

# Package ‘ITALICS’

November 14, 2025

**Version** 2.70.0

**Date** 2008-05-14

**Title** ITALICS

**Depends** R (>= 2.0.0), GLAD, ITALICSData, oligo, affxparser,  
pd.mapping50k.xba240

**Imports** affxparser, DBI, GLAD, oligo, oligoClasses, stats

**Suggests** pd.mapping50k.hind240, pd.mapping250k.sty, pd.mapping250k.nsp

**Author** Guillem Rigail, Philippe Hupe

**Maintainer** Guillem Rigail <italics@curie.fr>

**Description** A Method to normalize of Affymetrix GeneChip Human Mapping  
100K and 500K set

**License** GPL-2

**URL** <http://bioinfo.curie.fr>

**biocViews** Microarray, CopyNumberVariation

**git\_url** <https://git.bioconductor.org/packages/ITALICS>

**git\_branch** RELEASE\_3\_22

**git\_last\_commit** 4a9bedd

**git\_last\_commit\_date** 2025-10-29

**Repository** Bioconductor 3.22

**Date/Publication** 2025-11-13

## Contents

addInfo . . . . .	2
analyseCGH . . . . .	3
fromQuartetToSnp . . . . .	4
fromSnpToQuartet . . . . .	5
getConfDat . . . . .	5
getCorrection . . . . .	6
getEffet . . . . .	7
getModel . . . . .	7
getQuartet . . . . .	8
getResidu . . . . .	9
getSnpInfo . . . . .	9

ITALICS . . . . .	10
readQuartetCopyNb . . . . .	12
trainITALICS . . . . .	13
<b>Index</b>	<b>14</b>

---

addInfo	<i>add info to quartet annotation</i>
---------	---------------------------------------

---

### Description

This function merge information obtain from the getQuartet function and a given table

### Usage

```
addInfo(quartet, dat)
```

### Arguments

quartet	list obtain through the getQuartet Function
dat	a data.frame with additionnal information it must contain a fsetid and fid column

### Value

a data.frame similar to the quartetInfo item of quartet plus additionnal column

### Note

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

### Author(s)

Guillem Rigail, <[italics@curie.fr](mailto:italics@curie.fr)>.

### Source

Institut Curie, <[italics@curie.fr](mailto:italics@curie.fr)>.

---

analyseCGH	<i>GLAD analysis</i>
------------	----------------------

---

**Description**

Glad Analysis of the genomic profile

**Usage**

```
analyseCGH(data, amplicon, deletion, deltaN, forceGL, param, nbsigma, ...)
```

**Arguments**

data	A data frame containing SNP's intensity, chromosome and position on the genome. data must have a Chr, X and LogRatio columns
amplicon	see the amplicon parameter in the daglad function
deletion	see the deletion parameter in the daglad function
deltaN	see the deltaN parameter in the daglad function
forceGL	see the forceGL parameter in the daglad function
param	see the param parameter in the daglad function
nbsigma	see the nbsigma parameter in the daglad function
...	Other daglad parameters.

**Value**

An object of class profileCGH

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <[italics@curie.fr](mailto:italics@curie.fr)>.

**Source**

Institut Curie, <[italics@curie.fr](mailto:italics@curie.fr)>.

---

fromQuartetToSnp	<i>Compute the copy number of each SNP from its quartets intensities</i>
------------------	--

---

### Description

This function removes the LogRatio column of the snpInfo data.frame. Then compute the copy number of each SNP having its quartet intensities. And return the snpInfo data.frame with the newly computed LogRatio.

### Usage

```
fromQuartetToSnp(quartetInfo, snpInfo, cIntensity="quartetLogRatio", nLog=1)
```

### Arguments

quartetInfo	A table containing the quartet intensities and other quartet information. It must have a column called : fsetid.
snpInfo	A table containing snp information.
cIntensity	A vector containing the names of the quartet information to be aggregate. For example quartetLogRatio.
nLog	The position of the field which will be named LogRatio in the snpInfo data.frame. For example if cIntensity = c("a", "b") and you want b to be considered as the LogRatio you should set nLog=2

### Value

return the data.frame snpInfo with additionnal columns.

