Package 'SAScii'

July 21, 2025

Type Package

Title Import ASCII Files Directly into R using Only a 'SAS' Input Script

Version 1.0.2

Date 2023-08-28

Description Using any importation code designed for 'SAS' users to read ASCII files into 'sas7bdat' files, this package parses through the INPUT block of a '.sas' syntax file to design the parameters needed for a 'read.fwf()' function call. This allows the user to specify the location of the ASCII (often a '.dat') file and the location of the 'SAS' syntax file, and then load the data frame directly into R in just one step.

License GPL (>= 2)

URL https://github.com/ajdamico/SAScii

Depends R (>= 4.2.0)

LazyLoad Yes

NeedsCompilation no

Author Anthony Joseph Damico [aut, cre]

Maintainer Anthony Joseph Damico <ajdamico@gmail.com>

Repository CRAN

Date/Publication 2023-08-28 20:10:02 UTC

Contents

SAScii-package 2
parse.SAScii 3
read.SAScii 8

Index **17**

SAScii-package *Import ASCII files directly into R using only a SAS input script*

Description

Using importation code designed for SAS users to read ASCII files into sas7bdat files, the SAScii package parses through the INPUT block of a (.sas) syntax file to design the parameters needed for a read.fwf() function call. This allows the user to specify the location of the ASCII (often a .dat) file and the location of the .sas syntax file, and then load the data frame directly into R in just one step.

Author(s)

Anthony Joseph Damico

Maintainer: Anthony Joseph Damico <ajdamico@gmail.com>

Examples

```
## Not run:

##Load the 2009 Medical Expenditure Panel Survey Emergency Room Visits file as an R data frame

#Location of the ASCII 2009 Medical Expenditure Panel Survey Emergency Room Visits File
MEPS.09.ER.visit.file.location <-
"http://meps.ahrq.gov/mepsweb/data_files/pufs/h126edat.exe"

#Location of the SAS import instructions for the
#2009 Medical Expenditure Panel Survey Emergency Room Visits File
MEPS.09.ER.visit.SAS.read.in.instructions <-
"http://meps.ahrq.gov/mepsweb/data_stats/download_data/pufs/h126e/h126esu.txt"

#Load the 2009 Medical Expenditure Panel Survey Emergency Room Visits File
#NOTE: The SAS INPUT command occurs at line 273.
MEPS.09.ER.visit.df <-
read.SAScii ( MEPS.09.ER.visit.file.location ,
MEPS.09.ER.visit.SAS.read.in.instructions ,
zipped = T ,
beginline = 273 )

#save the data frame now for instantaneous loading later
save( MEPS.09.ER.visit.df , file = "MEPS.09.ER.visit.data.rda" )

##Load the 2011 National Health Interview Survey Persons file as an R data frame

NHIS.11.personsx.SAS.read.in.instructions <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Program_Code/NHIS/2011/personsx.sas"
NHIS.11.personsx.file.location <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Datasets/NHIS/2011/personsx.zip"
```

```
#store the NHIS file as an R data frame
NHIS.11.personsx.df <-
read.SAScii (
NHIS.11.personsx.file.location ,
NHIS.11.personsx.SAS.read.in.instructions ,
zipped = T )

#or store the NHIS SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
NHIS.11.personsx.sas <-
parse.SAScii( NHIS.11.personsx.SAS.read.in.instructions )

#save the data frame now for instantaneous loading later
save( NHIS.11.personsx.df , file = "NHIS.11.personsx.data.rda" )


##Load the 2011 National Health Interview Survey Sample Adult file as an R data frame

NHIS.11.samadult.SAS.read.in.instructions <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Program_Code/NHIS/2011/samadult.sas"
NHIS.11.samadult.file.location <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Datasets/NHIS/2011/samadult.zip"

#store the NHIS file as an R data frame!
NHIS.11.samadult.df <-
read.SAScii (
NHIS.11.samadult.file.location ,
NHIS.11.samadult.SAS.read.in.instructions ,
zipped = T )

#or store the NHIS SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
NHIS.11.samadult.sas <-
parse.SAScii( NHIS.11.samadult.SAS.read.in.instructions )

#save the data frame now for instantaneous loading later
save( NHIS.11.samadult.df , file = "NHIS.11.samadult.data.rda" )

## End(Not run)
```

`parse.SAScii` *Convert SAS import instructions into the arguments for a read.fwf function call.*

Description

Re-configure the INPUT block of a (.sas) syntax file into the arguments necessary to run the read.fwf function on an ASCII data set.

