Package 'SCAN.UPC'

November 12, 2026

Type Package

Title Single-channel array normalization (SCAN) and Universal exPression Codes (UPC)

Version 2.54.0

Author Stephen R. Piccolo and Andrea H. Bild and W. Evan Johnson

Maintainer Stephen R. Piccolo <stephen_piccolo@byu.edu>

Description SCAN is a microarray normalization method to facilitate personalized-medicine workflows. Rather than processing microarray samples as groups, which can introduce biases and present logistical challenges, SCAN normalizes each sample individually by modeling and removing probe- and array-specific background noise using only data from within each array. SCAN can be applied to one-channel (e.g., Affymetrix) or two-channel (e.g., Agilent) microarrays. The Universal exPression Codes (UPC) method is an extension of SCAN that estimates whether a given gene/transcript is active above background levels in a given sample. The UPC method can be applied to one-channel or two-channel microarrays as well as to RNA-Seq read counts. Because UPC values are represented on the same scale and have an identical interpretation for each platform, they can be used for cross-platform data integration.

License MIT

Depends R (>= 2.14.0), Biobase (>= 2.6.0), oligo, Biostrings, GEOquery, affy, affyio, foreach, sva

Suggests pd.hg.u95a

Imports utils, methods, MASS, tools, IRanges

URL http://bioconductor.org, http://jlab.bu.edu/software/scan-upc

biocViews ImmunoOncology, Software, Microarray, Preprocessing, RNASeq, TwoChannel, OneChannel

git_url https://git.bioconductor.org/packages/SCAN.UPC

git_branch RELEASE_3_23

git_last_commit 8172e68

git_last_commit_date 2026-04-28

Repository Bioconductor 3.23

Date/Publication 2026-11-11

Contents

BatchAdjust 2
BatchAdjustFromFile 3
InstallBrainArrayPackage 4
ParseMetaFromGtfFile 5
SCAN 6
SCAN_TwoColor 9
UPC_Generic 10
UPC_Generic_ExpressionSet 12
UPC_RNASeq 13
UPC_RNASeq_Single 15
UPC_TwoColor 16

Index **19**

`BatchAdjust` *Helper function for applying the ComBat batch-adjusting methodology to an ExpressionSet object*

Description

This is a convenience function for adjusting an ExpressionSet object for batch effects.

Usage

```
BatchAdjust(expressionSet, batchVariableName, covariateVariableNames=c())
```

Arguments

`expressionSet`	An `ExpressionSet` object that contains data to be adjusted for batch effects. This object should contain at least one `phenoData` column that indicates the batch number for each sample. Covariates may also be specified in other `phenoData` columns. Required.
`batchVariableName`	Name of `phenoData` column that indicates the batch number for each sample. Required.
`covariateVariableNames`	Name(s) of `phenoData` column(s) that indicate the covariate information for each sample. Optional.

Value

An ExpressionSet object indicating expression levels after batch adjustment.

Note

This function provides a convenient way to batch adjust an `ExpressionSet` object. It invokes the `ComBat` function within the `sva` package. Please see that package for additional details about how batch adjusting is performed. If multiple covariates are specified, a model matrix will be constructed that accounts for all covariates.

Author(s)

Stephen R. Piccolo

BatchAdjustFromFile *Helper function for applying the ComBat batch-adjusting methodology to an ExpressionSet object*

Description

This is a convenience function for adjusting an ExpressionSet object for batch effects.

Usage

```
BatchAdjustFromFile(expressionSet, batchFilePath)
```

Arguments

`expressionSet`	An `ExpressionSet` object that contains data to be adjusted for batch effects. This object should contain at least one `phenoData` column that indicates the batch number for each sample. Covariates may also be specified in other `phenoData` columns. Required.
`batchFilePath`	Tab-separated text file that indicates batch (and optionally, covariate information) for each sample. Required.

Value

An ExpressionSet object indicating expression levels after batch adjustment.

