Quiz 4

Mihai M. Craiu

16/7/2020

Question 1

The American Community Survey distributes downloadable data about United States communities. Download the 2006 microdata survey about housing for the state of Idaho using download.file() from here:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06hid.csv

and load the data into R. The code book, describing the variable names is here:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FPUMSDataDict06.pdf

Apply strsplit() to split all the names of the data frame on the characters "wgtp".

What is the value of the 123 element of the resulting list

```
# Download file...

Q1Url <- "https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2Fss06hid.csv"
Q1 <- read.csv(Q1Url)
head(Q1)
```

```
##   RT SERIALNO DIVISION PUMA REGION ST  ADJUST WGTP NP TYPE ACR AGS BDS BLD BUS
## 1  H      186        8  700      4 16 1015675   89  4    1   1  NA   4   2   2
## 2  H      306        8  700      4 16 1015675  310  1    1  NA  NA   1   7  NA
## 3  H      395        8  100      4 16 1015675  106  2    1   1  NA   3   2   2
## 4  H      506        8  700      4 16 1015675  240  4    1   1  NA   4   2   2
## 5  H      835        8  800      4 16 1015675  118  4    1   2   1   5   2   2
## 6  H      989        8  700      4 16 1015675  115  4    1   1  NA   3   2   2
##   CONP ELEP FS FULP GASP HFL INSP KIT MHP MRGI MRGP MRGT MRGX PLM RMS RNTM RNTP
## 1   NA  180  0    2    3   3  600   1  NA    1 1300    1    1   1   9   NA   NA
## 2   NA   60  0    2    3   3   NA   1  NA   NA   NA   NA   NA   1   2    2  600
## 3   NA   70  0    2   30   1  200   1  NA   NA   NA   NA    3   1   7   NA   NA
## 4   NA   40  0    2   80   1  200   1  NA    1  860    1    1   1   6   NA   NA
## 5   NA  250  0    2    3   3  700   1  NA    1 1900    1    1   1   7   NA   NA
## 6   NA  130  0    2    3   3  250   1  NA    1  700    1    1   1   6   NA   NA
##   SMP TEL TEN VACS VAL VEH WATP YBL FES  FINCP FPARC GRNTP GRPIP HHL HHT  HINCP
## 1  NA   1   1   NA  17   3  840   5   2 105600     2    NA    NA   1   1 105600
## 2  NA   1   3   NA  NA   1    1   3  NA    NA    NA   660    23   1   4  34000
## 3  NA   1   2   NA  18   2   50   5   7  9400     2    NA    NA   1   3   9400
## 4 400   1   1   NA  19   3  500   2   1 66000     1    NA    NA   1   1  66000
## 5 650   1   1   NA  20   5    2   3   1 93000     2    NA    NA   1   1  93000
## 6 400   1   1   NA  15   2 1200   5   2 61000     1    NA    NA   1   1  61000
##   HUGCL HUPAC HUPAOC HUPARC LNGI MV NOC NPF NPP NR NRC OCPIP PARTNER PSF R18
```

```
## 1     0     2      2      2    1  4   2   4   0  0   2    18       0   0   1
## 2     0     4      4      4    1  3   0  NA   0  0   0    NA       0   0   0
## 3     0     2      2      2    1  2   1   2   0  0   1    23       0   0   1
## 4     0     1      1      1    1  3   2   4   0  0   2    26       0   0   1
## 5     0     2      2      2    1  1   1   4   0  0   1    36       0   0   1
## 6     0     1      1      1    1  4   2   4   0  0   2    26       0   0   1
##   R60 R65 RESMODE SMOCP SMX SRNT SVAL TAXP WIF WKEXREL WORKSTAT FACRP FAGSP
## 1   0   0       1  1550   3    0    1   24   3       2        3     0     0
## 2   0   0       2    NA  NA    1    0   NA  NA      NA       NA     0     0
## 3   0   0       1   179  NA    0    1   16   1      13       13     0     0
## 4   0   0       2  1422   1    0    1   31   2       2        1     0     0
## 5   0   0       1  2800   1    0    1   25   3       1        1     0     0
## 6   0   0       2  1330   2    0    1    7   1       7        3     0     0
##   FBDSP FBLDP FBUSP FCONP FELEP FFSP FFULP FGASP FHFLP FINSP FKITP FMHP FMRGIP
## 1     0     0     0     0     0    0     0     0     0     0     0    0      0
## 2     0     0     0     0     0    0     0     0     0     0     0    0      0
## 3     0     0     0     0     0    0     0     0     0     0     0    0      0
## 4     0     0     0     0     0    0     0     0     0     0     0    0      0
## 5     0     0     0     0     0    0     0     0     0     0     0    0      0
## 6     0     0     0     0     0    0     0     0     0     1     0    0      0
##   FMRGP FMRGTP FMRGXP FMVYP FPLMP FRMSP FRNTMP FRNTP FSMP FSMXHP FSMXSP FTAXP
## 1     0      0      0     0     0     0      0     0    0      0      0     0
## 2     0      0      0     0     0     0      0     0    0      0      0     0
## 3     0      0      0     0     0     0      0     0    0      0      0     0
## 4     0      0      0     0     0     0      0     0    0      0      0     0
## 5     0      0      0     0     0     0      0     0    0      0      