Package ‘DirichletMultinomial’

May 3, 2024

Type Package

Title Dirichlet-Multinomial Mixture Model Machine Learning for Microbiome Data

Version 1.46.0

Author Martin Morgan <martin.morgan@roswellpark.org>

Maintainer Martin Morgan <martin.morgan@roswellpark.org>

Description Dirichlet-multinomial mixture models can be used to describe variability in microbial metagenomic data. This package is an interface to code originally made available by Holmes, Harris, and Quince, 2012, PLoS ONE 7(2): 1-15, as discussed further in the man page for this package, ?DirichletMultinomial.

License LGPL-3

Depends S4Vectors, IRanges

Imports stats4, methods, BiocGenerics

Suggests lattice, parallel, MASS, RColorBrewer, xtable


SystemRequirements gsl

biocViews ImmunoOncology, Microbiome, Sequencing, Clustering, Classification, Metagenomics

git_url https://git.bioconductor.org/packages/DirichletMultinomial

git_branch RELEASE_3_19

git_last_commit 40e13c6

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-05-03
DirichletMultinomial-package

Contents

DirichletMultinomial-package ............................................. 2
cvdmngroup ................................................................. 3
data ................................................................. 4
dmn ................................................................. 5
DMN-class ............................................................. 6
dmngroup ............................................................. 7
DMNGroup-class ............................................................. 8
heatmapdmn ............................................................... 9
model components .................................................. 10
roc ................................................................. 11
Utilities ................................................................. 13

Index ................................................................. 15

DirichletMultinomial-package

Dirichlet-Multinomial Mixture Model Machine Learning for Microbiome Data

Description

Dirichlet-multinomial mixture models can be used to describe variability in microbial metagenomic data. This package is an interface to code originally made available by Holmes, Harris, and Quince, 2012, PLoS ONE 7(2): 1-15.

Details

The estimation routine is from the LGPL-licensed (as stated on the corresponding googlecode page) source http://microbedmm.googlecode.com/files/MicrobeDMMv1.0.tar.gz, retrieved 17 February 2012.


Author(s)

Maintainer: Martin Morgan mailto:mtmorgan@fhcrc.org
Cross-validation on Dirichlet-Multinomial classifiers.

Description

Run cross-validation on Dirichlet-Multinomial generative classifiers.

Usage

cvdmngroup(ncv, count, k, z, ..., verbose = FALSE, .lapply = parallel::mclapply)

Arguments

ncv integer(1) number of cross-validation groups, between 2 and nrow(count).

count matrix of sample x taxon counts, subsets of which are used for training and cross-validation.

k named integer() vector of groups and number of Dirichlet components; e.g., c(Lean=1, Obese=3) performs cross-validation for models with k=1 Dirichlet components for the ‘Lean’ group, k=3 Dirichlet components for ‘Obese’.

z True group assignment.

... Additional arguments, passed to dmn during each cross-validation.

verbose logical(1) indicating whether progress should be reported

.lapply A function used to perform the outer cross-validation loop, e.g., lapply for calculation on a single processor, parallel::mclapply for parallel evaluation.

Value

A data.frame summarizing classifications of test samples in cross-validation groups. Columns are:

group The cross-validation group in which the individual was used for testing.

additional columns Named after classification groups, giving the posterior probability of assignment.

Author(s)

Martin Morgan mailto:mtmorgan@fhcrc.org

See Also

dmn, DirichletMultinomial-package, vignette("DirichletMultinomial")
Examples

data(xval)  ## result of following commands
head(xval)

## Not run:
## count matrix
fl <- system.file(package="DirichletMultinomial", "extdata",
"Twins.csv")
count <- t(as.matrix(read.csv(fl, row.names=1)))

## phenotype
fl <- system.file(package="DirichletMultinomial", "extdata",
"TwinStudy.t")
pheno0 <- scan(fl)
lvls <- c("Lean", "Obese", "Overwt")
pheno <- factor(lvls[pheno0 + 1], levels=lvls)
names(pheno) <- rownames(count)

## subset
keep <- c("Lean", "Obese")
count <- count[pheno]
pheno <- factor(pheno[pheno]

## cross-validation, single Dirichlet component for Lean, 3 for Obese
xval <- cvdmngroup(nrow(count), count, c(Lean=1, Obese=3), pheno,
verbos etrue, mc.preschedule=false)

## End(Not run)

---

Description

These data objects correspond to steps in a typical work flow, as described in the vignette to this package. fit corresponds to dmn fits to different values of k. bestgroup is the result of the two-
group generative classifier. xval summarizes leave-one-out cross validation of the classifier.

