Group 6 Assignment 1: R Programming

Contents

R Markdown

```
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 4.0.5
```

```
## Loading required package: ggplot2
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.1.0     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
## v purrr   0.3.4
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(janitor)  #janitor helps us clean datasets
```

```
## Warning: package 'janitor' was built under R version 4.0.5
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(here)  # here helps us know where files are
```

```
## Warning: package 'here' was built under R version 4.0.5
```

```
## here() starts at C:/Users/user/Downloads
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.0.5
```

```
library(readr)
library("formatR")
```

Load data - CSV

This data is a COVID dataset and downloaded open data

```
COVID19_Dataset <- readr::read_csv("https://opendata.ecdc.europa.eu/covid19/nationalcasedeath/csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   country = col_character(),
##   country_code = col_character(),
##   continent = col_character(),
##   population = col_double(),
##   indicator = col_character(),
##   weekly_count = col_double(),
##   year_week = col_character(),
##   rate_14_day = col_double(),
##   cumulative_count = col_double(),
##   source = col_character()
## )
```

Print the structure of your dataset

```
str(COVID19_Dataset)
```

```
## spec_tbl_df [25,818 x 10] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ country         : chr [1:25818] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
##  $ country_code    : chr [1:25818] "AFG" "AFG" "AFG" "AFG" ...
##  $ continent       : chr [1:25818] "Asia" "Asia" "Asia" "Asia" ...
##  $ population      : num [1:25818] 38928341 38928341 38928341 38928341 38928341 ...
##  $ indicator       : chr [1:25818] "cases" "cases" "cases" "cases" ...
##  $ weekly_count    : num [1:25818] 0 0 0 0 0 0 0 0 1 3 ...
##  $ year_week       : chr [1:25818] "2020-01" "2020-02" "2020-03" "2020-04" ...
##  $ rate_14_day     : num [1:25818] NA 0 0 0 0 ...
##  $ cumulative_count: num [1:25818] 0 0 0 0 0 0 0 0 1 4 ...
##  $ source          : chr [1:25818] "Epidemic intelligence, national weekly data" "Epidemic intelliger
##  - attr(*, "spec")=
##   .. cols(
##   ..   country = col_character(),
##   ..   country_code = col_character(),
##   ..   continent = col_character(),
##   ..   population = col_double(),
##   ..   indicator = col_character(),
##   ..   weekly_count = col_double(),
##   ..   year_week = col_character(),
```

```
##   ..   rate_14_day = col_double(),
##   ..   cumulative_count = col_double(),
##   ..   source = col_character()
##   .. )
```

List the variables in your dataset

```
names(COVID19_Dataset)
```

```
##  [1] "country"          "country_code"     "continent"        "population"
##  [5] "indicator"        "weekly_count"     "year_week"        "rate_14_day"
##  [9] "cumulative_count" "source"
```

Print the top 15 rows of your dataset

```
head(COVID19_Dataset, n = 15)
```

```
## # A tibble: 15 x 10
##    country    country_code continent population indicator weekly_count year_week
##    <chr>      <chr>        <chr>          <dbl> <chr>             <dbl> <chr>
##  1 Afghanist~ AFG          Asia        38928341 cases                 0 2020-01
##  2 Afghanist~ AFG          Asia        38928341 cases                 0 2020-02
##  3 Afghanist~ AFG          Asia        38928341 cases                 0 2020-03
##  4 Afghanist~ AFG          Asia        38928341 cases                 0 2020-04
##  5 Afghanist~ AFG          Asia        38928341 cases                 0 2020-05
##  6 Afghanist~ AFG          Asia        38928341 cases                 0 2020-06
##  7 Afghanist~ AFG          Asia        38928341 cases                 0 2020-07
##  8 Afghanist~ AFG          Asia        38928341 cases                 0 2020-08
##  9 Afghanist~ AFG          Asia        38928341 cases                 1 2020-09
## 10 Afghanist~ AFG          Asia        38928341 cases                 3 2020-10
## 11 Afghanist~ AFG          Asia        38928341 cases                12 2020-11
## 12 Afghanist~ AFG          Asia        38928341 cases                18 2020-12
## 13 Afghanist~ AFG          Asia        38928341 cases                80 2020-13
## 14 Afghanist~ AFG          Asia        38928341 cases               185 2020-14
## 15 Afghanist~ AFG          Asia        38928341 cases               308 2020-15
## # ... with 3 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>
```