Note

This function provides a convenient way to batch adjust an `ExpressionSet` object. It invokes the `ComBat` function within the `sva` package. Please see that package for additional details about how batch adjusting is performed.

The batch file must contain a row for each of the samples in the `ExpressionSet` object, and one of the columns must use the name "batch" to indicate that the values represent batch number. The remaining column(s), if present, will be treated as covariates. If multiple covariates are specified, a model matrix will be constructed that accounts for all covariates.

Author(s)

Stephen R. Piccolo

`InstallBrainArrayPackage`
Helper function for installing BrainArray packages

Description

When processing Affymetrix microarrays, users can specify alternative probe/gene mappings via the probeSummaryPackage parameter. Users can download such packages directly from the BrainArray web site and install them manually. Or they can use this helper function to download and install them in a single step.

Usage

```
InstallBrainArrayPackage(celFilePath, version, organism, annotationSource)
```

Arguments

`celFilePath`	Path to an example CEL file. The Affymetrix version name will be extracted from this file.
`version`	Version of BrainArray (for example, "17.1.0") you want to download.
`organism`	Abbreviated name (for example, "hs" = homo sapiens, "mm" = mus musculus, "gg" = gallus gallus) of the organism for which the microarrays are designed.
`annotationSource`	Abbreviated name of the annotation source (for example, "entrezg" = Entrez Gene, "ensg" = Ensembl Gene) you want to use.

Value

A character object that indicates the name of the installed package.

Note

Information about BrainArray versions, organism names, and annotation sources can be obtained via the BrainArray web site (http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/genomic_curated_CDF.asp).

Author(s)

Stephen R. Piccolo

Examples

```
## Not run:
pkgName = InstallBrainArrayPackage(celFilePath, "17.0.1", "hs", "entrezg")

## End(Not run)
```

`ParseMetaFromGtfFile` *Helper function to parse length and GC content information from a GTF file.*

Description

When applying the `UPC_RNASeq` function, it is possible to correct for the length and GC content of genomic features. To accomplish this, an annotation file indicating these values for each feature must be provided. This helper function enables users to generate an annotation file, using a GTF file and genome FASTA file as references.

Usage

```
ParseMetaFromGtfFile(gtfFilePath, fastaFilePattern, outFilePath,
                     featureTypes=c("protein_coding"), attributeType="gene_id")
```

Arguments

`gtfFilePath`	Path to the GTF file that will be parsed.
`fastaFilePattern`	File pattern that indicates where FASTA file(s) for the associated reference genome can be found.
`outFilePath`	Path where the output file will be stored.
`featureTypes`	One or more feature types (for example, "protein_coding," "unprocessed_pseudogene") that should be extracted from the GTF file. The default is "protein_coding."
`attributeType`	The type of attribute ("gene_id", "transcript_id") to be parsed. Values will be grouped according to these attributes.

Author(s)

Stephen R. Piccolo

Examples

```
## Not run:
ParseMetaFromGtfFile("GRCh37_XY.gtf", "GRCh37.fa", "GRCh37_Annotation.txt")

## End(Not run)
```

SCAN *Single-Channel Array Normalization (SCAN) and Universal exPression Codes (UPC) for Affymetrix microarrays*

Description

This function is used to normalize Affymetrix .CEL files via the SCAN and UPC methods.

Usage

```
SCAN(celFilePattern, outFilePath = NA, convThreshold = 0.01, annotationPackageName = NA,
     probeSummaryPackage = NA, probeLevelOutDirPath = NA, exonArrayTarget=NA,
     batchFilePath=NA, verbose = TRUE)
SCANfast(celFilePattern, outFilePath = NA, convThreshold = 0.50, annotationPackageName = NA,
     probeSummaryPackage = NA, probeLevelOutDirPath = NA, exonArrayTarget=NA,
     batchFilePath=NA, verbose = TRUE)
UPC(celFilePattern, outFilePath = NA, convThreshold = 0.01, annotationPackageName = NA,
     probeSummaryPackage = NA, probeLevelOutDirPath = NA, exonArrayTarget = NA,
     modelType="nn", batchFilePath=NA, verbose = TRUE)
UPCfast(celFilePattern, outFilePath = NA, convThreshold = 0.50, annotationPackageName = NA,
     probeSummaryPackage = NA, probeLevelOutDirPath = NA, exonArrayTarget = NA,
     modelType="nn", batchFilePath=NA, verbose = TRUE)
```

