

# Package ‘DEploid.utils’

May 7, 2026

**Title** 'DEploid' Data Analysis and Results Interpretation

**Version** 0.0.1

**Description** 'DEploid' (Zhu et.al. 2018 <[doi:10.1093/bioinformatics/btx530](https://doi.org/10.1093/bioinformatics/btx530)>) is designed for deconvoluting mixed genomes with unknown proportions. Traditional phasing programs are limited to diploid organisms. Our method modifies Li and Stephen's algorithm with Markov chain Monte Carlo (MCMC) approaches, and builds a generic framework that allows haplotype searches in a multiple infection setting. This package provides R functions to support data analysis and results interpretation.

**Depends** R (>= 3.1.0)

**Imports** Rcpp (>= 0.11.2), scales (>= 0.4.0), magrittr (>= 1.5),  
combinat

**Suggests** knitr, rmarkdown(>= 1.6), circlize, testthat (>= 0.9.0)

**LinkingTo** Rcpp

**RoxygenNote** 7.3.2

**Encoding** UTF-8

**Date** 2024-11-18

**License** Apache License (>= 2)

**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](mailto:sha.joe.zhu@gmail.com)>

**Repository** CRAN

**Date/Publication** 2024-12-19 16:00:18 UTC

## Contents

computeObsWSAF . . . . .	2
extractCoverageFromTxt . . . . .	3
extractCoverageFromVcf . . . . .	4
extractPLAF . . . . .	5
haplotypePainter . . . . .	5
histWSAF . . . . .	6
plotAltVsRef . . . . .	7
plotObsExpWSAF . . . . .	8
plotProportions . . . . .	9
plotWSAFvsPLAF . . . . .	9
<b>Index</b>	<b>11</b>

---

computeObsWSAF	<i>Compute observed WSAF</i>
----------------	------------------------------

---

### Description

Compute observed allele frequency within sample from the allele counts.

### Usage

```
computeObsWSAF(alt, ref)
```

### Arguments

alt	Numeric array of alternative allele count.
ref	Numeric array of reference allele count.

### Value

Numeric array of observed allele frequency within sample.

### See Also

[histWSAF](#) for histogram.

### Examples

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid.utils")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid.utils")
PG0390CoverageTxt <- extractCoverageFromTxt(refFile, altFile)
obsWSAF <- computeObsWSAF(PG0390CoverageTxt$altCount, PG0390CoverageTxt$refCount)

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

```
PG0390CoverageVcf <- extractCoverageFromVcf(vcfFile, "PG0390-C")
obsWSAF <- computeObsWSAF(PG0390CoverageVcf$altCount, PG0390CoverageVcf$refCount)
```

---

`extractCoverageFromTxt`

*Extract read counts from plain text file*

---

### **Description**

Extract read counts from tab-delimited text files of a single sample.

### **Usage**

```
extractCoverageFromTxt(refFileName, altFileName)
```

### **Arguments**

`refFileName` Path of the reference allele count file.  
`altFileName` Path of the alternative allele count file.

### **Value**

A data.frame contains four columns: chromosomes, positions, reference allele count, alternative allele count.

### **Note**

The allele count files must be tab-delimited. The allele count files contain three columns: chromosomes, positions and allele count.

### **Examples**

```
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid.utils")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid.utils")
PG0390 <- extractCoverageFromTxt(refFile, altFile)
```

---

`extractCoverageFromVcf`*Extract VCF information*

---

**Description**

Extract VCF information

**Usage**

```
extractCoverageFromVcf(filename, samplename)
```

**Arguments**

<code>filename</code>	VCF file name.
<code>samplename</code>	Sample name

**Value**

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

- CHROM SNP chromosomes.
- POS SNP positions.
- refCount reference allele count.
- altCount alternative allele count.

