genoCN

October 25, 2011

`code.genotype` *code bi-allele genotype to numerical value*

Description

code a genotype vector, e.g. ("AA", "AC", ...) to a numerical vector based on the count of minor allele, e.g., (0, 1, ...)

Usage

```
code.genotype(v)
```

Arguments

`v`	character vector of genotypes

Value

a numerical vector of genotype

Author(s)

Wei Sun wsun@bios.unc.edu

`genoCNA` *Copy Number Aberration*

Description

extract genotype and copy number calls for copy number aberrations, which are often observed in tumor tissues

Usage

```
genoCNA(snpNames, chr, pos, LRR, BAF, pBs, sampleID,
  Para=NULL, fixPara=FALSE, cnv.only=NULL, estimate.pi.r=TRUE,
  estimate.pi.b=TRUE, estimate.trans.m=TRUE, outputSeg = TRUE,
  outputSNP=3, outputTag=sampleID, outputViterbi=FALSE,
  Ds=c(1e10, 1e10, rep(1e8, 7)), pBs.alpha=0.001, contamination=TRUE,
  normalGtp=NULL, geno.error=0.01, min.tp=1e-4, max.diff=0.1,
  distThreshold=1e6, transB=c(0.5,.05,.05,0.1,0.1,.05,.05,.05,.05),
  epsilon=0.005, K=5, maxIt=200, seg.nSNP=3, traceIt=5)
```

Arguments

`snpNames`	a vector of SNP names. SNPs must be ordered by chromosme locations
`chr`	chromosomes of all the SNPs specified in `snpNames`
`pos`	positions of all the SNPs specified in `snpNames`
`LRR`	Log R Ratio of all the SNPs specified in `snpNames`
`BAF`	B Allele Frequency of all the SNPs specified in `snpNames`
`pBs`	population frequency of of all the SNPs specified in `snpNames`
`sampleID`	symbol/name of the studied sample. Only one sample is studied each time
`Para`	a list of initial parameters for the HMM. If Para is NULL, The default initial parameters: init.Para.CNA is used
`fixPara`	if fixPara is TRUE, the parameters in Para are fixed, and are used directly to calculate posterior probabilities. It is not recommended to set fixPara as TRUE for CNA studies.
`cnv.only`	a vector indicating those CNV-only probes, for which we only consider their Log R ratio. If it is NULL, there is no CNV-only probes
`estimate.pi.r`	to estimate pi.r (proportion of uniform component for LRR) or not. By default, estimate.pi.r=FALSE, and the initial value of pi.r is used to estimate other parameters
`estimate.pi.b`	to estimate pi.b (proportion of uniform component for BAF) or not. By default, estimate.pi.b=FALSE, and the initial value of pi.b is used to estimate other parameters
`estimate.trans.m`	to estimate transition probability matrix or not. By default, estimate.trans.m=FALSE, and the initial value of estimate.trans.m is used to estimate other parameters
`outputSeg`	wether to output the information of copy number altered segments
`outputSNP`	if `outputSNP` is 0, do not output SNP specific information; if `outputSNP` is 1, output the most likely copy number and genotype state of the SNPs that are within copy number altered regions; if `outputSNP` is 2, output the most likely copy number and genotype state of all the SNPs (whether it is within CNV regions or not), if `outputSNP` is 3, output the posterior probability for all the copy number and genotype states for the SNPs.
`outputTag`	the prefix of the output files, output of copy number altered segments is written into file outputTag_segment.txt, and output of SNP information is written into file outputTag_SNP.txt

`outputViterbi`
: whether to output the copy altered regions identified by the viterbi algorithm. see details

`Ds`
: Parameter to for transition probability of the HMM. A vector of length N, where N is the number of states in the HMM

`pBs.alpha`
: pBs.alpha is the lower limit of population B allele frequency, and the upper limit is 1 - pBs.alpha

`contamination`
: whether tissue contamination is considered

`normalGtp`
: `normalGtp` is specified only if paired tumor-normal SNP array is availalble. It is the normal tissue genotype for all the SNPs specified in `snpNames`, which can only take four different values: -1, 0, 1, and 2. Values 0, 1, 2 correspond to the number of B alleles, and value -1 indicates the normal genotype is missing. By default, it is NULL, then all the normal genotype are set missing (-1)

`geno.error`
: probability of genotyping error in normal tissue genotypes

`min.tp`
: the minimum of transition probability.

