

# Package ‘cummeRbund’

September 24, 2012

**Title** Analysis, exploration, manipulation, and visualization of Cufflinks high-throughput sequencing data.

**Version** 1.2.0

**Date** 2011-08-18

**Author** L. Goff, C. Trapnell

**Description** Allows for persistent storage, access, exploration, and manipulation of Cufflinks high-throughput sequencing data. In addition, provides numerous plotting functions for commonly used visualizations.

**Suggests** cluster

**Imports** methods

**Depends** R (>= 2.7.0), RSQLite, ggplot2, reshape2

**Maintainer** Loyal A. Goff <lgofff@csail.mit.edu>

**License** Artistic-2.0

**Collate** AllGenerics.R AllClasses.R database-setup.R methods-CuffSet.R methods-CuffData.R methods-CuffDist.R methods-CuffGeneSet.R methods-CuffFeatureSet.R methods-CuffGene.R methods-CuffFeature.R tools.R

**LazyLoad** yes

**biocViews** HighThroughputSequencing, HighThroughputSequencingData, RNAseq, RNAseq-Data, GeneExpression, DifferentialExpression, Infrastructure, DataImport, DataRepresentation, Visualization, Bioinformatics, Clustering, MultipleComparisons, QualityControl

## R topics documented:

cummeRbund-package	2
addFeatures	6
createDB	7
csBoxplot	7
csCluster	8
csClusterPlot	9
csDendro	10
csDensity	11

csHeatmap . . . . .	12
csScatter . . . . .	13
csSpecificity . . . . .	15
csVolcano . . . . .	16
CuffData-class . . . . .	17
CuffDist-class . . . . .	18
CuffFeature-class . . . . .	19
CuffFeatureSet-class . . . . .	20
CuffGene-class . . . . .	21
CuffGeneSet-class . . . . .	22
CuffSet-class . . . . .	23
diffData . . . . .	25
distValues . . . . .	26
expressionBarplot . . . . .	27
expressionPlot . . . . .	28
featureNames . . . . .	29
features . . . . .	30
findSimilar . . . . .	31
fpkm-methods . . . . .	32
fpkmMatrix . . . . .	32
getGene . . . . .	33
getGenes . . . . .	34
getLevels . . . . .	35
getSig . . . . .	36
getSigTable . . . . .	37
JSdist . . . . .	38
JSdistFromP . . . . .	39
JSdistVec . . . . .	40
makeprobs . . . . .	41
makeprobsvec . . . . .	42
MAplot . . . . .	43
PINK1 . . . . .	44
readCufflinks . . . . .	44
sampleGeneSet . . . . .	46
sampleIDs . . . . .	46
samples . . . . .	47
shannon.entropy . . . . .	48

## Index 49

---

cummeRbund-package	<i>cummeRbund: The finishing touch on your Tuxedo workflow. Analysis, manipulation, and visualization of Cufflinks HTS data. ~ package title ~</i>
--------------------	--

---

## Description

Allows for persistent storage, access, and manipulation of Cufflinks high-throughput sequencing data. In addition, provides numerous plotting functions for commonly used visualizations. ~ A concise (1-5 lines) description of the package ~

**Details**

```

Package:   cummeRbund
Version:   0.1.3
Suggests:
Depends:   R (>= 2.7.0), RSQLite, reshape2, ggplot2, methods
License:   MIT License
Collate:   AllGenerics.R AllClasses.R database-setup.R methods-CuffSet.R methods-CuffData.R methods-CuffDist.R
LazyLoad:  yes
biocViews: HighThroughputSequencing, HighThroughputSequencingData, RNAseq, RNAseqData, GeneExpression, D
Packaged:  2011-08-05 18:03:50 UTC; lgoff
Built:     R 2.12.1; ; 2011-08-05 18:03:57 UTC; unix

```

#### Index:

```

CuffData-class          Class "CuffData"
CuffDist-class          Class "CuffDist"
CuffFeature-class       Class "CuffFeature"
CuffFeatureSet-class    Class "CuffFeatureSet"
CuffGene-class          Class "CuffGene"
CuffGeneSet-class       Class "CuffGeneSet"
CuffSet-class           Class "CuffSet"
JSdlist                 Jensen-Shannon distance on columns
addFeatures             addFeatures
addFeatures-methods    ~ Methods for Function addFeatures in Package
                        'cummeRbund' ~
createDB                createDB
csBoxplot              csBoxplot
csBoxplot-methods     ~ Methods for Function csBoxplot in Package
                        'cummeRbund' ~
csDensity              Density plot of CuffData
csDensity-methods     ~ Methods for Function csDensity in Package
                        'cummeRbund' ~
csHeatmap              csHeatmap
csHeatmap-methods     ~ Methods for Function csHeatmap in Package
                        'cummeRbund' ~
csScatter              Scatter Plot
csScatter-methods     ~ Methods for Function csScatter in Package
                        'cummeRbund' ~
csVolcano              Volcano Plot
csVolcano-methods     ~ Methods for Function csVolcano in Package
                        'cummeRbund' ~
cummeRbund-package     cummeRbund: The finishing touch on your Tuxedo
                        workflow. Analysis, manipulation, and
                        visualization of Cufflinks HTS data.
diffData               Differential comparison data
diffData-methods      ~ Methods for Function diffData in Package
                        'cummeRbund' ~
dim-methods            ~ Methods for Function dim in Package 'base'
                        ~
expressionBarplot      Barplot
expressionBarplot-methods ~ Methods for Function expressionBarplot in

```

```

Package 'cummeRbund' ~~
expressionPlot      Expression Plot
expressionPlot-methods
  ~~ Methods for Function expressionPlot in
  Package 'cummeRbund' ~~
featureNames        Feature names
featureNames-methods
  ~~ Methods for Function featureNames in Package
  'cummeRbund' ~~
features            Features
features-methods    ~~ Methods for Function features in Package
  'cummeRbund' ~~
fpkm                Retrieve FPKM values
fpkm-methods        ~~ Methods for Function fpkm in Package
  'cummeRbund' ~~
fpkmMatrix          Retrieve FPKM values as matrix
fpkmMatrix-methods
  ~~ Methods for Function fpkmMatrix in Package
  'cummeRbund' ~~
getGene             getGene
getGene-methods     ~~ Methods for Function getGene in Package
  'cummeRbund' ~~
getGenes            getGenes
getGenes-methods    ~~ Methods for Function getGenes in Package
  'cummeRbund' ~~
getLevels           getLevels
getLevels-methods   ~~ Methods for Function getLevels in Package
  'cummeRbund' ~~
length-methods      ~~ Methods for Function length in Package
  'base' ~~
makeprobs           Transform a matrix into probabilities by
                    columns
readCufflinks       readCufflinks
samples             Get sample list from CuffData object
samples-methods     ~~ Methods for Function samples in Package
  'cummeRbund' ~~
shannon.entropy     Shannon entropy

```

Further information is available in the following vignettes:

`cummeRbund-manual` An R package for visualization and analysis of Cufflinks high-throughput sequencing data (source,

~~ An overview of how to use the package, including the most important ~~ functions ~~

### Author(s)

L. Goff, C. Trapnell

Maintainer: Loyal A. Goff <lgoff@csail.mit.edu>

### References

~~ Literature or other references for background information ~~

---

`addFeatures`*addFeatures*

---

**Description**

Adds a data.frame of features to a the SQLite backend database.

