

Assignment -2

Akash Chauhan

11/14/2021

```
#It is given that a collection of health-related surveys conducted each year of more than 400,000 American  
# Data field info  
# 1.MMSA      The name of the metropolitan area, metropolitan area or metropolitan division available in  
# 2.total_percent_at_risk  The percent of individuals in that area that are at high risk of becoming s  
#3.high_risk_per_ICU_bed   The number of high risk individuals per ICU bed in that area  
#4.high_risk_per_hospital  The number of high risk individuals per hospital in that area  
#5.ICU_beds  The number of ICU beds in the area, based on the Kaiser Family Foundation's data  
#6.hospitals  The number of hospitals in the area, based on the Kaiser Family Foundation's data  
#7.total_at_risk  The total number of high risk individuals in the area, per CDC's BRFSS  
  
#Data format : It is a .csv file downloaded from a data source on Github, where the values are separate
```

```
library(readr)  
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5      v dplyr    1.0.7  
## v tibble  3.1.5      v stringr 1.4.0  
## v tidyr   1.1.4      v forcats 0.5.1  
## v purrr   0.3.4
```

```
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag()     masks stats::lag()
```

```
library(dplyr)
```

```
#Creating a variable that has all the content from .csv file
```

```
local_file <- "mmsa-icu-beds.csv"  
# reading csv data using the read function  
Coviddata <- read_csv(local_file)
```

```
## Rows: 136 Columns: 7
```

```
## -- Column specification -----
```

```
## Delimiter: ","
```

```
## chr (2): MMSA, total_percent_at_risk
```

```
## dbl (5): high_risk_per_ICU_bed, high_risk_per_hospital, icu_beds, hospitals,...
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Now, previewing the data loaded from mmsa-icu-beds into R
```

```
head(Coviddata)
```

```
## # A tibble: 6 x 7
##   MMSA      total_percent_a~ high_risk_per_I~ high_risk_per_h~ icu_beds hospitals
##   <chr>      <chr>                <dbl>                <dbl>      <dbl>      <dbl>
## 1 San Jua~ 52.88%                NA                NA        NA        NA
## 2 Manhatt~ 47.29%                4490.            8980.        8         4
## 3 Hilton ~ 62.72%                3904.            36439.       28        3
## 4 Kahului~ 59.13%                3861.            19303.       20        4
## 5 Spartan~ 66.12%                3786.            85188.       45        2
## 6 Baton R~ 66.60%                3460.            39001.      124       11
## # ... with 1 more variable: total_at_risk <dbl>
```

```
#performing two data cleaning function as per question
#1. Find the rows with the missing values
#2.Remove the rows with the missing values from the data
```

```
# checking for missing values in the data frame
is.na(Coviddata)
```

```
##           MMSA total_percent_at_risk high_risk_per_ICU_bed high_risk_per_hospital
## [1,] FALSE FALSE TRUE TRUE
## [2,] FALSE FALSE FALSE FALSE
## [3,] FALSE FALSE FALSE FALSE
## [4,] FALSE FALSE FALSE FALSE
## [5,] FALSE FALSE FALSE FALSE
## [6,] FALSE FALSE FALSE FALSE
## [7,] FALSE FALSE FALSE FALSE
## [8,] FALSE FALSE FALSE FALSE
## [9,] FALSE FALSE FALSE FALSE
## [10,] FALSE FALSE FALSE FALSE
## [11,] FALSE FALSE FALSE FALSE
## [12,] FALSE FALSE FALSE FALSE
## [13,] FALSE FALSE FALSE FALSE
## [14,] FALSE FALSE FALSE FALSE
## [15,] FALSE FALSE FALSE FALSE
## [16,] FALSE FALSE FALSE FALSE
## [17,] FALSE FALSE FALSE FALSE
## [18,] FALSE FALSE FALSE FALSE
## [19,] FALSE FALSE FALSE FALSE
## [20,] FALSE FALSE FALSE FALSE
## [21,] FALSE FALSE FALSE FALSE
## [22,] FALSE FALSE FALSE FALSE
## [23,] FALSE FALSE FALSE FALSE
## [24,] FALSE FALSE FALSE FALSE
## [25,] FALSE FALSE FALSE FALSE
## [26,] FALSE FALSE FALSE FALSE
## [27,] FALSE FALSE FALSE FALSE
## [28,] FALSE FALSE FALSE FALSE
## [29,] FALSE FALSE FALSE FALSE
## [30,] FALSE FALSE FALSE FALSE
## [31,] FALSE FALSE FALSE FALSE
## [32,] FALSE FALSE FALSE FALSE
## [33,] FALSE FALSE FALSE FALSE
## [34,] FALSE FALSE FALSE FALSE
## [35,] FALSE FALSE FALSE FALSE
## [36,] FALSE FALSE FALSE FALSE
```