**See Also**

- `extractCoverageFromVcf`
- `extractCoverageFromTxt`

**Examples**

```
vcfFile = system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid.utils")  
vcf = extractCoverageFromVcf(vcfFile, "PG0390-C")
```

---

extractPLAF	<i>Extract PLAF</i>
-------------	---------------------

---

**Description**

Extract population level allele frequency (PLAF) from text file.

**Usage**

```
extractPLAF(plafFileName)
```

**Arguments**

plafFileName Path of the PLAF text file.

**Value**

A numeric array of PLAF

**Note**

The text file must have header, and population level allele frequency recorded in the "PLAF" field.

**Examples**

```
plafFile <- system.file("extdata", "labStrains.test.PLAF.txt", package = "DEploid.utils")
plaf <- extractPLAF(plafFile)
```

---

haplotypePainter	<i>Painting haplotype according the reference panel</i>
------------------	---

---

**Description**

Plot the posterior probabilities of a haplotype given the refernece panel.

**Usage**

```
haplotypePainter(  
  posteriorProbabilities,  
  title = "",  
  labelScaling,  
  numberOfInbreeding = 0  
)
```

**Arguments**

posteriorProbabilities      Posterior probabilities matrix with the size of number of loci by the number of reference strain.

title                          Figure title.

labelScaling                  Scaling parameter for plotting.

numberOfInbreeding          Number of inbreeding strains

**Value**

No return value called for side effects

---

histWSAF	<i>WSAF histogram</i>
----------	-----------------------

---

**Description**

Produce histogram of the allele frequency within sample.

**Usage**

```
histWSAF(
  obsWSAF,
  exclusive = TRUE,
  title = "Histogram 0<WSAF<1",
  cex.lab = 1,
  cex.main = 1,
  cex.axis = 1
)
```

**Arguments**

obsWSAF                      Observed allele frequency within sample

exclusive                    When TRUE  $0 < \text{WSAF} < 1$ ; otherwise  $0 \leq \text{WSAF} \leq 1$ .

title                          Histogram title

cex.lab                        Label size.

cex.main                      Title size.

cex.axis                      Axis text size.

**Value**

histogram

**Examples**

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid.utils")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid.utils")
PG0390CoverageTxt <- extractCoverageFromTxt(refFile, altFile)
obsWSAF <- computeObsWSAF(PG0390CoverageTxt$altCount, PG0390CoverageTxt$refCount)
histWSAF(obsWSAF)
myhist <- histWSAF(obsWSAF, FALSE)

# Example 2
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid.utils")
PG0390CoverageVcf <- extractCoverageFromVcf(vcfFile, "PG0390-C")
obsWSAF <- computeObsWSAF(PG0390CoverageVcf$altCount, PG0390CoverageVcf$refCount)
histWSAF(obsWSAF)
myhist <- histWSAF(obsWSAF, FALSE)
```

---

plotAltVsRef	<i>Plot coverage</i>
--------------	----------------------

---

**Description**

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

**Usage**

```
plotAltVsRef(
  ref,
  alt,
  title = "Alt vs Ref",
  exclude.ref = c(),
  exclude.alt = c(),
  potentialOutliers = c(),
  cex.lab = 1,
  cex.main = 1,
  cex.axis = 1
)
```

**Arguments**

ref	Numeric array of reference allele count.
alt	Numeric array of alternative allele count.
title	Figure title, "Alt vs Ref" by default
exclude.ref	Numeric array of reference allele count at sites that are not deconvoluted.
exclude.alt	Numeric array of alternative allele count at sites that are not deconvoluted
potentialOutliers	Potential outliers

cex.lab	Label size.
cex.main	Title size.
cex.axis	Axis text size.

**Value**

No return value called for side effects

**Examples**

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid.utils")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid.utils")
PG0390CoverageTxt <- extractCoverageFromTxt(refFile, altFile)
plotAltVsRef(PG0390CoverageTxt$refCount, PG0390CoverageTxt$altCount)

# Example 2
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid.utils")
PG0390CoverageVcf <- extractCoverageFromVcf(vcfFile, "PG0390-C")
plotAltVsRef(PG0390CoverageVcf$refCount, PG0390CoverageVcf$altCount)
```

---

plotObsExpWSAF

*Plot WSAF*

---

**Description**

Plot observed alternative allele frequency within sample against expected WSAF.