Arguments

`celFilePattern` Absolute or relative path to the input file to be processed. This is the only required parameter. To process multiple files, wildcard characters can be used (e.g., "*.CEL"). Alternatively, a Gene Expression Omnibus identifier (e.g., GSE22309 or GSM555237) can be specified.

`outFilePath` Absolute or relative path where the output file will be saved. This is optional.

`convThreshold` Convergence threshold that determines at what point the mixture-model parameters have stabilized. The default value should be suitable in most cases. However, if the model fails to converge, it may be useful to adjust this value. (This parameter is optional.)

`annotationPackageName`
The name of an annotation package that specifies the layout and sequences of the probes. This is optional. By default, the correct annotation package should be identified in most cases. However, with this option allows the user to specify the package explicitly if needed.

`probeSummaryPackage`
An R package that specifies alternative probe/gene mappings. This is optional. See note below for more details.

`probeLevelOutDirPath`
Absolute or relative path to a directory where probe-level normalized values can be saved. This is optional. By default, the probe-level values will be discarded after they have been summarized. However, if the user has a need to repeatedly

process the same file (perhaps to try various probe/gene mappings), this option can be useful because SCAN will retrieve previously normalized values if a probe-level file exists, rather than renormalize the raw data. The user should be aware that probe-level files may consume a considerable amount of disk space.

exonArrayTarget
: The type of probes to be used. This parameter is optional and should only be specified when Affymetrix Exon 1.0 ST arrays are being processed. This parameter allows the user to specify the subset of probes that should be used and how the probes should be grouped. Available options are NA, "core", "extended", "full", or "probeset". When "probeset" is used, all probes will be used, and the probes will be grouped according to the Affymetrix probeset definitions. When "core", "extended", or "full" are used, the probes that Affymetrix has defined to fall within each classification will be used, and probes will be grouped by Entrez Gene IDs (as defined in the corresponding annotation package). It is recommended to specify "probeset" when the *probeSummaryPackage* parameter is being used so that all probes will be considered.

modelType
: The type of mixture model to be used. This value can be either "nn" (default) or "nn_bayes."

batchFilePath
: Absolute or relative path to a tab-separated text file that indicates batch (and optionally, covariate information) for each sample. Optional.

verbose
: Whether to output more detailed status information as files are normalized. Default is TRUE.

Value

An ExpressionSet object that contains a row for each probeset/gene/transcript and a column for each input file. SCAN values will be on a log2 scale, centered at zero. UPC values will range between zero and one (lower values indicate that the gene is inactive and higher values indicating that the gene is active).

Note

If a Gene Expression Omnibus (GEO) identifier is specified for the celFilePattern parameter, an attempt will be made to download the sample(s) directly from GEO. If a study identifier (e.g., GSE22309) is specified, all CEL files from that study will be downloaded. If a sample identifier (e.g., GSM555237) is specified, only that sample will be downloaded.

By default, SCAN and UPC use the default mappings between probes and genes that have been provided by the manufacturer. However, these mappings may be outdated or may include problematic probes (for example, those that cross hybridize). The default mappings also may produce multiple summary values per gene. Alternative mappings, such as those provided by the BrainArray resource (see http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/genomic_curated_CDF.asp), allow SCAN and UPC to produce a single value per gene and to use updated gene definitions. Users can specify alternative mappings using the probeSummaryPackage parameter. If specified, this package must conform to the standards of the AnnotationDbi package. The BrainArray packages can be downloaded from http://brainarray.mbni.med.umich.edu/Brainarray/Database/CustomCDF/CDF_download.asp. When using BrainArray, be sure to download the R source package for probe-level mappings (see vignette for more information).