##	[37,]	FALSE	FALSE	FALSE	FALSE
##	[38,]	FALSE	FALSE	FALSE	FALSE
##	[39,]	FALSE	FALSE	FALSE	FALSE
##	[40,]	FALSE	FALSE	FALSE	FALSE
##	[41,]	FALSE	FALSE	FALSE	FALSE
##	[42,]	FALSE	FALSE	FALSE	FALSE
##	[43,]	FALSE	FALSE	FALSE	FALSE
##	[44,]	FALSE	FALSE	FALSE	FALSE
##	[45,]	FALSE	FALSE	FALSE	FALSE
##	[46,]	FALSE	FALSE	FALSE	FALSE
##	[47,]	FALSE	FALSE	FALSE	FALSE
##	[48,]	FALSE	FALSE	FALSE	FALSE
##	[49,]	FALSE	FALSE	FALSE	FALSE
##	[50,]	FALSE	FALSE	FALSE	FALSE
##	[51,]	FALSE	FALSE	FALSE	FALSE
##	[52,]	FALSE	FALSE	FALSE	FALSE
##	[53,]	FALSE	FALSE	FALSE	FALSE
##	[54,]	FALSE	FALSE	FALSE	FALSE
##	[55,]	FALSE	FALSE	FALSE	FALSE
##	[56,]	FALSE	FALSE	FALSE	FALSE
##	[57,]	FALSE	FALSE	FALSE	FALSE
##	[58,]	FALSE	FALSE	FALSE	FALSE
##	[59,]	FALSE	FALSE	FALSE	FALSE
##	[60,]	FALSE	FALSE	FALSE	FALSE
##	[61,]	FALSE	FALSE	FALSE	FALSE
##	[62,]	FALSE	FALSE	FALSE	FALSE
##	[63,]	FALSE	FALSE	FALSE	FALSE
##	[64,]	FALSE	FALSE	FALSE	FALSE
##	[65,]	FALSE	FALSE	FALSE	FALSE
##	[66,]	FALSE	FALSE	FALSE	FALSE
##	[67,]	FALSE	FALSE	FALSE	FALSE
##	[68,]	FALSE	FALSE	FALSE	FALSE
##	[69,]	FALSE	FALSE	FALSE	FALSE
##	[70,]	FALSE	FALSE	FALSE	FALSE
##	[71,]	FALSE	FALSE	FALSE	FALSE
##	[72,]	FALSE	FALSE	FALSE	FALSE
##	[73,]	FALSE	FALSE	FALSE	FALSE
##	[74,]	FALSE	FALSE	FALSE	FALSE
##	[75,]	FALSE	FALSE	FALSE	FALSE
##	[76,]	FALSE	FALSE	FALSE	FALSE
##	[77,]	FALSE	FALSE	FALSE	FALSE
##	[78,]	FALSE	FALSE	FALSE	FALSE
##	[79,]	FALSE	FALSE	FALSE	FALSE
##	[80,]	FALSE	FALSE	FALSE	FALSE
##	[81,]	FALSE	FALSE	FALSE	FALSE
##	[82,]	FALSE	FALSE	FALSE	FALSE
##	[83,]	FALSE	FALSE	FALSE	FALSE
##	[84,]	FALSE	FALSE	FALSE	FALSE
##	[85,]	FALSE	FALSE	FALSE	FALSE
##	[86,]	FALSE	FALSE	FALSE	FALSE
##	[87,]	FALSE	FALSE	FALSE	FALSE
##	[88,]	FALSE	FALSE	FALSE	FALSE
##	[89,]	FALSE	FALSE	FALSE	FALSE
##	[90,]	FALSE	FALSE	FALSE	FALSE

