

Tugas Modul 4

Regita Amelia Asnawi Putri

2022-09-15

Import dataset “murders”;

```
library(dslabs)
data(murders)
```

R Markdown

Soal Nomor 1:

Gunakan operator aksesoris (\$) untuk mengakses variabel populasi dan menyimpannya pada objek baru “pop”. Kemudian gunakan fungsi sort untuk mengurutkan variabel “pop”. Pada langkah terakhir, gunakan operator[] untuk menampilkan nilai populasi terkecil. Jawab :

```
pop <- (murders$population)
ps <- sort(pop) ##ps adalah popSort
ps[1]

## [1] 563626
```

Soal Nomor 2:

Tampilkan indeks dari data yang memiliki nilai populasi terkecil. Petunjuk : gunakan fungsi order. Jawab:

```
indeks <- order(pop)
indeks[1]

## [1] 51
```

Soal Nomor 3:

Dengan fungsi which.min. Tulis satu baris kode yang dapat menampilkan hasil yang sama dengan langkah diatas. Jawab :

```
minMurders <- which.min(murders$population)
minMurders

## [1] 51
```

Soal Nomor 4:

Tampilkan nama negara yang memiliki populasi terkecil. Jawab :

```
minMurders <- which.min(murders$population)
murders$state[minMurders]

## [1] "Wyoming"
```

Soal Nomor 5:

Untuk membuat data frame baru, contoh script yang dapat digunakan adalah sebagai berikut:

```
temp <- c(35, 88, 42, 84, 81, 30) city <- c("Beijing", "Lagos", "Paris", "Rio de Janeiro", "San
Juan", "Torontoo") city_temps <- data.frame(name = city, temperature = temp)
```

Gunakan fungsi rank untuk menentukan peringkat populasi dari tiap negara bagian, dimulai dari nilai terkecil hingga terbesar. Simpan hasil pemeringkatan di objek baru "ranks", lalu buat data frame baru yang berisi nama negara bagian dan peringkatnya dengan nama "my_df". Jawab :

```
ranks <- rank(murders$population)
ranks

## [1] 29 5 36 20 51 30 23 7 2 49 44 12 13 47 37 22 19 26 27 11 33 38 43
31 21
## [26] 34 8 14 17 10 41 16 48 42 4 45 24 25 46 9 28 6 35 50 18 3 40 39
15 32
## [51] 1

my_df <- data.frame(Name = murders$state, rank = ranks)
my_df

##           Name rank
## 1      Alabama  29
## 2       Alaska   5
## 3      Arizona  36
## 4    Arkansas  20
## 5   California  51
## 6     Colorado  30
## 7   Connecticut  23
## 8      Delaware   7
## 9 District of Columbia  2
## 10      Florida  49
## 11      Georgia  44
## 12      Hawaii  12
## 13      Idaho   13
## 14     Illinois  47
## 15     Indiana  37
## 16       Iowa   22
## 17      Kansas  19
## 18     Kentucky  26
## 19    Louisiana  27
## 20       Maine  11
```

```
## 21 Maryland 33
## 22 Massachusetts 38
## 23 Michigan 43
## 24 Minnesota 31
## 25 Mississippi 21
## 26 Missouri 34
## 27 Montana 8
## 28 Nebraska 14
## 29 Nevada 17
## 30 New Hampshire 10
## 31 New Jersey 41
## 32 New Mexico 16
## 33 New York 48
## 34 North Carolina 42
## 35 North Dakota 4
## 36 Ohio 45
## 37 Oklahoma 24
## 38 Oregon 25
## 39 Pennsylvania 46
## 40 Rhode Island 9
## 41 South Carolina 28
## 42 South Dakota 6
## 43 Tennessee 35
## 44 Texas 50
## 45 Utah 18
## 46 Vermont 3
## 47 Virginia 40
## 48 Washington 39
## 49 West Virginia 15
## 50 Wisconsin 32
## 51 Wyoming 1
```

Soal Nomor 6:

Ulangi langkah sebelumnya, namun kali ini urutkan my_df dengan fungsi order agar data yang ditampilkan merupakan data yang telah diurutkan dari populasi yang paling tidak padat hingga ke yang terdapat. Petunjuk: buat objek "ind" yang akan menyimpan indeks yang diperlukan dalam mengurutkan data populasi. Jawab :

```
ranks <- rank(murders$population)
my_df <- data.frame(Name = murders$state, rank = ranks)
ind = order(my_df$rank)
my_df$Name[ind]
```

