

Untitled

Priyanka

1/4/2021

Reading data from website <https://www.worldometers.info/coronavirus/>

```
library(rvest)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
#reading url using read_html
corona_rul <- read_html("https://www.worldometers.info/coronavirus/")
```

```
#reading table from the website
corona_file <- html_table(corona_rul)
```

```
head(corona_file)
```

```
## [[1]]
## # A tibble: 238 x 22
##   'Country,Other' TotalCases NewCases TotalDeaths NewDeaths
##   <int> <chr>          <chr>      <chr>      <chr>      <int>
## 1 NA "North America" 39,072,000 "+3,632" 876,154      311
## 2 NA "Asia"          45,698,206 "+754"    591,963       2
## 3 NA "South America" 26,415,596 ""        718,977      NA
## 4 NA "Europe"         45,863,910 ""        1,044,220    NA
## 5 NA "Africa"         4,708,637 ""        125,938      NA
## 6 NA "Oceania"        65,407      "+3"      1,222        NA
## 7 NA ""              721        ""        15          NA
## 8 NA "World"         161,824,477 "+4,389" 3,358,489    313
```

```
## 9      1 "USA"          33,626,097 ""          598,540          NA
## 10     2 "India"        24,046,120 ""          262,350          NA
## # ... with 228 more rows, and 16 more variables: TotalRecovered <chr>,
## #   NewRecovered <chr>, ActiveCases <chr>, Serious,Critical <chr>,
## #   Tot\U00a0Cases/1M pop <chr>, Deaths/1M pop <chr>, TotalTests <chr>,
## #   Tests/1M pop <chr>, Population <chr>, Continent <chr>,
## #   1 Caseevery X ppl <chr>, 1 Deathevery X ppl <chr>, 1 Testevery X ppl <int>,
## #   New Cases/1M pop <dbl>, New Deaths/1M pop <dbl>, Active Cases/1M pop <chr>
##
## [[2]]
## # A tibble: 238 x 22
##   '# 'Country,Other' TotalCases NewCases TotalDeaths NewDeaths
##   <int> <chr>          <chr>      <chr>      <chr>      <chr>
## 1    NA "Asia"          45,697,452 "+432,363" 591,961    "+5,496"
## 2    NA "North America" 39,068,368 "+57,706" 875,843    "+1,262"
## 3    NA "South America" 26,415,596 "+145,128" 718,977    "+4,108"
## 4    NA "Europe"        45,863,910 "+98,297" 1,044,220 "+2,270"
## 5    NA "Africa"        4,708,637 "+9,439" 125,938    "+234"
## 6    NA "Oceania"       65,404      "+258"    1,222      "+4"
## 7    NA ""              721         ""         15         ""
## 8    NA "World"         161,820,088 "+743,191" 3,358,176 "+13,374"
## 9      1 "China"        90,808      "+9"       4,636      ""
## 10     2 "USA"          33,626,097 "+39,825" 598,540    "+762"
## # ... with 228 more rows, and 16 more variables: TotalRecovered <chr>,
## #   NewRecovered <chr>, ActiveCases <chr>, Serious,Critical <chr>,
## #   Tot\U00a0Cases/1M pop <chr>, Deaths/1M pop <chr>, TotalTests <chr>,
## #   Tests/1M pop <chr>, Population <chr>, Continent <chr>,
## #   1 Caseevery X ppl <chr>, 1 Deathevery X ppl <chr>, 1 Testevery X ppl <int>,
## #   New Cases/1M pop <chr>, New Deaths/1M pop <dbl>, Active Cases/1M pop <chr>
##
## [[3]]
## # A tibble: 238 x 22
##   '# 'Country,Other' TotalCases NewCases TotalDeaths NewDeaths
##   <int> <chr>          <chr>      <chr>      <chr>      <chr>
## 1    NA "Asia"          45,265,089 "+449,547" 586,465    "+5,727"
## 2    NA "North America" 39,010,662 "+53,122" 874,581    "+1,328"
## 3    NA "South America" 26,270,468 "+140,207" 714,869    "+4,016"
## 4    NA "Europe"        45,765,613 "+97,716" 1,041,950 "+2,498"
## 5    NA "Africa"        4,699,198 "+8,877" 125,704    "+304"
## 6    NA "Oceania"       65,146      "+789"    1,218      "+9"
## 7    NA ""              721         ""         15         ""
## 8    NA "World"         161,076,897 "+750,258" 3,344,802 "+13,882"
## 9      1 "China"        90,799      "+16"       4,636      ""
## 10     2 "USA"          33,586,272 "+35,827" 597,778    "+843"
## # ... with 228 more rows, and 16 more variables: TotalRecovered <chr>,
## #   NewRecovered <chr>, ActiveCases <chr>, Serious,Critical <chr>,
## #   Tot\U00a0Cases/1M pop <chr>, Deaths/1M pop <chr>, TotalTests <chr>,
## #   Tests/1M pop <chr>, Population <chr>, Continent <chr>,
## #   1 Caseevery X ppl <chr>, 1 Deathevery X ppl <chr>, 1 Testevery X ppl <int>,
## #   New Cases/1M pop <chr>, New Deaths/1M pop <dbl>, Active Cases/1M pop <chr>
```

```
#writing the data to csv file
write.table(corona_file, file = "corona_file.csv",
            sep = ",",
```

```
row.names = FALSE)
```

```
corona_file_new <- corona_file[[1]]
head(corona_file_new)
```

```
## # A tibble: 6 x 22
##   '# 'Country,Other' TotalCases NewCases TotalDeaths NewDeaths TotalRecovered
##   <int> <chr>          <chr>      <chr>          <int> <chr>
## 1    NA North America  39,072,000 "+3,632"  876,154          311 31,184,603
## 2    NA Asia          45,698,206 "+754"    591,963           2 39,798,147
## 3    NA South America 26,415,596 ""      718,977          NA 23,962,329
## 4    NA Europe        45,863,910 ""      1,044,220         NA 41,405,659
## 5    NA Africa        4,708,637 ""      125,938          NA 4,243,248
## 6    NA Oceania       65,407    "+3"      1,222           NA 61,315
## # ... with 15 more variables: NewRecovered <chr>, ActiveCases <chr>,
## #   Serious,Critical <chr>, Tot\U00a0Cases/1M pop <chr>, Deaths/1M pop <chr>,
## #   TotalTests <chr>, Tests/1M pop <chr>, Population <chr>, Continent <chr>,
## #   1 Caseevery X ppl <chr>, 1 Deathevery X ppl <chr>, 1 Testevery X ppl <int>,
## #   New Cases/1M pop <dbl>, New Deaths/1M pop <dbl>, Active Cases/1M pop <chr>
```

```
str(corona_file_new)
```

```
## tibble [238 x 22] (S3: tbl_df/tbl/data.frame)
##  $ # : int [1:238] NA NA NA NA NA NA NA NA NA 1 2 ...
##  $ Country,Other : chr [1:238] "North America" "Asia" "South America" "Europe" ...
##  $ TotalCases : chr [1:238] "39,072,000" "45,698,206" "26,415,596" "45,863,910" ...
##  $ NewCases : chr [1:238] "+3,632" "+754" "" "" ...
##  $ TotalDeaths : chr [1:238] "876,154" "591,963" "718,977" "1,044,220" ...
##  $ NewDeaths : int [1:238] 311 2 NA NA NA NA NA NA 313 NA NA ...
##  $ TotalRecovered : chr [1:238] "31,184,603" "39,798,147" "23,962,329" "41,405,659" ...
##  $ NewRecovered : chr [1:238] "+2,247" "+502" "" "" ...
##  $ ActiveCases : chr [1:238] "7,011,243" "5,308,096" "1,734,290" "3,414,031" ...
##  $ Serious,Critical : chr [1:238] "16,430" "32,313" "28,211" "24,188" ...
##  $ Tot Cases/1M pop : chr [1:238] "" "" "" "" ...
##  $ Deaths/1M pop : chr [1:238] "" "" "" "" ...
##  $ TotalTests : chr [1:238] "" "" "" "" ...
##  $ Tests/1M pop : chr [1:238] "" "" "" "" ...
##  $ Population : chr [1:238] "" "" "" "" ...
##  $ Continent : chr [1:238] "North America" "Asia" "South America" "Europe" ...
##  $ 1 Caseevery X ppl : chr [1:238] "" "" "" "" ...
##  $ 1 Deathevery X ppl : chr [1:238] "" "" "" "" ...
##  $ 1 Testevery X ppl : int [1:238] NA NA NA NA NA NA NA NA 1 4 ...
##  $ New Cases/1M pop : num [1:238] NA NA NA NA NA NA NA NA NA NA ...
##  $ New Deaths/1M pop : num [1:238] NA NA NA NA NA NA NA NA NA NA ...
##  $ Active Cases/1M pop: chr [1:238] "" "" "" "" ...
```

```
#writing the data to csv file
write.table(corona_file_new, file = "corona_file.csv",
            sep = ",",
            row.names = FALSE)
```

```
corona_file_new <- read.csv("corona_file.csv")
corona_data <- data.frame(corona_file_new[-1:-8,])
head(corona_data)
```

```
##      X. Country.Other TotalCases NewCases TotalDeaths NewDeaths TotalRecovered
## 9      1          USA 33,626,097          598,540          NA      26,667,199
## 10     2          India 24,046,120          262,350          NA      20,073,367
## 11     3          Brazil 15,436,827          430,596          NA      13,979,329
## 12     4          France 5,841,129          107,250          NA      4,999,079
## 13     5          Turkey 5,083,996           44,059          NA      4,856,763
## 14     6          Russia 4,913,439          114,723          NA      4,527,878
##      NewRecovered ActiveCases Serious.Critical Tot.Cases.1M.pop Deaths.1M.pop
## 9              6,360,358           8,611          101,076           1,799
## 10             3,710,403           8,944           17,278            189
## 11             1,026,902           8,318           72,180           2,013
## 12              734,800           4,442           89,316           1,640
## 13              183,174           2,765           59,726            518
## 14              270,838           2,300           33,656            786
##      TotalTests Tests.1M.pop Population Continent X1.Caseevery.X.ppl
## 9 462,795,300   1,391,111 332,680,263 North America           10
## 10 309,448,585    222,350 1,391,716,282          Asia            58
## 11 46,970,130    219,625 213,864,852 South America            14
## 12 80,214,807   1,226,555  65,398,484          Europe            11
## 13 50,259,943    590,444  85,122,228          Asia             17
## 14 132,400,000    906,920 145,988,633          Europe             30
##      X1.Deathevery.X.ppl X1.Testevery.X.ppl New.Cases.1M.pop New.Deaths.1M.pop
## 9              556              1              NA              NA
## 10             5,305              4              NA              NA
## 11             497              5              NA              NA
## 12             610              1              NA              NA
## 13            1,932              2              NA              NA
## 14            1,273              1              NA              NA
##      Active.Cases.1M.pop
## 9             19,119
## 10             2,666
## 11             4,802
## 12            11,236
## 13             2,152
## 14             1,855
```

```
#head(corona_data)
```

```
corona_data <- data.frame(corona_file_new[ , -1 ])
corona_data <- data.frame(corona_file_new[c(-1:-8,-228 :-236), ])
```

```
#head(corona_data)
```

```
#chaning colnmaes by using rename
```

```
corona_data_updated <- corona_data %>%
  rename(S.No. ="X." , Country_Other = "Country.Other",
         Serious_Critical = "Serious.Critical",
```

```

Tot_Cases_1M_pop = "Tot.Cases.1M.pop",
Deaths_1M_pop = "Deaths.1M.pop",
Tests_1M_pop = "Tests.1M.pop" ,
X1_Caseevery_X_ppl = "X1.Caseevery.X.ppl",
X1_Deathevery_X_ppl = "X1.Deathevery.X.ppl",
X1_Testevery_X_ppl = "X1.Testevery.X.ppl", )

head(corona_data_updated)

```

```

##      S.No. Country_Other TotalCases NewCases TotalDeaths NewDeaths TotalRecovered
## 9      1      USA 33,626,097          598,540          NA      26,667,199
## 10     2      India 24,046,120          262,350          NA      20,073,367
## 11     3      Brazil 15,436,827          430,596          NA      13,979,329
## 12     4      France 5,841,129          107,250          NA      4,999,079
## 13     5      Turkey 5,083,996           44,059          NA      4,856,763
## 14     6      Russia 4,913,439          114,723          NA      4,527,878
##      NewRecovered ActiveCases Serious_Critical Tot_Cases_1M_pop Deaths_1M_pop
## 9              6,360,358           8,611          101,076           1,799
## 10             3,710,403           8,944           17,278            189
## 11             1,026,902           8,318           72,180           2,013
## 12              734,800           4,442           89,316           1,640
## 13              183,174           2,765           59,726            518
## 14              270,838           2,300           33,656            786
##      TotalTests Tests_1M_pop Population Continent X1_Caseevery_X_ppl
## 9 462,795,300 1,391,111 332,680,263 North America           10
## 10 309,448,585 222,350 1,391,716,282 Asia                58
## 11 46,970,130 219,625 213,864,852 South America           14
## 12 80,214,807 1,226,555 65,398,484 Europe                11
## 13 50,259,943 590,444 85,122,228 Asia                  17
## 14 132,400,000 906,920 145,988,633 Europe                30
##      X1_Deathevery_X_ppl X1_Testevery_X_ppl New.Cases.1M.pop New.Deaths.1M.pop
## 9              556           1          NA          NA
## 10             5,305           4          NA          NA
## 11             497           5          NA          NA
## 12             610           1          NA          NA
## 13            1,932           2          NA          NA
## 14            1,273           1          NA          NA
##      Active.Cases.1M.pop
## 9             19,119
## 10             2,666
## 11             4,802
## 12            11,236
## 13             2,152
## 14             1,855

```

```

#head(corona_data_updated)

#corona_data %>% select(-NewCases,-NewDeaths,-NewRecovered )
#head(corona_data_updated)

