Package 'expands'

March 18, 2018

Type Package

Title Expanding Ploidy and Allele-Frequency on Nested Subpopulations

Version 2.1.1

Date 2018-02-23

Author Noemi Andor

Maintainer Noemi Andor <expands.r@gmail.com>

Description Expanding Ploidy and Allele Frequency on Nested Subpopulations (expands) characterizes coexisting subpopulations in a single tumor sample using copy number and allele frequencies derived from exome- or whole genome sequencing input data (<http://www.ncbi.nlm.nih.gov/pubmed/24177718>). The model detects coexisting genotypes by leveraging run-specific tradeoffs between depth of coverage and breadth of coverage. This package predicts the number of clonal expansions, the size of the resulting subpopulations in the tumor bulk, the mutations specific to each subpopulation, tumor purity and phylogeny. The main function runExPANdS() provides the complete functionality needed to predict coexisting subpopulations from single nucleotide variations (SNVs) and associated copy numbers. The robustness of subpopulation predictions increases with the number of mutations provided. It is recommended that at least 200 mutations are used as input to obtain stable results. Updates in version 2.1 include: (i) new parameter ploidy in runExPANdS.R allows specification of non-diploid background ploidies (e.g. for near-triploid cell lines); (ii) parallel computing option is available. Further documentation and FAQ available at <http://dna-discovery.stanford.edu/software/expands>.

License GPL-2

URL http://dna-discovery.stanford.edu/software/expands,
https://github.com/noemiandor/expands,
https://groups.google.com/d/forum/expands

Depends R (>= 2.10)

Imports flexclust, plyr, RColorBrewer, gplots, NbClust, moments (>= 0.13), rJava (>= 0.5-0), flexmix (>= 2.3), matlab (>= 0.8.9), ape (>= 3.2), commonsMath (>= 1.1), parallel

Suggests phylobase (>= 0.6.8)

SystemRequirements Java (>= 5.0)

NeedsCompilation no

Repository CRAN

Date/Publication 2018-03-18 19:30:09 UTC

RoxygenNote 6.0.1

R topics documented:

`assignMutations` *Mutation Assignment*

Description

Assigns mutations to previously predicted subpopulations.

Usage

```
assignMutations(dm, finalSPs, max_PM=6, cnvSPs=NULL, ploidy = 2, verbose = T)
```

Arguments

`dm`	Matrix in which each row corresponds to a mutation. Has to contain at least the following column names: **chr** - the chromosome on which each mutation is located; **startpos** - the genomic position of each mutation; **AF_Tumor** - the allele-frequency of each mutation; **PN_B** - the count of the B-allele in normal (non-tumor) cells (binary variable: 1 if the mutation is a germline variant, 0 if somatic).

`finalSPs` Matrix in which each row corresponds to a subpopulation, as calculated by `clusterCellFrequencies`.

`max_PM` Upper threshold for the number of amplicons per mutated cell. See also `cellfrequency_pdf`.

`cnvSPs` Matrix in which each row corresponds to a subpopulation, as calculated by `clusterCellFrequencies`. If not set, finalSPs will be used to assign CNVs as well as SNVs.

`ploidy` The background ploidy of the sequenced sample (default: 2). Changing the value of this parameter is not recommended. Dealing with cell lines or tumor biopsies of very high (>=0.95) tumor purity is a necessary but not sufficient condition to change the value of this parameter.

`verbose` Give a more verbose output.

Details

Each mutated locus l is assigned to the subpopulation C, whose size f_C can best explain the allele frequency (AF) and copy number (CN) observed at l. Four alternative cell frequency probabilities, $P_x(f_C)$, are calculated for the SNV at locus l, with x denoting one of the four alternative evolutionary scenarios (see also `cellfrequency_pdf`).
The SNV is assigned to subpopulation:
$C := argmax_C(P_s(f_C), P_p(f_C), P_c(f_C), P_i(f_C))$ (see `cellfrequency_pdf`).

The mutated loci assigned to each subpopulation cluster represent the genetic profile of each predicted subpopulation.
The assignment between subpopulation C and locus l only implies that the SNV at l has been first propagated during the clonal expansion that gave rise to C. So SNVs present in C may not be exclusive to C but may also be present in subpopulations smaller than C. Whether or not this is the case can sometimes be inferred from the phylogenetic structure of the subpopulation composition. See also `buildPhylo`.

