Package: DirichletRF (via r-universe)

May 22, 2026

Type Package

Title ``Dirichlet Random Forest''

Version 0.1.0

Description Implementation of the Dirichlet Random Forest algorithm for compositional response data. Supports maximum likelihood estimation ('MLE') and method-of-moments ('MOM') parameter estimation for the Dirichlet distribution. Provides two prediction strategies; averaging-based predictions (average of responses within terminal nodes) and parameter-based predictions (expected value derived from the estimated Dirichlet parameters within terminal nodes). For more details see Masoumifard, van der Westhuizen, and Gardner-Lubbe (2026, ISBN:9781032903910).

License GPL-3

Encoding UTF-8

Imports Rcpp (>= 1.0.0), parallel

LinkingTo Rcpp

RoxygenNote 7.3.3

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation yes

Author Khaled Masoumifard [aut, cre] (ORCID: <https://orcid.org/0000-0003-1921-2145>), Stephan van der Westhuizen [aut] (ORCID: <https://orcid.org/0000-0001-9469-8427>), Sugnet Lubbe [aut] (ORCID: <https://orcid.org/0000-0003-2762-9944>)

Maintainer Khaled Masoumifard <masoumifardk@yahoo.com>

Repository https://cran.r-universe.dev

Date/Publication 2026-03-23 17:40:25 UTC

RemoteUrl https://github.com/cran/DirichletRF

RemoteRef HEAD

RemoteSha 1a4a8d32ca89407f17966a5b89df0589642513c0

Contents

`DirichletRF-package` *DirichletRF: Dirichlet Random Forest for Compositional Data*

Description

This package implements Dirichlet Random Forest for modeling and predicting compositional data using maximum likelihood estimation or method of moments.

`DirichletRF` *Build a Dirichlet Random Forest for Compositional Responses*

Description

Build a Dirichlet random forest for compositional responses. In compositional data analysis (CoDA), parts reside in the simplex, and this random forest ensures model output abide by CoDA principles. The implementation uses OpenMP for parallel tree building. Note that this implementation does not support out-of-bag (OOB) error estimation.

Usage

```
DirichletRF(
  X,
  Y,
  num.trees = 100,
  max.depth = 10,
  min.node.size = 5,
  mtry = -1,
  seed = 123,
  est.method = "mom",
  num.cores = -1
)
```

Arguments

X	A numeric (n x p) matrix of covariates. Note that the current version only allows numeric covariates. Users may use one-hot encoding to possibly include categorical covariates.
Y	A numeric (n x k) matrix of compositional responses. Each row should sum to 1. That is, data should already be normalised if needed.
num.trees	Number of trees grown in the forest. Default is 100.
max.depth	Maximum depth of trees. Default is 10.
min.node.size	Minimum size of observations in each tree leaf. Default is 5. Note that nodes with sizes smaller than min.node.size can occur.
mtry	Number of covariates randomly selected as candidates at each split. Default is sqrt(p), indicated by -1.
seed	The seed of the C++ random number generator.
est.method	Parameter estimation method for the Dirichlet distribution when splitting is done. Users may either use maximum likelihood ("mle") or method of moments ("mom"). Default is "mom".
num.cores	Number of OpenMP threads used for parallel tree building. The default is -1 which uses all the cores on the system minus 1. Users may also specify 1 which means that the forest will be built sequentially.

Details

The forest provides two types of fitted values: mean-based predictions (derived from sample means at each leaf) and parameter-based predictions (derived from normalised Dirichlet alpha parameters).

Value

A list of the forest which contains the following:

type Parallelisation type used: "openmp" or "sequential".

num.cores Number of cores used.

num.trees Total number of trees in the forest.

Y_train Training data used (no OOB).

fitted A list of fitted values:

- alpha_hat Estimated Dirichlet alpha parameters (n x k matrix).
- mean_based Mean-based fitted values (n x k matrix).
- param_based Parameter-based fitted values obtained by normalising alpha_hat (n x k matrix).

residuals A list of residuals (Y - fitted):

- mean_based Residuals from mean-based predictions.
- param_based Residuals from parameter-based predictions.

