

# Package ‘bibliometrix’

March 14, 2018

**Type** Package

**Title** An R-Tool for Comprehensive Science Mapping Analysis

**Version** 1.9.1

**Date** 2018-03-14

**Description** Tool for quantitative research in scientometrics and bibliometrics. It provides various routines for importing bibliographic data from SCOPUS (<<http://scopus.com>>) and Clarivate Analytics Web of Science (<<http://www.webofknowledge.com/>>) databases, performing bibliometric analysis and building data matrices for co-citation, coupling, scientific collaboration and co-word analysis.

**License** GPL-3

**URL** <http://www.bibliometrix.org>

**LazyData** FALSE

**Encoding** latin1

**Depends** R (>= 3.3.0)

**Imports** stats, factoextra, FactoMineR, ggplot2, ggrepel, igraph, Matrix, RColorBrewer, RISmed, rscopus, SnowballC, stringdist, stringr

**Suggests** knitr, rmarkdown

**RoxygenNote** 6.0.1

**NeedsCompilation** no

**Author** Massimo Aria [cre, aut],  
Corrado Cuccurullo [aut]

**Maintainer** Massimo Aria <[aria@unina.it](mailto:aria@unina.it)>

**VignetteBuilder** knitr

**Repository** CRAN

**Date/Publication** 2018-03-14 13:39:24 UTC

**R topics documented:**

bibliometrix-package	3
biblio	4
biblioAnalysis	5
biblioNetwork	6
biblio_df	8
citations	9
coeMatrix	10
conceptualStructure	11
convert2df	13
countries	15
dominance	15
duplicatedMatching	16
garfield	17
Hindex	18
histNetwork	19
histPlot	20
idByAuthor	21
isi2df	22
isibib2df	23
isiCollection	24
keywordAssoc	25
KeywordGrowth	26
localCitations	27
lotka	28
mergeDbSources	29
metaTagExtraction	30
networkPlot	31
normalizeSimilarity	33
plot.bibliometrix	34
pubmed2df	35
readFiles	36
retrievalByAuthorID	37
scientometrics	38
scientometrics_text	39
scopus2df	40
scopusCollection	41
stopwords	42
summary.bibliometrix	42
tableTag	43
termExtraction	44
thematicEvolution	46
thematicMap	47
timeslice	48
trim	49
trim.leading	50

---

bibliometrix-package *Tool for quantitative research in scientometrics and bibliometrics.*

---

## Description

It provides various routines for importing bibliographic data from SCOPUS and Thomson Reuters' ISI Web of Knowledge databases, performing bibliometric analysis and building data matrices for co-citation, coupling and scientific collaboration analysis.

## Details

Package: bibliometrix  
Type: Package  
Version: 0.1  
Date: 2016-05-05  
License: GPL-3

## Author(s)

Massimo Aria <massimo.aria@unina.it>, Corrado Cuccurullo <corrado.cuccurullo@unina2.it>  
Maintainer: Massimo Aria <massimo.aria@unina.it>

## References

- Cuccurullo C., Aria M., Sarto F. (2016) Foundations and Trends in Performance Management. A Twenty-five Years Bibliometric Analysis in Business and Public Administration Domains, Scientometrics, DOI: 10.1007/s11192-016-1948-8.
- Koseoglu, M. A. (2016). Growth and structure of authorship and co-authorship network in the strategic management realm: Evidence from the Strategic Management Journal. BRQ Business Research Quarterly.
- Batagelj, V., Cerinsek, M. (2013). On bibliographic networks. Scientometrics, 96(3), 845-864.
- Yan, E., Ding, Y. (2012). Scholarly network similarities: How bibliographic coupling networks, citation networks, cocitation networks, topical networks, coauthorship networks, and coword networks relate to each other. Journal of the American Society for Information Science and Technology, 63(7), 1313-1326.
- Rousseau, R. (2010). Bibliographic coupling and co-citation as dual notions. The Janus faced scholar. A Festschrift in honour of Peter Ingwersen, 173-183.
- Leydesdorff, L., Vaughan, L. (2006). Co-occurrence matrices and their applications in information science: Extending ACA to the Web environment. Journal of the American Society for Information Science and technology, 57(12), 1616-1628.

## Examples

```
## load scientometrics data set
# data scientometrics_text)

## Convert text data into a bibliographic data frame
# scient_df <- convert2df scientometrics_text, dbsource="isi", format="plaintext")

## Perform a bibliometric analysis of the bibliographic data frame
# results <- biblioAnalysis scient_df)

## summarize results
# summary results, k=10, pause=FALSE)

## plot results
# plot results, k=10, pause=FALSE)

## Estimate Lotka's law coefficients
# L=lotka results)
# L

## Perform authors' dominance analysis
#DF=dominance results)
#DF
```

---

biblio

*Dataset of "Bibliometrics" manuscripts.*

---

## Description

The set of manuscripts which the title containing the word "bibliometrics" and published in a journal indexed by ISI WoK database.

Period: 2006 - 2015

Database: [ISI Web of Knowledge](http://www.webofknowledge.com)

## Format

A large character with 9014 rows.

Data has been imported by an ISI Export file in bibtex format using the function [readLines](#).

## Source

<http://www.webofknowledge.com>

---

biblioAnalysis	<i>Bibliometric Analysis</i>
----------------	------------------------------

---

### Description

It performs a bibliometric analysis of a dataset imported from SCOPUS and Thomson Reuters' ISI Web of Knowledge databases.

### Usage

```
biblioAnalysis(M, sep = ";")
```

### Arguments

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
sep	is the field separator character. This character separates strings in each column of the data frame. The default is sep = ";".

### Value

biblioAnalysis returns an object of class "bibliometrix".

The functions [summary](#) and [plot](#) are used to obtain or print a summary and some useful plots of the results.

An object of class "bibliometrix" is a list containing the following components:

Articles	the total number of manuscripts
Authors	the authors' frequency distribution
AuthorsFrac	the authors' frequency distribution (fractionalized)
FirstAuthors	first author of each manuscript
nAUpperPaper	the number of authors per manuscript
Appearances	the number of author appearances
nAuthors	the number of authors
AuMultiAuthoredArt	the number of authors of multi authored articles
MostCitedPapers	The list of manuscripts sorted by citations
Years	publication year of each manuscript
FirstAffiliation	the affiliation of the first author
Affiliations	the frequency distribution of affiliations (of all co-authors for each paper)
Aff_frac	the fractionalized frequency distribution of affiliations (of all co-authors for each paper)
CO	the affiliation country of first author
Countries	the affiliation countries' frequency distribution
CountryCollaboration	Intracountry (SCP) and intercountry (MCP) collaboration indices
TotalCitation	the number of times each manuscript has been cited
TCperYear	the yearly average number of times each manuscript has been cited
Sources	the frequency distribution of sources (journals, books, etc.)
DE	the frequency distribution of authors' keywords
ID	the frequency distribution of keywords associated to the manuscript by SCOPUS and Thomson Reuters

**See Also**

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.

[summary](#) to obtain a summary of the results.

[plot](#) to draw some useful plots of the results.

**Examples**

```
data(scientometrics)

results <- biblioAnalysis(scientometrics)

summary(results, k = 10, pause = FALSE)
```

---

 biblioNetwork

*Creating Bibliographic networks*


---

**Description**

biblioNetwork creates different bibliographic networks from a bibliographic data frame.

**Usage**

```
biblioNetwork(M, analysis = "coupling", network = "authors", sep = ";")
```

**Arguments**

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
analysis	is a character object. It indicates the type of analysis have to be performed. analysis argument can be "collaboration", "coupling", "co-occurrences" or "co-citation". Default is analysis = "coupling".
network	is a character object. It indicates the network typology. The network argument can be "authors", "references", "sources", "countries", "keywords", "author_keywords", "titles", or "abstracts". Default is network = "authors".
sep	is the field separator character. This character separates strings in each column of the data frame. The default is sep = ";".

**Details**

The function [biblioNetwork](#) can create a collection of bibliographic networks following the approach proposed by Batagely and Cerinsek (2013).