Value

A list with two fields:

`dm` The input matrix with seven additional columns:
SP - subpopulation to which the point mutation has been assigned;
PM_B - count of the B-allele at the mutated genomic locus, in the assigned subpopulation (SP).
PM - total count of all alleles, in the assigned subpopulation (SP).
SP_cnv - if the point mutation lies within an amplified or deleted region: the subpopulation to which the copy number variation has been assigned. This entry has the same value as SP if and only if: i) the SNV and the CNV were propagated during the same clonal expansion or ii) the SNV lies within a copy neutral region.
PM_B_cnv - count of the B-allele, in the CNV harboring subpopulation (SP_cnv).
PM_cnv - total count of all alleles, in the CNV harboring subpopulation (SP_cnv).
%maxP - confidence of the assigned SP/SP_cnv scenario.

`finalSPs` The input matrix of subpopulations with column **nMutations** updated according to the total number of mutations assigned to each subpopulation.

Author(s)

Noemi Andor

References

Li, B. & Li, J. Z (2014). A general framework for analyzing tumor subclonality using SNP array and DNA sequencing data. Genome Biol.

See Also

`clusterCellFrequencies`

`assignQuantityToMutation`

Quantity assignment (copy number) to mutations

Description

Assigns a quantity to each mutated locus. Currently, the only assignable quantity is the average copy number (among all cells) of the locus in which the mutation is embedded.

Usage

```
assignQuantityToMutation(dm, cbs, quantityColumnLabel="CN_Estimate", verbose = T)
```

Arguments

`dm` Matrix in which each row corresponds to a mutation. Has to contain at least the following column names:
chr - the chromosome on which each mutation is located;
startpos - the genomic position of each mutation.

`cbs` Matrix in which each row corresponds to a copy number segment as calculated by a circular binary segmentation algorithm. Has to contain at least the following column names:
chr - chromosome;
startpos - the first genomic position of a copy number segment;
endpos - the last genomic position of a copy number segment;
CN_Estimate - the copy number estimated for each segment.

`quantityColumnLabel` The name of the new column. Valid options are: `FPKM, CN_Estimate.`

`verbose` Give a more verbose output.

Value

dm	The input matrix with three additional columns: **quantityID** - the ID of the assigned quantity; **quantityColumnLabel** - the quantity.

Author(s)

Noemi Andor

Examples

```
data(cbs)
data(snv)
dm=assignQuantityToMutation(snv,cbs,quantityColumnLabel="CN_Estimate")
```

---

assignQuantityToSP *Quantity assignment (copy number) to subpopulations*

Description

Assigns quantities to predicted subpopulations. Currently, the only assignable quantity are subpopulation specific copy number states for the input genome segments.

Usage

```
assignQuantityToSP(cbs, dm, C=list(sp=c("SP","SP_cnv"), pm=c("PM","PM_cnv")), e=1, v=T)
```

Arguments

cbs	Matrix in which each row corresponds to a copy number fragment as computed by a circular binary segmentation algorithm. Has to contain at least the following column names: **chr** - chromosome; **startpos** - the first genomic position of a copy number segment; **endpos** - the last genomic position of a copy number segment; **CN_Estimate** - the copy number estimated for each segment (weighted average value across all subpopulations in the sample).
dm	Matrix in which each row corresponds to a mutation. Has to contain at least the following column names: **chr** - chromosome on which each point mutation is located; **startpos** - genomic position of each mutation; **SP** - subpopulation to which the point mutation has been assigned; **SP_cnv** - subpopulation with a copy number variation within the same genomic segment in which SP has a point mutation; **PM** - total count of all alleles in the subpopulation with the point mutation (SP); **PM_cnv** - total count of all alleles in the subpopulation with the copy number variation (SP_cnv).

C	List referencing column names in the mutation matrix, with two fields: **sp** - column names holding subpopulation sizes (typically "SP", "SP_cnv"); **pm** - column names holding the total allele counts assigned for each subpopulation (typically "PM", "PM_cnv").
e	Maximum variance of subpopulation specific copy numbers for a given segment, above which segment will remain unassigned for the corresponding subpopulation. Determines whether or not to assign copy number to a subpopulation, SP_i, for a segment containing multiple SP_i specific copy numbers, at least two of which are distinct.
v	Give a more verbose output.

Value

The input copy number matrix with one additional column for each predicted subpopulation:
SP_xx - where xx is the size of the corresponding subpopulation. Column entries contain the copy number of each segment in SP; Value <NA> indicates that no copy number could be inferred for the segment in this subpopulation (either because the subpopulation had no point mutations/CNVs within the segment, or because it had multiple, ambiguous copy number assignments within the segment).

