

# Package ‘compas’

October 28, 2015

**Type** Package

**Title** Comparative alternative splicing detection

**Version** 1.0

**Date** 2015-10-28

**Author** Nilgun Donmez

**Maintainer** Nilgun Donmez <ndonmez@sfu.ca>

**Description** This package is a collation of the necessary functions to run COMPAS, a comparative tool to detect alternative splicing using RNA-Seq data.

**License** GPL-2

## R topics documented:

compas-package . . . . .	1
compas-internal . . . . .	2
compasCore . . . . .	2
gpsrBasic . . . . .	3
<b>Index</b>	<b>5</b>

---

compas-package	<i>Comparative alternative splicing detection</i>
----------------	---

---

## Description

This package is a collation of the necessary functions to run COMPAS, a comparative tool to detect alternative splicing using RNA-Seq data.

## Note

Part of compas package. More information can be found at:  
<https://github.com/nlgndnmz/COMPAS.git>

**Author(s)**

Nilgun Donmez

**See Also**

See also [compasCore](#), [gpsrBasic](#)

---

compas-internal

*Collection of utility functions used by compasCore*

---

**Description**

These functions implement various internal methods and not intended to be used as standalone functions.

**Note**

Part of compas package. More information can be found at:  
<https://github.com/nlgndnmz/COMPAS.git>

**Author(s)**

Nilgun Donmez

**See Also**

See also [compasCore](#), [gpsrBasic](#)

---

compasCore

*Core method of the compas package*

---

**Description**

This is the core function of the compas package, an R package to detect alternative splicing using RNA-Seq data. The best way to use this (or any other) function in the compas package is through the R script that is provided at <https://github.com/nlgndnmz/COMPAS.git>.

**Usage**

```
compasCore(numSamples, numReads, readLength, inputFile,
outputPrefix, baseReads, histFiles, enableLoess,
ignoranceCutoff, sigLevel)
```

**Arguments**

<code>numSamples</code>	Number of samples to be compared. Currently only 1 or 2 samples are allowed.
<code>numReads</code>	A vector of size <code>numSamples</code> containing the number of reads per each sample.
<code>readLength</code>	A vector of size <code>readLength</code> containing the maximum read length per each sample.
<code>inputFile</code>	The path to an input file containing the coverage values for exons and junctions in each gene.
<code>outputPrefix</code>	A prefix for output files to be written on disk.
<code>baseReads</code>	This is the base number of reads used to normalize read coverage in the given samples.
<code>histFiles</code>	A vector of size <code>numSamples</code> containing the paths for optional read histogram files per each sample. If provided, these files are used to estimate the 5' to 3' bias of the datasets.
<code>enableLoess</code>	A logical parameter that enables the Loess regression on the histogram files given in <code>histFiles</code> .
<code>ignoranceCutoff</code>	A threshold value that is used to decide if the coverage at a junction is sufficient to make calls.
<code>sigLevel</code>	A threshold value that is used to decide if a differential splicing event is significant enough to report.

**Value**

Does not return a value.

**Note**

Part of compas package. More information can be found at:  
<https://github.com/nlgndnmz/COMPAS.git>

**Author(s)**

Nilgun Donmez

**See Also**

See also [gpsrBasic](#)

---

`gpsrBasic`

*Functions implementing the GPSR algorithm*

---

**Description**

These functions solve the convex problem :  $\arg \min_x 0.5 \|y - Ax\|_2^2 + \tau \|x\|_1$  subject to the additional constraint  $x \geq 0$

**Usage**

```
gpsrBasic(y, A, dimx, tauVec, cont_steps = 3, initAty = TRUE,
tolA = 0.0001, maxiter = 10000)
```

```
gpsrBB(y, A, dimx, tauVec, cont_steps = 4, initAty = TRUE,
tolA = 0.0001, maxiter = 10000)
```

**Arguments**

y	A real valued vector y containing the observations.
A	A real valued matrix A of size dimx by len(y).
dimx	The length of hidden vector x.
tauVec	A regularization (i.e. penalty) vector with the same size as x.
cont_steps	Number of continuation steps. Default value is 3.
initAty	Logical value to decide whether to initialize x to t(A)y. If false, x is initialized to 0. True by default.
tolA	A tolerance threshold to stop iterations. If the difference between the objective scores in successive iterations is less than this value, the algorithm terminates. Default value is 0.0001.
maxiter	Maximum number of iterations to execute before terminating. Default value is 10000.

**Value**

Both functions return a list with the following elements:

solFound	A logical value indicating if a feasible solution is found
solMatrix	A matrix of size dimx by cont_steps

**Note**

Part of compas package. More information can be found at:  
<https://github.com/nlgndnmz/COMPAS.git>

**Author(s)**

Nilgun Donmez

**References**

These algorithms are based on the GPSR-Basic and GPSR-Barzilai-borwein algorithms given in: "Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems" by Mario A. T. Figueiredo, Robert D. Nowak, Stephen J. Wright

**See Also**

See also [compasCore](#)

# Index

`adjustWeights (compas-internal)`, 2  
`assignExp (compas-internal)`, 2  
  
`checkCulprit (compas-internal)`, 2  
`compas (compas-package)`, 1  
`compas-internal`, 2  
`compas-package`, 1  
`compasCore`, 2, 2, 4  
`compasDuo (compas-internal)`, 2  
  
`debiasplotter (compas-internal)`, 2  
  
`getAnnotation (compas-internal)`, 2  
`getCandidates (compas-internal)`, 2  
`getCanonicals (compas-internal)`, 2  
`getLoessFit (compas-internal)`, 2  
`getPaths (compas-internal)`, 2  
`getRawExp (compas-internal)`, 2  
`gpsrBasic`, 2, 3, 3  
`gpsrBB (gpsrBasic)`, 3  
  
`printGTF (compas-internal)`, 2  
  
`removeLowJunctions`  
    `(compas-internal)`, 2  
  
`selectBest (compas-internal)`, 2