Because the SCAN/UPC algorithm accounts for nucleotide-level genomic composition across thousands of probes, it may take several minutes to normalize a sample, depending on the computer's processor speed and the type of microarray. To enable users to normalize samples in a shorter period of time, we have provided alternative functions called `SCANfast` and `UPCfast`. In this approach, a smaller number of probes is used for normalization, and a less stringent convergence threshold is used by default. We have found that microarrays processed with `SCANfast` (using default parameters) require 75% less processing time (on average) but produce output values that correlate strongly (r = 0.998) with values produced by the `SCAN` function for the same arrays.

It is also possible to execute these functions in parallel. This approach uses the `foreach` package behind the scenes. If you have registered a parallel backend (for example, via the `doParallel` package), multiple CEL files can be processed in parallel. Otherwise, the files will be processed sequentially.

The `batchFilePath` parameter provides a convenient way to adjust the data for batch effects. It invokes the `ComBat` function within the `sva` package. Please see that package for additional details about how batch adjusting is performed. Batch adjusting will be performed after values have been SCAN normalized and summarized at the gene/probeset level. This is also true when `UPC` and `UPCfast` are being used—the data will be SCAN normalized and summarized, then batch adjusting will be performed, and lastly UPC transformation will occur. This process is different from when `UPC` or `UPCfast` are invoked without batch information; in this scenario, no SCAN normalization will occur.

The `modelType` parameter indicates which type of mixture model to use for UPC transformation. The "nn" model type has been used since the default implementation of this package. The "nn_bayes" model type is an experimental new approach intended for experiments where a subset of genes are expressed at extreme levels. When this model type is used, values will first be SCAN normalized and summarized at the gene/probeset level.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Sun Y, Campbell JD, Lenburg ME, Bild AH, and Johnson WE. A single-sample microarray normalization method to facilitate personalized-medicine workflows. *Genomics*, 2012, 100:6, pp. 337-344. Piccolo SR, Withers MR, Francis OE, Bild AH and Johnson WE. Multi-platform single-sample estimates of transcriptional activation. Proceedings of the National Academy of Sciences of the United States of America, 2013, 110(44):11778-17783.

Examples

```
## Not run:
# SCAN normalize a CEL file from GEO
normalized = SCAN("GSM555237")

# UPC normalize a CEL file from GEO
normalized = UPC("GSM555237")

# Normalize a CEL file and save output to a file
normalized = SCAN("GSM555237", "output_file.txt")
```

```
# Normalize a CEL file and summarize at the gene level using BrainArray
# mappings for Entrez Gene. First it is necessary to install the package
# and obtain the package name. For demonstration purposes, this file
# will be downloaded manually from GEO.
tmpDir = tempdir()
getGEOSuppFiles("GSM555237", makeDirectory=FALSE, baseDir=tmpDir)
celFilePath = file.path(tmpDir, "GSM555237.CEL.gz")
pkgName = InstallBrainArrayPackage(celFilePath, "17.0.1", "hs", "entrezg")
normalized = SCAN(celFilePath, probeSummaryPackage=pkgName)

# Normalize multiple files in parallel on multiple cores within a given
# computer. It is also possible using the doParallel package to spread
# the workload across multiple computers on a cluster.
library(doParallel)
registerDoParallel(cores=2)
result = SCAN("GSE22309")

## End(Not run)
```

SCAN_TwoColor *Single-Channel Array Normalization (SCAN) for Agilent two-color expression microarrays*

Description

This function is used to normalize Agilent two-color expression microarrays via the SCAN method.

Usage

```
SCAN_TwoColor(inFilePattern, outFilePath = NA, batchFilePath = NA, verbose = TRUE)
```

Arguments

inFilePattern	Absolute or relative path to the input file to be processed. To process multiple files, wildcard characters can be used (e.g., "*.txt"). Alternatively, a Gene Expression Omnibus identifier (e.g., GSE39655 or GSM1072833) can be specified. This is the only required parameter.
outFilePath	Absolute or relative path where the output file will be saved. This is optional.
batchFilePath	Absolute or relative path to a tab-separated text file that indicates batch (and optionally, covariate information) for each sample. Optional.
verbose	Whether to output more detailed status information as files are normalized. Default is TRUE.

