Emission from Cereal Food Production

Nomvume Sandi

Introduction

You should write actual content to introduce the topic and the data.

Instead of importing individual packages, a better option would be to import the tidyverse package once.

```
#Importing data from Csv the emission data
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
emission_data <- read_csv("https://raw.githubusercontent.com/xrander/bootcamp-test/master/dat
```

```
Rows: 132 Columns: 5
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (3): entity, code, products
dbl (2): emission, per_capital_emission

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.
```

```
emission_data
```

```
# A tibble: 132 x 5
   entity    code  products       emission per_capital_emission
   <chr>     <chr> <chr>             <dbl>                <dbl>
 1 Australia AUS   Rice             879389.                0.524
 2 Australia AUS   Wheat             41497.                0.524
 3 Australia AUS   Other Cereals     89034.                0.524
 4 Austria   AUT   Rice             184118.                0.353
 5 Austria   AUT   Wheat             15495.                0.353
 6 Austria   AUT   Other Cereals     30146.                0.353
 7 Belgium   BEL   Rice             458813.                0.971
 8 Belgium   BEL   Wheat             50685.                0.971
 9 Belgium   BEL   Other Cereals     86200.                0.971
10 Brazil    BRA   Rice           10277208.                2.71
# i 122 more rows
```

```
#converting emission from tonnes to kilo tonnes(1000 tons = 1kilo tonnes)
emission_data |>
  mutate(
   kilo_emission =emission/1000)
```

```
# A tibble: 132 x 6
   entity    code  products       emission per_capital_emission kilo_emission
   <chr>     <chr> <chr>             <dbl>                <dbl>         <dbl>
 1 Australia AUS   Rice             879389.                0.524         879.
 2 Australia AUS   Wheat             41497.                0.524          41.5
 3 Australia AUS   Other Cereals     89034.                0.524          89.0
 4 Austria   AUT   Rice             184118.                0.353         184.
 5 Austria   AUT   Wheat             15495.                0.353          15.5
 6 Austria   AUT   Other Cereals     30146.                0.353          30.1
 7 Belgium   BEL   Rice             458813.                0.971         459.
 8 Belgium   BEL   Wheat             50685.                0.971          50.7
 9 Belgium   BEL   Other Cereals     86200.                0.971          86.2
10 Brazil    BRA   Rice           10277208.                2.71        10277.
# i 122 more rows
```

Exploratory data analysis for emission_data

```
#head(data) represents the top 6 rows of the emission_data quick snapshot
head(emission_data)
```

```
# A tibble: 6 x 5
  entity    code  products      emission per_capital_emission
  <chr>     <chr> <chr>            <dbl>                <dbl>
1 Australia AUS   Rice           879389.                0.524
2 Australia AUS   Wheat           41497.                0.524
3 Australia AUS   Other Cereals   89034.                0.524
4 Austria   AUT   Rice           184118.                0.353
5 Austria   AUT   Wheat           15495.                0.353
6 Austria   AUT   Other Cereals   30146.                0.353
```

You do not need to explain what the functions are for. Instead, explain the results

```
#tail(data) represents the bottom 6 rows of the emission data quick snapshot
tail(emission_data)
```

```
# A tibble: 6 x 5
  entity         code  products      emission per_capital_emission
  <chr>          <chr> <chr>            <dbl>                <dbl>
1 United Kingdom GBR   Rice           1279479.                0.391
2 United Kingdom GBR   Wheat           119004.                0.391
3 United Kingdom GBR   Other Cereals   248065.                0.391
4 United States  USA   Rice           4564029.                0.347
5 United States  USA   Wheat           761895.                0.347
6 United States  USA   Other Cereals 1027924.                0.347
```

```
#summary provides overview of the data , five number summary(min,max,1st qartile , 3rd quarti
```

```
summary(emission_data)
```

```
    entity              code             products           emission
 Length:132         Length:132         Length:132         Min.   :    1309
 Class :character   Class :character   Class :character   1st Qu.:   17533
 Mode  :character   Mode  :character   Mode  :character   Median :   79844
                                                          Mean   :  933255
```

```
                                                    3rd Qu.:  316456
                                                    Max.   :62291319
 per_capital_emission
 Min.   :0.06882
 1st Qu.:0.20271
 Median :0.38503
 Mean   :0.50759
 3rd Qu.:0.50162
 Max.   :2.77977
```

```
#provides a summary of the data frame's columns
str(emission_data)
```

```
spc_tbl_ [132 x 5] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ entity              : chr [1:132] "Australia" "Australia" "Australia" "Austria" ...
 $ code                : chr [1:132] "AUS" "AUS" "AUS" "AUT" ...
 $ products            : chr [1:132] "Rice" "Wheat" "Other Cereals" "Rice" ...
 $ emission            : num [1:132] 879389 41497 89034 184118 15495 ...
 $ per_capital_emission: num [1:132] 0.524 0.524 0.524 0.353 0.353 ...
 - attr(*, "spec")=
  .. cols(
  ..   entity = col_character(),
  ..   code = col_character(),
  ..   products = col_character(),
  ..   emission = col_double(),
  ..   per_capital_emission = col_double()
  .. )
 - attr(*, "problems")=<externalptr>
```

