xlsReadWrite

Andrej Blejec

24. julij 2012

1 XLConnect

```
> #setwd("C:/Users/ablejec/Documents" )
> library(XLConnect)
>
```

```
> library(xlsReadWrite)

xlsReadWrite version (cran shlib)
Copyright (C) 2010 Hans-Peter Suter, Treetron, Switzerland.

!! Your installation contains the cran placeholder shlib (dll/so).
Please get the regular shlib (420 KB) by executing the following command:

   xls.getshlib()

Info, forum, issue tracker and manual download at http://www.swissr.org.

BACKGROUND: Our own xlsReadWrite code is free, but we also use proprietary cod
(Flexcel, tmssoftware.com) which can only be distributed legally in precompile
i.e. binary form. As CRAN 'generally does not accept submissions of precompile
binaries due to security reasons' we only provide a placeholder and you can
download the binary shlib separately. NO GUARANTEES: We have done thorough tes
initially and there are integrity checks, but we do _not_ give any guarantees.
You can check/clone the source code at http://github.com/swissr/xlsreadwrite,
in case of any issues we are happy to hear about them (bug tracker/forum/email
```

```
> # pripravimo podatkovno tabelo
> n <- 4
> #bla=data.frame(faktor=LETTERS[1:n],znak=as.character(letters[1:n]),prva=1:n
> bla=data.frame(faktor=LETTERS[1:n],znak=letters[1:n],prva=1:n,druga=(1:n/n),
> dimnames(bla)[[1]]=paste("vrstica",1:n)
> bla
```

```
          faktor znak prva druga    tretja cetrta
vrstica 1      A    a    1  0.25 0.3333333   TRUE
vrstica 2      B    b    2  0.50 0.6666667  FALSE
vrstica 3      C    c    3  0.75 1.0000000   TRUE
vrstica 4      D    d    4  1.00 1.3333333  FALSE
```

```
> str(bla)

'data.frame':        4 obs. of  6 variables:
 $ faktor: Factor w/ 4 levels "A","B","C","D": 1 2 3 4
 $ znak  : Factor w/ 4 levels "a","b","c","d": 1 2 3 4
 $ prva  : int  1 2 3 4
 $ druga : num  0.25 0.5 0.75 1
 $ tretja: num  0.333 0.667 1 1.333
 $ cetrta: logi  TRUE FALSE TRUE FALSE
```

```
> #write.xls(bla,file="../data/bla.xls")
> #read.xls(file="../data/bla.xls")
```

```
> file <- "Xbla.xls"
> writeWorksheetToFile(file, data = bla,sheet="bla")
>
```

Pika ali vejica, to ni več vprašanje!

- `write.xls`: zapis je tak, da se pravilno odpre ne glede na nastavitev Excela
- `read.xls`: pravilno prebere, ne glede na to, ali je Excel nastavljen na decimalno piko ali vejico

Datoteka `blav.xls` je shranjena z vejico, v R pa se prenese brez napak.

```
> #read.xls("../data/blav.xls")
> readWorksheetFromFile (file,sheet=1 )
```

```
  faktor znak prva druga    tretja cetrta
1      A    a    1  0.25 0.3333333   TRUE
2      B    b    2  0.50 0.6666667  FALSE
3      C    c    3  0.75 1.0000000   TRUE
4      D    d    4  1.00 1.3333333  FALSE
```

2 Manjkajoče vrednosti

poglejmo še, kako se zapišejo manjkajoče vrednosti. V drugo vrstico vpišemo po vrsti izjemne vrednosti, jih zapišemo z `write.xls` in preberemo z `read.xls`:

