Package ‘RNMF’

June 21, 2021

Type Package

Title An optimized Non-Negative Matrix Factorization based on R in cancer research

Version 1.0.9

Date 2021-06-21

Author Wen Luo, Zhenzhang Li

Maintainer Wen Luo <luowengjs2018@foxmail.com>

Description We present an optimized non-Negative Matrix Factorization (NMF) based on R to extract the mutational signature in cancer research. A key model of cumulative contribution abundance (CCA) was designed to highlight the association between genes and mutational signatures.

Depends R (>= 3.6.0)

License GPL (>= 2)

Imports data.table, GenomeInfoDb, BSgenome, BSgenome.Hsapiens.UCSC.hg19, BSgenome.Hsapiens.UCSC.hg38, Biostrings, GenomicRanges, MCMCpack, parallel, reshape, ggplot2, pheatmap, Deducer

Encoding UTF-8

LazyData true

URL https://github.com/zhenzhang-li/RNMF

BugReports https://github.com/zhenzhang-li/RNMF/issues

Date/Publication 2021-06-21 18:27:06

R topics documented:

Index

SigsInput *Convert the mutation dataset into input datas for the mutational signature*

Description

Convert the mutation dataset into input datas for the mutational signature. The format of the input dataset contains MAF and VCF. We used classifications for each type of mutation from PCAWG project. For SBSs, the primary classification comprised 96 classes (available at https://cancer.sanger.ac.uk/cosmic/signatures/SBS) constituted by the 6 base substitutions C>A, C>G, C>T, T>A, T>C and T>G (in which the mutated base is represented by the pyrimidine of the base pair), plus the flanking 5′ and 3′ bases. In some analyses, two flanking bases 5′ and 3′ to the mutated base were considered (producing 1,536 classes) or mutations within transcribed genome regions were selected and classified according to whether the mutated pyrimidine fell on the transcribed or untranscribed strand (producing 192 classes). We also derived a classification for DBSs (78 classes; available at https://cancer.sanger.ac.uk/cosmic/signatures/DBS). Indels were classified as deletions or insertions and—when of a single base—as C or T, and according to the length of the mononucleotide repeat tract in which they occurred. Longer indels were classified as occurring at repeats or with overlapping microhomology at deletion boundaries, and according to the size of indel, repeat and microhomology (83 classes; available at https://cancer.sanger.ac.uk/cosmic/signatures/ID).

Usage

SigsInput(file = NULL, Filetype = c('MAF','VCF'), AnalCOSMICSigType = c('SBS','DBS','ID'), genome.build = c("Ch37","Ch38"), sample.id = 'Tumor_Sample_Barcode', chr = 'Chromosome', pos = 'Start_position', ref = 'Reference_Allele', alt = 'Tumor_Seq_Allele2', ID.mc.cores = 1, ID.row83 = TRUE)

Arguments

- file - A mutation file obtained by annotation of mutation locations by Oncotator or other annotation software.
- Filetype - The format of the input dataset, contains MAF and VCF.

  1) File with 'MAF' form was generated by Oncotator,

  2) and File with 'VCF' form must have at least five columns with the colnames of "CHROM POS REF ALT SAMPLENAME",

  for SBS as: 1 1432333 T C T1,

for DBS as: 1 1432333 TA CG T1,
for ID as: 1 1432333 T - T1.

- AnalCOSMICSigType - Type of mutation based on PCAWG project, canbe SBS, DBS or ID.
- genome.build - The version of the reference genome used in the mutation set.
- sample.id - Column name in the mutation file corresponding to the Sample ID.
- chr - Column name in the mutation file corresponding to the chromosome.
- pos - Column name in the mutation file corresponding to the mutation position.
- ref - Column name in the mutation file corresponding to the reference base.
- alt - Column name in the mutation file corresponding to the alternate base.
- ID.mc.cores - The number of cores to use in ID type, i.e. at most how many child processes will be run simultaneously. Must be exactly 1 on Windows (which uses the master process).
- ID.row83 - Whether to keep 83 properties for ID type.