Visualisation

Total emissions of countries to plot (group the countries and sum up)

```
total_emissions <- emission_data|>
  group_by(code) |>
  summarize(total_emission = sum(emission, na.rm = TRUE))
print(total_emissions)
```

```
# A tibble: 44 x 2
```

```
   code  total_emission
   <chr>          <dbl>
 1 AUS         1009920.
 2 AUT          229760.
 3 BEL          595699.
 4 BGR           40871.
 5 BRA        15718377.
 6 CAN          882789.
 7 CHE          393396.
 8 CHN         6109723.
 9 CYP           25707.
10 CZE          158376.
# i 34 more rows
```

maximum and minimum of the total emission to only map them in a graph

```
top_20_emissions <- total_emissions |>
  slice_max(order_by = total_emissions, n = 20)
```

```
Bottom_20_emission <- total_emissions |>
  slice_min(order_by = total_emissions, n = 20)
```

```
print(top_20_emissions)
```

```
# A tibble: 20 x 2
   code  total_emission
   <chr>          <dbl>
 1 ZAF          567759.
 2 USA         6353848.
 3 TWN          901874.
 4 TUR          656277.
 5 SWE          270598.
 6 SVN           38107.
 7 SVK           98912.
 8 RUS         1128606.
 9 ROU          106113.
10 PRT          222558.
11 POL          398859.
12 NOR          244039.
13 NLD          799341.
14 MLT           14477.
```

```
15 MEX         5021201.
16 LVA           19240.
17 LUX           79637.
18 LTU           33607.
19 KOR         1473688.
20 JPN         2932992.
```

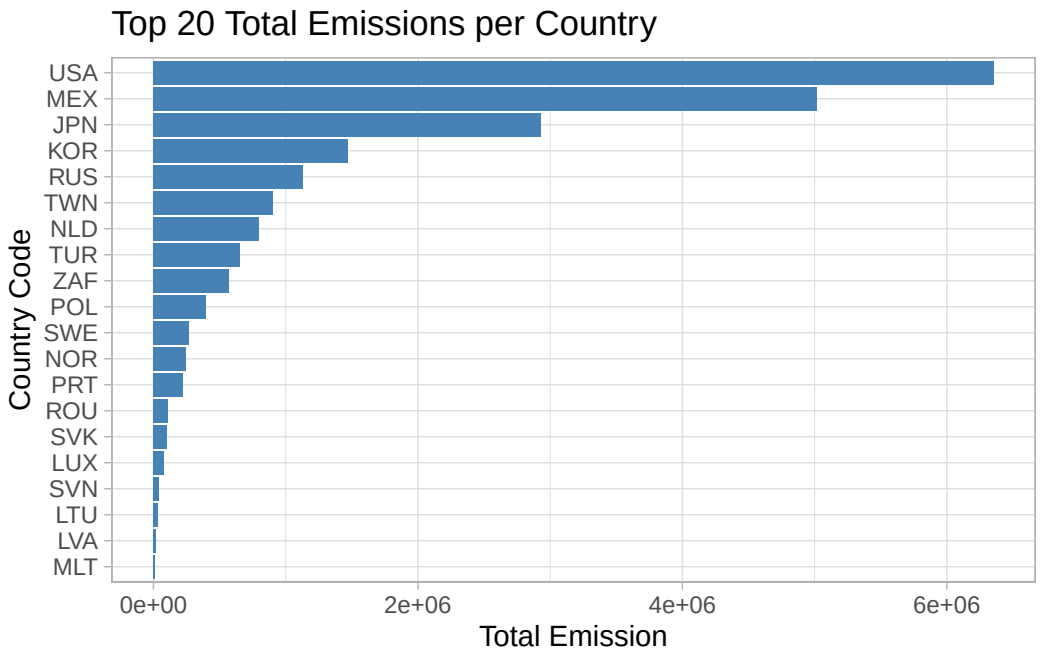
```r
print(Bottom_20_emission)
```

```
# A tibble: 20 x 2
   code  total_emission
   <chr>          <dbl>
 1 AUS          1009920.
 2 AUT           229760.
 3 BEL           595699.
 4 BGR            40871.
 5 BRA         15718377.
 6 CAN           882789.
 7 CHE           393396.
 8 CHN          6109723.
 9 CYP            25707.
10 CZE           158376.
11 DEU          1988006.
12 DNK           199789.
13 ESP          1196694.
14 EST            33110.
15 FIN           147278.
16 FRA          1885502.
17 GBR          1646549.
18 GRC           129023.
19 HRV            36099.
20 HUN            80175.
```

