Вероятности и статистика с R

Асен Чорбаджиев

October 12, 2017

1 Структури от данни

1.1 Таблици и категории

За категоризиране на данните в R се използва факторизация. Основната причина налични данни да бъдат категоризирани е използването им в различни статистическите модели, където категоризираните променливи влизат различно от количествените променливи. За тази цел запазването на даните като фактор осигурява, че съответния модел ще отчете разликите в данните коректно.

Факторизацията става с функцията factor(). Тя запазва факторите като вектор от целочислени стойности със съответното множество от текстови стойности готови за ползване. Във фактори могат да бъдат ползвани и числени и текстови променливи, но фактор нивата (levels) винаги са стрингове. Тяхното извеждане става с функцията levels(). Подредбата и множеството от нива може да се променя, като под подразбиране те са подредени възходящо. Изключването на нива става с параметъра exclude, а опцията за сортиране е логическото ordered.

Пример:

- data = c(1,2,2,3,1,2,3,3,1,2,3,3,1)
- fdata = factor(data)
- Превръщане в римски числа:
  1. rdata = factor(data,labels=c("I","II","III"))
  2. levels(fdata) = c('I','II','III')

Факторите представляват много ефикасен начин на представяне на текстови стойности - всеки char символ е запазен само веднъж и данните са представени

като вектор от integers. Най-простото представяне на това става с функцията table(). Функцията връща обект от клас "table", вектор от целочислени стойности на обобщените данни. Резултатът е винаги един или повече вектори в зависимост от броя на факторите. Например:

```
> mons = c("March","April","January","November","January",
+ "September","October","September","November","August",
+ "January","November","November","February","May","August",
+ "July","December","August","August","September","November",
+ "February","April")
> mons = factor(mons)
> table(mons)
```

Сортирането става:

```
> mons = factor(mons,levels=c("January","February","March",
+                "April","May","June","July","August","September",
+                "October","November","December"),ordered=TRUE)
> mons[1] < mons[2]
> table(mons)
```

Когато се налага вече факторизирани данни да бъдат превърнати в целочислени данни, директното обръщане с функции като as.numeric или as.integer връща грешка. Това обръщане става с атрибута levels:

```
> fert = c(10,20,20,50,10,20,10,50,20)
> fert = factor(fert,levels=c(10,20,50),ordered=TRUE)
> mean(as.numeric(levels(fert)[fert]))
```

За нормиране на данни в таблица се използва prop.table():

```
>  t=table(mons)
> prop.table(t)
```

1.2 Lists

В R List е структура, която съдържа в себеси една или повече структури от всякакъв друг вид и тип, включително вграден лист list. Създаването на такава структура става с функцията list():

```
mylist <- list (a = 1:5, b = "Hi There", c = function(x) x * sin(x))
```

Всеки вектор от подмножеството на един list, може да бъде от различен тип и с различен резмер. За достъп до елементите:

- връща подструктура с индекс i като лист: mylist[i]; i>=1.
  Затова скаларната операция mylist[1] + 1 връща грешка.
- достъп до подструктура като вектор с индекс: x[[i]]; i>=1
- достъп до подструктура като вектор с псевдоним: x[["name"]]

- достъп до подструктура в оргинален вид (втори начин): x$item
- достъп до i-ти елемент от подструктура с item: x$item[i]; i>=1
- размерност length():
  length(mylist) - брой подструктури в mylist
  length(mylist$a) - брой елеменити на a
- добавяне на нов елемент:
  mylist$d <- "New item"
  mylist[[*index*]] <- "New item".
  Какво ще стане следната операция: *mylist[[6]] = c(2,3,4,4,3)*
- премахване на елемент: mylist$b <- NULL
- какво връща mylist[[2]][1] ?
- unlist()

1.3 Data Frames

List с еднакъв размер на всичките си вектори се нарича Data Frame. Такъв тип таблици се се създават с функцията data.frame():
> n = c(2, 3, 5)
> s = c("aa", "bb", "cc")
> b = c(TRUE, FALSE, TRUE)
> df = data.frame(n, s, b) # df is a data frame
За достъп до елементите се ползват операторите на List. За таблично добавяна на редове и стълбове се ползват rbind() и cbind(). Допълнително, командите head(), tail(), връщат първите и съответно последните n-брой елементи от всяка колона, като по подразбиране броят им n=6.

