AN R EXPLORATORY DATA ANALYSIS PROJECT

CROPS PRODUCTION FERTILIZERS AND POPULATION

WADIA NORRI

27-08-2022

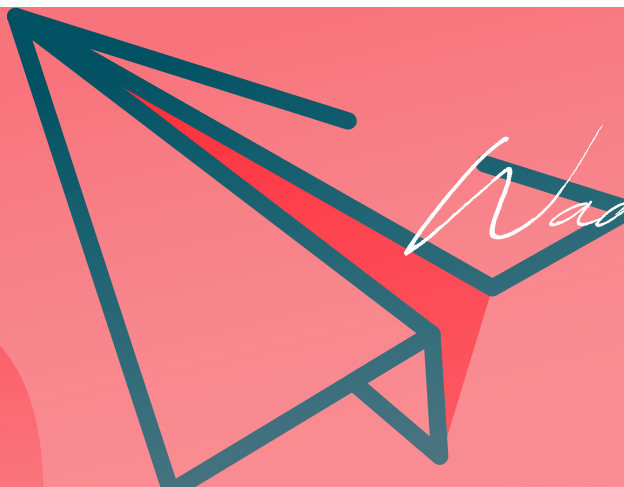
A WORD

The data used in this project was acquired from the Food and Agriculture Organization Of the United Nations And from NASA's Socioeconomic Data and Applications Center (sedac).

Accurate referencing of the data used in this project will be mentioned according to the appropriate regulations in the "references" section.

In this project, I will do an exploratory data analysis of different datasets using SQL, R, and Tableau.

The objective of such a project is to showcase my skills as a data analyst, my best practices, communication skills and to share my knowledge.



Nadia Norri

PROJECT PLAN

PROJECT PROPOSAL

CROPS PRODUCTION FERTILIZERS AND POPULATION

Summary

The objective of this EDA is to analyze the data on hand to draw insights, find patterns, and derive conclusions to be more informed and insightful when taking action regarding such an archaic yet sophisticated subject and economical sector of Agriculture.
In this project, I will be making use of the statistical programing language R mainly to manipulate the data through the different phases or processes of Data Analysis as well as I will be using SQL to view and query the data on the Cloud using the BigQuery server-less data warehouse and finally apart from the visualizations that we will make on R, we will be also using Tableau as a mean of visualization.
All of these tools will be employed in order to answer questions and make data-driven actions.

Things To Keep In Mind

The approach used in this project is divided and based on the six Data Analysis Processes Ask, Prepare, Process, Analyze, Share and Act.
Usually, I will share which **Analytical Skills**, **Analytical Aspects Of Thinking**, and in which stage the data is, regarding its **Data Life Cycle** in order to give insights to the reader on how to think analytically following a set of steps, but in this reading, I will not be sharing this information as I already did before on my ***Bike Theft In Toronto Analysis -Excel-***.
The data used in this project ROCCCs meaning it's Reliable, Original, Comprehensive, Current, and Cited. but since the data goes back to the 60s and include data-poor countries, some of the data can be :

- Aggregated may include official, semi-official, estimated, or calculated data
- Calculated data
- Not available
- FAO(Food and Agriculture Organization Of the United Nations) data based on imputation methodology
- FAO estimate
- Official data
- Unofficial figure

Although the data is gathered by reputable organizations the providers of such data is generally the countries themselves.
to know more about the data and the methodology used to gather the data check the refrences section.

PHILOSOPHY

Weird... isn't it, the universe is so vast, so big, so far-reaching, so... infinite you might say !? well, we could argue that.

The universe is only theoretically infinite. You see if we stop time right now the amount of objects, materials, and atoms in this universe are countable meaning it's finite but due to its vastness it doesn't matter how much time you have, as you will never be able to count it, and in our human way to reconcile ourselves, we call it infinite, take for example an ant, for the little puny ant, our blue planet is definitely infinite, yet we humans the dominant specie of this planet we know it's finite, we know we can only use much land, we know we can only drill much oil, we know we can only grow much food, yet we act like our small blue planet is infinite the same way the little puny ant thinks of it. But the fact is the universe is finite there's a big number out there of how many tons of materials are in the universe, the fact is Earth is finite it weights about 5.972 × 10~24 kg (give or take how many pizzas I ate) but the thing that is infinite weirdly enough it's life, it's people, it's their dreams, their innovations, their discoveries. Growth in the population shouldn't be looked at as a bad thing but rather as a great thing, with new people born new possibilities are also born but the key to success as a specie is to manage our finite resources in a proper way rather than think of it as the same way the ant thinks of this planet, infinite.

And the most important resource one would say is human innovation, turning imagination into reality, bringing ideas from a metaphysical realm to the physical world but for one to think, innovate, or research he needs to eat and drink, the basic needs for life to exist is the same basic needs for innovation to be created.

In other words, food and water is the fuel for all human discoveries. So let's discover the way we grow food today.

DATA ANALYSIS PROCESSES

1. ASK
2. PREPARE
3. PROCESS
4. ANALYZE
5. SHARE
6. ACT
7. REFERENCES

1.ASK:

Topic :

We are exploring crop production, fertilizer usage, and population growth and their impacts on each other.

Stakeholders :

This kind of analysis will be useful to environmentalists, people interested in demography, economists, and investors in the sector of agriculture such as farming companies, food companies or farmers.

Establishing metrics :

in this part, we will set some of the questions that we may need to answer :

- what is the average for each variable?
- What are the top 20 prominent crops by TOTAL production?
- What are the top 5 prominent crops by average production?
- What are the top 5 crops that use the biggest amount of area from 1961 till 2020?
- What are the top 5 crops that yield the highest?
- What are the top 5 crops that yield the lowest?
- What is the relation between production and the area of harvest?
- What is the normalized yield for each crop?
- What are the Trend of area harvested and production over time?
- What is the Change in area harvested between 2000 and 2020?
- Who are the top 10 countries that produce the most prominent crops By average production in the last 10 years?
- Distribution of fertilizers For the Top 10 Countries By Production
- What is the relationship between fertilizer usage and population?

2.PREPARE:

Data Sources :

In this EDA we will be using multiple datasets open to the public from multiple accredited sources.
the data sources are :

- Food and Agriculture Organization of the United Nations
- United Nations Population Division

Why these datasets :

The datasets need to contain variables that will allow us to answer our questions concerning crops, amount of land used by the crops, yield, fertilizers usage, and demographics of countries.
Moreover, the datasets should have different data format types such as continuous data or nominal data and it needs to be structured.
In this EDA we will focus more on quantitative data rather than on qualitative data.

Data Lifecycle :

Since we are mixing and matching different datasets, I will create a database on Google's BigQuery where we will store, organize, and fusion the different datasets.

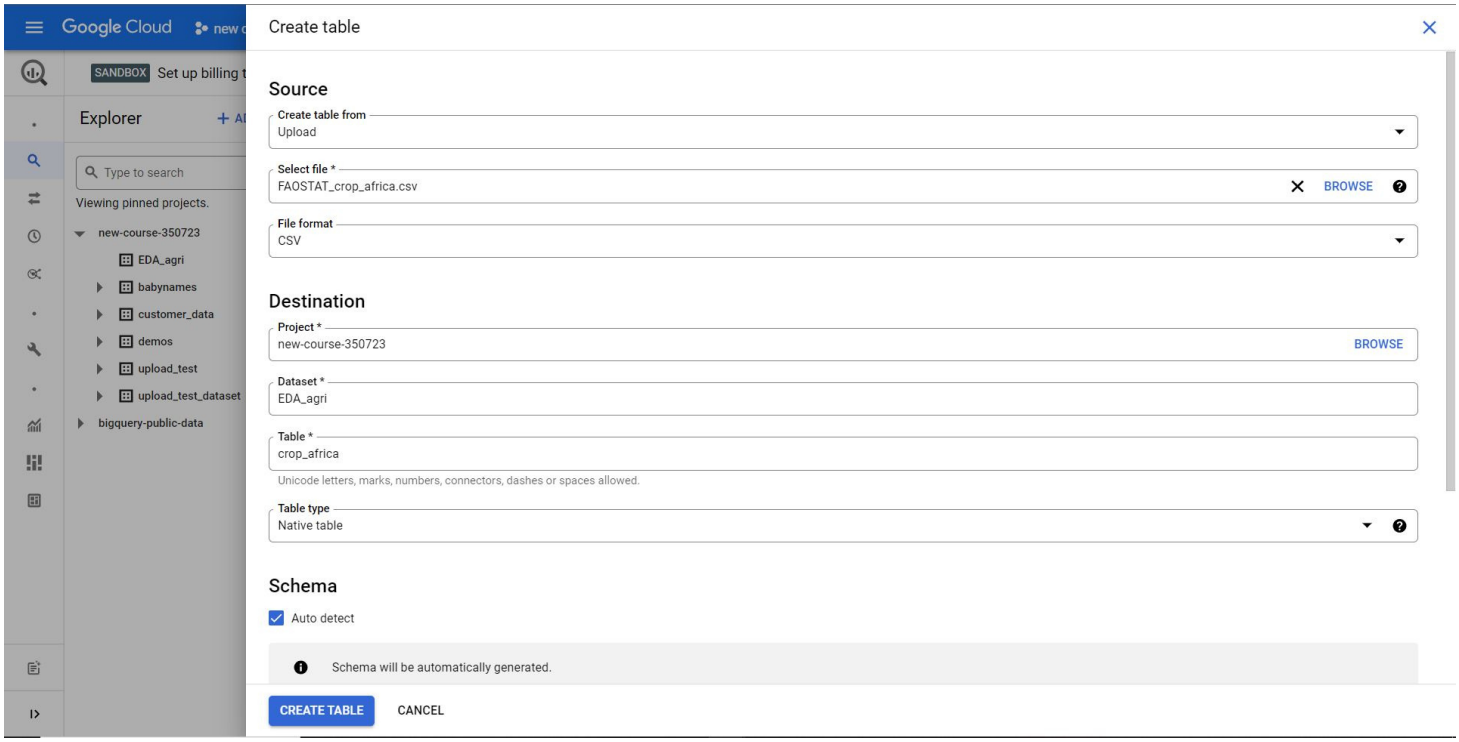
Uploading & understanding the data :

We will create a new dataset on BigQuery and then create tables for the different datasets.

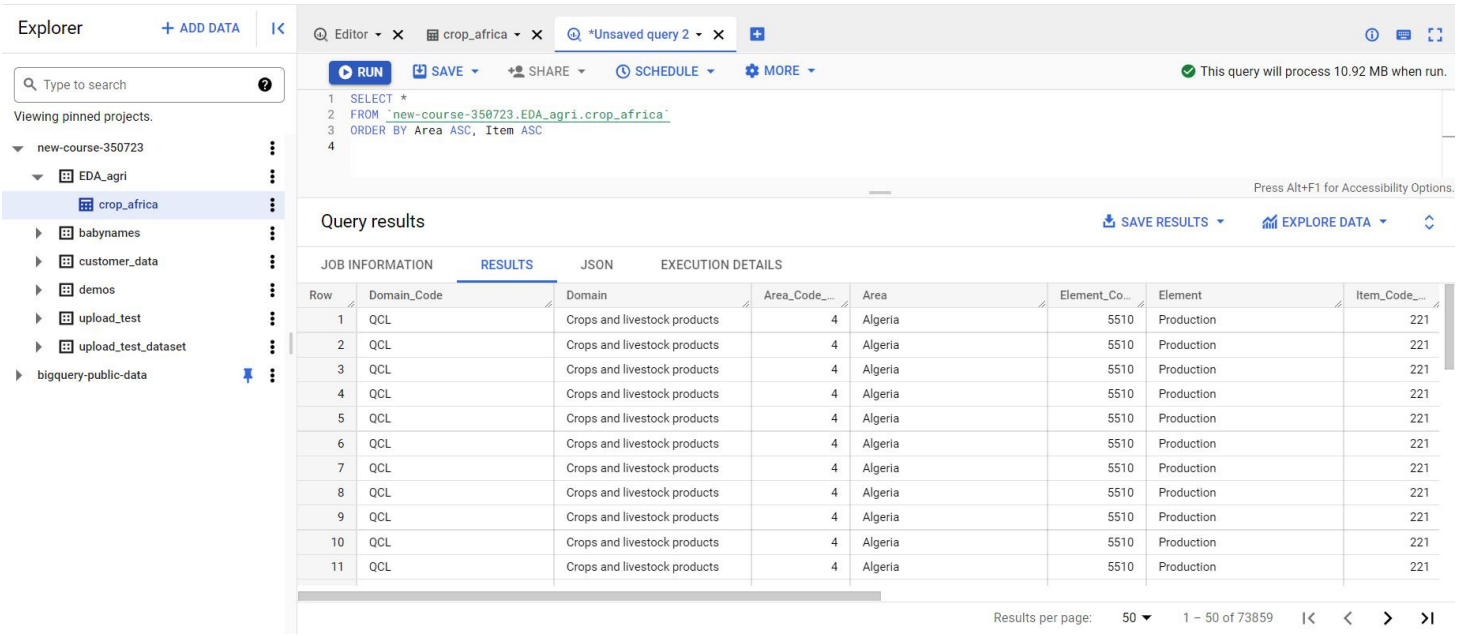
I could not download the dataset of the crops by "Crops Primary" and select all "Countries" and all available "Years" since the FAO website won't allow it stating that *'the selection is too large'*. In some cases even when choosing a subset of data by "Continent" it is deemed too large.

The maneuver around this problem is to download multiple subsets of data by subsets of the continent and then merge them, rather than download a bulk zip file containing all data about not only crops but also data on crops processed, live animals, livestock, and a lot more.

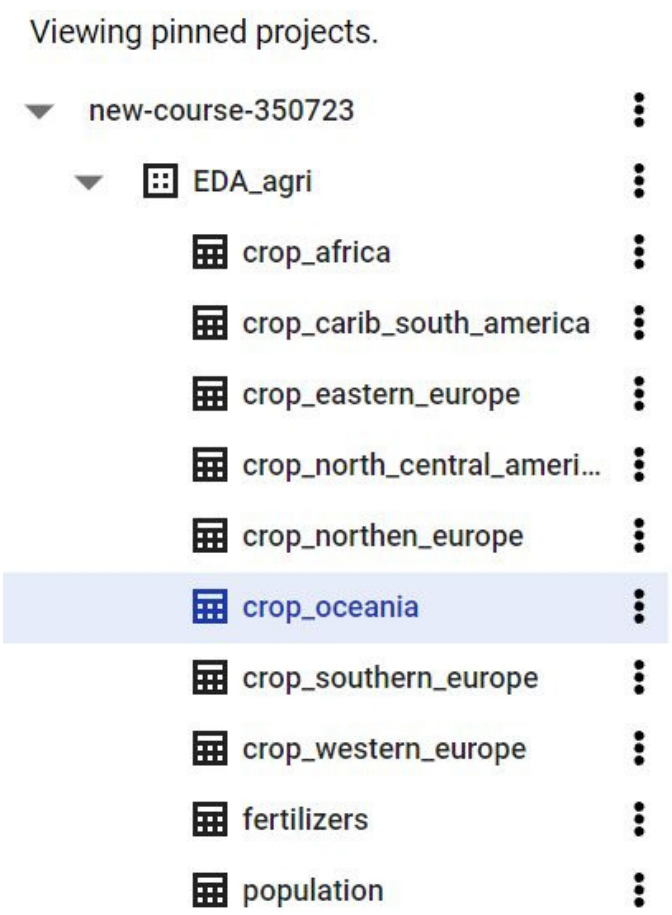
we will save the tables under the newly created dataset **EDA_agri**. The format of the files we are uploading is "CSV", the schema is set to Auto detect.



Querying the data we can see that all rows and columns were uploaded and in the right format *-see schema page-*.



We upload the other datasets.

