VanillaICE

November 11, 2009

R topics documented:

`breaks` *Identify breakpoints from the hidden Markov model predictions*

Description

Identify breakpoints: physical position of breaks, number of SNPs in region, and the called hidden state.

Usage

```
breaks(x, states, position, chromosome, sampleNames, chromosomeAnnotation = NULL
```

Arguments

`x`	Locus X sample matrix of hidden states where the hidden states are represented as integers
`states`	Labels for the hidden states
`position`	Physical position of loci
`chromosome`	integer indicating chromosome (23=X)
`sampleNames`	sample labels
`chromosomeAnnotation`	chromosome annotation. see details
`verbose`	verbose output

Details

One may provide their own chromosome annotation with centromere start and stop sites. The format must be the same as the chromosomeAnnotation dataset in the R package SNPchip.

Value

data.frame

`sample`	sample label
`chr`	chromosome (23 = X)
`start`	starting physical position of segment
`end`	last physical position of segment
`nbases`	number of bases in segment
`nprobes`	number of probes in segment
`state`	label for the state of the segment

Author(s)

R. Scharpf

Examples

```
x <- matrix(rep(c(1, 2, 3, 1, 2), each=50), ncol=1)
breaks(x, states=c("0", "1", "2"), position=1:nrow(x), chromosome=1,
       sampleNames="A")
```

`copynumberEmission` *Emission probabilities for copy number*

Description

Emission probabilities for copy number

Usage

```
copynumberEmission(copynumber, states, mu, sds, takeLog, verbose = TRUE,
na.rm=TRUE)
```

Arguments

`copynumber`	matrix
`states`	character string
`mu`	numeric: mean of hidden states for Gaussian
`sds`	standard deviations of copy number estimates
`takeLog`	logical: if TRUE, this function takes the log of the copy number AND mu arguments to this function
`verbose`	logical
`na.rm`	The default is to ignore missing values when calculating robust standard deviations

Details

By default, this func estimates the scale parameter for the Normal distribution from the supplied data using the median absolute deviation (MAD). However, different standard deviations can be supplied by the user with the argument `sds`. The supplied standard deviations must be of the same dimension as the copy number matrix.

Value

`array`	Array of emission probabilities on the log scale. Dimension 1: SNPs, Dimension 2: samples, Dimension3: states

See Also

`genotypeEmission`, `genotypeEmissionCrlmm`

`genotypeEmissionCrlmm` *Estimate the emission probabilities using confidence scores from CRLMM*

Description

Estimate the emission probabilities that incorporate information on the confidence scores for the genotype calls.

Usage

```
genotypeEmissionCrlmm(genotypes, conf, pHetCalledHom = 0.001, pHetCalledHet = 0.
```

Arguments

`genotypes`	Matrix of genotypes
`conf`	Matirx of confidence scores (see details).
`pHetCalledHom`	The probability that a truly heterozygous SNP is incorrectly called homozygous (incorrect call).
`pHetCalledHet`	The probability that a truly heterozygous SNP is called heterozygous (correct call).
`pHomInNormal`	The probability of a homozygous genotype call in the 'normal' state.
`pHomInRoh`	The probability of a homozygous genotype call in a region of homozygosity.
`annotation`	The cdf name (e.g., "genomewidesnp6")

Details

The confidence scores by crlmm are saved as an integer: 1000*log(1-p), where p is the probability that the genotype call is correct.

The reference distribution of confidence scores are available for the following Affymetrix platforms: affy6, nsp250, and sty250k.

Value

An R x C x X array of emission probabilities, where

R = number of loci (SNPs) C = number of samples S = number of states

Author(s)

R Scharpf

References

RB Scharpf et al. (2008), Annals of Applied Statistics

`genotypeEmission` *Emission probabilities for di-allelic genotype calls*

Description

Emission probabilities for di-allelic genotype calls

Usage

```
genotypeEmission(genotypes, conf, states, probHomCall, probMissing, verbose=TRUE
```

Arguments

`genotypes`	matrix of integers (1=AA, 2=AB, 3=BB, 4=other)
`conf`	Confidence estimates of the genotype calls obtained from crlmm (optional).
`states`	character string of hidden states
`probHomCall`	numeric: probability of a homozygous genotype call specified in the same order as the hidden states
`probMissing`	numeric: probability of a missing genotype call specified in the same order as the hidden states
`verbose`	logical

Details

CRLMM provides confidences estimates of the genotype calls that can be integrated to improve the HMM. Because CRLMM will genotype all SNPs, the probMissing argument is unecessary.

Value

`array`	Array of emission probabilities. Dimension 1: SNPs, Dimension 2: samples, Dimension3: states

hmm *Wrapper for fitting the HMM*

Description

A wrapper for fitting the HMM.

