

# Package ‘gnumeric’

July 3, 2015

**Version** 0.7-4

**Date** 2015-07-03

**Title** Read Data from Files Readable by Gnumeric

**Author** Karoly Antal <antalk2@gmail.com>.

**Maintainer** Karoly Antal <antalk2@gmail.com>

**Depends** R (>= 2.8.1), XML

**Imports** utils

**Description** Read data files readable by gnumeric into R. Can read whole sheet or a range, from several file formats, including the native format of gnumeric. Reading is done by using `ssconvert` (a file converter utility included in the gnumeric distribution <http://projects.gnome.org/gnumeric/>) to convert the requested part to CSV. From gnumeric files (but not other formats) can list sheet names and sheet sizes or read all sheets.

**License** GPL (>= 2)

**Repository** CRAN

**Date/Publication** 2015-07-03 13:43:59

**NeedsCompilation** no

## R topics documented:

read.gnumeric.sheet . . . . .	2
read.gnumeric.sheet.info . . . . .	6
read.gnumeric.sheets . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

`read.gnumeric.sheet`     *Read data from a gnumeric (or MS Excel, Openoffice Calc, Xbase, Quatro Pro, Paradox, HTML, etc) spreadsheet or database file using `ssconvert` from the gnumeric distribution*

---

## Description

Read data from a sheet of a gnumeric (or other common spreadsheet or database) file to a data.frame.

Requires an external program, ‘`ssconvert`’ (normally installed with gnumeric in ‘`PATH`’.

(Gnumeric home page is <http://projects.gnome.org/gnumeric/>)

Calls ‘`ssconvert`’ to convert the input to CSV. ‘`ssconvert`’ can read several file formats (see Details below).

Note: During conversion to CSV ‘`ssconvert`’ also evaluates formulas (e.g. ‘`=sum(A1:A3)`’) in cells, and emits the result instead of the formula.

‘`read.gnumeric.range`’ just calls ‘`read.gnumeric.sheet`’, but uses different default values for its arguments: by default drops no rows or columns and requires at least the bottom left corner of requested gnumeric cell range to be provided.

## Usage

```
read.gnumeric.sheet(file,
                    head=FALSE,
                    sheet.name='Sheet1',
                    top.left='A1',
                    bottom.right=NA,
                    drop.empty.rows="bottom",
                    drop.empty.columns="right",
                    colnames.as.sheet=FALSE,
                    rownames.as.sheet=colnames.as.sheet,
                    quiet=TRUE,
                    LANG='C',
                    locale='C',
                    import.encoding=NA,
                    field.format='automatic',
                    ...
                    );
```

```
read.gnumeric.range(file,
                    head=FALSE,
                    sheet.name='Sheet1',
                    top.left='A1',
                    bottom.right,
                    drop.empty.rows="none",
                    drop.empty.columns="none",
                    colnames.as.sheet=FALSE,
```

```

rownames.as.sheet=colnames.as.sheet,
quiet=TRUE,
LANG='C',
locale='C',
import.encoding=NA,
field.format='automatic',
...
);

```

## Arguments

file	Name of gnumeric file (or other file type readable by gnumeric) to read from. This may also be an URL, i.e. like 'http://example.com/path/file.gnumeric'
head	When TRUE, use first row of requested gnumeric sheet range as column names in the resulting data.frame
sheet.name	Name of sheet as appears in gnumeric. Sheet names containing space or hyphen characters do not work (ssconvert reports 'Invalid range specified'). sheet.name=NA Omits sheet name from the ssconvert command line. For gnumeric files this will read the sheet that was 'current' in gnumeric when the file was saved.
top.left	Top left corner of requested gnumeric sheet range, e.g. 'A1'
bottom.right	Bottom right corner of requested gnumeric sheet range.
	The default for read.gnumeric.sheet is NA: with top.left='A1' or top.left=NA this means read full sheet. If top.left is not 'A1' or NA (i.e. when reading partial sheet), then the interpretation of bottom.right=NA falls back to 'IV65536': this causes a lot of unused lines to be printed by 'ssconvert' then parsed by read.csv, thus you might want to override it to speed up reading. Use <a href="#">read.gnumeric.sheet.info</a> to read actual bottom.right cell name from a gnumeric file (but not other formats).
drop.empty.rows	One of c('none', 'top', 'bottom', 'both', 'all'). 'all' drops all empty lines from the requested range, even those that are between two non-empty rows. 'both' drops empty lines below the last non-empty row and above the first non-empty. 'top', 'bottom' and 'none' as you would expect.
drop.empty.columns	One of c('none', 'left', 'right', 'both', 'all') Similar to drop.empty.rows, but for columns.
colnames.as.sheet	Rename columns to 'A', 'B', 'C', ... to have names corresponding to gnumeric column names.