Usage

data(fit)
data(bestgrp)
data(xval)

Format

fit is a list of seven DMN objects.
bestgrp is a DMNGroup object.
xval is a data.frame with columns corresponding to the cross-validation group membership and
the Lean and Obese posterior probabilities.
dmn

Examples

```r
data(fit); fit[1:2]
plot(sapply(fit, laplace), type="b")
data(bestgrp); bestgrp
data(xval); head(xval, 3)
```

dmn

Fit Dirichlet-Multinomial models to count data.

Description

Fit Dirichlet-Multinomial models to a sample x taxon count matrix.

Usage

```r
dmn(count, k, verbose = FALSE, seed = runif(1, 0, .Machine$integer.max))
```

Arguments

- `count` matrix() of sample x taxon counts.
- `k` integer(1), the number of Dirichlet components to fit.
- `verbose` logical(1) indicating whether progress in fit should be reported.
- `seed` numeric(1) random number seed.

Details

This implements Dirichlet-multinomial mixture models describe in the package help page, `DirichletMultinomial-package`.

Value

An object of class dmn, with elements (elements are usually retrieved via functions defined in the package, not directly).

- `GoodnessOfFit` NLE, LogDet, Laplace, AIC, and BIC criteria assessing goodness-of-fit.
- `Group` matrix of dimension samples x k, providing the Dirichlet parameter vectors.
- `Mixture` `Weight` numeric() of length k, with relative weight of each component.
- `Fit` `Lower` matrix() of dimension taxa x k with 95% lower bounds on Dirichlet component vector estimates.
- `Estimate` `Upper` matrix() of dimension taxa x k with 95% upper bounds on Dirichlet component vector estimates.

Author(s)

Martin Morgan `mailto:mtmorgan@fhcrc.org`
References

See Also
DirichletMultinomial-package, vignette("DirichletMultinomial")

Examples
```
data(fit)
## k = 1:7; full example in vignette
lplc <- sapply(fit, laplace)
plot(lplc, type="b")
fit[[which.min(lplc)]]
```

DMN-class

Class "DMN"

Description
Result from fitting a Dirichlet-Multinomial model.

Objects from the Class
Objects can be created by calls to `dmn`.

Slots
The contents of a slot is usually retrieved via the methods described on the `mixture` help page.

`goodnessOfFit` NLE, LogDet, Laplace, AIC, and BIC criteria assessing goodness-of-fit.
`group` matrix of dimension samples x k, providing the Dirichlet parameter vectors.
`mixture` `Weight` numeric() of length k, with relative weight of each component.
`fit` `Lower` matrix() of dimension taxa x k with 95% lower bounds on Dirichlet component vector estimates.
`Estimate` matrix() of dimension taxa x k with Dirichlet component vector estimates.
`Upper` matrix() of dimension taxa x k with 95% upper bounds on Dirichlet component vector estimates.

Methods
See the `mixture` help page.

Author(s)
Martin Morgan `mailto:mtmorgan@fhcrc.org`
See Also
dmn, mixture.

Examples

data(fit)
fit[[4]]

dmngroup(count, group, k, ..., simplify = TRUE,
.lapply = parallel::mclapply)

Arguments
count matrix() of sample x taxon counts.
group factor() or vector to be coerced to a factor, with as many elements as there are rows in count, indicating the group to which the corresponding sample belongs.
k integer(), the number(s) of Dirichlet components to fit.
... Additional arguments, passed to dmn.
simplify Return only the best-fit model for each group?
.lapply An lapply-like function for application of group x k fits.

Details
This function divided count into groups defined by group, creates all combinations of group x k, and evaluates each using dmn. When simplify=TRUE, the best (Laplace) fit is selected for each group.

Value
An object of class dmngroup, a list of fitted models of class dmn. When simplify=TRUE, elements are named by the group to which they correspond.

Author(s)
Martin Morgan mailto:mtmorgan@fhcrc.org
References


See Also
dmn, DirichletMultinomial-package, vignette("DirichletMultinomial")

Examples

```r
## best fit for groups 'Lean' and 'Obese'; full example in vignette.
## Not run: bestgrp <- dmngroup(count, pheno, k=1:5, verbose=TRUE,
## mc.preschedule=FALSE)

## End(Not run)
data(bestgrp)
bestgrp
bestgrp["Obese"]
```

---

**DMNGroup-class**

**Class** "DMNGroup"

**Description**

Result from fitting a Dirichlet-Multinomial generative classifier.

**Objects from the Class**

Objects can be created by calls to `dmngroup`.

**Slots**

All slots in this class are inherited from `SimpleList`; see 'Methods', below, for information on how to manipulate this object.