##	[91,]	FALSE	FALSE	FALSE	FALSE
##	[92,]	FALSE	FALSE	FALSE	FALSE
##	[93,]	FALSE	FALSE	FALSE	FALSE
##	[94,]	FALSE	FALSE	FALSE	FALSE
##	[95,]	FALSE	FALSE	FALSE	FALSE
##	[96,]	FALSE	FALSE	FALSE	FALSE
##	[97,]	FALSE	FALSE	FALSE	FALSE
##	[98,]	FALSE	FALSE	FALSE	FALSE
##	[99,]	FALSE	FALSE	FALSE	FALSE
##	[100,]	FALSE	FALSE	FALSE	FALSE
##	[101,]	FALSE	FALSE	FALSE	FALSE
##	[102,]	FALSE	FALSE	FALSE	FALSE
##	[103,]	FALSE	FALSE	FALSE	FALSE
##	[104,]	FALSE	FALSE	FALSE	FALSE
##	[105,]	FALSE	FALSE	FALSE	FALSE
##	[106,]	FALSE	FALSE	FALSE	FALSE
##	[107,]	FALSE	FALSE	FALSE	FALSE
##	[108,]	FALSE	FALSE	FALSE	FALSE
##	[109,]	FALSE	FALSE	FALSE	FALSE
##	[110,]	FALSE	FALSE	FALSE	FALSE
##	[111,]	FALSE	FALSE	FALSE	FALSE
##	[112,]	FALSE	FALSE	FALSE	FALSE
##	[113,]	FALSE	FALSE	FALSE	FALSE
##	[114,]	FALSE	FALSE	FALSE	FALSE
##	[115,]	FALSE	FALSE	FALSE	FALSE
##	[116,]	FALSE	FALSE	FALSE	FALSE
##	[117,]	FALSE	FALSE	FALSE	FALSE
##	[118,]	FALSE	FALSE	FALSE	FALSE
##	[119,]	FALSE	FALSE	FALSE	FALSE
##	[120,]	FALSE	FALSE	FALSE	FALSE
##	[121,]	FALSE	FALSE	FALSE	FALSE
##	[122,]	FALSE	FALSE	FALSE	FALSE
##	[123,]	FALSE	FALSE	FALSE	FALSE
##	[124,]	FALSE	FALSE	FALSE	FALSE
##	[125,]	FALSE	FALSE	FALSE	FALSE
##	[126,]	FALSE	FALSE	FALSE	FALSE
##	[127,]	FALSE	FALSE	FALSE	FALSE
##	[128,]	FALSE	FALSE	FALSE	FALSE
##	[129,]	FALSE	FALSE	FALSE	FALSE
##	[130,]	FALSE	FALSE	FALSE	FALSE
##	[131,]	FALSE	FALSE	FALSE	FALSE
##	[132,]	FALSE	FALSE	FALSE	FALSE
##	[133,]	FALSE	FALSE	FALSE	FALSE
##	[134,]	FALSE	FALSE	FALSE	FALSE
##	[135,]	FALSE	FALSE	FALSE	FALSE
##	[136,]	FALSE	FALSE	FALSE	FALSE
##	icu_beds hospitals total_at_risk				
##	[1,]	TRUE	TRUE	FALSE	
##	[2,]	FALSE	FALSE	FALSE	
##	[3,]	FALSE	FALSE	FALSE	
##	[4,]	FALSE	FALSE	FALSE	
##	[5,]	FALSE	FALSE	FALSE	
##	[6,]	FALSE	FALSE	FALSE	
##	[7,]	FALSE	FALSE	FALSE	