`max.diff`
: Due to normalization procedure, the BAF may not be symmetric. Let's use state (AAA, AAB, ABB, BBB) as an example. Ideally, mean values of normal components AAB and ABB, denoted by mu1 and mu2, respectively, should have the relation mu1 = 1-mu2 if BAF is symmetric. However, this may not be true due to normalization procedures. We restrict the difference of mu1 and (1-mu2) by this parameter max.diff.

`distThreshold`
: If distance between adjacent probes is larger than distThreshold, restart the transition probability by the default values in `transB`.

`transB`
: The default transition probability.

`epsilon`
: see explanation of `K`

`K`
: epsilon and K are used to specify the convergence criteria. We say the estimate.para is converged if for K consecutive updates, the maximum change of parameter estimates in every adjacent step is smaller than epsilon

`maxIt`
: the maximum number of iterations of the EM algorithm to estimate parameters

`seg.nSNP`
: the minimum number of SNPs per segment

`traceIt`
: if traceIt is a integer n, then the running time is printed out in every n iterations of the EM algorithm. if traceIt is 0 or negative, no tracing information is printed out.

Value

results are written into output files

Note

Copy number altered regions are identified, by default, based on the SNP level copy number calls. A CNA region boundary is declared simply when the adjacent SNPs have different copy numbers. An alternative approach is to use viterbi algorithm to output the "best path". Most time the results based on the SNP level copy number calls are the same as the results from viterbi algorithm. For the following up association studies, the SNP level information is more relevant if we examine the association SNP by SNP.

Author(s)

Wei Sun and Zhengzheng Tang

Examples

```
data(snpData)
data(snpInfo)

dim(snpData)
dim(snpInfo)

snpData[1:2,]
snpInfo[1:2,]

snpInfo[c(1001,1100,10001,10200),]

plotCN(pos=snpInfo$Position, LRR=snpData$LRR, BAF=snpData$BAF,
main = "simulated data on Chr22")

snpNames = snpInfo$Name
chr = snpInfo$Chr
pos = snpInfo$Position
LRR = snpData$LRR
BAF = snpData$BAF
pBs = snpInfo$PFB
cnv.only=(snpInfo$PFB>1)
sampleID="simu1"

# Note this simulated data is more of CNV rather than CNA.
# For example, there is no tissue contamination.
# We just use it to illustrate the usage of genoCNA.

Theta = genoCNA(snpNames, chr, pos, LRR, BAF, pBs, contamination=TRUE,
  normalGtp=NULL, sampleID, cnv.only=cnv.only, outputSeg = TRUE,
            outputSNP = 1, outputTag = "simu1")
```

`genoCNV` *Copy Number Variation*

Description

extract genotype and copy number calls for copy number variation, which are inheritable DNA polymorphisms and are observed in normal tissues

Usage

```
genoCNV(snpNames, chr, pos, LRR, BAF, pBs, sampleID,
  Para=NULL, fixPara=FALSE, cnv.only=NULL, estimate.pi.r=TRUE,
  estimate.pi.b=FALSE, estimate.trans.m=FALSE, normLRR=TRUE,
  outputSeg=TRUE, outputSNP=3, outputTag=sampleID, outputViterbi=FALSE,
  Ds = c(1e6, 1e6, rep(1e5, 4)),
  pBs.alpha=0.001, loh=FALSE, output.loh=FALSE,
  min.tp=5e-5, max.diff=0.1, distThreshold=5000,
```

```
transB = c(0.995, 0.005*c(.01, .09, .8, .09, .01)),
epsilon=0.005, K=5, maxIt=200, seg.nSNP=3, traceIt=5)
```