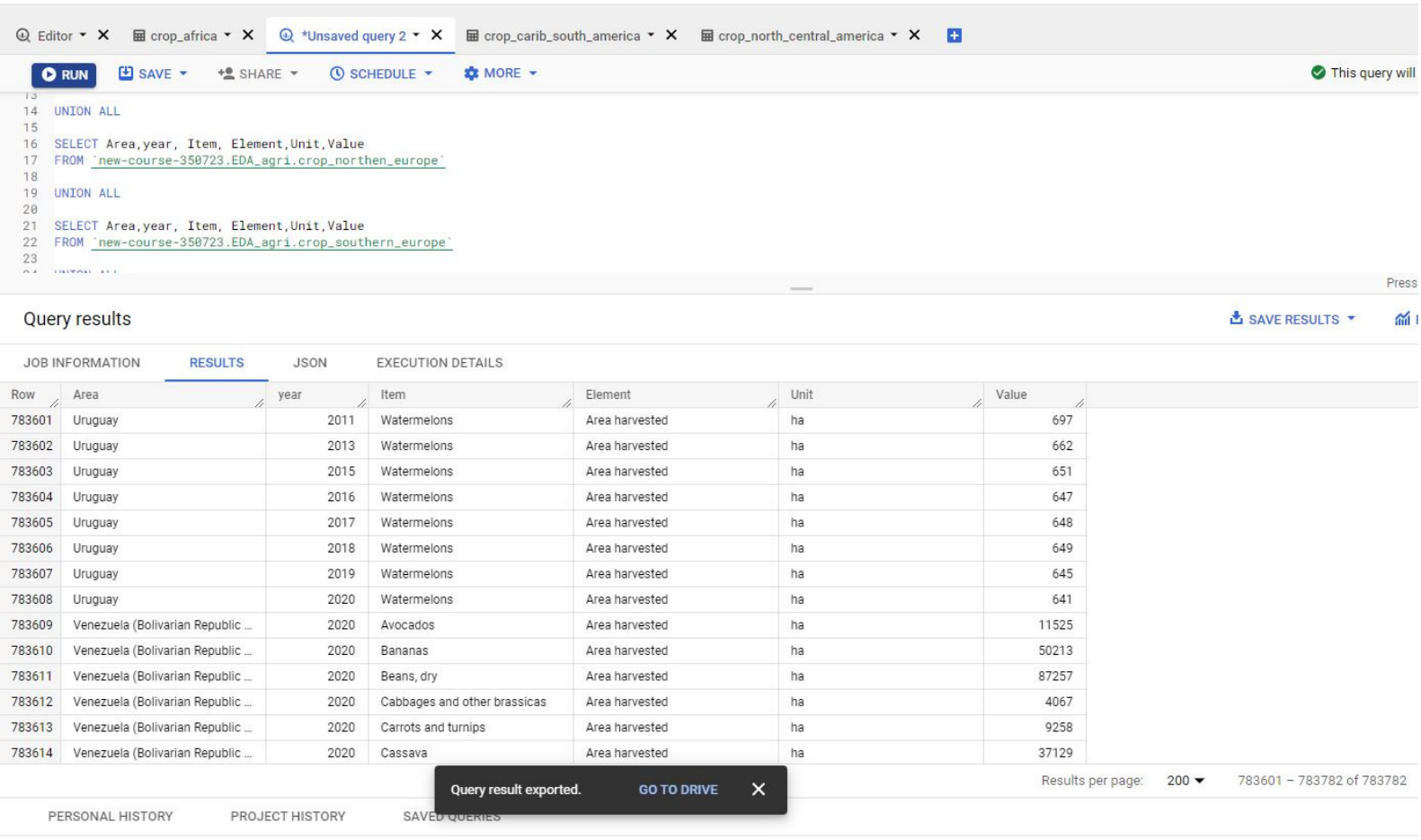


Usually, we would simply go to the next process funny enough it's the Process phase, but since we still need to merge the crop datasets into one there is still a job to do.
I could've done that using R but I won't be able to show off my SQL skills -hehe- moreover connecting to BigQuery using R is doable and easy using the **"bigquery"** and **"dplyr"** libraries but since the minimum billing for a query to run on R is 10MB even if the query is actually only some kbs I'm going to be billed for a whole 10MB which is bad a deal considering I have a free account, other than that some functions might not work right or not work at all.
The approach here is to fuse the data on BigQuery export it and work with it on R.

We merge the data using UNION ALL rather than JOINS since they aren't related by anything.

```
SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_africa`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_oceania`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_eastern_europe`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_northen_europe`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_southern_europe`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_western_europe`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_carib_south_america`

UNION ALL

SELECT Area,year, Item, Element,Unit,Value
FROM `new-course-350723.EDA_agri.crop_north_central_america`
```

The results of such query is 45.78MB of processed data



Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	Area	year	Item	Element	Unit	Value
783601	Uruguay	2011	Watermelons	Area harvested	ha	697
783602	Uruguay	2013	Watermelons	Area harvested	ha	662
783603	Uruguay	2015	Watermelons	Area harvested	ha	651
783604	Uruguay	2016	Watermelons	Area harvested	ha	647
783605	Uruguay	2017	Watermelons	Area harvested	ha	648
783606	Uruguay	2018	Watermelons	Area harvested	ha	649
783607	Uruguay	2019	Watermelons	Area harvested	ha	645
783608	Uruguay	2020	Watermelons	Area harvested	ha	641
783609	Venezuela (Bolivarian Republic ...	2020	Avocados	Area harvested	ha	11525
783610	Venezuela (Bolivarian Republic ...	2020	Bananas	Area harvested	ha	50213
783611	Venezuela (Bolivarian Republic ...	2020	Beans, dry	Area harvested	ha	87257
783612	Venezuela (Bolivarian Republic ...	2020	Cabbages and other brassicas	Area harvested	ha	4067
783613	Venezuela (Bolivarian Republic ...	2020	Carrots and turnips	Area harvested	ha	9258
783614	Venezuela (Bolivarian Republic ...	2020	Cassava	Area harvested	ha	37129

Query result exported. GO TO DRIVE

Results per page: 200 783601 – 783782 of 783782

PERSONAL HISTORY | PROJECT HISTORY | SAVED QUERIES

Datasets Report:

FULL_CROP_EDITED_DATA:

- Area: country
- year: the year of each Element
- Element : a simple group by shows there are 3 elements Area harvested, Production, and Yield for most items.
- Item : Crop name.
- Unit : the type of measurement regarding the Value of an Element.
- Value : numeric representation of the Element.

FERTILIZERS:

- Country Name
- Country Code
- Indicator Name: Fertilizer consumption (% of fertilizer production).
- Indicator Code: Code for the indicator.
- [1960-2021]: numeric representation of usage.

POPULATION:

- Country Name
- Country Code
- Indicator Name: Population, total.
- Indicator Code: Code for the indicator.
- [1960-2021]: numeric representation of the population.

Now we have every country, its area harvested, yield, and production of every possible crop by year!
We have a dataset but it still needs cleaning, the same goes for the fertilizers and population datasets.

3.PROCESS:

Tool used for cleaning :

The tool I used for cleaning all the data sets is the programing language **R** using **RStudio** as an IDE.

Why R ? :

Aside from being one of the most popular tools among the data science community, R is simply powerful and I emphasize "simply" although it has its own learning curve, it pays big by writing a simple line of code you can clean, visualize, or analyze the data, with a convenient and huge library of ready to use tools, being statistically focused, open source and its adaptability for both machine learning and analysis project R is a powerful tool.

Data integrity :

As mentioned before, the data used is historical data going as back as 1961 even if it was the year we sent the first human to space that's still before France outlawed the death penalty by guillotine in other words some regions or countries didn't log any information about their agriculture production, fertilizer's usage or even their demographics. So there are a lot of missing values in every data set. We will try to analyze or approach the data in three ways, the first being a total approach of aggregated measurement for the whole world regardless of missing values. The second way is to set a timeframe where we have the least number of missing values-even then we might still have missing values-. The third or final approach is to choose a handful of countries with no missing values -it is the same approach NASA's **SEDAC** once took regarding this matter in its **Twentieth Century Crop Statistics, v1 (1900–2017)** ***Food Security***- Apart from what's brought up above, we will check for the right Data types, Data range, mandatory values, duplicates, and more.

Cleaning The Data

In this part we will be cleaning the data, making sure to uphold a high standard for data integrity as well as manipulating the data in order to make it more organized and easier to read.
Clean data aligned with business objectives equals an accurate conclusion.
Using R allows us to replicate the procedures for manipulating, cleaning, and analyzing the data thus making the analysis trustworthy, the full steps will be available as an R Markdown file or/and R file.

R libraries used


```
library(tidyverse)
library(skimr)
library(naniar)
library(DataExplorer)
library(janitor)
library(stringdist)
```

FULL_CROP_EDITED_DATA:

Uploading the data :

```
df<-read.csv(file = 'C:/path')
```

Or

```
df<-read.csv(choose.file( ))
```

Understanding the data :

```
str(df)
glimpse(df)
skim_without_charts(df)
```

```
> str(df)
'data.frame':   783782 obs. of  6 variables:
 $ Area   : chr  "Austria" "Austria" "Austria" "Austria" ...
 $ year   : int  1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 ...
 $ Item   : chr  "Apples" "Apples" "Apples" "Apples" ...
 $ Element: chr  "Production" "Production" "Production" "Production" ...
 $ Unit   : chr  "tonnes" "tonnes" "tonnes" "tonnes" ...
 $ Value  : int  456000 431000 436000 447000 222000 366654 360937 328675 327690 308809 ...
> glimpse(df)
Rows: 783,782
Columns: 6
$ Area    <chr> "Austria", "Austria", "Austria", "Austria", "Austria", "Austria", "Austria", "Austria", "Austria", "Aus…
$ year    <int> 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 1970, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1…
$ Item    <chr> "Apples", "Apples", "Apples", "Apples", "Apples", "Apples", "Apples", "Apples", "Apples", "Apples", "Ap…
$ Element <chr> "Production", "Production", "Production", "Production", "Production", "Production", "Production", "Prod…
$ Unit    <chr> "tonnes", "tonnes", "tonnes", "tonnes", "tonnes", "tonnes", "tonnes", "tonnes", "tonnes", "tonnes", "to…
$ Value   <int> 456000, 431000, 436000, 447000, 222000, 366654, 360937, 328675, 327690, 308809, 243658, 156358, 287462,…
> skim_without_charts(df)
── Data Summary ────────────────────────
                           Values
Name                       df
Number of rows             783782
Number of columns          6
_____________________
Column type frequency:
  character                4
  numeric                  2
_____________________
Group variables            None

── Variable type: character ──────────────────────────────────────────────────────────────────────────────────────────
  skim_variable n_missing complete_rate min max empty n_unique whitespace
1 Area                  0             1   4  52     0      157          0
2 Item                  0             1   3  44     0      161          0
3 Element               0             1   5  14     0        3          0
4 Unit                  0             1   2   6     0        3          0

── Variable type: numeric ────────────────────────────────────────────────────────────────────────────────────────────
  skim_variable n_missing complete_rate    mean        sd   p0  p25   p50    p75      p100
1 year                  0         1       1995.      17.6 1961 1980  1997   2011       2020
2 Value             68290         0.913 312963. 4804303.     0 2090 16600 85103. 768594154
>
```

We have 783,782 rows with 6 columns so no data loss during export from Big Query.

Data types:

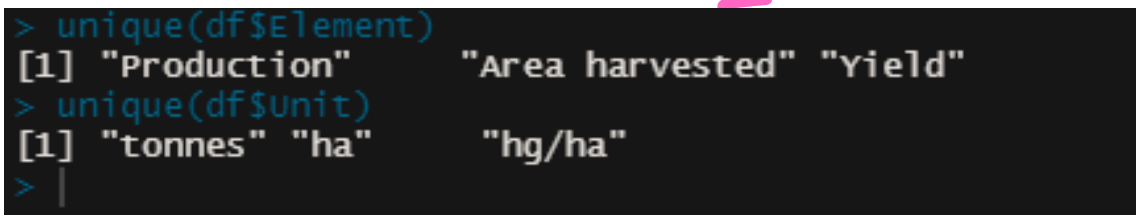
- Area: Character <right>
- Item : Character <right>
- Element : Character <right>
- Unit : Character <right>
- year : integer <right> *since it's only the year as like YYYY there's is no need to format it to date format*
- value : Character <right>

we can make use of most of the insights given by the *skim_without_charts()* function but in the "variable type: numeric" we can't take the calculations done on the variable Value as right since the value represents 3 unique Elements Production, Area harvested, and Yield and each one of these has a uniquely different type of Unit.

Except for n_missing values and complete rate where we have for Value:

- n_missing: 68290 entries missing
- complete_rate : 0.913

```
unique(df$Element)
unique(df$Unit)
```



Although the year isn't in the right format type we can still make use of the insight we have :

- p0: the oldest year in the data set is 1961
- p100: the latest year in the data set is 2020

The challenge :

The challenge with this data lies in :

- the values are mixed together: the values can either be a representation of Production, Area harvested, or Yield. Making manipulating the data rough and intuitive since the value doesn't only represent an Element but represents a combination of variables **Value = Area + year+ Item+Element+Unit.** Hence we need to divide the Elements into 3 new variables while keeping the representation of the Value of the other variables true. By way of explanation, we need to make the data wider. This will allow us to manipulate the data easier, analyze it more efficiently and compare variables through different means of calculations.

The solution :

The steps of the solution for our problem are:

1. Create 3 data sets from the original one based on each Element type.
2. Remove the Element & Unit from each data set we made and rename the Value column as the unique name of the Element we based our creation on.
3. Sort the data in an intuitive way for future usage.
4. To make sure we didn't lose any data combination before merging we make each data set of these 3 have all possible combinations.
5. We combine the 3 data sets into one with their complete possible combinations giving us a data set with **1,469,520 entries, and 6 total columns.**
6. With a complete data set, we have a lot of rows with multiple missing values that were not in the original one to combat this we make a function that deletes any row with more than 2 missing values since a row should at least represent 1 of 3 variables if not it means it was made during our process

```r
## create 3 data sets by dividing and sorting df by groups
area<-df %>% filter(df$Element == "Area harvested") %>%
  subset(select = -c(Element,Unit)) %>%
  rename("Area harvested"="Value") %>%
  arrange(Area,Item,year)
View(area)

prod<-df %>% filter(df$Element == "Production") %>%
  subset(select = -c(Element,Unit)) %>%
  rename("Production"="Value") %>%
  arrange(Area,Item,year)

yield<-df %>% filter(df$Element == "Yield") %>%
  subset(select = -c(Element,Unit)) %>%
  rename("Yield"="Value") %>%
  arrange(Area,Item,year)

## merge the data sets into one while keeping na values
yield<- complete(yield,Area,year,Item)##yield with all na
prod<- complete(prod,Area,year,Item)## prod with all na
yie_prod<-merge(yield,prod,by=c("Area","Item","year")) ## merging them
crop_na<-complete(area,Area,year,Item) %>% inner_join(.,yie_prod) ## adding area with nas and merging everything
View(crop_na)
```

This leaves us with the data set "crop_na" apart from the missing values we had originally on the "FULL_CROP_EDITED_DATA" data set, we made some more missing values while using the complete function, to combat this we make a function that deletes any row with more than 2 missing values since a row should at least represent 1 of 3 variables if not it means it was made during our process.

```r
##Cleaning rows with more than 2 nas in the production,Area harvested, and Yield
  # creating a function for that
delete.na <- function(DF, n=0) {
  DF[rowSums(is.na(DF)) <= n,]
}

crop_organized<-delete.na(crop_na,2)
View(crop_organized)
crop_organized<-arrange(crop_organized,Area,Item,year)
str(crop_organized)
crop_organized<-crop_organized[,c(1,2,3,4,6,5)]
str(crop_organized)
View(crop_organized)
```

- Now we have a data set named "crop_organized" with 6 columns and 248,331 rows.