Typical networks output of biblioNetwork are:

```
#### Collaboration Networks #####
- Authors collaboration (analysis = "collaboration", network = "authors")
- University collaboration (analysis = "collaboration", network = "universities")
- Country collaboration (analysis = "collaboration", network = "countries")

#### Co-citation Networks #####
- Authors co-citation (analysis = "co-citation", network = "authors")
- Reference co-citation (analysis = "co-citation", network = "references")
- Source co-citation (analysis = "co-citation", network = "sources")

#### Coupling Networks #####
- Manuscript coupling (analysis = "coupling", network = "references")
- Authors coupling (analysis = "coupling", network = "authors")
- Source coupling (analysis = "coupling", network = "sources")
- Country coupling (analysis = "coupling", network = "countries")

#### Co-occurrences Networks #####
- Authors co-occurrences (analysis = "co-occurrences", network = "authors")
- Source co-occurrences (analysis = "co-occurrences", network = "sources")
- Keyword co-occurrences (analysis = "co-occurrences", network = "keywords")
- Author-Keyword co-occurrences (analysis = "co-occurrences", network = "author_keywords")
- Title content co-occurrences (analysis = "co-occurrences", network = "titles")
- Abstract content co-occurrences (analysis = "co-occurrences", network = "abstracts")
```

## Value

It is a squared network matrix. It is an object of class `dgMatrix` of the package [Matrix](#).

## See Also

[convert2df](#) to import and convert a SCOPUS and Thomson Reuters' ISI Web of Knowledge export file in a data frame.

[cocMatrix](#) to compute a co-occurrence matrix.

[biblioAnalysis](#) to perform a bibliometric analysis.

## Examples

```
# EXAMPLE 1: Authors collaboration network

# data(scientometrics)

# NetMatrix <- biblioNetwork(scientometrics, analysis = "collaboration",
# network = "authors", sep = ";")

# net <- networkPlot(NetMatrix, n = 30, type = "kamada", Title = "Collaboration", labelsSize=0.5)
```

```
# EXAMPLE 2: Co-citation network

data scientometrics

NetMatrix <- biblioNetwork(scientometrics, analysis = "co-citation",
network = "references", sep = ";")

net <- networkPlot(NetMatrix, n = 30, type = "kamada", Title = "Co-Citation", labelsize=0.5)
```

---

biblio\_df

*Dataset of "Bibliometrics" manuscripts.*


---

### Description

The set of manuscripts which the title containing the word "bibliometrics" and published in a journal indexed by ISI WoK database.

Period: 2006 - 2015

Database: [ISI Web of Knowledge](http://www.webofknowledge.com)

### Format

#' A data frame with 99 rows (manuscripts) and 16 variables (ISI tag field):

**AU** Authors

**TI** Document Title

**SO** Publication Name (or Source)

**JI** ISO Source Abbreviation

**DT** Document Type

**DE** Author Keywords

**ID** Keywords associated by ISI or SCOPUS database

**AB** Abstract

**C1** Author Address

**RP** Reprint Address

**CR** Cited References

**TC** Times Cited

**PY** Year

**SC** Subject Category

**UT** Unique Article Identifier

**DB** Database

### Source

<http://www.webofknowledge.com>



---

citations	<i>Citation frequency distribution</i>
-----------	--

---

**Description**

It calculates frequency distribution of citations.

**Usage**

```
citations(M, field = "article", sep = ";")
```

**Arguments**

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
field	is a character. It can be "article" or "author" to obtain frequency distribution of cited citations or cited authors (only first authors for ISI database) respectively. The default is field = "article".
sep	is the field separator character. This character separates citations in each string of CR column of the bibliographic data frame. The default is sep = ";".

**Value**

an object of class "list" containing the following components:

Cited	the most frequent cited manuscripts or authors
Year	the publication year (only for cited article analysis)
Source	the journal (only for cited article analysis)

**See Also**

[biblioAnalysis](#) function for bibliometric analysis.

[summary](#) to obtain a summary of the results.

[plot](#) to draw some useful plots of the results.

**Examples**

```
## EXAMPLE 1: Cited articles

data(scientometrics)

CR <- citations(scientometrics, field = "article", sep = ";")

CR$Cited[1:10]
CR$Year[1:10]
CR$Source[1:10]
```

```
## EXAMPLE 2: Cited first authors

data(scientoMetrics)

CR <- citations(scientoMetrics, field = "author", sep = ";")

CR$Cited[1:10]
```

---

cocMatrix

*Co-occurrence matrix*


---

### Description

cocMatrix computes co-occurrences between elements of a Tag Field from a bibliographic data frame. Manuscript is the unit of analysis.

### Usage

```
cocMatrix(M, Field = "AU", type = "sparse", sep = ";")
```

### Arguments

**M** is a data frame obtained by the converting function [convert2df](#). It is a data matrix with cases corresponding to articles and variables to Field Tag in the original ISI or SCOPUS file.

**Field** is a character object. It indicates one of the field tags of the standard ISI WoS Field Tag codify. Field can be equal to one of this tags:

AU	Authors
SO	Publication Name (or Source)
JI	ISO Source Abbreviation
DE	Author Keywords
ID	Keywords associated by ISI or SCOPUS database
CR	Cited References

for a complete list of filed tags see: [Field Tags used in bibliometrix](#)

**type** indicates the output format of co-occurrences:

type = "matrix" produces an object of class matrix

type = "sparse" produces an object of class dgMatrix of the package [Matrix](#). "sparse" argument generates a compact

**sep** is the field separator character. This character separates strings in each column of the data frame. The default is sep = ";".

### Details

This co-occurrence matrix can be transformed into a collection of compatible networks. Through matrix multiplication you can obtain different networks. The function follows the approach proposed by Batagely and Cerinsek (2013).

### Value

a co-occurrence matrix with cases corresponding to manuscripts and variables to the objects extracted from the Tag Field.

### See Also

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a data frame.

[biblioAnalysis](#) to perform a bibliometric analysis.

[biblioNetwork](#) to compute a bibliographic network.

### Examples

```
# EXAMPLE 1: Articles x Authors co-occurrence matrix

data(scientometrics)
WA <- cocMatrix(scientometrics, Field = "AU", type = "sparse", sep = ";")

# EXAMPLE 2: Articles x Cited References co-occurrence matrix

# data(scientometrics)

# WCR <- cocMatrix(scientometrics, Field = "CR", type = "sparse", sep = ";")

# EXAMPLE 3: Articles x Cited First Authors co-occurrence matrix

# data(scientometrics)
# scientometrics <- metaTagExtraction(scientometrics, Field = "CR_AU", sep = ";")
# WCR <- cocMatrix(scientometrics, Field = "CR_AU", type = "sparse", sep = ";")
```

---

conceptualStructure *Creating and plotting conceptual structure map of a scientific field*

---

### Description

The function `conceptualStructure` creates a conceptual structure map of a scientific field performing Multiple Correspondence Analysis (MCA) and Clustering of a bipartite network of terms extracted from keyword, title or abstract fields.

**Usage**

```
conceptualStructure(M, field = "ID", quali.sup = NULL,
  quanti.sup = NULL, minDegree = 2, k.max = 5, stemming = FALSE,
  labelsize = 3)
```

**Arguments**

M	is a data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to articles and variables to Field Tag in the original ISI or SCOPUS file.
field	is a character object. It indicates one of the field tags of the standard ISI WoS Field Tag codify. field can be equal to one of this tags:
ID	Keywords Plus associated by ISI or SCOPUS database
DE	Author's keywords
ID_TM	Keywords Plus stemmed through the Porter's stemming algorithm
DE_TM	Author's Keywords stemmed through the Porter's stemming algorithm
TI	Terms extracted from titles
AB	Terms extracted from abstracts
quali.sup	is a vector indicating the indexes of the categorical supplementary variables.
quanti.sup	is a vector indicating the indexes of the quantitative supplementary variables.
minDegree	is an integer. It indicates the minimum occurrences of terms to analyze and plot. The default value is 2.
k.max	is an integer. It indicates the maximum number of cluster to keep. The default value is 5. The max value is 8.
stemming	is logical. If TRUE the Porter's Stemming algorithm is applied to all extracted terms. The default is stemming = FALSE.
labelsize	is an integer. It indicates the label size in the plot. Default is labelsize=2

**Value**

It is an object of the class `list` containing the following components:

net	bipartite network
res.mca	Results of Multiple Correspondence Analysis
km.res	Results of cluster analysis

**See Also**

[termExtraction](#) to extract terms from a textual field (abstract, title, author's keywords, etc.) of a bibliographic data frame.

[biblioNetwork](#) to compute a bibliographic network.

[cocMatrix](#) to compute a co-occurrence matrix.

[biblioAnalysis](#) to perform a bibliometric analysis.