**Usage**

```
## S4 method for signature 'CuffSet'  
addFeatures(object, features, level="genes", ...)
```

**Arguments**

<code>object</code>	An object of class ('CuffSet' or 'CuffData')
<code>features</code>	A data.frame of features to add. 1st column <b>MUST</b> contain ids (ie. <code>gene_id</code> for 'gene' features, <code>isoform_id</code> for 'isoform' features, etc)
<code>level</code>	One of <code>c('genes','isoforms','TSS','CDS')</code> to indicate which type of features you are being added, and to what data-level.
<code>...</code>	Additional arguments.

**Details**

None

**Value**

None

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
#None yet.
```

---

 createDB

*createDB*


---

**Description**

This should not be called directly by the user.

**Usage**

```
createDB(dbFname = "cuffData.db", driver = "SQLite")
```

**Arguments**

dbFname	File name for backend database (by default this is 'cuffData.db'). If you change this value, be sure to point to the new file with every call to 'readCufflinks'.
driver	DB driver for backend database (only SQLite supported at this time.)

**Details**

Should not be called directly

**Value**

Creates database backend file

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

---

 csBoxplot

*csBoxplot*


---

**Description**

Creates a ggplot2 plot object with a geom\_box layer displaying summary statistics for FPKM values across samples (x).

**Usage**

```
## S4 method for signature 'CuffData'
csBoxplot(object, logMode=TRUE, pseudocount=0.0001, ...)
```

**Arguments**

object	An object of class CuffData.
logMode	A logical argument to log10 -transform FPKM values.
pseudocount	Value added to FPKM to avoid log-transform issues.
...	Additional arguments to csBoxplot

**Details**

None

**Value**

A ggplot2 plot object with a geom\_box layer.

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Read cufflinks data and create CuffSet object
genes<-a@genes #CuffData object for all genes
csBoxplot(genes)
```

---

csCluster

*csCluster*

---

**Description**

Returns a ggplot2 plot object with geom\_line layer plotting FPKM values over conditions faceted by k-means clustering clusters. (Euclidean). This is very crude at this point. This does not return any of the clustering information directly, but if you want it, you can retrieve it from the ggplot object returned.

**Usage**

```
## S4 method for signature 'CuffFeatureSet'
csCluster(object,k,pseudocount=1,...)
```



**Arguments**

object	An object of class CuffFeatureSet.
k	Number of pre-defined clusters to attempt to find.
pseudocount	Value added to FPKM to avoid log-transform issues.
...	Additional arguments to pam.

**Details**

Uses 'kmeans' function.

**Author(s)**

Loyal A. Goff

**Source**

None

**References**

None.

**Examples**

```
data(sampleData)
csCluster(sampleGeneSet, 4)
```

---

csClusterPlot

*csClusterPlot*


---

**Description**

Replaces the default plotting behavior of the old csCluster. Takes as an argument the output of csCluster and plots expression profiles of features facet by cluster.

**Usage**

```
csClusterPlot(clustering, pseudocount=1.0, drawSummary=TRUE, sumFun=mean_cl_boot)
```

**Arguments**

clustering	The output of csCluster. (Must be the output of csCluster. Only this data format contains the necessary information for csClusterPlot.)
pseudocount	Value added to FPKM to avoid log transformation issues.
drawSummary	Logical value whether or not to draw a summary line for each cluster (by default this is the cluster mean)
sumFun	Summary function used to by drawSummary (default: mean_cl_boot)

**Details**

This replaces the default plotting behavior of the old `csCluster()` method. This was necessary so as to preserve the cluster information obtained by `csCluster` in a stable format. The output of `csClusterPlot` is a `ggplot2` object of `expressionProfiles` faceted by cluster ID.

**Value**

A `ggplot2` object of `expressionProfiles` faceted by cluster ID.

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
data(sampleData)
myClustering<-csCluster(sampleGeneSet,k=4)
csClusterPlot(myClustering)
```

---

csDendro

*csDendro*

---

**Description**

Creates a grid graphics plot of a dendrogram of Jensen-Shannon distances between conditions of a `CuffFeatureSet` or `CuffGeneSet` object.

**Usage**

```
## S4 method for signature 'CuffFeatureSet'
csDendro(object,logMode=T,pseudocount=1)
## S4 method for signature 'CuffData'
csDendro(object,logMode=T,pseudocount=1)
```

**Arguments**

<code>object</code>	An object of class <code>'CuffFeatureSet'</code> or <code>'CuffGeneSet'</code>
<code>logMode</code>	A logical argument to <code>log10</code> -transform FPKM values prior to plotting.
<code>pseudocount</code>	Value to be added to FPKM for appropriate log transformation and clustering. (Avoids zero-based errors)

**Details**

None

**Value**

Returns a dendrogram object and plots that object by default.

**Note**

None

**Author(s)**

Loyal A. Goff and Cole Trapnell

**References**

None.

**Examples**

```
data(sampleData)
csDendro(sampleGeneSet)
```

---

csDensity

*Density plot of CuffData*


---

**Description**

Creates a smoothed density plot, by sample, for log<sub>10</sub> FPKM values from a cuffdiff run.

**Usage**

```
## S4 method for signature 'CuffData'
csDensity(object, logMode=TRUE, pseudocount=1.0, labels, features=FALSE, ...)
## S4 method for signature 'CuffFeatureSet'
csDensity(object, logMode=TRUE, pseudocount=1.0, labels, features=FALSE, ...)
```

**Arguments**

object	An object of class CuffData.
logMode	A logical value of whether or not to log <sub>10</sub> -transform FPKM values. By default this is TRUE.
pseudocount	Pseudocount value added to FPKM to avoid errors in log-transformation of true zero values.
labels	A list of tracking_id values or gene_short_name values used for 'callout' points on the density plot for reference. (Not implemented yet).
features	Will include all fields from 'features' slot in returned ggplot object. Useful for further manipulations of plot object using feature-level attributes (e.g. gene_type, class_code, etc)
...	Additional arguments

**Details**

Creates a density plot, by sample, for log<sub>10</sub>-transformed FPKM values from a cuffdiff run.

**Value**

A ggplot2 plot object

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Create CuffSet object from sample data
genes<-a@genes #Create CuffData object for all 'genes'
d<-csDensity(genes) #Create csDensity plot
d #Render plot
```

---

csHeatmap

*csHeatmap*

---

**Description**

Creates a ggplot plot object with a geom\_tile layer of FPKM values per feature and sample.

