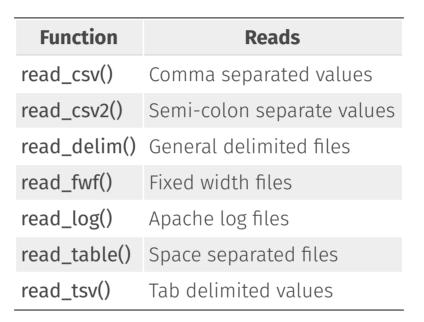
Reading and Writing Data

readr and haven

2021-10-23

readr





Importing Data

```
dataset <- read_csv("file_name.csv")
dataset</pre>
```

R functions

$$x < - f(arg = 1)$$

R functions

R functions

```
this saves it in your
    global environment
x < - f(arg = 1)
           assign results of
             f() to x
the name of
your results
```

Find diabetes.csv on your computer. Then read it into an object. Then view the results.

Find diabetes.csv on your computer. Then read it into an object. Then view the results.

```
diabetes <- read_csv("diabetes.csv")</pre>
```



new data alert!



diabetes

Where does it come from?

diabetes.csv (etc)
study: diabetes in
African Americans

How can I use it?

diabetes < readr::read_csv("diabetes.csv")
View(diabetes)</pre>



this saves it in your global environment

diabetes

```
## # A tibble: 403 × 19
4F4F
        id chol stab.glu hdl ratio glyhb location
                                                       age
     <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dr>
##
                                                     <dbl>
###
      1000
             203
                      82
                            56
                                3.60 4.31 Buckingham
                                                        46
   1
                      97
                                                        29
##
      1001 165
                            24 6.90 4.44 Buckingham
##
   3
      1002
            228
                      92
                            37 6.20 4.64 Buckingham
                                                        58
                                    4.63 Buckingham
##
      1003
            78
                      93
                            12 6.5
                                                        67
            249
                            28 8.90 7.72 Buckingham
##
   5
      1005
                      90
                                                        64
                      94
##
      1008
            248
                            69 3.60 4.81 Buckingham
                                                        34
           195
                            41 4.80 4.84 Buckingham
##
      1011
                      92
                                                        30
                            44 5.20 3.94 Buckingham
##
      1015
            227
                      75
                                                        37
           177
                            49 3.60 4.84 Buckingham
##
      1016
                      87
                                                        45
                            40 6.60 5.78 Buckingham
## 10
      1022 263
                      89
                                                        55
## # ... with 393 more rows, and 11 more variables:
## # gender <chr>, height <dbl>, weight <dbl>, frame <chr>,
      bp.1s <dbl>, bp.1d <dbl>, ...
4F4F 4F
```

Tibbles

data.frames are the basic form of rectangular data in R (columns of variables, rows of observations)

Tibbles

data.frames are the basic form of rectangular data in R (columns of variables, rows of observations"

read_csv() reads the data into a tibble, a modern version of the data frame.

Tibbles

data.frames are the basic form of rectangular data in R (columns of variables, rows of observations"

read_csv() reads the data into a tibble, a modern version of the data frame.

a tibble is a data frame

Missing values

It's common to use codes for missing values (-99, 8888)

Missing values

It's common to use codes for missing values (-99, 8888)

The na option can change these values to NA

```
read_csv(
   "a,b,c,d
   1,-99,3,4
   5,6,-99,8",
   na = "-99"
)

### # A tibble: 2 × 4

### a b c d

### <dbl> <dbl> <dbl> <dbl> <dbl> |
### 1 1 NA 3 4

### 2 5 6 NA 8
```

The read functions in readr try to guess each data type, but sometimes it's wrong

The read functions in readr try to guess each data type, but sometimes it's wrong

To tell readr how to parse the columns, add the argument col_types to read_csv()

The read functions in readr try to guess each data type, but sometimes it's wrong

To tell readr how to parse the columns, add the argument col_types to read_csv()

```
diabetes <- read_csv(
   "diabetes.csv",
   col_types = list(id = col_character())
)</pre>
```