Write a user defined function using any of the variables from the data set

```
my_function <- function(x, y) {
    output <- COVID19_Dataset %>%
        group_by(indicator, continent, year_week) %>%
        summarize(mean_weekly_count = mean(weekly_count, na.rm = TRUE)) %>%
        filter(indicator == x, year_week == y)
    return(output)
}
```

```
my_function("deaths", "2021-01")
```

```
## `summarise()` has grouped output by 'indicator', 'continent'. You can override using the `.groups` a
```

```
## # A tibble: 5 x 4
## # Groups:   indicator, continent [5]
##   indicator continent year_week mean_weekly_count
##   <chr>     <chr>     <chr>                 <dbl>
## 1 deaths    Africa    2021-01              198.
## 2 deaths    America   2021-01             1791.
## 3 deaths    Asia      2021-01              288.
## 4 deaths    Europe    2021-01             1742.
## 5 deaths    Oceania   2021-01                1.29
```

```
my_function("cases", "2020-12")
```

```
## `summarise()` has grouped output by 'indicator', 'continent'. You can override using the `.groups` a
```

```
## # A tibble: 5 x 4
## # Groups:   indicator, continent [5]
##   indicator continent year_week mean_weekly_count
##   <chr>     <chr>     <chr>                 <dbl>
## 1 cases     Africa    2020-12                48.5
## 2 cases     America   2020-12              2027.
## 3 cases     Asia      2020-12               793.
## 4 cases     Europe    2020-12              6501.
## 5 cases     Oceania   2020-12               390.
```

Use data manipulation techniques and filter rows based on any logical criteria that exist in your dataset

```
COVID19_Dataset_Canada = COVID19_Dataset %>%
    filter(COVID19_Dataset$weekly_count >= 1000, COVID19_Dataset$country ==
        "Canada")

head(COVID19_Dataset_Canada, n = 10)
```

```
## # A tibble: 10 x 10
##    country country_code continent population indicator weekly_count year_week
##    <chr>   <chr>        <chr>          <dbl> <chr>            <dbl> <chr>
##  1 Canada  CAN          America     37742157 cases             1126 2020-12
##  2 Canada  CAN          America     37742157 cases             4825 2020-13
##  3 Canada  CAN          America     37742157 cases             9241 2020-14
##  4 Canada  CAN          America     37742157 cases             8869 2020-15
##  5 Canada  CAN          America     37742157 cases            10412 2020-16
##  6 Canada  CAN          America     37742157 cases            12107 2020-17
##  7 Canada  CAN          America     37742157 cases            12590 2020-18
##  8 Canada  CAN          America     37742157 cases             9374 2020-19
##  9 Canada  CAN          America     37742157 cases             8143 2020-20
```

```
## 10 Canada  CAN          America     37742157 cases             7697 2020-21
## # ... with 3 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>
```

Identify the dependent & independent variables and use reshaping techniques and create a new data frame by joining those variables from your dataset

```
indept = data.frame(COVID19_Dataset$population, COVID19_Dataset$country,
    COVID19_Dataset$year_week)
dept = data.frame(COVID19_Dataset$weekly_count)
new_set = cbind.data.frame(indept, dept)
head(new_set, n = 10)
```

```
##    COVID19_Dataset.population COVID19_Dataset.country COVID19_Dataset.year_week
## 1                    38928341             Afghanistan                   2020-01
## 2                    38928341             Afghanistan                   2020-02
## 3                    38928341             Afghanistan                   2020-03
## 4                    38928341             Afghanistan                   2020-04
## 5                    38928341             Afghanistan                   2020-05
## 6                    38928341             Afghanistan                   2020-06
## 7                    38928341             Afghanistan                   2020-07
## 8                    38928341             Afghanistan                   2020-08
## 9                    38928341             Afghanistan                   2020-09
## 10                   38928341             Afghanistan                   2020-10
##    COVID19_Dataset.weekly_count
## 1                             0
## 2                             0
## 3                             0
## 4                             0
## 5                             0
## 6                             0
## 7                             0
## 8                             0
## 9                             1
## 10                            3
```