Author(s)

Noemi Andor

---

buildMultiSamplePhylo *Relations between inter- and intra-sample subpopulations*

Description

Predicts phylogenetic relations between subpopulations from subpopulation specific copy number and point mutation profiles, while including information about sample origin of each subpopulation. This function differs from buildPhylo in that it integrates the subpoulations predicted in multiple, geographically distinct tumor-samples into one common phylogeny and in that it includes point mutations in addition to copy number variations to infer inter-sample phylogenetic relations.

Usage

```
buildMultiSamplePhylo(samGr, out, treeAlgorithm="bionjs", e=0, plotF=1, spRes=1, v=F)
```

Arguments

samGr	List with three fields: **cbs** - Input of runExPANdS: matrix in which each row corresponds to a copy number segment. CBS is typically the output of a circular binary segmentation algorithm. Columns in CBS must be labeled and must include chr, startpos, endpos and CN_Estimate (see cbs). **Do not use the output of runExPANdS here.**

sps - Output of runExPANdS. Matrix in which each row corresponds to a somatic mutation. Columns must include: chr - the chromosome on which each mutation is located; startpos - the genomic position of each mutation; SP - the subpopulation to which the mutation has been assigned; PM - the total count of all alleles at the mutated genomic locus, in the assigned subpopulation; PM_B - the count of the B-allele at the mutated genomic locus, in the assigned subpopulation; CN_Estimate - the average copy number (among all cells) of the locus in which the mutation is embedded (see also assignQuantityToMutation).
labels - Label denoting sample origin of each subpopulation matrix. Entry is mandatory for each geographical sample.

out	Prefix of file to which multi-sample phylogeny will be saved.
treeAlgorithm	Neighbor joining algorithm used for phylogeny reconstruction (from library ape). Options: bionjs (default), njs.
e	Input parameter "e" for called function: assignQuantityToSP.
plotF	Option for displaying the phylogenetic tree (0 - no display; 1 - display).
spRes	Option on whether or not to ignore the subpopulations calculated for each sample and instead treat every geographical tumor-sample as one single tumor-metapopulation (Default value: 1 - subpopulation resolution; 0 - metapopulation resolution).
v	Give a more verbose output.

Details

This function does not change the subpopulation membership of SNVs. Instead it reconstructs phylogenetic relationships between subpopulations using neighbor-joining algorithms provided by R-package 'ape'. Pairwise distances between subpopulations i and j are calculated as:
$d_{ij} := (cnv_{i=j} + snv_{i=j})/(cnv_{ij} + snv_{ij})$, where $cnv_{i=j}$ is the number of copy number segments for which subpopulations i and j have the same copy number; $snv_{i=j}$ is the number of point mutations for which subpopulations i and j have the same mutation status and cnv_{ij}, snv_{ij} are the total number of copy number segments and mutations respectively, for which both subpopulations have available information. Subpopulations with insufficient copy number and point mutations information are excluded from phylogeny.

Value

An object of class "phylo" (library ape).

Author(s)

Noemi Andor

See Also

buildPhylo

buildPhylo *Relations between subpopulations*

Description

Predicts phylogenetic relations between subpopulations from subpopulation specific copy number profiles.

Usage

```
buildPhylo(sp_cbs, outF, treeAlgorithm="bionjs", dm=NA, add="Germline", verbose = T)
```

Arguments

sp_cbs	Subpopulation specific copy number matrix in which each row corresponds to a copy number segment. Has to contain at least one column for each predicted subpopulation. Subpopulation column names must be labeled SP_xx, where xx is the size of the corresponding subpopulation. Input parameter sp_cbs can be obtained by calling assignQuantityToSP.
outF	Prefix of file to which phylogeny will be saved.
treeAlgorithm	Neighbor joining algorithm used for phylogeny reconstruction (from library ape). Options: bionjs (default), njs.
dm	Optional matrix in which each row corresponds to a mutation. Only mutations located on autosomes should be included. Columns in dm must be labeled and must include: **SP** - subpopulation to which the point mutation has been assigned. **SP_cnv** - subpopulation to which the CNV (overlapping with the point mutation) has been assigned (if an CNV is present). **chr** - chromosome on which each point mutation is located; **startpos** - genomic position of each point mutation; **PM** - total count of all alleles at the mutated genomic locus, in the assigned subpopulation. **PM_B** - count of the B-allele at the mutated genomic locus, in the assigned subpopulation. If dm is available, an attempt will be made to assign every mutation to >1 subpopulation according to the inferred phylogenetic relations between subpopulations.
add	Artificial subpopulation to be included in phylogeny (options: 'Germline','Consensus', NULL).
verbose	Give a more verbose output.