	Area	year	Item	Element	Unit	Value
1	Austria	1961	Apples	Production	tonnes	456000
2	Austria	1962	Apples	Production	tonnes	431000
3	Austria	1963	Apples	Production	tonnes	436000
4	Austria	1964	Apples	Production	tonnes	447000
5	Austria	1965	Apples	Production	tonnes	222000
6	Austria	1966	Apples	Production	tonnes	366654
7	Austria					
8	Austria					
9	Austria					
10	Austria					
11	Austria					
12	Austria					
13	Austria					
14	Austria					
15	Austria					
16	Austria					
17	Austria					
18	Austria					
19	Austria					
20	Austria					

	Area	year	Item	Area harvested	Yield	Production
1	Albania	1961	Agave fibres nes	NA	NA	NA
2	Albania	1961	Almonds, with shell	NA	NA	NA
3	Albania	1961	Anise, badian, fennel, coriander	NA	NA	NA
4	Albania	1961	Apples	NA	NA	10004
5	Albania	1961	Apricots	NA	NA	900
6	Albania	1961	Artichokes	NA	NA	NA
7	Albania	1961	Asparagus	NA	NA	NA
8	Albania	1961	Avocados	NA	NA	NA
9	Albania	1961	Bambara beans	NA	NA	NA
10	Albania	1961	Bananas	NA	NA	NA
11	Albania	1961	Barley	8734	10224	8930
12	Albania	1961	Bastfib			
13	Albania	1961	Beans,			
14	Albania	1961	Beans,			
15	Albania	1961	Berries			
16	Albania	1961	Blueber			
17	Albania	1961	Brazil n			
18	Albania	1961	Broad b			
19	Albania	1961	Buckwh			
20	Albania	1961	Cabbag			

	Area	year	Item	Area harvested	Production	Yield
1	Albania	1961	Apples	NA	10004	NA
2	Albania	1962	Apples	NA	8039	NA
3	Albania	1963	Apples	NA	7931	NA
4	Albania	1964	Apples	NA	8498	NA
5	Albania	1965	Apples	NA	8000	NA
6	Albania	1966	Apples	NA	9000	NA
7	Albania	1967	Apples	NA	10000	NA
8	Albania	1968	Apples	NA	10000	NA
9	Albania	1969	Apples	NA	11000	NA
10	Albania	1970	Apples	NA	9000	NA
11	Albania	1971	Apples	NA	10000	NA
12	Albania	1972	Apples	NA	11000	NA
13	Albania	1973	Apples	NA	14000	NA
14	Albania	1974	Apples	NA	13000	NA
15	Albania	1975	Apples	NA	14000	NA
16	Albania	1976	Apples	NA	15000	NA
17	Albania	1977	Apples	NA	13000	NA
18	Albania	1978	Apples	NA	14000	NA
19	Albania	1979	Apples	NA	15000	NA
20	Albania	1980	Apples	NA	15000	NA
21	Albania	1981	Apples	NA	15000	NA
22	Albania	1982	Apples	NA	23000	NA
23	Albania	1983	Apples	NA	21000	NA
24	Albania	1984	Apples	NA	21000	NA
25	Albania	1985	Apples	3600	18200	50556
26	Albania	1986	Apples	3000	15000	50000
27	Albania	1987	Apples	3000	16000	53333
28	Albania	1988	Apples	2800	14000	50000
29	Albania	1989	Apples	3500	18000	51429
30	Albania	1990	Apples	2600	13100	50385
31	Albania	1991	Apples	2000	10000	50000
32	Albania	1992	Apples	2033	10000	49188

We did delete the "Unit" column so to give some context about measurement I will rename the newly created categories like this :

- Production -> Production (tonnes)
- Area harvested -> Area harvested (ha)
- Yield -> Yield (tonnes/ha) - it's in tonne/ha instead of the original measurement hg/ha because I want to change from hg to tonnes <1 hg=0.0001 t> making the variable more usable in our future analysis.

```
crop_organized<-rename(crop_organized, "Production(tonnes)"="Production",
                       "Area harvested(ha)"="Area harvested",
                       "Yield(tonnes/ha)"="Yield")
```

To actually change the measurement of the Yield(tonnes/ha) to an actual tonnes/ha *-tonne is metric while ton is imperial-* measurement we simply multiply the columns by 0.0001.

crop_organized$`Yield(tonnes/ha)` <-
crop_organized$`Yield(tonnes/ha)`*0.0001

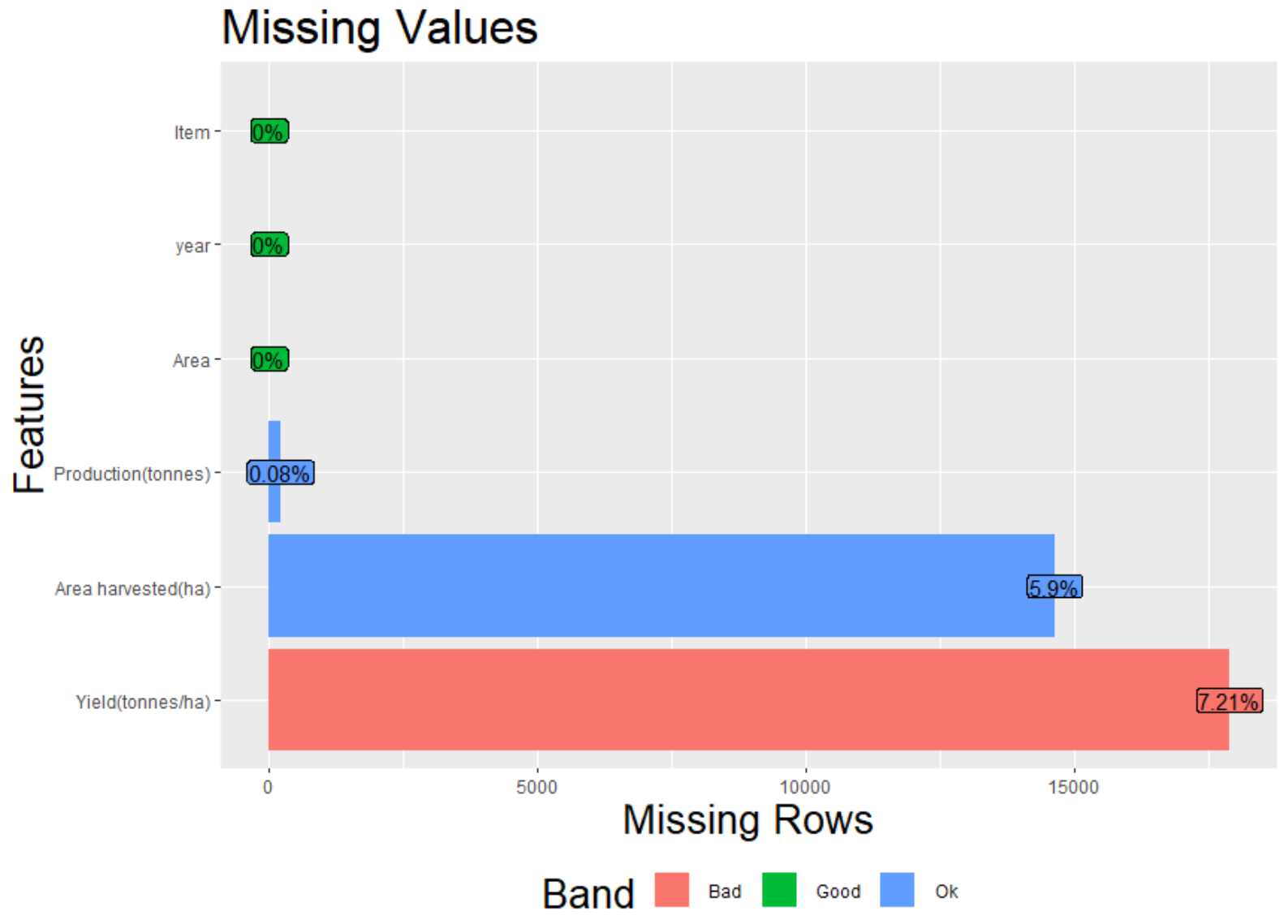
Using the skimr library we can have an apercu of the data frame.

```
> skim(crop_organized)
── Data Summary ────────────────────────
                           Values
Name                       crop_organized
Number of rows             248331
Number of columns          6
_______________________
Column type frequency:
  character                2
  numeric                  4
_______________________
Group variables            None

── Variable type: character ──────────────────────────────────────────────────────────────
  skim_variable n_missing complete_rate min max empty n_unique whitespace
1 Area                  0             1   4  52     0      157          0
2 Item                  0             1   3  44     0      156          0

── Variable type: numeric ────────────────────────────────────────────────────────────────
  skim_variable      n_missing complete_rate     mean       sd   p0     p25      p50     p75
1 year                   0         1       1995.      17.7 1961 1980     1998     2011
2 Area harvested(ha)     14644         0.941 135912.  1251286.     0  400     2903    20140.
3 Production(tonnes)       207         0.999 656196.  8040581.     0 1578    13588    98251.
4 Yield(tonnes/ha)       17902         0.928     12.1      50.1    0    1.71     5.30    12.8
       p100 hist
1      2020 ▂▂▂▂▇
2  70205008 ▇▁▁▁▁
3 768594154 ▇▁▁▁▁
4      5085. ▇▁▁▁▁
```

Visualize the missing values:

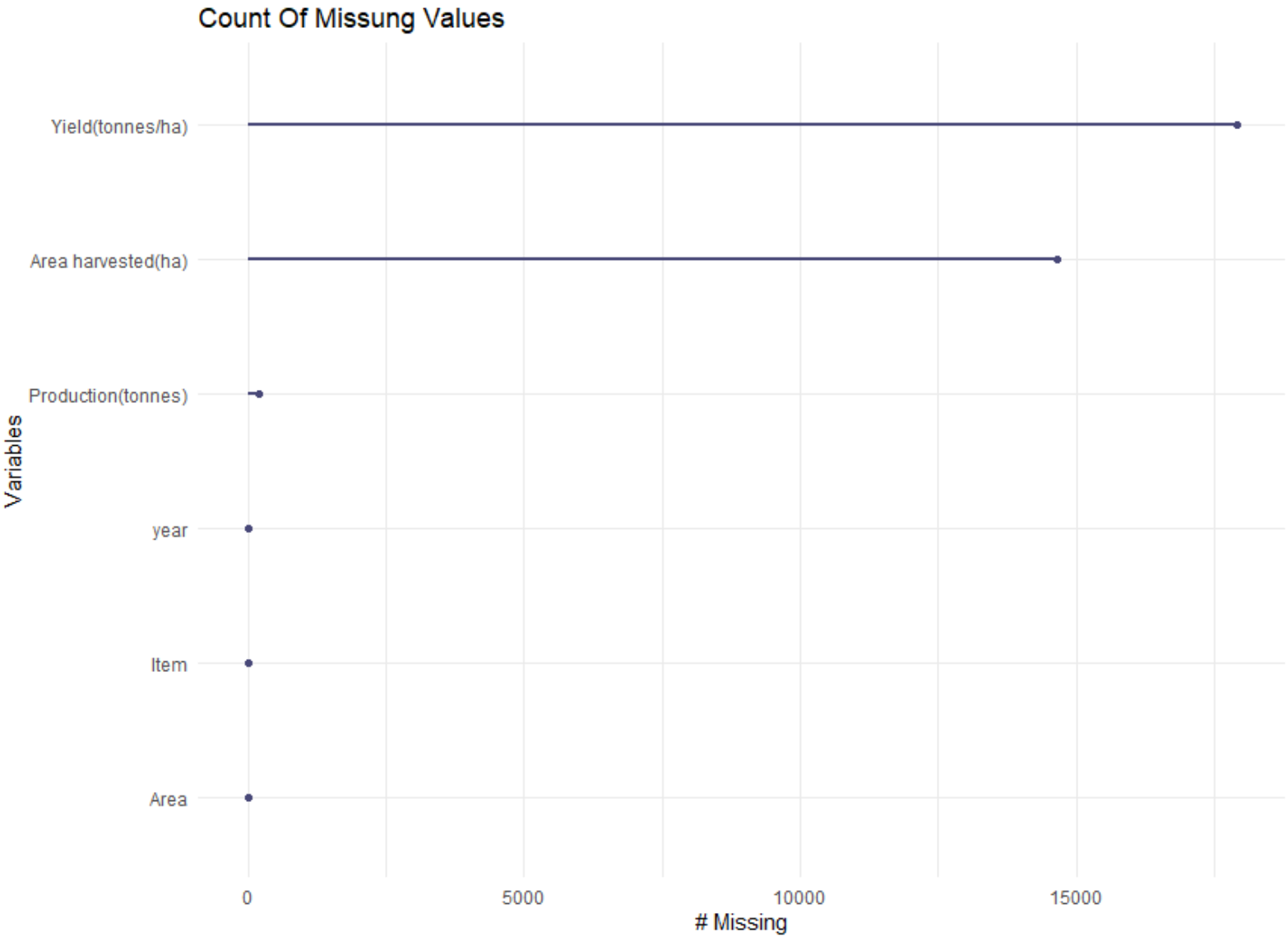
- Yield(tonnes/ha) has the highest amount of missing values at 7.21% of NAs of it's column. Still it's acceptable.

```
## visualizing missing values
plot_missing(
  crop_organized,
  group = list("Good" = 0, "Ok" = 0.06, "Bad" = 1),
  missing_only = FALSE,
  geom_label_args = list("size" = 4, "label.padding" = unit(0.1, "lines")),
  title = 'Missing Values',
  ggtheme = theme(title = element_text(size = 20)),
  theme_config = list(legend.position=c("bottom"))
  )
```



- Another way is visualizing the count of missing values per column, this lets us know the number of missing values where we can see about more than 17500 entries are missing from the Yield(tonnes/ha) but since we have 248,331 entries we can work with it.

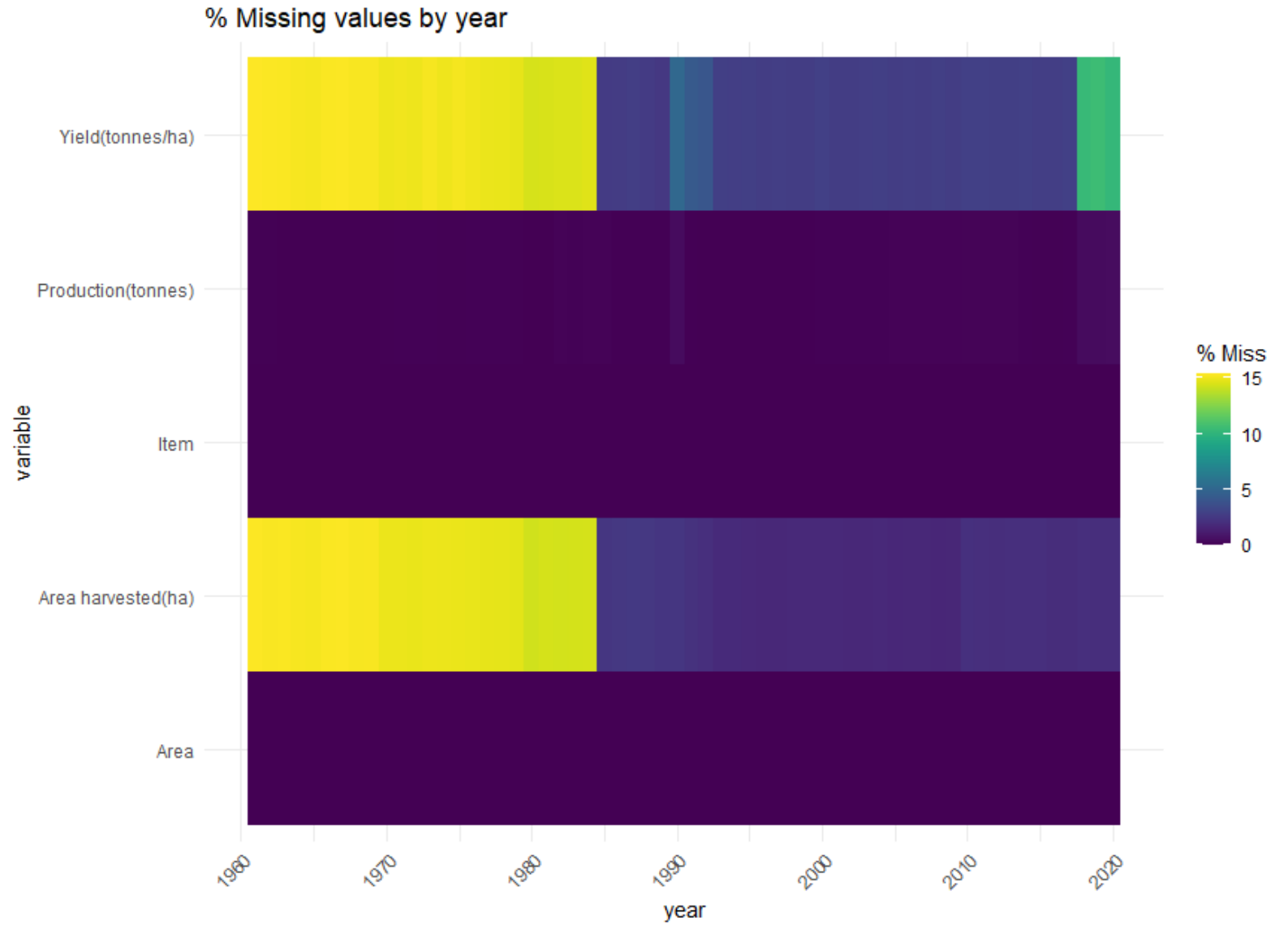
```
gg_miss_var(crop_organized)+labs(title = "Count Of Missung Values")
```



