

13 Noch mehr Details zur Sprache

Aufgabe 1:

In R gibt es die Funktion `rev()`, die einen Vektor umdreht. Schreiben Sie analog eine Funktion `strrev()`, die einen gegebenen Character String umdreht. Z.B. `strrev("ABC")` gibt `"CBA"`.

Aufgabe 2:

Laden Sie den Datensatz `crude` aus dem Paket `tm`. Im Normalfall werden Sie dieses Paket erst installieren müssen. Dies können Sie z.B. mithilfe von

```
> install.packages("tm")
```

machen.

Sie können ihn folgendermaßen in einen Character-Vektor umwandeln:

```
> library("tm")
> data("crude", package = "tm")
> crude <- sapply(crude, as.character)
```

- Ersetzen Sie die Zeilenumbrüche aus dem Datensatz durch Leerzeichen.
- Ersetzen Sie mehrere aufeinanderfolgende Leerzeichen durch eines.
- Splitten Sie die Zeichenketten jeweils an Punkten, Rufzeichen und Fragezeichen und bestimmen Sie die Anzahl der Teile, die Sie bekommen.
- Splitten Sie die Zeichenketten jeweils an Punkten, Rufzeichen und Fragezeichen, wo danach ein Leerzeichen und ein Großbuchstabe kommen, und bestimmen Sie die Anzahl der Teile, die Sie bekommen.

Hinweis: Alle Großbuchstaben können Sie z.B. entfernen mithilfe von

```
> x <- c("aBcD", "ABcd")
> gsub("[A-Z]", "", x)
```

```
[1] "ac" "cd"
```

Aufgabe 3:

Laden Sie den Datensatz `cheese` aus dem Paket `bayesm`. Im Normalfall werden Sie dieses Paket erst installieren müssen. Dies können Sie z.B. mithilfe von

```
> install.packages("bayesm")
```

machen. Dann können Sie das Paket und den Datensatz laden

```
> library("bayesm")  
> data("cheese", package = "bayesm")
```

Die Variable **RETAILER** enthält sowohl die Information über den Standort als auch die Supermarktkette.

- Splitten Sie die Information in 2 Variablen **Location** und **Chain** auf.
- Entfernen Sie bei der Variable **Location** alles, (1) was in runden Klammern steht und (2) was nach Beistrichen und Schrägstrichen kommt inklusive dieser Sonderzeichen.
- Bestimmen Sie das Subset der Beobachtungen, wo der Name der **Chain** "FOOD" enthält. Für welche **Location** sind die durchschnittlichen Preise am höchsten und für welche am niedrigsten, wenn man nur dieses Subset betrachtet?

Aufgabe 4:

Laden Sie den Datensatz **EUR04PlayerSkillsSep11** aus dem Paket **SportsAnalytics**. Im Normalfall werden Sie dieses Paket erst installieren müssen. Dies können Sie z.B. mithilfe von

```
> install.packages("SportsAnalytics")
```

machen. Dann können Sie das Paket und den Datensatz laden

```
> library("SportsAnalytics")  
> data("EUR04PlayerSkillsSep11", package = "SportsAnalytics")
```

Die Variable **Positions** enthält detaillierte Information über die Positionen, an denen der Spieler eingesetzt wird. Diese Positionen sind als Text aufgezählt und werden durch Beistriche getrennt. Die Position, der das ! folgt, ist die primäre Position des Spielers.

- Wieviel verschiedene Positionen gibt es?
- Wandeln Sie die Information folgendermaßen um:
 - Erzeugen Sie für jede Position eine eigene Variable. Diese soll den Namen der Position haben.
 - Die Variable soll ein Faktor sein, der 3 Levels hat, die angeben, ob der Spieler nie an der Position, sekundär oder primär an der Position eingesetzt wird. Die primäre Position wird durch das Rufzeichen angezeigt.
 - Hängen Sie die Variablen an den Datensatz als zusätzliche Spalten des Dataframes an.
- Überprüfen Sie, ob für jeden Spieler nur eine primäre Position angegeben ist.

Aufgabe 5:

Im Folgenden wird der Datensatz **EUR04PlayerSkillsSep11** aus dem Paket **SportsAnalytics** verwendet. Die Variable **Birthday** enthält den Geburtstag der Spieler.

- Berechnen Sie daraus das Alter (in Jahren) zum 17. September 2011.
- Vergleichen Sie das Ergebnis mit der Variable **Age**, die ebenfalls im Datensatz enthalten ist.