S2_IO_in_R

Enrique Audain Martinez

2023-09-14

Load libraries

```
library(openxlsx)
library(microbenchmark)
```

Import/export data into/from R

1. Import data

```
# 1. Import data

# Import data from a csv file
df <- read.csv("../../data/chd_genes.annotations.csv",
               sep = ",",
               header = TRUE)

# print the head of the data frame
head(df)
```

```
##   gene_symbol     category        pLI  plof gene_length obs_lof obs_syn exp_lof
## 1       ABCC9    syndromic 9.3524e-09 0.482      144002      30     298  84.399
## 2        ABL1    syndromic 9.9998e-01 0.176      173730       3     325  44.108
## 3       ACAD9    syndromic 4.5256e-08 0.814       36472      17     147  31.321
## 4       ACTA2 nonsyndromic 9.3017e-01 0.364       56317       2      72  17.293
## 5        ACTB    syndromic 9.8564e-01 0.232       36634       0     190  12.858
## 6       ACTC1 nonsyndromic 7.3668e-01 0.480        8044       2      89  13.125
##   exp_syn chromosome
## 1 298.160         12
## 2 314.370          9
## 3 144.410          3
## 4  84.674         10
## 5  96.859          7
## 6  89.788         15
```

```
# Import data using the read.table function
df <- read.table("../../data/chd_genes.annotations.tsv",
                 sep = "\t",
                 header = TRUE)
```

```
# print the head of the data frame
head(df)
```

```
##   gene_symbol     category        pLI  plof gene_length obs_lof obs_syn exp_lof
## 1       ABCC9    syndromic 9.3524e-09 0.482      144002      30     298  84.399
## 2        ABL1    syndromic 9.9998e-01 0.176      173730       3     325  44.108
## 3       ACAD9    syndromic 4.5256e-08 0.814       36472      17     147  31.321
## 4       ACTA2 nonsyndromic 9.3017e-01 0.364       56317       2      72  17.293
## 5        ACTB    syndromic 9.8564e-01 0.232       36634       0     190  12.858
## 6       ACTC1 nonsyndromic 7.3668e-01 0.480        8044       2      89  13.125
##   exp_syn chromosome
## 1 298.160         12
## 2 314.370          9
## 3 144.410          3
## 4  84.674         10
## 5  96.859          7
## 6  89.788         15
```

```
# Import data using the read.delim function
df <- read.delim("../../data/chd_genes.annotations.tsv",
                 sep = "\t",
                 header = TRUE)

# print the head of the data frame
head(df)
```

```
##   gene_symbol     category        pLI  plof gene_length obs_lof obs_syn exp_lof
## 1       ABCC9    syndromic 9.3524e-09 0.482      144002      30     298  84.399
## 2        ABL1    syndromic 9.9998e-01 0.176      173730       3     325  44.108
## 3       ACAD9    syndromic 4.5256e-08 0.814       36472      17     147  31.321
## 4       ACTA2 nonsyndromic 9.3017e-01 0.364       56317       2      72  17.293
## 5        ACTB    syndromic 9.8564e-01 0.232       36634       0     190  12.858
## 6       ACTC1 nonsyndromic 7.3668e-01 0.480        8044       2      89  13.125
##   exp_syn chromosome
## 1 298.160         12
## 2 314.370          9
## 3 144.410          3
## 4  84.674         10
## 5  96.859          7
## 6  89.788         15
```

```
# End of the section
```

2. Export data

```r
# 2. Export data

# load Iris data set
data(iris)

# print the head of the data frame
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```r
# Export data using the write.csv function
write.csv(iris, file = "iris.csv")

# Export data using the write.table function
write.table(iris, file = "iris.txt")

# End of the section
```

3. Import data from Excel

```
# 3. Import data from Excel

# Import data from Excel
df <- read.xlsx("../../data/chd_genes.annotations.xlsx", sheet = 1)

# print the head of the data frame
head(df)
```

```
##   gene_symbol     category        pLI  plof gene_length obs_lof obs_syn exp_lof
## 1       ABCC9    syndromic 9.3524e-09 0.482      144002      30     298  84.399
## 2        ABL1    syndromic 9.9998e-01 0.176      173730       3     325  44.108
## 3       ACAD9    syndromic 4.5256e-08 0.814       36472      17     147  31.321
## 4       ACTA2 nonsyndromic 9.3017e-01 0.364       56317       2      72  17.293
## 5        ACTB    syndromic 9.8564e-01 0.232       36634       0     190  12.858
## 6       ACTC1 nonsyndromic 7.3668e-01 0.480        8044       2      89  13.125
##   exp_syn chromosome
## 1 298.160         12
## 2 314.370          9
## 3 144.410          3
## 4  84.674         10
## 5  96.859          7
## 6  89.788         15
```

```
# End of the section
```

4. Export data to Excel

```
# 4. Export Iris data to Excel

# load Iris data set
data(iris)

# print the head of the data frame
head(iris)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa
```

```
# Export data to Excel
write.xlsx(iris, file = "iris.xlsx")

# End of the section
```

5. Evaluating speed performance of import functions.

Note: Differences between <read.csv>, <read.table> and <read.delim> read.csv is a special case of read.table, which is a special case of read.delim. read.csv uses a comma as a separator, read.table uses a tab, and read.delim uses a tab. read.csv is the most common of the three, and read.table is the most flexible. read.delim is used when the data is tab-delimited, but the extension is not txt.

```r
# What is the faster function to import data?
# read.csv is faster than read.table and read.delim.

# Import data using the read.csv function
microbenchmark(
  read.csv("../../data/chd_genes.annotations.tsv",
                        sep = "\t",
                        header = TRUE)
  )
```

```
## Unit: microseconds
##                                                                                expr
##  read.csv("../../data/chd_genes.annotations.tsv", sep = "\\t",      header = TRUE)
##      min       lq     mean   median       uq      max neval
##  986.015 1000.588 1172.397 1139.423 1322.633 1713.044   100
```

```r
# Import data using the read.table function
microbenchmark(
  read.table("../../data/chd_genes.annotations.tsv",
                          sep = "\t",
                          header = TRUE)
  )
```

```
## Unit: milliseconds
##                                                                                  expr
##  read.table("../../data/chd_genes.annotations.tsv", sep = "\\t",      header = TRUE)
##       min       lq     mean   median       uq      max neval
##  1.007881 1.122513 1.186612 1.191546 1.263893 1.745602   100
```

```r
# Import data using the read.delim function
microbenchmark(
  read.delim("../../data/chd_genes.annotations.tsv",
                          sep = "\t",
                          header = TRUE)
  )
```

```
## Unit: microseconds
##                                                                                  expr
##  read.delim("../../data/chd_genes.annotations.tsv", sep = "\\t",      header = TRUE)
##      min      lq     mean   median      uq      max neval
##  898.965 908.923 1093.086 1123.926 1243.84 1717.783   100
```

```r
# End of the section
```