Value

A list is returned, containing two elements: a matrix containing normalized data values and a vector of probe names that correspond to each row of the matrix. The matrix will contain two columns—one corresponding to each channel—for each sample. When the array design contains duplicate probe names (this is common for control probes), the vector of probe names will also contain duplicates.

Note

If a Gene Expression Omnibus (GEO) identifier is specified for the `inFilePattern` parameter, an attempt will be made to download the sample(s) directly from GEO. If a study identifier (e.g., GSE39655) is specified, all CEL files from that study will be downloaded. If a sample identifier (e.g., GSM1072833) is specified, only that sample will be downloaded.

The `batchFilePath` parameter provides a convenient way to adjust the data for batch effects. It invokes the `ComBat` function within the `sva` package. Please see that package for additional details about how batch adjusting is performed. Batch adjusting is performed after values have been SCAN normalized.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Sun Y, Campbell JD, Lenburg ME, Bild AH, and Johnson WE. A single-sample microarray normalization method to facilitate personalized-medicine workflows. *Genomics*, 2012, 100:6, pp. 337-344.

Examples

```
## Not run:
# Normalize a file from GEO and save output to a file
result = SCAN_TwoColor("GSM1072833", "output_file.txt")

## End(Not run)
```

UPC_Generic *Generic function to apply Universal exPression Codes (UPC) transformation*

Description

This function can be used to derive UPC values to any type of gene-expression data. It requires the user to specify expression values for many genes (or transcripts, exons, or probes). And optionally, the user can specify the length and/or GC content (proportion of G or C bases) for the corresponding genomic region (e.g., gene). If these values are specified, the UPC algorithm corrects for biases resulting from length or GC content.

Usage

```
UPC_Generic(expressionValues, lengths = NULL, gcContent = NULL, modelType = "nn",
  convThreshold = 0.001, higherValuesIndicateHigherExpression = TRUE, verbose = TRUE)
```

Arguments

expressionValues	A numeric vector that indicates expression levels. This is the only required parameter.
lengths	An integer vector that indicates the length of the genomic region that corresponds to each expression value. The order of these values must be identical to the order of the expression values. This parameter is optional.
gcContent	A numeric vector that indicates the proportion of bases in the genomic region that are a G or C nucleotide. The order of these values must be identical to the order of the expression values. This parameter is optional.
modelType	Various models can be used for the mixture model to differentiate between active and inactive probes. The default is the normal-normal model ("nn"), which uses the normal distribution. Other available options are log-normal ("ln"), negative-binomial ("nb"), and normal-normal Bayes ("nn_bayes").
convThreshold	Convergence threshold that determines at what point the mixture-model parameters have stabilized. The default value should be suitable in most cases. However, if the model fails to converge (or converges too quickly), it may be useful to adjust this value. Optional.
higherValuesIndicateHigherExpression	In most cases, higher expression values indicate relatively high expression. However, if higher values indicate relatively low expresssion, this parameter can be used to indicate such. Accordingly, UPC values closer to one will indicate higher expression, and UPC values closer to zero will indicate the opposite. Optional.
verbose	Whether to output more detailed status information as processing occurs. Default is TRUE.

Value

An numeric vector with UPC values whose order corresponds with the order of the input values.

Note

The `modelType` parameter indicates which type of mixture model to use for UPC transformation. The "nn_bayes" model type is an experimental new approach intended for experiments where a subset of genes are expressed at extreme levels.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Withers MR, Francis OE, Bild AH and Johnson WE. Multi-platform single-sample estimates of transcriptional activation. Proceedings of the National Academy of Sciences of the United States of America, 2013, 110:44 17778-17783.