```

```
#2nd method starts_with() to remove columns
```

```
corona_data_updated <- corona_data_updated %>% dplyr:: select(-starts_with("New"))
```

```
head(corona_data_updated)
```

```
##      S.No. Country_Other TotalCases TotalDeaths TotalRecovered ActiveCases
## 9         1          USA 33,626,097      598,540      26,667,199 6,360,358
## 10        2          India 24,046,120      262,350      20,073,367 3,710,403
## 11        3          Brazil 15,436,827      430,596      13,979,329 1,026,902
## 12        4          France 5,841,129      107,250      4,999,079 734,800
## 13        5          Turkey 5,083,996       44,059      4,856,763 183,174
## 14        6          Russia 4,913,439      114,723      4,527,878 270,838
##      Serious_Critical Tot_Cases_1M_pop Deaths_1M_pop TotalTests Tests_1M_pop
## 9              8,611          101,076          1,799 462,795,300 1,391,111
## 10             8,944           17,278           189 309,448,585 222,350
## 11             8,318           72,180          2,013 46,970,130 219,625
## 12             4,442           89,316          1,640 80,214,807 1,226,555
## 13             2,765           59,726           518 50,259,943 590,444
## 14             2,300           33,656           786 132,400,000 906,920
##      Population      Continent X1_Caseevery_X_ppl X1_Deathevery_X_ppl
## 9    332,680,263 North America          10          556
## 10 1,391,716,282          Asia           58          5,305
## 11 213,864,852 South America          14          497
## 12   65,398,484          Europe          11          610
## 13   85,122,228          Asia           17          1,932
## 14 145,988,633          Europe          30          1,273
##      X1_Testevery_X_ppl Active.Cases.1M.pop
## 9              1          19,119
## 10             4           2,666
## 11             5           4,802
## 12             1          11,236
## 13             2           2,152
## 14             1           1,855
```

```
#head(corona_data_updated)
```

```
library(dplyr)
```

```
#removing commas in the data
```

```
set.seed(1)
```

```
mysub <- function(x) {
  gsub(",", "", x)
}
```

```
#APPLYING mysub function to all applicable columns in the dataset
```

```
corona_data_updated[,3:12:15] <- apply(corona_data_updated[,3:12:15], MARGIN=2, FUN= mysub )
```

```
## Warning in 3:12:15: numerical expression has 10 elements: only the first used
```

```
## Warning in 3:12:15: numerical expression has 10 elements: only the first used
```

```
head(corona_data_updated)
```

```
##      S.No. Country_Other TotalCases TotalDeaths TotalRecovered ActiveCases
## 9      1      USA      33626097      598540      26667199      6360358
## 10     2      India      24046120      262350      20073367      3710403
## 11     3      Brazil      15436827      430596      13979329      1026902
## 12     4      France      5841129      107250      4999079      734800
## 13     5      Turkey      5083996      44059      4856763      183174
## 14     6      Russia      4913439      114723      4527878      270838
##      Serious_Critical Tot_Cases_1M_pop Deaths_1M_pop TotalTests Tests_1M_pop
## 9      8611      101076      1799      462795300      1391111
## 10     8944      17278      189      309448585      222350
## 11     8318      72180      2013      46970130      219625
## 12     4442      89316      1640      80214807      1226555
## 13     2765      59726      518      50259943      590444
## 14     2300      33656      786      132400000      906920
##      Population      Continent X1_Caseevery_X_ppl X1_Deathevery_X_ppl
## 9      332680263 North America      10      556
## 10 1391716282      Asia      58      5305
## 11 213864852 South America      14      497
## 12 65398484      Europe      11      610
## 13 85122228      Asia      17      1932
## 14 145988633      Europe      30      1273
##      X1_Testevery_X_ppl Active.Cases.1M.pop
## 9      1      19,119
## 10     4      2,666
## 11     5      4,802
## 12     1      11,236
## 13     2      2,152
## 14     1      1,855
```

```
str(corona_data_updated)
```

```
## 'data.frame':    221 obs. of  17 variables:
## $ S.No.      : int  1 2 3 4 5 6 7 8 9 10 ...
## $ Country_Other : chr  "USA" "India" "Brazil" "France" ...
## $ TotalCases   : chr  "33626097" "24046120" "15436827" "5841129" ...
## $ TotalDeaths  : chr  "598540" "262350" "430596" "107250" ...
## $ TotalRecovered : chr  "26667199" "20073367" "13979329" "4999079" ...
## $ ActiveCases  : chr  "6360358" "3710403" "1026902" "734800" ...
## $ Serious_Critical : chr  "8611" "8944" "8318" "4442" ...
## $ Tot_Cases_1M_pop : chr  "101076" "17278" "72180" "89316" ...
## $ Deaths_1M_pop : chr  "1799" "189" "2013" "1640" ...
## $ TotalTests    : chr  "462795300" "309448585" "46970130" "80214807" ...
## $ Tests_1M_pop   : chr  "1391111" "222350" "219625" "1226555" ...
## $ Population     : chr  "332680263" "1391716282" "213864852" "65398484" ...
## $ Continent      : chr  "North America" "Asia" "South America" "Europe" ...
## $ X1_Caseevery_X_ppl : chr  "10" "58" "14" "11" ...
## $ X1_Deathevery_X_ppl: chr  "556" "5305" "497" "610" ...
## $ X1_Testevery_X_ppl : int  1 4 5 1 2 1 0 1 1 1 ...
## $ Active.Cases.1M.pop: chr  "19,119" "2,666" "4,802" "11,236" ...
```

```
#converting chr type to num using lapply
```

```
colms <- c(3:12)
```

```
corona_data_updated[colms] <- lapply(corona_data_updated[colms], as.numeric)
```

```
corona_data_updated[14:16] <- lapply(corona_data_updated[14:16], as.numeric)
```

```
str(corona_data_updated)
```

```
## 'data.frame':    221 obs. of  17 variables:
## $ S.No.          : int  1 2 3 4 5 6 7 8 9 10 ...
## $ Country_Other  : chr   "USA" "India" "Brazil" "France" ...
## $ TotalCases     : num  33626097 24046120 15436827 5841129 5083996 ...
## $ TotalDeaths    : num  598540 262350 430596 107250 44059 ...
## $ TotalRecovered : num  26667199 20073367 13979329 4999079 4856763 ...
## $ ActiveCases    : num  6360358 3710403 1026902 734800 183174 ...
## $ Serious_Critical : num  8611 8944 8318 4442 2765 ...
## $ Tot_Cases_1M_pop : num  101076 17278 72180 89316 59726 ...
## $ Deaths_1M_pop   : num  1799 189 2013 1640 518 ...
## $ TotalTests      : num  4.63e+08 3.09e+08 4.70e+07 8.02e+07 5.03e+07 ...
## $ Tests_1M_pop    : num  1391111 222350 219625 1226555 590444 ...
## $ Population      : num  3.33e+08 1.39e+09 2.14e+08 6.54e+07 8.51e+07 ...
## $ Continent       : chr   "North America" "Asia" "South America" "Europe" ...
## $ X1_Caseevery_X_ppl : num  10 58 14 11 17 30 15 15 13 23 ...
## $ X1_Deathevery_X_ppl: num  556 5305 497 610 1932 ...
## $ X1_Testevery_X_ppl : num  1 4 5 1 2 1 0 1 1 1 ...
## $ Active.Cases.1M.pop: chr   "19,119" "2,666" "4,802" "11,236" ...
```

```
summary(corona_data_updated)
```

```
##      S.No.      Country_Other      TotalCases      TotalDeaths
## Min.   : 1.0   Length:221      Min.   : 3      Min.   : 1
## 1st Qu.: 55.5   Class :character 1st Qu.: 4895 1st Qu.: 108
## Median :110.0   Mode  :character  Median : 42578 Median : 783
## Mean   :110.0                      Mean   : 1464067 Mean   : 32117
## 3rd Qu.:164.5                      3rd Qu.: 324868 3rd Qu.: 6379
## Max.   :219.0                      Max.   :161824477 Max.   :3358489
## NA's   :2                          NA's   :12
## TotalRecovered      ActiveCases      Serious_Critical      Tot_Cases_1M_pop
## Min.   : 2      Min.   : 0      Min.   : 0.0      Min.   : 8
## 1st Qu.: 3908    1st Qu.: 148    1st Qu.: 12.0    1st Qu.: 2059
## Median : 37281    Median : 3297    Median : 83.0    Median : 17497
## Mean   : 1272520    Mean   : 161175    Mean   : 1389.7    Mean   : 32803
## 3rd Qu.: 288891    3rd Qu.: 26408    3rd Qu.: 481.8    3rd Qu.: 60083
## Max.   :140656007    Max.   :17809981    Max.   :104224.0    Max.   :174094
## NA's   :2                          NA's   :71      NA's   :3
## Deaths_1M_pop      TotalTests      Tests_1M_pop      Population
## Min.   : 0.10      Min.   : 1313      Min.   : 1146      Min.   :8.030e+02
## 1st Qu.: 37.75      1st Qu.: 133695    1st Qu.: 58380      1st Qu.:6.570e+05
## Median : 263.50      Median : 946627    Median : 262863      Median :6.617e+06
## Mean   : 573.15      Mean   : 10244503    Mean   : 661450      Mean   :2.942e+07
## 3rd Qu.: 916.75      3rd Qu.: 4729538    3rd Qu.: 767238      3rd Qu.:2.385e+07
## Max.   :3005.00      Max.   :462795300    Max.   :7774839      Max.   :1.392e+09
## NA's   :15          NA's   :12          NA's   :12          NA's   :4
## Continent          X1_Caseevery_X_ppl X1_Deathevery_X_ppl X1_Testevery_X_ppl
```



```
## Length:221      Min.   :    6      Min.   :   333      Min.   :  0.00
## Class :character 1st Qu.:   17      1st Qu.:  1090      1st Qu.:  1.00
## Mode  :character Median :   58      Median :   3936      Median :  4.00
##                Mean  : 2389      Mean  : 114275      Mean  : 27.43
##                3rd Qu.: 497      3rd Qu.: 27104      3rd Qu.: 17.00
##                Max.   :120222      Max.   :7366551      Max.   :872.00
##                NA's   :4          NA's   :16          NA's   :12
## Active.Cases.1M.pop
## Length:221
## Class :character
## Mode  :character
##
##
##
##
```

```
head(corona_data_updated)
```

```
##      S.No. Country_Other TotalCases TotalDeaths TotalRecovered ActiveCases
## 9         1          USA    33626097     598540      26667199      6360358
## 10        2          India   24046120     262350      20073367      3710403
## 11        3          Brazil  15436827     430596      13979329      1026902
## 12        4          France   5841129     107250       4999079       734800
## 13        5          Turkey   5083996      44059       4856763       183174
## 14        6          Russia   4913439     114723       4527878       270838
##      Serious_Critical Tot_Cases_1M_pop Deaths_1M_pop TotalTests Tests_1M_pop
## 9                8611         101076         1799  462795300      1391111
## 10               8944         17278          189  309448585       222350
## 11               8318         72180         2013  46970130       219625
## 12               4442         89316         1640  80214807      1226555
## 13               2765         59726          518  50259943       590444
## 14               2300         33656          786 132400000       906920
##      Population      Continent X1_Caseevery_X_ppl X1_Deathevery_X_ppl
## 9    332680263 North America          10          556
## 10 1391716282      Asia          58          5305
## 11 213864852 South America          14          497
## 12 65398484      Europe          11          610
## 13 85122228      Asia          17          1932
## 14 145988633      Europe          30          1273
##      X1_Testevery_X_ppl Active.Cases.1M.pop
## 9                1      19,119
## 10               4      2,666
## 11               5      4,802
## 12               1     11,236
## 13               2      2,152
## 14               1      1,855
```

```
write.table(corona_data_updated, file = "corona_data_updated.csv",
            sep = ",",
            row.names = FALSE)
```

```
library(tidyverse)
```

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

```
## v ggplot2 3.3.3      v purrr 0.3.4
## v tibble 3.1.1       v stringr 1.4.0
## v tidyr 1.1.3        v forcats 0.5.1
## v readr 1.4.0
```

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

```
## x dplyr::filter()      masks stats::filter()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()         masks stats::lag()
```

```
corona_data_updated <- na.omit(corona_data_updated)
summary(corona_data_updated)
```

```
##      S.No.      Country_Other      TotalCases      TotalDeaths
## Min.   : 1.00   Length:147      Min.    :    445   Min.    :    7
## 1st Qu.: 39.00   Class :character  1st Qu.:  22038   1st Qu.:   232
## Median : 85.00   Mode  :character  Median : 130380   Median :  1893
## Mean   : 86.06                      Mean  : 1066119   Mean   : 22165
## 3rd Qu.:126.50                      3rd Qu.: 584930   3rd Qu.: 11480
## Max.   :199.00                      Max.   :33626097   Max.   :598540
## TotalRecovered      ActiveCases      Serious_Critical Tot_Cases_1M_pop
## Min.    :    44   Min.    :    6   Min.    : 1.0   Min.    :   213
## 1st Qu.: 19444   1st Qu.:   989   1st Qu.: 13.0   1st Qu.: 10290
## Median : 114981   Median :   9157   Median : 85.0   Median : 31295
## Mean   : 925462   Mean   : 118492   Mean   : 709.0   Mean   : 41404
## 3rd Qu.: 538059   3rd Qu.: 42092   3rd Qu.: 481.5   3rd Qu.: 69347
## Max.   :26667199   Max.   :6360358   Max.   :8944.0   Max.   :174094
## Deaths_1M_pop      TotalTests      Tests_1M_pop      Population
## Min.    :    5.0   Min.    : 16700   Min.    : 1146   Min.    :1.107e+04
## 1st Qu.: 85.5     1st Qu.: 242594   1st Qu.: 104750   1st Qu.:1.938e+06
## Median : 430.0     Median : 1518411   Median : 342992   Median :8.709e+06
## Mean   : 705.9     Mean   : 13876857   Mean   : 668093   Mean   :3.665e+07
## 3rd Qu.:1088.0     3rd Qu.: 9039837   3rd Qu.: 833788   3rd Qu.:3.305e+07
## Max.   :3005.0     Max.   :462795300   Max.   :7774839   Max.   :1.392e+09
## Continent          X1_Caseevery_X_ppl X1_Deathevery_X_ppl X1_Testevery_X_ppl
## Length:147         Min.    :    6.0   Min.    :   333   Min.    :    0.00
## Class :character    1st Qu.:   14.5   1st Qu.:   919   1st Qu.:    1.00
## Mode  :character    Median :   32.0   Median :  2325   Median :    3.00
##                      Mean   :  224.1   Mean   : 13279   Mean   :   20.03
##                      3rd Qu.:   97.5   3rd Qu.: 11724   3rd Qu.:    9.50
##                      Max.   : 4686.0   Max.   :190003   Max.   :   872.00
## Active.Cases.1M.pop
## Length:147
## Class :character
## Mode  :character
##
##
##
```

```
#head(corona_data_updated)
```

```
corona_data_updated$Continent <- as.factor(corona_data_updated$Continent)
str(corona_data_updated$Continent)
```

```
## Factor w/ 6 levels "Africa","Asia",...: 5 2 6 4 2 4 4 4 4 4 ...
```

```
data1<-corona_data_updated
df <-corona_data_updated
```

```
library(forcats)
library(tidyverse)
library(ggplot2)
#Which continent is having the maximum number of cases?
```

```
continent_TotalCases <- corona_data_updated %>% mutate(Continent = fct_lump(Continent, n=5 )) %>%
  group_by(Continent) %>%
  summarise(TotalCases = max(TotalCases, na.rm=TRUE))%>% arrange(desc(TotalCases))
continent_TotalCases
```

```
## # A tibble: 6 x 2
##   Continent      TotalCases
##   <fct>          <dbl>
## 1 North America 33626097
## 2 Asia          24046120
## 3 South America 15436827
## 4 Europe        5841129
## 5 Africa        1605252
## 6 Australia/Oceania 29957
```

```
# Which continent has the least number of deaths?
```

```
library(dplyr)
continent_leastdeaths <- corona_data_updated %>% mutate(Continent = fct_lump(Continent, n=5 )) %>%
  group_by(Continent) %>% dplyr::summarise(TotalDeaths = min(TotalDeaths, na.rm=TRUE)) %>% arrange(
continent_leastdeaths
```

```
## # A tibble: 6 x 2
##   Continent      TotalDeaths
##   <fct>          <dbl>
## 1 Australia/Oceania 7
## 2 North America    12
## 3 Europe           29
## 4 Asia             31
## 5 Africa            67
## 6 South America    108
```

```
# What is the current status of China?
```

```
#There no data for China as it removed in the process of removing NAs
#I have taken 3 different countries for this question
```

```

library(dplyr)
# current status of Brazil
current_status_Brazil <- corona_data_updated %>%
  filter(corona_data_updated$Country_Other == "Brazil") %>%
  dplyr::summarise(Country_Other, TotalCases=max(TotalCases, na.rm=TRUE), TotalDeaths=max(TotalDeaths, na.rm=TRUE))
current_status_Brazil

##   Country_Other TotalCases TotalDeaths ActiveCases Serious_Critical
## 1      Brazil    15436827      430596    1026902             8318

# current status of USA
current_status_USA <- corona_data_updated %>%
  filter(corona_data_updated$Country_Other == "USA") %>%
  dplyr::summarise(Country_Other, TotalCases=max(TotalCases, na.rm=TRUE), TotalDeaths=max(TotalDeaths, na.rm=TRUE))
current_status_USA

##   Country_Other TotalCases TotalDeaths ActiveCases Serious_Critical
## 1      USA    33626097      598540    6360358             8611

# current status of India
current_status_India <- corona_data_updated %>%
  filter(corona_data_updated$Country_Other == "India") %>%
  dplyr::summarise(Country_Other, TotalCases=max(TotalCases, na.rm=TRUE), TotalDeaths=max(TotalDeaths, na.rm=TRUE))
current_status_India

##   Country_Other TotalCases TotalDeaths ActiveCases Serious_Critical
## 1      India    24046120      262350    3710403             8944

##### What is the current status of China?
##### Method-2 using TotalDeaths
#There no data for China as it removed in the process of removing NAs
#I have taken 3 different countries for this question
#If TotalDeaths is more than the mean of the value the rank allotted is "1" which indicate its in danger

library(dplyr)
TotalDeaths_status <- corona_data_updated %>% dplyr::summarise(TotalDeaths = mean(TotalDeaths, na.rm=TRUE))
#TotalDeaths_status
current_status <- mutate(corona_data_updated, rank = ifelse(corona_data_updated$TotalDeaths >= 13235, "1", "2"))
current_status$rank <- as.factor(current_status$rank)

current_status_country <- current_status %>% filter(current_status$rank== 1)%>% select(Country_Other, TotalDeaths)

#current_status_country
final_satus <- function(x){
  ifelse(x %in% current_status_country$Country_Other, "The country is in **Dangerous situation**", "The country is not in dangerous situation")
}
#status of different countries
final_satus("USA")

## [1] "The country is in **Dangerous situation**"