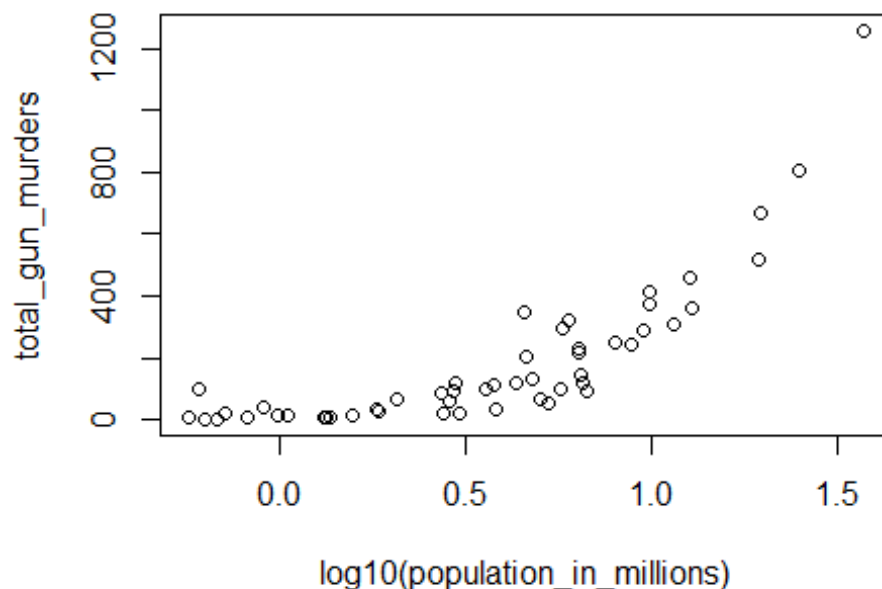
```
## [1] "Wyoming" "District of Columbia" "Vermont"
## [4] "North Dakota" "Alaska" "South Dakota"
## [7] "Delaware" "Montana" "Rhode Island"
## [10] "New Hampshire" "Maine" "Hawaii"
## [13] "Idaho" "Nebraska" "West Virginia"
## [16] "New Mexico" "Nevada" "Utah"
```

```
## [19] "Kansas"           "Arkansas"          "Mississippi"
## [22] "Iowa"             "Connecticut"       "Oklahoma"
## [25] "Oregon"           "Kentucky"          "Louisiana"
## [28] "South Carolina"   "Alabama"           "Colorado"
## [31] "Minnesota"        "Wisconsin"          "Maryland"
## [34] "Missouri"         "Tennessee"         "Arizona"
## [37] "Indiana"          "Massachusetts"     "Washington"
## [40] "Virginia"         "New Jersey"        "North Carolina"
## [43] "Michigan"         "Georgia"            "Ohio"
## [46] "Pennsylvania"     "Illinois"           "New York"
## [49] "Florida"          "Texas"              "California"
```

Soal Nomor 7:

Untuk keperluan analisis data, akan dibuat plot yang memvisualisasikan total pembunuhan terhadap populasi dan mengidentifikasi hubungan antara keduanya. Script yang digunakan : `population_in_millions <- murders$population/106` `total_gun_murders <- murders$total` `plot(population_in_millions, total_gun_murders)` Perlu diingat bahwa beberapa negara bagian memiliki populasi di bawah 5 juta, sehingga untuk mempermudah analisis, buat plot dalam skala log. Transformasi nilai variabel menggunakan transformasi log10, kemudian tampilkan plot-nya. Jawab :

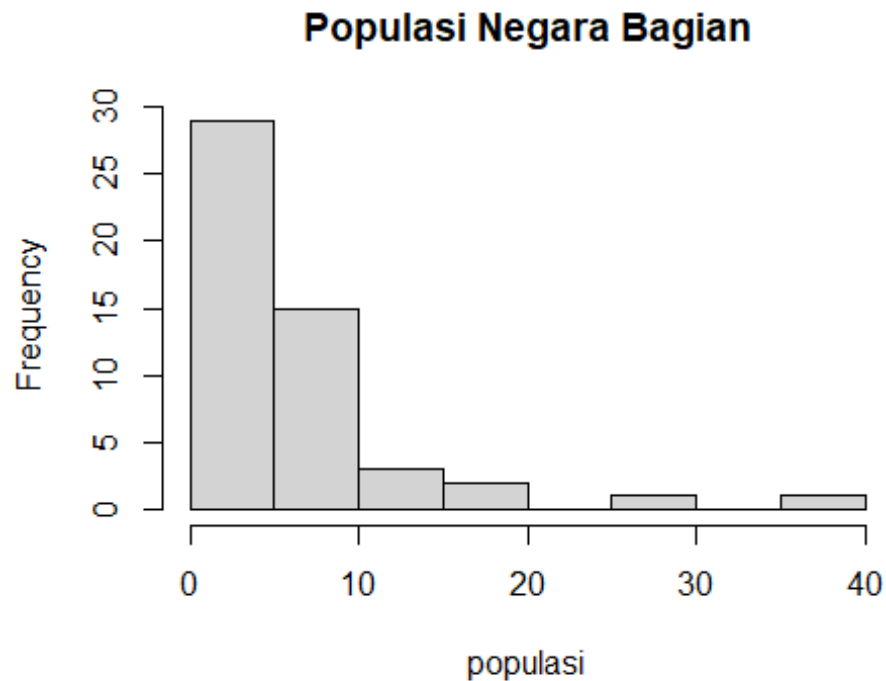
```
population_in_millions <- murders$population / 10^6
total_gun_murders <- murders$total
plot(log10(population_in_millions), total_gun_murders)
```



