zjin26_mini-assign05

Vincent Jin

2023-04-10

==========

Mini-assignment # 5

==========

using the following data.table, calculate the following (use built-in data.table functions):

```
library(data.table)
set.seed(1234)
id = 1:5000
age = abs(rnorm(5000,40,20))
race = sample(c('W','B','A',NA),5000,replace=T)
dxct = floor(sample(1:3,5000,replace=T)*(age/10))
cost = rnbinom(5000, size = 1, mu = 10000)*age
dt = data.table(id = id, age = age, race = race, dxct = dxct, cost = cost)
```

(1) count patients with missing race info

```
cat("The number of patients with missing race info is:\n")
```

```
## The number of patients with missing race info is:
```

```
dt[is.na(race), .N]
```

```
## [1] 1252
```

(2) count patients designated as Asian, more than 65 with a cost over 10k

```
cat("The number of patients designated as Asian, more than 65 with a cost over 10k is:\n")
```

```
## The number of patients designated as Asian, more than 65 with a cost over 10k is:
```

```
dt[race == "A" & age > 65 & cost > 10000, .N]
```

```
## [1] 125
```

(3) count patients over 65 with either a cost over 500k "or" more than 5 diagnoses

```
cat("The number of patients over 65 with either a cost over 500k or more than 5 diagnoses is:\n")
```

```
## The number of patients over 65 with either a cost over 500k or more than 5 diagnoses is:
```

```
dt[age > 65 & (cost > 500000 | dxct > 5), .N]
```

```
## [1] 514
```

(4) count patients in the top 1% of cost (use the quantile command)

```
cat("The number of patients in the top 1% of cost is:\n")
```

```
## The number of patients in the top 1% of cost is:
```

```
dt[cost >= quantile(dt$cost, probs = (1 - 0.01)), .N]
```

```
## [1] 50
```

(5) calculate the following per race: count of patients, average and SD of age, average and SD of number of diseases, and average and SD of cost [using built-in data.table functions]

```
cat("Count of patients by race:\n")
```

```
## Count of patients by race:
```

```r
dt[, .N, by = race]
```

```
##    race    N
## 1: <NA> 1252
## 2:    B 1229
## 3:    A 1260
## 4:    W 1259
```

```r
cat("average of age by race:\n")
```

```
## average of age by race:
```

```r
dt[, .(mean(age)), by = race]
```

```
##    race       V1
## 1: <NA> 39.84704
## 2:    B 40.86299
## 3:    A 39.88310
## 4:    W 40.17151
```

```r
cat("SD of age by race:\n")
```

```
## SD of age by race:
```

```r
dt[, .(sd(age)), by = race]
```

```
##    race       V1
## 1: <NA> 18.71699
## 2:    B 19.31096
## 3:    A 19.17043
## 4:    W 19.52287
```

```r
cat("average of cost by race:\n")
```

```
## average of cost by race:
```

```r
dt[, .(mean(cost)), by = race]
```

```
##    race       V1
## 1: <NA> 411056.8
## 2:    B 390848.5
## 3:    A 400782.6
## 4:    W 418355.4
```

```r
cat("SD of cost by race:\n")
```

```
## SD of cost by race:
```

```r
dt[, .(sd(cost)), by = race]
```

```
##    race       V1
## 1: <NA> 513273.7
## 2:    B 460389.3
## 3:    A 445071.6
## 4:    W 529598.2
```

```r
cat("aggregated:\n")
```

```
## aggregated:
```

```r
dt[, .(ct = .N, avg_age = mean(age), sd_age = sd(age), avg_cost = mean(cost), sd_cost = sd(cost)), by =
```

```
##    race   ct  avg_age   sd_age avg_cost  sd_cost
## 1: <NA> 1252 39.84704 18.71699 411056.8 513273.7
## 2:    B 1229 40.86299 19.31096 390848.5 460389.3
## 3:    A 1260 39.88310 19.17043 400782.6 445071.6
## 4:    W 1259 40.17151 19.52287 418355.4 529598.2
```

(6) join the data.table with a new data.table that provides some lab information (explain the "NA" in the new merged data.table)

```r
id = 1000:5000
lab = round(runif(4001, 50, 100))
dt_lab = data.table(id = id, lab = lab)
```

```r
merge(dt, dt_lab, by = c('id'), all = T)
```

```
##         id       age race dxct       cost lab
##    1:    1 15.858685 <NA>    4  111296.25  NA
##    2:    2 45.548585 <NA>    4  205378.57  NA
##    3:    3 61.688824    B   18  251752.09  NA
##    4:    4  6.913954    A    2   82566.44  NA
##    5:    5 48.582494 <NA>    4  818566.44  NA
##   ---
## 4996: 4996 16.071378    B    4   50030.20  99
## 4997: 4997 42.829031    B    4 2003756.23  71
## 4998: 4998 32.123351    W    6  779312.50  52
## 4999: 4999 38.760834    A    3  463463.29  87
## 5000: 5000 34.243983    A    6  584784.49  82
```

Explain the "NA" in the new merged data.table

Since the dt_lab dataset only contains information for patients whose id started from 1000 to 4000 but dt had patients whose id started from 0 - 999, these patients did not have any information in the dt_lab dataset thus had NA once the two sets merged together.