```

```
final_satus("Ireland")
```

```
## [1] "The country is Safe"
```

```
final_satus("Germany")
```

```
## [1] "The country is in **Dangerous situation**"
```

```
final_satus("India")
```

```
## [1] "The country is in **Dangerous situation**"
```

```
final_satus("Egypt")
```

```
## [1] "The country is in **Dangerous situation**"
```

```
# Please arrange all data based on the total number of cases per million population?
```

```
cases_per_million_population <- corona_data_updated %>%  
  mutate(corona_data_updated$TotalCases / (corona_data_updated$Population / 1000000 )) %>%  
  arrange(desc(TotalCases))
```

```
head(cases_per_million_population)
```

```
##   S.No. Country_Other TotalCases TotalDeaths TotalRecovered ActiveCases  
## 1      1      USA      33626097      598540      26667199      6360358  
## 2      2      India      24046120      262350      20073367      3710403  
## 3      3      Brazil      15436827      430596      13979329      1026902  
## 4      4      France      5841129      107250      4999079      734800  
## 5      5      Turkey      5083996      44059      4856763      183174  
## 6      6      Russia      4913439      114723      4527878      270838  
##   Serious_Critical Tot_Cases_1M_pop Deaths_1M_pop TotalTests Tests_1M_pop  
## 1              8611          101076          1799  462795300      1391111  
## 2              8944          17278           189  309448585      222350  
## 3              8318          72180           2013  46970130      219625  
## 4              4442          89316           1640  80214807      1226555  
## 5              2765          59726           518  50259943      590444  
## 6              2300          33656           786  132400000      906920  
##   Population      Continent X1_Caseevery_X_pp1 X1_Deathevery_X_pp1  
## 1  332680263 North America          10          556  
## 2 1391716282          Asia           58          5305  
## 3 213864852 South America          14          497  
## 4  65398484          Europe          11          610  
## 5  85122228          Asia           17          1932  
## 6 145988633          Europe          30          1273  
##   X1_Testevery_X_pp1 Active.Cases.1M.pop  
## 1              1          19,119  
## 2              4           2,666  
## 3              5           4,802  
## 4              1          11,236
```

```
## 5          2          2,152
## 6          1          1,855
##   corona_data_updated$TotalCases/(corona_data_updated$Population/1e+06)
## 1                                     101076.32
## 2                                     17278.03
## 3                                     72180.29
## 4                                     89315.97
## 5                                     59725.83
## 6                                     33656.31
```

```
#head(cases_per_million_population)
```

```
#Which country ranks first based on total number of cases per million population and which country rank
```

```
#country ranks first based on total number of cases per million population
cases_per_million_population <- corona_data_updated %>%
  mutate(corona_data_updated$TotalCases / (corona_data_updated$Population / 1000000 )) %>%
  select(Country_Other, TotalCases, Population)%>% arrange(corona_data_updated)

cases_per_rankONE <- cases_per_million_population %>%
  filter(TotalCases == max(TotalCases) ) %>%
  arrange(Country_Other) %>%
  head(10)
cases_per_rankONE
```

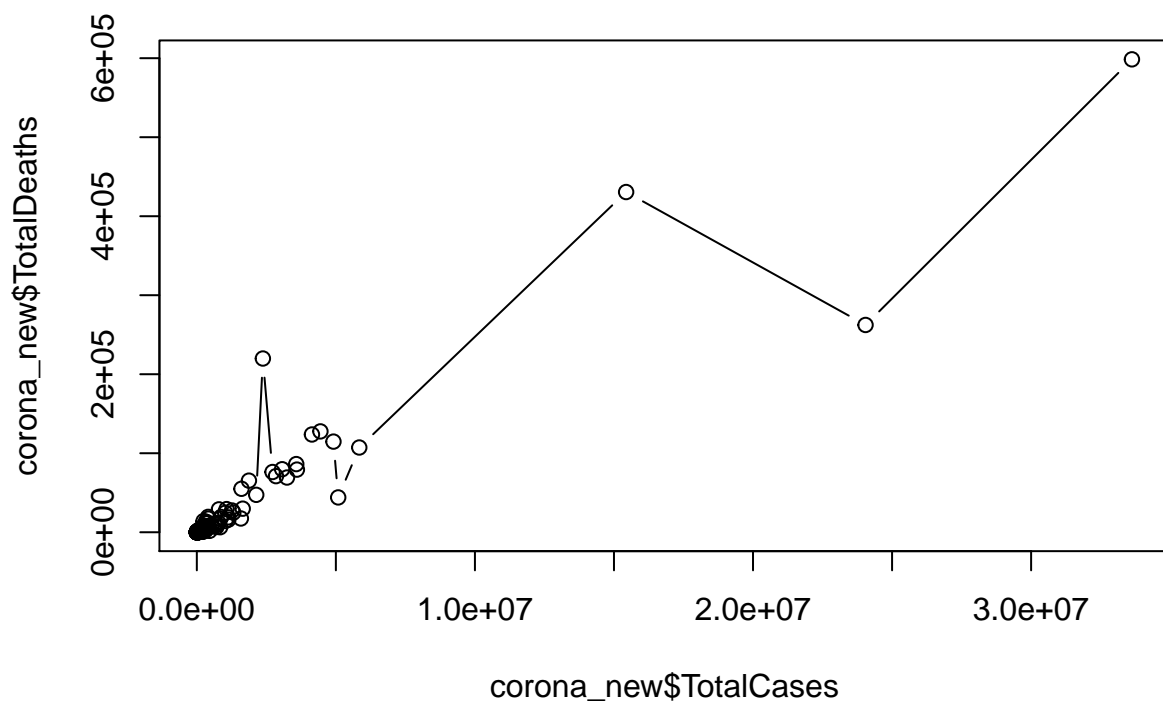
```
##   Country_Other TotalCases Population
## 1          USA    33626097   332680263
```

```
#country ranks last based on total number of cases per million population
cases_per_rankLAST <- cases_per_million_population %>%
  filter(TotalCases ==min(TotalCases) ) %>%
  arrange(Country_Other) %>%
  head(10)
cases_per_rankLAST
```

```
##           Country_Other TotalCases Population
## 1 Wallis and Futuna         445      11068
```

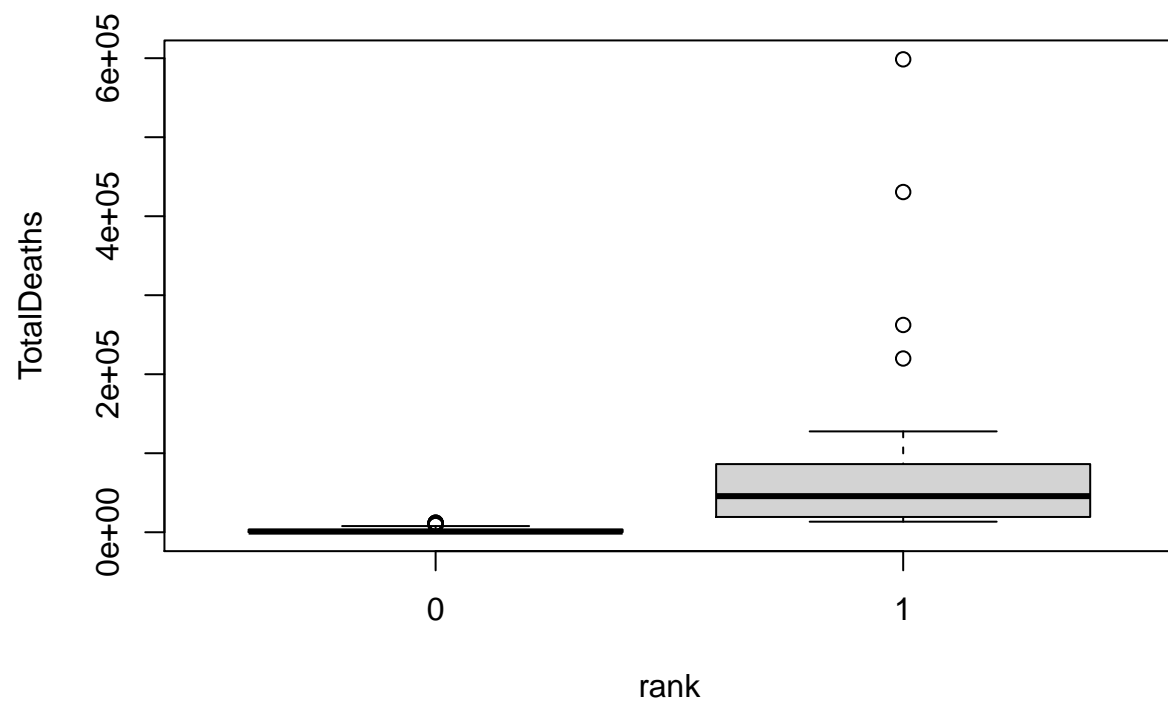
```
corona_new <- current_status %>% select(TotalCases,TotalDeaths,TotalRecovered,rank )
#head(corona_new)

plot(corona_new$TotalCases,corona_new$TotalDeaths , type="b")
```



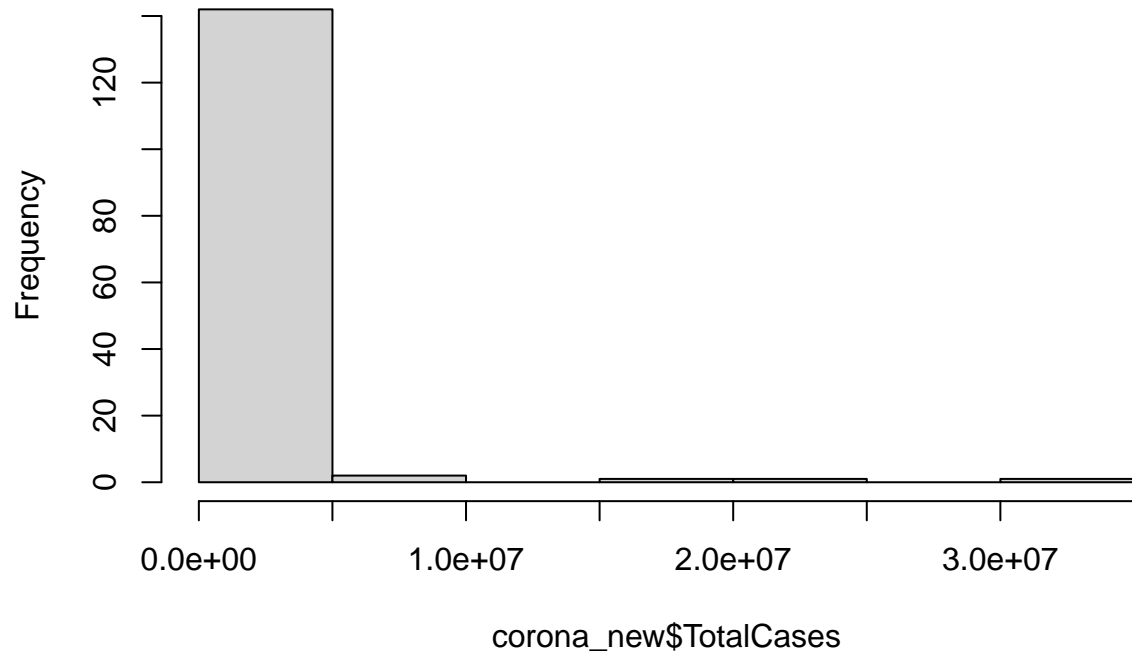
```
#head(corona_new)
```

```
boxplot(TotalDeaths ~ rank, corona_new, xlab = "rank", ylab = "TotalDeaths")
```