Or use a string for each variable type:

col_type = "cci"

Or use a string for each variable type: col_type = "cci"

letter	type
С	character
i	integer
n	number
d	double
l	logical
D	date
Т	date time
t	time
?	guess the type
_ or -	skip the column

Set the 4 column types to be: integer, double, character, and unknown (guess)

```
read_csv(
   "a,b,c,d
   1,2,3,4
   5,6,7,8",
   col_types = ""
)
```

<int> <dbl> <chr> <dbl>

1 1 2 3 ## 2 5 6 7

##

Set the 4 column types to be: integer, double, character, and unknown (guess)

```
read_csv(
    "a,b,c,d
    1,2,3,4
    5,6,7,8",
    col_types = "idc?"
)

### # A tibble: 2 × 4
### a b c d
```

haven

Function	Software
read_sas()	SAS
read_xpt()	SAS
read_spss()	SPSS
read_sav()	SPSS
read_por()	SPSS
read_stata()	Stata
read_dta()	Stata



haven





haven is not a core member of the tidyverse. That means you need to load it with library(haven).

There are several versions of the diabetes file besides CSV. Pick a file format you or your colleagues use and import them using the corresponding function from haven.

```
library(haven)
diabetes <- read_sas("diabetes.sas7bdat")</pre>
```

diabetes

```
## # A tibble: 403 × 19
4F4F
        id chol stab_glu
                           hdl ratio glyhb location
                                                       age
##
     <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dr>
                                                     <dbl>
##
      1000
             203
                      82
                                3.60 4.31 Buckingham
                                                        46
                            56
   1
##
      1001 165
                      97
                            24 6.90 4.44 Buckingham
                                                        29
##
      1002 228
                      92
                            37 6.20 4.64 Buckingham
                                                        58
##
      1003 78
                      93
                            12 6.5 4.63 Buckingham
                                                        67
##
      1005 249
                      90
                            28 8.90 7.72 Buckingham
                                                        64
##
      1008
           248
                      94
                            69 3.60 4.81 Buckingham
                                                        34
      1011 195
                      92
                            41 4.80 4.84 Buckingham
                                                        30
##
      1015 227
                      75
                            44 5.20 3.94 Buckingham
                                                        37
##
      1016 177
                      87
                            49 3.60 4.84 Buckingham
                                                        45
##
4F4F
  10
      1022 263
                      89
                            40
                                6.60 5.78 Buckingham
                                                        55
## # ... with 393 more rows, and 11 more variables:
## # gender <chr>, height <dbl>, weight <dbl>, frame <chr>,
      bp 1s <dbl>, bp 1d <dbl>, ...
### ##
```

Writing data

Function	Writes
write_csv()	Comma separated values
write_excel_csv()	CSV that you plan to open in Excel
write_delim()	General delimited files
write_file()	A single string, written as is
write_lines()	A vector of strings, one string per line
write_tsv()	Tab delimited values
write_rds()	A data type used by R to save objects
write_sas()	SAS .sas7bdat files
write_xpt()	SAS transport format, .xpt
write_sav()	SPSS .sav files
write_stata()	Stata .dta files

Writing data

Function	Writes
write_csv()	Comma separated values
write_excel_csv()	CSV that you plan to open in Excel
write_delim()	General delimited files
write_file()	A single string, written as is
write_lines()	A vector of strings, one string per line
write_tsv()	Tab delimited values
write_rds()	A data type used by R to save objects
write_sas()	SAS .sas7bdat files
write_xpt()	SAS transport format, .xpt
write_sav()	SPSS .sav files
write_stata()	Stata .dta files

write_csv(diabetes, file = "diabetes-clean.csv")

R has a few data file types, such as RDS and .Rdata. Save diabetes as "diabetes.Rds".

R has a few data file types, such as RDS and .Rdata. Save diabetes as "diabetes.Rds".

```
write rds(diabetes, "diabetes.Rds")
```