2 Входни и изходни данни

2.1 Входни данни

Данните могат да съществуват под различни форми:

- библиотеки в R. Зареждане на dataset(s) с функцията data().
- файлове на локална или отдалечена локация с разширения .dat, .txt, .csv, .xls, .xlsx, ...
- бази данни, ODBC (RODBC), ...

- Hadoop, hdf5, ...
- Кеширани данни от средата R - .RData. Зареждане с функцията load().

Тяхното зареждане в R става без ограничение, с единственото условие, че много от функционалностите са изнесени в специални библиотеки. Основните операции за четене от файлове е базирана на съответните C функции. Подобна функция е scan(). Когато трябва да бъде прочетен файл с подредени в таблица данни се използва read.table(). Подобни функции за четене са read.csv() и read.csv2() за четене на csv файлове.

2.2 Съхранение на данни

Съхранине на файлове:

- RData - save()
- Стандартната C функциионалност: file_handler = file("file_name", "wa")
  добавяне на ред: cat("list_of_elements",...,"file_handler")
  Затваряне на връзката: close("file_handler")
  конкатенация на файлове: file.append()
- write.table(); write.csv();write.csv2()
- бази данни, ODBC, ...

2.3 Descriptive statistic и графично представяне

Най-простата дескриптивна статистика е средната стойност. В R тя се смята с функцият *mean()*. Но използването на средните стойности при силно skewed разпределения не води до добри резултати. Тогава за целта се използва медиана (*median*). Медианата е средната стойност в наредените по големина стойности от извадката $x_1 <= x_2 < ... <= x_n$. Когато броят на елементи n е четен, за медиана се взима средната стойност от двете средни стойности в извадката. Функцията в R се нарича *median()*.

За визулното представяне на резултатите (descriptive statistics) в R съществува богата функционалност. Основните видове графики в основния пакет:

- plot() - Графика за съпоставяне последователно период на измерване и измерена стойност.

- barplot() - Графика за представяне на категоризирани данни с правоъгълни bars с височина и широчина пропорционална на стойностите, които те представляват. Бар-графиките може да бъдат вертикални или хоризонтални.
- pie() - Сумаризирана графика за представяне на категоризирани данни представящи различни стойности на дадена променлива в процентно/дялово разпределение. Обобщените данни трябва да бъдат предварително подготвени таблично.
- histogram() - графически метод за представяне на брой измервания попадащи в определени интервали от стойности. Тези интервали се наричат класове или bins. Честота с която данните попадат във всеки клас се описва с бар графика.
- boxplot() - медиана, квантили, outliers. Обяснение в следващите теми. Пример:
  > x = c(5, 5, 5, 13, 7, 11, 11, 9, 8, 9)
  > y = c(11, 8, 4, 5, 9, 5, 10, 5, 4, 19)
  > boxplot(x,y)
- Библиотеки за други визуални изображения, като картографски карти.

3 Упражнения:

- **Зад. 1.** : Сравнете резултатите:
  1.1 Защо командите връщат различен резултат
  x = list(1, 2, 3, 4)
  x2 = list(1:4)
  1.2 Защо командите връщат еднакъв резултат:
  x = list(1, 2, 3, 4)
  x[1]
  x[[1]]

- **Зад. 2.** Използвайте вградената data.frame структура mtcars за:
  1. Извадете списък на всички модели автомобили с 8 цилиндри.
  2. Средното и медианата на mpg за всички автомобили с 5 предавки.

- **Зад. 3.** Разглеждаме пакета "MASS". Да се обработи от таблицата survey:
  1.Начертайте бар-графика за пол и тютюнопушенето на анкетираните.
  2.Начертайте pie-chart за честотата на пушене.
  3. Начертайте хистограма за пулса на анкетираните. Начертайте и плътност, като използвате density().
  4. Направете таблица за разпределението на пушачите в различните възрасти. Представете графично.