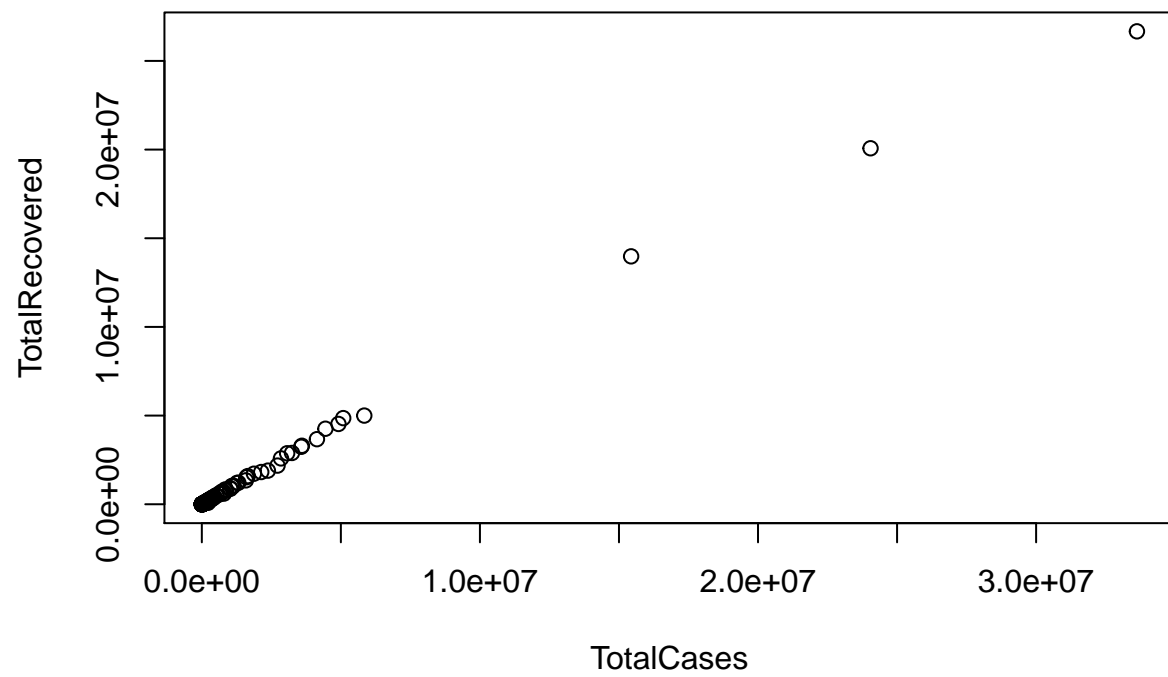


```
hist(corona_new$TotalCases)
```


Histogram of corona_new\$TotalCases

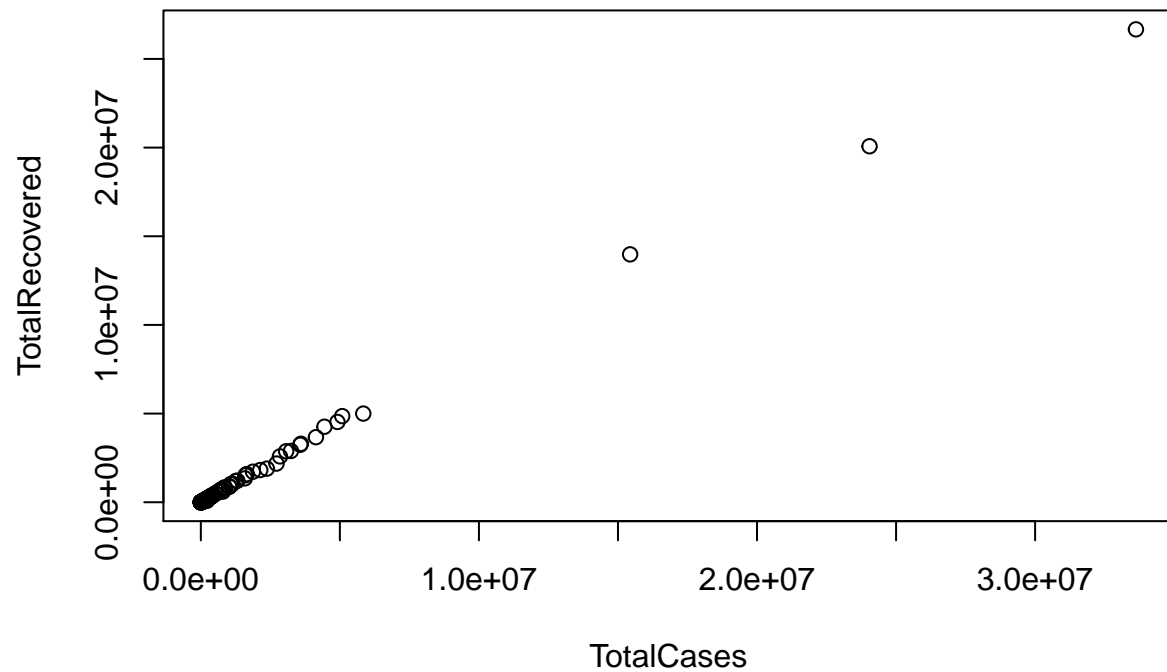


```
with(corona_new, plot(TotalCases, TotalRecovered))
```



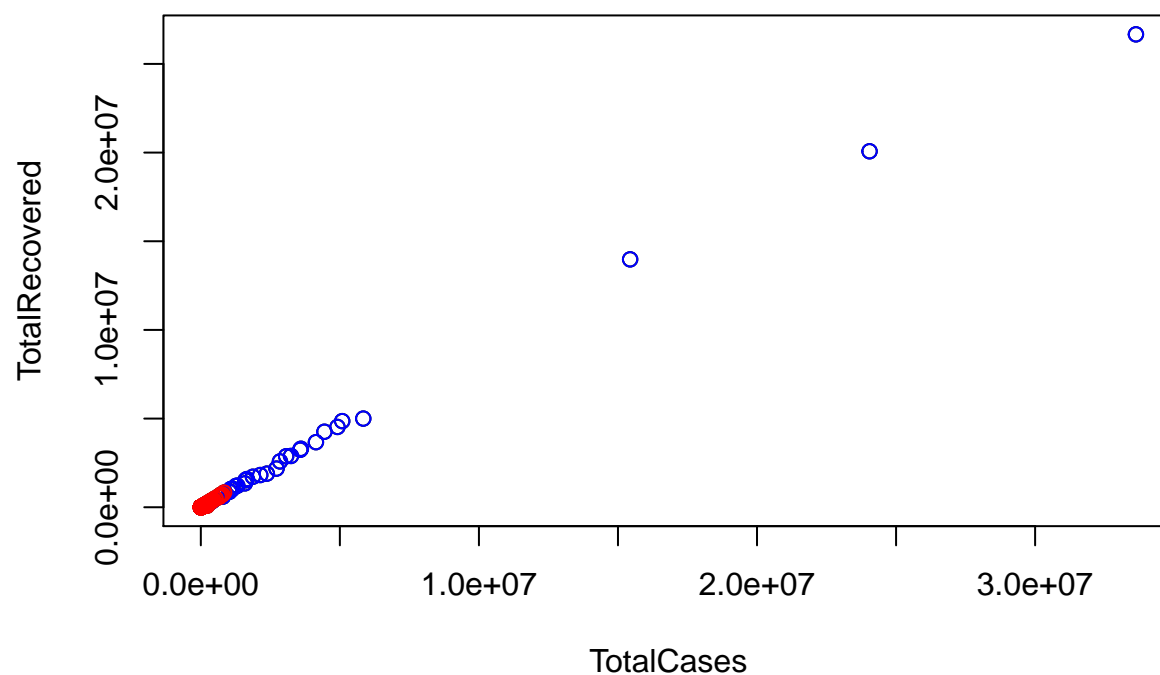
```
# adding title
with(corona_new, plot(TotalCases, TotalRecovered))
title(main = "TotalCases vs TotalRecovered")
```

TotalCases vs TotalRecovered



```
# adding colour
with(corona_new, plot>TotalCases, TotalRecovered, main = "TotalCases vs TotalRecovered"))
with(subset(corona_new, rank == 1), points>TotalCases, TotalRecovered, col = "blue"))
with(subset(corona_new, rank == 0), points>TotalCases, TotalRecovered, col = "red"))
```

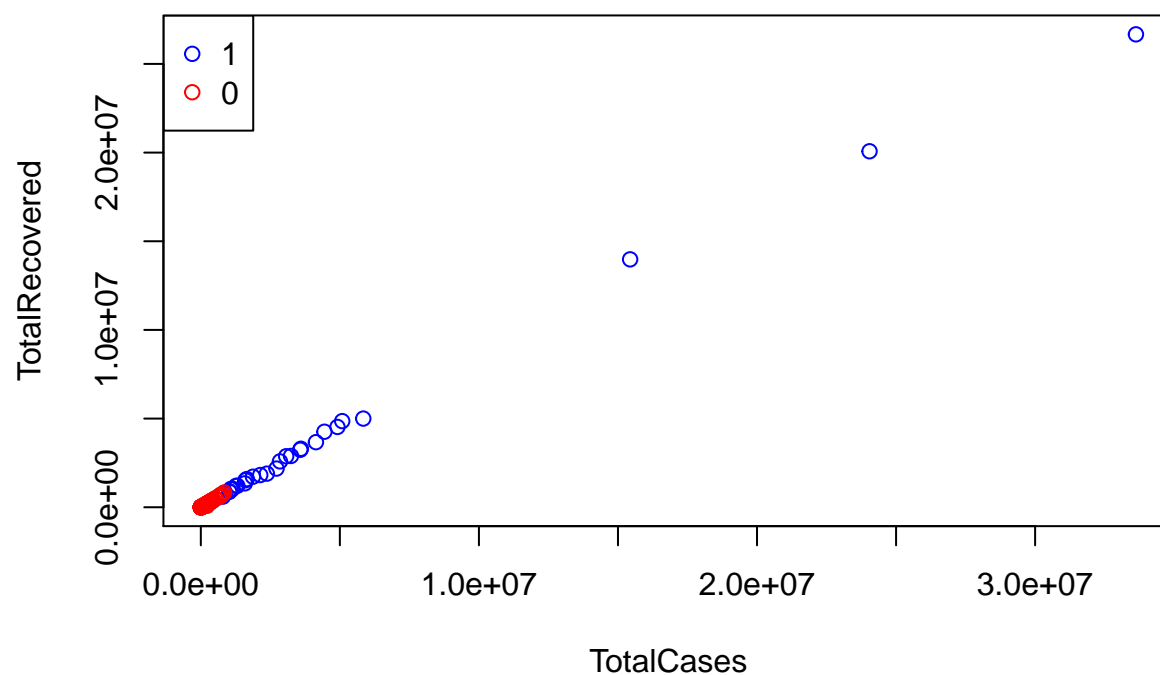
TotalCases vs TotalRecovered



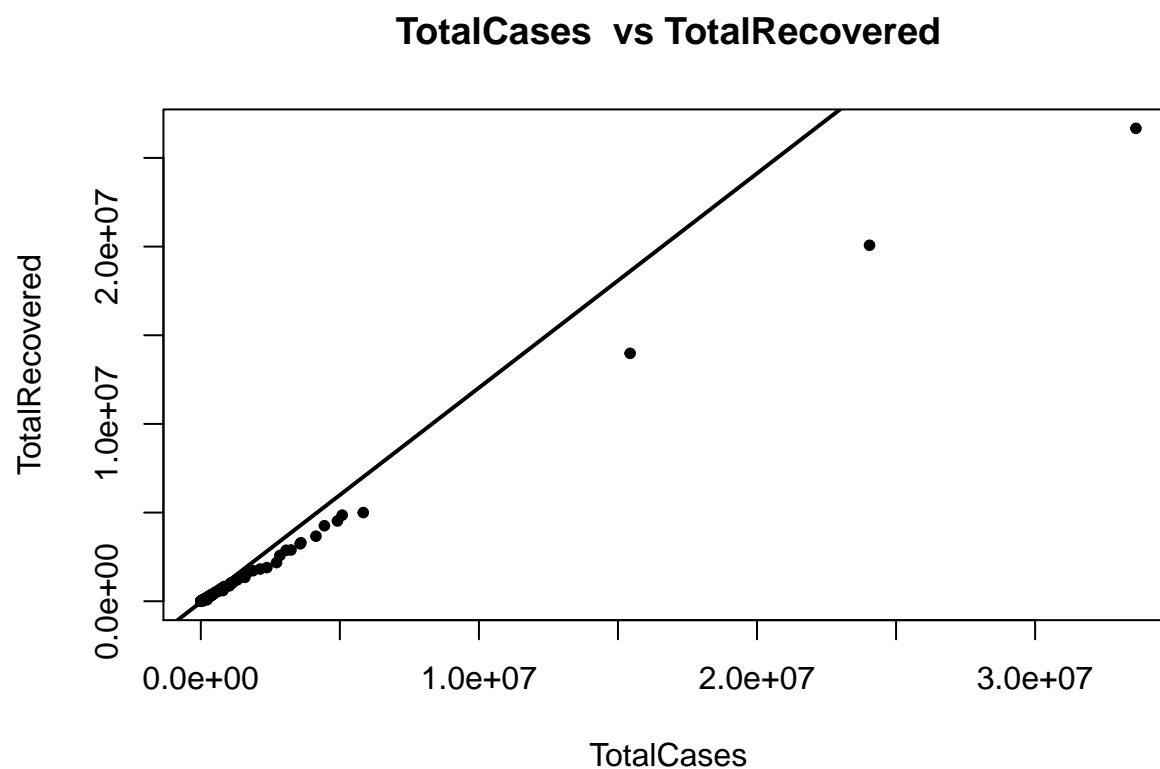
```
# Base Plot with Annotation
```

```
with(corona_new, plot(TotalCases, TotalRecovered, main = "TotalCases vs TotalRecovered", type = "n"))  
with(subset(corona_new, rank == 1), points(TotalCases, TotalRecovered, col = "blue"))  
with(subset(corona_new, rank == 0), points(TotalCases, TotalRecovered, col = "red"))  
  
legend("topleft", pch = 1, col = c("blue", "red"), legend = c("1", "0"))
```

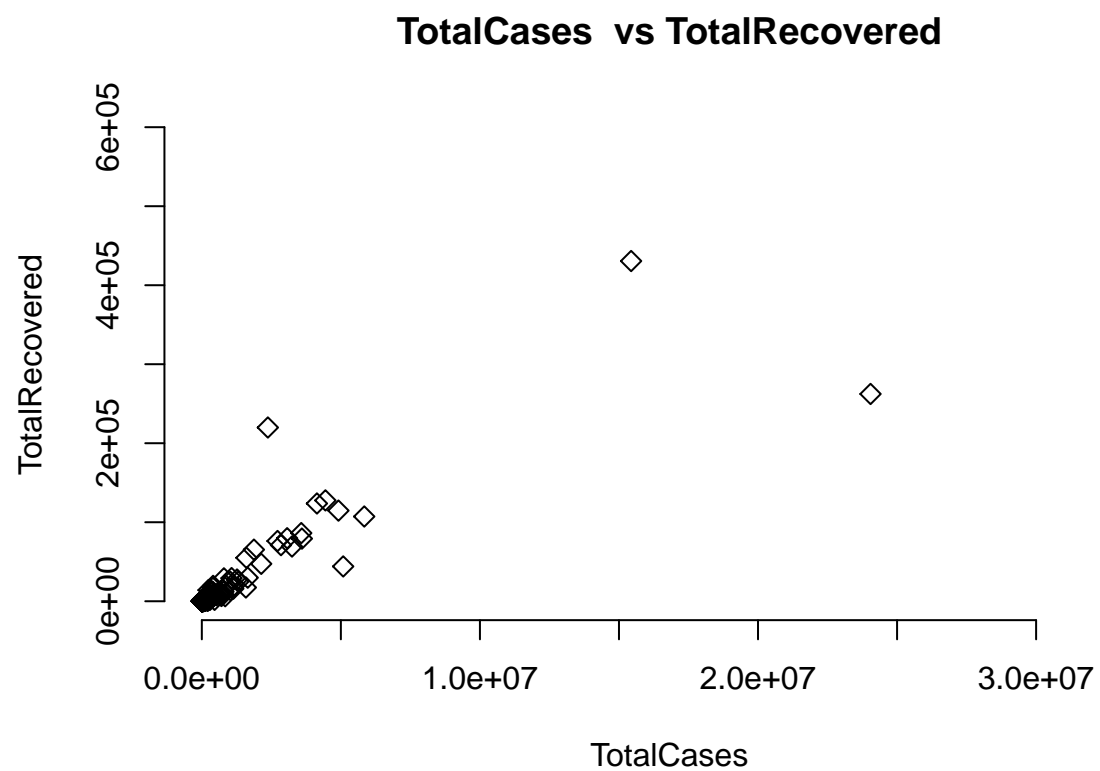
TotalCases vs TotalRecovered

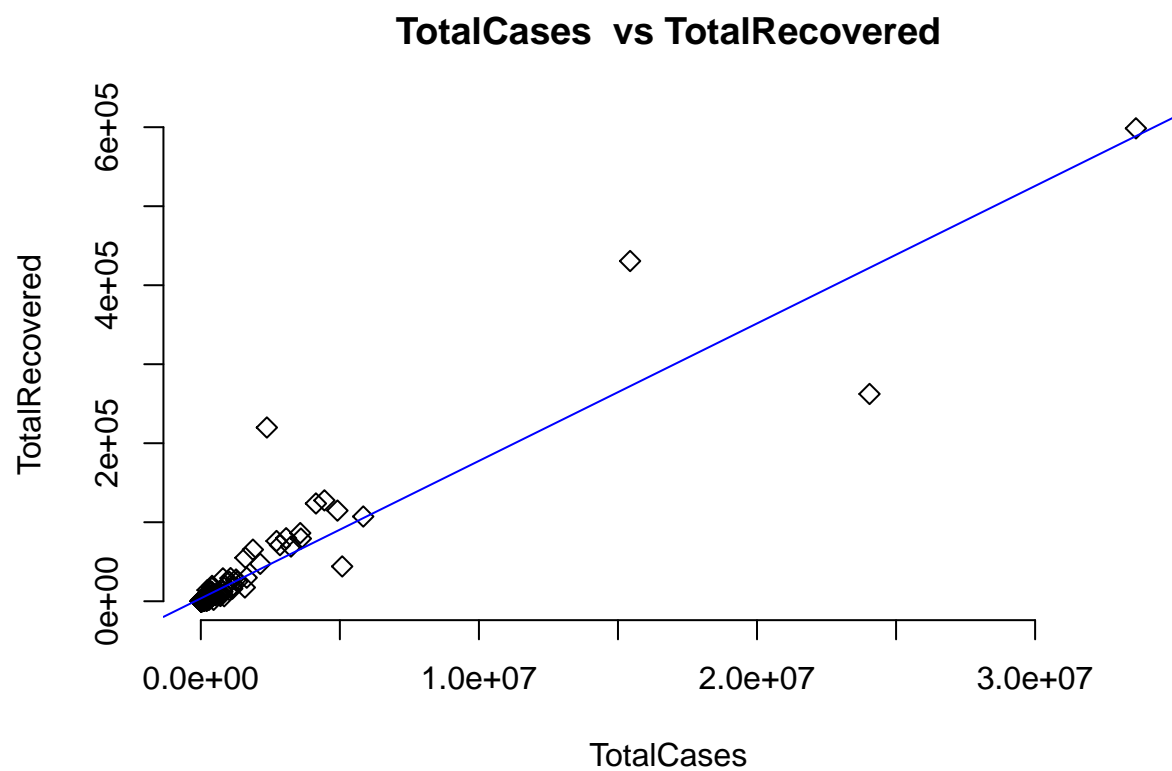


```
# Base Plot with Regression Line
with(corona_new, plot(TotalCases, TotalRecovered, main = "TotalCases vs TotalRecovered", pch = 20))
model <- lm(TotalCases ~ TotalRecovered, corona_new)
abline(model, lwd = 2)
```