**Examples**

```
# EXAMPLE Conceptual Structure using Keywords Plus

data(scientometrics)

S <- conceptualStructure(scientometrics, field="ID_TM", stemming=TRUE, minDegree=5, k.max = 5)
```

---

convert2df	<i>Convert a Clarivate Analytics WoS and SCOPUS Export files or RISmed PubMed/MedLine object into a data frame</i>
------------	--

---

**Description**

It converts a SCOPUS and Clarivate Analytics WoS export files or RISmed PubMed/MedLine object into a data frame, with cases corresponding to articles and variables to Field Tags as used in WoS.

**Usage**

```
convert2df(file, dbsource = "isi", format = "plaintext")
```

**Arguments**

file	can be: a) a character array containing data read from a Clarivate Analytics WoS Export file (in plain text or bibtex format) or SCOPUS Export file (exclusively in bibtex format); b) an object of the class pubmed (package RISmed) containing a collection obtained from a query performed with RISmed package.
dbsource	is a character indicating the bibliographic database. dbsource can be "isi", "scopus" or pubmed. Default is dbsource = "isi".
format	is a character indicating the format of the SCOPUS and Clarivate Analytics WoS export file. format can be "bibtex" or "plaintext". Default is format = "plaintext".

**Details**

Actually the function allows to convert both SCOPUS/WoS files in bibtex format and just WoS files in plain text format.

**Value**

a data frame with cases corresponding to articles and variables to Field Tags in the original export file.

data frame columns are named using the standard Clarivate Analytics WoS Field Tag codify. The main field tags are:

AU	Authors
TI	Document Title

SO	Publication Name (or Source)
JI	ISO Source Abbreviation
DT	Document Type
DE	Authors' Keywords
ID	Keywords associated by SCOPUS or WoS database
AB	Abstract
C1	Author Address
RP	Reprint Address
CR	Cited References
TC	Times Cited
PY	Year
SC	Subject Category
UT	Unique Article Identifier
DB	Database

for a complete list of field tags see: [Field Tags used in bibliometrix](#)

### See Also

[scopus2df](#) for converting SCOPUS Export file (in bibtex format)

[isibib2df](#) for converting ISI Export file (in bibtex format)

[isi2df](#) for converting ISI Export file (in plain text format)

[pubmed2df](#) for converting an object of the class pubmed (RISmed package)

Other converting functions: [isi2df](#), [isibib2df](#), [pubmed2df](#), [scopus2df](#)

### Examples

```
# An ISI or SCOPUS Export file can be read using \link{readLines} function:

# D <- readFiles('filename1.txt','filename2.txt','filename3.txt')

# filename1.txt, filename2.txt and filename3.txt are WoS or SCOPUS Export file
# in plain text or bibtex format.

# biblio <- readFiles('http://www.bibliometrix.org/datasets/bibliometrics_articles.txt')

data(biblio)

biblio_df_df <- convert2df(file = biblio, dbsource = "isi", format = "bibtex")
```

---

countries	<i>Index of Countries.</i>
-----------	----------------------------

---

**Description**

Data frame containing a normalized index of countries.

Data are used by `biblioAnalysis` function to extract Country Field of Cited References and Authors.

**Format**

A data frame with 198 rows and 1 variable:

**countries** country names

---

dominance	<i>Authors' dominance ranking</i>
-----------	-----------------------------------

---

**Description**

It calculates the authors' dominance ranking from an object of the class 'bibliometrix' as proposed by Kumar & Kumar, 2008.

**Usage**

```
dominance(results, k = 10)
```

**Arguments**

**results** is an object of the class 'bibliometrix' for which the analysis of the authors' dominance ranking is desired.

**k** is an integer, used for table formatting (number of authors). Default value is 10.

**Value**

The function `dominance` returns a data frame with cases corresponding to the first `k` most productive authors and variables to typical field of a dominance analysis.

the data frame variables are:

Dominance Factor	Dominance Factor (DF = FAA / MAA)
Multi Authored	N. of Multi Authored Articles (MAA)
First Authored	N. of First Authored Articles (FAA)
Rank by Articles	Author Ranking by N. of Articles
Rank by DF	Author Ranking by Dominance Factor

**See Also**

[biblioAnalysis](#) function for bibliometric analysis  
[summary](#) method for class 'bibliometrix'

**Examples**

```
data(scientometrics)
results <- biblioAnalysis(scientometrics)
DF=dominance(results)
DF
```

---

duplicatedMatching      *Searching of duplicated records in a bibliographic database*

---

**Description**

Search duplicated records in a dataframe.

**Usage**

```
duplicatedMatching(M, Field = "TI", tol = 0.95)
```

**Arguments**

M	is the bibliographic data frame.
Field	is a character object. It indicates one of the field tags used to identify duplicated records. Field can be equal to one of this tags: TI (title), AB (abstract), UT (manuscript ID).
tol	is a numeric value giving the minimum relative similarity to match two manuscripts. Default value is tol = 0.95.

**Details**

A bibliographic data frame is obtained by the converting function [convert2df](#). It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file. The function identifies duplicated records in a bibliographic data frame and deletes them. Duplicate entries are identified through the restricted Damerau-Levenshtein distance. Two manuscripts that have a relative similarity measure greater than tol argument are stored in the output data frame only once.

**Value**

the value returned from duplicatedMatching is a data frame without duplicated records.



**See Also**

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.

[biblioAnalysis](#) function for bibliometric analysis.

[summary](#) to obtain a summary of the results.

[plot](#) to draw some useful plots of the results.

**Examples**

```
data(scientometrics)

M=rbind(scientometrics[1:20,],scientometrics[10:30,])

newM <- duplicatedMatching(M, Field = "TI", tol = 0.95)

dim(newM)
```

---

garfield

*Eugene Garfield's manuscripts.*

---

**Description**

All manuscripts published by Eugene Garfield.

Period: 1954 - 2014

Database: [SCOPUS source](#)

**Format**

A data frame with 147 rows and 15 variables:

**AU** Authors

**TI** Document Title

**SO** Publication Name (or Source)

**JI** ISO Source Abbreviation

**DT** Document Type

**DE** Author Keywords

**ID** Keywords associated by ISI or SCOPUS database

**AB** Abstract

**C1** Author Address

**RP** Reprint Address

**CR** Cited References

**TC** Times Cited  
**PY** Year  
**UT** Unique Article Identifier  
**DB** Database

### Source

<http://www.scopus.com>

---

Hindex	<i>h-index calculation</i>
--------	----------------------------

---

### Description

It calculates the authors' h-index and its variants.

### Usage

```
Hindex(M, authors, sep = ";", years = 10)
```

### Arguments

<b>M</b>	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
<b>authors</b>	is a character vector. It contains the the authors' names list for which you want to calculate the H-index. The argument has the form C("SURNAME1 N", "SURNAME2 N", ...), in other words, for each author: surname and initials separated by one blank space. i.e for the authors SEMPRONIO TIZIO CAIO and ARIA MASSIMO authors argument is authors = c("SEMPRONIO TC", "ARIA M").
<b>sep</b>	is the field separator character. This character separates authors in each string of AU column of the bibliographic data frame. The default is sep = ";".
<b>years</b>	is a integer. It indicates the number of years to consider for Hindex calculation. Default is 10.

### Value

an object of class "list". It contains two elements: H is a data frame with h-index, g-index and m-index for each author; CitationList is a list with the bibliographic collection for each author.

### See Also

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.  
[biblioAnalysis](#) function for bibliometric analysis.  
[summary](#) to obtain a summary of the results.  
[plot](#) to draw some useful plots of the results.

## Examples

```
### EXAMPLE 1: ###

data(scientometrics)

authors <- c("SMALL H", "CHEN DZ")

Hindex(scientometrics, authors, sep = ";")$H

### EXAMPLE 2: Garfield h-index###

data(garfield)

indices=Hindex(garfield, authors="GARFIELD E", sep = ";")

# h-index, g-index and m-index of Eugene Garfield
indices$H

# Papers and total citations
indices$CitationList[[1]]
```

---

histNetwork

*Historical co-citation network*

---

## Description

histNetwork creates a historical citation network from a bibliographic data frame.

## Usage

```
histNetwork(M, n = 10, sep = ";")
```

## Arguments

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
n	is an integer, indicating the number of most cited references to select. Default value is 10.
sep	is the field separator character. This character separates strings in CR column of the data frame. The default is sep = ";".

## Value

histNetwork returns an object of class "list" containing the following components:

NetMatrix	the historical co-citation network matrix
Degree	the min degree of the network
histData	the set of n most cited references
M	the bibliographic data frame

**See Also**

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.