Visualization by heatmap shows the percent of missing values by year, and we can conclude from it that :

- Most missing values are between 1961 and 1985

```
gg_miss_fct(crop_organized, year)+
  ggtitle("% Missing values by year")+
  scale_x_continuous(breaks = scales::breaks_extended(n = 10))
```



Unique values:

- Checking unique values in each column.

```
## unique values
unique_values <- rapply(crop_organized,function(x) length(unique(x)))
unique_values
```

Area	year	Item	Area harvested(ha)	Production(tonnes)	Yield(tonnes/ha)
157	60	156	56739	93174	115364

- To check for each unique value by column we can use for example: unique(crop_organized$Area)

```
[15] "Bolivia (Plurinational State of)"
[17] "Botswana"
[19] "Bulgaria"
[21] "Burundi"
[23] "Cameroon"
[25] "Central African Republic"
[27] "Chile"
[29] "Comoros"
[31] "Cook Islands"
[33] "Côte d'Ivoire"
[35] "Cuba"
[37] "Czechoslovakia"
[39] "Denmark"
[41] "Dominica"
[43] "Ecuador"
[45] "El Salvador"
[47] "Eritrea"
[49] "Eswatini"
[51] "Faroe Islands"
[53] "Finland"
[55] "French Guyana"
[57] "Gabon"
[59] "Germany"
[61] "Greece"
[63] "Guadeloupe"
[65] "Guinea"
[67] "Guyana"
[69] "Honduras"
[71] "Iceland"
[73] "Italy"
[75] "Kenya"
[77] "Latvia"
[79] "Liberia"
[81] "Lithuania"
[83] "Madagascar"
[85] "Mali"
[87] "Marshall Islands"
[89] "Mauritania"
[91] "Mexico"
[93] "Montenegro"
[95] "Mozambique"
[97] "Nauru"
"Bosnia and Herzegovina"
"Brazil"
"Burkina Faso"
"Cabo Verde"
"Canada"
"Chad"
"Colombia"
"Congo"
"Costa Rica"
"Croatia"
"Czechia"
"Democratic Republic of the Congo"
"Djibouti"
"Dominican Republic"
"Egypt"
"Equatorial Guinea"
"Estonia"
"Ethiopia"
"Fiji"
"France"
"French Polynesia"
"Gambia"
"Ghana"
"Grenada"
"Guatemala"
"Guinea-Bissau"
"Haiti"
"Hungary"
"Ireland"
"Jamaica"
"Kiribati"
"Lesotho"
"Libya"
"Luxembourg"
"Malawi"
"Malta"
"Martinique"
"Mauritius"
"Micronesia (Federated States of)"
"Morocco"
"Namibia"
"Netherlands"
```

- Or to check for the types of crops we are dealing with :

```
> unique(crop_organized$Item)
  [1] "Apples"                                  "Apricots"
  [3] "Barley"                                  "Beans, dry"
  [5] "Beans, green"                            "Broad beans, horse beans, dry"
  [7] "Cabbages and other brassicas"            "Carrots and turnips"
  [9] "Cauliflowers and broccoli"               "Cherries"
 [11] "Cherries, sour"                          "Chestnut"
 [13] "Chillies and peppers, green"             "Cucumbers and gherkins"
 [15] "Dates"                                   "Eggplants (aubergines)"
 [17] "Figs"                                    "Fruit, citrus nes"
 [19] "Fruit, fresh nes"                        "Garlic"
 [21] "Grapes"                                  "Hops"
 [23] "Leeks, other alliaceous vegetables"      "Lemons and limes"
 [25] "Lettuce and chicory"                     "Maize"
 [27] "Melons, other (inc.cantaloupes)"         "Mushrooms and truffles"
 [29] "Nuts nes"                                "Oats"
 [31] "Okra"                                    "Olives"
 [33] "Onions, dry"                             "Onions, shallots, green"
 [35] "Oranges"                                 "Peaches and nectarines"
 [37] "Pears"                                   "Peas, green"
 [39] "Plums and sloes"                         "Potatoes"
 [41] "Pulses nes"                              "Pumpkins, squash and gourds"
 [43] "Quinces"                                 "Rice, paddy"
 [45] "Rye"                                     "Seed cotton"
 [47] "Sorghum"                                 "Soybeans"
 [49] "Spices nes"                              "Spinach"
 [51] "Strawberries"                            "Sugar beet"
 [53] "Sunflower seed"                          "Tangerines, mandarins, clementines, satsumas"
 [55] "Tobacco, unmanufactured"                 "Tomatoes"
 [57] "Vegetables, fresh nes"                   "Vegetables, leguminous nes"
 [59] "Vetches"                                 "Watermelons"
 [61] "Wheat"                                   "Almonds, with shell"
 [63] "Artichokes"                              "Bananas"
 [65] "Carobs"                                  "Chick peas"
 [67] "Chillies and peppers, dry"               "Fruit, stone nes"
 [69] "Grapefruit (inc. pomelos)"               "Groundnuts, with shell"
 [71] "Lentils"                                 "Peas, dry"
 [73] "Rapeseed"                                "Triticale"
 [75] "Bastfibres, other"                       "Cashew nuts, with shell"
 [77] "Cassava"                                 "Castor oil seed"
 [79] "Cocoa, beans"                            "Coffee, green"
 [81] "Millet"                                  "Oil palm fruit"
 [83] "Pineapples"                              "Sesame seed"
 [85] "Sisal"                                   "Sugar cane"
 [87] "Sweet potatoes"                          "Fruit, tropical fresh nes"
 [89] "Mangoes, mangosteens, guavas"            "Taro (cocoyam)"
 [91] "Yams"                                    "Anise, badian, fennel, coriander"
 [93] "Asparagus"                               "Avocados"
 [95] "Canary seed"                             "Cereals nes"
 [97] "Fibre crops nes"                         "Flax fibre and tow"
 [99] "Linseed"                                 "Lupins"
[101] "Maté"                                    "Oilseeds nes"
[103] "Papayas"                                 "Peppermint"
[105] "Safflower seed"                          "String beans"
[107] "Tea"                                     "Tung nuts"
[109] "Walnuts, with shell"                     "Berries nes"
```

Duplicates

- To check for duplicate rows we can use :.
  `sum(duplicated(crop_organized,fromLast = TRUE))`

It returns a zero, meaning we have no duplicate rows at all. the *duplicated* function returns TRUE or FALSE since a TRUE = 1, the sum of that would say how many duplicates we have.

With this our " FULL_CROP_EDITED_DATA" is processed into a clean & ready to be analyzed data frame called "crop_organized".

We still have 2 more datasets to process and clean, but I will only show one more here and pass through the used functions we did before to make this reading short! *-the full code will be available as an R Markdown file-*

FERTILIZERS:

Again we start by uploading our data and exploring it

Uploading the data :

```
fert<-read.csv(file = 'C:/path')
```

Or

```
fert<-read.csv(choose.file( ))
```

Understanding the data :

```
str(fert)
glimpse(fert)
skim_without_charts(fert)
```

```
> str(fert)
'data.frame':   266 obs. of  66 variables:
 $ Country.Name  : chr  "Aruba" "Africa Eastern and Southern" "Afghanistan" "Africa Western and Centra
 $ Country.Code  : chr  "ABW" "AFE" "AFG" "AFW" ...
 $ Indicator.Name: chr  "Fertilizer consumption (kilograms per hectare of arable land)" "Fertilizer co
r hectare of arable land)" "Fertilizer consumption (kilograms per hectare of arable land)" "Fertilizer
 per hectare of arable land)" ...
 $ Indicator.Code: chr  "AG.CON.FERT.ZS" "AG.CON.FERT.ZS" "AG.CON.FERT.ZS" "AG.CON.FERT.ZS" ...
 $ X1960         : logi  NA NA NA NA NA NA ...
 $ X1961         : num  NA NA 0.144 NA 0.375 ...
 $ X1962         : num  NA NA 0.143 NA 0.37 ...
 $ X1963         : num  NA NA 0.142 NA 0.515 ...
 $ X1964         : num  NA NA 0.141 NA 0.764 ...
 $ X1965         : num  NA NA 0.141 NA 1.444 ...
 $ X1966         : num  NA NA 0.191 0.984 1.857 ...
 $ X1967         : num  NA 11.28 1.27 1.14 2.37 ...
 $ X1968         : num  NA 11.85 1.91 1.01 2.66 ...
 $ X1969         : num  NA 12.24 2.163 0.929 3.621 ...
 $ X1970         : num  NA 13.65 2.47 1.24 3.9 ...
 $ X1971         : num  NA 15.47 2.59 1.38 8.62 ...
 $ X1972         : num  NA 16.63 3.68 1.78 7.69 ...
 $ X1973         : num  NA 17.01 3.11 1.71 7.38 ...
 $ X1974         : num  NA 17.48 4.29 2.42 5.21 ...
 $ X1975         : num  NA 17.82 4.6 3.11 1.41 ...
 $ X1976         : num  NA 18.414 5.592 3.55 0.724 ...
 $ X1977         : num  NA 19.4 6.86 4.66 10.9 ...
 $ X1978         : num  NA 19.93 6.78 4.3 8.1 ...
 $ X1979         : num  NA NA 6.29 4.98 4.72 ...
 $ X1980         : num  NA 24.26 6.46 5.62 5.79 ...
 $ X1981         : num  NA 27.27 5.78 7.22 4.17 ...
 $ X1982         : num  NA 26.63 6.67 6.73 1.72 ...
 $ X1983         : num  NA 22.46 7.15 8.09 2.97 ...
 $ X1984         : num  NA 21.53 9.18 7.99 2.41 ...
 $ X1985         : num  NA 20.57 9.22 6.81 7.01 ...
 $ X1986         : num  NA 19.9 8.51 6.42 4.14 ...
 $ X1987         : num  NA 18.9 9.99 6.09 3.55 ...
 $ X1988         : num  NA 20.81 7.05 7.18 5.49 ...
 $ X1989         : num  NA 20.08 7.03 7.65 8.14 ...
 $ X1990         : num  NA 19.72 5.63 8.05 3.28 ...
 $ X1991         : num  NA 19.31 6.14 8.09 2.34 ...
 $ X1992         : num  NA 18.81 5.79 8.38 3.03 ...
 $ X1993         : num  NA 20.7 5.11 8.8 2.67 ...
 $ X1994         : num  NA 19.74 NA 6.85 3.33 ...
 $ X1995         : num  NA 19.25 NA 5.46 2.67 ...
 $ X1996         : num  NA 22.02 0.654 5.422 2 ...
 $ X1997         : num  NA 22.355 0.651 6.009 0.667 ...
 $ X1998         : num  NA 20.346 0.902 6.078 1.133 ...
 $ X1999         : num  NA 20.365 0.653 6.261 1.133 ...
 $ X2000         : num  NA 19.484 0.651 5.935 0.467 ...
 $ X2001         : num  NA 19.62 2.39 6.24 NA ...
 $ X2002         : num  NA 23.39 3.19 NA 1.66 ...
 $ X2003         : num  NA 18.68 3.48 6.59 1.79 ...
 $ X2004         : num  NA 19.11 4.24 6.49 4.5 ...
 $ X2005         : num  NA 17.06 3.81 6.73 2.26 ...
 $ X2006         : num  NA 19.98 3.19 8.67 3.66 ...
 $ X2007         : num  NA 20.64 2.1 5.63 3.31 ...
 $ X2008         : num  NA 20.5 1.85 6.1 8.26 ...
```

The Challenges :

With this data frame, we have different types of problems to solve.

- When compared to " crop_organized" this data frame is much wider, column names are not clean. The naming scheme for country names does not only mismatch the country names we have in our "crop_organized" data frame but it adds to it by mentioning new groups of the countries and subcategories making the whole data frame confusing and hard to work with, another column related problem is the years naming where every year has a prefix before it (x1960,x1961,x1962,...).

```
> unique_values_fert <- rapply(fert,function(x) length(unique(x)))
> unique_values_fert
  Country.Name   Country.Code Indicator.Name Indicator.Code          X1960          X1961          X1962          X1963
           266            266              1              1              1            127            128            128
```

```
> unique_values_crop <- rapply(crop_organized,function(x) length(unique(x)))
> unique_values_crop
          Area           year           Item Area harvested(ha) Production(tonnes)   Yield(tonnes/ha)
           157             60            156          56739          93174         115364
```

Check the mismatching values :

- Using the *setdiff()* function.

```
setdiff(fert$Country.Name,crop_organized$Area)
```

We get 139 mismatching values. Here are some :

```
 [1] "Aruba"                                          "Africa Eastern and Southern"
 [3] "Afghanistan"                                    "Africa Western and Central"
 [5] "Andorra"                                        "Arab World"
 [7] "United Arab Emirates"                           "Armenia"
 [9] "American Samoa"                                 "Azerbaijan"
[11] "Bangladesh"                                     "Bahrain"
[13] "Bahamas, The"                                   "Bermuda"
[15] "Bolivia"                                        "Brunei Darussalam"
[17] "Bhutan"                                         "Central Europe and the Baltics"
[19] "Channel Islands"                                "China"
[21] "Cote d'Ivoire"                                  "Congo, Dem. Rep."
[23] "Congo, Rep."                                    "Caribbean small states"
[25] "Curacao"                                        "Cayman Islands"
[27] "Cyprus"                                         "Czech Republic"
[29] "East Asia & Pacific (excluding high income)"    "Early-demographic dividend"
[31] "East Asia & Pacific"                            "Europe & Central Asia (excluding high income)"
[33] "Europe & Central Asia"                          "Egypt, Arab Rep."
[35] "Euro area"                                      "European Union"
[37] "Fragile and conflict affected situations"       "Micronesia, Fed. Sts."
[39] "United Kingdom"                                 "Georgia"
[41] "Gibraltar"                                      "Gambia, The"
[43] "Greenland"                                      "Guam"
[45] "High income"                                    "Hong Kong SAR, China"
[47] "Heavily indebted poor countries (HIPC)"         "IBRD only"
[49] "IDA & IBRD total"                               "IDA total"
[51] "IDA blend"                                      "Indonesia"
[53] "IDA only"                                       "Isle of Man"
[55] "India"                                          "Not classified"
[57] "Iran, Islamic Rep."                             "Iraq"
[59] "Israel"                                         "Jordan"
[61] "Japan"                                          "Kazakhstan"
[63] "Kyrgyz Republic"                                "Cambodia"
[65] "St. Kitts and Nevis"                            "Korea, Rep."
[67] "Kuwait"                                         "Latin America & Caribbean (excluding high income)"
[69] "Lao PDR"                                        "Lebanon"
[71] "St. Lucia"                                      "Latin America & Caribbean"
[73] "Least developed countries: UN classification"   "Low income"
[75] "Liechtenstein"                                  "Sri Lanka"
[77] "Lower middle income"                            "Low & middle income"
[79] "Late-demographic dividend"                      "Macao SAR, China"
[81] "St. Martin (French part)"                       "Monaco"
[83] "Moldova"                                        "Maldives"
[85] "Middle East & North Africa"                     "Middle income"
[87] "Myanmar"                                        "Middle East & North Africa (excluding high income)"
[89] "Mongolia"                                       "Northern Mariana Islands"
[91] "Malaysia"                                       "North America"
[93] "Nepal"                                          "OECD members"
[95] "Oman"                                           "Other small states"
[97] "Pakistan"                                       "Philippines"
```

The Solution:

When we compare the characters in both "fert$Country.Name" and "crop_organized$Area", we can see that apart from the groupings in "fert$Country.Name" all the text or strings match partially for example "Bahamas the," v.s "Bahamas", "United states of" vs "United states of america" and so on. so the idea for this solution lies in replacing the "fert$Country.Name" by "crop_organized$Area" when a number of strings matches.