`UPC_Generic_ExpressionSet` *Generic function to apply Universal exPression Codes (UPC) transformation to ExpressionSet objects*

Description

This function can be used to derive UPC values for any type of gene-expression data. The key prerequisite is that the data be placed into an ExpressionSet object. And optionally, the user can specify the length and/or GC content (proportion of G or C bases) for the corresponding genomic region (e.g., gene). If these values are specified, the UPC algorithm will correct for biases resulting from length or GC content.

Usage

```
UPC_Generic_ExpressionSet(expressionSet, sequenceFeatureName = NA, modelType = "nn",
  convThreshold = 0.001, higherValuesIndicateHigherExpression = TRUE, verbose = TRUE)
```

Arguments

`expressionSet`	An `ExpressionSet` object that contains data to be UPC transformed. Required.
`sequenceFeatureName`	Optionally, the DNA sequence corresponding to each feature (e.g., microarray probe) can be specified in the `ExpressionSet` object's metadata. If specified, the UPC transformation will take the G/C content and length of each feature into account. Optional.
`modelType`	Various models can be used for the mixture model to differentiate between active and inactive probes. The default is the normal-normal model ("nn"), which uses the normal distribution. Other available options are log-normal ("ln"), negative-binomial ("nb"), and normal-normal Bayes ("nn_bayes").
`convThreshold`	Convergence threshold that determines at what point the mixture-model parameters have stabilized. The default value should be suitable in most cases. However, if the model fails to converge (or converges too quickly), it may be useful to adjust this value. (This parameter is optional.)
`higherValuesIndicateHigherExpression`	In most cases, higher expression values indicate relatively high expression. However, if higher values indicate relatively low expresssion, this parameter can be used to indicate such. Accordingly, UPC values closer to one will indicate higher expression, and UPC values closer to zero will indicate the opposite. (This parameter is optional.)
`verbose`	Whether to output more detailed status information as processing occurs. Default is TRUE.

Value

An ExpressionSet object with UPC values.

Note

The example below illustrates how an ExpressionSet object can be retrieved from a given Gene Expression Omnibus experiment and then UPC transformed, taking each probe's sequence into account.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Withers MR, Francis OE, Bild AH and Johnson WE. Multi-platform single-sample estimates of transcriptional activation. Proceedings of the National Academy of Sciences of the United States of America, 2013, 110:44 17778-17783.

Examples

```
## Not run:
upcData = UPC_Generic_ExpressionSet(getGEO("GSE56234", GSEMatrix=TRUE)[[1]], sequenceFeatureName="SEQUENCE")

## End(Not run)
```

UPC_RNASeq *Universal exPression Codes (UPC) for RNA-Seq data*

Description

This function is used to derive UPC values for RNA-Seq data. It requires at least one input file that specifies a read count for each genomic region (e.g., gene). This file should list a unique identifier for each region in the first column and corresponding read counts (not RPKM/FPKM values) in the second column.

This function also can correct for the GC content and length of each genomic region. Users who wish to enable this correction must provide a separate annotation file. This tab-separated file should contain a row for each genomic region. The first column should contain a unique identifier that corresponds to identifiers from the read-count input file. The second column should indicate the length of the genomic region. And the third column should specify the number of G or C bases in the region. The ParseMetaFromGtfFile function can be used to generate annotation files.

Usage

```
UPC_RNASeq(inFilePattern, annotationFilePath = NA, outFilePath = NA, modelType = "nn",
  convThreshold = 0.01, ignoreZeroes = FALSE, numDataHeaderRows=0,
  numAnnotationHeaderRows=0, batchFilePath=NA, verbose = TRUE)
```

