Lab5.R

praja

2021-12-31

```
#install.packages('RCurl') #installs RCurl package
#install.packages('jsonlite') #installs jsonlite package

library('RCurl') #use library RCurl
library('jsonlite') #use library jsonlite

station_link <- 'https://gbfs.citibikenyc.com/gbfs/en/station_status.json' #the JSON file data i
s stored in station_link
apiOutput <- getURL(station_link) #apiOutput stores the downloaded data from station_link
apiData <- fromJSON(apiOutput) #converts data from JSON format to R objects
stationStatus <- apiData$data$stations #stores the stations column from apiData
cols <- c('num_bikes_disabled','num_docks_disabled', 'station_id',
          'num_ebikes_available', 'num_bikes_available', 'num_docks_available') #forms a vector
 with given column names
stationStatus = stationStatus[,cols]

#1. Explain what you    see if  you type    in  the station_link    URL into    a   browser (in
 a   comment,    write   what    you see)
#Answer: The link contains a JSON file which gives a clear and easy to understand data format wh
ich can be used over the web.

#2. Provide a   comment explaining  each    line    of  code.

#3. Use str(    )   to  find    out the structure   of  apiOutput   and apiData.    Report  (via
a   comment)    what    you found   and explain the difference  between these   two objects
str(apiData)
```

```
## List of 3
##  $ data        :List of 1
##   ..$ stations:'data.frame': 1594 obs. of  15 variables:
##   .. ..$ num_bikes_available          : int [1:1594] 36 14 19 28 33 42 1 22 43 29 ...
##   .. ..$ last_reported                : int [1:1594] 1640847723 1640934254 1640931822 164093
1818 1640940863 1640931817 1640938040 1640935679 1640936715 1640931872 ...
##   .. ..$ num_docks_available          : int [1:1594] 16 19 8 33 15 9 18 9 12 19 ...
##   .. ..$ num_ebikes_available         : int [1:1594] 2 0 0 1 1 2 0 2 5 0 ...
##   .. ..$ is_renting                   : int [1:1594] 0 1 1 1 1 1 1 1 0 1 ...
##   .. ..$ is_returning                 : int [1:1594] 0 1 1 1 1 1 1 1 0 1 ...
##   .. ..$ station_status               : chr [1:1594] "active" "active" "active" "active" ...
##   .. ..$ eightd_has_available_keys    : logi [1:1594] FALSE FALSE FALSE FALSE FALSE FALSE
...
##   .. ..$ legacy_id                    : chr [1:1594] "72" "79" "82" "83" ...
##   .. ..$ num_docks_disabled           : int [1:1594] 0 0 0 0 0 0 0 0 0 0 ...
##   .. ..$ is_installed                 : int [1:1594] 1 1 1 1 1 1 1 1 1 1 ...
##   .. ..$ num_bikes_disabled           : int [1:1594] 3 0 0 1 2 2 0 0 1 2 ...
##   .. ..$ station_id                   : chr [1:1594] "72" "79" "82" "83" ...
##   .. ..$ eightd_active_station_services:List of 1594
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
```

```
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
##    .. .. ..$ : NULL
```