Author(s)

Wen Luo (design, implementation), Zhenzhang Li (design, testing)

Examples

SigsInput(file = 'Oncotator.HK100.maf', Filetype = 'MAF', AnalCOSMICSigType = 'SBS', genome.build = "Ch37", sample.id = 'Tumor_Sample_Barcode', chr = 'Chromosome', pos = 'Start_position', ref = 'Reference_Allele', alt = 'Tumor_Seq_Allele2')

denovoNMF *Non-Negative Matrix Factorization in cancer research*

Description

We present an optimized non-Negative Matrix Factorisation (NMF) based on R to extract the mutational signature in cancer research.

Usage

denovoNMF(originalGenomes, sampleNames, subtypes, kmin = 1, kmax = 10, AnalCOSMICSigType = 'SBS', steptol = 10^-9, totalIterations = 20, spacetime = 100, nlm = FALSE, mc.cores = 1)

Arguments

- originalGenomes - a file, data matrix for NMF, [m x n, m stands for feature counts and n stands for sample counts].

- sampleNames - a list file of sample names , data matrix for NMF, [n x 1, n stands for sample counts].
- subtypes - a list of types of mutation, data matrix for NMF, [m x 1, m stands for feature counts].
- kmin - Minimum classification of mutational signatures.
- kmax - Maximum classification of mutational signatures.
- AnalCOSMICSigType - Type of mutation based on PCAWG project, can be SBS, DBS or ID.
- steptol - A positive scalar providing the minimum allowable relative step length.
- totalIterations - The iterations number, must be at least 5.
- spacetime - The length of definite solution space of NMF, must be at least 20.
- nlm - Straighten the result to calculate the optimal solution. The default is false.
- mc.cores - The number of cores to use, i.e. at most how many child processes will be running simultaneously. Must be exactly 1 on Windows (which uses the master process).

Author(s)

Wen Luo (design, implementation), Zhenzhang Li (implementation, testing)

References

Alexandrov L B, Nik-Zainal S, Wedge D C, et al. Deciphering signatures of mutational processes operative in human cancer[J]. Cell reports, 2013, 3(1): 246-259.

Alexandrov L B, Kim J, Haradhvala N J, et al. The repertoire of mutational signatures in human cancer[J]. Nature, 2020, 578(7793): 94-101.

Examples

```
denovoNMF('originalGenomes', 'sampleNames', 'subtypes', kmin = 1, kmax = 10,
AnalCOSMICSigType = 'SBS', steptol = 10^-9, totalIterations = 20, spacetime =
100, nlm = TRUE, mc.cores = 20)
```

InverseNMF *Determine the contribution of known mutational processes*

Description

We present an optimized tool based on R to determine the weights of each mutational signature contributing to an individual tumor sample.

Usage

InverseNMF(originalGenomes, sampleNames, subtypes, AnalCOSMICSigType = c('SBS','DBS','ID'), genome.build = c("Ch37","Ch38"), cutoff = 0.06, SBS.version = c("V2","V3"), Pmatrix = NULL, steptol = 10^-10, plot = TRUE)

Arguments

- originalGenomes - a file, data matrix for NMF, [m x n, m stands for feature counts and n stands for sample counts].
- sampleNames - a list file of sample names , data matrix for NMF, [n x 1, n stands for sample counts].
- subtypes - a list of types of mutation, data matrix for NMF, [m x 1, m stands for feature counts].
- AnalCOSMICSigType - Type of mutation based on PCAWG project, canbe SBS, DBS or ID.
- genome.build - The version of the reference genome used in the mutation set.
- cutoff - A threshold between 0 and 1, and was chosen to correct the weight that are most likely caused by errors.
- SBS.version - Mutational signatures version 3 and version 2 signatures in COSMIC.
- Pmatrix - A mutational signature matrix file P with a format like the output result of RNMF software.
- steptol - A positive scalar providing the minimum allowable relative step length.
- plot - Whether to plot the image.

Author(s)

Wen Luo (design, implementation), Zhenzhang Li (implementation, testing)

Examples

InverseNMF('originalGenomes', 'sampleNames', 'subtypes', AnalCOSMICSigType = 'SBS', genome.build = "Ch37", cutoff = 0.06, SBS.version = "V2", Pmatrix = NULL, steptol = 10^-10, plot = TRUE)

cumulativeCA *Cumulative contribution abundance of genes*

Description

We present a tool to calculate the cumulative contribution abundance of genes in cancer research.