Arguments

inFilePattern	Absolute or relative path to the input file(s) to be processed. The input file(s) can contain one or more columns, where each column would contain data for a given sample. To process multiple files, wildcard characters can be used (e.g., "*.txt"). Required.
annotationFilePath	Absolute or relative path where the annotation file is located. This parameter is optional.
outFilePath	Absolute or relative path where the output file will be saved. This is optional.
modelType	Various models can be used for the mixture model to differentiate between active and inactive probes. The default is the normal-normal model ("nn"), which uses the normal distribution. Other available options are log-normal ("ln"), negative-binomial ("nb"), and normal-normal Bayes ("nn_bayes").
convThreshold	Convergence threshold that determines at what point the mixture-model parameters have stabilized. The default value should be suitable in most cases. However, if the model fails to converge (or converges too quickly), it may be useful to adjust this value. (This parameter is optional.)
ignoreZeroes	Whether to ignore read counts equal to zero when performing UPC calculations. Default is FALSE.
numDataHeaderRows	The number of header rows present in the input data file(s). If a header is present, the column names will be used as sample IDs.
numAnnotationHeaderRows	The number of header rows present in the annotation data file (if one has been specified).
batchFilePath	Absolute or relative path to a tab-separated text file that indicates batch (and optionally, covariate information) for each sample. Optional.
verbose	Whether to output more detailed status information as files are normalized. Default is TRUE.

Value

An ExpressionSet object that contains a row for each probeset/gene/transcript and a column for each input file.

Note

RNA-Seq data by nature have a lot of zero read counts. Samples with an excessive number of zeroes may lead to error messages because genes cannot be allocated properly to bins. The user can specify ignoreZeroes=TRUE to avoid this error. In practice, we have seen that the resulting UPC values are similar with either approach.

The batchFilePath parameter provides a convenient way to adjust the data for batch effects. It invokes the ComBat function within the sva package. Please see that package for additional details about how batch adjusting is performed. Batch adjusting is performed before the values are UPC transformed.

The modelType parameter indicates which type of mixture model to use for UPC transformation. The "nn_bayes" model type is an experimental new approach intended for experiments where a subset of genes are expressed at extreme levels.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Withers MR, Francis OE, Bild AH and Johnson WE. Multi-platform single-sample estimates of transcriptional activation. Proceedings of the National Academy of Sciences of the United States of America, 2013, 110:44 17778-17783.

Examples

```
## Not run:
result = UPC_RNASeq("ReadCounts.txt", "Annotation.txt")

## End(Not run)
```

UPC_RNASeq_Single *Apply Universal exPression Codes (UPC) transformation to a single RNA-Seq sample*

Description

This function is used to derive UPC values for a single RNA-Seq sample. It requires an input vector that specifies a read count for each genomic region (e.g., gene). Optionally, this function can correct for the GC content and length of each genomic region.

Usage

```
UPC_RNASeq_Single(expressionValues, featureNames, lengths = NULL,
  gcContent = NULL, modelType = "nn", convThreshold = 0.01,
  ignoreZeroes = FALSE, verbose = TRUE)
```

Arguments

expressionValues	A vector of RNA-Seq count values. Required.
featureNames	A vector of unique names that correspond to the count values. Required.
lengths	A vector indicating the length (in genomic bases) of the genomic region that corresponds to the count values.
gcContent	A vector indicating the number of G/C bases in the genomic region that corresponds to the count values.

modelType	Various models can be used for the mixture model to differentiate between active and inactive probes. The default is the normal-normal model ("nn"), which uses the normal distribution. Other available options are log-normal ("ln"), negative-binomial ("nb"), and normal-normal Bayes ("nn_bayes").
convThreshold	Convergence threshold that determines at what point the mixture-model parameters have stabilized. The default value should be suitable in most cases. However, if the model fails to converge (or converges too quickly), it may be useful to adjust this value. (This parameter is optional.)
ignoreZeroes	Whether to ignore read counts equal to zero when performing UPC calculations. Default is FALSE.
verbose	Whether to output more detailed status information as files are normalized. Default is TRUE.

Value

A vector that contains a UPC value for each probeset/gene/transcript.

Note

RNA-Seq data by nature have a lot of zero read counts. Samples with an excessive number of zeroes may lead to error messages because genes cannot be allocated properly to bins. The user can specify `ignoreZeroes=TRUE` to avoid this error. In practice, we have seen that the resulting UPC values are similar with either approach.