showing the top 20 emissions per country

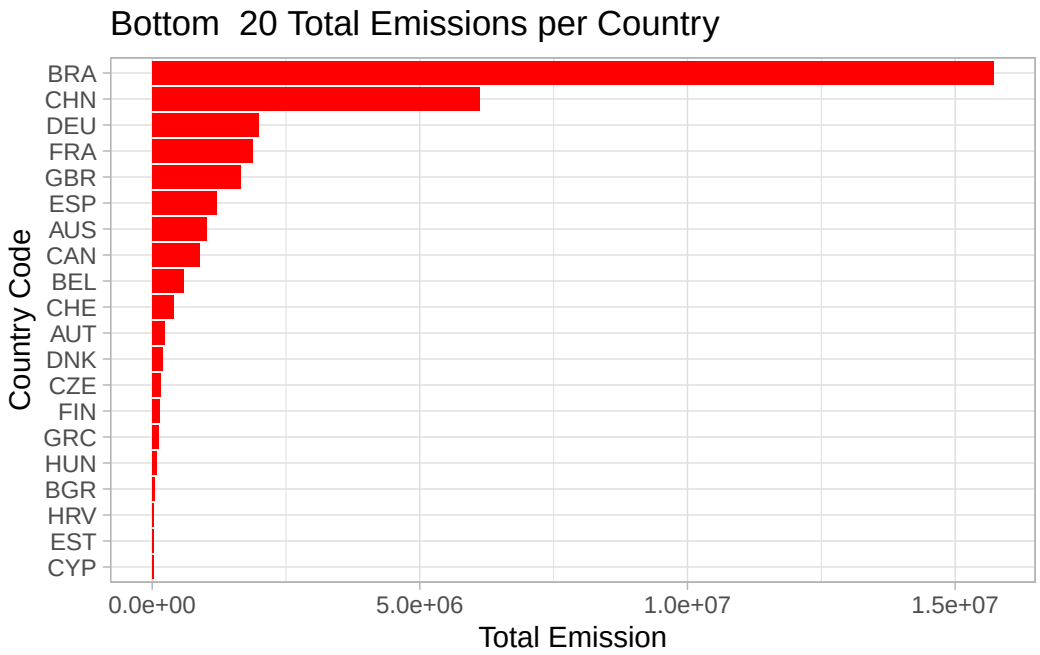
```r
ggplot(top_20_emissions, aes(x = reorder(code, total_emission), y = total_emission)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(
    x = "Country Code",
    y = "Total Emission",
    title = "Top 20 Total Emissions per Country"
```

```
  ) +
  theme_light() +
  coord_flip()
```



Bottom emissions

```
ggplot(Bottom_20_emission, aes(x = reorder(code, total_emission), y = total_emission)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(
    x = "Country Code",
    y = "Total Emission",
    title = "Bottom  20 Total Emissions per Country"
  ) +
  theme_light() +
  coord_flip()
```



Total emissions per product

```
product_emissions <- emission_data|>
  group_by(products) |>
  summarize(total_emission = sum(emission, na.rm = TRUE))
print(product_emissions)
```

```
# A tibble: 3 x 2
  products      total_emission
  <chr>                  <dbl>
1 Other Cereals       9637471.
2 Rice              101948790.
3 Wheat              11603388.
```

Sum of product contribution

```
ggplot(product_emissions, aes(x = products, y = total_emission, fill = products)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Total Emissions by Product",
    x = " By Product",
    y = "Total Emissions",
    fill = "Product"
```

```
  ) +
  theme_classic()
```