Remove missing values in your dataset

```
COVID19_Dataset_notnull = COVID19_Dataset[complete.cases(COVID19_Dataset),
    ]
head(COVID19_Dataset_notnull, n = 10)
```

```
## # A tibble: 10 x 10
##    country    country_code continent population indicator weekly_count year_week
##    <chr>      <chr>        <chr>          <dbl> <chr>            <dbl> <chr>
##  1 Afghanist~ AFG          Asia        38928341 cases                0 2020-02
##  2 Afghanist~ AFG          Asia        38928341 cases                0 2020-03
##  3 Afghanist~ AFG          Asia        38928341 cases                0 2020-04
##  4 Afghanist~ AFG          Asia        38928341 cases                0 2020-05
```

```
##  5 Afghanist~ AFG          Asia        38928341 cases                0 2020-06
##  6 Afghanist~ AFG          Asia        38928341 cases                0 2020-07
##  7 Afghanist~ AFG          Asia        38928341 cases                0 2020-08
##  8 Afghanist~ AFG          Asia        38928341 cases                1 2020-09
##  9 Afghanist~ AFG          Asia        38928341 cases                3 2020-10
## 10 Afghanist~ AFG          Asia        38928341 cases               12 2020-11
## # ... with 3 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>
```

Identify and remove duplicated data in your dataset

```
COVID19_Dataset_distinct = COVID19_Dataset %>%
    distinct()
head(COVID19_Dataset_distinct, n = 10)
```

```
## # A tibble: 10 x 10
##    country    country_code continent population indicator weekly_count year_week
##    <chr>      <chr>        <chr>          <dbl> <chr>            <dbl> <chr>
##  1 Afghanist~ AFG          Asia        38928341 cases                0 2020-01
##  2 Afghanist~ AFG          Asia        38928341 cases                0 2020-02
##  3 Afghanist~ AFG          Asia        38928341 cases                0 2020-03
##  4 Afghanist~ AFG          Asia        38928341 cases                0 2020-04
##  5 Afghanist~ AFG          Asia        38928341 cases                0 2020-05
##  6 Afghanist~ AFG          Asia        38928341 cases                0 2020-06
##  7 Afghanist~ AFG          Asia        38928341 cases                0 2020-07
##  8 Afghanist~ AFG          Asia        38928341 cases                0 2020-08
##  9 Afghanist~ AFG          Asia        38928341 cases                1 2020-09
## 10 Afghanist~ AFG          Asia        38928341 cases                3 2020-10
## # ... with 3 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>
```

Reorder multiple rows in descending order

```
head(COVID19_Dataset %>%
    arrange(desc(COVID19_Dataset$country, COVID19_Dataset$continent)),
    n = 10)
```

```
## # A tibble: 10 x 10
##    country  country_code continent population indicator weekly_count year_week
##    <chr>    <chr>        <chr>          <dbl> <chr>            <dbl> <chr>
##  1 Zimbabwe ZWE          Africa      14862927 cases                2 2020-12
##  2 Zimbabwe ZWE          Africa      14862927 cases                5 2020-13
##  3 Zimbabwe ZWE          Africa      14862927 cases                2 2020-14
##  4 Zimbabwe ZWE          Africa      14862927 cases                5 2020-15
##  5 Zimbabwe ZWE          Africa      14862927 cases               11 2020-16
##  6 Zimbabwe ZWE          Africa      14862927 cases                6 2020-17
##  7 Zimbabwe ZWE          Africa      14862927 cases                3 2020-18
##  8 Zimbabwe ZWE          Africa      14862927 cases                2 2020-19
##  9 Zimbabwe ZWE          Africa      14862927 cases               10 2020-20
```

```
## 10 Zimbabwe ZWE          Africa      14862927 cases               10 2020-21
## # ... with 3 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>
```