[summary](#) to obtain a summary of the results.

[plot](#) to draw some useful plots of the results.

[biblioNetwork](#) to compute a bibliographic network.

**Examples**

```
data(scientometrics)
```

```
histResults <- histNetwork(scientometrics, n = 10, sep = ";")
```

---

histPlot

*Plotting historical co-citation network*


---

**Description**

histPlot plots a historical co-citation network.

**Usage**

```
histPlot(histResults, remove.isolates = FALSE, size = F, labelsize = 0.8,
         label = TRUE, arrowsize = 0.1)
```

**Arguments**

histResults is an object of class "list" containing the following components:

NetMatrix	the historical citation network matrix
Degree	the min degree of the network
histData	the set of n most cited references
M	the bibliographic data frame

remove.isolates is a network matrix obtained by the function [histNetwork](#).

remove.isolates

is logical. If TRUE isolates vertices are not plotted.

size

is logical. If TRUE the point size of each vertex is proportional to its degree.

labelsize

is an integer. It indicates the label size in the plot. Default is labelsize=1

label is logical. If TRUE vertex labels are plotted.  
 arrowsize is numerical. It indicates the edge arrow size.

### Details

The function [histPlot](#) can plot a historical co-citation network previously created by [histNetwork](#).

### Value

It is a network object of the class `igraph`.

### See Also

[histNetwork](#) to compute a historical co-citation network.

[cocMatrix](#) to compute a co-occurrence matrix.

[biblioAnalysis](#) to perform a bibliometric analysis.

### Examples

```
# EXAMPLE Citation network

data(scientometrics)

histResults <- histNetwork(scientometrics, n = 20, sep = ";")

net <- histPlot(histResults, size = TRUE)
```

---

idByAuthor

*Get Complete Author Information and ID from Scopus*

---

### Description

Uses SCOPUS API author search to identify author identification information.

### Usage

```
idByAuthor(df, api_key)
```

### Arguments

df is a dataframe composed of three columns:

lastname	author's last name
firstname	author's first name
affiliation	Part of the affiliation name (university name, city, etc.)

i.e. `df[1,1:3]<-c("aria","massimo","naples")` When affiliation is not specified,

the field `df$affiliation` have to be NA. i.e. `df[2,1:3]<-c("cuccurullo","corrado", NA)`

`api_key` is a character. It contains the Elsevier API key. Information about how to obtain an API Key [Elsevier API website](#)

### Value

a data frame with cases corresponding to authors and variables to author's information and ID got from SCOPUS.

### See Also

[retrievalByAuthorID](#) for downloading the complete author bibliographic collection from SCOPUS

### Examples

```
## Request a personal API Key to Elsevier web page https://dev.elsevier.com/sc_apis.html
#
# api_key="your api key"

## create a data frame with the list of authors to get information and IDs
# i.e. df[1,1:3]<-c("aria","massimo","naples")
#      df[2,1:3]<-c("cuccurullo","corrado", NA)

## run idByAuthor function
#
# authorsID <- idByAuthor(df, api_key)
```

---

isi2df

*Convert an ISI WoK Export file into a data frame*

---

### Description

It converts an ISI Wok Export file and create a data frame from it, with cases corresponding to articles and variables to Field Tag in the original file.

### Usage

```
isi2df(D)
```

### Arguments

`D` is a character array containing data read from a ISI Export file (in plain text format).

**Value**

a data frame with cases corresponding to articles and variables to Field Tag in the original ISI file.

**See Also**

[scopus2df](#) for converting SCOPUS Export file (in bibtex format)

Other converting functions: [convert2df](#), [isibib2df](#), [pubmed2df](#), [scopus2df](#)

**Examples**

```
# An ISI Export file can be read using \code{\link{readLines}} function:
# largechar <- readLines('filename.txt')

# filename.txt is an ISI Export file in plain text format.
# The file have to be saved without Byte order mark (U+FEFF) at the beginning
# and EoF code at the end of file.
# The original file (exported by ISI search web site) can be modified
# using an advanced text editor like Notepad++ or Emacs.

# scientometrics_text <- readLines('http://www.bibliometrix.org/datasets/scientometrics.txt')
# data scientometrics_text)
# scient_df <- isi2df(scientometrics_text)
```

---

isibib2df

---

*Convert an Clarivate Analytics WoS Export file into a data frame*


---

**Description**

It converts an Clarivate Analytics WoS Export file and create a data frame from it, with cases corresponding to articles and variables to Field Tag in the original file.

**Usage**

```
isibib2df(D)
```

**Arguments**

D is a character array containing data read from an WoS Export file (in bibtex format).

**Value**

a data frame with cases corresponding to articles and variables to Field Tag in the original SCOPUS file.

**See Also**

[isi2df](#) for converting ISI Export file (in plain text format)

Other converting functions: [convert2df](#), [isi2df](#), [pubmed2df](#), [scopus2df](#)

**Examples**

```
# A ISI Export file can be read using \link{readLines} function:  
  
# largechar <- readFiles('filename1.bib', 'filename2.bib2,...')  
  
# filename.bib is a Clarivate Analytics WoS Export file in plain text format.  
  
# largechar <- readFiles('http://www.bibliometrix.org/datasets/ranking.bib')  
  
# ranking <- isibib2df(largechar)
```

---

isiCollection

*"Bibliometrics" manuscripts from ISI WOS.*

---

**Description**

Manuscripts including the term "bibliometrics" in the title.

Period: 1985 - 2017

Database: [ISI Web of Knowledge](#)

Format: bibtex

**Format**

A data frame with 329 rows and 16 variables:

**AU** Authors

**TI** Document Title

**SO** Publication Name (or Source)

**JI** ISO Source Abbreviation

**DT** Document Type

**DE** Author Keywords

**ID** Keywords associated by ISI or SCOPUS database

**AB** Abstract

**C1** Author Address

**RP** Reprint Address

**CR** Cited References

**TC** Times Cited



**PY** Year  
**SC** Subject Category  
**UT** Unique Article Identifier  
**DB** Database

### Source

<http://www.webofknowledge.com>

---

keywordAssoc	<i>ID and DE keyword associations</i>
--------------	---------------------------------------

---

### Description

It associates authors' keywords to keywords plus.

### Usage

```
keywordAssoc(M, sep = ";", n = 10, excludeKW = NA)
```

### Arguments

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
sep	is the field separator character. This character separates keywords in each string of ID and DE columns of the bibliographic data frame. The default is sep = ";".
n	is a integer. It indicates the number of authors' keywords to associate to each keyword plus. The default is n = 10.
excludeKW	is character vector. It contains authors' keywords to exclude from the analysis.

### Value

an object of class "list".

### See Also

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.

[biblioAnalysis](#) function for bibliometric analysis.

[summary](#) to obtain a summary of the results.

[plot](#) to draw some useful plots of the results.

## Examples

```

data(scientometrics)

KWlist <- keywordAssoc(scientometrics, sep = ";",n = 10, excludeKW = NA)

# list of first 10 Keywords plus
names(KWlist)

# list of first 10 authors' keywords associated to the first Keyword plus
KWlist[[1]][1:10]

```

---

KeywordGrowth	<i>Yearly occurrences of top keywords/terms</i>
---------------	---

---

## Description

It calculates yearly occurrences of top keywords/terms.

## Usage

```
KeywordGrowth(M, Tag = "ID", sep = ";", top = 10, cdf = TRUE)
```

## Arguments

M	is a data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to articles and variables to Field Tag in the original ISI or SCOPUS file.
Tag	is a character object. It indicates one of the keyword field tags of the standard ISI WoS Field Tag codify (ID or DE) or a field tag created by <a href="#">termExtraction</a> function (TI_TM, AB_TM, etc.).
sep	is the field separator character. This character separates strings in each keyword column of the data frame. The default is sep = ";".
top	is a numeric. It indicates the number of top keywords to analyze. The default value is 10.
cdf	is a logical. If TRUE, the function calculates the cumulative occurrences distribution.

## Value

an object of class data.frame

## Examples

```
data(scientometrics)
topKW=KeywordGrowth(scientometrics, Tag = "ID", sep = ";", top=5, cdf=TRUE)
topKW

# Plotting results
#
# library(reshape2)
# library(ggplot2)
# DF=melt(topKW, id='Year')
# ggplot(DF,aes(Year,value, group=variable, color=variable))+geom_line()
```

---

localCitations	<i>Author local citations</i>
----------------	-------------------------------

---

## Description

It calculates local citations (LCS) of authors and documents of a bibliographic collection.