Arguments

snpNames	a vector of SNP names. SNPs must be ordered by chromosome locations
chr	chromosomes of all the SNPs specified in snpNames
pos	positions of all the SNPs specified in snpNames
LRR	Log R Ratio of all the SNPs specified in snpNames
BAF	B Allele Frequency of all the SNPs specified in snpNames
pBs	population frequency of of all the SNPs specified in snpNames
sampleID	symbol/name of the studied sample. Only one sample is studied each time
Para	a list of initial parameters for the HMM. If Para is NULL, The default initial parameters: init.Para.CNV is used
fixPara	if fixPara is TRUE, the parameters in Para are fixed, and are used directly to calculate posterior probabilities
cnv.only	a vector indicating those CNV-only probes, for which we only consider their Log R ratio. If it is NULL, there is no CNV-only probes
estimate.pi.r	to estimate pi.r (proportion of uniform component for LRR) or not. By default, estimate.pi.r=FALSE, and the initial value of pi.r is used to estimate other parameters
estimate.pi.b	to estimate pi.b (proportion of uniform component for BAF) or not. By default, estimate.pi.b=FALSE, and the initial value of pi.b is used to estimate other parameters
estimate.trans.m	to estimate transition probability matrix or not. By default, estimate.trans.m=FALSE, and the initial value of estimate.trans.m is used to estimate other parameters
normLRR	If normLRR is TRUE, we normalize the LRR data by subtracting the median LRR for those LRR between -2 and 2. This strategy has been used by PennCNV.
outputSeg	wether to output the information of copy number altered segments
outputSNP	if outputSNP is 0, do not output SNP specific information; if outputSNP is 1, output the most likely copy number and genotype state of the SNPs that are within copy number altered regions; if outputSNP is 2, output the most likely copy number and genotype state of all the SNPs (whether it is within CNV regions or not), if outputSNP is 3, output the posterior probability for all the copy number and genotype states for the SNPs.
outputTag	the prefix of the output files, output of copy number altered segments is written into file outputTag_segment.txt, and output of SNP information is written into file outputTag_SNP.txt
outputViterbi	whether to output the copy altered regions identified by the viterbi algorithm. see details
Ds	Parameter to for transition probability of the HMM. A vector of length N, where N is the number of states in the HMM
pBs.alpha	pBs.alpha is the lower limit of population B allele frequency, and the upper limit is 1 - pBs.alpha

loh	Whether we use the copy-number-neutral loss of heterozygosity state for CNV studies.
output.loh	Whether we output the loh information.
min.tp	the minimum of transition probability.
max.diff	Due to normalization procedure, the BAF may not be symmetric. Let's use state (AAA, AAB, ABB, BBB) as an example. Ideally, mean values of normal components AAB and ABB, denoted by mu1 and mu2, respectively, should have the relation mu1 = 1-mu2 if BAF is symmetric. However, this may not be true due to normalization procedures. We restrict the difference of mu1 and (1-mu2) by this parameter max.diff.
distThreshold	If distance between adjacent probes is larger than distThreshold, restart the transition probability by the default values in transB.
transB	The default transition probability.
epsilon	see explanation of K
K	epsilon and K are used to specify the convergence criteria. We say the estimate.para is converged if for K consecutive updates, the maximum change of parameter estimates in every adjacent step is smaller than epsilon
maxIt	the maximum number of iterations of the EM algorithm to estimate parameters
seg.nSNP	the minimum number of SNPs per segment
traceIt	if traceIt is a integer n, then the running time is printed out in every n iterations of the EM algorithm. if traceIt is 0 or negative, no tracing information is printed out.

Value

results are written into output files

Note

Copy number altered regions are identified, by default, based on the SNP level copy number calls. A CNV region boundary is declared simply when the adjacent SNPs have different copy numbers. An alternative approach is to use viterbi algorithm to output the "best path". Most time the results based on the SNP level copy number calls are the same as the results from viterbi algorithm. For the following up association studies, the SNP level information is more relevant if we examine the association SNP by SNP.