0     0
## 6     0      0      0     0     0     0      0     0    0      0      0     1
##   FTELP FTENP FVACSP FVALP FVEHP FWATP FYBLP wgtp1 wgtp2 wgtp3 wgtp4 wgtp5
## 1     0     0      0     0     0     0     0    87    28   156    95    26
## 2     0     0      0     0     0     0     1   539   363   293   422   566
## 3     0     0      0     0     0     0     0   187    35   184   178    83
## 4     0     0      0     0     0     0     0   232   406   234   270   249
## 5     0     0      0     0     0     0     0   107   194   129    41   156
## 6     0     0      0     0     0     1     0   191   197   127   115   115
##   wgtp6 wgtp7 wgtp8 wgtp9 wgtp10 wgtp11 wgtp12 wgtp13 wgtp14 wgtp15 wgtp16
## 1    25    95    93    93     91     87    166     90     25    153     89
## 2   289    87   242   453    453    334    358    414    102    281     99
## 3    95    31    32   177    118    110    114    184    107     95    115
## 4   242   406   249   287     67     72    413    399     77    245    424
## 5   174    47   113   101     33    115     52    113     95    135    206
## 6   107   119    34    32     30    123    199    117     33    109    117
##   wgtp17 wgtp18 wgtp19 wgtp20 wgtp21 wgtp22 wgtp23 wgtp24 wgtp25 wgtp26 wgtp27
## 1    148     82     25    180     90     24    140     92     25     27     86
## 2    108    278    131    407    447    264    352    238    390    336    122
## 3     33    118    120     37    184     35    176    176    110    103     29
## 4     67     63    226    254    238     69    238    255    239    248     69
## 5    100    185    135    279    116     33    105    244     38     30    230
## 6     31    115    201    190    184    198    113    109    117    111    110
##   wgtp28 wgtp29 wgtp30 wgtp31 wgtp32 wgtp33 wgtp34 wgtp35 wgtp36 wgtp37 wgtp38
## 1     84     87     93     90    149     91     28    143     81    144     95
## 2    374    482    468    335    251    613    104    284    116     91    326
## 3     30    197    127     92    118    177     99     99    109     34    100
## 4    234    247    437    423     74     61    401    267     72    388    335
## 5    123    123    243    120    238     98     90    107     44    122     32
```

```
## 6     33     37     36    110    183    114     35    134    119     32    121
##   wgtp39 wgtp40 wgtp41 wgtp42 wgtp43 wgtp44 wgtp45 wgtp46 wgtp47 wgtp48 wgtp49
## 1     27     22     90    171     27     83    153    148     92     91     91
## 2    102    361    107    253    321    289     96    343    564    274    118
## 3    105     33    173     36    168    175     99    103     30     35    155
## 4    229    236    239     65    259    247    230    225     82    220    233
## 5    127    195    116     36    135    237     33     33    249    102     84
## 6    188     33     34     32    109    115    115    112    119    192    186
##   wgtp50 wgtp51 wgtp52 wgtp53 wgtp54 wgtp55 wgtp56 wgtp57 wgtp58 wgtp59 wgtp60
## 1     93     90     26     94    142     24     91     29     84    148     30
## 2    118    321    261    130    463    294    479    391    307    476    283
## 3    102     95    107    185    120    114    113     36    115    103     29
## 4    419    390     69     74    391    276     70    422    409    223    245
## 5    224    119    250    119    125    126     32    112     33    131     45
## 6    213    106     34    124    179    106    107    190    112     34     35
##   wgtp61 wgtp62 wgtp63 wgtp64 wgtp65 wgtp66 wgtp67 wgtp68 wgtp69 wgtp70 wgtp71
## 1     93    143     24     88    147    145     91     83     83     86     81
## 2    116    353    323    374    106    236    380    313     90     94    292
## 3    183     35    179    169     95    110     28     34    233     97    123
## 4    269    488    221    250    247    240    415    234    219     66     68
## 5    101    165    125     41    191    195     49    119     92     44    127
## 6     32     34    119    123    122    121    123    196    196    207    120
##   wgtp72 wgtp73 wgtp74 wgtp75 wgtp76 wgtp77 wgtp78 wgtp79 wgtp80
## 1     27     93    151     28     79     25    101    157    129
## 2    401     81    494    346    496    615    286    454    260
## 3    119    168    107     95    101     30    124    106     31
## 4    359    385     71    234    421     76     77    242    231
## 5     36    119    121    116    209     97    176    144     38
## 6     34    109    199    116    110    211    120     31    189
```

```
# Computing solution...

Q1_colnames <- names(Q1)
strsplit(Q1_colnames, "^wgtp")[[123]]
```

```
## [1] ""   "15"
```