## Usage

```
localCitations(M, sep = ";")
```

## Arguments

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
sep	is the field separator character. This character separates citations in each string of CR column of the bibliographic data frame. The default is sep = ";".

## Details

Local citations measure how many times an author (or a document) included in this collection have been cited by the documents also included in the collection.

## Value

an object of class "list" containing author local citations and document local citations.

## See Also

[citations](#) function for citation frequency distribution.  
[biblioAnalysis](#) function for bibliometric analysis.  
[summary](#) to obtain a summary of the results.  
[plot](#) to draw some useful plots of the results.

**Examples**

```
data(scientometrics)

CR <- localCitations(scientometrics, sep = ";")

CR$Authors[1:10,]
CR$Papers[1:10,]
```

---

lotka	<i>Lotka's law coefficient estimation</i>
-------	---

---

**Description**

It estimates Lotka's law coefficients for scientific productivity (*Lotka A.J., 1926*)

**Usage**

```
lotka(results)
```

**Arguments**

`results` is an object of the class 'bibliometrix' for which the analysis of the authors' dominance ranking is desired.

**Value**

The function `lotka` returns a list of summary statistics of the Lotka's law estimation of an object of class `bibliometrix`.

the list contains the following objects:

Beta	Beta coefficient
C	Constant coefficient
R2	Goodness of Fit
fitted	Fitted Values
p.value	Pvalue of two-sample Kolmogorov-Smirnov test between the empirical and the theoretical Lotka's Law distribution
AuthorProd	Authors' Productivity frequency table

**See Also**

[biblioAnalysis](#) function for bibliometric analysis

[summary](#) method for class 'bibliometrix'

**Examples**

```
data(scientometrics)
```

```
results <- biblioAnalysis(scientometrics)
L=lotka(results)
L
```

---

mergeDbSources	<i>Merge bibliographic data frames from SCOPUS and ISI WOS</i>
----------------	--

---

### Description

Merge bibliographic data frames from different databases (ISI and SCOPUS) into a single one.

### Usage

```
mergeDbSources(..., remove.duplicated = TRUE)
```

### Arguments

... are the bibliographic data frames to merge.  
remove.duplicated is logical. If TRUE duplicated documents will be deleted from the bibliographic collection.

### Details

bibliographic data frames are obtained by the converting function [convert2df](#). The function merges data frames identifying common tag fields and duplicated records.

### Value

the value returned from mergeDbSources is a bibliographic data frame.

### See Also

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.  
[biblioAnalysis](#) function for bibliometric analysis.  
[summary](#) to obtain a summary of the results.  
[plot](#) to draw some useful plots of the results.

### Examples

```
data(isiCollection)
data(scopusCollection)
M <- mergeDbSources(isiCollection, scopusCollection, remove.duplicated=TRUE)
```

dim(M)

---

metaTagExtraction      *Meta-Field Tag Extraction*

---

### Description

It extracts other field tags, different from the standard ISI/SCOPUS codify.

### Usage

```
metaTagExtraction(M, Field = "CR_AU", sep = ";")
```

### Arguments

**M** is a data frame obtained by the converting function [convert2df](#). It is a data matrix with cases corresponding to articles and variables to Field Tag in the original ISI or SCOPUS file.

**Field** is a character object. New tag extracted from aggregated data is specified by this string. Field can be equal to one of this tags:

"CR_AU"	First Author of each cited reference
"CR_SO"	Source of each cited reference
"AU_CO"	Country of affiliation for each co-author
"AU1_CO"	Country of affiliation for the first author
"AU_UN"	University of affiliation for each co-author
"SR"	Short tag of the document (as used in reference lists)

**sep** is the field separator character. This character separates strings in each column of the data frame. The default is sep = ";".

### Value

the bibliometric data frame with a new column containing data about new field tag indicated in the argument Field.

### See Also

[scopus2df](#) for converting ISO or SCOPUS Export file into a data frame.

[biblioAnalysis](#) function for bibliometric analysis

### Examples

```
# Example 1: First Authors for each cited reference
```

```

data(scientometrics)
scientometrics <- metaTagExtraction(scientometrics, Field = "CR_AU", sep = ";")
unlist(strsplit(scientometrics$CR_AU[1], ";"))

#Example 2: Source for each cited reference

data(scientometrics)
scientometrics <- metaTagExtraction(scientometrics, Field = "CR_SO", sep = ";")
unlist(strsplit(scientometrics$CR_SO[1], ";"))

#Example 3: Affiliation country for co-author

data(scientometrics)
scientometrics <- metaTagExtraction(scientometrics, Field = "AU_CO", sep = ";")
scientometrics$AU_CO[1:10]

```

---

networkPlot

*Plotting Bibliographic networks*


---

## Description

networkPlot plots a bibliographic network.

## Usage

```

networkPlot(NetMatrix, normalize = NULL, n = NULL, degree = NULL,
  Title = "Plot", type = "kamada", label = TRUE, labelsize = 1,
  label.cex = FALSE, label.short = TRUE, label.n = NULL, halo = FALSE,
  cluster = "walktrap", vos.path = NULL, size = 3, curved = FALSE,
  noloops = TRUE, remove.multiple = TRUE, remove.isolates = FALSE,
  weighted = NULL, edgesize = 1, edges.min = 0)

```

## Arguments

NetMatrix	is a network matrix obtained by the function <a href="#">biblioNetwork</a> .
normalize	is a character. It can be "association", "jaccard", "inclusion", "salton" or "equivalence" to obtain Association Strength, Jaccard, Inclusion, Salton or Equivalence similarity index respectively. The default is type = NULL.
n	is an integer. It indicates the number of vertices to plot.
degree	is an integer. It indicates the min frequency of a vertex. If degree is not NULL, n is ignored.
Title	is a character indicating the plot title.

type	is a character object. It indicates the network map layout:
type="circle"	Circle layout
type="sphere"	Sphere layout
type="mds"	Multidimensional Scaling layout
type="fruchterman"	Fruchterman-Reingold layout
type="kamada"	Kamada-Kawai layout
type="vosviewer"	Network is plotted using VOSviewer software
label	is logical. If TRUE vertex labels are plotted.
labelsize	is an integer. It indicates the label size in the plot. Default is labelsize=1
label.cex	is logical. If TRUE the label size of each vertex is proportional to its degree.
label.short	is a logical. If TRUE label are plotted in short format.
label.n	is an integer. It indicates the number of vertex labels to draw.
halo	is logical. If TRUE communities are plotted using different colors. Default is halo=FALSE
cluster	is a character. It indicates the type of cluster to perform among ("none", "optimal", "lovain", "infomap", "edge_betweenness", "walktrap").
vos.path	is a character indicating the full path where VOSviewer.jar is located.
size	is integer or logical. If TRUE the point size of each vertex is proportional to its degree. If it is a integer, the point size of each vertex is constant equal to size. Default is size=3.
curved	is a logical. If TRUE edges are plotted with an optimal curvature. Default is curved=FALSE
noloops	is logical. If TRUE loops in the network are deleted.
remove.multiple	is logical. If TRUE multiple links are plotted using just one edge.
remove.isolates	is logical. If TRUE isolates vertices are not plotted.
weighted	This argument specifies whether to create a weighted graph from an adjacency matrix. If it is NULL then an unweighted graph is created and the elements of the adjacency matrix gives the number of edges between the vertices. If it is a character constant then for every non-zero matrix entry an edge is created and the value of the entry is added as an edge attribute named by the weighted argument. If it is TRUE then a weighted graph is created and the name of the edge attribute will be weight.
edgesize	is an integer. It indicates the network edge size.
edges.min	is an integer. It indicates the min frequency of edges between two vertices. If edge.min=0, all edges are plotted.

### Details

The function `networkPlot` can plot a bibliographic network previously created by `biblioNetwork`. The network map can be plotted using internal R routines or using `VOSviewer` by Nees Jan van Eck and Ludo Waltman.



**Value**

It is a list containing the following elements:

graph	a network object of the class igraph
cluster_obj	a <a href="#">communities</a> object of the package igraph
cluster_res	a data frame with main results of clustering procedure.

**See Also**

[biblioNetwork](#) to compute a bibliographic network.

[cocMatrix](#) to compute a co-occurrence matrix.

[biblioAnalysis](#) to perform a bibliometric analysis.