##	[8,]	FALSE	FALSE	FALSE
##	[9,]	FALSE	FALSE	FALSE
##	[10,]	FALSE	FALSE	FALSE
##	[11,]	FALSE	FALSE	FALSE
##	[12,]	FALSE	FALSE	FALSE
##	[13,]	FALSE	FALSE	FALSE
##	[14,]	FALSE	FALSE	FALSE
##	[15,]	FALSE	FALSE	FALSE
##	[16,]	FALSE	FALSE	FALSE
##	[17,]	FALSE	FALSE	FALSE
##	[18,]	FALSE	FALSE	FALSE
##	[19,]	FALSE	FALSE	FALSE
##	[20,]	FALSE	FALSE	FALSE
##	[21,]	FALSE	FALSE	FALSE
##	[22,]	FALSE	FALSE	FALSE
##	[23,]	FALSE	FALSE	FALSE
##	[24,]	FALSE	FALSE	FALSE
##	[25,]	FALSE	FALSE	FALSE
##	[26,]	FALSE	FALSE	FALSE
##	[27,]	FALSE	FALSE	FALSE
##	[28,]	FALSE	FALSE	FALSE
##	[29,]	FALSE	FALSE	FALSE
##	[30,]	FALSE	FALSE	FALSE
##	[31,]	FALSE	FALSE	FALSE
##	[32,]	FALSE	FALSE	FALSE
##	[33,]	FALSE	FALSE	FALSE
##	[34,]	FALSE	FALSE	FALSE
##	[35,]	FALSE	FALSE	FALSE
##	[36,]	FALSE	FALSE	FALSE
##	[37,]	FALSE	FALSE	FALSE
##	[38,]	FALSE	FALSE	FALSE
##	[39,]	FALSE	FALSE	FALSE
##	[40,]	FALSE	FALSE	FALSE
##	[41,]	FALSE	FALSE	FALSE
##	[42,]	FALSE	FALSE	FALSE
##	[43,]	FALSE	FALSE	FALSE
##	[44,]	FALSE	FALSE	FALSE
##	[45,]	FALSE	FALSE	FALSE
##	[46,]	FALSE	FALSE	FALSE
##	[47,]	FALSE	FALSE	FALSE
##	[48,]	FALSE	FALSE	FALSE
##	[49,]	FALSE	FALSE	FALSE
##	[50,]	FALSE	FALSE	FALSE
##	[51,]	FALSE	FALSE	FALSE
##	[52,]	FALSE	FALSE	FALSE
##	[53,]	FALSE	FALSE	FALSE
##	[54,]	FALSE	FALSE	FALSE
##	[55,]	FALSE	FALSE	FALSE
##	[56,]	FALSE	FALSE	FALSE
##	[57,]	FALSE	FALSE	FALSE
##	[58,]	FALSE	FALSE	FALSE
##	[59,]	FALSE	FALSE	FALSE
##	[60,]	FALSE	FALSE	FALSE
##	[61,]	FALSE	FALSE	FALSE