### Note

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

### Author(s)

Guillem Rigaille, <italics@curie.fr>.

### Source

Institut Curie, <italics@curie.fr>.

---

fromSnpToQuartet	<i>Function to get from snp to quartet</i>
------------------	--

---

**Description**

This function put the smoothing value of each SNP in front of its corresponding quartet in the quartetInfo data.frame.

**Usage**

```
fromSnpToQuartet(quartetInfo, profilSNP)
```

**Arguments**

quartetInfo	a data frame containing all the quartet values plus there GC content, fragment length and GC content and Quartet effect
profilSNP	a data frame, corresponding to the profileValues argument of a profilCGH object (see GLAD)

**Value**

return the data.frame quartetInfo with an additionnal column: "Smoothing" corresponding to the estimated smoothing value.

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigaille, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

---

getConfDat	<i>Elimination of badly predicted probes</i>
------------	--

---

**Description**

This function eliminate badly predicted probes using a regression table and an estimated model given by the function getModel or getBestBICModelLight. Then it computes the corrected intensity.

**Usage**

```
getConfDat(confidence, quartetInfo, model)
```

**Arguments**

confidence	The confidence interval : 0.95
quartetInfo	A Regression table containing the variables in the model
model	The class lm object given by the function getModel

**Value**

A data frame with the corrected intensity. Only goodly predicted probes are taken into account. SNP's with more than 8 badly predicted probes get a NA.

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <[italics@curie.fr](mailto:italics@curie.fr)>.

**Source**

Institut Curie, <[italics@curie.fr](mailto:italics@curie.fr)>.

---

getCorrection	<i>Correction</i>
---------------	-------------------

---

**Description**

This function computes the corrected intensity.

**Usage**

```
getCorrection(effet, model, regTab)
```

**Arguments**

effet	The name of the biological effect
model	The class lm object given by the getModel function
regTab	The regression table used to estimate the linear model, and containing the variables in the model

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <[italics@curie.fr](mailto:italics@curie.fr)>.

**Source**

Institut Curie, <[italics@curie.fr](mailto:italics@curie.fr)>.

---

`getEffet`*Effet*

---

**Description**

This function retrieves the estimated biological effect

**Usage**

```
getEffet(effet, model, regTab)
```

**Arguments**

<code>effet</code>	The name of the biological effect
<code>model</code>	The class <code>lm</code> object given by the <code>getModel</code> function
<code>regTab</code>	The regression table used to estimate the linear model, and containing the variables in the model

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <[italics@curie.fr](mailto:italics@curie.fr)>.

**Source**

Institut Curie, <[italics@curie.fr](mailto:italics@curie.fr)>.

---

`getModel`*Regression Model*

---

**Description**

Computes the linear regression model and and return an object of class `lm`.

**Usage**

```
getModel(formule, response, regTab)
```

**Arguments**

<code>formule</code>	A symbolic description of the term of the model. It is a string
<code>response</code>	The parameter you want to explain (the response) : the SNP "LogRatio". Y is a string
<code>regTab</code>	A Regression table containing the variables in the model

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

---

getQuartet

*Function to retrieve the information of each quartet*

---

**Description**

This function retrieve information of each quartet. This function use the pd.mapping50k.xba240, pd.mapping50k.hind240, pd.mapping250k.sty and pd.mapping250k.nsp package.

**Usage**

```
getQuartet(pkgname, snpInfo)
```

**Arguments**

pkgname	the chip type pd.mapping50k.xba240, pd.mapping50k.hind240, pd.mapping250k.sty or pd.mapping250k.nsp
snpInfo	a data frame containing SNPs position along the genome

**Value**

return a list with two fields. fid : containing the position of each quartet on the CEL file. quartetInfo : a data fame containing the columns : fsetid, fid, FL (fragment length) and GC (content of the quartet)

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

---

getResidu	<i>Correction</i>
-----------	-------------------

---

**Description**

This function retrieves the residual values

**Usage**

```
getResidu(model)
```

**Arguments**

model            The class lm object given by the getModel function

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

---

getSnInfo	<i>Function to retrieve the chromosome and the position of each SNP on a given Affymetrix SNP array</i>
-----------	---

---

**Description**

This function retrieve the chromosome and position in bp of each SNP of a given Affymetrix SNP array. This function use the pd.mapping50k.xba240, pd.mapping50k.hind240, pd.mapping250k.sty and pd.mapping250k.nsp package.