**Examples**

```
# EXAMPLE Co-citation network

data(scientometrics)

NetMatrix <- biblioNetwork(scientometrics, analysis = "co-citation",
network = "references", sep = ";")

net <- networkPlot(NetMatrix, n = 30, type = "kamada", Title = "Co-Citation",labelsize=0.5)
```

---

normalizeSimilarity    *Calculate similarity indices*

---

**Description**

It calculates a relative measure of bibliographic co-occurrences.

**Usage**

```
normalizeSimilarity(NetMatrix, type = "association")
```

**Arguments**

NetMatrix        is a coupling matrix obtained by the network functions [biblioNetwork](#) or [cocMatrix](#).

type             is a character. It can be "association", "jaccard", "inclusion", "salton" or "equivalence" to obtain Association Strength, Jaccard, Inclusion, Salton or Equivalence similarity index respectively. The default is type = "association".

## Details

couplingSimilarity calculates Association strength, Inclusion, Jaccard or Salton similarity from a co-occurrence bibliographic matrix.

The association strength is used by Van Eck and Waltman (2007) and Van Eck et al. (2006). Several works refer to the measure as the proximity index, while Leydesdorff (2008) and Zitt et al. (2000) refer to it as the probabilistic affinity (or activity) index.

The inclusion index, also called Simpson coefficient, is an overlap measure used in information retrieval.

The Jaccard index (or Jaccard similarity coefficient) gives us a relative measure of the overlap of two sets. It is calculated as the ratio between the intersection and the union of the reference lists (of two manuscripts).

The Salton index, instead, relates the intersection of the two lists to the geometric mean of the size of both sets. The square of Salton index is also called Equivalence index.

The indices are equal to zero if the intersection of the reference lists is empty.

## Value

a similarity matrix.

## See Also

[biblioNetwork](#) function to compute a bibliographic network.

[cocMatrix](#) to compute a bibliographic bipartite network.

## Examples

```
data(scientometrics)
NetMatrix <- biblioNetwork(scientometrics, analysis = "co-occurrences",
  network = "keywords", sep = ";")
S=normalizeSimilarity(NetMatrix, type = "association")
```

---

plot.bibliometrix      *Plotting bibliometric analysis results*

---

## Description

plot method for class 'bibliometrix'

## Usage

```
## S3 method for class 'bibliometrix'
plot(x, ...)
```

**Arguments**

`x` is the object for which plots are desired.

`...` can accept two arguments:  
`k` is an integer, used for plot formatting (number of objects). Default value is 10.  
`pause` is a logical, used to allow pause in screen scrolling of results. Default value is `pause = FALSE`.

**Value**

none. The function `plot` returns a set of plots of the object of class `bibliometric`.

**See Also**

The bibliometric analysis function [biblioAnalysis](#).  
[summary](#) to compute a list of summary statistics of the object of class `bibliometric`.

**Examples**

```
data(scientometrics)

results <- biblioAnalysis(scientometrics)

plot(results, k = 10, pause = FALSE)
```

---

pubmed2df

---

*Convert a PubMed/MedLine collection into a data frame*


---

**Description**

It converts a PubMed/MedLine collection (obtained through a query performed with `RISmed` package) and create a data frame from it, with cases corresponding to articles and variables to Field Tags as proposed by Clarivate Analytics WoS.

**Usage**

```
pubmed2df(D)
```

**Arguments**

`D` is an object of class `MedLine` (package "`RISmed`") containing data resulting of a query performed on MedLine using the package `RISmed`.

**Value**

a data frame with cases corresponding to articles and variables to Field Tags as proposed by Clarivate Analytics WoS.

**See Also**

[scopus2df](#) for converting SCOPUS Export file (in bibtex format)

[isi2df](#) for converting Clarivate Analytics WoS Export file (in plaintext format)

[isibib2df](#) for converting Clarivate Analytics WoS Export file (in bibtex format)

Other converting functions: [convert2df](#), [isi2df](#), [isibib2df](#), [scopus2df](#)

**Examples**

```
# library(RISmed)
# search_topic <- 'epidermolysis bullosa'
# search_query <- EUtilsSummary(search_topic, retmax=200, mindate=2014, maxdate=2014)
# summary(search_query)
# D <- EUtilsGet(search_query)

# M <- pubmed2df(D)
```

---

readFiles

*Load a sequence of ISI or SCOPUS Export files into a large character object*

---

**Description**

It loads a sequence of SCOPUS and Thomson Reuters' ISI Web of Knowledge export files and create a large character vector from it.

**Usage**

```
readFiles(...)
```

**Arguments**

... is a sequence of names of files downloaded from ISI WOS.(in plain text or bibtex format) or SCOPUS Export file (exclusively in bibtex format).

**Value**

a character vector of length the number of lines read.

**See Also**

[convert2df](#) for converting SCOPUS of ISI Export file into a dataframe

**Examples**

```
# ISI or SCOPUS Export files can be read using \code{\link{readFiles}} function:

# largechar <- readFiles('filename1.txt','filename2.txt','filename3.txt')

# filename1.txt, filename2.txt and filename3.txt are ISI or SCOPUS Export file
# in plain text or bibtex format.

D <- readFiles('http://www.bibliometrix.org/datasets/bibliometrics_articles.txt')
```

---

retrievalByAuthorID    *Get Author Content on SCOPUS by ID*

---

**Description**

Uses SCOPUS API search to get information about documents on a set of authors using SCOPUS ID.

**Usage**

```
retrievalByAuthorID(id, api_key, remove.duplicated = TRUE)
```

**Arguments**

`id` is a vector of characters containing the author's SCOPUS IDs. SCOPUS IDs can be obtained using the function [idByAuthor](#).

`api_key` is a character. It contains the Elsevier API key. Information about how to obtain an API Key [Elsevier API website](#)

`remove.duplicated` is logical. If TRUE duplicated documents will be deleted from the bibliographic collection.

**Value**

a list containing two objects: (i) `M` which is a data frame with cases corresponding to articles and variables to main Field Tags named using the standard ISI WoS Field Tag codify. `M` includes the entire bibliographic collection downloaded from SCOPUS. The main field tags are:

AU	Authors
TI	Document Title
SO	Publication Name (or Source)
DT	Document Type
DE	Authors' Keywords
ID	Keywords associated by SCOPUS or ISI database
AB	Abstract
C1	Author Address

RP	Reprint Address
TC	Times Cited
PY	Year
UT	Unique Article Identifier
DB	Database

(ii) `authorDocuments` which is a list containing a bibliographic data frame for each author.

**LIMITATIONS:** Currently, SCOPUS API does not allow to download document references. As consequence, it is not possible to perform co-citation analysis (the field CR is empty).

### See Also

[idByAuthor](#) for downloading author information and SCOPUS ID.

### Examples

```
## Request a personal API Key to Elsevier web page https://dev.elsevier.com/sc_apis.html

## api_key="your api key"

## create a data frame with the list of authors to get information and IDs
# i.e. df[1,1:3] <- c("aria","massimo","naples")
#       df[2,1:3] <- c("cuccurullo","corrado", "naples")

## run idByAuthor function
#
# authorsID <- idByAuthor(df, api_key)
#

## extract the IDs
#
# id <- authorsID[,3]
#

## create the bibliographic collection
#
# res <- retrievalByAuthor(id, api_key)
#
# M <- res$M # the entire bibliographic data frame
# M <- res$authorDocuments # the list containing a bibliographic data frame for each author
```

**Description**

Manuscripts about the topics "co-citation analysis" and "coupling analysis" published on Scientometrics Journal.

Period: 1985 - 2015

Database: [ISI Web of Knowledge](#)

**Format**

A data frame with 147 rows and 16 variables:

**AU** Authors

**TI** Document Title

**SO** Publication Name (or Source)

**JI** ISO Source Abbreviation

**DT** Document Type

**DE** Author Keywords

**ID** Keywords associated by ISI or SCOPUS database

**AB** Abstract

**C1** Author Address

**RP** Reprint Address

**CR** Cited References

**TC** Times Cited

**PY** Year

**SC** Subject Category

**UT** Unique Article Identifier

**DB** Database

**Source**

<http://www.webofknowledge.com>

---

scientometrics\_text    *"Co-citation analysis" and "Coupling analysis" manuscripts.*

---

**Description**

Manuscripts about the topics "co-citation analysis" and "coupling analysis" published on Scientometrics Journal.

Period: 1985 - 2015

Database: [ISI Web of Knowledge](#)

**Format**

A large character with 12731 rows.

Data has been imported by an ISI Export file in plain text format using the function [readLines](#).