Details

Reconstructs phylogenetic relationships between subpopulations using neighbor-joining algorithms provided by R-package 'ape'. Pairwise distances between subpopulations are calculated as the number of copy number segments for which both subpopulations have the same copy number,

divided by the total number of copy number segments for which both subpopulations have available copy number information. Subpopulations with insufficient copy number information are excluded from phylogeny.

Value

List with two fields:

`tree`	An object of class "phylo" (library ape).
`dm`	The input matrix with each row representing a point mutation and additional columns: **SP_xx** - where xx is the size of the corresponding subpopulation. Column entries contain a binary indicator of whether or not the point mutation in this row is present in SP_xx.

Author(s)

Noemi Andor

See Also

`assignQuantityToSP`

---

cbs *Matrix of copy number fragments*

Description

Copy number segments as obtained by circular binary segmentation. Data is derived from a Glioblastoma tumor (TCGA-06-0152-01).

Usage

```
data(cbs)
```

Format

Numeric matrix with 120 rows (one per copy-number segment) and 4 columns:
chr - the chromosome
startpos - genomic position at which copy-number segment starts.
endpos - genomic position at which copy-number segment ends.
CN_Estimate - average copy-number of the segment among all cells.

Source

Data derived from The Cancer Genome Atlas (TCGA).

`cellfrequency_pdf` *Computes the probability distribution of cellular frequencies for a single mutation.*

Description

Calculates P - the probability density distribution of cellular frequencies for one single point mutation or CNV. For each cell-frequency f, the value of $P(f)$ reflects the probability that the mutation is present in a fraction f of cells.

Usage

```
cellfrequency_pdf(af, cnv, pnb, freq, max_PM=6, ploidy = 2)
```

Arguments

`af`	The allelic frequency at which the point mutation has been observed.
`cnv`	The average copy number of the locus in which the mutation is embedded.
`pnb`	The count of the B-allele in normal cells (binary variable: 1 if the mutation is a germline variant, 0 if somatic). B-alleles that have >1 copy in normal cells are not modeled.
`freq`	Vector of cellular frequencies at which the probabilities will be calculated.
`max_PM`	Upper threshold for the number of amplicons per mutated cell. max_PM is the maximum number of amplicons above which solutions are rejected in the cell-frequency estimation step described below, i.e. PM <= max_PM. The choice of max_PMshould depend on genomic depth of coverage and on the fraction of the genome sequenced: the higher the quality and abundance of data, the higher max_PM.
`ploidy`	The background ploidy of the sequenced sample (default: 2). Changing the value of this parameter is not recommended. Dealing with cell lines or tumor biopsies of very high (>=0.95) tumor purity is a necessary but not sufficient condition to change the value of this parameter.

Details

We consider two types of molecular mechanisms that convert a locus into its mutated state: copy number variation (CNV) inducing events and single nucleotide variation (SNV) inducing events. We assume that a normal state is defined by a total allele count of two and B allele count below two, whereas a mutated state has an increased fraction of B alleles. The conditions defining these states for each locus l are as follows:
i) $PM_B, PN_B, PM, PN \in N$; **ii)** $PM_B \geq 1; PN_B \leq 1; PN = 2$; **iii)** $\frac{PM_B}{PM} \geq \frac{PN_B}{PN}$.

PM_B and PN_B denote the count of the B allele in each cell type: mutated cells and normal cells, respectively. The value of PN_B is one if l has a germline variant, zero otherwise. PM, PN are the total allele count of mutated cells and normal cells. PM is required to be between one and

max_PM, that is, we exclude solutions for which the maximum number of amplicons per cell exceeds the user defined value of max_PM.
The function returns the probability distribution, $P(f)$, that the mutation at locus l is present in a fraction f of cells, where $f \in [0, 1]$.

Four alternative cell frequency probability distribution scenarios, $P(f)$, can be obtained for each allele-frequency + copy number pair (AF, CN). For each scenario, model starts with a germline population that will be the root of all other modeled subpopulations. First subpopulation (f_{cnv}) modeled to evolve from the germline population is always the one carrying a CNV:
$pm * f_{cnv} + PN * (1 - f_{cnv}) = CN$, where pm is the total allele count of f_{cnv}.
A subsequent subpopulation (f_{snv}) is always defined by an SNV and is modeled in relation to f_{cnv}, either as:
1. $P_s(f)$ - its sibling: $PM_B * f_{snv} + PN_B * (1 - f_{snv}) = AF * CN$, where $f_{snv} + f_{cnv} <= 1; PM_B <= 2$.
2. $P_p(f)$ - its parent: $PM_B * (f_{snv} - f_{cnv}) + pm_B * f_{cnv} + PN_B * (1 - f_{snv}) = AF * CN$, where $f_{snv} > f_{cnv}; PM_B <= 2$ and pm_B is the B-allele count of f_{cnv}.
3. $P_c(f)$ - its child: $PM_B * f_{snv} + PN_B * (1 - f_{snv}) = AF * CN$, where $f_{snv} < f_{cnv}; PM_B <= pm$.
4. $P_i(f)$ - itself: $PM_B * f + PN_B * (1 - f) = AF * CN$, where $f = f_{snv} = f_{cnv}; PM_B <= pm$.