##	[62,]	FALSE	FALSE	FALSE
##	[63,]	FALSE	FALSE	FALSE
##	[64,]	FALSE	FALSE	FALSE
##	[65,]	FALSE	FALSE	FALSE
##	[66,]	FALSE	FALSE	FALSE
##	[67,]	FALSE	FALSE	FALSE
##	[68,]	FALSE	FALSE	FALSE
##	[69,]	FALSE	FALSE	FALSE
##	[70,]	FALSE	FALSE	FALSE
##	[71,]	FALSE	FALSE	FALSE
##	[72,]	FALSE	FALSE	FALSE
##	[73,]	FALSE	FALSE	FALSE
##	[74,]	FALSE	FALSE	FALSE
##	[75,]	FALSE	FALSE	FALSE
##	[76,]	FALSE	FALSE	FALSE
##	[77,]	FALSE	FALSE	FALSE
##	[78,]	FALSE	FALSE	FALSE
##	[79,]	FALSE	FALSE	FALSE
##	[80,]	FALSE	FALSE	FALSE
##	[81,]	FALSE	FALSE	FALSE
##	[82,]	FALSE	FALSE	FALSE
##	[83,]	FALSE	FALSE	FALSE
##	[84,]	FALSE	FALSE	FALSE
##	[85,]	FALSE	FALSE	FALSE
##	[86,]	FALSE	FALSE	FALSE
##	[87,]	FALSE	FALSE	FALSE
##	[88,]	FALSE	FALSE	FALSE
##	[89,]	FALSE	FALSE	FALSE
##	[90,]	FALSE	FALSE	FALSE
##	[91,]	FALSE	FALSE	FALSE
##	[92,]	FALSE	FALSE	FALSE
##	[93,]	FALSE	FALSE	FALSE
##	[94,]	FALSE	FALSE	FALSE
##	[95,]	FALSE	FALSE	FALSE
##	[96,]	FALSE	FALSE	FALSE
##	[97,]	FALSE	FALSE	FALSE
##	[98,]	FALSE	FALSE	FALSE
##	[99,]	FALSE	FALSE	FALSE
##	[100,]	FALSE	FALSE	FALSE
##	[101,]	FALSE	FALSE	FALSE
##	[102,]	FALSE	FALSE	FALSE
##	[103,]	FALSE	FALSE	FALSE
##	[104,]	FALSE	FALSE	FALSE
##	[105,]	FALSE	FALSE	FALSE
##	[106,]	FALSE	FALSE	FALSE
##	[107,]	FALSE	FALSE	FALSE
##	[108,]	FALSE	FALSE	FALSE
##	[109,]	FALSE	FALSE	FALSE
##	[110,]	FALSE	FALSE	FALSE
##	[111,]	FALSE	FALSE	FALSE
##	[112,]	FALSE	FALSE	FALSE
##	[113,]	FALSE	FALSE	FALSE
##	[114,]	FALSE	FALSE	FALSE
##	[115,]	FALSE	FALSE	FALSE

```
## [116,] FALSE FALSE FALSE
## [117,] FALSE FALSE FALSE
## [118,] FALSE FALSE FALSE
## [119,] FALSE FALSE FALSE
## [120,] FALSE FALSE FALSE
## [121,] FALSE FALSE FALSE
## [122,] FALSE FALSE FALSE
## [123,] FALSE FALSE FALSE
## [124,] FALSE FALSE FALSE
## [125,] FALSE FALSE FALSE
## [126,] FALSE FALSE FALSE
## [127,] FALSE FALSE FALSE
## [128,] FALSE FALSE FALSE
## [129,] FALSE FALSE FALSE
## [130,] FALSE FALSE FALSE
## [131,] FALSE FALSE FALSE
## [132,] FALSE FALSE FALSE
## [133,] FALSE FALSE FALSE
## [134,] FALSE FALSE FALSE
## [135,] FALSE FALSE FALSE
## [136,] FALSE FALSE FALSE
```

```
# removing missing values from coviddata
```

```
Coviddata <- Coviddata[complete.cases(Coviddata), ]
```

```
# partial_records <- sampledata[complete.cases(sampledata)]
```

```
# For cleaning process, renaming the column names using dplyr
```

```
library(dplyr)
```

```
# modifying the coloumn names in coviddata and storing
```

```
colnames(Coviddata)<- c("MMSA","percent_at_risk","patient_per_ICU_bed","Patient_per_hospital", "no_icu"
```

```
head(Coviddata)
```

```
## # A tibble: 6 x 7
```

```
##   MMSA      percent_at_risk patient_per_ICU~ Patient_per_hos~ no_icu no_hospital
##   <chr>      <chr>          <dbl>          <dbl>  <dbl>      <dbl>
## 1 Manhatta~ 47.29%          4490.          8980.    8         4
## 2 Hilton H~ 62.72%          3904.          36439.   28         3
## 3 Kahului~ 59.13%          3861.          19303.   20         4
## 4 Spartanb~ 66.12%          3786.          85188.   45         2
## 5 Baton Ro~ 66.60%          3460.          39001.  124        11
## 6 Rockingh~ 57.72%          3365.          40381.   60         5
```

```
## # ... with 1 more variable: total_at_risk <dbl>
```

```
# extracting the top 10 rows to subset the row values
```

```
Coviddata %>% sample_n(10, replace=FALSE)
```

```
## # A tibble: 10 x 7
```

```
##   MMSA      percent_at_risk patient_per_ICU~ Patient_per_hos~ no_icu no_hospital
##   <chr>      <chr>          <dbl>          <dbl>  <dbl>      <dbl>
## 1 Fargo, ~ 51.26%          870.          48306.  111         2
## 2 Wichita~ 67.11%          3279.          19677.   24         4
## 3 Baton R~ 66.60%          3460.          39001.  124        11
## 4 Tuscalo~ 62.40%          1371.          40201.   88         3
## 5 Corpus ~ 61.89%          1868.          53710.  115         4
```

```
## 6 Sacrame~ 54.51%      1924.      65148.    508      15
## 7 San Ant~ 56.62%      1558.      58783.    679      18
## 8 Columbu~ 59.99%      1735.      52809.    548      18
## 9 Anchora~ 58.80%      2055.      44706.    87       4
## 10 Scottsb~ 68.20%      899.      19784.    22       1
## # ... with 1 more variable: total_at_risk <dbl>
```

```
#extracting the top 10 rows to subset the row values
cov <- Coviddata %>% sample_n(10, replace=FALSE)
print(cov)
```

```
## # A tibble: 10 x 7
##   MMSA      percent_at_risk patient_per_ICU~ Patient_per_hos~ no_icu no_hospital
##   <chr>      <chr>              <dbl>          <dbl>    <dbl>    <dbl>
## 1 Portlan~ 55.37%                2649.        63107.    405      17
## 2 Anchora~ 58.80%                2055.        44706.    87       4
## 3 Salisbu~ 68.32%                3292.        37312.    68       6
## 4 Sioux C~ 60.83%                2320.        19139.    33       4
## 5 Hartfor~ 57.26%                2150.        55043.   256     10
## 6 Hagerst~ 66.46%                2023.        68778.    68       2
## 7 Allento~ 60.18%                1667.        36367.   240     11
## 8 Montgom~ 59.69%                2329.        40299.   398     23
## 9 Minneap~ 50.99%                2474.        43764.   566     32
## 10 Richmon~ 60.29%                1572.        60855.   387     10
## # ... with 1 more variable: total_at_risk <dbl>
```

```
#extracting the column values and concatenating using c and storing in dfcol
dfcol<- Coviddata[,c(1,2,3,4)]
```

```
#inline code output
```

