

Practical Exam

Arjay De Guzman

2024-03-07

Load the built-in warpbreaks dataset.

```
data ("warpbreaks")
```

Find out, in a single command, which columns of warpbreaks are either numeric or integer. What are the data types of each column?

```
str(warpbreaks)
```

```
## 'data.frame':   54 obs. of  3 variables:
## $ breaks : num  26 30 54 25 70 52 51 26 67 18 ...
## $ wool   : Factor w/ 2 levels "A","B": 1 1 1 1 1 1 1 1 1 1 ...
## $ tension: Factor w/ 3 levels "L","M","H": 1 1 1 1 1 1 1 1 2 ...
```

How many observations does it have?

```
nrow(warpbreaks)
```

```
## [1] 54
```

Is numeric a natural data type for the columns which are stored as such? Convert to integer when necessary.
#yes it is a natural data type for columns which are stored in warpbreaks

Error messages in R sometimes report the underlying type of an object rather than the user-level class. Derive from the following code and error message what the underlying type. Explain what is the error all about. Do not just copy the error message that was displayed.

#A few operations require matching data types to function. You are attempting to convert a list, which can contain a variety of data kinds, straight into an integer, which is a single numeric number, this is where you are going to encounter the problem “Error: ‘list’ object cannot be coerced to type ‘integer’”.

```
#warpbreaks1 <- as.integer(warpbreaks)
```

Read the complete file using readLines.

```
exampleFile <- readLines ("exampleFile.txt")
```

```
## Warning in readLines("exampleFile.txt"): incomplete final line found on
## 'exampleFile.txt'
```

```
exampleFile
```

```
## [1] "// Survey data. Created : 21 May 2013"
## [2] "// Field 1: Gender"
## [3] "// Field 2: Age (in years)"
## [4] "// Field 3: Weight (in kg)"
## [5] "M;28;81.3"
## [6] "male;45;"
## [7] "Female;17;57,2"
```

```
## [8] "fem.;64;62.8"
```

Separate the vector of lines into a vector containing comments and a vector containing the data. Hint: use `grepl`.

```
commentLine<- exampleFile[grepl("^//", exampleFile)]
dataLine<- exampleFile[!grepl("^//", exampleFile)]
commentLine
```

```
## [1] "// Survey data. Created : 21 May 2013"
## [2] "// Field 1: Gender"
## [3] "// Field 2: Age (in years)"
## [4] "// Field 3: Weight (in kg)"
```

```
dataLine
```

```
## [1] "M;28;81.3"      "male;45;"      "Female;17;57,2" "fem.;64;62.8"
```

Extract the date from the first comment line and display on the screen “It was created data.”

```
date <- gsub(".*Created data: (.*)<.*", "\\1", commentLine[1])
cat("It was created data", date, "\n")
```

```
## It was created data // Survey data. Created : 21 May 2013
```

A. Split the character vectors in the vector containing data lines by semicolon (;) using `strsplit`.

```
#splitData<- lapply(data, function(x) unlist(strsplit(x, ";")))
#splitData
```

```
#[[1]] #[1] "M" "28" "81.3"
#[[2]] #[1] "male" "45"
#[[3]] #[1] "Female" "17" "57,2"
#[[4]] #[1] "fem." "64" "62.8"
```

B. Find the maximum number of fields retrieved by split. Append rows that are shorter with NA's.

```
#maxFields <- max(sapply(splitData, length))
#splitData<- lapply(splitData, function(x) c(x, rep(NA, maxFields - length(x))))
#maxFields
#splitData
```

```
#[1] 3 #[[1]] #[1] "M" "28" "81.3"
#[[2]] #[1] "male" "45" NA
#[[3]] #[1] "Female" "17" "57,2"
#[[4]] #[1] "fem." "64" "62.8"
```

Use `unlist` and `matrix` to transform the data to row-column format.

```
#matrixData <- matrix(unlist(splitData), ncol = maxFields, byrow = TRUE)
#matrixData
```

```
[,1] [,2] [,3]
```

```
#[1,] "M" "28" "81.3" #[2,] "male" "45" NA
#[3,] "Female" "17" "57,2" #[4,] "fem." "64" "62.8"
```

From comment lines 2-4, extract the names of the fields. Set these as colnames for the matrix you just created.

```
#field_names <- strsplit(comments[2:4], ": ")  
#colnames(matrixData) <- field_names  
#print(matrixData)
```

**c("// Field 1", "Gender") c("// Field 2", "Age (in years)") c("//
Field 3", "Weight (in kg)")**

```
#[1,] "M" "28" "81.3"  
#[2,] "male" "45" NA  
#[3,] "Female" "17" "57,2"  
#[4,] "fem." "64" "62.8"
```