Quiz3

Mihai M. Craiu

14/7/2020

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:

data.csv

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

codebook.pdf

Create a logical vector that identifies the households on greater than 10 acres who sold more than $10,000 worth of agriculture products. Assign that logical vector to the variable agricultureLogical. Apply the which() function like this to identify the rows of the data frame where the logical vector is TRUE.

which(agricultureLogical)

What are the first 3 values that result?

```r
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
```

```r
#Computing solution...

agricultureLogical <- Q1$ACR == 3 & Q1$AGS == 6
which(agricultureLogical)
```

```
##  [1]  125  238  262  470  555  568  608  643  787  808  824  849  952  955 1033
## [16] 1265 1275 1315 1388 1607 1629 1651 1856 1919 2101 2194 2403 2443 2539 2580
## [31] 2655 2680 2740 2838 2965 3131 3133 3163 3291 3370 3402 3585 3652 3852 3862
## [46] 3912 4023 4045 4107 4113 4117 4185 4198 4310 4343 4354 4448 4453 4461 4718
## [61] 4817 4835 4910 5140 5199 5236 5326 5417 5531 5574 5894 6033 6044 6089 6275
## [76] 6376 6420
```

Options:

a. 125, 238,262

b. 403, 756, 798

c. 236, 238, 262

d. 59, 460, 474

Question 2

Using the jpeg package read in the following picture of your instructor into R

picture.jpg

Use the parameter native=TRUE. What are the 30th and 80th quantiles of the resulting data? (some Linux systems may produce an answer 638 different for the 30th quantile)

```
# Loading package...

library(jpeg)

# Downloading file...

Q2Url <- "https://d396qusza40orc.cloudfront.net/getdata%2Fjeff.jpg"
Q2Path = ’C:/Users/Mihai/Desktop/Data_Science_JHU_Coursera/Getting_and_Cleaning_Data/Week_3/Q2.jpg’
download.file(Q2Url, Q2Path, mode = ’wb’)
Q2 <- readJPEG(Q2Path, native = TRUE)

# Computing solution...

quantile(Q2, probs = c(0.3, 0.8))
```

```
##       30%       80%
## -15258512 -10575416
```

‘some Linux systems may produce an answer 638 different for the 30th quantile.’

Mine is Linux, so…

```
paste(quantile(Q2, probs = 0.3) - 638, quantile(Q2, probs = 0.8))
```

```
## [1] "-15259150 -10575416"
```

Options:

a. -15259150 -10575416

b. -10904118 -10575416

c. 10904118 -594524

d. -16776430 -15390165