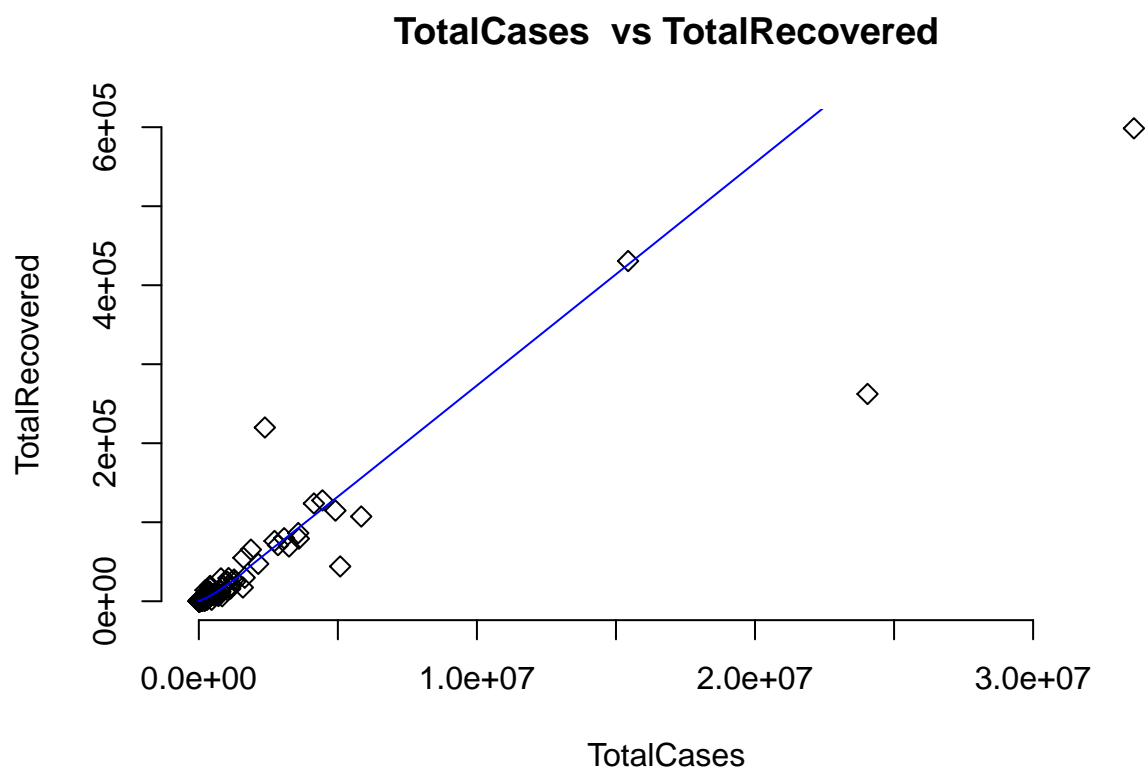
```
# R base scatter plot: plot()
x <- corona_new$TotalCases
y <- corona_new$TotalDeaths
# Plot with main and axis titles
# Change point shape (pch = 19) and remove frame.
plot(x, y, main = "TotalCases vs TotalRecovered",
      xlab = "TotalCases", ylab = "TotalRecovered",
      pch = 5, frame = FALSE)
```





```
# Add loess fit
plot(x, y, main = "TotalCases vs TotalRecovered",
     xlab = "TotalCases", ylab = "TotalRecovered",
     pch = 5, frame = FALSE)

lines(lowess(x, y), col = "blue")
```

```
library(ggplot2)
library("car")
```

```
## Loading required package: carData
```

```
##
```

```
## Attaching package: 'car'
```

```
## The following object is masked from 'package:purrr':
```

```
##
```

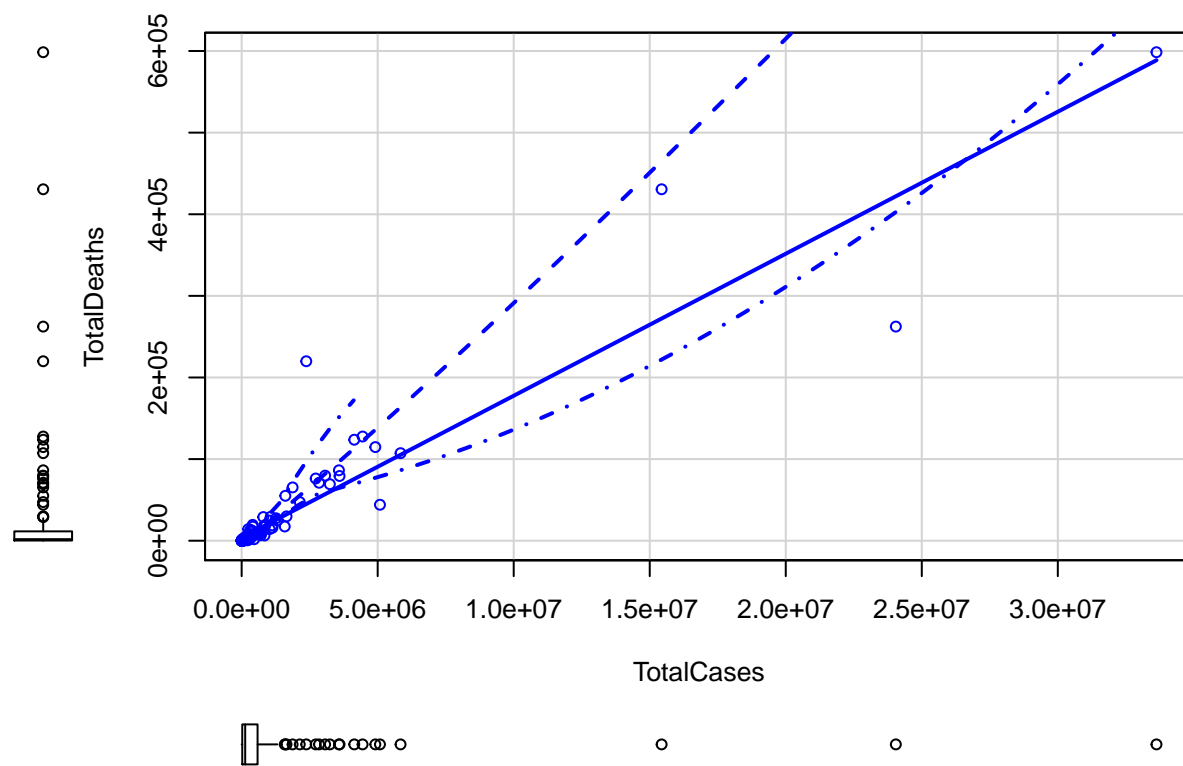
```
## some
```

```
## The following object is masked from 'package:dplyr':
```

```
##
```

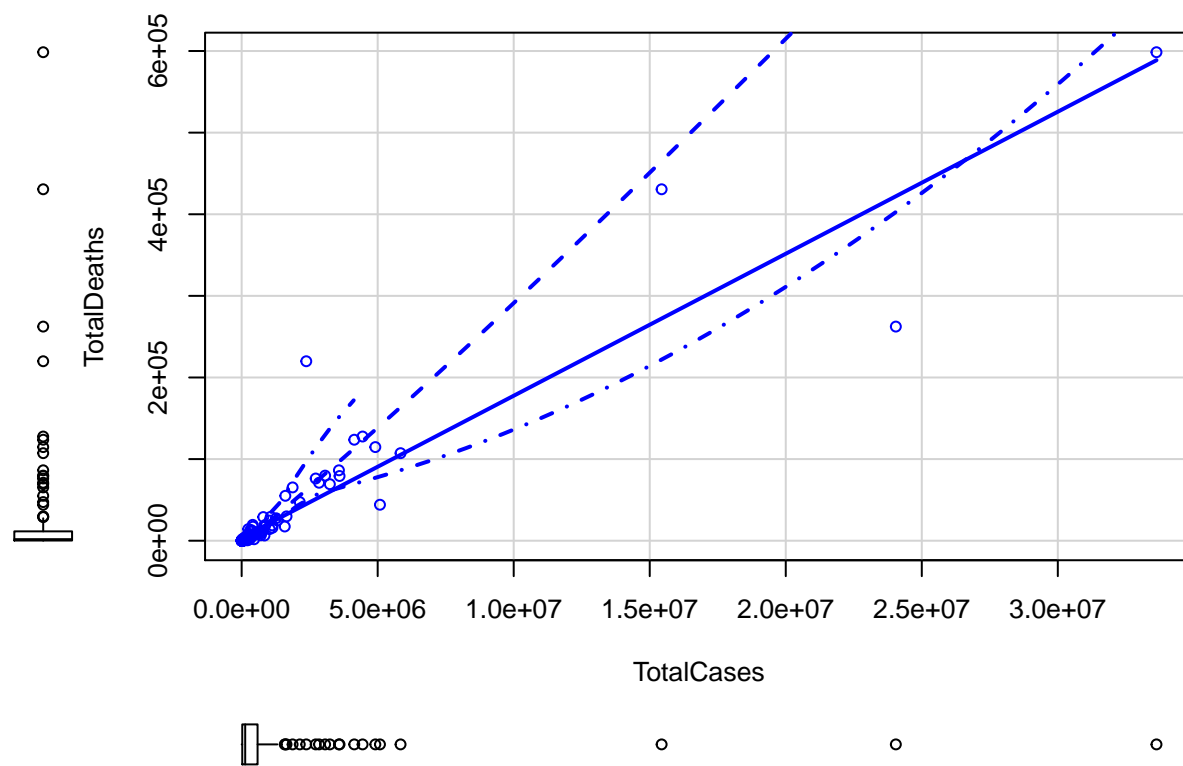
```
## recode
```

```
scatterplot(TotalDeaths ~ TotalCases, data = corona_new)
```

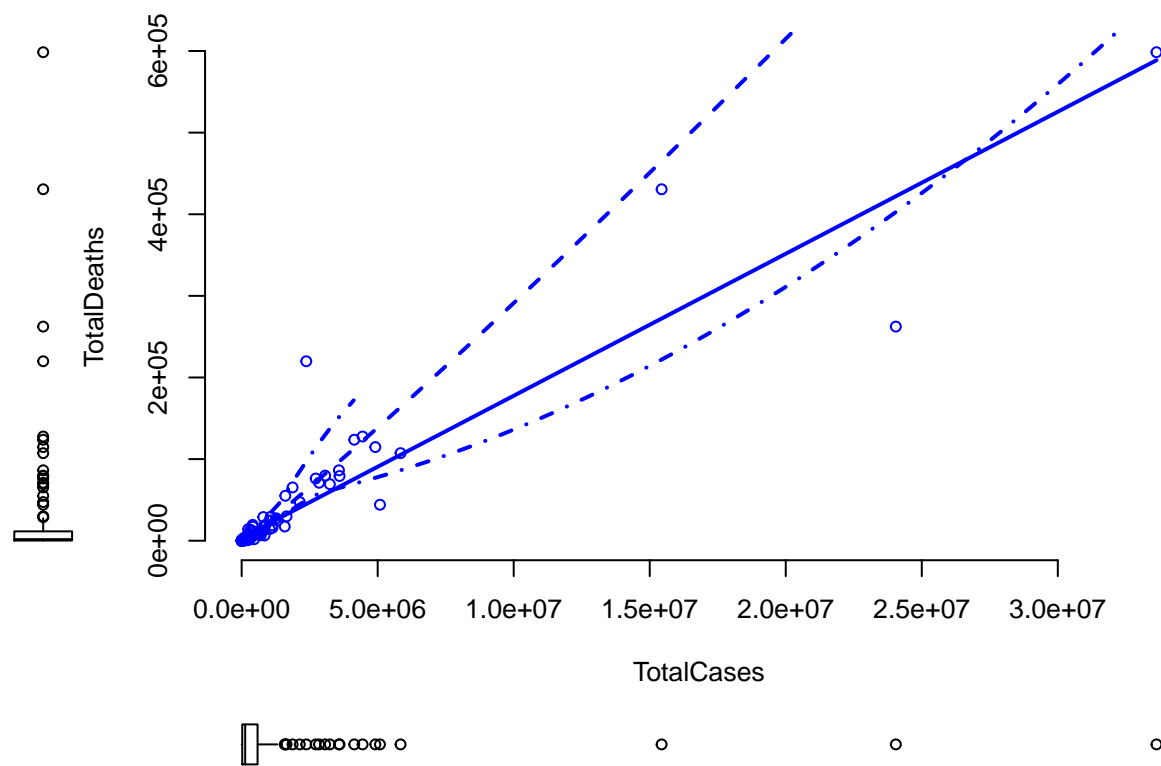


Suppress the smoother and frame

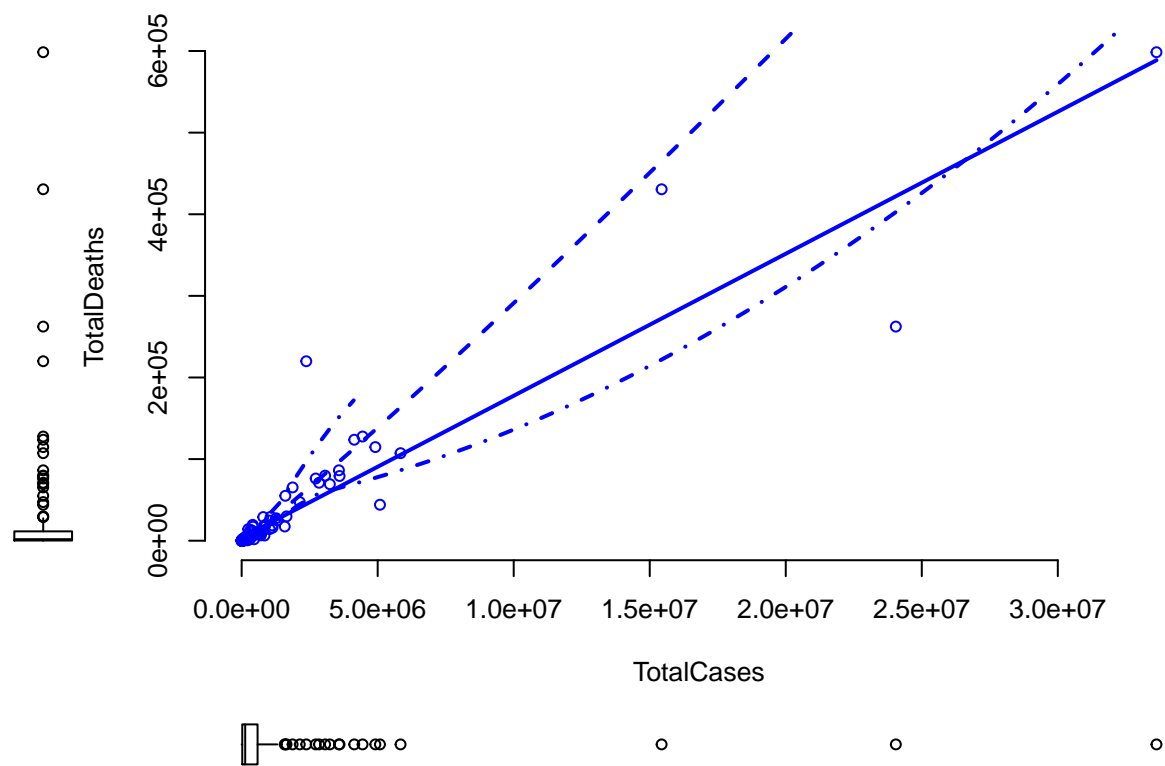
```
scatterplot(TotalDeaths ~ TotalCases, data = corona_new,
            grid = TRUE, frame = FALSE)
```