**Usage**

```
getSnInfo(pkgname)
```

**Arguments**

pkgname            the chip type pd.mapping50k.xba240, pd.mapping50k.hind240, pd.mapping250k.sty or pd.mapping250k.nsp

**Value**

Return a data.frame with five columns : fsetid, dbsnp\_rs\_id, Chr, X and fragment\_length corresponding to the fsetid, the rs\_id, the chromosome, the position on the chromosome and the PCR amplified fragment length respectively.

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigail, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

---

ITALICS

*Affymetrix SNP chip normalization*

---

**Description**

Normalize and analyse Affymetrix SNP array 100K and 500K set (see the vignette)

**Usage**

```
ITALICS(quartetInfo, snpInfo, confidence=0.95, iteration=2,
        formule="Smoothing+QuartetEffect+FL+I(FL^2)+I(FL^3)+GC+I(GC^2)+I(GC^3)", prc=0.3,
        amplicon=2.1, deletion=-3.5, deltaN=0.15, forceGL=c(-0.2,0.2), param=c(d=2), nbsigma=1, ... )
```

**Arguments**

quartetInfo	a data frame containing all the raw quartet intensities plus there GC content, fragment length, and Quartet effect
snpInfo	a data frame containing SNPs position along the genome and raw copy number
confidence	The confidence interval. After the last bias estimation step, quartets outside this confidence interval are flagged. The lower confidence is, the more quartets will be flagged. See also the parameter prc.
iteration	The number of iteration you d'like to do
formule	A symbolic description of the term of the model. The default value of formule means that we want correct the observed quartetLogRatio using the estimated copy number (Smoothing), the Quartet Effect, the quartet Fragment Length (FL) and the quartet GC content.
prc	prc is a frequence (between 0 and 1). After the final iteration of ITALICS, badly predicted probes are flagged (see also the parameter confidence). Only SNPs having more than prc of their probes non-flagged are kept for the final GLAD analysis. The higher prc is, the more SNPs are removed before the final GLAD analysis.
amplicon	see the amplicon parameter in the daglad function
deletion	see the deletion parameter in the daglad function
deltaN	see the deltaN parameter in the daglad function
forceGL	see the forceGL parameter in the daglad function
param	see the param parameter in the daglad function
nbsigma	see the nbsigma parameter in the daglad function
...	Other daglad parameters.

**Details**

The function ITALICS implements the methodology which is described in the article : ITALICS: an algorithm for normalization and DNA copy number calling for Affymetrix SNP arrays (Rigaill et al., Bioinformatics Advance Access published on February 5, 2008).

The principle of the ITALICS algorithm: ITALICS, is a normalization method that estimates both the biological and the non-relevant effects in an alternate and iterative way to accurately remove the non-relevant effects.

ITALICS deals with known systematic sources of variation such as the GC-content of the quartets, the PCR amplified fragment length and the GC-content of the PCR amplified fragment . It also takes into account the quartet effect which corresponds to the fact that some quartets systematically have a small intensity while others tend to have a high intensity. ITALICS is also able to correct spatial artifacts which sometimes arise on Affymetrix SNP arrays 100K and 500K set.

**Value**

Return an object of class profileCGH

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigaill, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