**Usage**

```
## S4 method for signature 'CuffFeatureSet'
csHeatmap(object, rescaling='none', clustering='none', labCol=T, labRow=T, logMode=T, pseudocount
border=FALSE, heatscale= c(low='darkred',mid='orange',high='white'), heatMidpoint=NULL, fullname
```

**Arguments**

object	An object of class 'CuffFeatureSet' or 'CuffGeneSet'
rescaling	Rescaling can either be 'row' or 'column' OR you can pass rescale a function that operates on a matrix to do your own rescaling. Default is 'none'.
clustering	Clustering can either be 'row','column','none', or 'both', in which case the appropriate indices are re-ordered based on the pairwise Jensen-Shannon distance of FPKM values. Alternatively you can pass your own clustering function so long as the returned value is a re-ordered matrix.
labCol	A logical argument to display column labels.
labRow	A logical argument to display row labels.
logMode	A logical argument to log10-transform FPKM values prior to plotting.
pseudocount	Value to be added to FPKM for appropriate log transformation and clustering. (Avoids zero-based errors)
border	A logical argument to draw border around plot.

heatscale	A list with min length=2, max length=3 that detail the low,mid,and high colors to build the color scale.
heatMidpoint	Value for midpoint of color scale.
fullnames	A logical value whether to use 'fullnames' (concatenated gene_short_name and gene_id) for rows in heatmap. Default [ TRUE ].
...	Additional arguments to csHeatmap

**Details**

None

**Value**

A ggplot2 plot object with a geom\_tile layer to display FPKM values by sample (x) and feature (y)

**Note**

None

**Author(s)**

Loyal A. Goff and Cole Trapnell

**References**

None.

**Examples**

```
data(sampleData)
csHeatmap(sampleGeneSet)
```

---

csScatter

*Scatter Plot*


---

**Description**

A scatter plot comparing the FPKM values from two samples in a cuffdiff run.

**Usage**

```
## S4 method for signature 'CuffData'
csScatter(object, x, y, logMode=TRUE, pseudocount=1.0, labels, smooth=FALSE, colorByStatus = FALSE)
```

**Arguments**

object	An object of class ('CuffData','CuffFeatureSet')
x	Sample name for x axis
y	Sample name for y axis
logMode	Logical argument to log2-transform data (default: T)
pseudocount	Value to add to zero FPKM values for log transformation (default: 0.0001)
smooth	Logical argument to add a smooth-fit regression line
labels	A list of tracking_ids or gene_short_names that will be 'callout' points in the plot for reference. Useful for finding genes of interest in the field. Not implemented yet.
colorByStatus	A logical argument whether or not to color the points by 'significant' Y or N. [Default = FALSE]
drawRug	A logical argument whether or not to draw the rug for x and y axes [Default = TRUE]
...	Additional arguments to csScatter

**Details**

None

**Value**

ggplot object with geom\_point and geom\_rug layers

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Create CuffSet object from sample data
genes<-a@genes #Create CuffData object for all genes
s<-csScatter(genes,'hESC','Fibroblasts',smooth=TRUE) #Create plot object
s #render plot object
```

---

csSpecificity	<i>csSpecificity</i>
---------------	----------------------

---

**Description**

Returns a matrix of 'Specificity scores' (S) defined as  $1 - \text{JSD}(p_g, q_i)$  where  $p_g$  is the Log10+1 expression profile of a gene  $g$  across all conditions  $j$ , collapsed into a probability distribution, and  $q_i$  is the unit vector of 'perfect expression' in a given condition  $i$ .

**Usage**

```
## S4 method for signature 'CuffFeatureSet'
csSpecificity(object, logMode=T, pseudocount=1, relative=FALSE, ...)
## S4 method for signature 'CuffData'
csSpecificity(object, logMode=T, pseudocount=1, relative=FALSE, ...)
```

**Arguments**

object	An object of class CuffFeatureSet, CuffGeneSet, or CuffData.
logMode	A logical argument to log10-transform FPKM values prior to plotting.
pseudocount	Value to be added to FPKM for appropriate log transformation and clustering. (Avoids zero-based errors)
relative	A logical argument that when TRUE, will scale the S values from 0-1 by dividing by max(S)
...	Additional arguments to fpkmMatrix.

**Details**

None

**Author(s)**

Loyal A. Goff

**Source**

None

**References**

None.

**Examples**

```
data(sampleData)
csSpecificity(sampleGeneSet)
```

csVolcano

*Volcano Plot***Description**

Creates a volcano plot of log fold change in expression vs  $-\log(\text{pval})$  for a pair of samples (x,y)

**Usage**

```
## S4 method for signature 'CuffData'
csVolcano(object, x, y, features=FALSE, xlimits = c(-20, 20), ...)
```

**Arguments**

object	An object of class CuffData, CuffFeatureSet, or CuffGeneSet
x	Sample name from 'samples' table for comparison
y	Sample name from 'samples' table for comparison
features	Will include all fields from 'features' slot in returned ggplot object. Useful for further manipulations of plot object using feature-level attributes (e.g. gene_type, class_code, etc)
xlimits	Set boundaries for x limits to avoid infinity plotting errors. [Default c(-20,20)]
...	Additional arguments

**Details**

This creates a 'volcano' plot of fold change vs. significance for a pairwise comparison of genes or features across two different samples.

**Value**

A ggplot2 plot object

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Create CuffSet object
genes<-a@genes #Create cuffData object for all genes
v<-csVolcano(genes,"hESC","Fibroblasts") # Volcano plot of all genes for conditions x='hESC' and y='Fibroblasts'
v #print plot
```



---

CuffData-class	Class "CuffData"
----------------	------------------

---

### Description

A 'pointer' class for all information (FPKM, annotation, differential expression) for a given feature type (genes, isoforms, TSS, CDS). The methods for this function communicate directly with the SQL backend to present data to the user.

### Objects from the Class

Objects can be created by calls of the form `new("CuffData", DB, tables, filters, type, idField, ...)`.

### Slots

DB: Object of class "SQLiteConnection" ~~  
tables: Object of class "list" ~~  
filters: Object of class "list" ~~  
type: Object of class "character" ~~  
idField: Object of class "character" ~~

### Methods

**dim** signature(x = "CuffData"): ...  
**getFeatures** signature(object = "CuffData"): ...

### Note

None

### Author(s)

Loyal A. Goff

### References

None

### See Also

None

### Examples

```
showClass("CuffData")
```

---

CuffDist-class      *Class "CuffDist"*

---

**Description**

A 'pointer' class to information relative to the distribution-level tests (promoters, splicing, and relative CDS usage)

**Objects from the Class**

Objects can be created by calls of the form `new("CuffDist", DB, table, type, idField, ...)`.

**Slots**

DB: Object of class "SQLiteConnection" ~~  
table: Object of class "character" ~~  
type: Object of class "character" ~~  
idField: Object of class "character" ~~

**Methods**

**dim** signature(x = "CuffDist"): ...  
**samples** signature(x = "CuffDist"): ...

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**See Also**

None

**Examples**

```
showClass("CuffDist")
```

---

CuffFeature-class      *Class "CuffFeature"*

---

### Description

A 'data' container class for all FPKM, annotation, and differential expression data for a single feature (gene, isoform, TSS, or CDS).