The `modelType` parameter indicates which type of mixture model to use for UPC transformation. The "nn_bayes" model type is an experimental new approach intended for experiments where a subset of genes are expressed at extreme levels.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Withers MR, Francis OE, Bild AH and Johnson WE. Multi-platform single-sample estimates of transcriptional activation. Proceedings of the National Academy of Sciences of the United States of America, 2013, 110:44 17778-17783.

`UPC_TwoColor` *Universal exPression Codes (UPC) for two-channel microarrays*

Description

This function is used to normalize two-channel expression microarrays (from Agilent) using the Universal exPression Codes (UPC) approach. In raw form, such microarray data come in the form of tab-separate data files.

Usage

```
UPC_TwoColor(inFilePattern, outFilePath = NA, modelType="nn",
  convThreshold=0.01, batchFilePath = NA, verbose = TRUE)
```

Arguments

inFilePattern	Absolute or relative path to the input file to be processed. To process multiple files, wildcard characters can be used (e.g., "*.txt"). Alternatively, a Gene Expression Omnibus identifier (e.g., GSE39655 or GSM1072833) can be specified.(This is the only required parameter.)
outFilePath	Absolute or relative path where the output file will be saved. (This parameter is optional.)
modelType	Various models can be used for the mixture model to differentiate between active and inactive probes. The default is the normal-normal model ("nn"), which uses the normal distribution. Other available options are log-normal ("ln"), negative-binomial ("nb"), and normal-normal Bayes ("nn_bayes").
convThreshold	Convergence threshold that determines at what point the mixture-model parameters have stabilized. The default value should be suitable in most cases. However, if the model fails to converge, it may be useful to adjust this value. Optional.
batchFilePath	Absolute or relative path to a tab-separated text file that indicates batch (and optionally, covariate information) for each sample. Optional.
verbose	Whether to output more detailed status information as files are processed. Default is TRUE.

Value

A list is returned, containing two elements: a matrix containing UPC values and a vector of probe names that correspond to each row of the matrix. The matrix will contain two columns—one corresponding to each channel—for each sample. When the array design uses duplicate probe names (this is common for control probes), the vector of probe names will also contain duplicates.

Note

If a Gene Expression Omnibus (GEO) identifier is specified for the inFilePattern parameter, an attempt will be made to download the sample(s) directly from GEO. If a study identifier (e.g., GSE39655) is specified, all CEL files from that study will be downloaded. If a sample identifier (e.g., GSM1072833) is specified, only that sample will be downloaded.

The batchFilePath parameter provides a convenient way to adjust the data for batch effects. It invokes the ComBat function within the sva package. Please see that package for additional details about how batch adjusting is performed. Batch adjusting is performed before UPC transformation occurs.

The modelType parameter indicates which type of mixture model to use for UPC transformation. The "nn_bayes" model type is an experimental new approach intended for experiments where a subset of genes are expressed at extreme levels.

Author(s)

Stephen R. Piccolo

References

Piccolo SR, Withers MR, Francis OE, Bild AH and Johnson WE. Multi-platform single-sample estimates of transcriptional activation. Proceedings of the National Academy of Sciences of the United States of America, 2013, 110:44 17778-17783.

Examples

```
## Not run:
# Normalize a file from GEO and save output to a file
result = UPC_TwoColor("GSM1072833", "output_file.txt")

## End(Not run)
```

Index

BatchAdjust, 2
BatchAdjustFromFile, 3

InstallBrainArrayPackage, 4

ParseMetaFromGtfFile, 5, *13*

SCAN, 6
SCAN_TwoColor, 9
SCANfast *(*SCAN*)*, 6

UPC *(*SCAN*)*, 6
UPC_Generic, 10
UPC_Generic_ExpressionSet, 12
UPC_RNASeq, *5*, 13
UPC_RNASeq_Single, 15
UPC_TwoColor, 16
UPCfast *(*SCAN*)*, 6