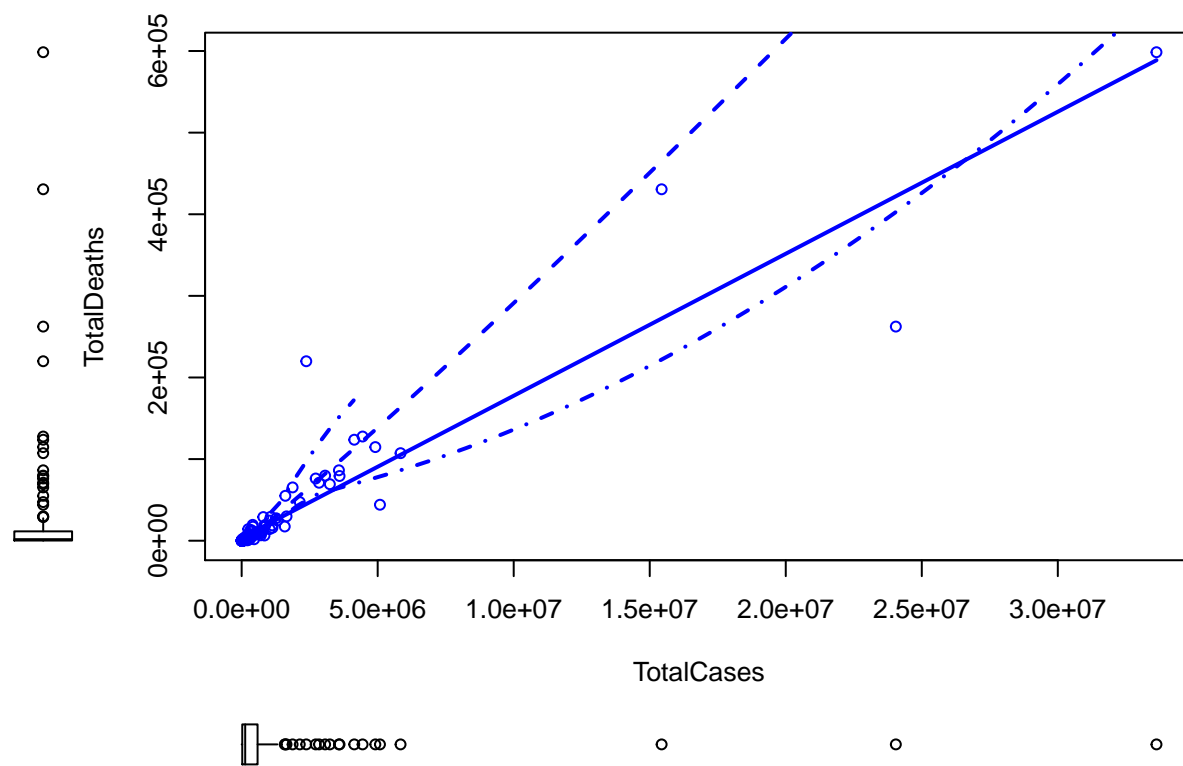
```
scatterplot(TotalDeaths ~ TotalCases, data = corona_new,
            grid = FALSE, frame = FALSE)
```



```
# smoother = FALSE,
# Scatter plot by groups ("cyl")
library(ggplot2)
scatterplot(TotalDeaths ~ TotalCases,
            data = corona_new,
            grid = FALSE, frame = FALSE)
```

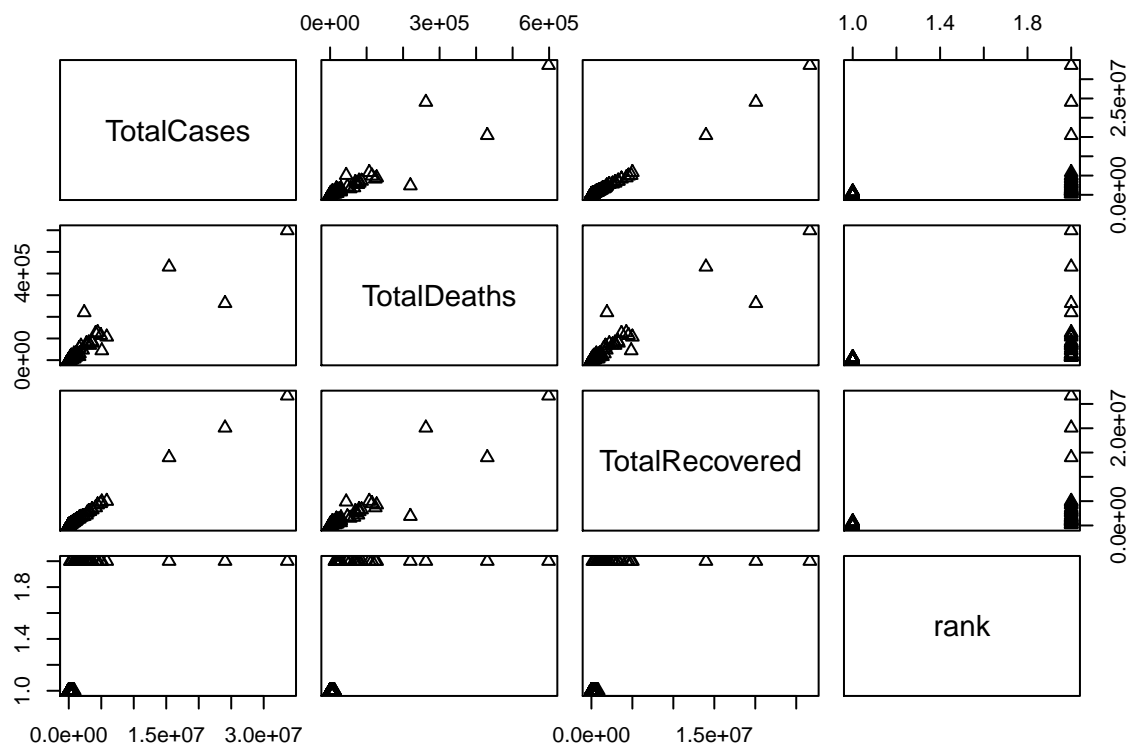


```
scatterplot(TotalDeaths ~ TotalCases,
            data = corona_new,
            grid = FALSE, frame = TRUE)
```

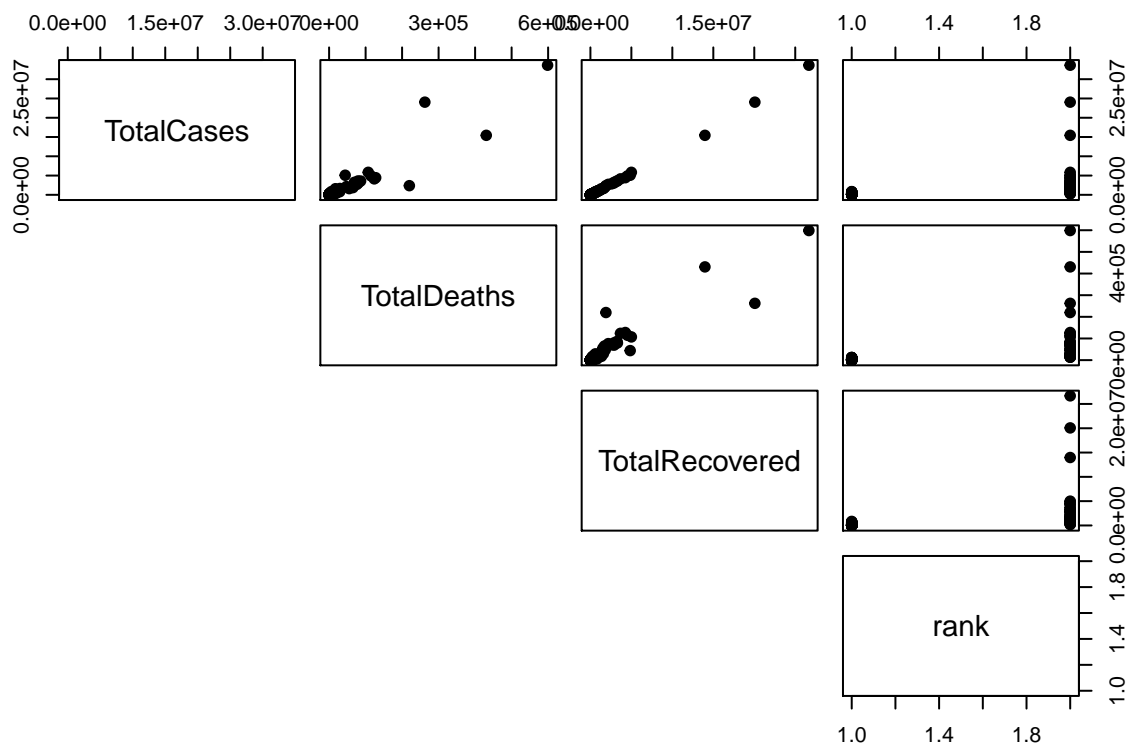


```
# Scatter Plot Matrices - R Base Graphs
# Basic plots:
```

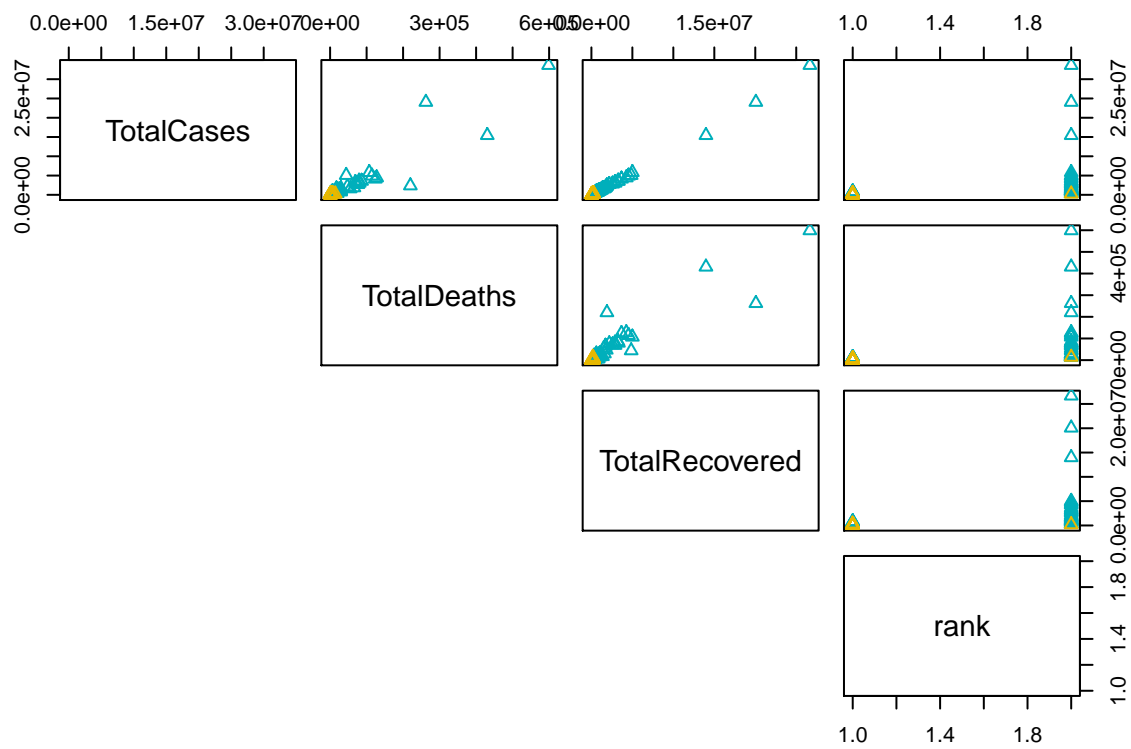
```
pairs(corona_new[,1:4], pch = 2)
```



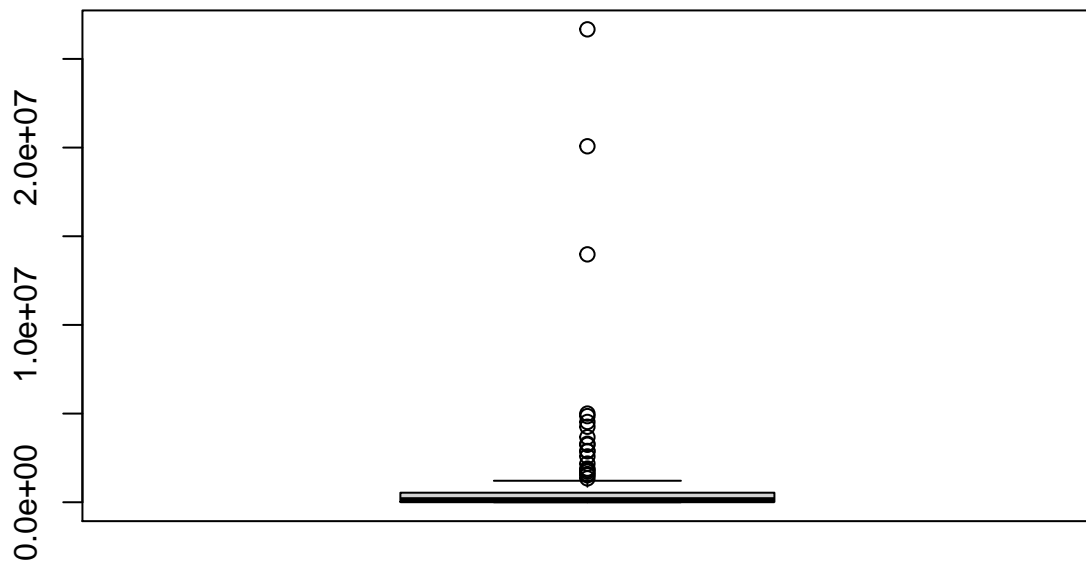
```
# Show only upper panel:
pairs(corona_new[,1:4], pch = 19, lower.panel = NULL)
```



