SMAP
April 19, 2009

GBM  
Glioblastoma multiforme array CGH data

Description
Array CGH data measurements of glioblastoma multiforme sample G24460.

Usage
data(GBM)

Source
Genome wide array CGH data from Diaz de Stahl, T., et al. (2005).

References

See Also
smap

Examples
data(GBM)
observations <- SMAPObservations{
  value=as.numeric(GBM[,2]),
  chromosome=as.character(GBM[,3]),
  startPosition=as.numeric(GBM[,4]),
  endPosition=as.numeric(GBM[,5]),
  name="G24460",
  reporterId=as.character(GBM[,1])
}

plot(observations)
SMAPHMM-class

Class "SMAPHMM": A class to manage HMMs for the SMAP package

Description

Holds parameters for a Hidden Markov Model (HMM) used in the SMAP package.

Objects from the Class

Objects are not intended to be created directly but via the constructor function SMAPHMM.

Slots

A: Object of class "matrix" The transition probability matrix between states.

Pi: Object of class "numeric" The initial probabilities of starting in a certain state.

Phi: Object of class "matrix" A matrix that specifies the parameters of Gaussian distributions associated with each hidden state. The first column specifies standard deviations, the second specifies means.

noStates: Object of class "numeric" The number of hidden states in the HMM.

Methods

A signature(object = "SMAPHMM"): Returns the transition matrix.

Pi signature(object = "SMAPHMM"): Returns the initial probabilities.

Phi signature(object = "SMAPHMM"): Returns the distribution parameter matrix.

noStates signature(object = "SMAPHMM"): Returns the number of hidden states in the HMM.

Author(s)

Robin Andersson, (robin.andersson@lcb.uu.se)

References


See Also

smap, SMAPHMM
SMAPHMM

Constructor for "SMAPHMM" objects

Description

A constructor for SMAPHMM-class objects.

Usage

SMAPHMM(noStates, Phi, A=NULL, Pi=rep(1/noStates,noStates), initTrans=0.2/(noStates - 1))

Arguments

| noStates   | The number of hidden states in the HMM (numeric). |
| Phi        | A Gaussian distribution parameter matrix (numeric). |
| A          | A noStates * noStates matrix of transition probabilities between the hidden states (numeric). |
| Pi         | A vector of initial probabilities of starting in a certain state (numeric). |
| initTrans  | Specifies the transition probability between non-equal states (numeric). |

Details

Phi is a noStates * 2 matrix that specifies the parameters of Gaussian distributions associated with each hidden state. The first column specifies standard deviations, the second specifies means.

If A == NULL, initTrans specifies the transition probability between states i and j in 1:noStates, such that i != j. Only used if A == NULL. initTrans * noStates must be smaller than (or equal to) 1.

Value

An object of class SMAPHMM-class.

Author(s)

Robin Andersson, (robin.andersson@lcb.uu.se)

References


See Also

smap, SMAPHMM-class, SMAPObservations-class
Class "SMAPObservations": A class to manage microarray observations for the SMAP package

Description

Holds observed microarray intensity ratios and clone annotations for the SMAP package.

Objects from the Class

Objects can be created by calls of the form `new("SMAPObservations", value, chromosome, startPosition, endPosition, name, reporterId)`. You can also use the convenience function `SMAPObservations`.

Slots

- **value**: Object of class "numeric" Microarray intensity ratios.
- **chromosome**: Object of class "character" Associated chromosomes for the observations.
- **startPosition**: Object of class "numeric" Associated start positions for the observations.
- **endPosition**: Object of class "numeric" Associated end positions for the observations.
- **reporterId**: Object of class "character" Identifiers of the observations, e.g., probe ids.
- **name**: Object of class "character" An identifier of the observation set.
- **noObservations**: Object of class "numeric" The number of observations in the set.

Methods

- **value signature(object = "SMAPObservations")**: Returns the values of the observations.
- **chromosome signature(object = "SMAPObservations")**: Returns the chromosome annotations of the observations.
- **startPosition signature(object = "SMAPObservations")**: Returns the start positions of the observations.
- **endPosition signature(object = "SMAPObservations")**: Returns the end positions of the observations.
- **reporterId signature(object = "SMAPObservations")**: Returns the identifiers of the observations.
- **name signature(object = "SMAPObservations")**: Returns the name of the observation set.
- **noObservations signature(object = "SMAPObservations")**: Returns the number of observations in the set.
- **initialize signature(.Object = "SMAPObservations")**: Creates an instance.
- **plot signature(x = "SMAPObservations", y = "missing")**: A plot method for the observations.
- **[ signature(x = "SMAPObservations")**: Creates a new object of class SMAPObservations with extracted elements as specified by the indices provided.
SMAPObservations

Author(s)
Robin Andersson, (robin.andersson@lcb.uu.se)

References

See Also
smap, SMAPObservations

SMAPObservations  Constructor for "SMAPObservations" objects

Description
A constructor for SMAPObservations-class objects.