Rename some of the column names in your dataset

```
COVID19_Dataset_renamed = COVID19_Dataset %>%
    rename(Country_Name = country, Cotinent_Name = continent)
head(COVID19_Dataset_renamed, n = 10)
```

```
## # A tibble: 10 x 10
##    Country_Name country_code Cotinent_Name population indicator weekly_count
##    <chr>        <chr>        <chr>              <dbl> <chr>            <dbl>
##  1 Afghanistan  AFG          Asia            38928341 cases                0
##  2 Afghanistan  AFG          Asia            38928341 cases                0
##  3 Afghanistan  AFG          Asia            38928341 cases                0
##  4 Afghanistan  AFG          Asia            38928341 cases                0
##  5 Afghanistan  AFG          Asia            38928341 cases                0
##  6 Afghanistan  AFG          Asia            38928341 cases                0
##  7 Afghanistan  AFG          Asia            38928341 cases                0
##  8 Afghanistan  AFG          Asia            38928341 cases                0
##  9 Afghanistan  AFG          Asia            38928341 cases                1
## 10 Afghanistan  AFG          Asia            38928341 cases                3
## # ... with 4 more variables: year_week <chr>, rate_14_day <dbl>,
## #   cumulative_count <dbl>, source <chr>
```

Add new variables in your data frame by using a mathematical function (for e.g. – multiply an existing column by 2 and add it as a new variable to your data frame)

```
COVID19_Dataset_added = COVID19_Dataset %>%
    mutate(count_by_population = cumulative_count/population)
head(COVID19_Dataset_added, n = 10)
```

```
## # A tibble: 10 x 11
##    country    country_code continent population indicator weekly_count year_week
##    <chr>      <chr>        <chr>           <dbl> <chr>            <dbl> <chr>
##  1 Afghanist~ AFG          Asia         38928341 cases                0 2020-01
##  2 Afghanist~ AFG          Asia         38928341 cases                0 2020-02
##  3 Afghanist~ AFG          Asia         38928341 cases                0 2020-03
##  4 Afghanist~ AFG          Asia         38928341 cases                0 2020-04
##  5 Afghanist~ AFG          Asia         38928341 cases                0 2020-05
##  6 Afghanist~ AFG          Asia         38928341 cases                0 2020-06
##  7 Afghanist~ AFG          Asia         38928341 cases                0 2020-07
##  8 Afghanist~ AFG          Asia         38928341 cases                0 2020-08
##  9 Afghanist~ AFG          Asia         38928341 cases                1 2020-09
## 10 Afghanist~ AFG          Asia         38928341 cases                3 2020-10
## # ... with 4 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>, count_by_population <dbl>
```

Create a training set using random number generator engine

```
set.seed(1234)
trainingset = COVID19_Dataset %>%
    sample_frac(0.05, replace = FALSE)
head(trainingset, n = 10)
```

```
## # A tibble: 10 x 10
##    country    country_code continent population indicator weekly_count year_week
##    <chr>      <chr>        <chr>           <dbl> <chr>            <dbl> <chr>
##  1 El Salvad~ SLV          America       6486201 cases              403 2020-19
##  2 Eswatini   SWZ          Africa        1160164 deaths              64 2020-53
##  3 Dominican~ DOM          America      10847904 deaths              14 2020-53
##  4 Ethiopia   ETH          Africa      114963583 cases            14626 2021-13
##  5 Trinidad ~ TTO          America       1399491 cases              461 2020-39
##  6 Gabon      GAB          Africa        2225728 deaths               1 2020-21
##  7 America (~ <NA>         America    1021703563 deaths           18680 2020-38
##  8 Mauritania MRT          Africa        4649660 deaths               1 2020-34
##  9 Honduras   HND          America       9904608 cases               99 2020-15
## 10 Anguilla   AIA          America         15002 deaths               0 2020-13
## # ... with 3 more variables: rate_14_day <dbl>, cumulative_count <dbl>,
## #   source <chr>
```