rownames.as.sheet	Rename rows to '1', '2', '3', ... to have names corresponding to gnumeric row indices. Note: this means <code>df['1',]</code> , not <code>df[1,]</code> in the result (row-names are strings, not integers). Note: when deciding row names only top.left and head are accounted for, but not e.g. skip (which may be passed to <code>read.csv</code> via ...).
quiet	When TRUE, do not print command executed, and (on unix platforms) also redirect stderr of the external program 'ssconvert' to /dev/null
LANG	Under unix, passed to ssconvert in the environment variable 'LANG'. The default value ('C') is intended to avoid using decimal comma in the emitted CSV file. It is probably always overridden by the locale argument.
locale	Passed to ssconvert <code>-O "locale=C"</code> The default value ('C') is intended to avoid using decimal comma in the emitted CSV file.
import.encoding	If not NA, passed to ssconvert as its <code>--import-encoding</code> parameter.
field.format	Passed to ssconvert <code>-O "format=value"</code> . Allowed values: "raw", "automatic", "preserve". "raw" emits date and datetime values as number of days since an (unspecified) epoch. E.g.: <code>as.numeric(as.character(x))+as.Date('1899-12-30')</code> might work for date values and <code>as.POSIXct(as.numeric(as.character(x))*(60*60*24), origin="1899-12-29 23:59:59", tz='UTC')</code> might work for datetime values. See <code>help(as.Date)</code> for some comments on Excel epoch values.
...	Extra arguments, passed to <a href="#">read.csv</a>

## Details

Data from the gnumeric file is dumped as .csv using the 'ssconvert' program provided with gnumeric.

'ssconvert' supports several input formats, thus the input file does not have to be a gnumeric file. The formats supported may be listed with

```
ssconvert --list-importers
```

from a shell prompt.

For me this prints (with ssconvert version '1.8.4')

ID	Description
Gnumeric_xbase:xbase	Xbase (*.dbf) file format
Gnumeric_Excel:excel	MS Excel (tm) (*.xls)
Gnumeric_Excel:xlsx	MS Excel (tm) 2007
Gnumeric_html:html	HTML (*.html, *.htm)
Gnumeric_oleo:oleo	GNU Oleo (*.oleo)
Gnumeric_applix:applix	Applix (*.as)

Gnumeric_QPro:qpro	Quattro Pro (*.wb1, *.wb2, *.wb3)
Gnumeric_paradox:paradox	Paradox database or   primary index file
Gnumeric_sc:sc	SC/xspread
Gnumeric_XmlIO:sax	Gnumeric XML (*.gnumeric)
Gnumeric_lotus:lotus	Lotus 123 (*.wk1, *.wks, *.123)
Gnumeric_XmlIO:dom	Gnumeric XML (*.gnumeric) Old   slow importer
Gnumeric_dif:dif	Data Interchange Format (*.dif)
Gnumeric_Excel:excel_xml	MS Excel (tm) 2003 SpreadsheetML
Gnumeric_OpenCalc:openoffice	Open/Star Calc (*.sxc, *.ods)
Gnumeric_plan_perfect:pln	Plan Perfect Format (PLN) import
Gnumeric_sylk:sylk	MultiPlan (SYLK)
Gnumeric_mps:mps	Linear and integer program (*.mps)   file format
Gnumeric_stf:stf_csvtab	Comma or tab separated   values (CSV/TSV)
Gnumeric_stf:stf_assistant	Text import (configurable)

But the actual list may depend on which import plugins are installed for gnumeric.

Format	Source	Status
.gnumeric	gnumeric	works
.xls	gnumeric	works
.html	gnumeric '[Save as / HTML 4.0]'	works
.html	Openoffice Calc '[Save as/HTML Document]'	works
.ods	Openoffice Calc	works
Other formats		not tested

### See Also

[read.gnumeric.range](#) for a variant with default arguments more suited for reading an exact cell range of a sheet.

[read.gnumeric.sheet.info](#) to read actual bottom.right cell name from a gnumeric file (but not other formats).

[read.gnumeric.sheets](#) to read all sheets from a gnumeric file (but not other formats).

[read.xls](#) for reading Microsoft Excel files (possibly from a 'http://' URL)

[read.DIF](#) for reading Data Interchange Format (DIF) files.

[read.dbf](#) for Xbase (.dbf) files.