- factor: ne zapise NA, javi napako
- character: ne zapise NA
- numeric: vcasih ? NA zapise prazno celico, prebere ko 0 zapiše in prebere NA
- numeric: NaN zapise kot NaN in pravilno prebere kot NaN
- logical: zapise kot 1/0 ter NA in prebere TRUE/FALSE in NA

```
> bla[2,3] <- NA
> bla[2,4] <- NaN
> bla[2,5] <- NA
> bla[2,6] <- NA
> bla
```

```
          faktor znak prva druga    tretja cetrta
vrstica 1      A    a    1  0.25 0.3333333   TRUE
vrstica 2      B    b   NA   NaN        NA     NA
vrstica 3      C    c    3  0.75 1.0000000   TRUE
vrstica 4      D    d    4  1.00 1.3333333  FALSE
```

```
> str(bla)
```

```
'data.frame':        4 obs. of  6 variables:
 $ faktor: Factor w/ 4 levels "A","B","C","D": 1 2 3 4
 $ znak  : Factor w/ 4 levels "a","b","c","d": 1 2 3 4
 $ prva  : int  1 NA 3 4
 $ druga : num  0.25 NaN 0.75 1
 $ tretja: num  0.333 NA 1 1.333
 $ cetrta: logi  TRUE NA TRUE FALSE
```

```
> #write.xls(bla,file="../data/bla2.xls")
> file <- "Xbla2.xls"
> writeWorksheetToFile(file, data = bla,sheet="bla")
> ble <- readWorksheetFromFile (file,sheet="bla" )
> ble
```

```
  faktor znak prva druga    tretja cetrta
1      A    a    1  0.25 0.3333333   TRUE
2      B    b   NA    NA        NA     NA
3      C    c    3  0.75 1.0000000   TRUE
4      D    d    4  1.00 1.3333333  FALSE
```

```
> str(ble)

'data.frame':        4 obs. of  6 variables:
 $ faktor: chr  "A" "B" "C" "D"
 $ znak  : chr  "a" "b" "c" "d"
 $ prva  : num  1 NA 3 4
 $ druga : num  0.25 NA 0.75 1
 $ tretja: num  0.333 NA 1 1.333
 $ cetrta: logi  TRUE NA TRUE FALSE
```

Pri branju z `read.xls` je sicer možna navedba razredov spremenljivk, vedar niso podprti vsi razredi (factor, double)

Z manjkajočimi vrednostmi je treba biti pazljiv!!!

3 RODBC

OPOZORILO: RODBC je slabo podprt za 64-bit Windows!!

Druga možnost je morda tale uporaba RODBC paketa, ki deluje dobro. Opazil sem, da argument `rownames` ne deluje tako, kot je opisano v pomoči oz. da pričakuje, da bo prvi stolpec poimenovan "rownames". Argument `rownames=0` omogoči branje tabel, ki nimajo imen vrstic.

```
> library(RODBC)
> read.xls <- function(file,...,sheet="Sheet1",cch="#",comment=FALSE,rownames=
+ z <- odbcConnectExcel(file)
+ myframe <- sqlFetch(z, sheet,rownames=rownames,colnames=!colnames,...)
+ close(z)
+ rownames(myframe) <- myframe[,1]
+ myframe <- myframe[,-1]
+ commentLines<-grep(paste("^",cch,sep=""),rownames(myframe))
+ if(!is.null(commentLines)&length(commentLines)>0&!comment) myframe <- myfram
+ if(comment) myframe <- rownames(myframe[commentLines,])
+ invisible(myframe)
+ }
```

Preberimo podatke

```
> data <- read.xls("../data/bla2.xls")
> data
```

Struktura podatkov

```
> str(data)
```

Branje z drugega lista

```
> (read.xls("../data/bla3.xls",sheet="Sheet2"))
```

Izpis dokumentacije zapisane v komentarjih

```
> (read.xls("../data/bla3.xls",sheet="Sheet2",comment=TRUE))
```

Glej: http://tolstoy.newcastle.edu.au/R/help/04/12/8638.html

4 Linux

Po navedbi avtorjev deluje **XLCOnnect** tudi na MAC in Linux.

Za Linux predlagajo uporabo perl:
http://www-106.ibm.com/developerworks/linux/library/l-pexcel/
(povezava ni preverjena)