Usage

```
hmm(object, states, mu = NULL, probs = NULL, takeLog = FALSE, initialP, returnSe
```

Arguments

object	SnpCallSet, SnpCopyNumberSet, or oligoSnpSet object
states	Labels for the hidden states. See details for order.
mu	The latent copy number. See details for order.
probs	See details.
takeLog	Whether to take the log of the copy number before computing emission probabilities and standard deviations
initialP	Initial state probabilities
returnSegments	Logical: whether to return the segments or the loci x sample matrix of predicted states
TAUP	Scaling parameter for transition probabilities.
verbose	Logical: Verbose output?
ice	Whether to use CRLMM confidence scores of the genotype calls.
envir	Optional. An environment for storing intermediate files created for fitting the HMM.

Details

For oligoSnpSet objects, the hidden state labels are assumed to be 1: hemizygous deletion 2: normal 3: region of homozygosity (ROH) 4: amplification

The argument mu should have copy number values corresponding to the above states. For instance on the absolute scale, the copy number states should be 1, 2, 2, and 4.

probs: If ice is FALSE, the elements in probs should correspond to the probability of a homozygous genotype in each of the above states. If ice is TRUE, the elements in probs should correspond to 1. Pr(homozygous call | truth is heterozyous) 2. Pr(heterozygous call | truth is heterozygous) 3. Pr(homozygous call | truth is ROH) 4. Pr(homozygous call | truth is normal) . 'Normal' meaning copy number 2 and a typical frequency of heterozygosity for autosomes.

Value

If returnSegments is TRUE, a data.frame containing the coordinates of the predicted segments is returned. Otherwise, a loci X sample matrix is returned. The elements of the matrix correspond to the predict hidden state for a specific locus and sample.

Author(s)

R. Scharpf

References

RB Scharpf et al. (2008) Hidden Markov Models for the assessment of chromosomal alterations using high-throughput SNP arrays, Annals of Applied Statistics

---

locusLevelData *Basic data elements required for the HMM*

Description

This object is a list containing the basic data elements required for the HMM

Usage

```
data(locusLevelData)
```

Format

A list

Details

The basic assay data elements that can be used for fitting the HMM are:

1. a mapping of platform identifiers to chromosome and physical position

2. (optional) a matrix of copy number estimates

3. (optional) a matrix of confidence scores for the copy number estimates (e.g., inverse standard deviations)

4. (optional) a matrix of genotype calls

5. (optional) CRLMM confidence scores for the genotype calls

At least (2) or (4) is required. The locusLevelData is a list that contains (1), (2), (4), and (5).

Source

A HapMap sample on the Affymetrix 50k platform. Chromosomal alterations were simulated. The last 100 SNPs on chromosome 2 are, in fact, a repeat of the first 100 SNPs on chromosome 1 – this was added for internal use.

Examples

```
data(locusLevelData)
str(locusLevelData)
```

`transitionProbability` *Compute the transition probability*

Description

Wrapper for computing the locus-specific transition probability

Usage

```
transitionProbability(chromosome, position, TAUP = 1e+08, chromosomeAnnotation,
```

Arguments

`chromosome`	chromosome (integer representation)
`position`	physical position
`TAUP`	Scalar for computing transition probabilities (see Details).
`chromosomeAnnotation`	Optional: chromosome annotation
`verbose`	Logical: verbose output

Details

The HMM uses locus-specific transition probabilities that are calculated as a function of the physical distance between loci. Specifically, the probability that the locus at position $t-1$ is not informative for the locus at position t is calculated as $1 - exp(-d/TAUP)$, where d is the physical distance between locus t and locus t-1. The default for TAUP is $1x10^8$ and can be specified to acheive a desired amount of sensitivity and specificity. Larger values of TAUP decreases the probability of transitioning to other states, and therefore provides a more smooth fit.

Value

The transitionProbability function (i) transforms the physical distance between adjacent loci to an estimate of the genomic distance and (ii) adds an 'arm' variable to the annotation matrix.

`chromosome`	chromosome
`position`	physical position
`arm`	an integer. The HMM uses the arm variable as a factor and is fit independently to each 'arm'.
`transitionPr`	locus-specific transition probabilities

Author(s)

R. Scharpf

See Also

`chromosomeAnnotation`

`viterbi` *viterbi algorithm*

Description

The Viterbi algorithm for computing the most likely state sequence given a model

Usage

```
viterbi(initialStateProbs, emission, tau, arm, tau.scale, verbose =
FALSE, chromosome, position, sampleNames, locusNames, normalIndex, returnLikelih
```

Arguments

`initialStateProbs`	initial state probabilities (log scale)
`emission`	matrix of log emission probabilities (one sample is a matrix)
`tau`	transition probabilities (original scale)
`arm`	numeric or character string indicating chromosomal arm
`tau.scale`	matrix to scale the probability of transitioning between states.
`verbose`	Logical
`chromosome`	chromosome
`position`	physical position
`sampleNames`	sample labels
`locusNames`	labels for loci
`normalIndex`	index corresponding to the normal state. See details
`returnLikelihood`	whether to return the 'loglikelihood'

Details

The Viterbi algorithm is fit independently to each chromosomal arm if arm is specified.

Argument `tau.scale` is a matrix that scales the probability of transitioning from an altered state to a normal state to the probability of transitioning between two altered states. If missing, tau.scale is 1 (no scaling)

Value

`matrix`	predicted states

Author(s)

R. Scharpf

Index