Homework 8

Conor Craig

2023-10-19

Creating a data frame object ——————————————————————

Create a vector called ID that contains a number from 1 through 6

```r
ID <- sample (1:6)
```

Create a vector called Race that contains the following:

"black", "white", "black", "hispanic", "white", "white"

```r
race <- c("Black", "White", "Black", "hispanic", "White", "White")
```

What is the type of the Race vector?

Is it a factor? Is it a character? Is it a logical vector?

Find out by using some commands.

```r
# Three commands are used since it asked for "some" commands
class(race)
```

```
## [1] "character"
```

```r
typeof(race)
```

```
## [1] "character"
```

```
str(race)
```

```
##  chr [1:6] "Black" "White" "Black" "hispanic" "White" "White"
```

Create a vector called Voted_For_Obama that contains thefollowing:

TRUE, TRUE, TRUE, FALSE, FALSE, TRUE

```
voted_for_obama <- c(TRUE, TRUE, TRUE, FALSE, FALSE, TRUE)
```

Find out the data type of the Voted_For_Obama vector

```
class(voted_for_obama)
```

```
## [1] "logical"
```

Create a vector called Party_ID that contains the following:

"Dem", "Dem", "Dem", "Rep", "Ind", "Rep"

```
party_id <- c("dem", "dem", "dem", "rep", "ind", "rep")
```

Create a vector called Income_Level that contains the following:

"High", "Low", "Low", "High", "High", "Low"

```
income_level <- c("high", "low", "low", "high", "high", "low")
```

Create a vector called Approval that contains the following:

70, 80, 68, 20, 10, 60

```r
approval <- c(70, 80, 68, 20, 10, 60)
```

Create a data frame called vote.data by combining the six vectors
you have created above.

```r
voter_data <- data.frame(ID, race, voted_for_obama, party_id, income_level, approval,
                         stringsAsFactors = TRUE)
```

We have seen above that the Race vector was a character vector.
We have learned in the joint exercise that we should treat a
vector like this one as a factor, not as a character.
A nice thing about data frame objects is that, it will
convert vectors like the Race vector into factors automatically.
To convince yourself that the Race vector included in the vote.data
is indeed a factor vector, apply the is.factor function on it.

```r
is.factor(voter_data$race)
```

```
## [1] TRUE
```

Notice that is.factor(Race) returns FALSE but the command you wrote above should return TRUE.

How many people in this data set voted for Obama?

That is, for how many observations does the variable Voted_For_Obama take the value of TRUE? Write R commands that gives you the answer. Note that I know the answer is 4. What you need to give me is the command that gives us the answer 4.

Hint 1: there is a function called length that returns the length of a vector. One way to do this task is to measure the length of a subset of the Voted_For_Obama vector where the values are TRUE.

Hint 2: there is a function called nrow that returns the number of rows of a matrix or a data frame. Another way to do this task is to measure the number of rows of a subset of the vote.data object for which Voted_For_Obama is equal to TRUE.

```r
number_voted_for_obama <- nrow(subset(voter_data, voted_for_obama == TRUE))
number_voted_for_obama
```

```
## [1] 4
```

There is one person in this data set who identifies himself as "Ind" (Independent). Did he vote for Obama?

Again, I know he didn't. Give me the command that gives us the answer FALSE

```r
voter_data$voted_for_obama[voter_data$party_id == "ind"]
```

```
## [1] FALSE
```

```r
subset(voter_data, party_id == "ind")$voted_for_obama
```

```
## [1] FALSE
```

Create a subset of the data set that contains only "white" people. Store this smaller data set into an object called vote.white

```r
vote.white <- subset(voter_data, race == "White")
```

Show the third column of the newly created data set vote.white

```r
vote.white[3]
```

```
##   voted_for_obama
## 2            TRUE
## 5           FALSE
## 6            TRUE
```

How many white voters in this mini data set voted for Obama?

Again, I know the answer is 2. Write a command that gives the answer 2.

```
sum(vote.white$voted_for_obama)
```

```
## [1] 2
```

```
# This counts just the number of TRUEs within a logical variable by treating
# TRUE as 1 and FALSE as 0
```

Some extra exercises

Modifying a data frame object ——————————————————————-

We have created a data frame called my.data in gv900-week5-Review.R.

Let's see how we can add variables to an existing data frame.

To add a new variable, you also use the $ symbol.

Specifically, we write DATAFRAME $ NEW_VARIABLE_NAME <- VALUES

For example, in order to add a new variable called Population

that contains the following values:

318946000, 64105654, 127090000, 1367420000, 203322000, 80781000, 87354300

We write

my.data $ Population <- c(318946000, 64105654, 127090000, 1367420000, 203322000, 80781000, 87354300)

my.data

If the Console window is wide enough, it should be showing up like this:

> my.data

Country_ID Country_Name Regime_Type GDP_PC EU_Member Population

1 1 United States Democracy 51163 FALSE 318946000

2 2 United Kingdom Democracy 39367 TRUE 64105654

3 3 Japan Democracy 46838 FALSE 127090000

4 4 China Dictatorship 6070 FALSE 1367420000

5 5 Brazil Democracy 11347 FALSE 203322000

6 6 Germany Democracy 41376 TRUE 80781000

7 7 Egypt Dictatorship 3115 FALSE 87354300

We can see that a new column is now added at the end (far right).

If the Console window is not wide enough, it may show up like this:

> my.data

Country_ID Country_Name Regime_Type GDP_PC EU_Member

1 1 United States Democracy 51163 FALSE

2 2 United Kingdom Democracy 39367 TRUE

3 3 Japan Democracy 46838 FALSE

4 4 China Dictatorship 6070 FALSE

5 5 Brazil Democracy 11347 FALSE

6 6 Germany Democracy 41376 TRUE

If you want to browse a data frame object, use the View function

View(my.data)

We can create a new variable that is an answer to some operations.

For example, GDP_PC measures per capita GDP (in 2013 US dollars).

This was calculated as a country's GDP divided by its population:

GDP_PC = GDP / Population

Therefore, if we multiply GDP_PC and Population, we can obtain its

GDP:

GDP = GDP_PC * Population

Create a variable within the my.data object called GDP which is equal to

the product of GDP_PC and Population (GDP_PC times Population).

world$gdppcap08
Write your commands here
my.data

It is a little bit difficult to read numbers like 1.631823e+13, which

means 1.631823 * 10^13. Let's create another variable that shows

re-scaled GDPs by dividing the raw GDP by 1000000 (on million).

my.data $ GDP_mil <- my.data $ GDP / 1000000
my.data

So, now this new variable GDP_mil is shown in 1 million dollars.

For example, the value of GDP_mil is 16318234.2 for the United

States, which means that GDP of the US is 16318234.2 million dollars.

Create a new variable within my.data called Is_Democracy that

is a logical vector that tells us whether or not a country is democratic.

Hint: utilize the Regime_Type variable included in the my.data

object.

Write your commands here

End of file