### Examples

```
## Read all data from 'Sheet1'
## Not run:
df <- read.gnumeric.sheet( file="file.gnumeric" );

df <- read.gnumeric.sheet( file="file.gnumeric",
```

```

sheet.name='Sheet1' );

## Read from Excel sheet named 'Sheet3' the range C3:D50,
## rename columns to 'C' and 'D', rows to '3' ... '50',
## then drop all empty rows.
##
df<-read.gnumeric.sheet( "file.xls",
                        sheet.name='Sheet3',
                        top.left='C3',
                        bottom.right='D50',
                        drop.empty.rows="all",
                        drop.empty.columns="none",
                        colnames.as.sheet=TRUE
                        )

## Read from "file.gnumeric", 'Sheet1' data in 'A1:E100',
## Use first row (of selected range) as column names.
## Drop empty rows and columns from bottom and right.
df<-read.gnumeric.sheet("file.gnumeric", head=TRUE,
                        bottom.right='E100')

## Why does it not work? Set quiet=FALSE to see
## the command executed (and on unix, diagnostic
## messages from ssconvert).
df<-read.gnumeric.sheet( "file.ods", quiet=FALSE )

## End(Not run)

```

---

```
read.gnumeric.sheet.info
```

*Read names and sizes of sheets from a gnumeric spreadsheet file*

---

### Description

Read `sheet.name`, width and height of sheets of a gnumeric file to a `data.frame`. Also constructs cell name for the `bottom.right` cells, these can be passed to [read.gnumeric.sheet](#) or [read.gnumeric.range](#)

### Usage

```
read.gnumeric.sheet.info(file);
```

### Arguments

`file`                    Name of gnumeric file to read from.

**Value**

A data.frame with columns

Column name	Type	Note
sheet.name	string	
width	integer	Number of columns
height	integer	Number of rows
bottom.right	string	Bottom right cell name or NA if sheet is empty

**Examples**

```
## Read names and sizes of sheets from 'file.gnumeric'
## Not run:
df <- read.gnumeric.sheet.info( file="file.gnumeric" );

## End(Not run)
```

---

`read.gnumeric.sheets` *Read each sheet from a gnumeric spreadsheet file*

---

**Description**

Read data from each non-empty sheet of a gnumeric file to a list of data.frames.

All arguments are passed to [read.gnumeric.sheet](#).

**Usage**

```
read.gnumeric.sheets(file,
  head=FALSE,
  drop.empty.rows="none",
  drop.empty.columns="none",
  colnames.as.sheet=FALSE,
  rownames.as.sheet=colnames.as.sheet,
  quiet=TRUE,
  LANG='C',
  locale='C',
  import.encoding=NA,
  field.format='automatic',
  ...
);
```

**Arguments**

file                   Name of gnumeric file to read from.  
head, drop.empty.rows, drop.empty.columns  
                      See [read.gnumeric.sheet](#)  
colnames.as.sheet, rownames.as.sheet, quiet  
                      See [read.gnumeric.sheet](#)  
LANG, locale, import.encoding, field.format, ...  
                      See [read.gnumeric.sheet](#)

**Value**

A list of data frames.

**See Also**

[read.gnumeric.sheet.info](#) to list sheet names and sizes.  
[read.gnumeric.range](#) to read an exact cell range of a single sheet.

**Examples**

```
## Read all sheets from 'file.gnumeric'  
## Not run:  
df.list <- read.gnumeric.sheets( file="file.gnumeric" );  
  
df1 <- df.list['Sheet1'];  
  
df.list <- read.gnumeric.sheets( file="file.gnumeric",  
                                  head=TRUE );  
  
names(df.list); ## sheet names  
  
## End(Not run)
```



# Index

## \*Topic **IO**

- read.gnumeric.sheet, [2](#)
- read.gnumeric.sheet.info, [6](#)
- read.gnumeric.sheets, [7](#)

## \*Topic **file**

- read.gnumeric.sheet, [2](#)
- read.gnumeric.sheet.info, [6](#)
- read.gnumeric.sheets, [7](#)

- read.csv, [4](#)
- read.dbf, [5](#)
- read.DIF, [5](#)
- read.gnumeric.range, [5](#), [6](#), [8](#)
- read.gnumeric.range
  - (read.gnumeric.sheet), [2](#)
- read.gnumeric.sheet, [2](#), [6–8](#)
- read.gnumeric.sheet.info, [3](#), [5](#), [6](#), [8](#)
- read.gnumeric.sheets, [5](#), [7](#)
- read.xls, [5](#)