Under 1), SNV and CNV are completely independent as they are never co-propagated during the same clonal expansion. Under 2) and 3), SNV and CNV are partially dependent, yet present in two distinct subpopulations. Under 4), both the SNV and an CNV at l were propagated during the same clonal expansion.

Value

List with four components:

`p`	The probability that the point mutation/CNV is present in a fraction f of cells, for each input frequency f in parameter $freq$.
`bestF`	The cellular frequency that best explains the observed allele frequency and/or copy number.

Author(s)

Noemi Andor

References

Noemi Andor, Julie Harness, Sabine Mueller, Hans Werner Mewes and Claudia Petritsch. (2013) ExPANdS: Expanding Ploidy and Allele Frequency on Nested Subpopulations. Bioinformatics.

Examples

```
freq=seq(0.1,1.0,by=0.01);
cfd=cellfrequency_pdf(af=0.26,cnv=1.95,pnb=0,freq=freq, max_PM=6)
plot(freq,cfd$p,type="l",xlab="f",ylab="P(f)");
```

`clusterCellFrequencies` *Clustering of cellular frequency probability distributions*

Description

Calculates overrepresented cell frequencies using a two-step approach. Based on the assumption that passenger mutations occur within a cell prior to the driver event that initiates the expansion, each clonal expansion should be marked by multiple mutations. Thus mutations and copy number variations that took place in a cell prior to a clonal expansion should be present in a similar fraction of cells and leave a similar "frequency-trace" during their propagation.

Usage

```
clusterCellFrequencies(densities, p, nrep=30, min_CF=0.1, verbose = T)
```

Arguments

`densities`	Matrix as obtained by `computeCellFrequencyDistributions`. Each row corresponds to a mutation and each column corresponds to a cellular frequency. Each value $densities[i, j]$ represents the probability that mutation i is present in a fraction f of cells, where f is given by: $colnames(densities[, j])$.
`p`	Precision with which subpopulation size is predicted, a small value reflects a high resolution and can lead to a higher number of predicted subpopulations.
`nrep`	Positive integer indicating the number of algorithm repetitions (default: 30).
`min_CF`	Lower threshold for the prevalence of a mutated cell (default: 0.1).
`verbose`	Give a more verbose output.

Details

In the first step, mutations with similar cellular frequencies are grouped together by hierarchical cluster analysis of the probability distributions using the Kullback-Leibler divergence as a distance measure. The cell frequency at each cluster-maxima denotes the size of the subpopulation that harbors the clustered mutations. In the second step, each cluster is extended by members with similar distributions in an interval around the cluster-maxima.

Value

`SPs`	Matrix of predicted subpopulations. Each row corresponds to a subpopulation and each column contains information about that subpopulation, such as the size in the sequenced tumor bulk (column **Mean Weighted**) and the noise score at which the subpopulation has been detected (column **score**: lower values ~ higher subpopulation detection confidence).

Author(s)

Noemi Andor

References

Noemi Andor, Julie Harness, Sabine Mueller, Hans Werner Mewes and Claudia Petritsch. (2013) ExPANdS: Expanding Ploidy and Allele Frequency on Nested Subpopulations. Bioinformatics.

---

computeCellFrequencyDistributions

Gathering of cell frequency probability distributions

Description

Computes the probability distributions of cell frequencies, by calling `cellfrequency_pdf` for each mutation separately.

Usage

```
computeCellFrequencyDistributions(dm, max_PM=6, p, min_CF=0.1, ploidy = 2, nc = 1, v = T)
```

Arguments

`dm`	Matrix in which each row corresponds to a mutation. Has to contain at least the following column names: **chr** - the chromosome on which each mutation is located; **startpos** - the position of each mutation; **AF_Tumor** - the allele-frequency of each mutation; **PN_B** - the count of the B-allele in normal cells (binary variable: 1 if the mutation is a germline variant, 0 if somatic).
`max_PM`	Upper threshold for the number of amplicons per mutated cell (default: 6). See also `cellfrequency_pdf`.
`p`	Precision with which subpopulation size is predicted, a small value reflects a high resolution and can lead to a higher number of predicted subpopulations.
`min_CF`	Lower boundary for the prevalence of a mutated cell (default: 0.1).
`ploidy`	The background ploidy of the sequenced sample (default: 2). Changing the value of this parameter is not recommended. Dealing with cell lines or tumor biopsies of very high (>=0.95) tumor purity is a necessary but not sufficient condition to change the value of this parameter.
`nc`	The number of nodes to be forked to run R in parallel.
`v`	Give a more verbose output.