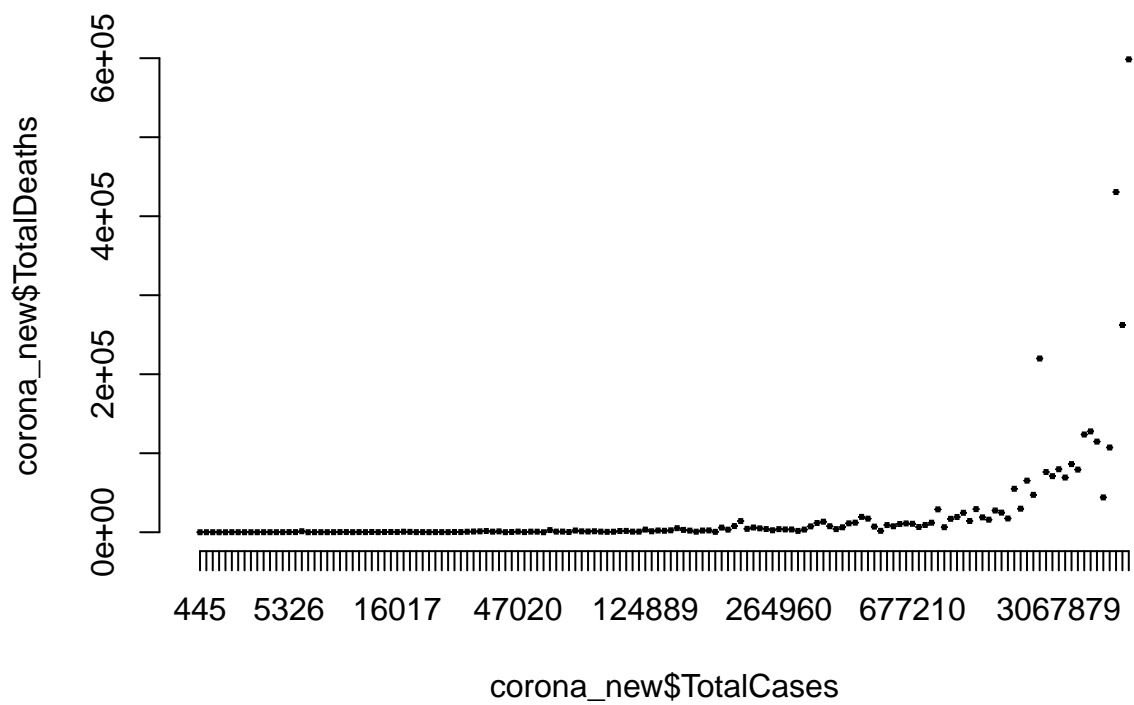
```
# Color points by groups (species)
my_cols <- c("#00AFBB", "#E7B800")
pairs(corona_new[,1:4], pch = 2, cex = 1,
      col = my_cols[iris$Species],
      lower.panel=NULL)
```

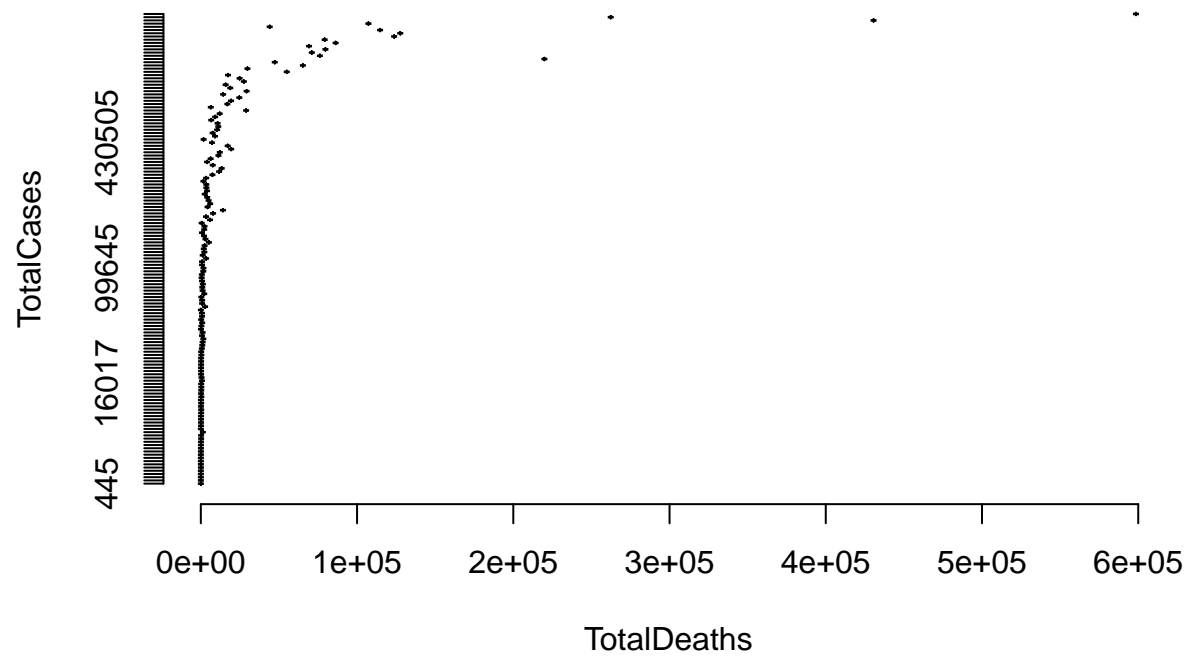
```
# Basic box plots
# Box plot of one variable
boxplot(corona_new$TotalRecovered)
```



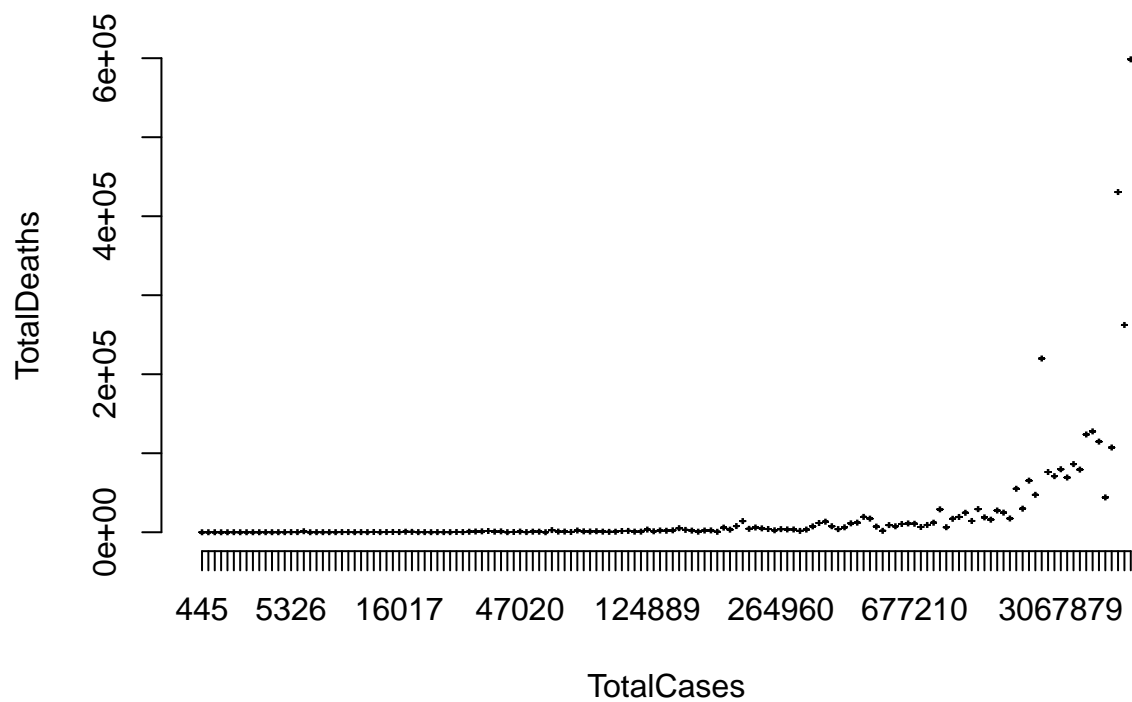
```
# Box plots by groups (dose)
# remove frame
boxplot(corona_new$TotalDeaths ~ corona_new$TotalCases,
        data = corona_new, frame = FALSE)
```



```
# Horizontal box plots
boxplot(TotalDeaths ~ TotalCases, data = corona_new, frame = FALSE,
        horizontal = TRUE)
```

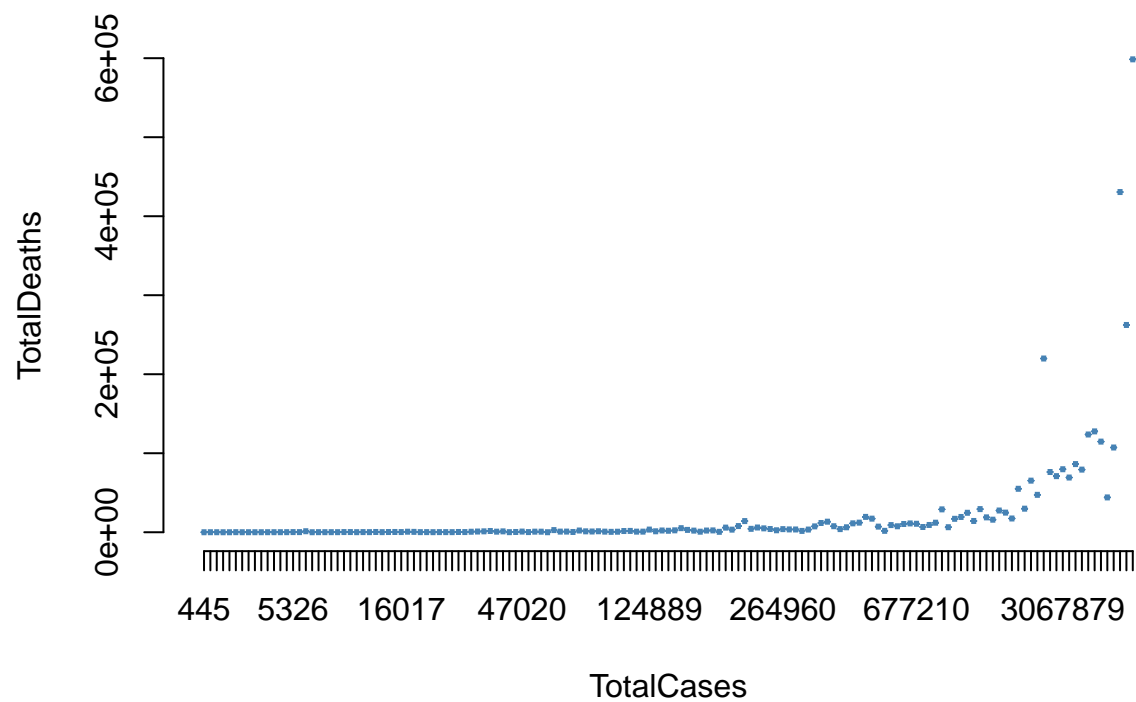


```
# Notched box plots  
boxplot(TotalDeaths ~ TotalCases, data = corona_new, frame = FALSE,  
        notch = TRUE)
```

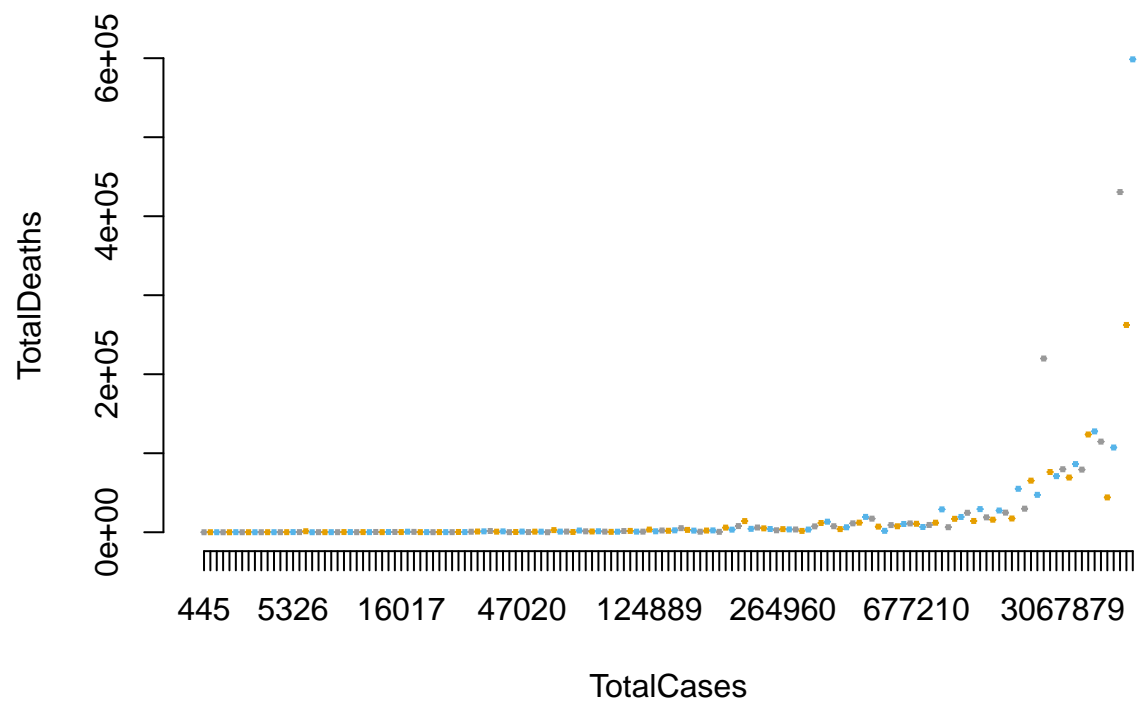


```
# Change group names
#boxplot(TotalDeaths ~ TotalCases, data = corona_new, frame = FALSE, names = c("D0.5", "D1", "D2"))

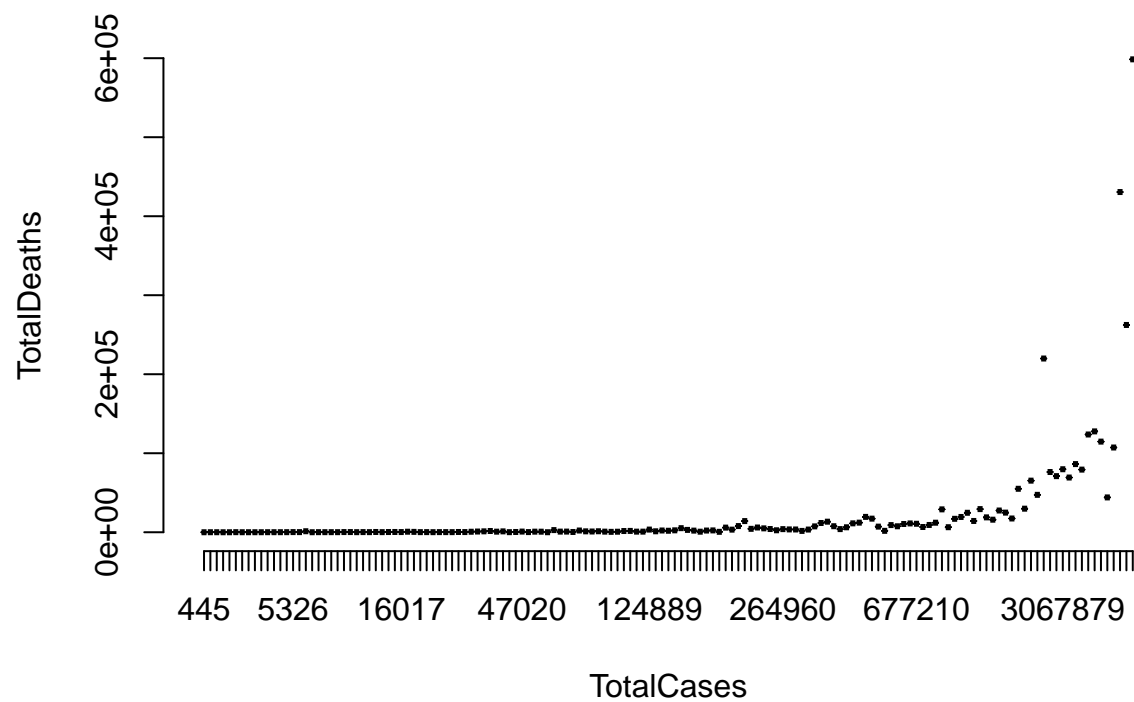
# Change color
# Change the color of border using one single color
boxplot(TotalDeaths ~ TotalCases, data = corona_new, frame = FALSE,
        border = "steelblue")
```