Options:

a. "wgt" "15"

b. "wgtp"

c. "" "15"

d. "wgtp" "15"

Question 2

Load the Gross Domestic Product data for the 190 ranked countries in this data set:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FGDP.csv

Remove the commas from the GDP numbers in millions of dollars and average them. What is the average?

Original data sources:

http://data.worldbank.org/data-catalog/GDP-ranking-table

```
# Downloading file...

Q2_Url <- "https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FGDP.csv"
Q2_Path <- "C:/Users/Mihai/Desktop/Data_Science_JHU_Coursera/Getting_and_Cleaning_Data/Week_4/Q2GDP.csv
download.file(Q2_Url, Q2_Path, method = "curl")

# Loading and tidying data...

Q2_File <- read.csv(Q2_Path, nrow = 190, skip = 4)
Q2_File <- Q2_File[,c(1, 2, 4, 5)]
colnames(Q2_File) <- c("CountryCode", "Rank", "Country", "Total")
head(Q2_File)
```

```
##   CountryCode Rank        Country        Total
## 1         USA    1  United States  16,244,600
## 2         CHN    2          China   8,227,103
## 3         JPN    3          Japan   5,959,718
## 4         DEU    4        Germany   3,428,131
## 5         FRA    5         France   2,612,878
## 6         GBR    6 United Kingdom   2,471,784
```

```
# Computing solution...

Q2_File$Total <- as.integer(gsub(",", "", Q2_File$Total))
mean(Q2_File$Total, na.rm = T)
```

```
## [1] 377652.4
```

Options:

a. 377652.4

b. 381668.9

c. 387854.4

d. 293700.3