SNP major mode. Since BayesRCO includes all genotypes in the model, samples missing a genotype call cannot simply be omitted. Missing genotypes are replaced by the mean genotype value of a given marker.

Phenotype data: The program reads column 6 as the phenotype column from a PLINK *.fam file. A different phenotype column can be specified by using the –n `[num]` option, where –n `1` uses the original 6th column (default), –n `2` uses column 7 and so forth. Missing phenotypes (or phenotypes to be predicted) must be coded as `NA`.

4.2 Prior annotation categories for SNPs

We can represent SNP annotation categories as a binary design matrix, with SNPs in rows and annotation categories in columns. We differentiate two types of annotation matrix, non-overlapping (for BayesRC) or potentially overlapping (for BayesRC+ or BayesRCπ). An example of a **non-overlapping annotation** matrix, such that all SNPs are assigned to a single annotation, is as follows:

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ . & . & . \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

An example of an **overlapping** annotation matrix, such that all SNPs are assigned to *at least* one annotation, is as follows:

$$\begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \\ . & . & . \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$$

In the latter example, the first SNP has been categorized as belonging to annotations 2 and 3, while the second SNP has been categorized as belonging only to annotation 2.

As recommended by MacLeod et al. (2016), it is important to have sufficiently large annotation categories ($\geq$ 1000 SNPs) to avoid difficulties to estimate the π_a parameters.

4.3 General Inputs

Input	Description	Default
-bfile	prefix PLINK binary files	None
-out	prefix for output	None
-n	phenotype column	1
-vara	SNP variance prior	0.01
-vare	error variance prior	0.01
-dfvara	degrees of freedom Va	-2.0
-dfvare	degrees of freedom Ve	-2.0
-delta	prior for Dirichlet	1.0
-msize	number of SNPs in reduced update	0
-mrep	number of full cycles in reduced update	5000
-numit	length of MCMC chain	50000
-burnin	burnin steps	20000
-thin	thinning rate	10
-ndist	number of mixture distributions	4
-gpin	effect sizes of mixtures (% x Va)	0.0,0.0001,0.001,0.01
-seed	initial value for random number	0
-predict	perform prediction	f
-snpout	output detailed SNP info	f
-cat	output SNP categories per iteration	None
-beta	output SNP effect per iteration	None
-permute	permute order of SNP	f
-model	model summary file (for prediction)	None
-freq	SNP frequency file (for prediction)	None
-param	SNP effect file (for prediction)	None
-ncat	number of SNP annotations	1
-catfile	SNP annotation matrix file	None
-additive	run BayesRC+	f
-bayesCpi	run BayesCπ	f

4.4 Running BayesRCO

BayesRCO is run in two steps: a first for the training data (and thus the estimation of model parameters) and a second for prediction; we thus use two separate datasets, one including phenotype values, and one without (be careful, the SNPs must match between the two datasets!). We illustrate here how to launch these two features. By default, the software runs a BayesRCπ model.

4.4.1 BayesRCπ

```
path/bayesRC0 -bfile [prefix_learning] -out
[prefix_learning] -ncat [number of annotations]
-catfile [annotation_matrix]

path/bayesRC0 -bfile [prefix_validation] -out
[prefix_validation] -model [prefix_learning].model
-freq [prefix_learning].frq -param
[prefix_learning].param -ncat [number of annotations]
-catfile [annotation_matrix]
```

4.4.2 BayesRC+

To run BayesRC+, use the flag `-additive` in the training step:

```
path/bayesRC0 -bfile [prefix_learning] -out
[prefix_learning] -ncat [number of annotations]
-catfile [annotation_matrix] -additive

path/bayesRC0 -bfile [prefix_validation] -out
[prefix_validation] -model [prefix_learning].model
-freq [prefix_learning].frq -param
[prefix_learning].param -ncat [number of annotations]
-catfile [annotation_matrix]
```

4.4.3 BayesRC

As BayesRC is a special case of BayesRCπ or BayesRC+ where no SNPs are assigned to more than one prior annotation category, you can simply run either of the two previous methods with the appropriate disjoint annotation matrix.

4.4.4 BayesR

As BayesR is a special case of BayesRC with a single prior annotation category to which all SNPs are assigned, it can be run in the same way as for BayesRC using an annotation matrix corresponding to a vector (the same length as the number of SNPs) of 1's. In this case, as the default number of `ncat` is 1, there is no need to specify this option.

4.4.5 BayesCπ

Finally, BayesCπ can be run in a similar manner as for BayesR with the additional flag `-bayesCpi`.

4.4.6 Options

Prior distributions for variance components: Prior inverted-chi squared distribution can be specified for both additive and residual variances (σ_g^2 and ^2_e). Scale and degrees of freedom (df) for the variance components are required. Flat (improper) distributions can be specified by setting `df` to -2. It is also possible to specify the heritability of the trait by setting `dfvara` to -3.0 (i.e. `-dfvara -3.0`). In this case the scale parameter is treated as the heritability and the SNP-based variance is set (fixed) to σ_g^2=heritability$\times\sigma_y^2$ (σ_y^2 being the phenotypic variance).

Effect size Dirichlet prior (all): The default is to use a uniform and almost uninformative prior for the mixture distribution with a pseudo-observation of 1 (SNP) for each class. Different priors can be specified using the `delta [num]` option. For example, `-delta 3,2,1` specifies a prior with 3, 2 and 1 pseudo-observations for classes 1 to 3 of a 3-component mixture model, `-delta 2` sets the prior to 2 for all mixture components.

Annotation Dirichlet prior (BayesRCπ) For the moment there is no parameter to change the value of the annotation assignment prior. Such an option may be added in future versions.