```
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. ..$ : NULL
##   .. .. .. [list output truncated]
##   .. ..$ valet                         :'data.frame':    1594 obs. of  7 variables:
##   .. .. ..$ dock_blocked_count: int [1:1594] NA NA NA NA NA NA NA NA NA NA ...
##   .. .. ..$ off_dock_capacity : int [1:1594] NA NA NA NA NA NA NA NA NA NA ...
##   .. .. ..$ off_dock_count    : int [1:1594] NA NA NA NA NA NA NA NA NA NA ...
##   .. .. ..$ active            : logi [1:1594] NA NA NA NA NA NA ...
##   .. .. ..$ valet_revision    : int [1:1594] NA NA NA NA NA NA NA NA NA NA ...
##   .. .. ..$ region            : chr [1:1594] NA NA NA NA ...
##   .. .. ..$ station_id        : chr [1:1594] NA NA NA NA ...
##  $ last_updated: int 1640941680
##  $ ttl         : int 5
```

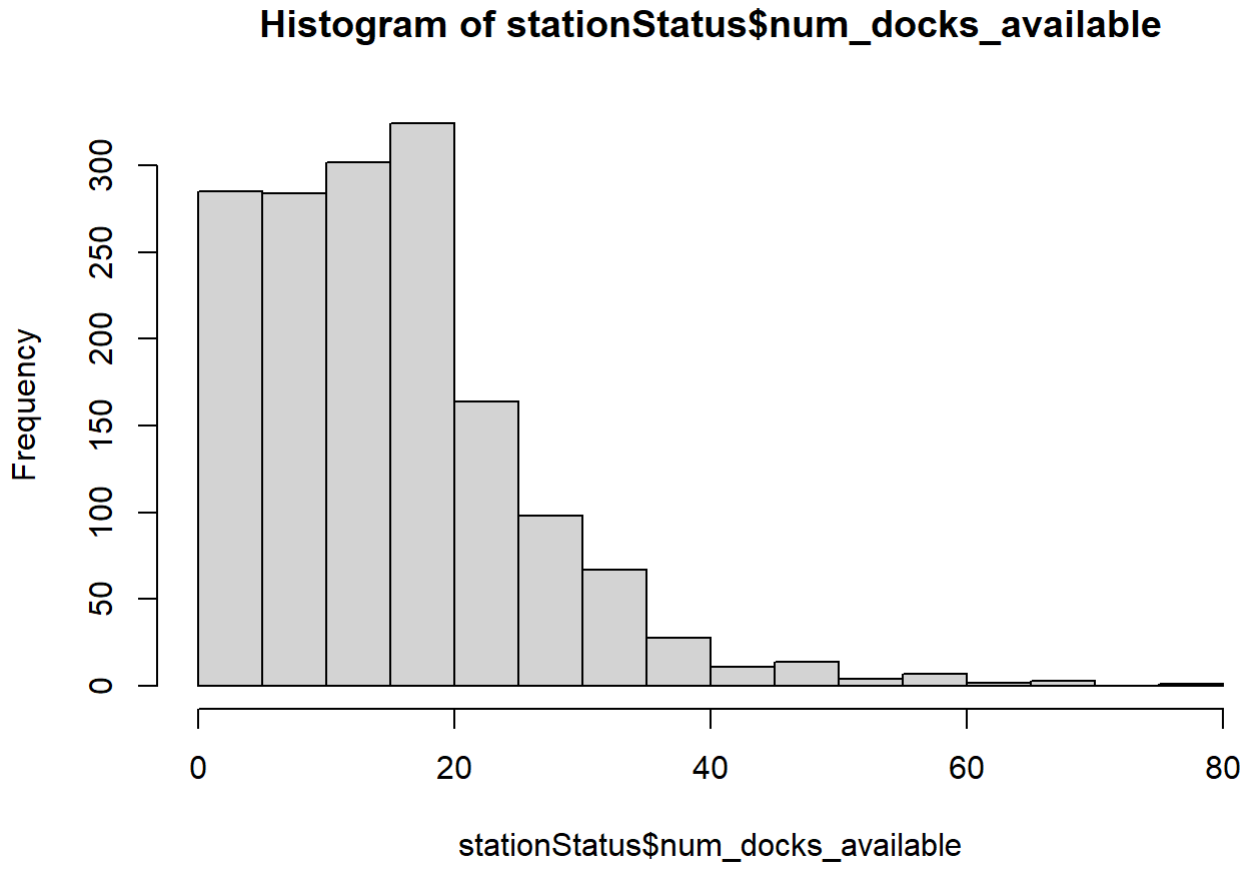
```
str(apiOutput)
```

```
##  chr "{\"data\":{\"stations\":[{\"num_bikes_available\":36,\"last_reported\":1640847723,\"num
_docks_available\":16,\""| __truncated__
```

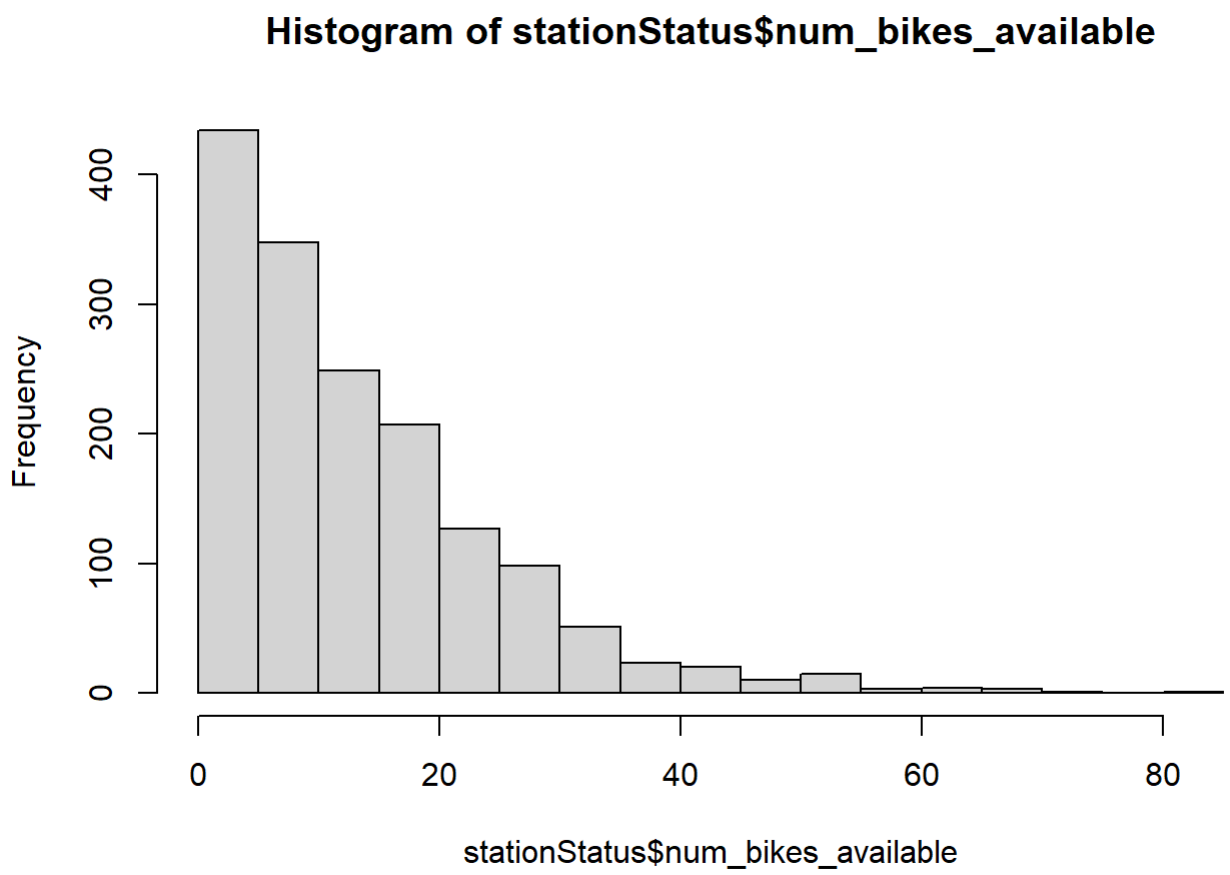
```
#Answer: ApiData is more structured than ApiOutput since we converted it to R objects

#4. The apiOutput   object  can also    be  examined    with    a   custom  function    from
 the
#jsonlite   package called  prettify(   ).  Run this    command and explain what    you
#found (in  a   comment).
#prettify(apiOutput)
#We find out that using prettify function to apiOutput it gives a more simplified, indented
#and clearer definition to apiOutput, ie presentation of data is clear.

#5.Explain  stationStatus   (what   type    of  object, what    information is  available)
str(stationStatus)
```

```
## 'data.frame':    1594 obs. of  6 variables:
##  $ num_bikes_disabled  : int  3 0 0 1 2 2 0 0 1 2 ...
##  $ num_docks_disabled  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ station_id          : chr  "72" "79" "82" "83" ...
##  $ num_ebikes_available: int  2 0 0 1 1 2 0 2 5 0 ...
##  $ num_bikes_available : int  36 14 19 28 33 42 1 22 43 29 ...
##  $ num_docks_available : int  16 19 8 33 15 9 18 9 12 19 ...
```

```
#In stationStatus, we have 1578 observations with 6 variables, 5 numeric and 1 character(station
_id)
#stationStatus is a dataframe.