Print the summary statistics of your dataset

```
COVID19_Dataset %>%
    group_by(COVID19_Dataset$continent) %>%
    summarise_if(is.numeric, median, na.rm = TRUE)
```

```
## # A tibble: 5 x 5
##   `COVID19_Dataset$contine~ population weekly_count rate_14_day cumulative_count
##   <chr>                         <dbl>        <dbl>       <dbl>            <dbl>
## 1 Africa                     13132792           18       1.27              443
## 2 America                     2860840           22      11.5               232
## 3 Asia                       23816775           45       1.98             1068
## 4 Europe                      5457873          109      25.1              2016.
## 5 Oceania                      686878            0       0.112              23
```

Use any of the numerical variables from the dataset and perform the following statistical functions.

Mean • Median • Mode • Range

```
mode(COVID19_Dataset$cumulative_count)
```

```
## [1] "numeric"
```

```
range(COVID19_Dataset$cumulative_count)
```

```
## [1]        0 58946038
```

```
median(COVID19_Dataset$cumulative_count)
```

```
## [1] 682
```

```
mean(COVID19_Dataset$cumulative_count)
```

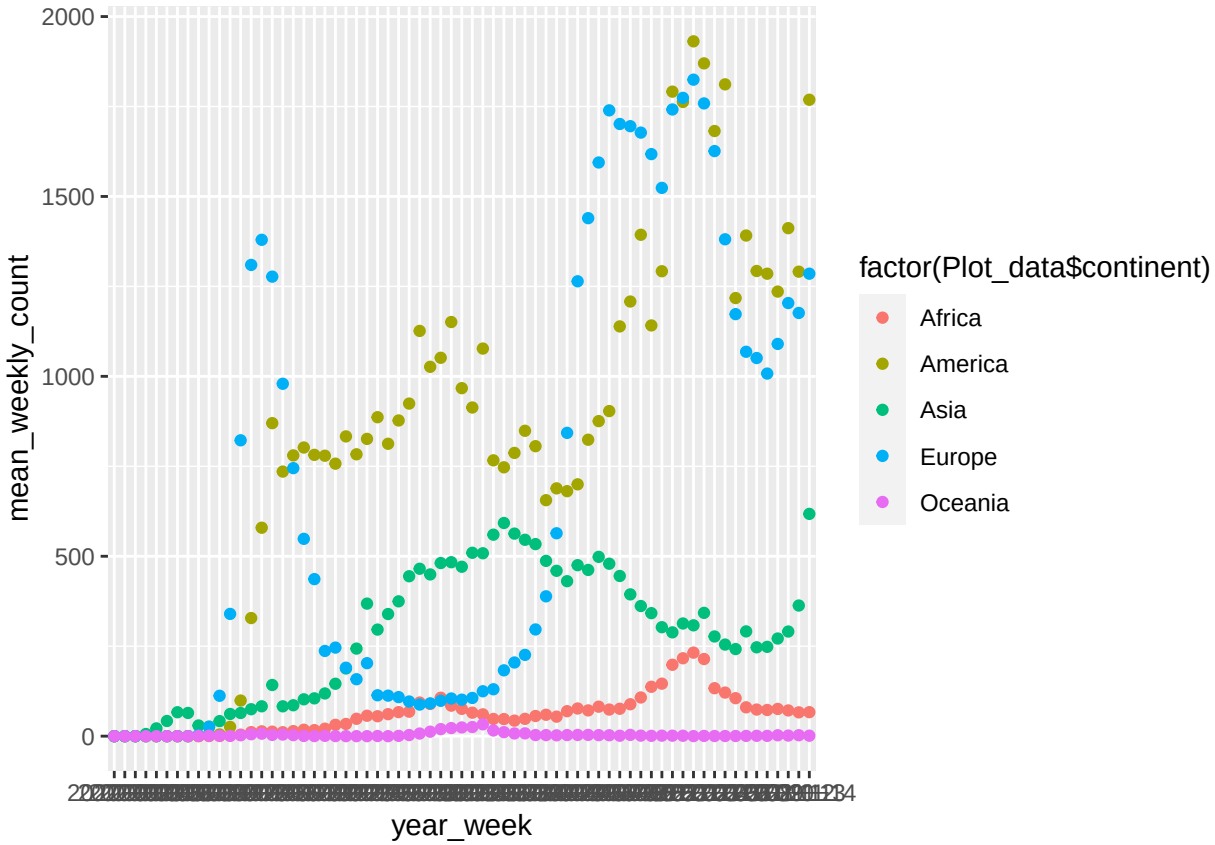
```
## [1] 238342.9
```

Plot a scatter plot for any 2 variables in your dataset.

```
my_function2 <- function(x) {
    output <- COVID19_Dataset %>%
        group_by(indicator, continent, year_week) %>%
        summarize(mean_weekly_count = mean(weekly_count, na.rm = TRUE)) %>%
        filter(indicator == x)
    return(output)
}

Plot_data = my_function2("deaths")
```

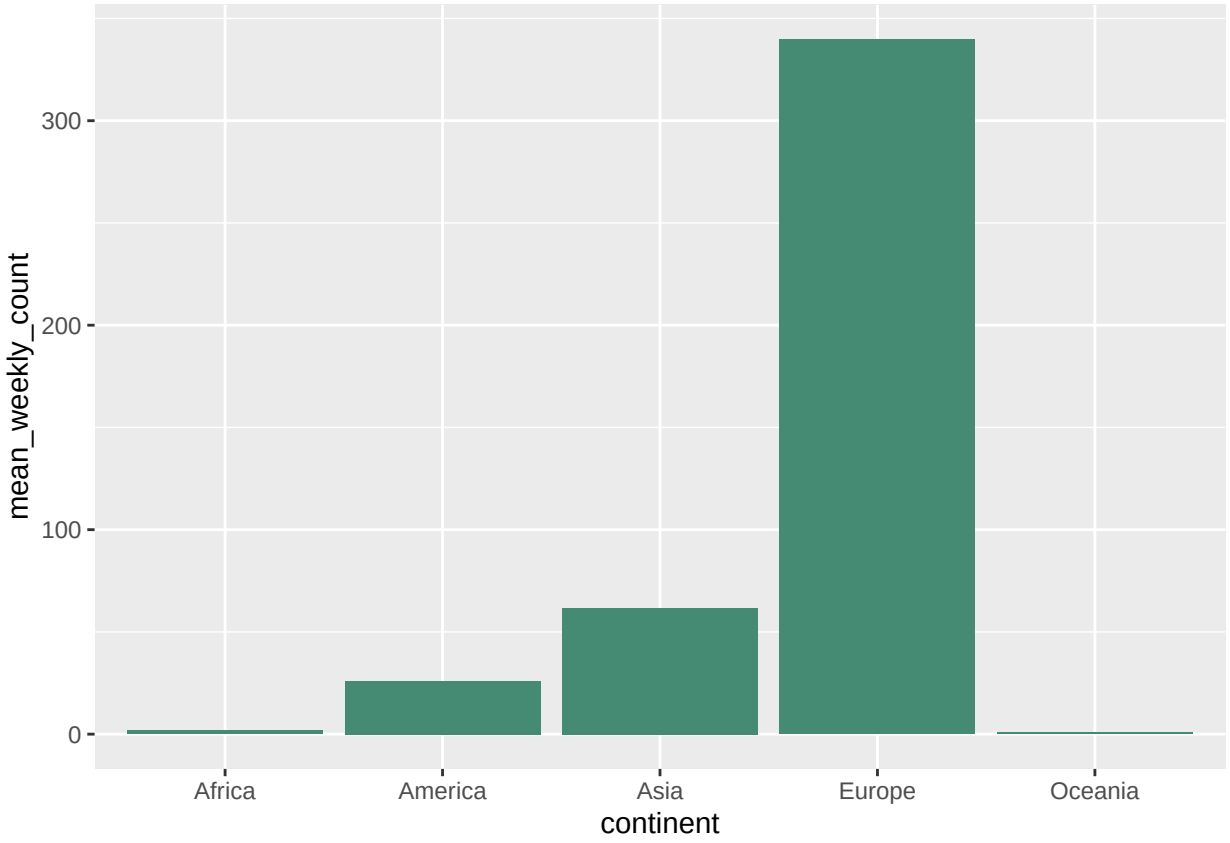
```
## `summarise()` has grouped output by 'indicator', 'continent'. You can override using the `.groups` a
```

```
ggplot(data = Plot_data, aes(x = year_week, y = mean_weekly_count,
    col = factor(Plot_data$continent))) + geom_point()
```