Usage

```
parse.SAScii( sas_ri, beginline = 1 , lrecl = NULL )
```

Arguments

sas_ri	Character string containing location of SAS import instructions.
beginline	Line number in SAS import instructions where the INPUT statement begins. If the word INPUT appears before the actual INPUT block, the function will return an error.
lrecl	LRECL option from SAS code. Only necessary if the width of the ASCII file is longer than the actual columns containing data (if the file contains empty space on the right side)

Details

This function cannot handle overlapping columns. For example, in the 2009 National Ambulatory Medical Care Survey (NAMCS) SAS import instructions, columns DIAG1 and DIAG13D will create an error because both start at space 55.
ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/dataset_documentation/namcs/sas/nam09inp.txt.

Value

An R data frame containing:

varname	The name of the variable field
width	The width of the field
char	A logical flag indicating a character field if T and numeric if F
divisor	A fraction to later be multiplied by numeric fields containing decimal points

Author(s)

Anthony Joseph Damico

Examples

```
############
#Some Data#
############

#write an example ASCII data set
some.data <-
"0154hello2304coolgreatZZ
2034puppy0023nicesweetok
9900buddy4495    swell!!"

#create temporary ASCII file
some.data.tf <- tempfile()
#write the sas code above to that temporary file
```

```
writeLines ( some.data , con = some.data.tf )

#write an example SAS import script using the @ method
sas.import.with.at.signs <-
"INPUT
@1 NUMBERS1 4.2
@5 WORDS1 $ 5.
@10 NUMBERS2 2.0
@12 NUMBERS3 2.0
@14 WORDS2 $4.
@18 WORDS3 $5
@23 WORDS4 $ 1
@24 WORDS5 $ 1
;"


#create a temporary file
sas.import.with.at.signs.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( sas.import.with.at.signs , con = sas.import.with.at.signs.tf )

parse.SAScii( sas.import.with.at.signs.tf )

#write an example SAS import script using the dash method
sas.import.with.lengths <-
"INPUT
NUMBERS1 1 - 4 .2
WORDS1 $ 5-9
NUMBERS2 10 -11
NUMBERS3 12- 13 .0
WORDS2 $14-17
WORDS3$ 18-22
WORDS4   $   23-23
WORDS5 $24
;"

#create a temporary file
sas.import.with.lengths.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( sas.import.with.lengths , con = sas.import.with.lengths.tf )

parse.SAScii( sas.import.with.lengths.tf )


##########################################
#Survey of Income and Program Participation#
##########################################

#first fourteen lines pulled from the Survey of Income and Program Participation
#http://smpbff2.dsd.census.gov/pub/sipp/2008/l08puw1.sas

sipp.sas <-
"LIBNAME sas8 v8 'current directory';
```

```
FILENAME INPUTDAT 'l08puw1.dat' ;
DATA sas8.l08puw1 ;
INFILE INPUTDAT PAD LRECL=2341 ;
INPUT
 SSUSEQ 1- 5
 SSUID $ 6- 17
 SPANEL 18- 21
 SWAVE 22- 23
 SROTATON 24- 24
 SREFMON 25- 25
 RHCALMN 26- 27
 RHCALYR 28- 31
 SHHADID 32- 34
 GVARSTR 35- 37
 GHLFSAM 38- 38
 GRGC $ 39- 41
 TFIPSST 42- 43
 TMOVRFLG 44- 45
; RUN;"

#create a temporary file
sipp.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( sipp.sas , con = sipp.tf )
#parse that temporary file
sipp.fwf.parameters <- parse.SAScii( sipp.tf , beginline = 5 )
#print the results to the screen
sipp.fwf.parameters

##################################################
#NATIONAL HEALTH INTERVIEW SURVEY - IMPUTED INCOME#
##################################################

#INPUT lines pulled from the 2011 National Health Interview Survey's Imputed Income file
#ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Datasets/NHIS/2011_imputed_income/incmimp.sas

nhis.incmimp.sas <-
"   * INPUT ALL VARIABLES;

   INPUT
  RECTYPE  1-2   SRVY_YR  3-6
  HHX   7-12  FMX   13-14
  FPX   15-16  IMPNUM   17
  FAMINCF2 18   TCINCM_F  19
  FAMINCI2  20-25  POVRATI3  26-34
  EMPLOY_F  35    EMPLOY_I  36
  ERNYR_F  37    TCEARN_F  38
  ERNYR_I2  39-44
  ;

   * DEFINE VARIABLE LABELS;"

#create a temporary file
```

```
nhis.incmimp.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( nhis.incmimp.sas , con = nhis.incmimp.tf )
#parse that temporary file
nhis.incmimp.fwf.parameters <- parse.SAScii( nhis.incmimp.tf )
#print the results to the screen
nhis.incmimp.fwf.parameters


################################################
#NATIONAL HEALTH INTERVIEW SURVEY - IMPUTED INCOME#
################################################

#INPUT lines pulled from the 2011 National Health Interview Survey's Sample Adult file
#ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Program_Code/NHIS/2011/samadult.sas

nhis.samadult.sas <-
"   * INPUT ALL VARIABLES;

   INPUT

  /* IDN LOCATIONS */

  RECTYPE       1 -   2    SRVY_YR       3 -   6
  HHX      $    7 -  12    INTV_QRT     13 -  13
  ASSIGNWK     14 -  15    FMX      $   16 -  17
  FPX      $   18 -  19    WTIA_SA      20 -  26 .1
  WTFA_SA      27 -  32

  /* UCF LOCATIONS */

  REGION       33 -  33    STRAT_P      34 -  36
  PSU_P        37 -  38
;"

#create a temporary file
nhis.samadult.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( nhis.samadult.sas , con = nhis.samadult.tf )
#parse that temporary file
nhis.samadult.fwf.parameters <- parse.SAScii( nhis.samadult.tf )
#print the results to the screen
nhis.samadult.fwf.parameters


## Not run:

#############################################################################################
#Create the read.fwf parameters required to load the
#2009 Medical Expenditure Panel Survey Emergency Room Visits file

#Location of the SAS import instructions for the
#2009 Medical Expenditure Panel Survey Emergency Room Visits File
```

```
MEPS.09.ER.visit.SAS.read.in.instructions <-
"http://meps.ahrq.gov/mepsweb/data_stats/download_data/pufs/h126e/h126esu.txt"

#Load the 2009 Medical Expenditure Panel Survey Emergency Room Visits File
#NOTE: The SAS INPUT command occurs at line 273.
MEPS.09.ER.visit.sas <-
parse.SAScii ( MEPS.09.ER.visit.SAS.read.in.instructions , beginline = 273 )


################################################################################################
#Create the read.fwf parameters required to load the
#2011 National Health Interview Survey Persons file

NHIS.11.personsx.SAS.read.in.instructions <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Program_Code/NHIS/2011/personsx.sas"

#store the NHIS SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
NHIS.11.personsx.sas <-
parse.SAScii( NHIS.11.personsx.SAS.read.in.instructions )

## End(Not run)
```

