

hw_2

Andrew Mikolinski

2023-09-24

```
library(dplyr) library(tidyverse)
data("mtcars")
```

Question 1

This code appears to be attempting a subset, the proper code for a subset is as follows

```
mtcars[mtcars$cyl < 6, ]
```

```
##           mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Datsun 710   22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## Merc 240D   24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
## Merc 230    22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
## Fiat 128    32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## Honda Civic 30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## Toyota Corolla 33.9  4  71.1  65 4.22 1.835 19.90  1  1    4    1
## Toyota Corona 21.5  4 120.1  97 3.70 2.465 20.01  1  0    3    1
## Fiat X1-9    27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
## Porsche 914-2 26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## Lotus Europa 30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
## Volvo 142E   21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

Avoid the -x:y situation, theres a few options for fixing this depending on what you're trying to accomplish but this is one of them

```
mtcars[-1:0 & 0:3, ]
```

```
##           mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Datsun 710   22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## Duster 360   14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 280C    17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## Cadillac Fleetwood 10.4  8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Honda Civic 30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## AMC Javelin  15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Porsche 914-2 26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## Maserati Bora 15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
```

This command just needed a second “=” sign

```
mtcars[mtcars$cyl == 8, ]
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
## Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
## Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
## Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
## Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
## Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
## Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
## Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
## Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4
## Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0	3	2
## AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0	3	2
## Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0	3	4
## Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0	0	3	2
## Ford Pantera L	15.8	8	351.0	264	4.22	3.170	14.50	0	1	5	4
## Maserati Bora	15.0	8	301.0	335	3.54	3.570	14.60	0	1	5	8

This command needed more specificity because it was basically saying “either 4 cylinders or a 6 anywhere in the data” so if you’re looking for either 4 or 6 cylinders this is the code for it

```
mtcars[mtcars$cyl == 4 | mtcars$cyl == 6, ]
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
## Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
## Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
## Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
## Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
## Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
## Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
## Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
## Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
## Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
## Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1	4	1
## Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
## Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1	4	1
## Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0	3	1
## Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1	1	4	1
## Porsche 914-2	26.0	4	120.3	91	4.43	2.140	16.70	0	1	5	2
## Lotus Europa	30.4	4	95.1	113	3.77	1.513	16.90	1	1	5	2
## Ferrari Dino	19.7	6	145.0	175	3.62	2.770	15.50	0	1	5	6
## Volvo 142E	21.4	4	121.0	109	4.11	2.780	18.60	1	1	4	2

Question 2

When you set `x=1:5` you're setting `x` as a vector so when you then put the command `x[NA]` you're telling R to find the subset of `x` that is NA (NA represents missing values). But there is no NA value in `x=1:5`. Since NA is a logical vector by itself, R just keeps repeating NA for the number of values in `x` as the output.

```
x = 1:5
x[NA]
```

```
## [1] NA NA NA NA NA
```

Question 3

This returns an error because without the comma the command doesn't let you keep all rows or columns in the matrix/array. The comma tells R that you're looking for just rows 1-15, if you were to plug in a value after the comma like 7, R would return rows 1-15 and only column 7. In a multi dimensional array/matrix, the commas specify the values for specific dimensions that you're looking for.

```
mtcars[1:15, ]
```

```
##           mpg cyl  disp  hp drat   wt  qsec vs am gear carb
## Mazda RX4      21.0   6 160.0 110 3.90 2.620 16.46 0  1    4    4
## Mazda RX4 Wag  21.0   6 160.0 110 3.90 2.875 17.02 0  1    4    4
## Datsun 710      22.8   4 108.0  93 3.85 2.320 18.61 1  1    4    1
## Hornet 4 Drive  21.4   6 258.0 110 3.08 3.215 19.44 1  0    3    1
## Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02 0  0    3    2
## Valiant         18.1   6 225.0 105 2.76 3.460 20.22 1  0    3    1
## Duster 360      14.3   8 360.0 245 3.21 3.570 15.84 0  0    3    4
## Merc 240D       24.4   4 146.7  62 3.69 3.190 20.00 1  0    4    2
## Merc 230        22.8   4 140.8  95 3.92 3.150 22.90 1  0    4    2
## Merc 280        19.2   6 167.6 123 3.92 3.440 18.30 1  0    4    4
## Merc 280C       17.8   6 167.6 123 3.92 3.440 18.90 1  0    4    4
## Merc 450SE      16.4   8 275.8 180 3.07 4.070 17.40 0  0    3    3
## Merc 450SL      17.3   8 275.8 180 3.07 3.730 17.60 0  0    3    3
## Merc 450SLC     15.2   8 275.8 180 3.07 3.780 18.00 0  0    3    3
## Cadillac Fleetwood 10.4   8 472.0 205 2.93 5.250 17.98 0  0    3    4
```

Question 4

The first line of this code sets up a matrix comprised of 1, 2, 3, 5, 6, 7, and 3 missing values, and tells R that the values should be split into 3 rows in the order in which each of the values appear in the code.

The second line is checking if there are any missing values in the provided matrix. Normally, without the “= 0” this would return 3 NA values, and the command “is.na(x)” would return a 3 row 3 column table with 6 “False” and 3 “True” values. With “= 0”, you’re telling R that even though there are a few missing values in the set it should not consider them missing values. So now if you put the command is.na(x) it will return a table with 3 rows and 3 columns and all of them will say “False”.

```
x = matrix(c(1:3, NA, 5:7, NA, NA), nrow = 3)
x[is.na(x)]
```

```
## [1] NA NA NA
```

Question 5

```
data("mtcars")
mtcars$mpg
```

```
## [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 17.8 16.4 17.3 15.2 10.4
## [16] 10.4 14.7 32.4 30.4 33.9 21.5 15.5 15.2 13.3 19.2 27.3 26.0 30.4 15.8 19.7
## [31] 15.0 21.4
```

```
mpg_2 <-
  ifelse(mtcars$mpg < 16, "Low",
        ifelse(mtcars$mpg >= 16 | mtcars$mpg < 21, "Low_intermediate",
              ifelse(mtcars$mpg >= 21 | mtcars$mpg < 26, "Intermediate_high",
                    ifelse(mtcars$mpg >= 26, "High"))))
mpg_2
```

```
## [1] "Low_intermediate" "Low_intermediate" "Low_intermediate" "Low_intermediate"
## [5] "Low_intermediate" "Low_intermediate" "Low" "Low_intermediate"
## [9] "Low_intermediate" "Low_intermediate" "Low_intermediate" "Low_intermediate"
## [13] "Low_intermediate" "Low" "Low" "Low"
## [17] "Low" "Low_intermediate" "Low_intermediate" "Low_intermediate"
## [21] "Low_intermediate" "Low" "Low" "Low"
## [25] "Low_intermediate" "Low_intermediate" "Low_intermediate" "Low_intermediate"
## [29] "Low" "Low_intermediate" "Low" "Low_intermediate"
```