Usage
SMAPObservations(value, chromosome, startPosition, endPosition,
   name=character(0),
   reporterId=as.character(1:length(value)))

Arguments
value  A vector of microarray intensity ratios (numeric).
chromosome  A vector of chromosome annotations (character).
startPosition  A vector of start positions (numeric).
endPosition  A vector of end positions (numeric).
name  The name of the observation set (character).
reporterId  A vector of observation identifiers, e.g., probe ids (character).

Details
The vectors value, chromosome, startPosition, endPosition, and reporterId must be of equal length.

Value
An object of class SMAPObservations-class.

Author(s)
Robin Andersson, (robin.andersson@lcb.uu.se)
SMAPProfile-class

References


See Also

smap, SMAPObservations-class, SMAPHMM-class

Examples

```r
## Load Glioblastoma multiforme data
data(GBM)
observations <- SMAPObservations(value=as.numeric(GBM[,2]),
                                   chromosome=as.character(GBM[,3]),
                                   startPosition=as.numeric(GBM[,4]),
                                   endPosition=as.numeric(GBM[,5]),
                                   name="G24460",
                                   reporterId=as.character(GBM[,1]))

## plot observations
plot(observations, ylim=c(0,2))
## plot subset of observations (chromosome 9)
ids <- which(chromosome(observations) == "9")
plot(observations[ids])
```

---

SMAPProfile-class  
Class "SMAPProfile"

Description

Holds results from running `smap`.

Objects from the Class

Objects are not intended to be created directly but as a result from running `smap`.

Slots

- **HMM**: Object of class "SMAPHMM"
- **observations**: Object of class "SMAPObservations"
- **P**: Object of class "numeric" The log joint posterior probability of the state sequence $Q$ and parameters of HMM given the observations.
- **Q**: Object of class "numeric" The optimal state sequence (path) in the HMM.
- **name**: Object of class "character" The name of the object.
Methods

P  signature(object = "SMAPProfile"): Returns the log joint posterior probability.
Q  signature(object = "SMAPProfile"): Returns the optimal state sequence.
HMM signature(object = "SMAPProfile"): Returns the optimized HMM.
name signature(object = "SMAPProfile"): Returns the name of the profile.
observations signature(object = "SMAPProfile"): Returns the observations.
plot signature(x = "SMAPProfile", y = "missing"): A plot method for the result profile.
[ signature(x = "SMAPProfile"): Creates a new object of class SMAPProfile with extracted elements as specified by the indices provided.

Author(s)

Robin Andersson, (robin.andersson@lcb.uu.se)

References


See Also

smap, SMAPProfiles-class
Methods

- **Q** signature(object = "SMAPProfiles"): Returns the optimal state sequence of the list elements.
- **observations** signature(object = "SMAPProfiles"): Returns the observations of the list elements.
- **name** signature(object = "SMAPProfiles"): Returns the name of the profile.
- **plot** signature(x = "SMAPProfiles", y = "missing"): A plot method for the result profiles.

Author(s)

Robin Andersson, ⟨robin.andersson@lcb.uu.se⟩

References


See Also

- smap, SMAPProfile-class

smap

*smap: A Segmental Maximum A Posteriori Approach to Array-CGH Copy Number Profiling*

**Description**

This function fits a Hidden Markov Model (HMM) to a set of observed microarray intensity ratios and outputs the most plausible state sequence in the HMM through segmental a posteriori maximization.

Briefly, given an HMM with initial parameter settings \( \lambda \) and a set of observations \( O \), the method alternates maximization of the joint posterior probability of the state sequence \( Q \) and \( \lambda \) given \( O, p(Q, \lambda | O) \), over \( Q \) (using a modified Viterbi algorithm) and \( \lambda \) (using a gradient descent scheme with individual learning rate adaptation).

**Usage**

smap(x, Obs, sd.min=0.05, mean.sd=0.05, max.iters=Inf, gd.max.iters=Inf, tau=0.05, eta=0.01, e.change=0.5, e.same=1.2, e.min=0.0001, e.max=0.5, adaptive=TRUE, overlap=TRUE, distance=TRUE, chrom.wise=FALSE, verbose=1, L=5000000)
Arguments