Plot a bar plot for any 2 variables in your dataset

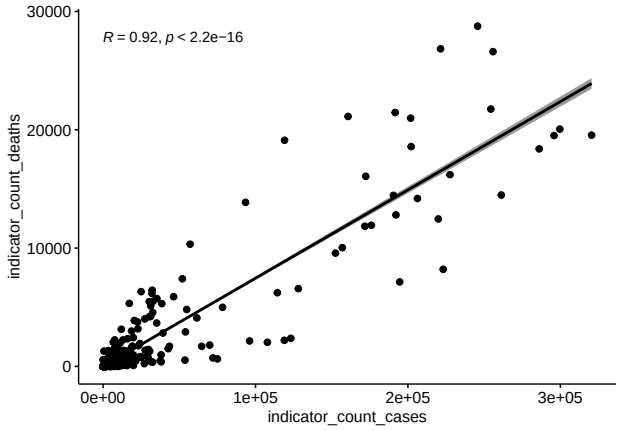
```
plot_data2 = Plot_data %>%
    filter(year_week == "2020-12")
ggplot(data = plot_data2, aes(continent, mean_weekly_count)) +
    geom_bar(stat = "identity", fill = "aquamarine4")
```



Find the correlation between any 2 variables by applying least square linear regression model

```
COVID19_Dataset_cases = COVID19_Dataset %>%
    filter(indicator == "cases", year_week == "2020-12" | year_week ==
        "2020-13" | year_week == "2020-14" | year_week == "2020-15" |
        year_week == "2020-16" | year_week == "2020-17" | year_week ==
        "2020-18" | year_week == "2020-19" | year_week == "2020-20")

COVID19_Dataset_cases_renamed = COVID19_Dataset_cases %>%
    rename(indicator_count_cases = weekly_count)

COVID19_Dataset_deaths = COVID19_Dataset %>%
    filter(indicator == "deaths", year_week == "2020-12" | year_week ==
        "2020-13" | year_week == "2020-14" | year_week == "2020-15" |
        year_week == "2020-16" | year_week == "2020-17" | year_week ==
        "2020-18" | year_week == "2020-19" | year_week == "2020-20")

COVID19_Dataset_deaths_renamed = COVID19_Dataset_deaths %>%
    rename(indicator_count_deaths = weekly_count)

new_data_set = cbind.data.frame(COVID19_Dataset_cases_renamed,
    COVID19_Dataset_deaths_renamed$indicator_count_deaths)
```

```
X = new_data_set[, "indicator_count_cases"]
Y = new_data_set[, "COVID19_Dataset_deaths_renamed$indicator_count_deaths"]
correlation = cor(Y, X, method = "pearson")
ggscatter(new_data_set, x = "indicator_count_cases", y = "COVID19_Dataset_deaths_renamed$indicator_coun
    add = "reg.line", conf.int = TRUE, cor.coef = TRUE, cor.method = "pearson",
    xlab = "indicator_count_cases", ylab = "indicator_count_deaths")
```




```
## Provide a conclusion of your analysis if any in the .RMD file

'''r
# According to our scatter plot, different continents have
# given different responses to Covid-19. For example, Oceania
# has the lowest death toll. On the flip side, Europe has
# completely different counts. It can easily be seen that, in
# the middle of Covid-19, Europe took some precautions, and
# decreased the deaths toll. Additionally, we can say that
# America has the highest death toll.

# When we look at the correlation between cases and deaths,
# we can say that two variable depend on each other. They
# have a high correlation according to the correlation
# graphic.At the 12 th week of 2020, every case almost ended
# up with a death. But after a while, 20 th week of 2021, the
# death toll is lower than case counts. We can say that
# during these times, people has increased their immune
# system.
```