Value

List with three fields:

`freq`	The cellular frequencies for which probabilities are computed.
`densities`	Matrix in which each row corresponds to a point mutation and each column corresponds to a cellular frequency. Each value $densities[i, j]$ represents the probability that mutation i is present in a fraction $freq[j]$ of cells.
`dm`	The input matrix with column f updated according to the cellular frequency that best explains the observed allele frequency and copy number.

Author(s)

Noemi Andor

`gatherEXPANDSoutput` *Reading EXPANDS' output files*

Description

Reads EXPANDS output files from user-specified input directory.

Usage

```
gatherEXPANDSoutput(outDirEXPANDS,regex="")
```

Arguments

`outDirEXPANDS`	Absolute path to input directory in which EXPANDS results are stored, as generated by `runExPANdS`.
`regex`	Pattern that path has to match in order to be read.

Value

Two-level nested list. The outer level contains one entry per each output of `runExPANdS`. Each entry constitutes an inner list with fields:
snv - The assignment of each SNV to a subpopulation.
cbs - The copy number of each genomic segment in each subpopulation.
spstats - Matrix of predicted subpopulations.
tree - The inferred phylogenetic relationships between subpopulations as an object of class "phylo4" (library phylobase).
treeApe - The inferred phylogenetic relationships between subpopulations as an object of class "phylo" (library ape).

Author(s)

Noemi Andor

See Also

runExPANdS

---

plotSPs *Subpopulation Visualization*

Description

Plots coexistent subpopulations determined by ExPANdS.

Usage

```
plotSPs(dm, sampleID=NA, cex=0.5, legend="CN_Estimate", orderBy="chr", rawAF=F)
```

Arguments

dm	Matrix in which each row corresponds to a point mutation (for example, the matrix output of assignMutations). Has to contain at least the following column names: **chr** - the chromosome on which each mutation is located; **startpos** - the genomic position of each mutation; **AF_Tumor** - the allele-frequency of each mutation; **CN_Estimate** - the absolute copy number estimated for each segment; **PN_B** - the count of the B-allele in normal cells (binary variable: 1 if the mutation is a germline variant, 0 if somatic); **SP** - the subpopulation to which each point mutation has been assigned (as fraction of cells in the tumor bulk); **%maxP** - the confidence with which the mutation has been assigned to the corresponding subpopulation; **SP_cnv** - the subpopulation to which the CNV has been assigned; **PM** - the total count of all alleles at the mutated genomic locus, in subpopulation SP. **PM_cnv** - the total count of all alleles at the mutated genomic locus in subpopulation SP_cnv. **PM_B** - the count of the mutated allele in subpopulation SP. **PM_B_cnv** - the count of the mutated allele in subpopulation SP_cnv.
sampleID	The name of the sample in which the mutations have been detected.
cex	The amount by which plotting text and symbols should be magnified relative to the default. See also help(par).
legend	Allele frequencies and subpopulation specific copy numbers are colored based on the chromosome on which the mutation is located (option: 'chr') or based on the average copy number of the locus in the sample (option: 'CN_Estimate').
orderBy	Loci within a subpopulation are sorted by genomic location (option 'chr') or by the confidence with which they have been assigned to the subpopulation (option '%maxP').
rawAF	Specifies whether the allele frequency of SNVs should be adjusted relative to the assigned subpopulation (options: true, false).

Value

For each point mutation (x-axis) the function displays:
- the size of the subpopulation to which the mutation has been assigned (squares). Each square is colored based on the confidence with which the mutation has been assigned to the corresponding subpopulation (black - highest, white - lowest).
- the total count of all alleles at the mutated genomic locus in that subpopulation (dots).
- only for loci with an CNV and an SNV each in distinct subpopulations: the total count of all alleles at the mutated genomic locus in the subpopulation which harbors the CNV (crosses).
- the allele frequency of the mutation (stars - somatic mutations, triangles - loss of heterozygosity).

Author(s)

Noemi Andor

---

roi *Regions of interest*

Description

For internal use only. Default regional boundary for mutations included during clustering, comprising ca. 468 MB centered on the human exome. Relevant if number of input mutations exceeds user defined threshold (often applies to whole genome sequencing data). A saved image of this object is in sysdata.rda.