Soal Nomor 8

Buat histogram dari populasi negara bagian. Jawab :

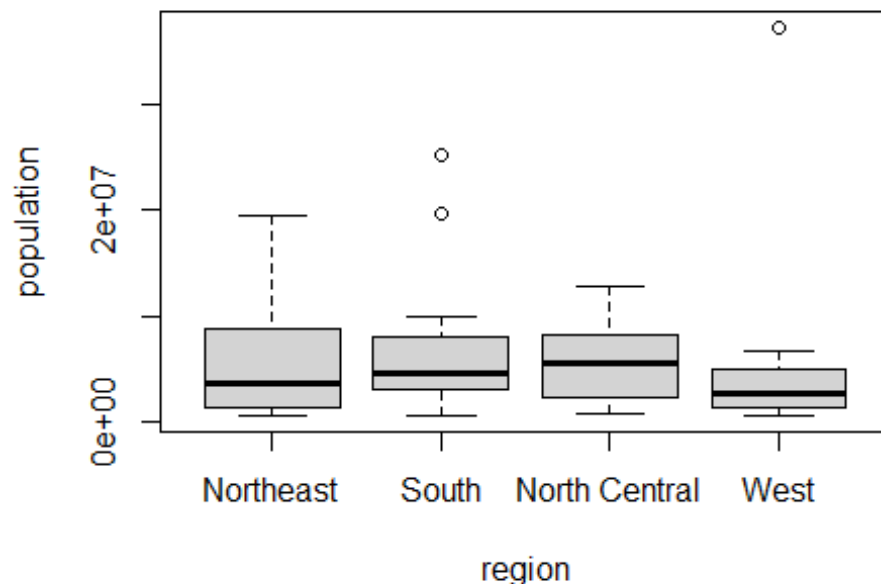
```
populasi <- with(murders, murders$population / 10^6)  
hist(populasi, main = "Populasi Negara Bagian")
```



Soal Nomor 9

Hasilkan boxplot dari populasi negara berdasarkan wilayahnya. Jawab :

```
boxplot(population~region, data = murders)
```



This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

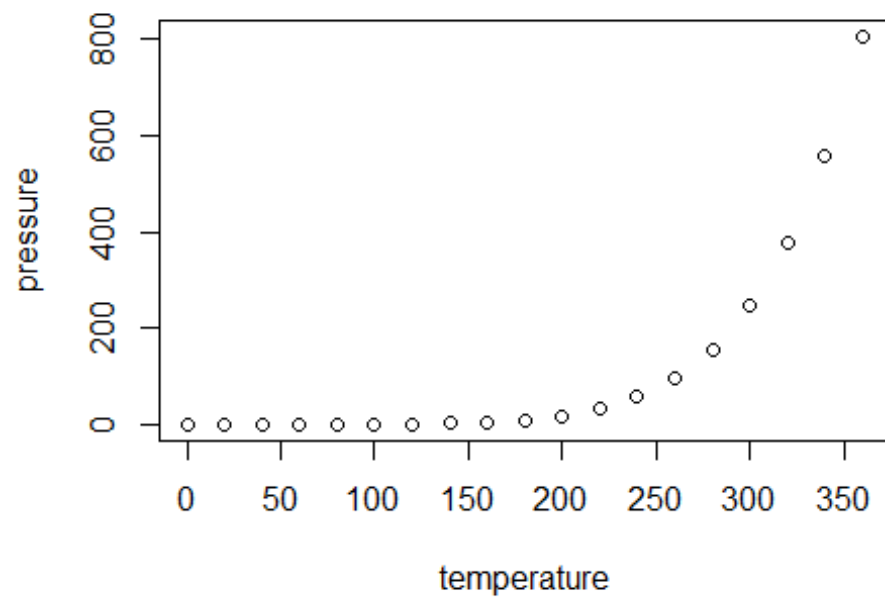
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0    Min.   :  2.00
##  1st Qu.:12.0    1st Qu.: 26.00
##  Median :15.0    Median : 36.00
##  Mean   :15.4    Mean   : 42.98
##  3rd Qu.:19.0    3rd Qu.: 56.00
##  Max.   :25.0    Max.   :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.