Package 'RandomGaussianNB'

<div align="center">January 7, 2026</div>

Type Package

Title Randomized Feature and Bootstrap-Enhanced Gaussian Naive Bayes Classifier

Version 0.2.4

Date 2025-12-21

Description Provides an accessible and efficient implementation of a randomized feature and bootstrap-enhanced Gaussian naive Bayes classifier. The method combines stratified bootstrap resampling with random feature subsampling and aggregates predictions via posterior averaging. Support is provided for mixed-type predictors and parallel computation. Methods are described in Srisuradetchai (2025) <doi:10.3389/fdata.2025.1706417> ``Posterior averaging with Gaussian naive Bayes and the R package RandomGaussianNB for big-data classification''.

License MIT + file LICENSE

Encoding UTF-8

Imports parallel, stats

RoxygenNote 7.3.3

Suggests mlbench, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Patchanok Srisuradetchai [aut, cre]

Maintainer Patchanok Srisuradetchai <patchanok@mathstat.sci.tu.ac.th>

Repository CRAN

Date/Publication 2026-01-07 08:00:14 UTC

Contents

predict.random_gaussian_nb

Predict from a random_gaussian_nb model

Description

Predict from a random_gaussian_nb model

Usage

```
## S3 method for class 'random_gaussian_nb'
predict(object, newdata = NULL, type = c("class", "prob"), ...)
```

Arguments

object	A fitted random_gaussian_nb object.
newdata	A data.frame of predictors. If NULL, uses training predictors.
type	"class" (default) or "prob".
...	currently unused.

Value

If type = "prob", returns a data.frame with one column per class giving posterior probabilities averaged over the bootstrap ensemble (rows correspond to observations in newdata).

If type = "class", returns a factor of predicted class labels with levels equal to the training classes.

---

print.random_gaussian_nb

Train a Random Naive Bayes Model via Bootstrap + Random Subspace (Mixed Types)

Description

Fits an ensemble Naive Bayes classifier by repeating (i) stratified bootstrap resampling of rows and (ii) random feature-subset selection, then aggregates predictions by posterior averaging.

Usage

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

## S3 method for class 'random_gaussian_nb'
summary(object, ...)
```

```
## S3 method for class 'random_gaussian_nb'
str(object, ...)

## S3 method for class 'random_gaussian_nb'
nobs(object, ...)

## S3 method for class 'random_gaussian_nb'
fitted(object, ...)

## S3 method for class 'random_gaussian_nb'
plot(
  x,
  which = c("feature_frequency", "prior_variability", "prob_entropy"),
  newdata = NULL,
  top = 20,
  ...
)

random_gaussian_nb(
  data,
  response,
  n_iter = 100,
  feature_fraction = 0.5,
  cores = 1,
  laplace = 1
)
```

Arguments

x	A random_gaussian_nb object.
...	Passed to the underlying plotting function (e.g., barplot(), boxplot(), hist()).
object	A random_gaussian_nb object.
which	Diagnostic to plot: "feature_frequency", "prior_variability", or "prob_entropy".
newdata	Optional new data for "prob_entropy". If NULL, uses the training data.
top	Number of top features to show for "feature_frequency".
data	A data.frame containing predictors and the response.
response	Name of the response column (string).
n_iter	Positive integer; number of bootstrap iterations.
feature_fraction	Numeric in (0,1]; fraction of features sampled each iteration.
cores	Positive integer; number of parallel workers.
laplace	Numeric >= 0; Laplace smoothing parameter for categorical features.

Details

Numeric predictors use Gaussian likelihoods; categorical predictors (factor/character/logical) use multinomial likelihoods with Laplace smoothing.

Numeric predictors use Gaussian likelihoods; categorical predictors (factor/character/logical) use multinomial likelihoods with Laplace smoothing.

The following S3 methods are available for this class:

`print(x, ...)` Returns x invisibly (called for side effects).

`summary(object, ...)` Returns `object` invisibly (prints a summary).

`str(object, ...)` Returns `object` invisibly (prints a compact structure).

`nobs(object, ...)` Returns an integer: number of training observations.

`fitted(object, ...)` Returns a factor of fitted class labels for the training data.

`plot(x, ...)` Returns x invisibly (called for its side effects).

Value

An object of class `"random_gaussian_nb"` containing the fitted bootstrap ensemble and training metadata.

Index