

Package ‘DEploid’

May 7, 2026

Type Package

Title Deconvolute Mixed Genomes with Unknown Proportions

Version 0.5.7

Description Traditional phasing programs are limited to diploid organisms. Our method modifies Li and Stephens algorithm with Markov chain Monte Carlo (MCMC) approaches, and builds a generic framework that allows haplotype searches in a multiple infection setting. This package is primarily developed as part of the Pf3k project, which is a global collaboration using the latest sequencing technologies to provide a high-resolution view of natural variation in the malaria parasite *Plasmodium falciparum*. Parasite DNA are extracted from patient blood sample, which often contains more than one parasite strain, with unknown proportions. This package is used for deconvoluting mixed haplotypes, and reporting the mixture proportions from each sample.

URL <https://github.com/DEploid-dev/DEploid-r>

BugReports <https://github.com/DEploid-dev/DEploid-r/issues>

License GPL (>= 3)

Depends R (>= 3.1.0), DEploid.utils (>= 0.0.1)

Imports Rcpp (>= 0.11.2), scales (>= 0.4.0), plotly (>= 4.7.1), magrittr (>= 1.5), rmarkdown(>= 1.6), htmlwidgets (>= 1.0)

Suggests knitr, testthat (>= 0.9.0)

VignetteBuilder knitr

LinkingTo Rcpp

RoxygenNote 7.3.2

Encoding UTF-8

Date 2024-12-22

NeedsCompilation yes

Author Joe Zhu [aut, cre] (ORCID: <<https://orcid.org/0000-0001-7566-2787>>),
Jacob Almagro-Garcia [aut],
Gil McVean [aut],
University of Oxford [cph],

Yinghan Liu [ctb],
 CodeCogs Zyba Ltd [com, cph],
 Deepak Bandyopadhyay [com, cph],
 Lutz Kettner [com, cph]

Maintainer Joe Zhu <sha.joe.zhu@gmail.com>

Repository CRAN

Date/Publication 2025-01-20 03:20:02 UTC

Contents

dEplod	2
plotAltVsRefPlotly	3
plotHistWSAFPlotly	4
plotObsExpWSAFPlotly	5
plotWSAFVsPLAFPlotly	6
Index	8

dEplod	<i>Deconvolute mixed haplotypes</i>
--------	-------------------------------------

Description

Deconvolute mixed haplotypes, and reporting the mixture proportions from each sample This function provides an interface for calling *dEplod* from R. The command line options are passed via the `args` argument

Usage

```
dEplod(args)
```

Arguments

`args` String of dEplod input.

Value

A list with members of haplotypes, proportions and log likelihood of the MCMC chain.

- Haps Haplotypes at the final iteration in plain text file.
- Proportions MCMC updates of the proportion estimates.
- llks Log likelihood of the MCMC chain.

Seeding

The R version of DEplod uses random number from R's random generator. Therefore, the `'-seed'` argument of the command line version will be ignored, and no seed is given in the output. Use the R function `'set.seed'` prior to calling this function to ensure reproducibility of results.

See Also

- vignette('dEplod-Arguments') for an overview of commandline arguments

Examples

```
## Not run:
vcfFile = system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEplod")
plafFile = system.file("extdata", "labStrains.test.PLAF.txt", package = "DEplod")
set.seed(1234)
PG0390.deconv = dEplod(paste("-vcf", vcfFile, "-plaf", plafFile, "-noPanel"))

## End(Not run)
```

plotAltVsRefPlotly *Plot coverage*

Description

Plot alternative allele count vs reference allele count at each site.

Usage

```
plotAltVsRefPlotly(ref, alt, title = "Alt vs Ref", potentialOutliers = c())
```

Arguments

ref	Numeric array of reference allele count.
alt	Numeric array of alternative allele count.
title	Figure title, "Alt vs Ref" by default
potentialOutliers	Index of potential outliers.