Format

Numeric matrix in which each row corresponds to a genomic segment.
Columns:
chr - the chromosome of the segment ;
start - the first genomic position of the segment;
end - the last genomic position of the segment.

Source

Data derived from human SureSelectExome_hg19 50MB library kit annotation.

See Also

runExPANdS

runExPANdS *Main Function*

Description

Given a set of mutations, ExPANdS predicts the number of clonal expansions in a tumor, the size of the resulting subpopulations in the tumor bulk and which mutations accumulate in a cell prior to its clonal expansion. Input-parameters SNV and CBS hold the paths to tab-delimited files containing the point mutations and the copy numbers respectively. Alternatively SNV and CBS can be read into the workspace and passed to runExPANdS as numeric matrices. The robustness of the subpopulation predictions by ExPANdS increases with the number of mutations provided. It is recommended that SNV contains at least 200 point mutations to obtain stable results.

Usage

```
runExPANdS(SNV, CBS, maxS=0.7, max_PM=6, min_CF=0.1, p=NA, ploidy=2,
                  nc=1, plotF=2, snvF=NULL, maxN=8000, region=NA, verbose=T)
```

Arguments

SNV	Matrix in which each row corresponds to a point mutation. Only mutations located on autosomes should be included. Columns in SNV must be labeled and must include: **chr** - the chromosome on which each mutation is located; **startpos** - the genomic position of each mutation; **AF_Tumor** - the allele-frequency of each mutation; **PN_B** - count of B-allele in normal cells. A value of 0 indicates that the variant has only been detected in the tumor sample (i.e. somatic mutation). A value of 1 indicates that the variant is also present in the normal (control) sample, albeit at reduced allele frequency (i.e. this is a germline variant, which passed the calling filter due to the presence of an LOH event). Mutations, for which the allele frequency in the tumor sample is lower than the corresponding allele frequency in the normal sample, should not be included.
CBS	Matrix in which each row corresponds to a copy number segment. CBS is typically the output of a circular binary segmentation algorithm. Columns in CBS must be labeled and must include: **chr** - chromosome; **startpos** - the first genomic position of a copy number segment; **endpos** - the last genomic position of a copy number segment; **CN_Estimate** - the absolute copy number estimated for each segment.
maxS	Upper threshold for the noise score of subpopulation detection. Only subpopulations identified at a score below $maxS$ are kept.
max_PM	Upper threshold for the number of amplicons per mutated cell. Increasing the value of this variable is not recommended unless extensive depth and breadth of coverage underlie the measurements of copy numbers and allele frequencies. See also cellfrequency_pdf.

`min_CF` Lower boundary for the cellular prevalence interval of a mutated cell. Mutations for which allele frequency * copy number are below $min_{CellFreq}$, are excluded from further computation. Decreasing the value of this variable is not recommended unless extensive depth and breadth of coverage underlie the measurements of copy numbers and allele frequencies.

`p` Precision with which subpopulation size is predicted, a small value reflects a high resolution and can lead to a higher number of predicted subpopulations.

`plotF` Option for displaying a visual representation of the identified subpopulations (0 - no display; 1 - display subpopulation size; 2 - display subpopulation size and phylogeny).

`snvF` Prefix of file to which predicted subpopulation composition will be saved. Default: the name of the file from which mutations have been read or "out.expands" if input mutations are not handed over as file path.

`maxN` Upper limit for number of point mutations used during clustering. If number of user supplied point mutations exceeds $maxN$, the clustering of cellular frequency distributions will be restricted to point mutations found within $region$.

`region` Regional boundary for mutations included during clustering. Matrix in which each row corresponds to a genomic segment. Columns must include:
chr - the chromosome of the segment;
start - the first genomic position of the segment;
end - the last genomic position of the segment.
Default: SureSelectExome_hg19, comprising ca. 468 MB centered on the human exome. Alternative user supplied regions should also be coding regions, as the selective pressure is higher as compared to non-coding regions.

`ploidy` The background ploidy of the sequenced sample (default: 2). Changing the value of this parameter is not recommended. Dealing with cell lines or tumor biopsies of very high (>=0.95) tumor purity is a necessary but not sufficient condition to change the value of this parameter.

`nc` The number of nodes to be forked to run R in parallel.

`verbose` Give a more verbose output.

Value

List with fields:

`finalSPs` Matrix of predicted subpopulations. Each row corresponds to a subpopulation and each column contains information about that subpopulation, such as the size in the sequenced tumor bulk (column **Mean Weighted**) and the noise score at which the subpopulation has been detected (column **score**).