### Objects from the Class

Objects can be created by calls of the form `new("CuffFeature", annotation, fpkm, diff, ...)`.

### Slots

**annotation:** Object of class "data.frame" ~~

**fpkm:** Object of class "data.frame" ~~

**diff:** Object of class "data.frame" ~~

### Methods

**fpkmMatrix** signature(object="CuffFeature"): ...

**length** signature(x = "CuffFeature"): ...

### Accessors

**annotation** signature(object="CuffFeature"): Access @annotation slot

**diffData** signature(object="CuffFeature"): Access @diff slot

### Note

'CuffGene' is a superclass of 'CuffFeature' that links gene information for a given gene with all isoform-, TSS-, and CDS-level data for the given gene.

### Author(s)

Loyal A. Goff

### References

None

### See Also

[CuffGene](#)

### Examples

```
showClass("CuffFeature")
```

---

CuffFeatureSet-class    *Class "CuffFeatureSet"*

---

### Description

A 'data' container class for all FPKM, annotation, and differential expression data for a set of features (genes, isoforms, TSS, CDS).

### Objects from the Class

Objects can be created by calls of the form `new("CuffFeatureSet", annotation, fpkm, diff, ...)`.

### Slots

**annotation:** Object of class "data.frame" ~~

**fpkm:** Object of class "data.frame" ~~

**diff:** Object of class "data.frame" ~~

### Methods

**diffData** signature(object = "CuffFeatureSet"): ...

**featureNames** signature(object = "CuffFeatureSet"): ...

**features** signature(object = "CuffFeatureSet"): ...

**fpkmMatrix** signature(object = "CuffFeatureSet"): ...

**samples** signature(object = "CuffFeatureSet"): ...

**length** signature(object = "CuffFeatureSet"): ...

### Accessors

**annotation** signature(object="CuffFeatureSet"): Access @annotation slot

### Note

None.

### Author(s)

Loyal A. Goff

### References

None.

### See Also

[CuffGeneSet](#)

### Examples

```
showClass("CuffFeatureSet")
```

---

CuffGene-class	Class "CuffGene"
----------------	------------------

---

### Description

A 'data' container class for all FPKM, annotation, and differential expression Data (as well as for all linked features) for a given gene.

### Objects from the Class

Objects can be created by calls of the form `new("CuffGene", id, isoforms, TSS, CDS, promoters, splicing, r`

### Slots

id: Object of class "character" ~~  
 isoforms: Object of class "CuffFeature" ~~  
 TSS: Object of class "CuffFeature" ~~  
 CDS: Object of class "CuffFeature" ~~  
 promoters: Object of class "CuffFeature" ~~  
 relCDS: Object of class "CuffFeature" ~~  
 splicing: Object of class "CuffFeature" ~~  
 annotation: Object of class "data.frame" ~~  
 fpkm: Object of class "data.frame" ~~  
 diff: Object of class "data.frame" ~~

### Extends

Class "[CuffFeature](#)", directly.

### Methods

No methods defined with class "CuffGene" in the signature.

### Accessors

**genes** signature(object = "CuffGene"): Access @genes slot  
**isoforms** signature(object = "CuffGene"): Access @isoforms slot  
**TSS** signature(object = "CuffGene"): Access @TSS slot  
**CDS** signature(object = "CuffGene"): Access @CDS slot  
**promoters** signature(object = "CuffGene"): Access @CDS slot  
**relCDS** signature(object = "CuffGene"): Access @CDS slot  
**splicing** signature(object = "CuffGene"): Access @CDS slot

### Note

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**See Also**

[CuffFeature](#)

**Examples**

```
showClass("CuffGene")
```

---

CuffGeneSet-class      *Class "CuffGeneSet"*

---

**Description**

A 'data' container class for all FPKM, annotation, and differential expression data (an associated features) for a given set of genes.

**Objects from the Class**

Objects can be created by calls of the form `new("CuffGeneSet", annotation, fpkm, diff, ...)`.

**Slots**

ids: Object of class "character" ~~  
 isoforms: Object of class "CuffFeatureSet" ~~  
 TSS: Object of class "CuffFeatureSet" ~~  
 CDS: Object of class "CuffFeatureSet" ~~  
 promoters: Object of class "CuffFeatureSet" ~~  
 reICDS: Object of class "CuffFeatureSet" ~~  
 splicing: Object of class "CuffFeatureSet" ~~  
 annotation: Object of class "data.frame" ~~  
 fpkm: Object of class "data.frame" ~~  
 diff: Object of class "data.frame" ~~

**Extends**

Class "[CuffFeatureSet](#)", directly.

**Methods**

No methods defined with class "CuffGeneSet" in the signature.

**Accessors**

**genes** signature(object = "CuffGeneSet"): Access @genes slot  
**isoforms** signature(object = "CuffGeneSet"): Access @isoforms slot  
**TSS** signature(object = "CuffGeneSet"): Access @TSS slot  
**CDS** signature(object = "CuffGeneSet"): Access @CDS slot  
**promoters** signature(object = "CuffGeneSet"): Access @promoters slot  
**relCDS** signature(object = "CuffGeneSet"): Access @relCDS slot  
**splicing** signature(object = "CuffGeneSet"): Access @splicing slot

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**See Also**

[CuffFeatureSet](#)

**Examples**

```
showClass("CuffGeneSet")
```

---

CuffSet-class

*Class "CuffSet"*

---

**Description**

A 'pointer' class to connect to, and retrieve data from the SQLite backend database.

**Objects from the Class**

Objects can be created by calls of the form `new("CuffSet", DB, conditions, genes, isoforms, TSS, CDS, promoters)`. Available methods are primary accessors to retrieve CuffGeneSet or CuffGene objects for manipulation.

**Slots**

DB: Object of class "SQLiteConnection" ~~  
conditions: Object of class "data.frame" ~~  
genes: Object of class "CuffData" ~~  
isoforms: Object of class "CuffData" ~~  
TSS: Object of class "CuffData" ~~  
CDS: Object of class "CuffData" ~~  
promoters: Object of class "CuffDist" ~~  
splicing: Object of class "CuffDist" ~~  
relCDS: Object of class "CuffDist" ~~

**Methods**

[ signature(x = "CuffSet"): ...

**Accessors**

**genes** signature(object = "CuffSet"): Access @genes slot  
**isoforms** signature(object = "CuffSet"): Access @isoforms slot  
**TSS** signature(object = "CuffSet"): Access @TSS slot  
**CDS** signature(object = "CuffSet"): Access @CDS slot  
**promoters** signature(object = "CuffSet"): Access @promoters slot  
**splicing** signature(object = "CuffSet"): Access @splicing slot  
**relCDS** signature(object = "CuffSet"): Access @relCDS slot

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**See Also**

None.