Usage

cumulativeCA(file = NULL, Filetype = c('MAF','VCF'), Pfile = NULL, Sfile = NULL, sample.id = 'Tumor_Sample_Barcode', chr = 'Chromosome', pos = 'Start_position', ref = 'Reference_Allele', alt = 'Tumor_Seq_Allele2', Hugo = 'Hugo_Symbol', AnalCOSMICSigType = c('SBS','DBS','ID'), genome.build = c("Ch37","Ch38"), groupFile = NULL, geneListSortFile = NULL, ID.mc.cores = 2, ID.row83 = TRUE, plot = TRUE)

Arguments

- file - A mutation file obtained by annotation of mutation locations by Oncotator or other annotation software.
- Filetype - The format of the input dataset, contains MAF and VCF.
  1) File with 'MAF' form was generated by Oncotator,
  2) and File with 'VCF' form must have at least five columns with the colnames of "CHROM POS REF ALT SAMPLENAME",
     for SBS as: 1 1432333 T C T1,
     for DBS as: 1 1432333 TA CG T1,
     for ID as: 1 1432333 T - T1.
- Pfile - A mutational signature matrix file P with a format like the output result of RNMF software.
- Sfile - A abundance fractions matrix S with a format like the output result of RNMF software.
- sample.id - Column name in the mutation file corresponding to the Sample ID.
- chr - Column name in the mutation file corresponding to the chromosome.
- pos - Column name in the mutation file corresponding to the mutation position.
- ref - Column name in the mutation file corresponding to the reference base.
- alt - Column name in the mutation file corresponding to the alternate base.
- Hugo - Column name in the mutation file corresponding to the gene.
- AnalCOSMICSigType - Type of mutation based on PCAWG project, canbe SBS, DBS or ID.
- genome.build - The version of the reference genome used in the mutation set.
- groupFile - Group files with two columns: 'SampleID', 'Group'.
- geneListSortFile - Sorted gene list with two columns: 'Hugo_Symbol', 'Chromosome'.
- ID.mc.cores - The number of cores to use in ID type, i.e. at most how many child processes will be run simultaneously. Must be exactly 1 on Windows (which uses the master process).
- ID.row83 - Whether to keep 83 properties for ID type.
- plot - Whether to plot the image.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

cumulativeCA(file = 'Inputtest.maf', Pfile = 'SignatureComposition.12.Normalized.txt', Sfile = 'SampleContribution.12.Normalized.txt', sample.id = 'Tumor_Sample_Barcode', chr = 'Chromosome', pos = 'Start_position', ref = 'Reference_Allele', alt = 'Tumor_Seq_Allele2', Hugo = 'Hugo_Symbol', AnalCOSMICSigType = 'SBS')

---

eachMutationCA *Contribution abundance of each mutation*

Description

We present a tool to calculate the contribution abundance of each mutation in cancer research.

Usage

eachMutationCA(file = NULL, Pfile = NULL, Sfile = NULL, sample.id = 'Tumor_Sample_Barcode', chr = 'Chromosome', pos = 'Start_position', ref = 'Reference_Allele', alt = 'Tumor_Seq_Allele2', Hugo = 'Hugo_Symbol', Subtype = 'tricontext', AnalCOSMICSigType = c('SBS','DBS','ID'), genome.build = c("Ch37","Ch38"), significantGenesList = NULL, ID.mc.cores = 2, ID.row83 = TRUE)

Arguments

- file - A file with non-silent mutations obtained by annotation of mutation locations by Oncotator or other annotation software. And then can be generated through the SigsInput function. The format of the input dataset, contains MAF and VCF. File must have at least seven columns with the colnames of "Hugo_Symbol CHROM POS REF ALT
  SAMPLENAME SUBTYPE",
  for SBS as: 1 1432333 T C T1 C[C>T]T,
  for DBS as: 1 1432333 GC AA T1 GC>AA,
  for ID as: 1 1432333 T - T1 1:Del:T:3.
- Pfile - A mutational signature matrix file P with a format like the output result of RNMF software.
- Sfile - A abundance fractions matrix S with a format like the output result of RNMF software.
- sample.id - Column name in the mutation file corresponding to the Sample ID.
- chr - Column name in the mutation file corresponding to the chromosome.
- pos - Column name in the mutation file corresponding to the mutation

position.