`dm` Matrix containing the input mutations with at least seven additional columns:
SP - the subpopulation to which the point mutation has been assigned;
SP_cnv - the subpopulation to which the CNV has been assigned (if an CNV exists at this locus);
%maxP - the confidence of mutation assignment.
f - Deprecated. The maximum likelihood cellular prevalence of this point mutation, before it has been assigned to SP. This value is based on the copy number

and allele frequency of the mutation exclusively and is independent of other point mutations. Column SP is less sensitive to noise and considered the more accurate estimation of cellular mutation prevalence.
PM - the total count of all alleles in the subpopulation harboring the point mutation (SP).
PM_B - the count of the B-allele in the subpopulation harboring the point mutation (SP).
PM_cnv - the total count of all alleles in the subpopulation harboring an CNV (SP_cnv).
PM_B_cnv - the count of the B-allele, in the CNV harboring subpopulation (SP_cnv).
If phylogeny reconstruction was successful, matrix includes one additional column for each subpopulation from the phylogeny, indicating whether or not the point mutation is present in the corresponding subpopulation.

`densities` Matrix as obtained by `computeCellFrequencyDistributions`. Each row corresponds to a mutation and each column corresponds to a cellular frequency. Each value $densities[i, j]$ represents the probability that mutation i is present in a fraction f of cells, where f is given by: $colnames(densities[, j])$.

`sp_cbs` Matrix as obtained by `assignQuantityToSP`. Each row corresponds to a copy number segment, e.g. as obtained from a circular binary segmentation algorithm. Includes one additional column for each predicted subpopulation, containing the copy number of each segment in the corresponding subpopulation.

`tree` An object of class "phylo" (library ape) as obtained by `buildPhylo`. Contains the inferred phylogenetic relationships between subpopulations.

Author(s)

Noemi Andor

References

Noemi Andor, Julie Harness, Sabine Mueller, Hans Werner Mewes and Claudia Petritsch. (2013) ExPANdS: Expanding Ploidy and Allele Frequency on Nested Subpopulations. Bioinformatics.

Examples

```
data(snv);
data(cbs);
maxS=2.5;
set.seed(4); idx=sample(1:nrow(snv), 60, replace=FALSE);
#out= runExPANdS(snv[idx,], cbs, maxS);
```

simulation *Simulated heterogeneous samples*

Description

A total of 50 samples with various numbers of subpopulations per sample were simulated at variable noise rates and constant number of 200 mutations per sample.

Usage

```
data(simulation)
```

Format

List with 50 entries - one per simulated sample. Subpopulation composition can be predicted for each sample and the predictions compared to the the simulated entries:
snv - the matrix of simulated point mutations (including ground truth columns SP*, PM*).
cbs - the matrix of simulated copy number segments (including ground truth columns SP*).
spstats - matrix of subpopulation statistics (ground truth).

Examples

```
data(simulation)
snvcols=c("chr", "startpos", "CN_Estimate", "AF_Tumor","PN_B")
cbscols=c("chr", "startpos", "endpos")
sI=1:50;#set to 1:200 to run on entire simulation
#out=runExPANdS(simulation[[1]]$snv[sI,snvcols],simulation[[1]]$cbs[,cbscols],plotF = 0);
#truePhy=buildPhylo(simulation[[1]]$cbs,outF='truePhylo'); ##simulated
#predPhy=buildPhylo(out$sp_cbs,outF='truePhylo'); ##predicted
#par(mfrow=c(1,2))
#plot(truePhy$tree,cex=2,main='simulated')
#plot(predPhy$tree,cex=2,main='predicted')
```

snv *Single Nucleotide Variations*

Description

Somatic mutations and Loss of Heterozygosity (LOH) of a Glioblastoma tumor (TCGA-06-0152-01).

Usage

```
data(snv)
```

Format

Numeric matrix with 773 rows (one per mutation) and 7 columns:
chr - the chromosome
startpos - genomic position
endpos - same as above
REF - ASCII code of the reference nucleotide (in hg18/hg19)
ALT - ASCII code of the B-allele nucleotide
AF_Tumor - allele frequency of B-allele
PN_B - count of B-allele in normal cells. A value of 0 indicates that the mutation has only been detected in the tumor sample (i.e. somatic mutations). A value of 1 indicates that the variant is also present in the normal (control) sample, albeit at reduced allele frequency (i.e. this is a germline variant, which passed the calling filter due to the presence of an LOH event). Other mutations should not be included.

Source

Data derived from The Cancer Genome Atlas (TCGA).

Index