**Usage**

```
plotObsExpWSAF(
  obsWSAF,
  expWSAF,
  title = "WSAF(observed vs expected)",
  cex.lab = 1,
  cex.main = 1,
  cex.axis = 1
)
```

**Arguments**

obsWSAF	Numeric array of observed WSAF.
expWSAF	Numeric array of expected WSAF.
title	Figure title.
cex.lab	Label size.
cex.main	Title size.
cex.axis	Axis text size.

**Value**

No return value called for side effects

---

plotProportions	<i>Plot proportions</i>
-----------------	-------------------------

---

**Description**

Plot the MCMC samples of the proportion, indexed by the MCMC chain.

**Usage**

```
plotProportions(  
  proportions,  
  title = "Components",  
  cex.lab = 1,  
  cex.main = 1,  
  cex.axis = 1  
)
```

**Arguments**

proportions	Matrix of the MCMC proportion samples. The matrix size is number of the MCMC samples by the number of strains.
title	Figure title.
cex.lab	Label size.
cex.main	Title size.
cex.axis	Axis text size.

**Value**

No return value called for side effects

---

plotWSAFvsPLAF	<i>Plot WSAF vs PLAF</i>
----------------	--------------------------

---

**Description**

Plot allele frequencies within sample against population level.

**Usage**

```
plotWSAFvsPLAF(
  plaf,
  obsWSAF,
  expWSAF = c(),
  potentialOutliers = c(),
  title = "WSAF vs PLAF",
  cex.lab = 1,
  cex.main = 1,
  cex.axis = 1
)
```

**Arguments**

plaf	Numeric array of population level allele frequency.
obsWSAF	Numeric array of observed alternative allele frequencies within sample.
expWSAF	Numeric array of expected WSAF from model.
potentialOutliers	Potential outliers
title	Figure title, "WSAF vs PLAF" by default
cex.lab	Label size.
cex.main	Title size.
cex.axis	Axis text size.

**Value**

No return value called for side effects

**Examples**

```
# Example 1
refFile <- system.file("extdata", "PG0390-C.test.ref", package = "DEploid.utils")
altFile <- system.file("extdata", "PG0390-C.test.alt", package = "DEploid.utils")
PG0390CoverageTxt <- extractCoverageFromTxt(refFile, altFile)
obsWSAF <- computeObsWSAF(PG0390CoverageTxt$altCount, PG0390CoverageTxt$refCount)
plafFile <- system.file("extdata", "labStrains.test.PLAF.txt", package = "DEploid.utils")
plaf <- extractPLAF(plafFile)
plotWSAFvsPLAF(plaf, obsWSAF)

# Example 2
vcfFile <- system.file("extdata", "PG0390-C.test.vcf.gz", package = "DEploid.utils")
PG0390CoverageVcf <- extractCoverageFromVcf(vcfFile, "PG0390-C")
obsWSAF <- computeObsWSAF(PG0390CoverageVcf$altCount, PG0390CoverageVcf$refCount)
plafFile <- system.file("extdata", "labStrains.test.PLAF.txt", package = "DEploid.utils")
plaf <- extractPLAF(plafFile)
plotWSAFvsPLAF(plaf, obsWSAF)
```

# Index

`computeObsWSAF`, 2  
`extractCoverageFromTxt`, 3  
`extractCoverageFromVcf`, 4  
`extractPLAF`, 5  
`haplotypePainter`, 5  
`histWSAF`, 2, 6  
`plotAltVsRef`, 7  
`plotObsExpWSAF`, 8  
`plotProportions`, 9  
`plotWSAFvsPLAF`, 9