- ref - Column name in the mutation file corresponding to the reference base.
- alt - Column name in the mutation file corresponding to the alternate base.
- Hugo - Column name in the mutation file corresponding to the gene.
- Subtype - Column name in the mutation file corresponding to the Subtype.
- AnalCOSMICSigType - Type of mutation based on PCAWG project, canbe SBS, DBS or ID.
- genome.build - The version of the reference genome used in the mutation set.
- significantGenesList - Sorted gene list with two columns: 'Hugo_Symbol', 'Chromosome'.
- ID.mc.cores - The number of cores to use in ID type, i.e. at most how many child processes will be run simultaneously. Must be exactly 1 on Windows (which uses the master process).
- ID.row83 - Whether to keep 83 properties for ID type.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

eachMutationCA(file = 'MutationInputSigsTypes.txt', Pfile = 'SignatureComposition.12.Normalized.txt', Sfile = 'SampleContribution.12.Normalized.txt', sample.id = 'Tumor_Sample_Barcode', chr = 'Chromosome', pos = 'Start_position', ref = 'Reference_Allele', alt = 'Tumor_Seq_Allele2', Hugo = 'Hugo_Symbol', Subtype = 'tricontext', AnalCOSMICSigType = 'SBS')

---

genePerMutSigs *Mutation enrichment analysis*

Description

We present a method based on R for association analysis of genes and mutational signatures. Mutation enrichment analysis identifies an association between somatic mutations and activity of signature in a discovery cohort.

Usage

genePerMutSigs(Sfile = NULL, choose.Sigs = 'SBS1', Gfile = NULL, file = NULL, sample.id = 'Tumor_Sample_Barcode', Hugo = 'Hugo_Symbol', Mutfreq = 0.04, threshold = 0.06, Qvalue = 0.1, plot = TRUE)

Arguments

- Sfile - A abundance fractions matrix S with a format like the output result of

RNMF software.

- choose.Sigs - Select a feature for subsequent analysis, and the feature name should match the feature header name in Sfile.
- Gfile - A abundance fractions matrix of genes for all samples with a format like the output result of RNMF software.
- file - A a non-silent mutation dataset, such as the annotation result file in MAF format.
- sample.id - Column name in the mutation file corresponding to the Sample ID.
- Hugo - Column name in the mutation file corresponding to the gene.
- Mutfreq - A minimum sample mutation rate threshold is used to determine the genes for subsequent analysis.
- threshold - A threshold is used to define the minimum contribution abundance of genes that can be displayed.
- Qvalue - A FDR threshold of genes are used to highlight in red.
- plot - Whether to plot the image.

Author(s)

Wen Luo (design, implementation, testing), Zhenzhang Li (design)

Examples

genePerMutSigs(Sfile = 'SampleContribution.12.Normalized.txt', choose.Sigs = 'SBS1', Gfile = 'SBS1.SBS.geneCumulativeContributionAbundance.txt', file = 'Non-silent.mutation.snv.txt', sample.id = 'Tumor_Sample_Barcode', Hugo = 'Hugo_Symbol', Mutfreq = 0.04, threshold = 0.2, Qvalue = 0.1, plot = TRUE)

samFisherSigs *Mutation enrichment analysis*

Description

We present a method based on R for association analysis of genes and mutational signatures. Mutation enrichment analysis identifies an association between somatic mutations and activity of signature in a discovery cohort.

Usage

samFisherSigs(Sfile = NULL, choose.Sigs = 'SBS1', file = NULL, sample.id = 'Tumor_Sample_Barcode', Hugo = 'Hugo_Symbol', Mutfreq = 0.04, threshold = 0.06, Qvalue = 0.1, plot = TRUE)

Arguments

- Sfile - An abundance fractions matrix S with a format like the output result from RNMF software.