Mixture model: The BayesR, BayesRC, BayesRC+ and BayesRCπ models assume that the true SNP effect is derived from a series of normal distributions. The default models uses 4 mixture distributions with SNP variances of 0, 0.0001, 0.001 and 0.01, so that the variance (S) of the jth SNP has 4 possible values: S1=0, S2=0.0001$\times\sigma_g^2$, S3=0.001$\times\sigma_g^2$, S4=0.01$\times\sigma_g^2$. Different mixture models can be specified using the `–ndist [num]` and `–gpin [num]` options. For example, `–ndist 3 –gpin 0.0,0.001,0.05` fits a 3 component mixture with SNP variances S1=0, S2=0.001$\times\sigma_g^2$, S3=0.05$\times\sigma_g^2$.

MCMC sampling: The default is to use a chain length of 50,000 samples (`–numit`) with the first 20,000 samples (`–burnin`) being discarded, and using every 10th sample (`–thin`) for posterior inference. To improve mixing, one can use the option `–permute` to update SNP effects in random order.

5 Outputs

The outputs all have the same name as specified when launching the software, followed by a suffix corresponding to their type, as follows:

5.1 Frequency file

name_output.type: One column containing the SNP allele 2 frequency.

5.2 Log File

The file name prefix is as specified by –`out [prefix_training]`. The suffix '.log' is appended to give the file name. This is a descriptive file and provides a summary of the run parameters used and the number of records processed.

5.3 Frequency File

Contains allele frequency of the '2' allele. The suffix '.frq' is appended to the prefix. This file is required for scaling and centering genotypes for prediction analysis. The SNP order has to be the same as the genotype input file.

5.4 Model File

The suffix 'model' is appended to the output prefix. This file contains means of the posterior samples of model parameters:

Mean: intercept

Nsnp: number of SNPs in model

Va: genetic variance explained by SNPs (σ_g^2)

Nk1_1,...,Nkk_j: residual variance (σ_e^2)

Pk1_1,...,Pkk_j: proportion of SNPs in mixture component 1 to *k* and annotation 1 to *j*

Vk1_1,...,Vkk_j: sum of squares of SNP effects in mixture component 1 to *k* and annotation 1 to *j*

5.5 Hyperparameter file

The file *prefix.hyp* gives posterior parameter estimates for each MCMC sample:

Replicate: iteration number

Nsnp: number of SNPs in model

Va: genetic variance explained by SNPs

Ve: residual variance

Nk1_1,...,Nkk_j: number of SNPs in mixture components 1 to *k* and annotation 1 to *j*

Pk1_1,...,Pkk_j: proportion of SNPs in mixture component 1 to *k* and annotation 1 to *j*

Vk1_1,...,Vkk_j: sum of squares of SNP effects in mixture component 1 to *k* and annotation 1 to *j*

5.6 Parameter file

The suffix 'param' is appended to the output prefix. The SNP order is the same as the genotype input file. This file contains mean posterior estimates for each individual SNP:

PIP_1, ..., PIP_k: Posterior inclusion probabilities of the SNP in mixture classes 1 to *k*

beta: posterior SNP effect

PAIP_1, ..., PAIP_j: Posterior annotation inclusion probabilities of the SNP in annotation 1 to *j* (useful for BayesRCπ, otherwise gives the annotations each SNP belong to)

Vbeta: variance of the posterior SNP effects across iterations

Vi: posterior variance of the SNP effects

5.7 Genetic value file

This file outputs the predicted genomic values (GVs). The output prefix is used to give the file name *prefix.gv*.

5.8 Optional files

-snpout: provide output in sparse format `mixture class:SNP:effect size`. The SNP number (SNP #) corresponds to the row number of the SNP in the PLINK *.bim file.

-cat: output SNP categories per iteration

-beta: output SNP effect per iteration

References

M. Erbe, B. Hayes, L. Matukumalli, S. Goswami, P. Bowman, C. Reich, B. Mason, and M. Goddard. Improving accuracy of genomic predictions within and between dairy cattle breeds with imputed high-density single nucleotide polymorphism panels. *Journal of Dairy Science*, 95(7):4114–4129, July 2012. ISSN 00220302. doi: 10.3168/jds.2011-5019. URL https://linkinghub.elsevier.com/retrieve/pii/S0022030212003918.

D. Habier, R. L. Fernando, K. Kizilkaya, and D. J. Garrick. Extension of the bayesian alphabet for genomic selection. *BMC Bioinformatics*, 12 (1):186, Dec. 2011. ISSN 1471-2105. doi: 10.1186/1471-2105-12-186. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-12-186.

I. M. MacLeod, P. J. Bowman, C. J. Vander Jagt, M. Haile-Mariam, K. E. Kemper, A. J. Chamberlain, C. Schrooten, B. J. Hayes, and M. E. Goddard. Exploiting biological priors and sequence variants enhances QTL discovery and genomic prediction of complex traits. *BMC Genomics*, 17 (1):144, Dec. 2016. ISSN 1471-2164. doi: 10.1186/s12864-016-2443-6. URL http://www.biomedcentral.com/1471-2164/17/144.

G. Moser, S. H. Lee, B. J. Hayes, M. E. Goddard, N. R. Wray, and P. M. Visscher. Simultaneous Discovery, Estimation and Prediction Analysis of Complex Traits Using a Bayesian Mixture Model. *PLOS Genetics*, 11(4):e1004969, Apr. 2015. ISSN 1553-7404. doi: 10.1371/journal.pgen.1004969. URL https://dx.plos.org/10.1371/journal.pgen.1004969.

Funding

This work is part of the GENE-SWitCH project that has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the grant agreement number 817998.