`read.SAScii` *Create an R data frame by reading in an ASCII file and SAS import instructions*

Description

Using importation code designed for SAS users to read ASCII files into sas7bdat files, the read.SAScii function parses through the INPUT block of a (.sas) syntax file to design the parameters needed for a read.fwf function call, and then runs that command. This allows the user to specify the location of the ASCII (often a .dat) file and the location of the .sas syntax file, and then load the data frame directly into R in just one step.

Usage

```
read.SAScii( fn,
sas_ri,
beginline = 1,
buffersize = 50,
zipped = F ,
n = -1 ,
intervals.to.print = 1000 ,
lrecl = NULL ,
skip.decimal.division = NULL
)
```

Arguments

fn	Character string containing location of ASCII filename (or if zipped = T, a filename ending in .zip).
sas_ri	Character string containing location of SAS import instructions.
beginline	Line number in SAS import instructions where the INPUT statement begins. If the word INPUT appears before the actual INPUT block, the function will return an error.
buffersize	Maximum number of lines to read at one time, passed to read.fwf().
zipped	Flag noting if ASCII file should be unzipped / decompressed before loading. Useful when downloading larger data sets directly from a website.
n	the maximum number of records (lines) to be passed to read.fwf(), defaulting to no limit.
intervals.to.print	the number of records to wait before printing current progress to the screen.
lrecl	LRECL option from SAS code. Only necessary if the width of the ASCII file is longer than the actual columns containing data (if the file contains empty space on the right side)
skip.decimal.division	whether numeric columns should be divided based on how many decimal places are specified by the SAS import instructions. recommended: ignore this parameter (or set it to NULL) to let the function attempt to determine whether numeric columns have already been divided to hit the appropriate number of decimal places or not. TRUE tells read.SAScii to not perform any decimal-related division of numeric columns. FALSE tells read.SAScii to perform decimal-related division according to the SAS import instructions, ignoring the presence of numeric fields that already contain decimals.