Make new data frame :

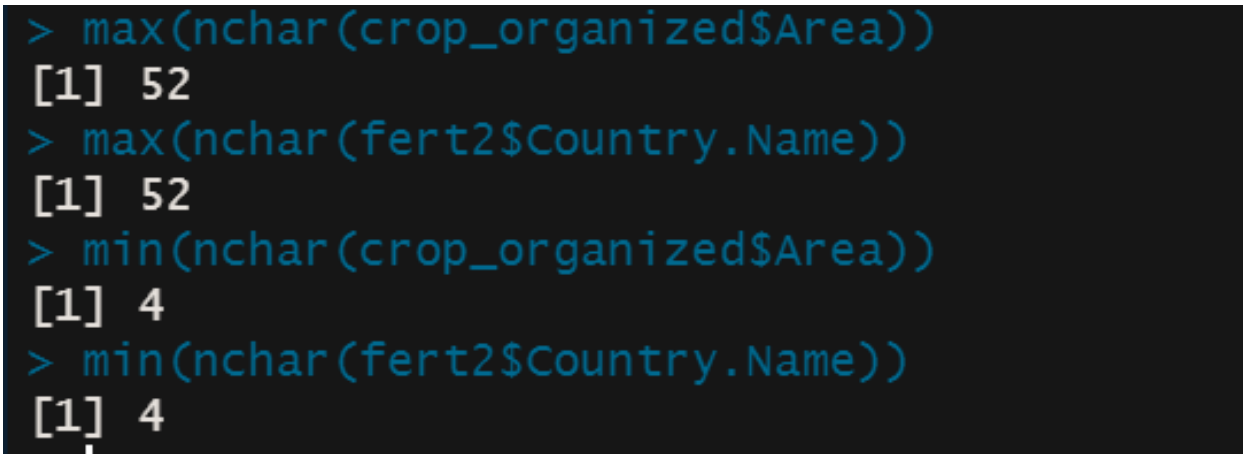
We make a new data frame called "fert2" based on exactly the data frame"fert".

The right number of strings:

To find the right number of strings we need we use *nchar().*

```
max(nchar(crop_organized$Area))
```

```
max(nchar(fert2$Country.Name))
```



I used it here with *max()* and *min()* to showcase the dissimilarity we have in the length of characters in our data. hence we will need to have something self-regulating for each of our 266 values.

Updating & Cleaning the Country.Name column:

For this we are going to use *amatch()*,*substring()*, conditional brackets, and some tricks. For the *substring()* we start from **1** but since the number of characters changes from one row to the other we use *nchar()* to tell *substr()* the number of character for each row, in *amatch()* I found the best method/algorithm to work for this problem is **lcs** *-The longest common substring algorithm-* in which the weight is ignored completely.

```
fert2$Country.Name <- crop_organized$Area[amatch(
  substr(fert2$Country.Name,1,nchar(fert2$Country.Name)),
  substr(crop_organized$Area,1,nchar(fert2$Country.Name)),method = "lcs",maxDist = 3)]
```

- We need to remove the rows left with missing values, and rearrange by alphabetic order.

```
fert2<-drop_na(fert2,Country.Name)
fert2<-arrange(fert2,Country.Name)
```

Checking if both data frames countries match :

- Here is some ways to check if the values in both data frames match

```
setdiff(fert2$Country.Name,crop_organized$Area)
sum(fert2$Country.Name %in% unique(crop_organized$Area))
crop_organized[!(crop_organized$Area %in% fert2$Country.Name),  ]
fert2[fert2$Country.Name %in% unique(crop_organized$Area),  ]
```

setdif() returrns 0 mismatches but that does not mean everything is good.

```
> setdiff(fert2$Country.Name,crop_organized$Area)
character(0)
```

using the %in% operator and sum() function shows us they do match on 143 country names which is the number of rows for "fert2"

```
> sum(fert2$Country.Name %in% unique(crop_organized$Area))
[1] 143
```

For further details we can use the other 2 last lines of codes, but lets investigate further more. But this means 14 countries are missing or are they ?

Let's start by checking duplicates, by using set of functions duplicated() and subset() we can see how many duplicates there are and what are they.

```
sum(duplicated(fert2$Country.Name,fromLast = TRUE))
dups_fert<-subset(fert2,duplicated(Country.Name))
View(dups_fert)
```

- We have 9 duplicates.

```
> sum(duplicated(fert2$Country.Name,fromLast = TRUE))
[1] 9
```

Checking by country codes using the **ISO -international organization for standardization-** we can update the countries to the right naming schemes of **"crop_organized"**, but even then we have some countries that simply don't exist in both data frames like the "USSR" and others because of political causes that disturb the naming flow.

	Country.Name	Country.Code
2	Albania	ABW
25	Canada	CHN
57	Grenada	GRD
67	Ireland	IRL
68	Ireland	ISR
71	Jamaica	JPN
82	Malawi	PLW
110	Romania	ROU
123	South Africa	SAS

Aruba — doesn't exist in crop_organized needs to be deleted

China

it's right but the other one should be Greenland

it's right but the other one should be Iraq

Israel

Japan

Palau — doesn't exist in crop_organized needs to be deleted

it's right but the other one should be Oman

Its a subgroup needs to be deleted

Updating to the right country names:

- We substitute the the wrong country names by the right ones and delete the ones which doesn't exist in "crop_organized".

```
fert2$Country.Name[fert2$Country.Code %in% dups_fert$Country.Code]<-
  c("d","China","Greenland","Iraq","Israel","Japan","d","Oman","d") # put
fert2<-subset(fert2, Country.Name!="d") # delete rows with d
```

Cleaning the year columns :

X1963	X1964	X1965	X1966	X1967	X1968

- We have an "X" before every year so before making the data frame wider we have to clean that.

```
names(fert2)<-sub("^X", "", names(fert2))
```

	Country.Name	Country.Code	Indicator.Name	Indicator.Code	X1960	X1961	X1962	X1963	X1964
190	Palau	PLW	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
191	Panama	PAN	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	1.141553e+01	22.83105023	22.83105023	2.511416
192	Papua New Guinea	PNG	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	1.333333e+00	1.33333333	1.33333333	1.333333
193	Paraguay	PRY	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	8.571429e-01	1.05485232	1.37174211	1.888305
194	Peru	PER	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	5.091370e+01	58.65215137	44.54545455	3.438487
195	Philippines	PHL	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	1.453193e+01	18.09610656	19.13580247	1.822314

	Country.Name	Country.Code	Indicator.Name	Indicator.Code	1960	1961	1962	1963	1964
1	Albania	ALB	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	14.58796296	11.92660550	12.04545455	1.444695e+
3	Algeria	DZA	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	7.57107540	8.73015873	8.54838710	8.561644e+
4	Angola	AGO	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.37453183	0.37037037	0.51470588	7.636364e-
5	Antigua and Barbuda	ATG	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
6	Argentina	ARG	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.87331290	0.66272159	1.07551020	1.475000e+
7	Australia	AUS	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	46.64923143	42.01770334	48.67514689	5.959731e+
8	Austria	AUT	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	148.90882180	172.62641280	196.86231880	2.090960e+
9	Bahamas	BHR	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
10	Barbados	BRB	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	320.37500000	477.43750000	446.56250000	5.716875e+
11	Belarus	BLR	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
12	Belgium	BEL	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	384.61569020	462.51593630	479.24075920	4.318657e+
13	Belize	BLZ	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	10.00000000	25.50000000	28.62162162	7.513514e+
14	Benin	BEN	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.64239130	0.63510638	0.44270833	8.265306e-
15	Bolivia (Plurinational State of)	BOL	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.61823802	0.69444444	0.88190184	1.081283e+
16	Bosnia and Herzegovina	BIH	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
17	Botswana	BWA	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	2.78195489	2.75689223	2.75689223	4.010025e+
18	Brazil	BRA	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	11.42728966	10.60073685	12.06230000	9.821185e+
19	Bulgaria	BGR	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	33.80362183	36.30019259	42.28250362	6.790813e+
20	Burkina Faso	BFA	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.01365348	0.05154639	0.03591418	4.513727e-
21	Burundi	BDI	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
22	Cabo Verde	CPV	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	NA	NA	NA	
23	Cameroon	CMR	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.65000000	0.64356436	1.03921569	1.339806e+
24	Canada	CAN	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	10.96976236	12.39003655	13.40391841	1.490057e+
25	China	CHN	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	7.04082324	9.59844811	12.11820841	1.632832e+
26	Central African Republic	CAF	Fertilizer consumption (kilograms per hectare of arable land)	AG.CON.FERT.ZS	NA	0.18273809	0.18934911	0.21764706	2.339181e-

Removing some columns :

- Since we have no use for the "Indicator.Name","Indicator.Code" and the empty "1960" columns we will delete them.

```
fert2<-select(fert2,-c(3,4,5))
```

Making the data frame longer :

- The data frame is too wide and doesn't match our "crop_organized" structure, so will group the columns from "1961" to "2020" under one column called "year" as for the values inside them will be called "fertilizer_utilization".

```
fert_organized<-fert2
fert_organized<- pivot_longer(fert_organized, cols = 3:63,names_to = "year")
fert_organized<-rename(fert_organized,"fertilizer_utilization"="value")#Renaming columns
```

```
> head(fert_organized)
# A tibble: 6 × 4
  Country.Name Country.Code year  fertilizer_utilization
  <chr>        <chr>        <chr>                  <dbl>
1 Albania      ALB          1961                    14.6
2 Albania      ALB          1962                    11.9
3 Albania      ALB          1963                    12.0
4 Albania      ALB          1964                    14.4
5 Albania      ALB          1965                    17.4
6 Albania      ALB          1966                    20
```

Changing year data type :

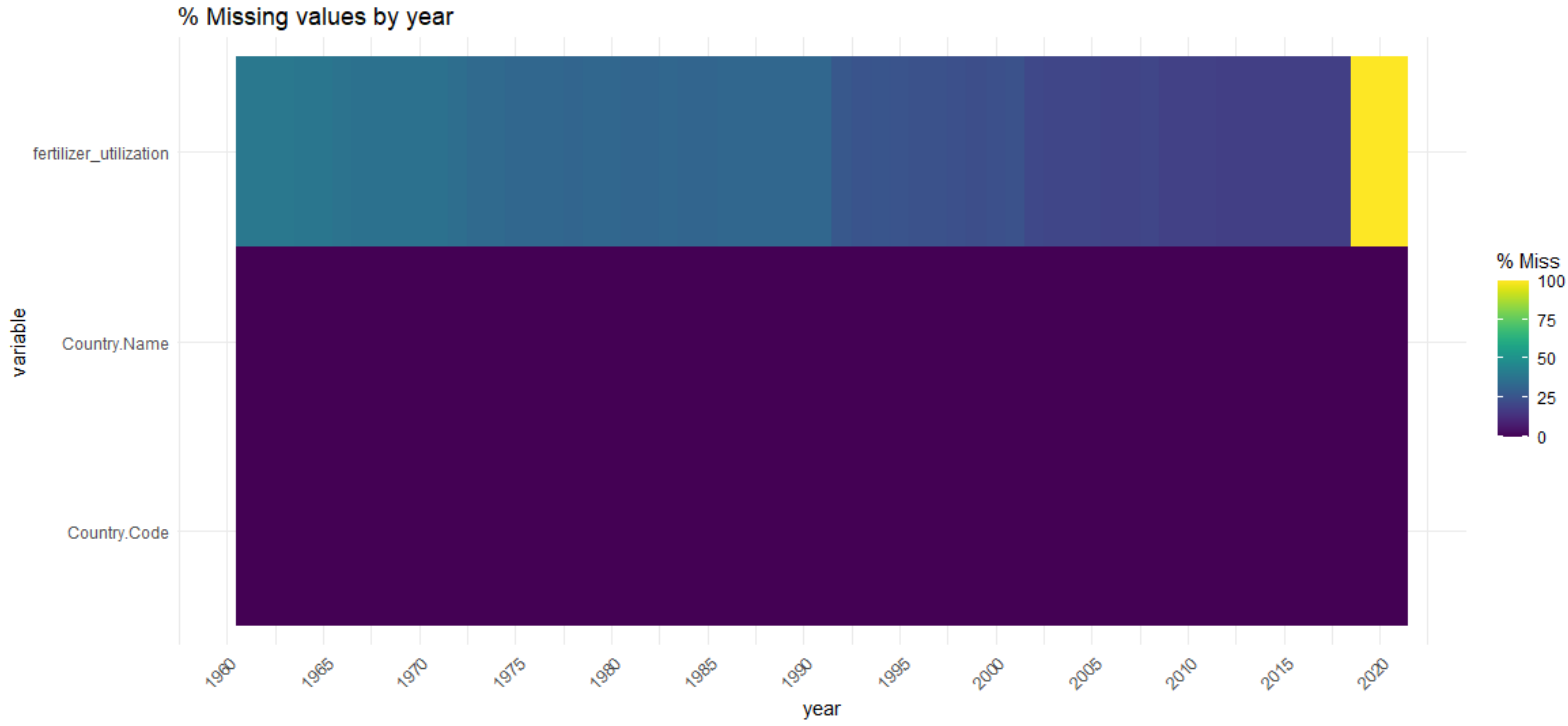
- We can see from the head() function that year is set as a character so we will change it to integer.

```
######## Changing year data type
fert_organized$year<-as.integer(fert_organized$year)
```

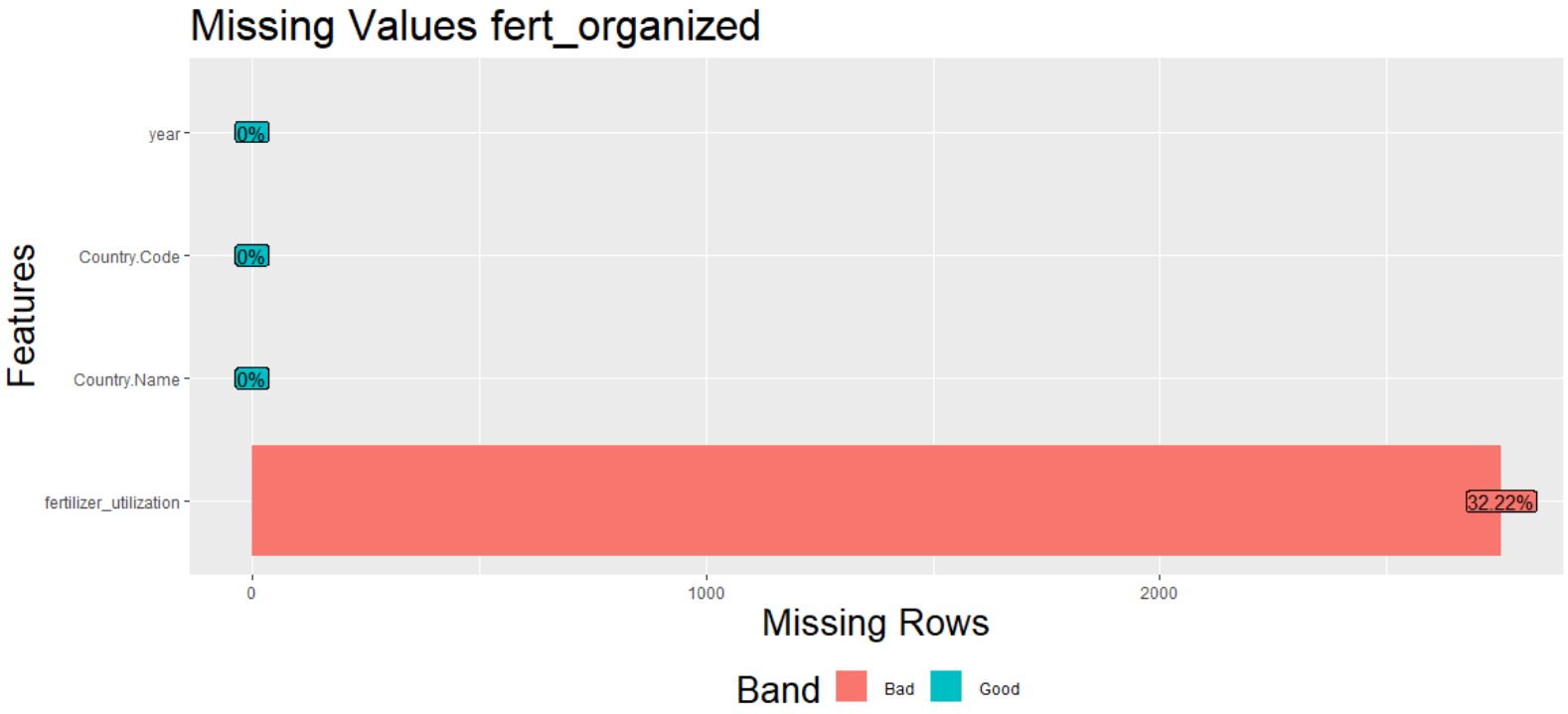
```
> head(fert_organized)
# A tibble: 6 × 4
  Country.Name Country.Code  year fertilizer_utilization
  <chr>        <chr>        <int>                  <dbl>
1 Albania      ALB           1961                    14.6
2 Albania      ALB           1962                    11.9
3 Albania      ALB           1963                    12.0
4 Albania      ALB           1964                    14.4
5 Albania      ALB           1965                    17.4
6 Albania      ALB           1966                    20
```