**Extends**


**Methods**

See the `mixture` help page for functions that operate on `DMNGroup` and `DMN`.
`DMNGroup` can be manipulated as a list; see `SimpleList` for a description of typical list-like functions.

**Author(s)**

Martin Morgan mailto:mtmorgan@fhcrc.org
See Also

mixture, DMN, SimpleList.

Examples

data(bestgrp)
bestgrp
bestgrp[[1]]

heatmapdmn

Heatmap representation of samples assigned to Dirichlet components.

Description

Produce a heat map summarizing count data, grouped by Dirichlet component.

Usage

heatmapdmn(count, fit1, fitN, ntaxa = 30, ..., transform = sqrt, lblwidth = 0.2 * nrow(count), col = .gradient)

Arguments

count  A matrix of sample x taxon counts, as supplied to dmn.
fit1  An instance of class dmn, from a model fit to a single Dirichlet component, k=1 in dmn.
fitN  An instance of class dmn, from a model fit to N != 1 components, k=N in dmn.
ntaxa  The ntaxa most numerous taxa to display counts for.
...  Additional arguments, ignored.
transform  Transformation to apply to count data prior to visualization; this does not influence mixture membership or taxonomic ordering.
lblwidth  The proportion of the plot to dedicate to taxonomic labels, as a fraction of the number of samples to be plotted.
col  The colors used to display (possibly transformed, by transform) count data, as used by image.

Details

Columns of the heat map correspond to samples. Samples are grouped by Dirichlet component, with average (Dirichlet) components summarized as a separate wide column. Rows correspond to taxonomic groups, ordered based on contribution to Dirichlet components.

Author(s)

Martin Morgan mailto:mtmorgan@fhcrc.org
Examples

```r
## counts
fl <- system.file(package="DirichletMultinomial", "extdata", "Twins.csv")
count <- t(as.matrix(read.csv(fl, row.names=1)))

## all and best-fit clustering
data(fit)
lplc <- sapply(fit, laplace)
best <- fit[[which.min(lplc)]]
heatmapdmn(count, fit[[1]], best, 30)
```

### Description

The accessors `mixture` and `mixturewt` return information about the estimated Dirichlet components of the fitted model. Return values are described in the Values section, below.

### Usage

```r
mixture(object, ..., assign=FALSE)
mixturewt(object, ...)
goodnessOfFit(object, ...)
laplace(object, ...)
## S4 method for signature 'DMN'
AIC(object, ..., k = 2)
## S4 method for signature 'DMN'
BIC(object, ...)

## S4 method for signature 'DMN'
fitted(object, ..., scale=FALSE)
## S4 method for signature 'DMN'
predict(object, newdata, ..., logevidence=FALSE)
## S4 method for signature 'DMNGroup'
fitted(object, ...)
## S4 method for signature 'DMNGroup'
predict(object, newdata, ..., assign=FALSE)
## S4 method for signature 'DMNGroup'
summary(object, ...)
```

### Arguments

- `object` An instance of class `dmn`.
- `newdata` A matrix of new sample x taxon data to be fitted to the model of `object`.
... Additional arguments, available to methods, when applicable.

assign logical(1) indicating whether the maximum per-sample mixture component should be returned (assign=FALSE), or the full mixture matrix (assign=TRUE).

scale logical(1) indicating whether fitted values should be returned unscaled (default, scaled=FALSE) or scaled by the variability of mixturewt parameter theta.

logevidence logical(1) indicating whether posterior probability (default, logevidence=FALSE) or log evidence logical=TRUE should be returned.

k ignored.

Value

mixture with assign=FALSE returns a matrix of sample x Dirichlet component estimates. With assign=TRUE mixture returns a named vector indexing the maximal Dirichlet component of each sample.
mixturewt returns a matrix with rows corresponding to mixture components, and columns pi (component weight) and theta (component variability). Small values of theta correspond to highly variable components.
goodnessOfFit returns a named numeric vector of measures of goodness of fit.
laplace, AIC, and BIC return the corresponding measures of goodness of fit.

Author(s)

Martin Morgan mailto:mtmorgan@fhcrc.org

Examples

data(fit)
best <- fit[[4]]
mixturewt(best)
head(mixture(best), 3)
head(mixture(best, assign=TRUE), 3)
goodnessOfFit(best)

fl <- system.file(package="DirichletMultinomial", "extdata", "Twins.csv")
count <- t(as.matrix(read.csv(fl, row.names=1)))
data(bestgrp)
bestgrp
head(predict(bestgrp, count))

roc Summarize receiver-operator characteristics

Description

Returns a data.frame summarizing the cumulative true- and false-positive probabilities from expected and observed classifications.
Usage

roc(exp, obs, ...)

