Homework 6

Christopher Brunswick

11/6/2021

* 1. Compute the per 100,000 murder rate for each state and store it in an object called murder\_rate. Then use logical operators to create a logical vector named low that tells us which entries of murder\_rate are lower than 1.

library(dslabs)

## Warning: package 'dslabs' was built under R version 4.0.5

data(murders)  
  
murder\_rate <- murders$total/murders$population\*100000  
murder\_rate < 1

low <- murder\_rate < 1

## [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE  
## [13] TRUE FALSE FALSE TRUE FALSE FALSE FALSE TRUE FALSE FALSE FALSE TRUE  
## [25] FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE TRUE FALSE  
## [37] FALSE TRUE FALSE FALSE FALSE TRUE FALSE FALSE TRUE TRUE FALSE FALSE  
## [49] FALSE FALSE TRUE

1. Now use the results from the previous exercise and the function which to determine the indices of murder\_rate associated with values lower than 1.

which(low)

## [1] 12 13 16 20 24 30 35 38 42 45 46 51

1. Use the results from the previous exercise to report the names of the states with murder rates lower than 1.

murders$state[which(low)]

lowmurders <- murders$state[which(low)]

## [1] "Hawaii" "Idaho" "Iowa" "Maine"   
## [5] "Minnesota" "New Hampshire" "North Dakota" "Oregon"   
## [9] "South Dakota" "Utah" "Vermont" "Wyoming"

1. Now extend the code from exercises 2 and 3 to report the states in the Northeast with murder rates lower than 1. Hint: use the previously defined logical vector low and the logical operator &.

murders$region == "Northeast"

## [1] FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE  
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE TRUE FALSE FALSE  
## [25] FALSE FALSE FALSE FALSE FALSE TRUE TRUE FALSE TRUE FALSE FALSE FALSE  
## [37] FALSE FALSE TRUE TRUE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE  
## [49] FALSE FALSE FALSE

northeast <- murders$region == "Northeast"  
states\_n\_lowregion <- low & northeast  
murders$state[states\_n\_lowregion]

## [1] "Maine" "New Hampshire" "Vermont"

1. In a previous exercise we computed the murder rate for each state and the average of these numbers. How many states are below the average?

mean(murder\_rate)

## [1] 2.779125

average\_mr <- mean(murder\_rate)  
newlow <- murder\_rate < average\_mr  
murders$state[newlow]

## [1] "Alaska" "Colorado" "Connecticut" "Hawaii"   
## [5] "Idaho" "Indiana" "Iowa" "Kansas"   
## [9] "Kentucky" "Maine" "Massachusetts" "Minnesota"   
## [13] "Montana" "Nebraska" "New Hampshire" "New York"   
## [17] "North Dakota" "Ohio" "Oregon" "Rhode Island"   
## [21] "South Dakota" "Utah" "Vermont" "Washington"   
## [25] "West Virginia" "Wisconsin" "Wyoming"

length(murders$state[newlow])

## [1] 27

1. Use the match function to identify the states with abbreviations AK, MI, and IA. Hint: start by defining an index of the entries of murders$abb that match the three abbreviations, then use the [ operator to extract the states.

abbrev <- match(c("AK", "MI", "IA"), murders$abb)  
murders$state[abbrev]

## [1] "Alaska" "Michigan" "Iowa"

1. Use the %in% operator to create a logical vector that answers the question: which of the following are actual abbreviations: MA, ME, MI, MO, MU?

c("MA", "ME", "MI", "MO", "MU") %in% murders$abb

## [1] TRUE TRUE TRUE TRUE FALSE

1. Extend the code you used in exercise 7 to report the one entry that is not an actual abbreviation. Hint: use the ! operator, which turns FALSE into TRUE and vice versa, then which to obtain an index.

!(c("MA", "ME", "MI", "MO", "MU") %in% murders$abb)

## [1] FALSE FALSE FALSE FALSE TRUE

which(!(c("MA", "ME", "MI", "MO", "MU") %in% murders$abb))

## [1] 5