- **x**: An object of class `SMAPHMM-class`.
- **Obs**: An object of class `SMAPObservations-class`.
- **sd.min**: The minimum allowed standard deviation of state associated Gaussian distributions (numeric).
- **mean.sd**: Prior standard deviation of state associated Gaussian means (numeric).
- **max.iters**: Maximum number of iterations in the SMAP algorithm (numeric).
- **gd.max.iters**: Maximum number of iterations in the gradient descent algorithm per SMAP iteration (numeric).
- **tau**: Minimum log probability improvement required in the SMAP and gradient descent optimization (numeric).
- **eta**: Initial learning rate in the gradient descent optimization (numeric).
- **e.change**: Multiplier for individual learning rate adaptation if the sign of partial derivative changes (numeric). Only used if `adaptive == TRUE`.
- **e.same**: Multiplier for individual learning rate adaptation if the sign of partial derivative stays the same (numeric). Only used if `adaptive == TRUE`.
- **e.min**: Minimum allowed learning rate (numeric).
- **e.max**: Maximum allowed learning rate (numeric).
- **adaptive**: If `TRUE`, individual learning rate adaptation according to Algorithm 1 in Bagos et al. (2004) is used in the gradient descent optimization.
- **overlap**: If `TRUE`, genomic overlap of clones is considered in the optimization.
- **distance**: If `TRUE`, genomic distance between clones is considered in the optimization, in terms of distance based transition probabilities.
- **chrom.wise**: If `TRUE`, the observations are analyzed chromosome-wise rather than genome-wise.
- **verbose**: Specifies the amount of output produced; 0 means no information and 3 a lot of information (numeric).
- **L**: A positive length parameter that controls the convergence of distance based transition probabilities towards $1 / \text{noStates}(x)$ (numeric).

Details

- `sd.min`, `mean.sd`, and `eta` must all be greater than 0. `tau` must be greater than 0 if `max.iters` or `gd.max.iters` are infinite, and can be 0 otherwise. If `adaptive` is `TRUE`, then `e.change` is required to be in the interval $(0,1]$, `e.same` must be greater than or equal to 1, and `e.max` must be greater than 0.

Value

The method returns an object of class `SMAPProfile-class` or `SMAPProfiles-class` if `chrom.wise` is set to `FALSE` or `TRUE`, respectively.

Author(s)

Robin Andersson (robin.andersson@lcb.uu.se)
References


See Also

SMAPHMM, SMAPObservations

Examples

```r
## Load Glioblastoma multiforme data
data(GBM)
observations <- SMAPObservations(value=as.numeric(GBM[,2]),
                                  chromosome=as.character(GBM[,3]),
                                  startPosition=as.numeric(GBM[,4]),
                                  endPosition=as.numeric(GBM[,5]),
                                  name="G24460",
                                  reporterId=as.character(GBM[,1]))
plot(observations, ylim=c(0,2))
## Initiate HMM
init.means <- c(0.4, 0.7, 1, 1.3, 1.6, 3)
init.sds <- rep(0.1, 6)
phi <- cbind(init.means, init.sds)
hmm <- SMAPHMM(6, phi, initTrans=0.02)
hmm
## RUN SMAP:
profile <- smap(hmm, observations, verbose=2)
## genome profile
plot(profile, ylim=c(0,2))
## chromosome 9 profile
ids <- which(chromosome(observations) == "9")
plot(profile[ids], ylim=c(0,2), main="chromosome 9")
## output results for chromosome 9
#cbind(reporterId(observations[ids]), Q(profile[ids]))
```
Index

∗Topic classes
  SMAPHMM, 3
  SMAPHMM-class, 2
  SMAPObservations, 5
  SMAPObservations-class, 4
  SMAPProfile-class, 6
  SMAPProfiles-class, 7

∗Topic datasets
  GBM, 1

∗Topic methods
  smap, 8

A (SMAPHMM-class), 2
chromosome
  (SMAPObservations-class), 4
endPosition
  (SMAPObservations-class), 4

GBM, 1

HMM (SMAPProfile-class), 6
name (SMAPObservations-class), 4
noObservations
  (SMAPObservations-class), 4
noStates (SMAPHMM-class), 2
observations (SMAPProfile-class), 6

P (SMAPProfile-class), 6
Phi (SMAPHMM-class), 2
Pi (SMAPHMM-class), 2
plot, SMAPObservations, missing-method
  (SMAPObservations-class), 4

Q (SMAPProfile-class), 6
reporterId
  (SMAPObservations-class), 4

smap, 1–3, 5–7, 8, 8
SMAPHMM, 2, 3, 10
SMAPHMM-class, 3, 6, 9

SMAPHMM-class, 2
SMAPObservations, 4, 5, 5, 10
SMAPObservations-class, 3, 5, 6, 9
SMAPObservations-class, 4
SMAPProfile-class, 8, 9
SMAPProfile-class, 6, 7
SMAPProfiles-class, 7, 9
SMAPProfiles-class, 7
startPosition
  (SMAPObservations-class), 4
value (SMAPObservations-class), 4