Visualizing missing values :

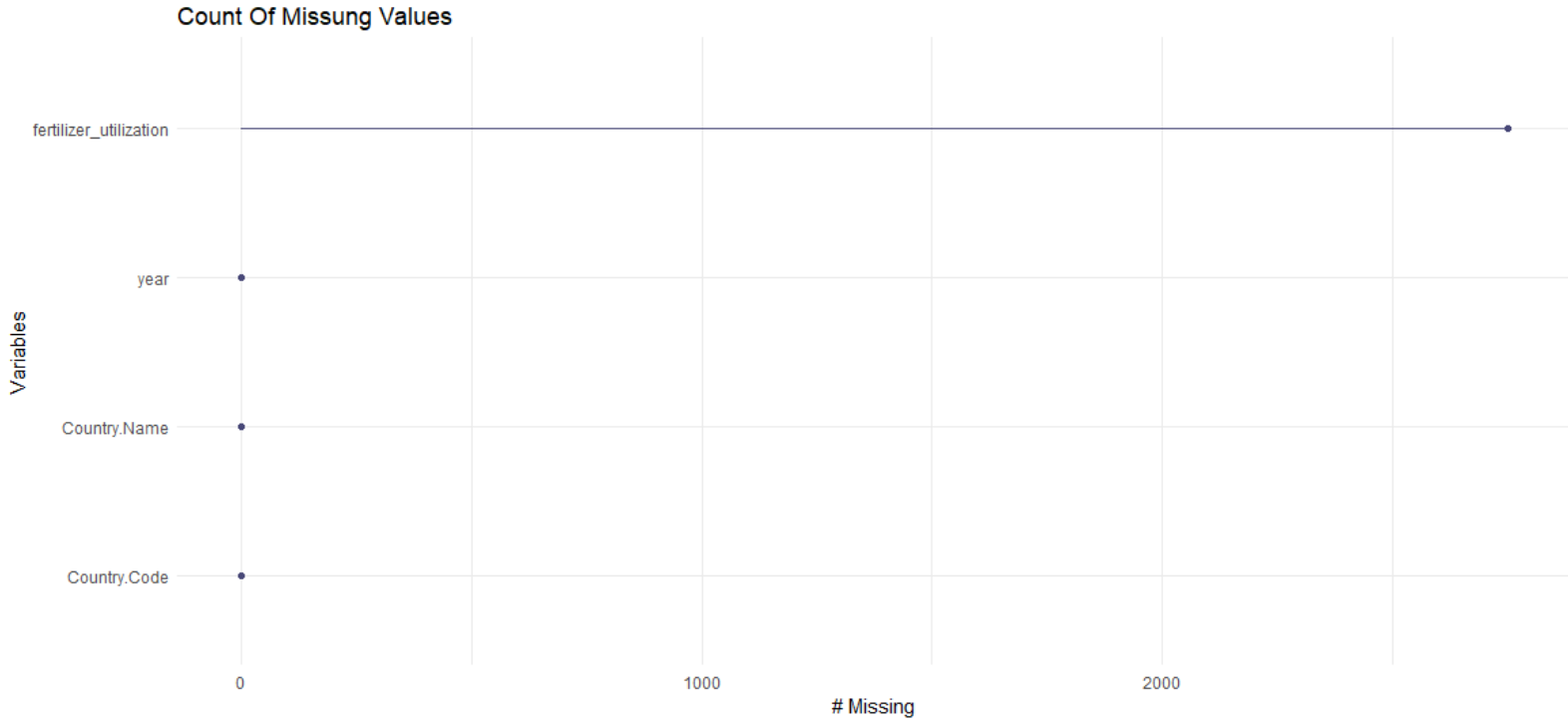
- Most missing values happen to be in the fertilizer_utilization column between 2019 and 2020



- fertilizer_utilization have a 32.22% of missing values

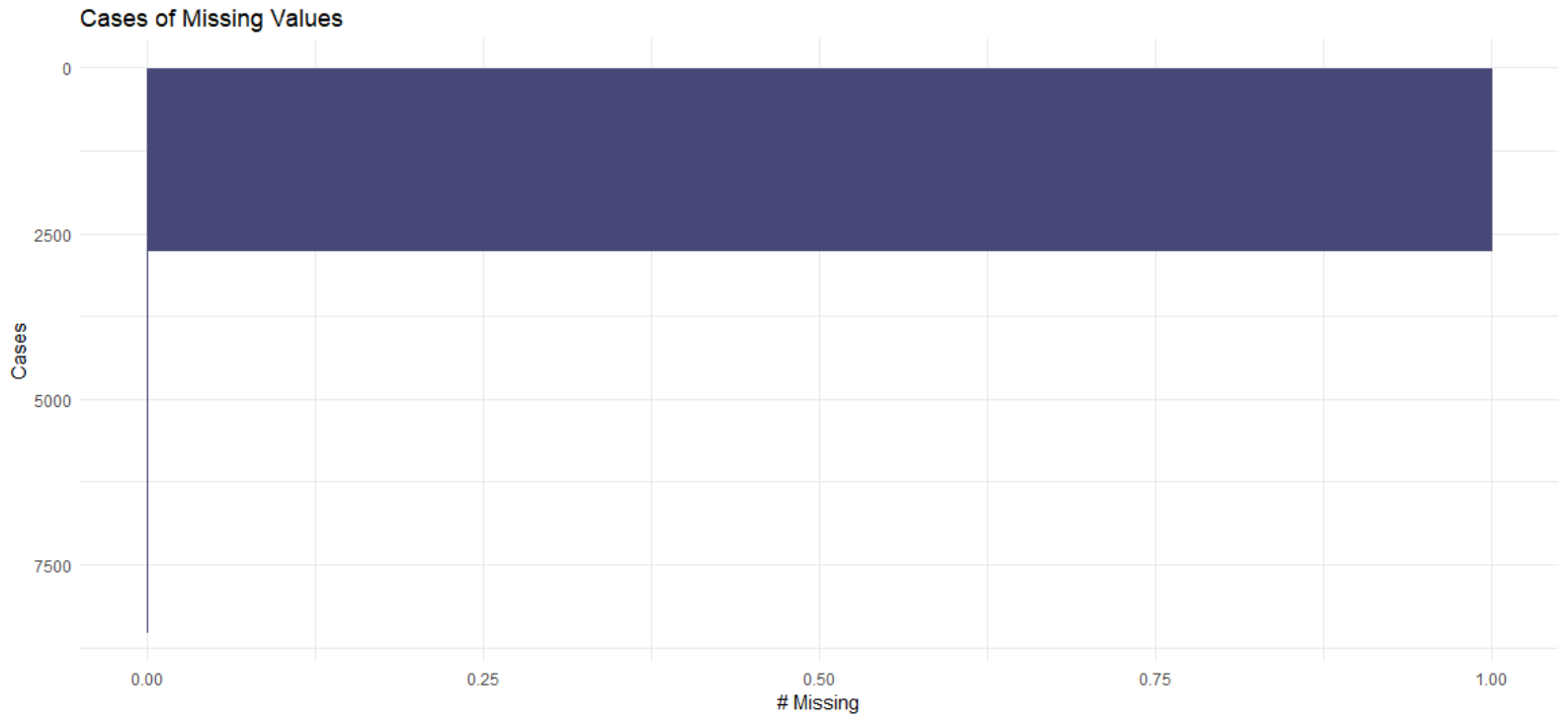


- fertilizer_utilization have 2752 missing values



- Using the *gg_miss_case()* we can see about 2700 cases have 100% missing values

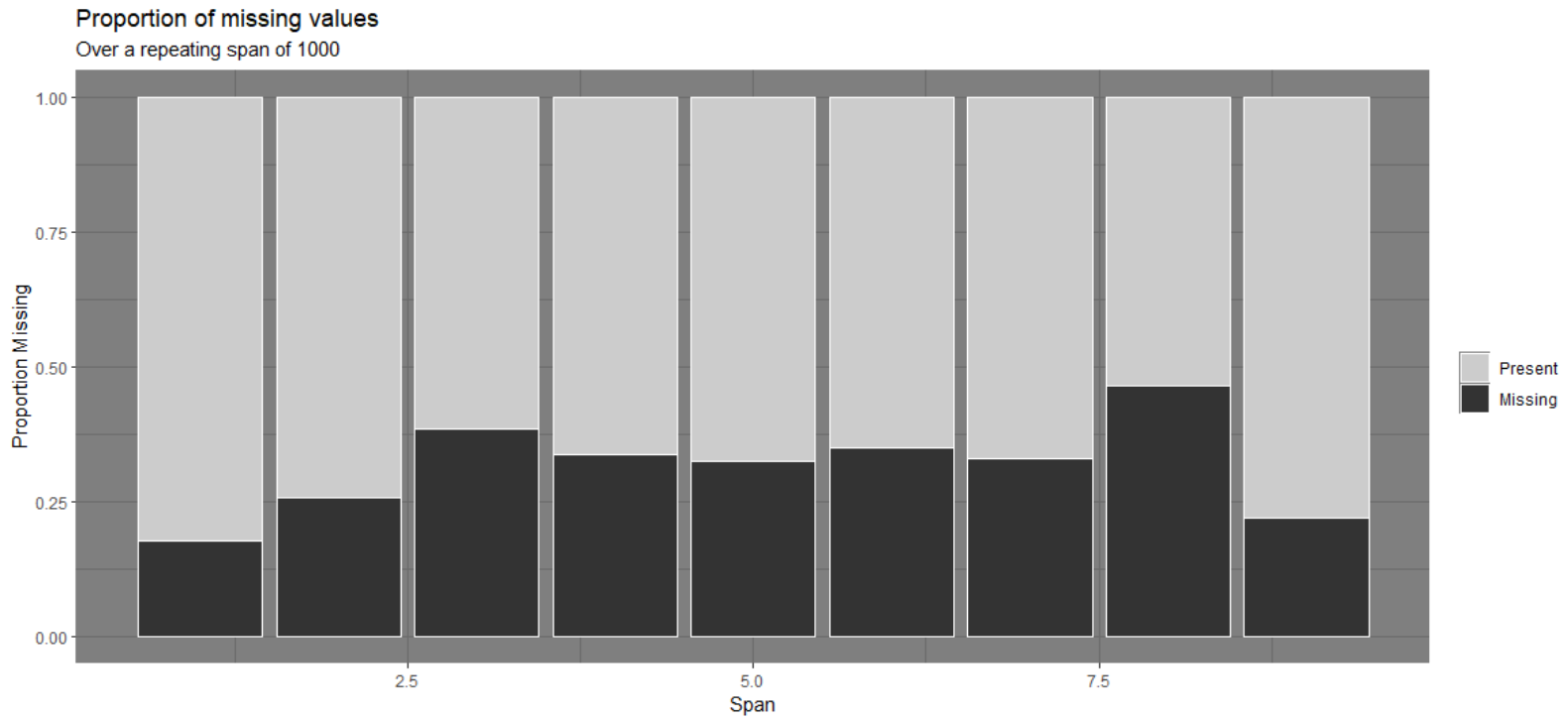
```
gg_miss_case(fert_organized)+labs(title = "Cases of Missing Values")
gg_miss_case(fert_organized, facet = year)+labs(title = "Cases of Missing Values")
```



- To show cases of missing values by year we use facets, this shows us that from 2019 to 2021 it's all missing values and that there are certain countries(about 50) that have missing values for all years

- Using *gg_miss_span()* we can see the number of missings in a given span, or breaksize

```
gg_miss_span(fert_organized, fertilizer_utilization, span_every = 1000)+theme_dark()
```



- From this analysis we need to delete rows where the year is 2019,2020, or 2021.

```
fert_organized<-fert_organized %>%
  subset(fert_organized$year!=2019|temp$year!=2020|temp$year!=2021)
```

We do see a lot of missing values in the "fert_organized" data frame, but we can still derive useful insights, to be more accurate we can use a frame of quantifiable data from this dataset where there are no missing values, but we will leave that for our analysis phase.

With this our " fert" data frame is processed into a clean & ready to be analyzed data frame called "crop_organized".

POPULATION:

Since it has the same structure as **"fert"** we will repeat the same prcocess. that's what nice about R !

Even if the structure is the same we still have to check for missing values.

```
####### Checking for missing values
sum(is.na(pop_organized))
temp<-pop_organized
temp<-pop_organized[rowSums(is.na(pop_organized)) > 0,]
View(temp)
```

```
> sum(is.na(pop_organized))
[1] 10
```

	Country.Name	Country.Code	year	Population
1	Eritrea	ERI	2012	*NA*
2	Eritrea	ERI	2013	*NA*
3	Eritrea	ERI	2014	*NA*
4	Eritrea	ERI	2015	*NA*
5	Eritrea	ERI	2016	*NA*
6	Eritrea	ERI	2017	*NA*
7	Eritrea	ERI	2018	*NA*
8	Eritrea	ERI	2019	*NA*
9	Eritrea	ERI	2020	*NA*
10	Eritrea	ERI	2021	*NA*

Although we can delete I rather just leave it there since it's an MNAR (Missing not at random).

With this our " population" data frame is processed into a clean & ready to be analyzed data frame called "pop_organized".

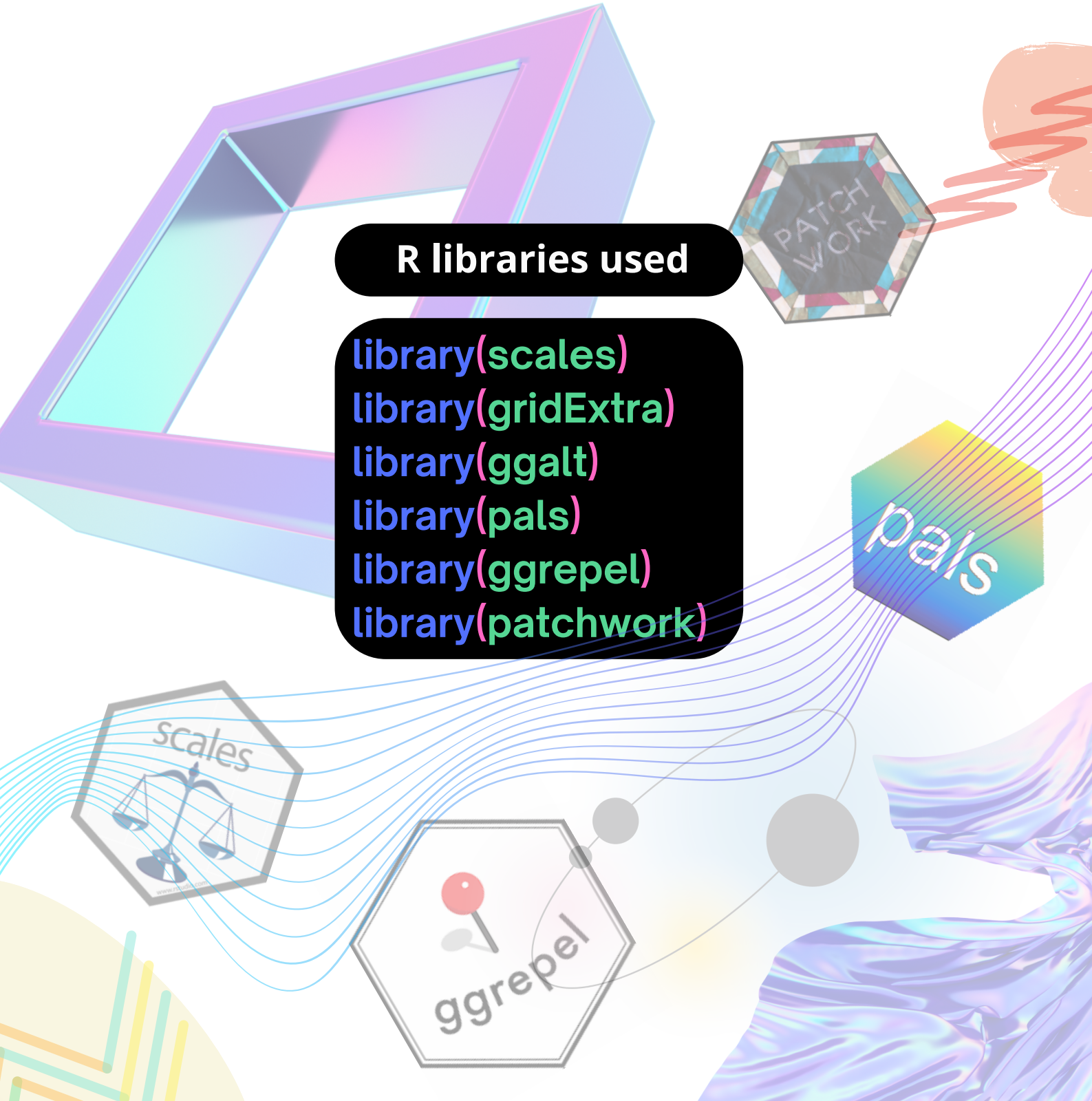

We cleaned 3 different datasets, using a multitude of methods and functions, we visualized the missing values, renamed columns, calculated and extracted the unique values, turned a long data frame into a wide data frame and turned wide data frames into long ones. For the crop dataset we went from 783,782 rows of 3 unique variables into 248,331 rows with 6 total unique variables, as for the fertilizer usage data set we went from 266 rows of 66 columns into 8,120 entries with 4 total columns same as fertilizer dataset we transformed the population dataset from 266 rows of 66 columns into 8,680 rows with 4 columns.