#6. Generate    a   histogram   of  the number  of  docks   available
hist(stationStatus$num_docks_available)
```




```
#7. Generate    a   histogram   of  the number  of  bikes   available
hist(stationStatus$num_bikes_available)
```




```
#8. How many stations have at least one ebike?
sum(stationStatus$num_ebikes_available > 0)
```

```
## [1] 1010
```

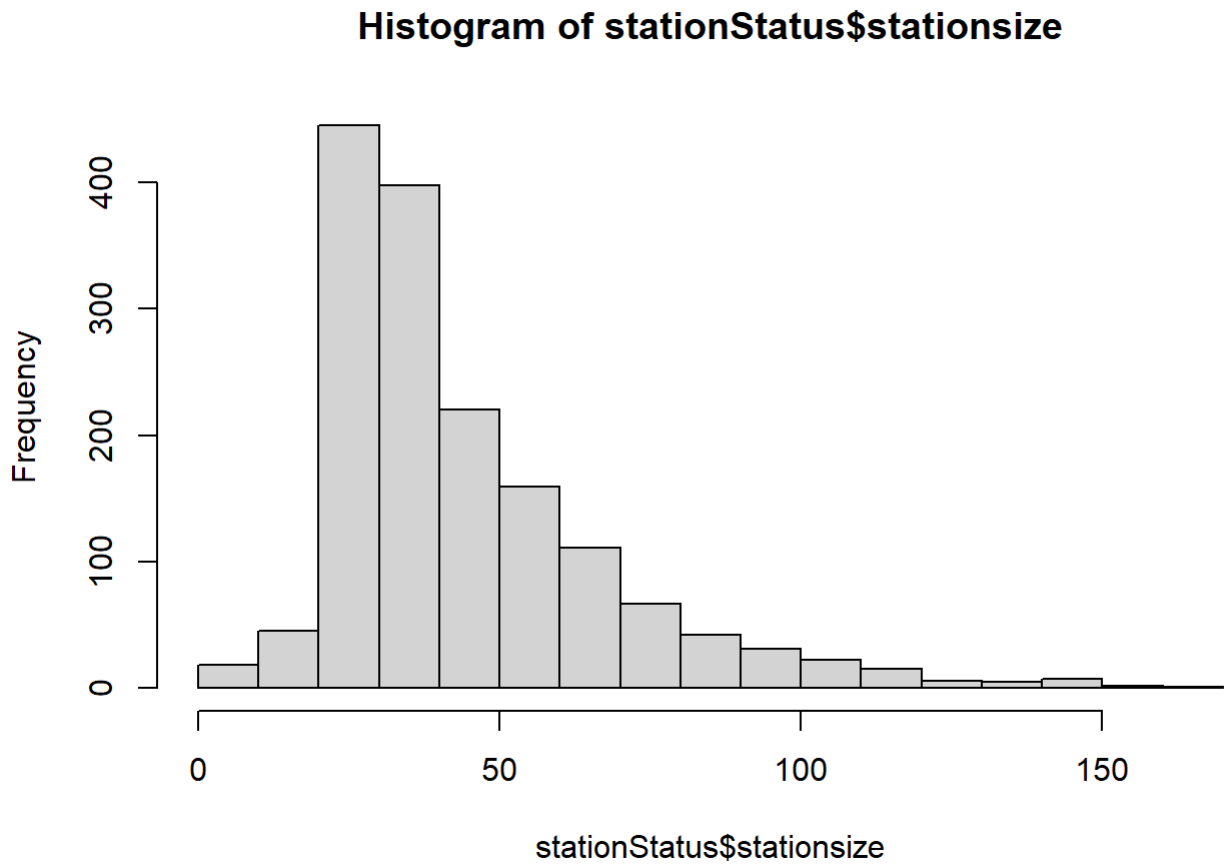
```
#9. Explore stations with at least one ebike by create a new dataframe, that
 only has
#stations with at least one eBike.