**Examples**

```
showClass("CuffSet")
```



---

diffData	<i>Differential comparison data</i>
----------	-------------------------------------

---

**Description**

An accessor method to retrieve differential expression data from a 'CuffData', 'CuffFeatureSet', or 'CuffFeature' object

**Usage**

```
## S4 method for signature 'CuffData'  
diffData(object, x, y, features=FALSE)
```

**Arguments**

object	An object of class ('CuffData' or 'CuffFeatureSet')
x	Optional, if x and y are both missing, data for all pairwise differential comparisons are returned, otherwise if x and y are sample names from the 'samples' table, than only differential data pertaining to those two samples are returned.
y	See 'x'
features	A logical value that returns all feature-level data as part of data.frame when true. object must be of class 'CuffData'.
...	Additional arguments.

**Details**

None

**Value**

A data.frame object

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
data(sampleData)  
diff<-diffData(sampleGeneSet) #returns a dataframe of differential expression data from sample CuffGeneSet
```

---

`distValues`*distValues*

---

**Description**

Returns a data.frame of distribution-level test values from a CuffDist object (@promoters, @splicing, @relCDS)

**Usage**

```
## S4 method for signature 'CuffDist'  
distValues(object)
```

**Arguments**

object	An object of class 'CuffDist'
...	Additional arguments to distValues

**Details**

None

**Value**

Returns a data.frame of distribution-level test values.

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**See Also**

None

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) # Read cufflinks data and create CuffSet obj.  
distValues(a@promoters) # returns data.frame of values from CuffDist object in slot 'promoters'
```

---

expressionBarplot      *Barplot*

---

### Description

A barplot of FPKM values with confidence intervals for a given gene, set of genes, or features of a gene (e.g. isoforms, TSS, CDS, etc).

### Usage

```
## S4 method for signature 'CuffFeatureSet'
expressionBarplot(object, logMode=TRUE, pseudocount=1.0, showErrorbars=TRUE, showStatus=TRUE, ..
```

### Arguments

object	An object of class ('CuffFeatureSet','CuffGeneSet','CuffFeature','CuffGene')
logMode	A logical value whether or not to draw y-axis on log10 scale. Default = FALSE.
pseudocount	Numerical value added to each FPKM during log-transformation to avoid errors.
showErrorbars	A logical value whether or not to draw error bars. Default = TRUE
showStatus	A logical value whether or not to draw visual queues for quantification status of a given gene:condition. Default = TRUE
...	Additional arguments.

### Details

None

### Value

A ggplot2 plot object

### Note

Need to implement logMode and features for this plotting method.

### Author(s)

Loyal A. Goff

### References

None

### Examples

```
data(sampleData)
PINK1 # sample CuffFeature object
expressionBarplot(PINK1) #Barplot of PINK1 FPKM values
expressionBarplot(PINK1@isoforms) #Barplot of PINK1 FPKM values faceted by isoforms
```

---

expressionPlot                      *Expression Plot*

---

**Description**

A line plot (optionally with confidence intervals) detailing FPKM expression levels across conditions for a given gene(s) or feature(s)

**Usage**

```
## S4 method for signature 'CuffFeature'
expressionPlot(object, logMode=FALSE, pseudocount=1.0, drawSummary=FALSE, sumFun=mean_cl_boot, s
```

**Arguments**

object	An object of class ('CuffFeature' or 'CuffGene')
logMode	A logical value to draw y-axis (FPKM) on log-10 scale. Default = FALSE.
pseudocount	A numeric value added to FPKM to avoid errors on log-10 transformation.
drawSummary	A logical value. Draws a 'summary' line with mean FPKM values for each condition.
sumFun	Function used to determine values for summary line. Default = mean_cl_boot
showErrorbars	A logical value whether or not to draw error bars.
showStatus	A logical value whether or not to draw visual queues for quantification status of a given gene:condition. Default = TRUE
...	Additional arguments

**Details**

None

**Value**

A ggplot2 plot object

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
data(sampleData)
PINK1 # sample CuffFeature object
expressionPlot(PINK1) #Line plot of PINK1 FPKM values
expressionPlot(PINK1@isoforms) #Line plot of PINK1 FPKM values faceted by isoforms
```

---

featureNames	<i>Feature names</i>
--------------	----------------------

---

**Description**

Retrieve a vector of feature names from a 'CuffData' or 'CuffFeatureSet' object

**Usage**

```
## S4 method for signature 'CuffData'  
featureNames(object)
```

**Arguments**

object            An object of class ('CuffData' or 'CuffFeatureSet')

**Details**

None

**Value**

A list of feature names

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
data(sampleData)  
featureNames(sampleGeneSet)
```

---

features

*Features*

---

### Description

Returns a data frame of features from a CuffData, CuffFeatureSet, or CuffFeature object

### Usage

```
## S4 method for signature 'CuffData'  
features(object)
```

### Arguments

object            An object of class ('CuffData', 'CuffFeatureSet', or 'CuffFeature')

### Details

If features have been added to cuffData via addFeatures() then these will be presented as well.

### Value

A data.frame of feature-level information

### Note

None

### Author(s)

Loyal A. Goff

### References

None

### Examples

```
data(sampleData)  
features(sampleGeneSet)
```

---

findSimilar	<i>findSimilar</i>
-------------	--------------------

---

**Description**

Returns a CuffGeneSet containing n genes with the most similar expression profiles to gene/profile x.

**Usage**

```
## S4 method for signature 'CuffSet'  
findSimilar(object, x, n)
```

**Arguments**

object	A object of class 'CuffSet'
x	A 'gene_id' or 'gene_short_name' from which to look up an expression profile OR a vector of expression values to compare all genes (vector must have same length and order of 'samples')
n	Number of similar genes to return

**Details**

By default, returns a CuffGeneSet object with n similar genes. This may change in the future.

**Value**

A CuffGeneSet object of n most similar genes to x.

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund"))  
mySimilarGenes<-findSimilar(a,"PINK1",10)
```

fpkm-methods

*Retrieve FPKM values*

---

**Description**

Returns a data.frame from @FPKM slot

**Details**

Returns a data.frame of FPKM values.

**Value**

A data.frame of FPKM-level values for a set of features.

**Methods**

```
signature(object = "CuffData")  
signature(object = "CuffFeature")  
signature(object = "CuffFeatureSet")
```

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
data(sampleData)  
fpkm(PINK1)
```

---

fpkmMatrix*fpkmMatrix*

---

**Description**

Retrieve FPKM values as gene by condition matrix

**Usage**

```
## S4 method for signature 'CuffData'  
fpkmMatrix(object)
```



**Arguments**

object            An object of class ('CuffData', 'CuffFeatureSet', 'CuffGeneSet', 'CuffGene', or 'Cuff-Feature')

**Details**

None.

**Value**

A feature x condition matrix of FPKM values.