**Source**

<http://www.webofknowledge.com>

---

scopus2df

*Convert a SCOPUS Export file into a data frame*

---

**Description**

It converts a SCOPUS Export file and create a data frame from it, with cases corresponding to articles and variables to Field Tag in the original file.

**Usage**

```
scopus2df(D)
```

**Arguments**

D is a character array containing data read from a SCOPUS Export file (in bibtex format).

**Value**

a data frame with cases corresponding to articles and variables to Field Tag in the original SCOPUS file.

**See Also**

[isi2df](#) for converting ISI Export file (in plain text format)

Other converting functions: [convert2df](#), [isi2df](#), [isibib2df](#), [pubmed2df](#)

**Examples**

```
# A SCOPUS Export file can be read using \link{readFiles} function:  
# largechar <- readFiles('filename1.bib', 'filename2.bib2,...')  
# filename.bib is a SCOPUS Export file in plain text format.  
#largechar <- readFiles('http://www.bibliometrix.org/datasets/scopus.bib')  
  
#scopus_df <- scopus2df(largechar)
```



---

scopusCollection      *"Bibliometrics" manuscripts from SCOPUS.*

---

**Description**

Manuscripts including the term "bibliometrics" in the title.  
Period: 1975 - 2017  
Database: **SCOPUS**  
Format: bibtex

**Format**

A data frame with 487 rows and 15 variables:

**AU** Authors

**TI** Document Title

**SO** Publication Name (or Source)

**JI** ISO Source Abbreviation

**DT** Document Type

**DE** Author Keywords

**ID** Keywords associated by ISI or SCOPUS database

**AB** Abstract

**C1** Author Address

**RP** Reprint Address

**CR** Cited References

**TC** Times Cited

**PY** Year

**UT** Unique Article Identifier

**DB** Database

**Source**

<http://www.scopus.com>

---

stopwords	<i>List of English stopwords.</i>
-----------	-----------------------------------

---

**Description**

A character vector containing a complete list of English stopwords  
 Data are used by `biblioAnalysis` function to extract Country Field of Cited References and Authors.

**Format**

A character vector with 665 rows.

---

summary.bibliometrix	<i>Summarizing bibliometric analysis results</i>
----------------------	--

---

**Description**

summary method for class 'bibliometrix'

**Usage**

```
## S3 method for class 'bibliometrix'
summary(object, ...)
```

**Arguments**

object	is the object for which a summary is desired.
...	can accept two arguments: k integer, used for table formatting (number of rows). Default value is 10. pause logical, used to allow pause in screen scrolling of results. Default value is pause = FALSE.

**Value**

The function `summary` computes and returns a list of summary statistics of the object of class `bibliometrics`.

the list contains the following objects:

MainInformation	Main Information about Data
AnnualProduction	Annual Scientific Production
AnnualGrowthRate	Annual Percentage Growth Rate
MostProdAuthors	Most Productive Authors
MostCitedPapers	Top manuscripts per number of citations

MostProdCountries	Most Productive Countries
TCperCountries	Total Citation per Countries
MostRelSources	Most Relevant Sources
MostRelKeywords	Most Relevant Keywords

**See Also**

[biblioAnalysis](#) function for bibliometric analysis

[plot](#) to draw some useful plots of the results.

**Examples**

```
data(scientometrics)

results <- biblioAnalysis(scientometrics)

summary(results)
```

---

tableTag	<i>Tabulate elements from a Tag Field column</i>
----------	--

---

**Description**

It tabulates elements from a Tag Field column of a bibliographic data frame.

**Usage**

```
tableTag(M, Tag = "CR", sep = ";")
```

**Arguments**

M	is a data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to articles and variables to Field Tag in the original ISI or SCOPUS file.
Tag	is a character object. It indicates one of the field tags of the standard ISI WoS Field Tag codify.
sep	is the field separator character. This character separates strings in each column of the data frame. The default is sep = ";".

**Details**

tableTag is an internal routine of main function [biblioAnalysis](#).

**Value**

an object of class table

## Examples

```
data(scientometrics)
Tab <- tableTag(scientometrics, Tag = "CR", sep = ";")
Tab[1:10]
```

---

termExtraction	<i>Term extraction tool from textual fields of a manuscript</i>
----------------	---

---

## Description

It extracts terms from a textual field (abstract, title, author's keywords, etc.) of a bibliographic data frame.

## Usage

```
termExtraction(M, Field = "TI", stemming = FALSE, language = "english",
  remove.numbers = TRUE, remove.terms = NULL, keep.terms = NULL,
  synonyms = NULL, verbose = TRUE)
```

## Arguments

- |                |   |      |                  |      |                     |      |                          |      |                              |
|----------------|---|------|------------------|------|---------------------|------|--------------------------|------|------------------------------|
| M              | is a data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to articles and variables to Field Tag in the original ISI or SCOPUS file.  |      |                  |      |                     |      |                          |      |                              |
| Field          | is a character object. It indicates the field tag of textual data : <table> <tr> <td>"TI"</td> <td>Manuscript title</td> </tr> <tr> <td>"AB"</td> <td>Manuscript abstract</td> </tr> <tr> <td>"ID"</td> <td>Manuscript keywords plus</td> </tr> <tr> <td>"DE"</td> <td>Manuscript author's keywords</td> </tr> </table> <p>The default is Field = "TI".</p> | "TI" | Manuscript title | "AB" | Manuscript abstract | "ID" | Manuscript keywords plus | "DE" | Manuscript author's keywords |
| "TI"           | Manuscript title  |      |                  |      |                     |      |                          |      |                              |
| "AB"           | Manuscript abstract   |      |                  |      |                     |      |                          |      |                              |
| "ID"           | Manuscript keywords plus  |      |                  |      |                     |      |                          |      |                              |
| "DE"           | Manuscript author's keywords  |      |                  |      |                     |      |                          |      |                              |
| stemming       | is logical. If TRUE the Porter Stemming algorithm is applied to all extracted terms. The default is stemming = FALSE.   |      |                  |      |                     |      |                          |      |                              |
| language       | is a character. It is the language of textual contents ("english", "german", "italian", "french", "spanish"). The default is language="english".  |      |                  |      |                     |      |                          |      |                              |
| remove.numbers | is logical. If TRUE all numbers are deleted from the documents before term extraction. The default is remove.numbers = TRUE.  |      |                  |      |                     |      |                          |      |                              |
| remove.terms   | is a character vector. It contains a list of additional terms to delete from the documents before term extraction. The default is remove.terms = NULL.  |      |                  |      |                     |      |                          |      |                              |
| keep.terms     | is a character vector. It contains a list of compound words "formed by two or more terms" to keep in their original form in the term extraction process. The default is keep.terms = NULL.  |      |                  |      |                     |      |                          |      |                              |

synonyms	is a character vector. Each element contains a list of synonyms, separated by ";", that will be merged into a single term (the first word contained in the vector element). The default is synonyms = NULL.
verbose	is logical. If TRUE the function prints the most frequent terms extracted from documents. The default is verbose=TRUE.

### Value

the bibliometric data frame with a new column containing terms about the field tag indicated in the argument Field.

### See Also

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.

[biblioAnalysis](#) function for bibliometric analysis

### Examples

```
# Example 1: Term extraction from titles

data(scientometrics)

# vector of compound words
keep.terms <- c("co-citation analysis","bibliographic coupling")

# term extraction
scientometrics <- termExtraction(scientometrics, Field = "TI",
remove.numbers=TRUE, remove.terms=NULL, keep.terms=keep.terms, verbose=TRUE)

# terms extracted from the first 10 titles
scientometrics$TI_TM[1:10]

#Example 2: Term extraction from abstracts

data(scientometrics)

# vector of terms to remove
remove.terms=c("analysis","bibliographic")

# term extraction
scientometrics <- termExtraction(scientometrics, Field = "AB", stemming=TRUE,language="english",
remove.numbers=TRUE, remove.terms=remove.terms, keep.terms=NULL, verbose=TRUE)

# terms extracted from the first abstract
scientometrics$AB_TM[1]

# Example 3: Term extraction from keywords with synonyms

data(scientometrics)
```

```
# vector of synonyms
synonyms <- c("citation; citation analysis", "h-index; index; impact factor")

# term extraction
scientometrics <- termExtraction(scientometrics, Field = "ID",
synonyms=synonyms, verbose=TRUE)
```

---

thematicEvolution      *Create an Evolution Thematic Map*

---

## Description

It creates an Evolution thematic map based on co-word network analysis and clustering. The methodology is inspired by the proposal of Cobo et al. (2011).

