Package ‘banter’

February 12, 2023

Type Package
Title BioAcoustic eveNT classifier
Version 0.9.6
Depends R (>= 4.1.0), magrittr
Imports dplyr (>= 1.0.6), ggplot2 (>= 3.3.3), gridExtra (>= 2.3), methods, parallel, randomForest (>= 4.6), rfPermute (>= 2.5.1), rlang, stats, swfscMisc (>= 1.5), tibble (>= 3.1.2), tidyR (>= 1.1.1)
Suggests testthat
License GNU General Public License
RoxygenNote 7.2.3
Encoding UTF-8
NeedsCompilation no
Author Eric Archer [aut, cre], Taiki Sakai [aut]
Maintainer Eric Archer <eric.archer@noaa.gov>
Repository CRAN
Date/Publication 2023-02-12 21:32:29 UTC

R topics documented:

banter-package .................................................. 2
addBanterDetector .......................................... 2
banterGuide .................................................. 4
getBanterModel ............................................. 4
getBanterModelData ........................................ 5
getDetectorNames .......................................... 6
getSampSize ................................................ 7
addBanterDetector

Description

Add a BANTER Detector Model

Usage

addBanterDetector(
  x,
  data,
  name,
  ntree,
  sampsize = 1,
  importance = FALSE,
  num.cores = 1
)

removeBanterDetector(x, name)
**Arguments**

- **x**: a `banter_model` object.
- **data**: detector data.frame or named list of detector data.frames. If a data.frame, then name must be provided.
- **name**: detector name.
- **ntree**: number of trees.
- **sampsize**: number or fraction of samples to use in each tree. If < 1, then it will be used to select this fraction of the smallest sample size.
- **importance**: retain importance scores in model? Defaults to `FALSE` and will be ignored if num.cores > 1.
- **num.cores**: number of cores to use for Random Forest model. Set to `NULL` to use the maximum number detected on your system - 1.

**Value**

A `banter_model` object with the detector model added or removed.

**Author(s)**

Eric Archer <eric.archer@noaa.gov>

**References**


**Examples**

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add the 'bp' (burst pulse) detector model
bant.mdl <- addBanterDetector(
  x = bant.mdl,
  data = train.data$detectors$bp,
  name = "bp",
  ntree = 50, sampsize = 1, num.cores = 1
)
bant.mdl

# remove the 'bp' detector model
bant.mdl <- removeBanterDetector(bant.mdl, "bp")
bant.mdl
banterGuide  

**BANTER Guide**

**Description**
Open a browser window displaying "BANTER: A User’s Guide to Acoustic Classification".

**Usage**
banterGuide()

**Author(s)**
Eric Archer <eric.archer@noaa.gov>

**References**

getBanterModel  

**Extract Random Forest Model**

**Description**
Extract BANTER event or detector Random Forest model.

**Usage**
getBanterModel(x, model = "event")

**Arguments**
- `x` a *banter_model* object.
- `model` name of model to extract. Default is "event" to extract the event-level model. Can also be the name of a detector.

**Value**
a *randomForest* model object.

**Author(s)**
Eric Archer <eric.archer@noaa.gov>
References


Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
    bant.mdl, train.data$detectors,
    ntree = 50, sampsize = 1, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)

# extract the event randomForest model
event.rf <- getBanterModel(bant.mdl)
event.rf

# extract the burst pulse (bp) detector model
bp.rf <- getBanterModel(bant.mdl, "bp")
bp.rf

getBanterModelData  Extract Random Forest Model Data

Description

Extract BANTER event data used for the Random Forest model.

Usage

ggetBanterModelData(x)

Arguments

x  a banter_model object.

Value

the event data frame used to build the input model x.

Author(s)

Eric Archer <eric.archer@noaa.gov>
getDetectorNames

References


Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
    bant.mdl, train.data$detectors, 
    ntree = 50, sampsize = 1, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)

event.df <- getBanterModelData(bant.mdl)
head(event.df)

generic function

getDetectorNames

Description

Return names of detectors loaded in BANTER model.

Usage

generic function

Arguments

x

a banter_model object.

Value

a vector of names.

Author(s)

Eric Archer <eric.archer@noaa.gov>

References

getSampSize

Examples

data(train.data)
    # initialize BANTER model with event data
    bant.mdl <- initBanterModel(train.data$events)
    # add all detector models
    bant.mdl <- addBanterDetector(
        bant.mdl, train.data$detectors,
        ntree = 50, sampsize = 1, num.cores = 1
    )
    getDetectorNames(bant.mdl)

---

getSampSize | Sample Size

### Description

Return sample sizes used for a BANTER model.

### Usage

getSampSize(x, model = "event")

### Arguments

- **x**: a \texttt{banter\_model} object.
- **model**: name of model to extract. Default is "event" to return values for the event-level model. Can also be name of a detector.