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
data(sampleData)
fpmMatrix(sampleGeneSet)
```

---

getGene

*getGene*

---

**Description**

Primary accessor from a CuffSet object to retrieve all related information for 1 (one) given gene, indexed by gene\_id or gene\_short\_name.

**Usage**

```
## S4 method for signature 'CuffSet'
getGene(object, geneId, sampleIdList=NULL)
```

**Arguments**

object            An object of class 'CuffSet' (Primary 'pointer' object for Cufflinks data).

geneId            A character string to identify which gene for which you would like to retrieve all information.

sampleIdList     A vector of sample names used to subset or re-order samples in returned object

**Details**

None.

**Value**

Returns a CuffGene object containing all related information for a given gene\_id or gene\_short\_name

**Note**

Right now, this does not return an error if it cannot find a gene. (this is probably a bad thing...)

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Read cufflinks data and create master CuffSet
myGene<-getGene(a,"PINK1") # Retrieve all information for gene "PINK1"
```

---

getGenes

*getGenes*

---

**Description**

Primary accessor from a CuffSet object to retrieve all related information for >1 (MANY) given genes, indexed by gene\_id or gene\_short\_name.

**Usage**

```
## S4 method for signature 'CuffSet'
getGenes(object, geneIdList, sampleIdList=NULL)
```

**Arguments**

object	An object of class 'CuffSet' (Primary 'pointer' object for Cufflinks data).
geneIdList	A vector of gene_ids or gene_short_namesto identify which genes for which you would like to retrieve all information.
sampleIdList	A vector of sample names used to subset or re-order samples in returned object

**Details**

None.

**Value**

Returns a CuffGeneSet object containing all related information for a given set of gene\_id or gene\_short\_name values

**Note**

Right now, this does not return an error if it cannot find a gene. (this is probably a bad thing...)

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Read cufflinks data and create master CuffData
data(sampleData)
sampleIDs
myGene<-getGenes(a,sampleIDs) # Retrieve all information for a set of 20 'sample' genes.
```

---

*getLevels*

*getLevels*

---

**Description**

Returns a list of samples as levels. This should not be called directly by user.

**Usage**

```
## S4 method for signature 'CuffData'
getLevels(object)
```

**Arguments**

object            An object of class 'CuffData' or 'CuffFeatureSet' or 'CuffFeature'

**Details**

For internal usage only.

**Value**

A vector of sample names as factors.

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

---

`getSig`*getSig*

---

**Description**

Returns the identifiers of significant genes in a vector format.

**Usage**

```
## S4 method for signature 'CuffSet'  
getSig(object,x,y,alpha=0.05,level='genes')
```

**Arguments**

<code>object</code>	A CuffSet object (e.g. cuff)
<code>x</code>	Optional argument to restrict significance results to one pairwise comparison. Must be used with a 'y' argument to specify the other half of the pair.
<code>y</code>	See x.
<code>alpha</code>	An alpha value by which to filter multiple-testing corrected q-values to determine significance
<code>level</code>	Feature level to be queried for significance (must be one of c('genes','isoforms','TSS','CDS'))

**Details**

This is a convenience function to quickly retrieve vectors of identifiers for genes or features that were determined to be significantly regulated between conditions by cuffdiff. This function only returns tracking IDs that correspond to tests with an 'OK' status from cuffdiff, NOTEST values are ignored. By default getSig returns a vector of IDs for all pairwise comparisons together. If you specify an 'x' AND 'y' values as sample names, then only the features that are significant in that particular pairwise comparison are reported, after appropriate multiple testing correction of output p-values.

**Value**

A vector of feature IDs.

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Read cufflinks data in sample directory and
mySig<-getSig(a,x='hESC',y='Fibroblasts',alpha=0.05,level='genes')
head(mySig)
```

---

getSigTable	<i>getSigTable</i>
-------------	--------------------

---

**Description**

Returns the identifiers of significant genes in a test table - like format.

**Usage**

```
## S4 method for signature 'CuffSet'
getSigTable(object,alpha=0.05,level='genes')
```

**Arguments**

object	A CuffSet object (e.g. cuff)
alpha	An alpha value by which to filter multiple-testing corrected q-values to determine significance
level	Feature level to be queried for significance (must be one of c('genes','isoforms','TSS','CDS'))

**Details**

This is a convenience function to quickly retrieve lists of identifiers for genes or features that were determined to be significantly regulated between conditions by cuffdiff. This function only returns tracking IDs that correspond to tests with an 'OK' status from cuffdiff, NOTEST values are ignored or reported as NA. By default getSig returns a table of genes x conditions, where the column names represent the pairwise comparisons from the cuffdiff analysis. The values in the table are 1 for features that are significant for this comparison and 0 for genes that are not, any failed tests are reported as <NA>. Only includes the features that are significant in at least one comparison.

**Value**

A data.frame of pairwise test results.

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Read cufflinks data in sample directory and
mySigTable<-getSigTable(a,alpha=0.05,level='genes')
head(mySigTable)
```

---

 JSdist

*Jensen-Shannon distance on columns*


---

**Description**

JSdist takes a matrix of expression probabilities (calculated directly or output from makeprobs()) and returns a dist object of the pairwise Jensen-Shannon distances between columns

**Usage**

```
JSdist(mat)
```

**Arguments**

mat                    A matrix of expression probabilities (e.g. from makeprobs())

**Details**

Returns pairwise Jensen-Shannon distance (in the form of a dist object) for a matrix of probabilities (by column)

**Value**

A dist object of pairwise J-S distances between columns.

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
mat<-matrix(sample(1:50,50),10)
probs<-makeprobs(mat)
js.distance<-JSdist(probs)
```

---

JSdistFromP	<i>Jensen-Shannon distance on rows from a pre-defined vector of probabilities</i>
-------------	---

---

**Description**

JSdist takes a matrix of expression probabilities (calculated directly or output from makeprobs()) and returns a matrix of Jensen-Shannon distances between individual rows and a specific vector of probabilities (q)

**Usage**

```
JSdistFromP(mat,q)
```

**Arguments**

mat	A matrix of expression probabilities (e.g. from makeprobs())
q	A vector of expression probabilities.

**Details**

Returns Jensen-Shannon distance for each row of a matrix of probabilities against a provided probability distribution (q)

**Value**

A vector of JS distances

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
mat<-matrix(sample(1:50,50),10)
q<-c(100,4,72,8,19)
q<-q/sum(q)
js.distance<-JSdistFromP(mat,q)
```

JSdistVec

*JSdistVec*

---

**Description**

Returns the Jensen-Shannon Distance (square root of JS divergence) between two probability vectors.

**Usage**

```
JSdistVec(p, q)
```

**Arguments**

p	A vector of probabilities
q	A vector of probabilities

**Details**

Should not be called directly by user.

**Value**

Returns the JS distance as a numeric

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
p<-sample(1:5000,20)
q<-sample(1:5000,20)
p<-makeprobsvec(p)
q<-makeprobsvec(q)
JSdistVec(p,q)
```



---

`makeprobs`*Transform a matrix into probabilities by columns*

---

**Description**

This function takes a matrix of expression values (must be greater than 0) and returns a matrix of probabilities by column. This is a required transformation for the Jensen-Shannon distance which is a metric that operates on probabilities.

**Usage**

```
makeprobs(a)
```

**Arguments**

`a` A matrix of expression values (values must be greater than 0).

**Details**

To make a matrix of probabilities by row, use `t()` to transpose prior to calling `makeprobs`.

**Value**

A matrix of expression probabilities by column.