- choose.Sigs - Select a feature for subsequent analysis, and the feature name should match the feature header name in Sfile.
- file - A a non-silent mutation dataset, such as the annotation result file in MAF format.
- sample.id - Column name in the mutation file corresponding to the Sample ID.
- Hugo - Column name in the mutation file corresponding to the gene.
- Mutfreq - A minimum sample mutation rate threshold is used to determine the genes for subsequent analysis.
- threshold - A threshold is used to define the minimum mutational exposure to samples that can be displayed.
- Qvalue - A FDR threshold of genes are used to highlight in red.
- plot - Whether to plot the image.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

References

Xing R, Zhou Y, Yu J, et al. Whole-genome sequencing reveals novel tandem-duplication hotspots and a prognostic mutational signature in gastric cancer[J]. Nature communications, 2019, 10(1): 1-13.

Examples

samFisherSigs(Sfile = 'SampleContribution.12.Normalized.txt', choose.Sigs = 'SBS1', file = 'Non-silent.mutation.snv.txt', sample.id = 'Tumor_Sample_Barcode', Hugo = 'Hugo_Symbol', Mutfreq = 0.04, threshold = 0.2, Qvalue = 0.1, plot = TRUE)

samPerMutSigs *Mutation enrichment analysis*

Description

We present a method based on R for association analysis of genes and mutational signatures. Mutation enrichment analysis identifies an association between somatic mutations and activity of signature in a discovery cohort.

Usage

samPerMutSigs(Sfile = NULL, choose.Sigs = 'SBS1', file = NULL, sample.id = 'Tumor_Sample_Barcode', Hugo = 'Hugo_Symbol', Mutfreq = 0.04, threshold = 0.06, Qvalue = 0.1, plot = TRUE)

Arguments

- Sfile - A abundance fractions matrix S with a format like the output result of RNMF software.
- choose.Sigs - Select a feature for subsequent analysis, and the feature name should match the feature header name in Sfile.
- file - A a non-silent mutation dataset, such as the annotation result file in MAF format.
- sample.id - Column name in the mutation file corresponding to the Sample ID.
- Hugo - Column name in the mutation file corresponding to the gene.
- Mutfreq - A minimum sample mutation rate threshold is used to determine the genes for subsequent analysis.
- threshold - A threshold is used to define the minimum mutational exposure of samples that can be displayed.
- Qvalue - A FDR threshold of genes are used to highlight in red.
- plot - Whether to plot the image.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

samPerMutSigs(Sfile = 'SampleContribution.12.Normalized.txt', choose.Sigs = 'SBS1', file = 'Non-silent.mutation.snv.txt', sample.id = 'Tumor_Sample_Barcode', Hugo = 'Hugo_Symbol', Mutfreq = 0.04, threshold = 0.2, Qvalue = 0.1, plot = TRUE)

IDlego *The tool is used to plot lego image for mutation spectrum*

Description

The tool is used to plot lego image for mutation spectrum, and its method references ID Signatures.

Usage

IDlego(file='originalGenomes', subtype='subtypes', genome.build = c("Ch37","Ch38"), SequenceType = c("WGS","WES"))

Arguments

- originalGenomes - a file, data matrix for NMF, [m x n, m stands for feature counts and n stands for sample counts].

- subtypes - a list of types of mutation, data matrix for NMF, [m x 1, m stands for feature counts].
- genome.build - The version of the reference genome used in the mutation set.
- SequenceType - Type of Sequence, canbe WGS, WES.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

IDlego(file='originalGenomes', subtype='subtypes', genome.build = "Ch37", SequenceType = "WGS")

DBSlego *The tool is used to plot lego image for mutation spectrum*

Description

The tool is used to plot lego image for mutation spectrum, and its method references DBS Signatures.

Usage

DBSlego(file='originalGenomes', subtype='subtypes', genome.build = c("Ch37","Ch38"), SequenceType = c("WGS","WES"))

Arguments

- originalGenomes - a file, data matrix for NMF, [m x n, m stands for feature counts and n stands for sample counts].
- subtypes - a list of types of mutation, data matrix for NMF, [m x 1, m stands for feature counts].
- genome.build - The version of the reference genome used in the mutation set.
- SequenceType - Type of Sequence, can be WGS, WES.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

DBSlego(file='originalGenomes', subtype='subtypes', genome.build = "Ch37", SequenceType = "WGS")

SBSlego *The tool is used to plot lego image for mutation spectrum*

Description

The tool is used to plot lego image for mutation spectrum, and its method comes from lwlegopt (https://github.com/BGI-LuoWen/lwlegopt).