Author(s)

Wei Sun and Zhengzheng Tang

Examples

```
data(snpData)
data(snpInfo)

dim(snpData)
dim(snpInfo)

snpData[1:2,]
snpInfo[1:2,]
```

```
snpInfo[c(1001,1100,10001,10200),]

plotCN(pos=snpInfo$Position, LRR=snpData$LRR, BAF=snpData$BAF,
main = "simulated data on Chr22")

snpNames = snpInfo$Name
chr = snpInfo$Chr
pos = snpInfo$Position
LRR = snpData$LRR
BAF = snpData$BAF
pBs = snpInfo$PFB
cnv.only=(snpInfo$PFB>1)
sampleID="simu1"

Theta = genoCNV(snpNames, chr, pos, LRR, BAF, pBs,
            sampleID, cnv.only=cnv.only, outputSeg = TRUE,
            outputSNP = 1, outputTag = "simu1")
```

`init.Para.CNA` *Initial parameters for the HMM*

Description

a list of initial values for the parameters of genoCNA.

Usage

```
data(init.Para.CNA)
```

Format

The format is a list of 16 items

- pi.r a vector of length N, where N is the number of states. pi.r[j] is the prior probability of the uniform component of log R ratio for state j
- mu.r a vector of length N, where N is the number of states. mu.r[j] is mean value of the normal component of log R ratio for state j
- sd.r a vector of length N, where N is the number of states. sd.r[j] is standard deviation of the normal component of log R ratio for state j
- mu.r.upper, mu.r.lower two vectors of the same size of mu.r, indicating the upper/lower bound of mu.r
- sd.r.upper, sd.r.lower two vectors of the same size of sd.r, indicating the upper/lower bound of sd.r
- pi.b a vector of length N, where N is the number of states. pi.b[j] is the prior probability of the uniform component of B allele frequency for state j
- mu.b a matrix of N*M, where N is the number of states, and M is the maximum number of components of each states. mu.b[i,j] indicates the mean value of the j-th component of the i-th state
- sd.b a matrix of the same size of mu.b, specifying the standard deviations

- mu.b.upper, mu.b.lower two matrices of the same size of mu.b, incating the upper/lower bound of mu.b
- sd.b.upper, sd.b.lower two matrices of the same size of sd.b, indicating the upper/lower bound of sd.b
- trans.m transition probability matrix of size N*N. The diagonal elements are not used.
- trans.begin a matrix of size S*N, where S is the number of chromosomes, and N is the number of states. trans.begin[s,] are the state probabilities for the fist probe of the s-th chromosome. By default, we assume there is only one chromosome, therefore it is a matrix of 1*N.

Examples

```
data(init.Para.CNA)
```

`init.Para.CNV` *Initial parameters for the HMM of genoCNV*

Description

a list of initial values for the parameters genoCNV.

Usage

```
data(init.Para.CNV)
```

Format

The format is a list of 16 items

- pi.r a vector of length N, where N is the number of states. pi.r[j] is the prior probability of the uniform component of log R ratio for state j
- mu.r a vector of length N, where N is the number of states. mu.r[j] is mean value of the normal component of log R ratio for state j
- sd.r a vector of length N, where N is the number of states. sd.r[j] is standard deviation of the normal component of log R ratio for state j
- mu.r.upper, mu.r.lower two vectors of the same size of mu.r, incating the upper/lower bound of mu.r
- sd.r.upper, sd.r.lower two vectors of the same size of sd.r, indicating the upper/lower bound of sd.r
- pi.b a vector of length N, where N is the number of states. pi.b[j] is the prior probability of the uniform component of B allele frequency for state j
- mu.b a matrix of N*M, where N is the number of states, and M is the maximum number of components of each states. mu.b[i,j] indicates the mean value of the j-th component of the i-th state
- sd.b a matrix of the same size of mu.b, specifying the standard deviations
- mu.b.upper, mu.b.lower two matrices of the same size of mu.b, incating the upper/lower bound of mu.b
- sd.b.upper, sd.b.lower two matrices of the same size of sd.b, indicating the upper/lower bound of sd.b

- trans.m transition probability matrix of size N*N. The diagonal elements are not used.
- trans.begin a matrix of size S*N, where S is the number of chromosomes, and N is the number of states. trans.begin[s,] are the state probabilities for the fist probe of the s-th chromosome. By default, we assume there is only one chromosome, therefore it is a matrix of 1*N.