## Usage

```
thematicEvolution(..., weighted = FALSE)
```

## Arguments

... is a sequence of names of thematic maps created by [thematicMap](#) function.

weighted is a logical. If FALSE, a thematic nexus is measures by the classical inclusion index (calculated using the number of keywords). If TRUE, the inclusion index is calculated considering the occurrences of keywords.

## Details

[thematicEvolution](#) starts from two or more thematic maps created by [thematicMap](#) function.

## Value

a list containing:

nets	The thematic nexus graph for each comparison
incMatrix	Some useful statistics about the thematic nexus

## See Also

[thematicMap](#) function to create a thematic map based on co-word network analysis and clustering.

[cocMatrix](#) to compute a bibliographic bipartite network.

[networkPlot](#) to plot a bibliographic network.

## Examples

```

data(scientometrics)
years=c(2000)
list_df=timeslice(scientometrics, breaks = years)
M1=list_df[[1]]
M2=list_df[[2]]
NetMatrix1 <- biblioNetwork(M1, analysis = "co-occurrences",
                           network = "keywords", sep = ";")
S1 <- normalizeSimilarity(NetMatrix1, type = "association")
net1 <- networkPlot(NetMatrix1, normalize = "association",n = 50,
                  Title = "co-occurrence network",type="fruchterman",
                  labelsize = 0.7, halo = FALSE, cluster = "walktrap",remove.isolates=FALSE,
                  remove.multiple=FALSE, noloops=TRUE, weighted=TRUE)
res1 <- thematicMap(net1, NetMatrix1, S1)
#plot(res1$map)

NetMatrix2 <- biblioNetwork(M2, analysis = "co-occurrences",
                           network = "keywords", sep = ";")
S2 <- normalizeSimilarity(NetMatrix2, type = "association")
net2 <- networkPlot(NetMatrix2, normalize = "association",n = 50,
                  Title = "co-occurrence network",type="fruchterman",
                  labelsize = 0.7, halo = FALSE, cluster = "walktrap",remove.isolates=FALSE,
                  remove.multiple=FALSE, noloops=TRUE, weighted=TRUE)
res2 <- thematicMap(net2, NetMatrix2, S2)
#plot(res2$map)

nexus <- thematicEvolution(res1,res2,weighted=FALSE)

```

---

thematicMap

*Create a thematic map*


---

## Description

It creates a thematic map based on co-word network analysis and clustering. The methodology is inspired by the proposal of Cobo et al. (2011).

## Usage

```
thematicMap(Net, NetMatrix, S = NULL, minfreq = 5)
```

## Arguments

Net	is a igraph object created by <a href="#">networkPlot</a> function.
NetMatrix	is a co-occurrence matrix obtained by the network functions <a href="#">biblioNetwork</a> or <a href="#">cocMatrix</a> .
S	is a similarity matrix obtained by the <a href="#">normalizeSimilarity</a> function. If S is NULL, map is created using co-occurrence counts.
minfreq	is a integer. It indicates the minimum frequency of a cluster.

**Details**

thematicMap starts from a co-occurrence keyword network to plot in a two-dimensional map the typological themes of a domain.

**Value**

a list containing:

map	The thematic map as ggplot2 object
clusters	Centrality and Density values for each cluster.
words	A list of words following in each cluster

**See Also**

[biblioNetwork](#) function to compute a bibliographic network.

[cocMatrix](#) to compute a bibliographic bipartite network.

[networkPlot](#) to plot a bibliographic network.

**Examples**

```
data(scientometrics)
NetMatrix <- biblioNetwork(scientometrics, analysis = "co-occurrences",
  network = "keywords", sep = ";")
S <- normalizeSimilarity(NetMatrix, type = "association")
net <- networkPlot(S, n = 100, Title = "co-occurrence network", type="fruchterman",
  labelsizesize = 0.7, halo = FALSE, cluster = "walktrap", remove.isolates=FALSE,
  remove.multiple=FALSE, noloops=TRUE, weighted=TRUE)
res <- thematicMap(net, NetMatrix, S)
plot(res$map)
```

---

timeslice

*Bibliographic data frame time slice*

---

**Description**

Divide a bibliographic data frame into time slice

**Usage**

```
timeslice(M, breaks = NA, k = 5)
```



**Arguments**

M	is a bibliographic data frame obtained by the converting function <a href="#">convert2df</a> . It is a data matrix with cases corresponding to manuscripts and variables to Field Tag in the original SCOPUS and Thomson Reuters' ISI Web of Knowledge file.
breaks	is a numeric vector of two or more unique cut points.
k	is a integer value giving the number of intervals into which the data frame is to be cut. k is used only in case breaks argument is not provided. The default is k = 5.

**Value**

the value returned from `split` is a list containing the data frames for each sub-period.

**See Also**

[convert2df](#) to import and convert an ISI or SCOPUS Export file in a bibliographic data frame.

[biblioAnalysis](#) function for bibliometric analysis.

[summary](#) to obtain a summary of the results.

[plot](#) to draw some useful plots of the results.

**Examples**

```
data(scientometrics)
list_df <- timeslice(scientometrics, breaks = c(1995, 2005))
names(list_df)
```

---

trim

*Deleting leading and ending white spaces*

---

**Description**

Deleting leading and ending white spaces from a character object.

**Usage**

```
trim(x)
```

**Arguments**

x is a character object.

**Details**

tableTag is an internal routine of bibliometrics package.

**Value**

an object of class character

**Examples**

```
char <- c(" Alfred", "Mary", " John")
char
trim(char)
```

---

trim.leading	<i>Deleting leading white spaces</i>
--------------	--------------------------------------

---

**Description**

Deleting leading white spaces from a character object.

**Usage**

```
trim.leading(x)
```

**Arguments**

x is a character object.

**Details**

tableTag is an internal routine of bibliometrics package.

**Value**

an object of class character

**Examples**

```
char <- c(" Alfred", "Mary", " John")
char
trim.leading(char)
```

# Index

## \*Topic **package**

- [bibliometrix-package](#), 3
- [biblio](#), 4
- [biblio\\_df](#), 8
- [biblioAnalysis](#), 5, 7, 9, 11, 12, 15–18, 21, 25, 27–30, 33, 35, 42, 43, 45, 49
- [bibliometrix \(bibliometrix-package\)](#), 3
- [bibliometrix-package](#), 3
- [biblioNetwork](#), 6, 6, 11, 12, 20, 31–34, 47, 48
  
- [citations](#), 9, 27
- [cocMatrix](#), 7, 10, 12, 21, 33, 34, 46–48
- [communities](#), 33
- [conceptualStructure](#), 11
- [convert2df](#), 5–7, 9–12, 13, 16–20, 23–27, 29, 30, 36, 40, 43–45, 49
- [countries](#), 15
  
- [dominance](#), 15
- [duplicatedMatching](#), 16
  
- [garfield](#), 17
  
- [Hindex](#), 18
- [histNetwork](#), 19, 20, 21
- [histPlot](#), 20, 21
  
- [idByAuthor](#), 21, 37, 38
- [isi2df](#), 14, 22, 24, 36, 40
- [isibib2df](#), 14, 23, 23, 36, 40
- [isiCollection](#), 24
  
- [keywordAssoc](#), 25
- [KeywordGrowth](#), 26
  
- [localCitations](#), 27
- [lotka](#), 28
  
- [Matrix](#), 7, 10
- [mergeDbSources](#), 29
  
- [metaTagExtraction](#), 30
  
- [networkPlot](#), 31, 32, 46–48
- [normalizeSimilarity](#), 33, 47
  
- [plot](#), 5, 6, 9, 17, 18, 20, 25, 27, 29, 43, 49
- [plot.bibliometrix](#), 34
- [pubmed2df](#), 14, 23, 24, 35, 40
  
- [readFiles](#), 36
- [readLines](#), 4, 40
- [retrievalByAuthorID](#), 22, 37
  
- [scientometrics](#), 38
- [scientometrics\\_text](#), 39
- [scopus2df](#), 14, 23, 24, 30, 36, 40
- [scopusCollection](#), 41
- [stopwords](#), 42
- [summary](#), 5, 6, 9, 16–18, 20, 25, 27–29, 35, 49
- [summary.bibliometrix](#), 42
  
- [tableTag](#), 43
- [termExtraction](#), 12, 26, 44
- [thematicEvolution](#), 46, 46
- [thematicMap](#), 46, 47
- [timeslice](#), 48
- [trim](#), 49
- [trim.leading](#), 50