### Value

a vector of sample sizes.

### Author(s)

Eric Archer <eric.archer@noaa.gov>

### References

Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
    bant.mdl, train.data$detectors,
    ntree = 50, sampsize = 2, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)

# sample size for the event model
getSampSize(bant.mdl)

# sample size for the burst pulse (bp) detector model
getSampSize(bant.mdl, "bp")

initBanterModel  Initialize BANTER model

Description

Initialize a BANTER model with event data.

Usage

initBanterModel(x)

Arguments

x  a data.frame of events. Every row is a unique event. Must have columns named event.id and species. All other columns will be used as predictor variables for the BANTER event classifier model.

Value

a banter_model object without any detector models.

Note

Values in the column species are passed through the make.names function on creation to ensure they don't include invalid characters.

Author(s)

Eric Archer <eric.archer@noaa.gov>
References


Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
bant.mdl

###

<table>
<thead>
<tr>
<th>modelPctCorrect</th>
<th>Model Percent Correct</th>
</tr>
</thead>
</table>

Description

Extract percent correctly classified by species for detector and event models.

Usage

modelPctCorrect(x)

Arguments

x a banter_model object.

Value

a data.frame with the percent correctly classified for each model in x.

Author(s)

Eric Archer <eric.archer@noaa.gov>

References

Examples

```r
data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
  bant.mdl, train.data$detectors,
  ntree = 50, sampsize = 1, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)
modelPctCorrect(bant.mdl)
```

numCalls

<table>
<thead>
<tr>
<th>numCalls</th>
<th>Number and Proportion of Calls</th>
</tr>
</thead>
</table>

Description

Return the number and proportion of calls in BANTER detector models.

Usage

```r
numCalls(x, by = c("species", "event"))
propCalls(x, by = c("species", "event"))
```

Arguments

- `x` a `banter_model` object.
- `by` return summary by "species" or "event".

Author(s)

Eric Archer <eric.archer@noaa.gov>

References

Examples

```r
data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
  bant.mdl, train.data$detectors,
  ntree = 50, sampsize = 1, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)

# number of calls by species and event
numCalls(bant.mdl, "species")
numCalls(bant.mdl, "event")

# proportion of calls by species and event
propCalls(bant.mdl, "species")
propCalls(bant.mdl, "event")
```

<table>
<thead>
<tr>
<th>numEvents</th>
<th>Number of Events</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Description

Return the number of events in a BANTER model by species.

Usage

```r
numEvents(x, model = "event")
```

Arguments

- `x` a `banter_model` object.
- `model` name of model to extract. Default is "event" to summarize the event-level model. Can also be name of a detector.

Value

a data.frame giving the number of events available for each species.

Author(s)

Eric Archer <eric.archer@noaa.gov>
plotplotDetectorTrace

References


Examples

data(train.data)

# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)

# add all detector models
bant.mdl <- addBanterDetector(
bant.mdl, train.data$detectors,
    ntree = 50, sampsize = 1, num.cores = 1
)

# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)

# number of events in event model
numEvents(bant.mdl)

# number of events in burst pulse (bp) detector model
numEvents(bant.mdl, "bp")

plotDetectorTrace

Plot BANTER Detector Traces

Description

Plot traces of OOB error rates for detector Random Forest models.

Usage

plotDetectorTrace(x, detector = NULL)

Arguments

x

a banter_model object.

detector

names of models to plot. If set to NULL, traces for all models will be shown.

Author(s)

Eric Archer <eric.archer@noaa.gov>

References

predict

See Also

plotTrace

Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
  bant.mdl, train.data$detectors,
  ntree = 50, sampsize = 1, num.cores = 1
)

plotDetectorTrace(bant.mdl)

---

predict  

Predict BANTER events

Description

Predict species of events for novel data from a BANTER model.

Usage

predict(object, ...)

## S3 method for class 'banter_model'
predict(object, new.data, ...)

## S4 method for signature 'banter_model'
predict(object, new.data, ...)

Arguments

object  
a banter_model object.

...  
unused.

new.data  
a list of event and detector data that has the same predictors as in the banter_model. It must contain elements called events and detectors. The events element must be a data.frame that has a column called event.id and the same predictor columns as the event data used to initialize the banter model (see initBanterModel). The detectors element must be a named list with the same detectors used to build the model (see addBanterDetector).
Value

A list with the following elements:

- **events**  the data frame used in the event model for predictions.
- **predict.df**  data.frame of predicted species and assignment probabilities for each event.
- **detector.freq**  data.frame giving the number of events available for each detector.
- **validation.matrix**  if species is a column in new.data, a table giving the classification rate for each event.

Note

At least one detector in the model must be present in new.data. Any detectors in the training model that are absent will have all species proportions and the the detector proportion set to 0. If a column called species is in new.data, columns for the original species designation and if that matches predicted (correct) will be added to the predict.df data.frame of the output.