**Examples**

```
## Not run:
## step to get the path of the HF0844_Hind.CEL file
ITALICSDataPATH <- attr(as.environment(match("package:ITALICSData", search())),"path")
filename <- paste(ITALICSDataPATH, "/extdata/HF0844_Hind.CEL", sep="")
quartetEffectFile <- paste(ITALICSDataPATH, "/data/Hind.QuartetEffect.csv", sep="")

## load quartet effect
quartetEffect <- read.table(quartetEffectFile, sep=";", header=TRUE)

## load annotation using the pd.mapping50k.xba24 or pd.mapping50k.hind240 or pd.mapping250k.sty or pd.mapping
headdetails <- readCelHeader(filename[1])
pkgname <- cleanPlatformName(headdetails[["chiptype"]])
snpInfo <- getSnpInfo(pkgname)
quartet <- getQuartet(pkgname, snpInfo)

## read cel files and format data
tmpExprs <- readCellIntensities(filename, indices=quartet$fid)
quartet$quartetInfo$quartetLogRatio <- readQuartetCopyNb(tmpExprs)
quartet$quartetInfo <- addInfo(quartet, quartetEffect)
snpInfo <- fromQuartetToSnp(cIntensity="quartetLogRatio", quartetInfo=quartet$quartetInfo, snpInfo=snpInfo)
```

```
## ITALICS normalization
profilSNPHind <- ITALICS(quartet$quartetInfo, snpInfo,
  formule="Smoothing+QuartetEffect+FL+I(FL^2)+I(FL^3)+GC+I(GC^2)+I(GC^3)")

## plot the profile
data(cytoband)
plotProfile(profilSNPHind, Smoothing="Smoothing", Bkp=TRUE, cytoband = cytoband)

## End(Not run)
```

---

readQuartetCopyNb      *Read PM probes of selected quartets and compute the quartet intensity*

---

## Description

This function read the cel files and return the raw-value of each quartet = mean of allele A and B

## Usage

```
readQuartetCopyNb(tmpExprs)
```

## Arguments

tmpExprs      A vector of the perfect match intensity of allele A and B of the quartets. This vector should be sorted in a specific order. See the example given in the help of the ITALICS function.

## Value

return a vector with the raw-value of each quartet

## Note

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

## Author(s)

Guillem Rigaiil, <italics@curie.fr>.

## Source

Institut Curie, <italics@curie.fr>.

---

trainITALICS	<i>ITALICS training</i>
--------------	-------------------------

---

**Description**

Estimation of the quartet effect based on several normal sample chips

**Usage**

```
trainITALICS (dir, amplicon=2.1, deletion=-3.5, deltaN=0.15, forceGL=c(-0.2,0.2), param=c(d=2), nb
```

**Arguments**

dir	The directory containing the normal sample chips. All these chips should be of the same type hind, xba, nsp or sty. Only .CEL files be considered
amplicon	see the amplicon parameter in the daglad function
deletion	see the deletion parameter in the daglad function
deltaN	see the deltaN parameter in the daglad function
forceGL	see the forceGL parameter in the daglad function
param	see the param parameter in the daglad function
nbsigma	see the nbsigma parameter in the daglad function
...	Other daglad parameters.

**Details**

The ITALICS function take into account a quartet effect which is computed on a reference data set of normal women samples. The ITALICSData provide quartetEffect for the Xba, Hind, Sty and Nsp chip computed on our own reference data set.

We recommand that you use your own reference data set to compute the quartet Effect by using the trainITALICS function. ITALICS reference data should contain only woman normal samples. Furthermore we recommand that you check that none of these chip have obvious spatial artifact. To so read the cel files using the read.affybatch (form the affy package). Then use the image function on the obtain affybatch object.

**Value**

a data.frame with two column fsetid and quartetEffect

**Note**

People interested in tools dealing with array CGH analysis and DNA copy number analysis can visit our web-page <http://bioinfo.curie.fr>.

**Author(s)**

Guillem Rigaille, <italics@curie.fr>.

**Source**

Institut Curie, <italics@curie.fr>.

# Index

addInfo, [2](#)  
analyseCGH, [3](#)  
  
fromQuartetToSnp, [4](#)  
fromSnpToQuartet, [5](#)  
  
getConfDat, [5](#)  
getCorrection, [6](#)  
getEffet, [7](#)  
getModel, [7](#)  
getQuartet, [8](#)  
getResidu, [9](#)  
getSnpInfo, [9](#)  
  
ITALICS, [10](#)  
Italics (ITALICS), [10](#)  
italics (ITALICS), [10](#)  
  
readQuartetCopyNb, [12](#)  
  
trainITALICS, [13](#)  
trainItalics (trainITALICS), [13](#)  
trainitalics (trainITALICS), [13](#)