This dataframe has 135 rows and 7 columns. The names of the columns and a brief description of each are in the table below :

```
#showing description of each coloumn in a table format
```

```
library(knitr)
description <- c('The name of the metropolitan area, metropolitan area or metropolitan division available
description_table <- data.frame(names(Coviddata), description)
kable(description_table, "pipe", col.names = NULL)
```

MMSA	The name of the metropolitan area, metropolitan area or metropolitan division available in the CDC's BRFSS
percent_at_risk	The percent of individuals in that area that are at high risk of becoming seriously ill from COVID-19 as per CDC's BRFSS
patient_per_ICU	The number of high risk individuals per ICU bed in that area
Patient_per_hospital	The number of high risk individuals per hospital in that area
no_icu	The number of ICU beds in the area, based on the Kaiser Family Foundation's data
no_hospital	The number of hospitals in the area, based on the Kaiser Family Foundation's data
total_at_risk	The total number of high risk individuals in the area, per CDC's BRFSS

```
#Summary of Coloumns choosen are as follows:
```

```
#1. patient_per_ICU_bed"
#2 Patient_per_hospital
#3 no_icu
#4 no_hospital
```



```
#5 total_at_risk
```

```
total_summary <- summarise_each(Coviddata[,c("patient_per_ICU_bed", "Patient_per_hospital", "no_icu", "no_hospital")],
```

```
## Warning: `summarise_each()` was deprecated in dplyr 0.7.0.
```

```
## Please use `across()` instead.
```

```
## This warning is displayed once every 8 hours.
```

```
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
```

```
## Please use a list of either functions or lambdas:
```

```
##
```

```
## # Simple named list:
```

```
## list(mean = mean, median = median)
```

```
##
```

```
## # Auto named with `tibble::lst()`:
```

```
## tibble::lst(mean, median)
```

```
##
```

```
## # Using lambdas
```

```
## list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
```

```
## This warning is displayed once every 8 hours.
```

```
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```

```
# print summary
```

```
total_summary
```

```
## # A tibble: 1 x 5
```

```
##   patient_per_ICU_bed Patient_per_hospital no_icu no_hospital total_at_risk
```

```
##           <dbl>           <dbl> <dbl>           <dbl>           <dbl>
```

```
## 1             1947.             43787.   360.             13.8             665288.
```

```
““
```