Details

This function cannot handle overlapping columns. For example, in the 2009 National Ambulatory Medical Care Survey (NAMCS) SAS import instructions, columns DIAG1 and DIAG13D will create an error because both start at space 55.
ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/dataset_documentation/namcs/sas/nam09inp.txt.

Value

A data.frame as produced by read.fwf() which is called internally.

Note

Some of the commands below take days to run, depending on your machine. If you need the Survey of Income and Program Participation, start the program before you quit working for the weekend.

Author(s)

Anthony Joseph Damico

Examples

```
############
#Some Data#
############

#write an example ASCII data set
some.data <- "0154hello2304coolgreatZZ\n2034puppy0023nicesweetok\n9900buddy4495   swell!!"

#create temporary ASCII file
some.data.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( some.data , con = some.data.tf )

#write an example SAS import script using the at method
sas.import.with.at.signs <-
"INPUT
@1 NUMBERS1 4.2
@5 WORDS1 $ 5.
@10 NUMBERS2 2.0
@12 NUMBERS3 2.0
@14 WORDS2 $4.
@18 WORDS3 $5
@23 WORDS4 $ 1
@24 WORDS5 $ 1
;"


#create a temporary file
sas.import.with.at.signs.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( sas.import.with.at.signs , con = sas.import.with.at.signs.tf )

parse.SAScii( sas.import.with.at.signs.tf )

#using at signs sas script
read.SAScii( some.data.tf , sas.import.with.at.signs.tf )


#write an example SAS import script using the dash method
sas.import.with.lengths <-
"INPUT
NUMBERS1 1 - 4 .2
WORDS1 $ 5-9
NUMBERS2 10 -11
NUMBERS3 12- 13 .0
WORDS2 $14-17
WORDS3$ 18-22
WORDS4   $   23-23
```

```
WORDS5 $24
;"

#create a temporary file
sas.import.with.lengths.tf <- tempfile()
#write the sas code above to that temporary file
writeLines ( sas.import.with.lengths , con = sas.import.with.lengths.tf )

parse.SAScii( sas.import.with.lengths.tf )

#using dash method sas script
read.SAScii( some.data.tf , sas.import.with.lengths.tf )

## Not run:


################################################################################################
#Load the 2009 Medical Expenditure Panel Survey Emergency Room Visits file as an R data frame

#Location of the ASCII 2009 Medical Expenditure Panel Survey Emergency Room Visits File
MEPS.09.ER.visit.file.location <-
"http://meps.ahrq.gov/mepsweb/data_files/pufs/h126edat.exe"

#Location of the SAS import instructions for the
#2009 Medical Expenditure Panel Survey Emergency Room Visits File
MEPS.09.ER.visit.SAS.read.in.instructions <-
"http://meps.ahrq.gov/mepsweb/data_stats/download_data/pufs/h126e/h126esu.txt"

#Load the 2009 Medical Expenditure Panel Survey Emergency Room Visits File
#NOTE: The SAS INPUT command occurs at line 273.
MEPS.09.ER.visit.df <-
read.SAScii (
MEPS.09.ER.visit.file.location ,
MEPS.09.ER.visit.SAS.read.in.instructions ,
zipped = T ,
beginline = 273 )

#save the data frame now for instantaneous loading later
save( MEPS.09.ER.visit.df , file = "MEPS.09.ER.visit.data.rda" )


################################################################################################
#Load the 2011 National Health Interview Survey Persons file as an R data frame

NHIS.11.personsx.SAS.read.in.instructions <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Program_Code/NHIS/2011/personsx.sas"
NHIS.11.personsx.file.location <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Datasets/NHIS/2011/personsx.zip"

#store the NHIS file as an R data frame!
NHIS.11.personsx.df <-
read.SAScii (
NHIS.11.personsx.file.location ,
```

```
NHIS.11.personsx.SAS.read.in.instructions ,
zipped = T )

#or store the NHIS SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
NHIS.11.personsx.sas <- parse.SAScii( NHIS.11.personsx.SAS.read.in.instructions )

#save the data frame now for instantaneous loading later
save( NHIS.11.personsx.df , file = "NHIS.11.personsx.data.rda" )