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
myMat<-matrix(sample(1:50,50),10)
probs<-makeprobs(myMat)
```

makeprobsvec      *makeprobsvec*

---

**Description**

Sums a vector of numerics and divides by the sum

**Usage**

```
makeprobsvec(p)
```

**Arguments**

p                      A vector of numerics

**Details**

None

**Value**

A vector of probabilities

**Note**

Should not be called directly by user.

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
p<-sample(1:5000,20)
makeprobsvec(p)
```

---

MAplot

*MAplot*

---

### Description

Creates an M vs A plot (Avg intensity vs log ratio) for a given pair of conditions across all fpkms

### Usage

```
## S4 method for signature 'CuffData'  
MAplot(object, x, y, logMode=T, pseudocount=1, smooth=F)
```

### Arguments

object	An object of class 'CuffData'.
x	Sample name from 'samples' table for comparison
y	Sample name from 'samples' table for comparison
logMode	A logical argument to log10-transform FPKM values prior to plotting.
pseudocount	Value to be added to FPKM for appropriate log transformation and clustering. (Avoids zero-based errors)
smooth	Logical argument whether or not to draw a smoothed line fit through data.

### Details

None

### Value

Returns a ggplot MvsA plot object.

### Note

None

### Author(s)

Loyal A. Goff and Cole Trapnell

### References

None.

### Examples

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Create CuffSet object from sample data  
genes<-a@genes #Create CuffData object for all 'genes'  
d<-MAplot(genes,'hESC','Fibroblasts') #Create csDensity plot  
d #Render plot
```

PINK1

*PINK1*

---

**Description**

A sample 'CuffGene' dataset

**Usage**

```
data(sampleData)
```

**Format**

PINK1 is a CuffGene object (extends CuffFeature) with all sample gene-, isoform-, TSS-, and CDS-level data for the gene 'PINK1'.

**Details**

Sample CuffGene data for gene 'PINK1'

**Source**

None

**References**

None

**Examples**

```
data(sampleData)  
PINK1
```

---

readCufflinks*readCufflinks*

---

**Description**

This initializes the backend SQLite table and provides a DB connection for all downstream data analysis.

**Usage**

```
readCufflinks(dir = getwd(), dbFile = "cuffData.db", geneFPKM = "genes.fpkm_tracking", geneDiff
```

**Arguments**

<code>dir</code>	Directory in which all CuffDiff output files can be located. Defaults to current working directory.
<code>dbFile</code>	Name of backend database. Default is 'cuffData.db'
<code>geneFPKM</code>	genes.fpkm_tracking file
<code>geneDiff</code>	gene_exp.diff file
<code>isoformFPKM</code>	isoforms.fpkm_tracking file
<code>isoformDiff</code>	isoform_exp.diff file
<code>TSSFPKM</code>	tss_groups.fpkm_tracking file
<code>TSSDiff</code>	tss_group_exp.diff file
<code>CDSFPKM</code>	cds.fpkm_tracking file
<code>CDSExpDiff</code>	cds_exp.diff file
<code>CDSDiff</code>	cds.diff file (distribution tests on CDS)
<code>promoterFile</code>	promoters.diff file (distribution tests on promoters)
<code>splicingFile</code>	splicing.diff (distribution tests on isoforms)
<code>driver</code>	Driver for backend database. (Currently only "SQLite" is supported).
<code>rebuild</code>	A logical argument to rebuild database backend.
<code>...</code>	Additional arguments to readCufflinks

**Details**

This is the initialization function for the cummeRbund package. It creates the SQLite backend database, populates the data, and provides a connection object for all future interactions with the dataset. Once the initial build is complete, this function will default to using the database for all future sessions. **IMPORTANT:** - Each R session should begin with a call to readCufflinks to re-open the connection to the database. - Should any connectivity issues to the database arise, another call to readCufflinks should create a new connection object and repair any issue. - The database can always be rebuild (using rebuild=TRUE) from the original CuffDiff output files.

**Value**

A CuffSet object. A 'pointer' class that allows interaction with cufflinks/cuffdiff data via a SQLite database backend.

**Note**

None.

**Author(s)**

Loyal A. Goff

**References**

None.

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Read cufflinks data in sample directory and
```

sampleGeneSet

*sampleGeneSet*

---

**Description**

A sample CuffGeneSet data set for 20 genes.

**Usage**

```
data(sampleData)
```

**Format**

sampleGeneSet is a CuffGeneSet (extends CuffFeatureSet) object containing all sample gene-, isoform-, TSS-, and CDS-level data for 20 different genes. These data were derived from a toy set of hESC-vs-iPSC-vs-Fibroblast RNA-Seq expression data.

**Details**

None

**Source**

None

**References**

None

**Examples**

```
data(sampleData)
```

---

sampleIDs

*sampleIDs*

---

**Description**

A vector of gene\_ids used to create 'sampleGeneSet' example

**Usage**

```
data(sampleData)
```

**Format**

The format is: chr "sampleIDs"

**Details**

None

**Source**

None

**References**

None

**Examples**

```
data(sampleData)
```

---

samples

*Get sample list from CuffData object*

---

**Description**

Returns a list of sample names from a CuffData or CuffFeatureSet object

**Usage**

```
## S4 method for signature 'CuffData'  
samples(object)
```

**Arguments**

object            An object of class ('CuffData','CuffFeatureSet','CuffFeature')

**Details**

None

**Value**

A list of sample names

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**See Also**

None

**Examples**

```
a<-readCufflinks(system.file("extdata", package="cummeRbund")) #Create CuffSet object
samples(a@genes)
```

---

shannon.entropy	<i>Shannon entropy</i>
-----------------	------------------------

---

**Description**

Calculates the Shannon entropy for a probability distribution

**Usage**

```
shannon.entropy(p)
```

**Arguments**

p                    A vector of probabilities (must sum to ~1)

**Details**

None

**Value**

Returns a numeric value for the Shannon entropy of the supplied probability distribution