4

ANALYZING

4.ANALYZE:

In this section, we will answer the questions we set in the ASK process by summarizing & aggregating values, making calculations, and creating plots to draw conclusions, make predictions, recommendations, and data-driven decisions.



CROPS:

- The average area harvested is **135,912.5 ha.**
- The average production is **656,195.6 tonnes.**
- The average yield is **12.11 tonnes/ha.**

```
mean(crop_organized$`Area_harvested(ha)`,na.rm=TRUE)
mean(crop_organized$`Production(tonnes)`,na.rm = TRUE)
mean(crop_organized$`Yield(tonnes/ha)`,na.rm = TRUE)
```

- The max area harvested is **70,205,008 ha.**
- The max production is **768,594,154 tonnes.**
- The max yield is **5084.74 tonnes/ha.**

```
max(crop_organized$`Area_harvested(ha)`,na.rm=TRUE)
max(crop_organized$`Production(tonnes)`,na.rm = TRUE)
max(crop_organized$`Yield(tonnes/ha)`,na.rm = TRUE)
```

- The minimum for each variable is **0**.

```
min(crop_organized$`Area_harvested(ha)`,na.rm=TRUE)
min(crop_organized$`Production(tonnes)`,na.rm = TRUE)
min(crop_organized$`Yield(tonnes/ha)`,na.rm = TRUE)
```

Let's check our findings.

- It seems that the **USSR** had the highest area harvested and it was in **1965** growing **wheat** but even so, it still had a low yield when compared to the average.

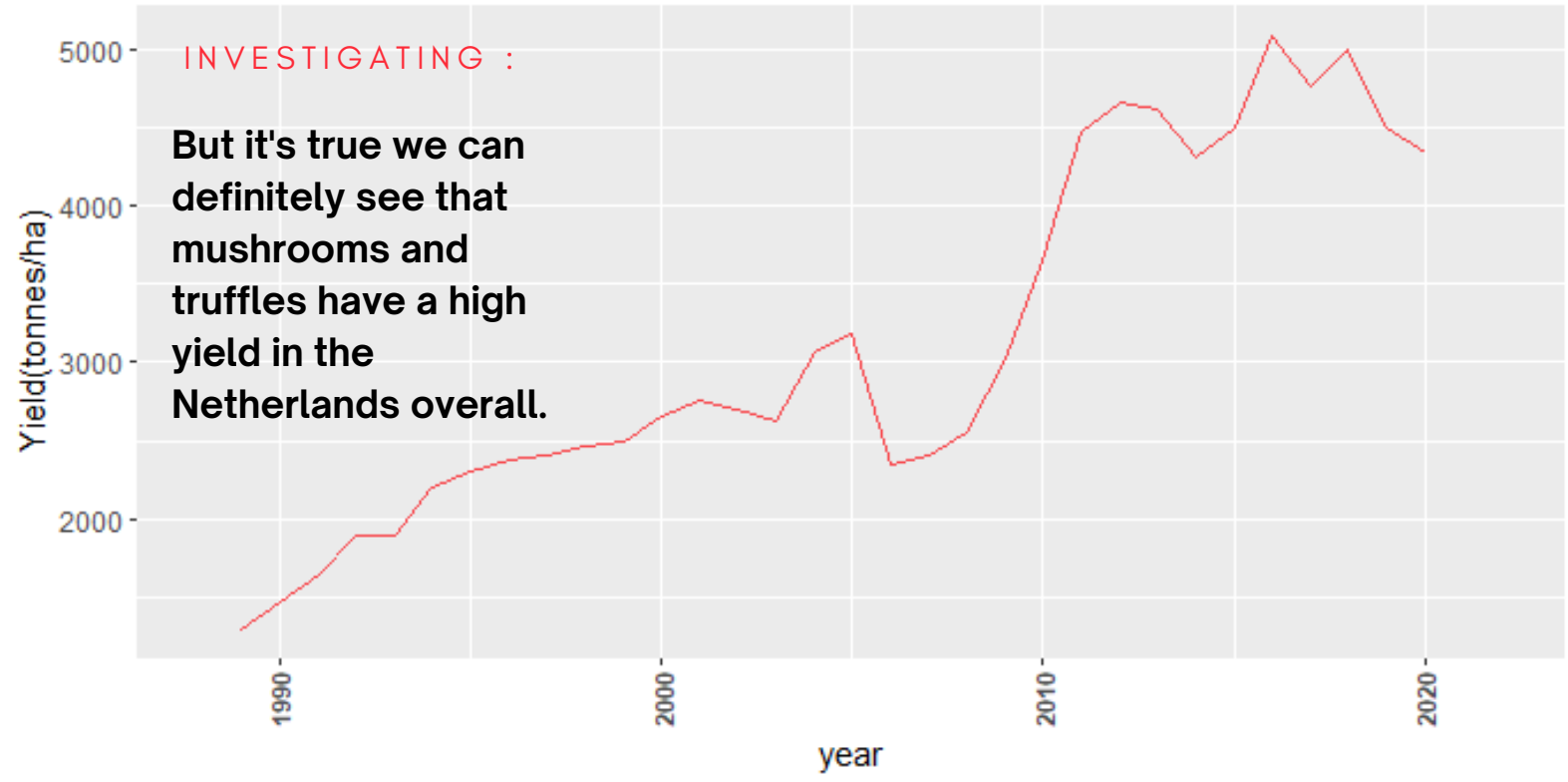
```
> filter(crop_organized, crop_organized$`Area_harvested(ha)` > 70205007)
# A tibble: 1 × 6
  Area   year Item  `Area_harvested(ha)` `Production(tonnes)` `Yield(tonnes/ha)`
  <chr> <int> <chr>                <int>                <int>               <dbl>
1 USSR   1965 Wheat             70205008             56105008               0.799
```

- **Brazil** had the highest production in **2016** growing **sugar cane** with an area of harvest lower than the average.

```
> filter(crop_organized,crop_organized$`Production(tonnes)` > 768594153)
# A tibble: 1 × 6
  Area    year Item       `Area_harvested(ha)` `Production(tonnes)` `Yield(tonnes/ha)`
  <chr>  <int> <chr>                     <int>                <int>              <dbl>
1 Brazil  2016 Sugar cane              10223894            768594154               75.2
```

- **2016** seems also a good year for the **Netherlands** with the highest yield harvesting **mushrooms and truffles** but it seems too good to be true.

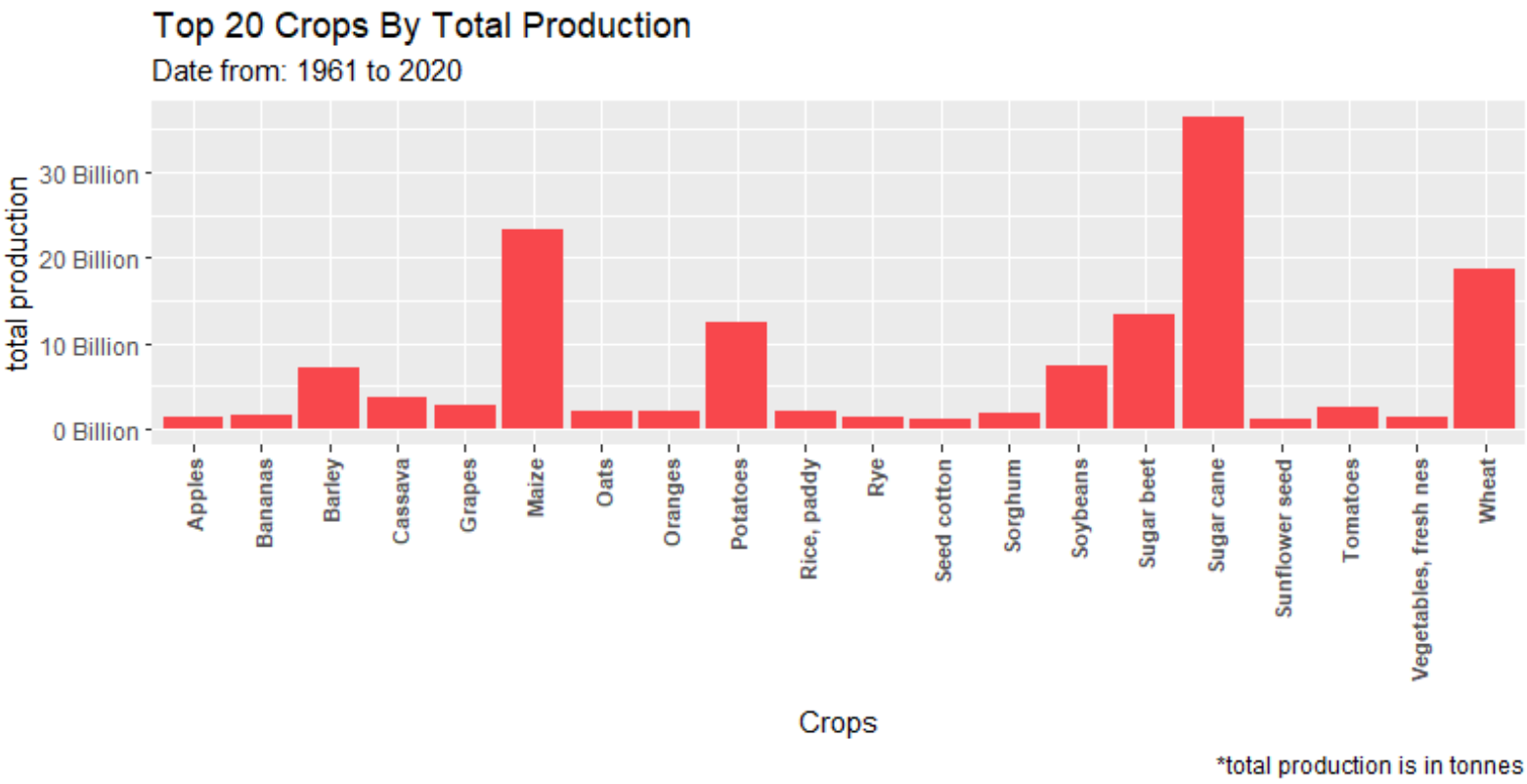
```
> filter(crop_organized,crop_organized$`Yield(tonnes/ha)` > 5084.745)
# A tibble: 1 × 6
  Area        year Item                `Area_harvested(ha)` `Production(tonnes)` Yield(to…¹
  <chr>      <int> <chr>                              <int>                <int>      <dbl>
1 Netherlands  2016 Mushrooms and truffles                59               300000      5085.
```




```
invest<-crop_organized %>%
  filter(.,Area == "Netherlands",Item =="Mushrooms and truffles")

ggplot(invest,aes(x= year,y=`Yield(tonnes/ha)`))+
  geom_line(color="#f8474c")+
  xlim(c(1988,2022))+
  theme(axis.text.x = element_text(size=8,face="bold",angle = 90, vjust = 0.5, hjust=1))
```

WHAT ARE THE TOP 20 PROMINENT CROPS BY TOTAL PRODUCTION FROM 1961 TILL 2020?

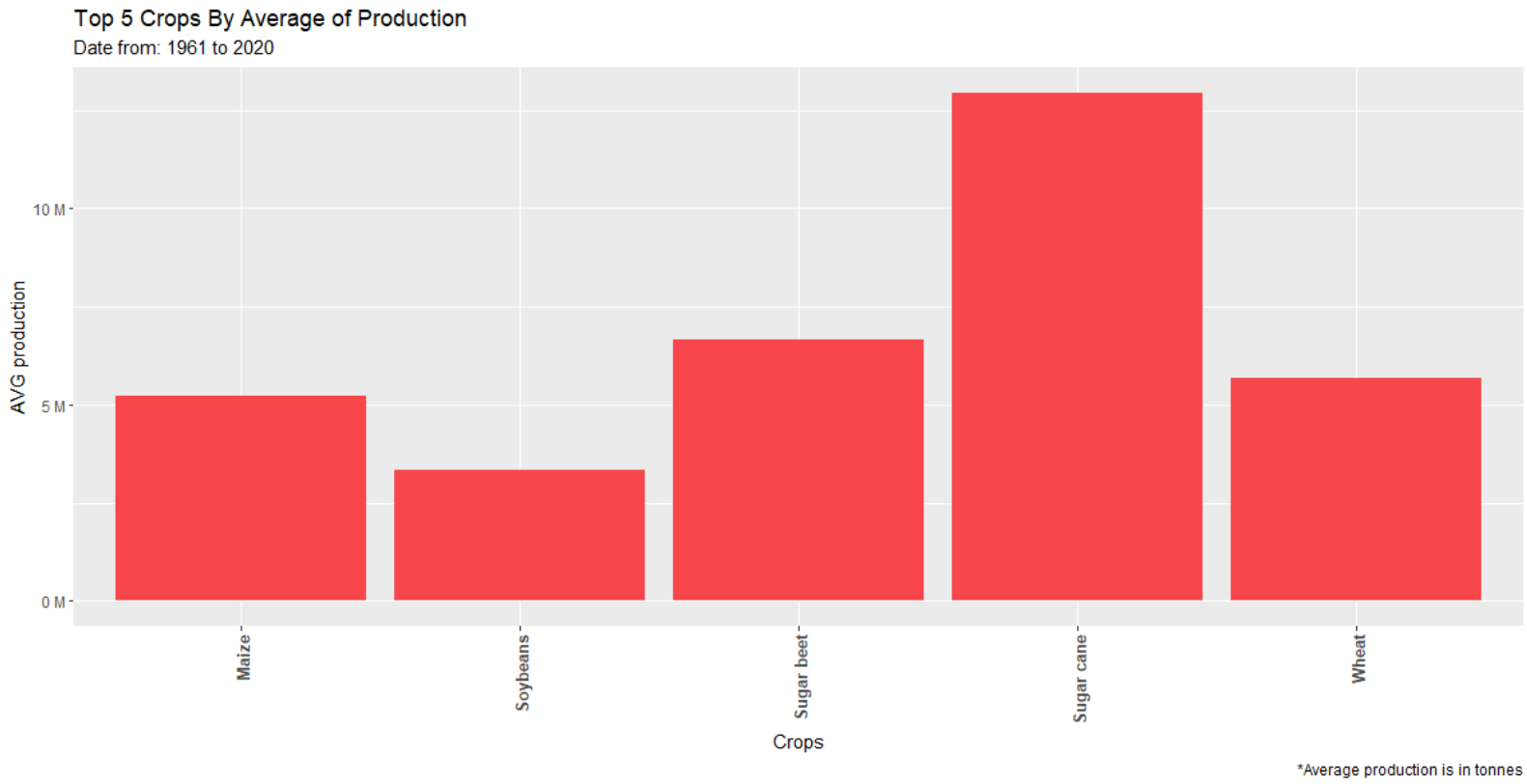
The top 20 crops by productions are as follow :




```r
mindate<-min(crop_organized$year)
maxdate<-max(crop_organized$year)

sumpro<-crop_organized %>%
  group_by(Item) %>%
  drop_na() %>%
  summarize(total_production=sum(`Production(tonnes)`)) %>%
  arrange(desc(total_production)) %>%
  top_n(total_production,n = 20)

ggplot(data = sumpro,aes(x=Item,y=total_production))+
  geom_bar(stat = "identity",fill="#f8474c")+
  scale_y_continuous(labels = unit_format(unit = "Billion", scale = 1e-9))+
  theme(axis.text.x = element_text(size=7,face="bold",angle = 90, vjust = 0.5, hjust=1))+
  ylab("total production")+
  xlab("Crops")+
  labs(title = "Top 20 Crops By Total Production",
       subtitle = paste0("Date from: ",mindate," to ",maxdate),
       caption = "*total production is in tonnes")
```

WHAT ARE THE TOP 5 PROMINENT CROPS BY AVERAGE PRODUCTION FROM 1961 TILL 2020?