Examples

```
data(init.Para.CNV)
```

`plotCN` *plot LRR, BAF, and the copy number estimates*

Description

plot LRR, BAF, and the copy number estimates of genoCNV and/or PennCNV.

Usage

```
plotCN(pos, LRR, BAF, chr2plot = NULL, sampleIDs = NULL, fileNames=NULL,
types = "genoCN", CNA = TRUE, main = "", LRR.ylim=NULL,
cex=0.5, plot.lowess=TRUE)
```

Arguments

`pos`	position of all the SNPs
`LRR`	a vector of the log R ratio, should be one-to-one correspondence of pos
`BAF`	a vector of the B allele frequency, should be one-to-one correspondence of pos
`chr2plot`	which chromosome to plot. Only one chromosome can be plotted each time
`sampleIDs`	sample ID, could be a vector of the same length as fileNames so that different sample IDs are used for different input files.
`fileNames`	one or more names of the output files of genoCN or PennCNV. If it is NULL, only plot the LRR and BAF.
`types`	should be the same length as fileNames, indicating the type of output, currently only support "genoCN" and "pennCNV"
`CNA`	whether this is a copy number aberration study.
`main`	title of the plot
`LRR.ylim`	Range of y-axis for LRR plot
`cex`	the amount by which plotting text and symbols should be magnified relative to the default
`plot.lowess`	to plot the lowess curve for LRR or not

Author(s)

Wei Sun

See Also

`genoCNA`, `genoCNV`

Examples

```
data(snpData)
data(snpInfo)

dim(snpData)
dim(snpInfo)

snpData[1:2,]
snpInfo[1:2,]

snpInfo[c(1001,1100,10001,10200),]

plotCN(pos=snpInfo$Position, LRR=snpData$LRR, BAF=snpData$BAF,
main = "simulated data on Chr22")
```

---

snpData *Simulated LRR and BAF data for 17,348 SNPs on chromosome 22.*

Description

Simulated LRR and BAF data for 17,348 SNPs on chromosome 22. Two CNVs are simulated. One is from the 1001-th probe to the 1100-th probe, with copy number 1. The other one is from the 10,001-th probe to the 10,200-th probe, with copy number 3.

Usage

```
data(snpData)
```

Format

A data frame with 17,348 observations on the following 3 variables.

`Name` a character vector of probe Names

`LRR` a numeric vector of LRR values of each probe

`BAF` a numeric vector of BAF of each probe

Examples

```
data(snpData)
data(snpInfo)

dim(snpData)
dim(snpInfo)

snpData[1:2,]
snpInfo[1:2,]

plotCN(pos=snpInfo$Position, LRR=snpData$LRR, BAF=snpData$BAF,
main = "simulated data on Chr22")
```

snpInfo *Information of 17,348 SNPs on chromosome 22.*

Description

Information of 17,348 SNPs on chromosome 22.

Usage

```
data(snpInfo)
```

Format

A data frame with 17348 observations on the following 4 variables.

`Name` a character vector of probe Names

`Chr` a character vector of chromosomes of each probe

`Position` a numeric vector of genomic position of each probe

`PFB` a numeric vector of population frequency of B allele for each probe. For copy number only probes, PFB=2.0

Examples

```
data(snpData)
data(snpInfo)

dim(snpData)
dim(snpInfo)

snpData[1:2,]
snpInfo[1:2,]

plotCN(pos=snpInfo$Position, LRR=snpData$LRR, BAF=snpData$BAF,
main = "simulated data on Chr22")
```

Index