Package ‘bbmix’

July 6, 2023

Title Bayesian Model for Genotyping using RNA-Seq

Version 1.0.0

Description The method models RNA-seq reads using a mixture of 3 beta-binomial distributions to generate posterior probabilities for genotyping bi-allelic single nucleotide polymorphisms. Elena Vigorito, Anne Barton, Costantino Pitzalis, Myles J. Lewis and Chris Wallace (2023) <doi:10.1093/bioinformatics/btad393> ``BBmix: a Bayesian beta-binomial mixture model for accurate genotyping from RNA-sequencing.''

License GPL-2

Encoding UTF-8

Biarch true

Depends R (>= 3.4.0)

Imports methods, Rcpp (>= 0.12.0), rstan (>= 2.18.1), R.utils, data.table, rmutil

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

SystemRequirements GNU make

RoxygenNote 7.1.0

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation yes

Author Elena Vigorito [aut, cre],
Chris Wallace [aut]

Maintainer Elena Vigorito <`elena.vigorito@mrc-bsu.cam.ac.uk`>

Repository CRAN

Date/Publication 2023-07-06 14:00:09 UTC

R topics documented:

bbmix-package *The 'bbmix' package.*

Description

Bayesian Beta-Binomial mixture model for RNA-seq genotyping

References

Stan Development Team (2018). RStan: the R interface to Stan. R package version 2.18.2. https://mc-stan.org

call_gt *Call genotypes using beta binomial after model training*

Description

Call genotypes using beta binomial after model training

Usage

```
call_gt(
  allele_counts_f,
  depth = 10,
  stan_f = NULL,
  legend_f,
  pop = "EUR",
  prob = 0.99,
  fisher_f = NULL,
  fisher = 30,
  cluster_f = NULL,
  out
)
```

Arguments

`allele_counts_f` vector with file names with allele counts for SNPs

`depth` min read count to call variant

`stan_f` full name to stan object with model fit to extract mean of parameters. Defaults to the model trained with genome in a bottle reads. Otherwise this object can be generated with fit_bb function.

`legend_f` full name for file with SNP info to get allele frequency for prior

`pop` population to select AF for GT prior, defaults to EUR

`prob` cut-off for making hard calls, defaults to 0.99

`fisher_f` file with Fisher test to detect strand bias

`fisher` cut_off for Fisher test to detect strand bias

`cluster_f` file with info about SNP clusters

`out` character with file name to save genotype output

Value

data table with genotype probabilities

Examples

```
## Retrive input files for running call_gt
counts_f <- system.file("extdata/input", "NA12878.chr22.Q20.allelicCounts.txt",
package = "bbmix",
mustWork = TRUE)

legend <- system.file("extdata/input", "1000GP_Phase3_chr22.legend",
package = "bbmix", mustWork = TRUE)

fisher_f <- system.file("extdata/input", "chr22.FS.Q20.alleleCounts.txt",
package = "bbmix", mustWork = TRUE)

cluster_f <- system.file("extdata/input", "fSNPs_22_RP_maf0_01_cluster3window35.txt",
package = "bbmix", mustWork = TRUE)

out <- paste0(tempdir() , "/NA12878.chrom22.gt.txt")

## Run call_gt:
call_gt(allele_counts_f = counts_f,
legend_f = legend,
fisher_f = fisher_f,
cluster_f = cluster_f,
out = out)

unlink(out)
```

`call_help` *call gt helper, calculate mean dbetabinom from all posterior samples*

Description

call gt helper, calculate mean dbetabinom from all posterior samples

Usage

```
call_help(n, m, mu, lambda)
```

Arguments

`n`	counts alt allele
`m`	total counts
`mu`	vector with posterior draws for mu param
`lambda`	vector with posterior draws for lambda param

Value

mean of dbetabinom

`ex_alt_hom` *Exclude fSNPs with no alternative allele in any sample. Also exclude fSNPs if all samples are hom.*

Description

Exclude fSNPs with no alternative allele in any sample. Also exclude fSNPs if all samples are hom.

Usage

```
ex_alt_hom(gt_f, out)
```

Arguments

`gt_f`	character vector with file names with genotype calls per sample
`out`	file name to save output

Value

save file

Examples

```
gt_f <- system.file("extdata/output", "gt.NA12878.chr22.txt",
package = "bbmix",
mustWork = TRUE)
out <- tempfile()

## Running function
ex_alt_hom(gt_f, out)

unlink(out)
```

`fit_bb` *Fit beta binomial distribution to allelic counts for homozygous reference, heterozygous, homozygous alternative*

Description

Fit beta binomial distribution to allelic counts for homozygous reference, heterozygous, homozygous alternative

Usage

```
fit_bb(
  counts_f,
  depth = 10,
  N = 1000,
  prefix = NULL,
  k = 3,
  alpha_p = c(1, 10, 499),
  beta_p = c(499, 10, 1),
  out,
  mc.cores = NULL
)
```

Arguments

`counts_f` file name with allele counts for SNPs

`depth` depth cut-off to use to select SNPs to fit distributions

`N` number of SNPs to use for fitting

`prefix` charcter with prefix to add for saving files, defaults to NULL

`k` number of components for mixture model, defaults to 3

`alpha_p` alpha parameter for the k components of alpha parameter

`beta_p` beta paramenter for the k components of Beta parameter

`out` character with dir name to save output

`mc.cores` number of cores to use, defaults to parallel detected cores

Value

saves stan object to file

Examples

```
## Not run:
## Retrive input files for running call_gt
counts_f <- system.file("extdata/input", "NA12878.chr22.Q20.allelicCounts.txt",
package = "bbmix",
mustWork = TRUE)

out <- tempdir()
fit_bb(counts_f = counts_f, N=10,
out = out, mc.cores=1)
unlink(out)

## End(Not run)
```

gt_help *call gt helper, get posterior mean, expected gt and sd expected gt across all samples*

Description

call gt helper, get posterior mean, expected gt and sd expected gt across all samples

Usage

```
gt_help(stan_samples, pop, data)
```

Arguments

`stan_samples` matrix with samples extracted from stan fit object, params mu and lambda

`pop` population to select AF for GT prior, defaults to EUR

`data` data table 1 row with counts and EAF to apply model

Value

gt_help()

`poolreads` *Pool randomly selected reads from different files*

Description

Pool randomly selected reads from different files

Usage

```
poolreads(files, N = 1000, d = 10, out)
```

Arguments

`files`	names for files to extract reads
`N`	number of reads to extract
`d`	depth for reads
`out`	file name to save reads

Value

save files

Examples

```
counts_f <- system.file("extdata/input", "NA12878.chr22.Q20.allelicCounts.txt",
package = "bbmix",
mustWork = TRUE)

## In this example we only use one file and we take a pool of 10 reads

out <- tempfile()

poolreads(files=counts_f,
N=10,
d=10,
out = out)

unlink(out)
```

Index