##############################################################################################
#Load the 2011 National Health Interview Survey Sample Adult file as an R data frame

NHIS.11.samadult.SAS.read.in.instructions <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Program_Code/NHIS/2011/SAMADULT.sas"
NHIS.11.samadult.file.location <-
"ftp://ftp.cdc.gov/pub/Health_Statistics/NCHS/Datasets/NHIS/2011/samadult.zip"

#store the NHIS file as an R data frame!
NHIS.11.samadult.df <-
read.SAScii (
NHIS.11.samadult.file.location ,
NHIS.11.samadult.SAS.read.in.instructions ,
zipped = T )

#or store the NHIS SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
NHIS.11.samadult.sas <- parse.SAScii( NHIS.11.samadult.SAS.read.in.instructions )

#save the data frame now for instantaneous loading later
save( NHIS.11.samadult.df , file = "NHIS.11.samadult.data.rda" )


##############################################################################################
#Load an IPUMS - American Community Survey Extract into R

#DOES NOT RUN without downloading ACS ASCII files to
#your local drive from http://www.ipums.org/

#MINNESOTA POPULATION CENTER - IPUMS ASCII EXTRACTS & SAS import instructions
IPUMS.file.location <- "./IPUMS/usa_00001.dat"
IPUMS.SAS.read.in.instructions <- "./IPUMS/usa_00001.sas"

#store the IPUMS extract as an R data frame!
IPUMS.df <-
read.SAScii (
IPUMS.file.location ,
IPUMS.SAS.read.in.instructions ,
zipped = F )

#or store the IPUMS extract SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
```

```
IPUMS.sas <- parse.SAScii( IPUMS.SAS.read.in.instructions )


##############################################################################################
#Load the Current Population Survey -
#Annual Social and Economic Supplement - March 2011 as an R data frame

#census.gov website containing the current population survey's main file
CPS.ASEC.mar11.file.location <-
"http://smpbff2.dsd.census.gov/pub/cps/march/asec2011_pubuse.zip"
CPS.ASEC.mar11.SAS.read.in.instructions <-
"http://www.nber.org/data/progs/cps/cpsmar11.sas"

#create a temporary file and a temporary directory..
tf <- tempfile() ; td <- tempdir()
#download the CPS repwgts zipped file
download.file( CPS.ASEC.mar11.file.location , tf , mode = "wb" )
#unzip the file's contents and store the file name within the temporary directory
fn <- unzip( tf , exdir = td , overwrite = T )

#the CPS March Supplement ASCII/FWF contains household-, family-, and person-level records.
#throw out records that are not person-level.
#according to the SAS import instructions, person-level record lines begin with a "3"

#create a second temporary file
tf.sub <- tempfile()

input <- fn
output <- tf.sub

incon <- file(input, "r")
outcon <- file(output, "w")

#cycle through every line in the downloaded CPS file..
while(length(line <- readLines(incon, 1))>0){
#and if the first letter is a 3, add it to the new person-only CPS file.
if ( substr( line , 1 , 1 ) == "3" ){
writeLines(line,outcon)
}
}
close(outcon)
close(incon , add = T)

#the SAS file produced by the National Bureau of Economic Research (NBER)
#begins the person-level INPUT after line 1209,
#so skip SAS import instruction lines before that.
#NOTE that the beginline of 1209 will change for different years.

#store the CPS ASEC March 2011 file as an R data frame!
cps.asec.mar11.df <-
read.SAScii (
tf.sub ,
CPS.ASEC.mar11.SAS.read.in.instructions ,
```

```
beginline = 1209 ,
zipped = F )

#or store the CPS ASEC March 2011 SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
cps.asec.mar11.sas <-
parse.SAScii( CPS.ASEC.mar11.SAS.read.in.instructions , beginline = 1209 )


##############################################################################################
#Load the Replicate Weights file of the Current Population Survey
#March 2011 as an R data frame

#census.gov website containing the current population survey's replicate weights file