Arguments

exp logical() vector of expected classifications to a particular group.
obs Predicted probability of assignment to the group identified by TRUE values in exp. The length of exp and obs must be identical.
... Additional arguments, available to methods.

Value

A data.frame with columns

TruePositive Cummulative probability of correct assignment.
FalsePositive Cummulative probability of incorrect assignment.

Author(s)

Martin Morgan mailto:mtmorgan@fhcrc.org

Examples

library(lattice)

## count matrix
fl <- system.file(package="DirichletMultinomial", "extdata", "Twins.csv")
count <- t(as.matrix(read.csv(fl, row.names=1)))

## phenotype
fl <- system.file(package="DirichletMultinomial", "extdata", "TwinStudy.t")
pheno0 <- scan(fl)
lvls <- c("Lean", "Obese", "Overwt")
pheno <- factor(lvls[pheno0 + 1], levels=lvls)
names(pheno) <- rownames(count)

## count data used for cross-validation, and cross-validation
count <- csubset(c("Lean", "Obese"), count, pheno)
data(bestgrp)

## true, false positives from single-group classifier
bst <- roc(pheno[rownames(count)] == "Obese",
predict(bestgrp, count)[,"Obese"])
head(bst)

## lattice plot
xyplot(TruePostive ~ FalsePositive, bst, type="l",
    xlab="False Positive", ylab="True Positive")
**Utilities**

*Helpful utility functions*

---

**Description**

csubset creates a subset of a count matrix, based on identity of column phenotypes to a specified value.

**Usage**

csubset(val, x, pheno, cidx = TRUE)

**Arguments**

- **val** character(1) specifying the subset of phenotype to select.
- **x** A matrix of counts, with rows corresponding to samples and columns to taxonomic groups.
- **pheno** A character() vector of length equal to the number of rows in count, indicating the phenotype of the corresponding sample.
- **cidx** A logical(1) indicating whether columns (taxa) with zero counts in the count matrix following removal of taxa not satisfying pheno %in% val should be removed. cidx=FALSE removes the 0-count columns.

**Value**

A matrix of counts, with rows satisfying pheno %in% val and with columns equal either to ncol(x) (when cidx=TRUE) or the number of columns with non-zero counts after row subsetting (cidx=FALSE).

**Author(s)**

Martin Morgan mailto:mtmorgan@fhcrc.org

**Examples**

```r
## count matrix
fl <- system.file(package="DirichletMultinomial", "extdata", "Twins.csv")
count <- t(as.matrix(read.csv(fl, row.names=1)))

## phenotype
fl <- system.file(package="DirichletMultinomial", "extdata", "TwinStudy.t")
pheno0 <- scan(fl)
lvls <- c("Lean", "Obese", "Overwt")
pheno <- factor(lvls[pheno0 + 1], levels=lvls)
names(pheno) <- rownames(count)

## subset
```
dim(count)
sum("Lean" == pheno)
dim(csubset("Lean", count, pheno))
dim(csubset("Lean", count, pheno, cidx=FALSE))
Index

* classes
  DMN-class, 6
  DMNGroup-class, 8
* datasets
  data, 4
* manip
  dmn, 5
  dmngroup, 7
  heatmapdmn, 9
  model components, 10
  Utilities, 13
* package
  DirichletMultinomial-package, 2
* stats
  cvdmngroup, 3
  roc, 11
  AIC, DMN-method (model components), 10
  Annotated, 8
  bestgrp (data), 4
  BIC, DMN-method (model components), 10
  csubset (Utilities), 13
  cvdmngroup, 3
  data, 4
  DirichletMultinomial-package, 2, 3, 5, 6, 8
  DMN, 4, 9
  dmn, 3, 5, 6–10
  DMN-class, 6
  DMNGroup, 4
  dmngroup, 7, 8
  DMNGroup-class, 8
  fit (data), 4
  fitted, DMN-method (model components), 10
  fitted, DMNGroup-method (model components), 10
  goodnessOfFit (model components), 10
  heatmapdmn, 9
  image, 9
  laplace (model components), 10
  List, 8
  mixture, 6–9
  mixture (model components), 10
  mixturewt, 11
  mixturewt (model components), 10
  model components, 10
  predict, DMN-method (model components), 10
  predict, DMNGroup-method (model components), 10
  roc, 11
  show, DMN-method (model components), 10
  show, DMNGroup-method (model components), 10
  SimpleList, 8, 9
  summary, DMNGroup-method (model components), 10
  Utilities, 13
  Vector, 8
  xval (data), 4