References

Masoumifard, K., van der Westhuizen, S., & Gardner-Lubbe, S. (In press). Dirichlet-random forest for predicting compositional data. In A. Bekker, J. Ferreira, & P. Nagar (Eds.), *Environmental Statistics: Innovative Methods and Applications*. CRC Press.

See Also

predict.dirichlet_forest for making predictions on new data.

Examples

```
# Small toy example (auto-tested)
set.seed(42)
n <- 50; p <- 2
X <- matrix(rnorm(n * p), n, p)
G <- matrix(rgamma(n * 3, shape = rep(c(2, 3, 4), each = n)), n, 3)
Y <- G / rowSums(G)
forest <- DirichletRF(X, Y, num.trees = 5, num.cores = 1)
print(forest)
Xtest <- matrix(rnorm(5 * p), 5, p)
pred  <- predict(forest, Xtest)


# Larger example
n <- 500; p <- 4
X <- matrix(rnorm(n * p), n, p)
alpha <- c(2, 3, 4)
G <- matrix(rgamma(n * length(alpha), shape = rep(alpha, each = n)),
            n, length(alpha))
Y <- G / rowSums(G)

forest1 <- DirichletRF(X, Y, num.trees = 100, num.cores = 1)
forest2 <- DirichletRF(X, Y, num.trees = 100)

alpha_hat   <- forest1$fitted$alpha_hat
mean_fit    <- forest1$fitted$mean_based
param_fit   <- forest1$fitted$param_based
resid_mean  <- forest1$residuals$mean_based
resid_param <- forest1$residuals$param_based

Xtest <- matrix(rnorm(10 * p), 10, p)
pred  <- predict(forest1, Xtest)
param_pred <- pred$alpha_predictions / rowSums(pred$alpha_predictions)
```

---

predict.dirichlet_forest

Predict with a Dirichlet Forest

Description

Makes predictions using a fitted `dirichlet_forest` object returned by `DirichletRF`.

Usage

```
## S3 method for class 'dirichlet_forest'
predict(object, newdata, ...)
```

Arguments

`object`	A `dirichlet_forest` object.
`newdata`	A numeric matrix of new covariates (n_new x p).
`...`	Currently unused.

Value

A list with the following elements:

`alpha_predictions` Estimated Dirichlet alpha parameters for each new observation (n_new x k matrix).

`mean_predictions` Mean-based compositional predictions (n_new x k matrix).

Examples

```
# Small toy example (auto-tested)
set.seed(42)
n <- 50; p <- 2
X <- matrix(rnorm(n * p), n, p)
G <- matrix(rgamma(n * 3, shape = rep(c(2, 3, 4), each = n)), n, 3)
Y <- G / rowSums(G)
forest  <- DirichletRF(X, Y, num.trees = 5, num.cores = 1)
Xtest   <- matrix(rnorm(5 * p), 5, p)
pred    <- predict(forest, Xtest)
pred$mean_predictions


n <- 500; p <- 4
X <- matrix(rnorm(n * p), n, p)
alpha <- c(2, 3, 4)
G <- matrix(rgamma(n * length(alpha), shape = rep(alpha, each = n)),
            n, length(alpha))
Y <- G / rowSums(G)
forest <- DirichletRF(X, Y, num.trees = 50, num.cores = 1)
Xtest  <- matrix(rnorm(10 * p), 10, p)
pred   <- predict(forest, Xtest)
param_pred  <- pred$alpha_predictions / rowSums(pred$alpha_predictions)
single_pred <- predict(forest, Xtest[1, , drop = FALSE])
```

`print.dirichlet_forest` *Custom Print Method for dirichlet_forest Objects*

Description

Suppresses the display of large data matrices (Y_train, fitted, residuals) when the object is printed, while keeping them accessible via `$`.

Usage

```
## S3 method for class 'dirichlet_forest'
print(x, ...)
```

Arguments

`x`	A `dirichlet_forest` object.
`...`	Further arguments passed to or from other methods.

Value

Invisibly returns `x`, the `dirichlet_forest` object unchanged. Called primarily for its side effect of printing a summary of the model to the console.

Index