Usage

SBSlego(file='originalGenomes', subtype='subtypes', scale=10, name=NULL, title='GC', sort=NULL, top=TRUE, color=NULL, border=NULL, genome.build = c("Ch37","Ch38"), SequenceType = c("WGS","WES"), RegionLength = NULL)

Arguments

- originalGenomes - a file, data matrix for NMF, [m x n, m stands for feature counts and n stands for sample counts].
- subtypes - a list of types of mutation, data matrix for NMF, [m x 1, m stands for feature counts].
- scale - Scale interval of Y axis.
- name - Prefix_Name of output file.
- title - Cancer type.
- sort - Sort for trinucleotide percentage value: TRUE, FALSE and NULL.
- top - Location of icon displaying: TRUE, FALSE.
- color - Colours for lego figure.
- border - Colours for border.
- genome.build - The version of the reference genome used in the mutation set.
- SequenceType - Type of Sequence, canbe WGS, WES.
- RegionLength - The length of sequence region.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

SBSlego(file='originalGenomes', subtype='subtypes', scale=10, name=NULL, title='GC', sort=NULL, top=TRUE, color=NULL, border=NULL, genome.build = "Ch37", SequenceType = "WGS")

similarityCOSMIC *Similarity calculation based on COSMIC Mutational Signatures*

Description

Calculate the similarities between features and draw heat maps.

Usage

similarityCOSMIC(Pfile, AnalCOSMICSigType = c('SBS','DBS','ID'), genome.build = c('Ch37', 'Ch38'), SBS.version = c("V2","V3"), plot = TRUE, fontsize = 6, filename = NULL)

Arguments

- Pfile - A mutational signature matrix file P with a format like the output result of RNMF software.
- AnalCOSMICSigType - Type of mutation based on PCAWG project, canbe SBS, DBS or ID.
- genome.build - The version of the reference genome used in the mutation set.
- SBS.version - Mutational signatures version 3 and version 2 signatures in COSMIC.
- plot - Whether to plot the image.
- fontsize - base fontsize for the plot.
- filename - file path where to save the picture. Filetype is decided by the extension in the path. Currently following formats are supported: png, pdf, tiff, bmp, jpeg. Even if the plot does not fit into the plotting window, the file size is calculated so that the plot would fit there, unless specified otherwise.

Author(s)

Wen Luo (design), Zhenzhang Li (implementation, testing)

Examples

similarityCOSMIC(Pfile = 'SignatureComposition.12.Normalized.txt', AnalCOSMICSigType = 'SBS', genome.build = 'Ch37', SBS.version = "V2", plot = TRUE, fontsize = 6, filename = NULL)

similarityAB *Similarity calculation*

Description

Calculate the similarities between features and draw heat maps.

Usage

similarityAB(Pfile1, Pfile2, AnalCOSMICSigType = c('SBS','DBS','ID'), plot = TRUE, fontsize = 6, filename = NULL)

Arguments

- Pfile1 - A mutational signature matrix file A with a format like the output result from RNMF software.
- Pfile2 - A mutational signature matrix file B with a format like the output result of RNMF software.
- AnalCOSMICSigType - Type of mutation based on PCAWG project, canbe SBS, DBS or ID.
- plot - Whether to plot the image.
- fontsize - base fontsize for the plot.
- filename - file path where to save the picture. Filetype is decided by the extension in the path. Currently following formats are supported: png, pdf, tiff, bmp, jpeg. Even if the plot does not fit into the plotting window, the file size is calculated so that the plot would fit there, unless specified otherwise.

Author(s)

Wen Luo (design, implementation), Zhenzhang Li (implementation, testing)

Examples

similarityAB(Pfile1 = 'A.12.Normalized.txt', Pfile2 = 'B.12.Normalized.txt', AnalCOSMICSigType = 'SBS', plot = TRUE, fontsize = 6, filename = NULL)

Index