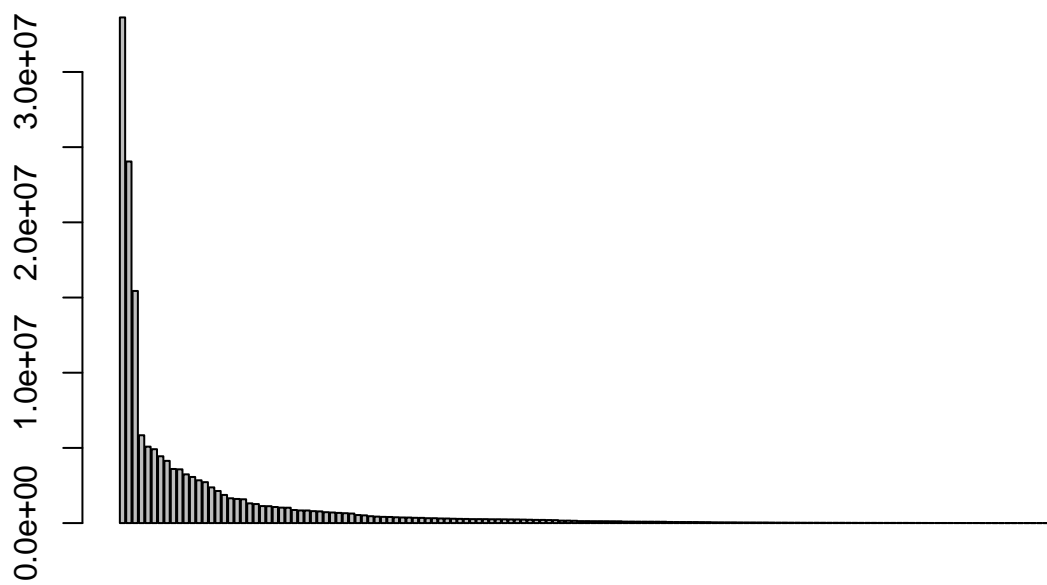
```
# Change the color of border.
# Use different colors for each group
boxplot(TotalDeaths ~ TotalCases, data = corona_new, frame = FALSE,
        border = c("#999999", "#E69F00", "#56B4E9"))
```



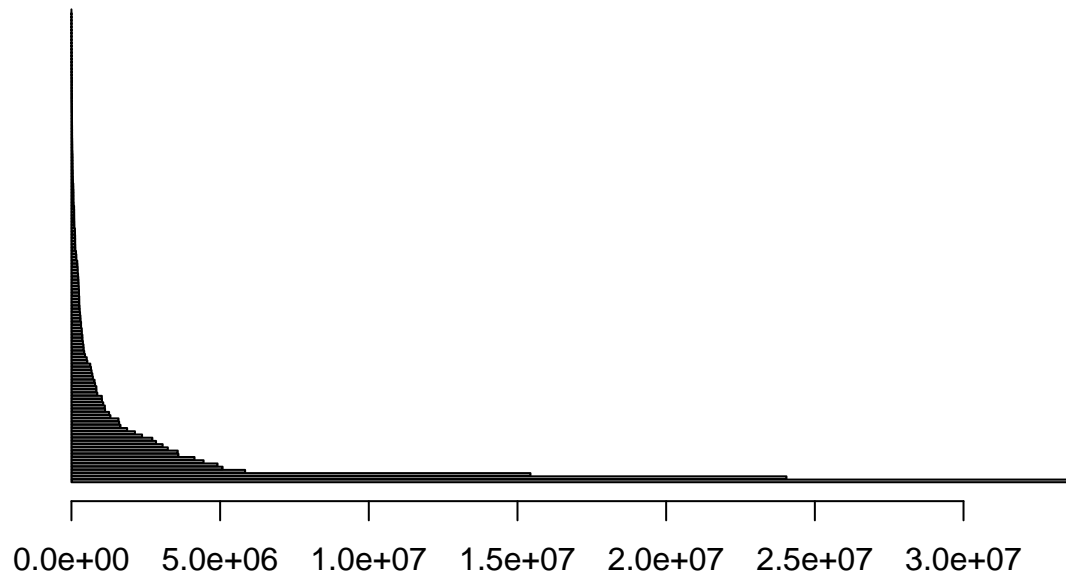
```
# Change fill color : single color
boxplot(TotalDeaths ~ TotalCases, data = corona_new, frame = FALSE,
        col = "steelblue")
```



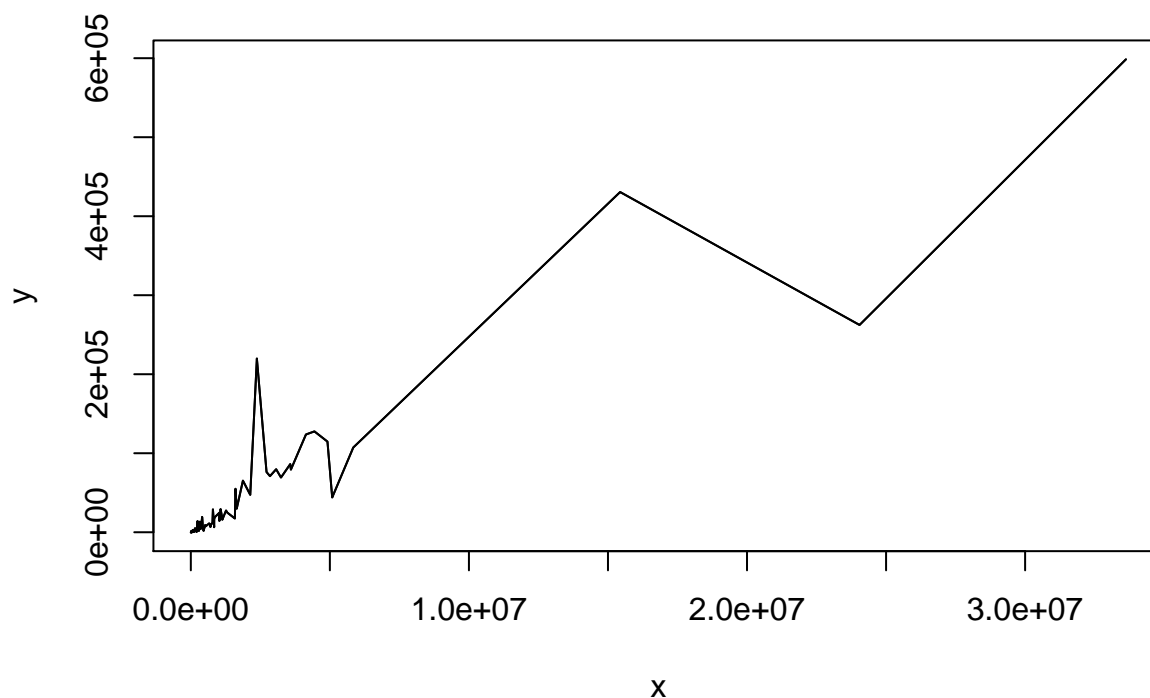
```
# Change group names
# barplot(corona_new$TotalCases, names.arg = c("A", "B", "C"))
# Bar plot of one variable
barplot(corona_new$TotalCases)
```

```
# Horizontal bar plot  
barplot(corona_new$TotalCases, horiz = TRUE)
```



```
# Line Plots - R Base Graphs  
plot(x, y, type = "l", lty = 1)  
lines(x, y, type = "l", lty = 1)
```



```
#### Plotly
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##   last_plot
```

```
## The following object is masked from 'package:stats':
##
##   filter
```

```
## The following object is masked from 'package:graphics':
##
##   layout
```

```
# Scatter plot
fig <- plot_ly(data = corona_data_updated, x = ~corona_data_updated$TotalCases,
               y = ~corona_data_updated$TotalRecovered,
               marker = list(size = 10,
                             color = 'pink',
                             line = list(color = 'green',
```

```

                                width = 1)))
fig <- fig %>% layout(title = 'Customized Scatter Plot',
                     yaxis = list(zeroline = FALSE),
                     xaxis = list(zeroline = FALSE))
fig

```

PhantomJS not found. You can install it with `webshot::install_phantomjs()`. If it is installed, please

No trace type specified:

Based on info supplied, a 'scatter' trace seems appropriate.

Read more about this trace type -> <https://plotly.com/r/reference/#scatter>

No scatter mode specified:

Setting the mode to markers

Read more about this attribute -> <https://plotly.com/r/reference/#scatter-mode>

```

# head(data1)
fig <- data1 %>%
  plot_ly(
    x = ~data1$Tests_1M_pop,
    y = ~Deaths_1M_pop,
    size = ~Tests_1M_pop,
    frame = ~Continent ,
    text = ~Country_Other ,
    hoverinfo = "text",
    type = 'scatter',
    mode = 'markers'
  )

fig <- fig %>% layout(
  xaxis = list(
    type = "log"
  )
)

fig

```

Warning: 'line.width' does not currently support multiple values.

Warning: 'line.width' does not currently support multiple values.

Warning: 'line.width' does not currently support multiple values.

Warning: 'line.width' does not currently support multiple values.

Warning: 'line.width' does not currently support multiple values.

Warning: 'line.width' does not currently support multiple values.

```
#fig <- plot_ly(x = ~data1$TotalCases, y = ~data1$TotalDeaths, z = ~data1$TotalRecovered, data=, type = )
```

```
fig <- plot_ly(data1, x = ~Tot_Cases_1M_pop, y = ~Continent, name = "TotalCases",
               type = 'scatter', mode = "markers",
               marker = list(color = "red", opacity = 0.4), size = 5)%>%
  add_trace(x = ~ActiveCases, y = ~Continent, name = "ActiveCases",
            type = 'scatter', mode = "markers", marker = list(color = "blue", opacity = 0.4)) %>%
  layout(title = "Total Deaths vs Active vs Recovered cases")

fig
```

```
# Data Preparation
labels <- data1$Continent
values <- data1$Deaths_1M_pop
# Data Visualization
fig <- plot_ly(type='pie', labels=labels, values=values,
               textinfo='label+percent',
               insidetextorientation='radial') %>% layout(title = "Total deaths")

fig
```

```
# Data Preparation
labels <- data1$Continent
values <- data1$ActiveCases
# Data Visualization
fig <- plot_ly(type='pie', labels=labels, values=values,
               textinfo='label+percent',
               insidetextorientation='radial') %>% layout(title = "Total ActiveCases")

fig
```

```
#precautions = c(Low, Medium, High)
#sum(is.na(corona_data_updated))
final_data <- corona_data_updated %>% select(TotalCases,
                                             Country_Other,
                                             TotalDeaths,
                                             ActiveCases,
                                             Serious_Critical)

precautions <- case_when(data1$Serious_Critical < mean(data1$Serious_Critical)~ "Low",
                         data1$Serious_Critical > mean(data1$Serious_Critical)~ "High")

new_data <- data1 %>% mutate(precautions) %>%
  select(Country_Other,
         Continent,
         TotalCases,
         TotalDeaths,
         TotalRecovered,
         Serious_Critical,
         precautions)
head(new_data)
```

```
##      Country_Other      Continent TotalCases TotalDeaths TotalRecovered
## 9              USA North America  33626097      598540      26667199
```

```
## 10      India      Asia  24046120      262350      20073367
## 11      Brazil South America 15436827      430596      13979329
## 12      France     Europe   5841129      107250      4999079
## 13      Turkey     Asia    5083996      44059      4856763
## 14      Russia     Europe   4913439      114723      4527878
## Serious_Critical precautions
## 9          8611      High
## 10         8944      High
## 11         8318      High
## 12         4442      High
## 13         2765      High
## 14         2300      High
```

```
# load libraries
library(mlbench)
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
## lift
```

```
control <- trainControl(method="repeatedcv", number=10, repeats=3)
seed <- 7
```

```
metric <- c("RMSE", "Rsquared" )
preProcess=c("center", "scale")
#head(data1)
#sum(is.na(data1))
```

```
library(caTools)
set <- sample(2, nrow(new_data),
             replace = TRUE,
             prob = c(0.7, 0.3))
train <- new_data[set==1,]

test <- new_data[set==2,]
train$precautions <- as.factor(train$precautions)
test$precautions <- as.factor(test$precautions)
str(train$precautions)
```

```
## Factor w/ 2 levels "High","Low": 1 1 1 1 1 1 1 1 1 1 ...
```

```
# Random Forest
set.seed(seed)
fit.rf <- train(precautions~ Serious_Critical,
```

```

      data=train,
      method="rf",
      trControl=control)
fit.rf

## Random Forest
##
## 105 samples
## 1 predictor
## 2 classes: 'High', 'Low'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 95, 95, 95, 94, 94, 94, ...
## Resampling results:
##
## Accuracy Kappa
## 1 1
##
## Tuning parameter 'mtry' was held constant at a value of 2

```

```
summary(fit.rf)
```

```

##           Length Class      Mode
## call           4  -none-    call
## type            1  -none- character
## predicted       105 factor    numeric
## err.rate       1500 -none-    numeric
## confusion        6  -none-    numeric
## votes          210 matrix    numeric
## oob.times       105 -none-    numeric
## classes         2  -none- character
## importance        1  -none-    numeric
## importanceSD      0  -none-     NULL
## localImportance  0  -none-     NULL
## proximity        0  -none-     NULL
## ntree           1  -none-    numeric
## mtry            1  -none-    numeric
## forest          14  -none-    list
## y              105 factor    numeric
## test            0  -none-     NULL
## inbag           0  -none-     NULL
## xNames          1  -none- character
## problemType      1  -none- character
## tuneValue        1 data.frame list
## obsLevels        2  -none- character
## param            0  -none-    list

```

```

pred <- predict(fit.rf, test)

Scores_comp <- data.frame(Actual = test$precautions,
                          Predcited = pred)
head(Scores_comp)

```

```
##   Actual Predicted
## 1   High      High
## 2   High      High
## 3   Low       Low
## 4   High      High
## 5   High      High
## 6   Low       Low
```

```
plot(Scores_comp)
```