**Note**

None

**Author(s)**

Loyal A. Goff

**References**

None

**Examples**

```
x<-sample(1:500,50)
p<-x/sum(x)
shannon.entropy(p)
```



# Index

- \*Topic **JS**
  - JSdist, [38](#)
  - JSdistFromP, [39](#)
- \*Topic **Jensen-Shannon**
  - JSdist, [38](#)
  - JSdistFromP, [39](#)
- \*Topic **classes**
  - CuffData-class, [17](#)
  - CuffDist-class, [18](#)
  - CuffFeature-class, [19](#)
  - CuffFeatureSet-class, [20](#)
  - CuffGene-class, [21](#)
  - CuffGeneSet-class, [22](#)
  - CuffSet-class, [23](#)
- \*Topic **datasets**
  - csCluster, [8](#)
  - csSpecificity, [15](#)
  - PINK1, [44](#)
  - sampleGeneSet, [46](#)
  - sampleIDs, [46](#)
- \*Topic **distance**
  - JSdist, [38](#)
  - JSdistFromP, [39](#)
- \*Topic **heatmap**
  - csDendro, [10](#)
  - csHeatmap, [12](#)
  - MPlot, [43](#)
- \*Topic **metric**
  - JSdist, [38](#)
  - JSdistFromP, [39](#)
- \*Topic **package**
  - cummeRbund-package, [2](#)
- [, CuffSet-method (CuffSet-class), [23](#)
- addFeatures, [6](#)
- addFeatures, CuffData-method
  - (addFeatures), [6](#)
- addFeatures, CuffSet-method
  - (addFeatures), [6](#)
- annotation (CuffFeatureSet-class), [20](#)
- annotation, CuffFeature-method
  - (CuffFeature-class), [19](#)
- annotation, CuffFeatureSet-method
  - (CuffFeatureSet-class), [20](#)
- CDS (CuffSet-class), [23](#)
- CDS, CuffGene-method (CuffGene-class), [21](#)
- CDS, CuffGeneSet-method
  - (CuffGeneSet-class), [22](#)
- CDS, CuffSet-method (CuffSet-class), [23](#)
- createDB, [7](#)
- csBoxplot, [7](#)
- csBoxplot, CuffData-method (csBoxplot), [7](#)
- csCluster, [8](#)
- csCluster, CuffFeatureSet-method
  - (csCluster), [8](#)
- csClusterPlot, [9](#)
- csDendro, [10](#)
- csDendro, CuffData-method (csDendro), [10](#)
- csDendro, CuffFeatureSet-method
  - (csDendro), [10](#)
- csDensity, [11](#)
- csDensity, CuffData-method (csDensity),  
[11](#)
- csDensity, CuffFeatureSet-method
  - (csDensity), [11](#)
- csHeatmap, [12](#)
- csHeatmap, CuffFeatureSet-method
  - (csHeatmap), [12](#)
- csScatter, [13](#)
- csScatter, CuffData-method (csScatter),  
[13](#)
- csScatter, CuffFeatureSet-method
  - (csScatter), [13](#)
- csSpecificity, [15](#)
- csSpecificity, CuffData-method
  - (csSpecificity), [15](#)
- csSpecificity, CuffFeatureSet-method
  - (csSpecificity), [15](#)
- csVolcano, [16](#)
- csVolcano, CuffData-method (csVolcano),  
[16](#)
- csVolcano, CuffFeatureSet-method
  - (csVolcano), [16](#)
- CuffData-class, [17](#)
- CuffDist-class, [18](#)
- CuffFeature, [21](#), [22](#)
- CuffFeature-class, [19](#)

- CuffFeatureSet, 22, 23
- CuffFeatureSet-class, 20
- CuffGene, 19
- CuffGene-class, 21
- CuffGeneSet, 20
- CuffGeneSet-class, 22
- CuffSet-class, 23
- cummeRbund (cummeRbund-package), 2
- cummeRbund-package, 2
  
- diffData, 25
- diffData, CuffData-method (diffData), 25
- diffData, CuffFeature-method (CuffFeature-class), 19
- diffData, CuffFeatureSet-method (CuffFeatureSet-class), 20
- dim, CuffData-method (CuffData-class), 17
- dim, CuffDist-method (CuffDist-class), 18
- distValues, 26
- distValues, CuffDist-method (distValues), 26
  
- expressionBarplot, 27
- expressionBarplot, CuffFeature-method (expressionBarplot), 27
- expressionBarplot, CuffFeatureSet-method (expressionBarplot), 27
- expressionPlot, 28
- expressionPlot, CuffFeature-method (expressionPlot), 28
- expressionPlot, CuffFeatureSet-method (expressionPlot), 28
  
- featureNames, 29
- featureNames, CuffData-method (featureNames), 29
- featureNames, CuffFeatureSet-method (CuffFeatureSet-class), 20
- features, 30
- features, CuffData-method (features), 30
- features, CuffFeatureSet-method (CuffFeatureSet-class), 20
- findSimilar, 31
- findSimilar, CuffSet-method (findSimilar), 31
- fpkm (fpkm-methods), 32
- fpkm, CuffData-method (fpkm-methods), 32
- fpkm, CuffFeature-method (fpkm-methods), 32
- fpkm, CuffFeatureSet-method (fpkm-methods), 32
- fpkm-methods, 32
- fpkmMatrix, 32
- fpkmMatrix, CuffData-method (fpkmMatrix), 32
- fpkmMatrix, CuffFeature-method (CuffFeature-class), 19
- fpkmMatrix, CuffFeatureSet-method (CuffFeatureSet-class), 20
  
- genes (CuffSet-class), 23
- genes, CuffGene-method (CuffGene-class), 21
- genes, CuffGeneSet-method (CuffGeneSet-class), 22
- genes, CuffSet-method (CuffSet-class), 23
- getFeatures, CuffData-method (CuffData-class), 17
- getGene, 33
- getGene, CuffSet-method (getGene), 33
- getGenes, 34
- getGenes, CuffSet-method (getGenes), 34
- getLevels, 35
- getLevels, CuffData-method (getLevels), 35
- getLevels, CuffSet-method (getLevels), 35
- getSig, 36
- getSig, CuffSet-method (getSig), 36
- getSigTable, 37
- getSigTable, CuffSet-method (getSigTable), 37
  
- isoforms (CuffSet-class), 23
- isoforms, CuffGene-method (CuffGene-class), 21
- isoforms, CuffGeneSet-method (CuffGeneSet-class), 22
- isoforms, CuffSet-method (CuffSet-class), 23
  
- JSdist, 38
- JSdistFromP, 39
- JSdistVec, 40
  
- length, CuffFeature-method (CuffFeature-class), 19
- length, CuffFeatureSet-method (CuffFeatureSet-class), 20
  
- makeprobs, 41
- makeprobsvec, 42
- MPlot, 43
- MPlot, CuffData-method (MPlot), 43
  
- PINK1, 44
- promoters (CuffSet-class), 23

promoters, CuffGene-method  
(CuffGene-class), 21

promoters, CuffGeneSet-method  
(CuffGeneSet-class), 22

promoters, CuffSet-method  
(CuffSet-class), 23

readCufflinks, 44

relCDS (CuffSet-class), 23

relCDS, CuffGene-method  
(CuffGene-class), 21

relCDS, CuffGeneSet-method  
(CuffGeneSet-class), 22

relCDS, CuffSet-method (CuffSet-class),  
23

sampleGeneSet, 46

sampleIDs, 46

samples, 47

samples, CuffData-method (samples), 47

samples, CuffDist-method  
(CuffDist-class), 18

samples, CuffFeatureSet-method  
(CuffFeatureSet-class), 20

samples, CuffSet-method (samples), 47

shannon.entropy, 48

splicing (CuffSet-class), 23

splicing, CuffGene-method  
(CuffGene-class), 21

splicing, CuffGeneSet-method  
(CuffGeneSet-class), 22

splicing, CuffSet-method  
(CuffSet-class), 23

TSS (CuffSet-class), 23

TSS, CuffGene-method (CuffGene-class), 21

TSS, CuffGeneSet-method  
(CuffGeneSet-class), 22

TSS, CuffSet-method (CuffSet-class), 23