Author(s)

Eric Archer <eric.archer@noaa.gov>

References


Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
  bant.mdl, train.data$detectors,
  ntree = 50, sampsize = 2, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)

# predict test data
data(test.data)
test.pred <- predict(bant.mdl, test.data)
test.pred
runBanterModel

Description

Build full event classifier model

Usage

runBanterModel(x, ntree, sampsize = 1)

Arguments

<table>
<thead>
<tr>
<th>x</th>
<th>a banter_model object.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ntree</td>
<td>number of trees.</td>
</tr>
<tr>
<td>sampsize</td>
<td>number or fraction of samples to use in each tree.</td>
</tr>
</tbody>
</table>

Value

a banter_model object with the complete BANTER model.

Author(s)

Eric Archer <eric.archer@noaa.gov>

References


Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
  bant.mdl, train.data$detectors,
  ntree = 50, sampsize = 1, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)
summary(bant.mdl)
subsampleDetections  
Subsample Detections

Description

Extract a random subsample of detections for each event and detector.

Usage

subsampleDetections(data, n)

Arguments

data  
a detector data.frame or list of detector data.frames.

n  
a value giving the number (n >= 1) or fraction (n between 0 and 1) of detections per event per detector to select. Detections are randomly selected without replacement. If n is greater than the number of detections in an event, all detections for that event will be retained.

Value

a detector data.frame or list of detector data.frames with no more than n detections per event per detector.

Author(s)

Eric Archer <eric.archer@noaa.gov>

References


Examples

data(train.data)

# initial number of detections per event per detector
sapply(train.data$detectors, function(x) table(x$event.id))

# select half of all detectors
detect.half <- subsampleDetections(train.data$detectors, 0.5)
sapply(detect.half, function(x) table(x$event.id))

# select 20 detections
detect.20 <- subsampleDetections(train.data$detectors, 20)
sapply(detect.20, function(x) table(x$event.id))
# select 10 detections from 'ec' detector
ec.10 <- subsampleDetections(train.data$detectors$ec, 10)
table(ec.10$event.id)

---

**Summary**

**BANTER Classifier Model Summary**

**Description**

Display summaries for event and detector models

**Usage**

```r
summary(object, ...)  
## S3 method for class 'banter_model'
summary(object, model = "event", n = 0.5, bins = 20, ...)

## S4 method for signature 'banter_model'
summary(object, model = "event", n = 0.5, bins = 20, ...)
```

**Arguments**

- `object`: a `banter_model` object.
- `...`: ignored.
- `model`: name of model to summarize. Default is "event" to summarize the event-level model. Can also be name of a detector.
- `n`: number of final iterations to summarize OOB error rate for. If between 0 and 1 is taken as a proportion of chain.
- `bins`: number of bins in inbag histogram.

**Value**

In the plot that is created, the upper panel shows the trace of the Random Forest model OOB rate across sequential trees in the forest. The lower plot shows a frequency histogram of the number of times each sample was inbag (used as training data in a tree in the forest). The vertical red lines indicate the expected inbag rate for samples of each species.

**Author(s)**

Eric Archer <eric.archer@noaa.gov>
References


Examples

data(train.data)
# initialize BANTER model with event data
bant.mdl <- initBanterModel(train.data$events)
# add all detector models
bant.mdl <- addBanterDetector(
   bant.mdl, train.data$detectors,
   ntree = 50, sampsize = 1, num.cores = 1
)
# run BANTER event model
bant.mdl <- runBanterModel(bant.mdl, ntree = 1000, sampsize = 1)
summary(bant.mdl)

data(test.data)

Description

A list of events and call data from detectors for testing BANTER model

Usage

data(test.data)

Format

list

train.data

Description

A list of events and call data from detectors for training BANTER model

Usage

data(train.data)

Format

list
Index

* datasets
  test.data, 18
  train.data, 18

* package
  banter-package, 2
  addBanterDetector, 2, 13
  banter (banter-package), 2
  banter-package, 2
  banter_model, 3–13, 15, 17
  banterGuide, 4
  getBanterModel, 4
  getBanterModelData, 5
  getDetectorNames, 6
  getSampSize, 7
  initBanterModel, 8, 13
  make.names, 8
  modelPctCorrect, 9
  numCalls, 10
  numEvents, 11
  plotDetectorTrace, 12
  plotTrace, 13
  predict, 13
  predict, banter_model-method (predict), 13
  predict.banter_model (predict), 13
  propCalls (numCalls), 10
  randomForest, 4
  removeBanterDetector
    (addBanterDetector), 2
  runBanterModel, 15
  subsampleDetections, 16
  summary, 17

summary, banter_model-method (summary), 17
summary.banter_model (summary), 17

test.data, 18
train.data, 18