eBikesDF <- stationStatus[stationStatus$num_ebikes_available > 0,]

#10. Calculate the mean of 'num_docks_available' for this new dataframe
mean(eBikesDF$num_docks_available)
```

```
## [1] 14.8604
```

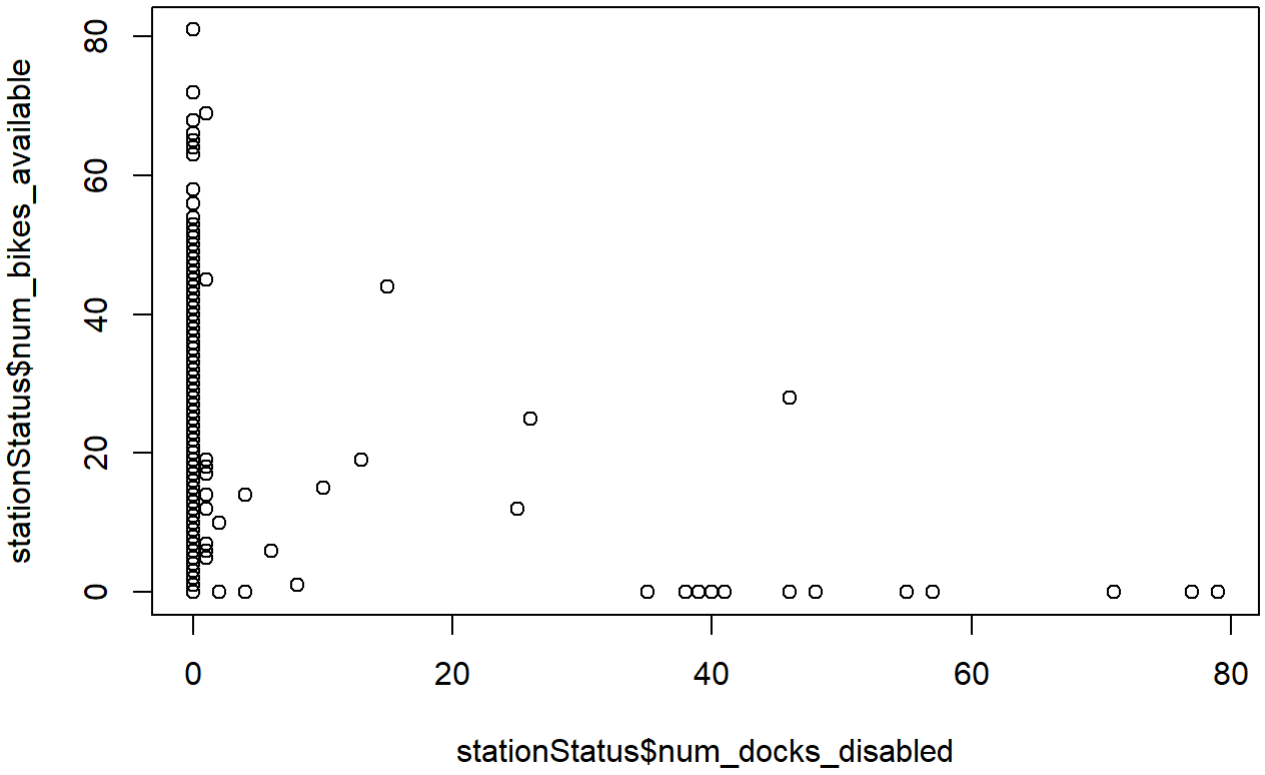
```
#mean is 15.15

#11. Calculate the mean of 'num_docks_available' for for the full 'stationStatus'
#dataframe. In a comment, explain how different are the two means?
mean(stationStatus$num_docks_available)
```

```
## [1] 15.3463
```

```
#mean is 15.52 and the previous mean is 15.15, so there is a deviation of 0.37

#12. . Create   a   new attribute,  called  'stationSize',  which   is  the total   number  of
 "slots"
#available  for a   bike    (that   might,  or  might   not,    have    a   bike    in  it  no
w).   Run a
#histogram  on  this    variable    and review  the distribution.
stationStatus$stationsize <- stationStatus$num_bikes_available +
                             stationStatus$num_bikes_disabled +
                             stationStatus$num_docks_available +
                             stationStatus$num_bikes_available +
                             stationStatus$num_docks_disabled

hist(stationStatus$stationsize)
```



```
#highest frequency is approx 900 between 0 to 50 stationSize and the frequency gradually increas
es and then decreases again.

#13. Use the plot( ) command to produce an X-Y scatter plot with the number of
#occupied docks on the X-axis and the number of available bikes on the Y-axis.
# Explain the results plot.
plot(stationStatus$num_docks_disabled , stationStatus$num_bikes_available)
```




```
#It displays a scatter plot between the docks that are occupied and bikes that are available. It
shows that as and when the number of docks get disabled the number of bikes get available.
```