CPS.replicate.weight.file.location <-
"http://smpbff2.dsd.census.gov/pub/cps/march/CPS_ASEC_ASCII_REPWGT_2011.zip"
CPS.replicate.weight.SAS.read.in.instructions <-
"http://smpbff2.dsd.census.gov/pub/cps/march/CPS_ASEC_ASCII_REPWGT_2011.SAS"

#store the CPS repwgt file as an R data frame!
cps.repwgt.df <-
read.SAScii (
CPS.replicate.weight.file.location ,
CPS.replicate.weight.SAS.read.in.instructions ,
zipped = T )

#or store the CPS repwgt SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
cps.repwgt.sas <- parse.SAScii( CPS.replicate.weight.SAS.read.in.instructions )


##############################################################################################
#Load the 2008 Survey of Income and Program Participation Wave 1 as an R data frame
SIPP.08w1.SAS.read.in.instructions <-
"http://smpbff2.dsd.census.gov/pub/sipp/2008/l08puw1.sas"
SIPP.08w1.file.location <-
"http://smpbff2.dsd.census.gov/pub/sipp/2008/l08puw1.zip"

#store the SIPP file as an R data frame

#note the text "INPUT" appears before the actual INPUT block of the SAS code
#so the parsing of the SAS instructions will fail without a beginline parameter specifying
#where the appropriate INPUT block occurs

SIPP.08w1.df <-
read.SAScii (
SIPP.08w1.file.location ,
SIPP.08w1.SAS.read.in.instructions ,
beginline = 5 ,
buffersize = 10 ,
zipped = T )

#or store the SIPP SAS import instructions for use in a
```

```
#read.fwf function call outside of the read.SAScii function
SIPP.08w1.sas <- parse.SAScii( SIPP.08w1.SAS.read.in.instructions , beginline = 5 )


###############################################################################################
#Load the Replicate Weights file of the
#2008 Survey of Income and Program Participation Wave 1 as an R data frame
SIPP.repwgt.08w1.SAS.read.in.instructions <-
"http://smpbff2.dsd.census.gov/pub/sipp/2008/rw08wx.sas"
SIPP.repwgt.08w1.file.location <-
"http://smpbff2.dsd.census.gov/pub/sipp/2008/rw08w1.zip"

#store the SIPP file as an R data frame

#note the text "INPUT" appears before the actual INPUT block of the SAS code
#so the parsing of the SAS instructions will fail without a beginline parameter specifying
#where the appropriate INPUT block occurs

SIPP.repwgt.08w1.df <-
read.SAScii (
SIPP.repwgt.08w1.file.location ,
SIPP.repwgt.08w1.SAS.read.in.instructions ,
beginline = 5 ,
zipped = T )

#store the SIPP SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
SIPP.repwgt.08w1.sas <-
parse.SAScii(
SIPP.repwgt.08w1.SAS.read.in.instructions ,
beginline = 5 )


###############################################################################################
#Load all twelve waves of the 2004 Survey of Income and Program Participation as R data frames

SIPP.04w1.SAS.read.in.instructions <-
"http://smpbff2.dsd.census.gov/pub/sipp/2004/l04puw1.sas"

#store the SIPP SAS import instructions for use in a
#read.fwf function call outside of the read.SAScii function
SIPP.04w1.sas <- parse.SAScii( SIPP.04w1.SAS.read.in.instructions , beginline = 5 )

#note the text "INPUT" appears before the actual INPUT block of the SAS code
#so the parsing of the SAS instructions will fail without a beginline parameter specifying
#where the appropriate INPUT block occurs

#loop through all 12 waves of SIPP 2004
for ( i in 1:12 ){

SIPP.04wX.file.location <-
paste(
"http://smpbff2.dsd.census.gov/pub/sipp/2004/l04puw" ,
```

```
i ,
".zip" ,
sep = ""
)

#name the data frame based on the current wave
df.name <- paste( "SIPP.04w" , i , ".df" , sep = "" )

#store the SIPP file as an R data frame!
assign(

df.name ,

read.SAScii (
SIPP.04wX.file.location ,
SIPP.04w1.SAS.read.in.instructions ,
beginline = 5 ,
buffersize = 5 ,
zipped = T )
)

}

## End(Not run)
```

Index

parse.SAScii, 3

read.SAScii, 8

SAS.uncomment *(*parse.SAScii*)*, 3
SAScii *(*SAScii-package*)*, 2
SAScii-package, 2

withWarnings *(*read.SAScii*)*, 8