Examples

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEplod")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEplod")
PG0390CoverageT <- extractCoverageFromTxt(refFile, altFile)
plotAltVsRefPlotly(PG0390CoverageT$refCount, PG0390CoverageT$altCount)

# Example 2
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEplod")
PG0390CoverageV <- extractCoverageFromVcf(vcfFile, "PG0390-C")
plotAltVsRefPlotly(PG0390CoverageV$refCount, PG0390CoverageV$altCount)
```

plotHistWSAFPlotly *WSAF histogram*

Description

Produce histogram of the allele frequency within sample.

Usage

```
plotHistWSAFPlotly(obsWSAF, exclusive = TRUE, title = "Histogram 0<WSAF<1")
```

Arguments

obsWSAF	Observed allele frequency within sample
exclusive	When TRUE $0 < \text{WSAF} < 1$; otherwise $0 \leq \text{WSAF} \leq 1$.
title	Figure title, "Histogram 0<WSAF<1" by default

Value

histogram

Examples

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid")
PG0390Coverage <- extractCoverageFromTxt(refFile, altFile)
obsWSAF <- computeObsWSAF(PG0390Coverage$altCount, PG0390Coverage$refCount)
plotHistWSAFPlotly(obsWSAF)
myhist <- plotHistWSAFPlotly(obsWSAF)

# Example 2
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid")
PG0390CoverageV <- extractCoverageFromVcf(vcfFile, "PG0390-C")
obsWSAF <- computeObsWSAF(PG0390CoverageV$altCount, PG0390CoverageV$refCount)
plotHistWSAFPlotly(obsWSAF)
myhist <- plotHistWSAFPlotly(obsWSAF)
```

plotObsExpWSAFPlotly *Plot WSAF*

Description

Plot observed alternative allele frequency within sample against expected WSAF.

Usage

```
plotObsExpWSAFPlotly(obsWSAF, expWSAF, title = "WSAF(observed vs expected)")
```

Arguments

obsWSAF	Numeric array of observed WSAF.
expWSAF	Numeric array of expected WSAF.
title	Figure title, "WSAF(observed vs expected)" by default

Examples

```
## Not run:
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid")
PG0390CoverageV <- extractCoverageFromVcf(vcfFile, "PG0390-C")
obsWSAF <- computeObsWSAF(PG0390CoverageV$altCount, PG0390CoverageV$refCount)
plafFile <- system.file("extdata", "labStrains.test.PLAF.txt",
  package = "DEploid"
)
PG0390CoverageV.deconv <- dEplid(paste(
  "-vcf", vcfFile,
  "-plaf", plafFile, "-noPanel"
))

prop <- PG0390CoverageV.deconv$Proportions[dim(PG0390CoverageV.deconv
$Proportions)[1], ]

expWSAF <- t(PG0390CoverageV.deconv$Haps) %*% prop
plotObsExpWSAFPlotly(obsWSAF, expWSAF)

## End(Not run)
```

plotWSAFVsPLAFPlotly *Plot WSAF vs PLAF*

Description

Plot allele frequencies within sample against population level.

Usage

```
plotWSAFVsPLAFPlotly(
  plaf,
  obsWSAF,
  ref,
  alt,
  title = "WSAF vs PLAF",
  potentialOutliers = c()
)
```

Arguments

plaf	Numeric array of population level allele frequency.
obsWSAF	Numeric array of observed alternative allele frequencies within sample.
ref	Numeric array of reference allele count.
alt	Numeric array of alternative allele count.
title	Figure title, "WSAF vs PLAF" by default
potentialOutliers	Index of potential outliers.

Examples

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid")
PG0390CoverageT <- extractCoverageFromTxt(refFile, altFile)
obsWSAF <- computeObsWSAF(PG0390CoverageT$altCount, PG0390CoverageT$refCount)
plafFile <- system.file("extdata", "labStrains.test.PLAF.txt",
  package = "DEploid"
)
plaf <- extractPLAF(plafFile)
plotWSAFVsPLAFPlotly(
  plaf, obsWSAF, PG0390CoverageT$refCount,
  PG0390CoverageT$altCount
)

# Example 2
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid")
PG0390CoverageV <- extractCoverageFromVcf(vcfFile, "PG0390-C")
```

```
obsWSAF <- computeObsWSAF(PG0390CoverageV$saltCount, PG0390CoverageV$refCount)
plafFile <- system.file("extdata", "labStrains.test.PLAF.txt",
  package = "DEploid"
)
plaf <- extractPLAF(plafFile)
plotWSAFVsPLAFPlotly(
  plaf, obsWSAF, PG0390CoverageV$refCount,
  PG0390CoverageV$saltCount
)
```

Index

dEplod, [2](#)

plotAltVsRefPlotly, [3](#)

plotHistWSAFPlotly, [4](#)

plotObsExpWSAFPlotly, [5](#)

plotWSAFVsPLAFPlotly, [6](#)