The story changes when we aggregate by average.

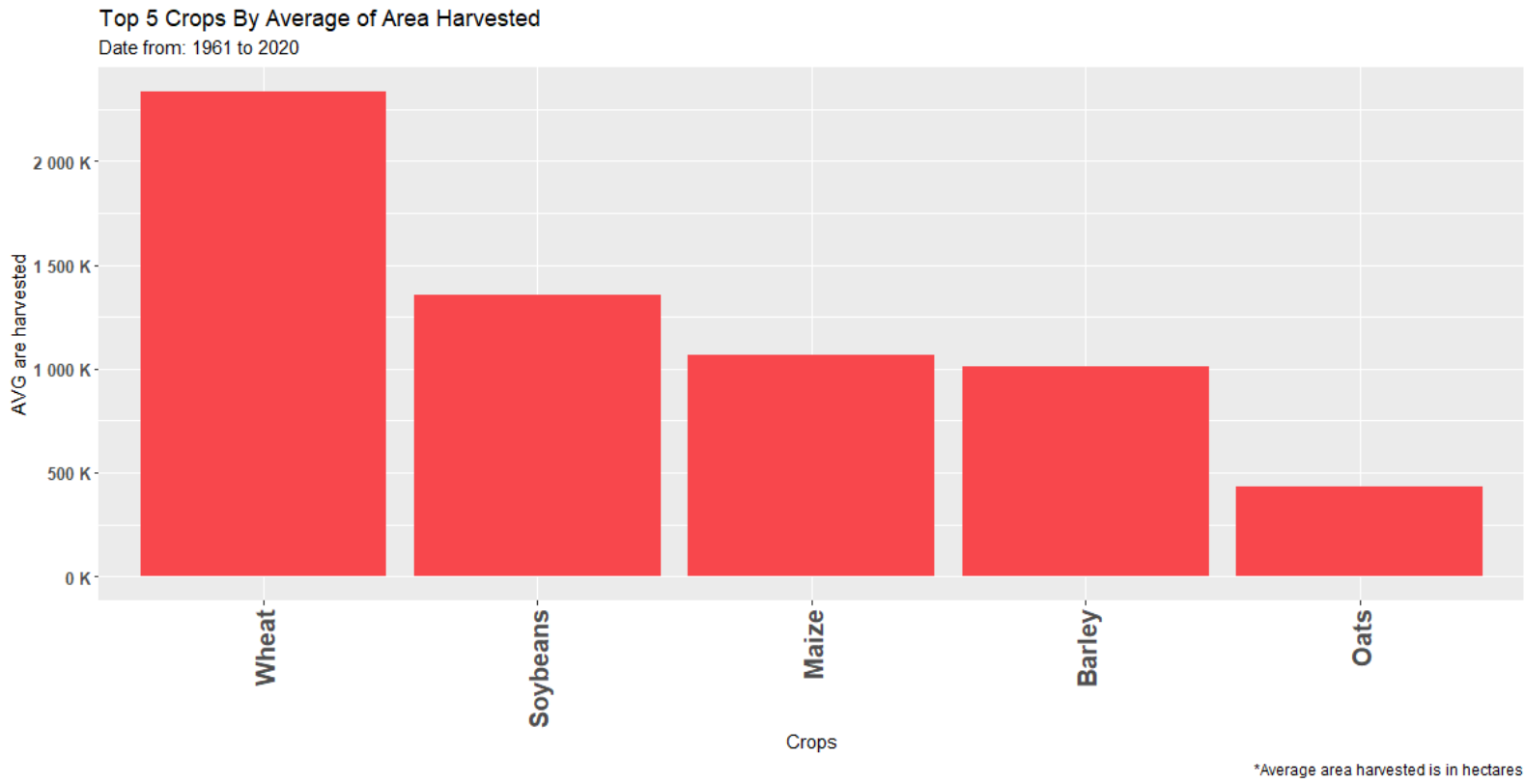


When compared by total production the **potatoes** is not in the top 5 this time, and instead it's taken by **soybeans**.

```
meanpro<-crop_organized %>%
  group_by(Item) %>%
  drop_na() %>%
  summarize(avg_production=mean(`Production(tonnes)`)) %>%
  arrange(desc(avg_production)) %>%
  top_n(avg_production,n = 5)

ggplot(data = meanpro,aes(x=Item,y=avg_production))+
  geom_bar(stat = "identity",fill="#f8474c")+
  scale_y_continuous(labels = unit_format(unit = "M", scale = 1e-6))+
  theme(axis.text.x = element_text(size=10,face="bold",angle = 90, vjust = 0.5, hjust=1))+
  ylab("AVG production")+
  xlab("Crops")+
  labs(title = "Top 5 Crops By Average of Production",
       subtitle = paste0("Date from: ",mindate," to ",maxdate),
       caption = "*Average production is in tonnes")
```

WHAT ARE THE TOP 5 CROPS THAT USES THE BIGGEST AMOUNT OF AREA FROM 1961 TILL 2020?

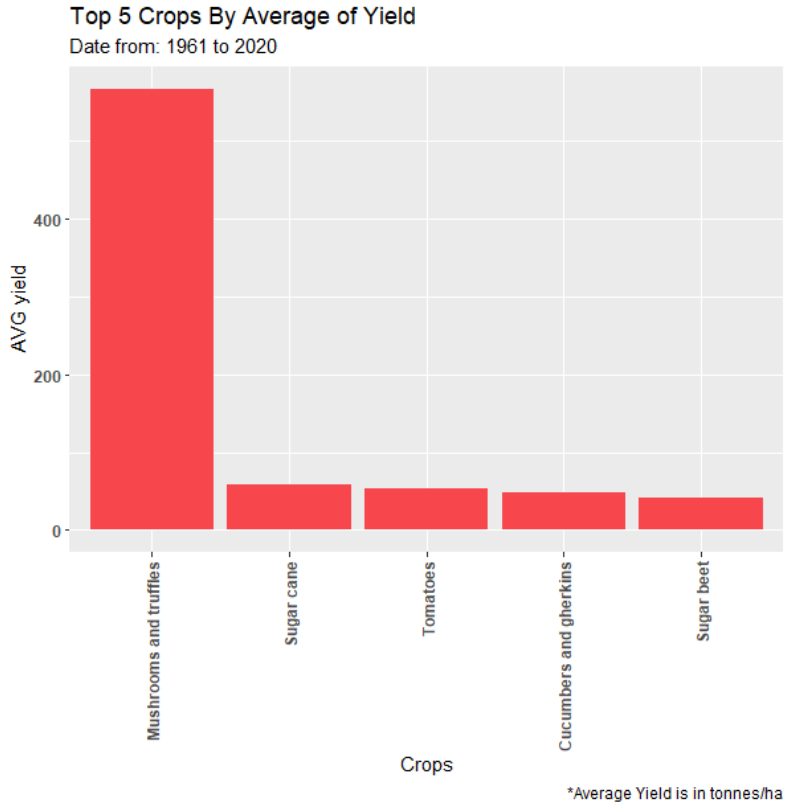
We can derive that only **wheat**, **soybeans**, and **maize** are in the top 5 in both the average production as well as the average area used for harvest, with **barley** and **oats** in the 4th and 5th place respectively .

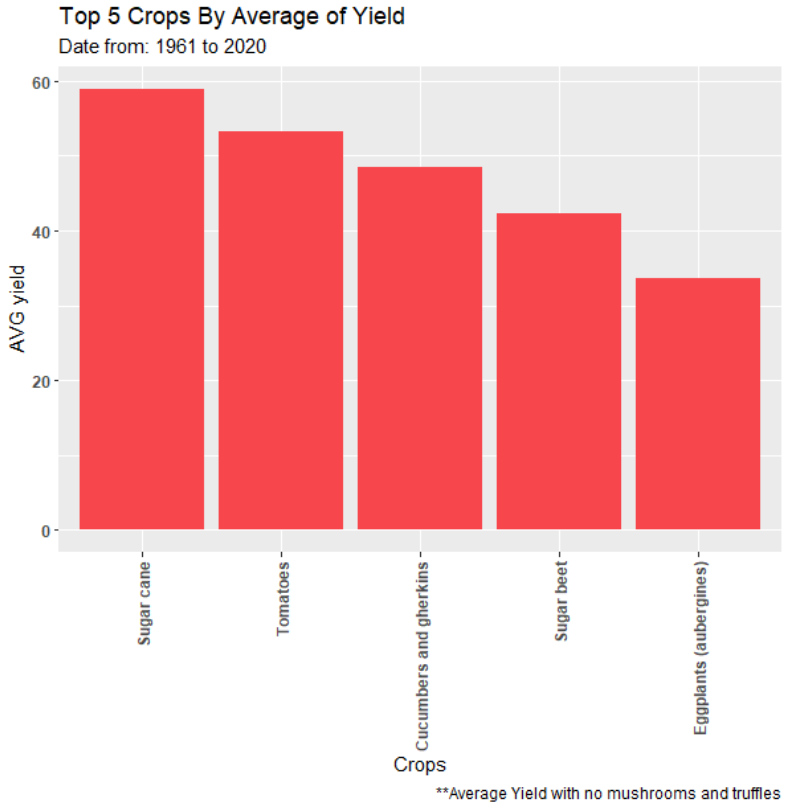



```r
meanarea<-crop_organized %>%
  group_by(Item) %>%
  drop_na() %>%
  summarize(avg_area=mean(`Area_harvested(ha)`)) %>%
  arrange(desc(avg_area)) %>%
  top_n(avg_area,n = 5)


ggplot(data = meanarea,aes(x=reorder(Item, -avg_area),y=avg_area))+
  geom_bar(stat = "identity",fill="#f8474c")+
  scale_y_continuous(labels = unit_format(unit = "K", scale = 1e-3))+
  theme(axis.text.x = element_text(size=15,face="bold",angle = 90, vjust = 0.5, hjust=1),
        axis.text.y = element_text(size=10,face="bold"))+
  ylab("AVG are harvested")+
  xlab("Crops")+
  labs(title = "Top 5 Crops By Average of Area Harvested",
       subtitle = paste0("Date from: ",mindate," to ",maxdate),
       caption = "*Average area harvested is in hectares")
```

WHAT ARE THE TOP 5 CROPS THAT YIELD THE HIGHEST ?

Apart from **mushrooms and truffles** having the highest average yield, **sugar cane**, **tomatoes**, **cucumbers and gherkins**, **sugar beet**, and **eggplants** all have the highest average yield, stating that we found the average yield is 12.11 tonnes/ha.

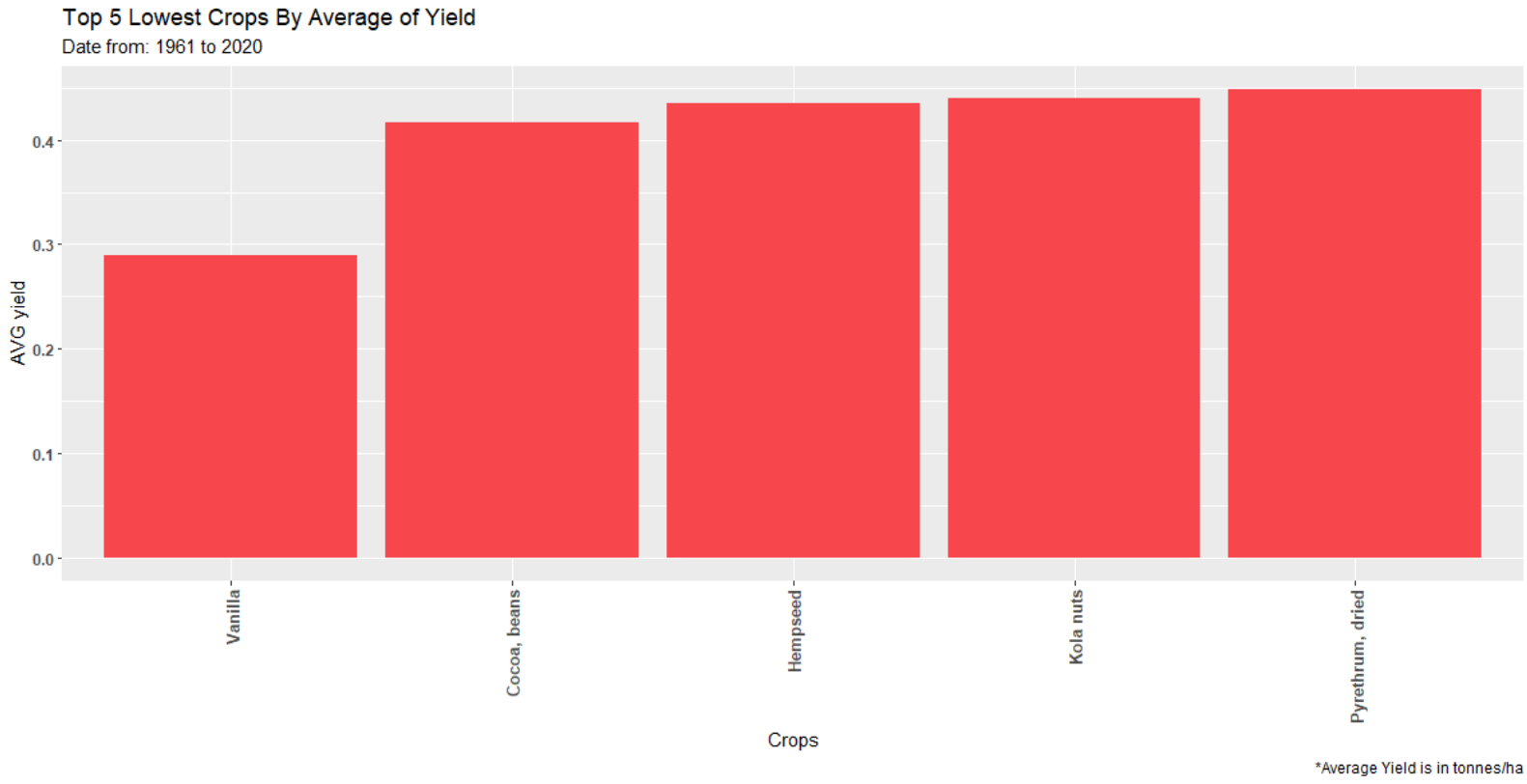





```r
meanyield_nomush<-crop_organized %>%
  filter(.,crop_organized$Item !="Mushrooms and truffles") %>%
  group_by(Item) %>%
  drop_na() %>%
  summarize(avg_yield=mean(`Yield(tonnes/ha)`)) %>%
  arrange(desc(avg_yield)) %>%
  top_n(avg_yield,n = 5)

nomush<-ggplot(data = meanyield_nomush,aes(x=reorder(Item, -avg_yield),y=avg_yield))+
  geom_bar(stat = "identity",fill="#f8474c")+
  theme(axis.text.x = element_text(face="bold",angle = 90, vjust = 0.5, hjust=1),
        axis.text.y = element_text(face="bold"))+
  ylab("AVG yield")+
  xlab("Crops")+
  labs(title = "Top 5 Crops By Average of Yield",
       subtitle = paste0("Date from: ",mindate," to ",maxdate),
       caption = "**Average Yield with no mushrooms and truffles")

meanyield<-crop_organized %>%
  group_by(Item) %>%
  drop_na() %>%
  summarize(avg_yield=mean(`Yield(tonnes/ha)`)) %>%
  arrange(desc(avg_yield)) %>%
  top_n(avg_yield,n = 5)

mush<-ggplot(data = meanyield,aes(x=reorder(Item, -avg_yield),y=avg_yield))+
  geom_bar(stat = "identity",fill="#f8474c")+
  theme(axis.text.x = element_text(face="bold",angle = 90, vjust = 0.5, hjust=1),
        axis.text.y = element_text(face="bold"))+
  ylab("AVG yield")+
  xlab("Crops")+
  labs(title = "Top 5 Crops By Average of Yield",
       subtitle = paste0("Date from: ",mindate," to ",maxdate),
       caption = "*Average Yield is in tonnes/ha")

grid.arrange(mush, nomush, nrow = 1)
```

WHAT ARE THE TOP 5 CROPS THAT YIELD THE LOWEST?

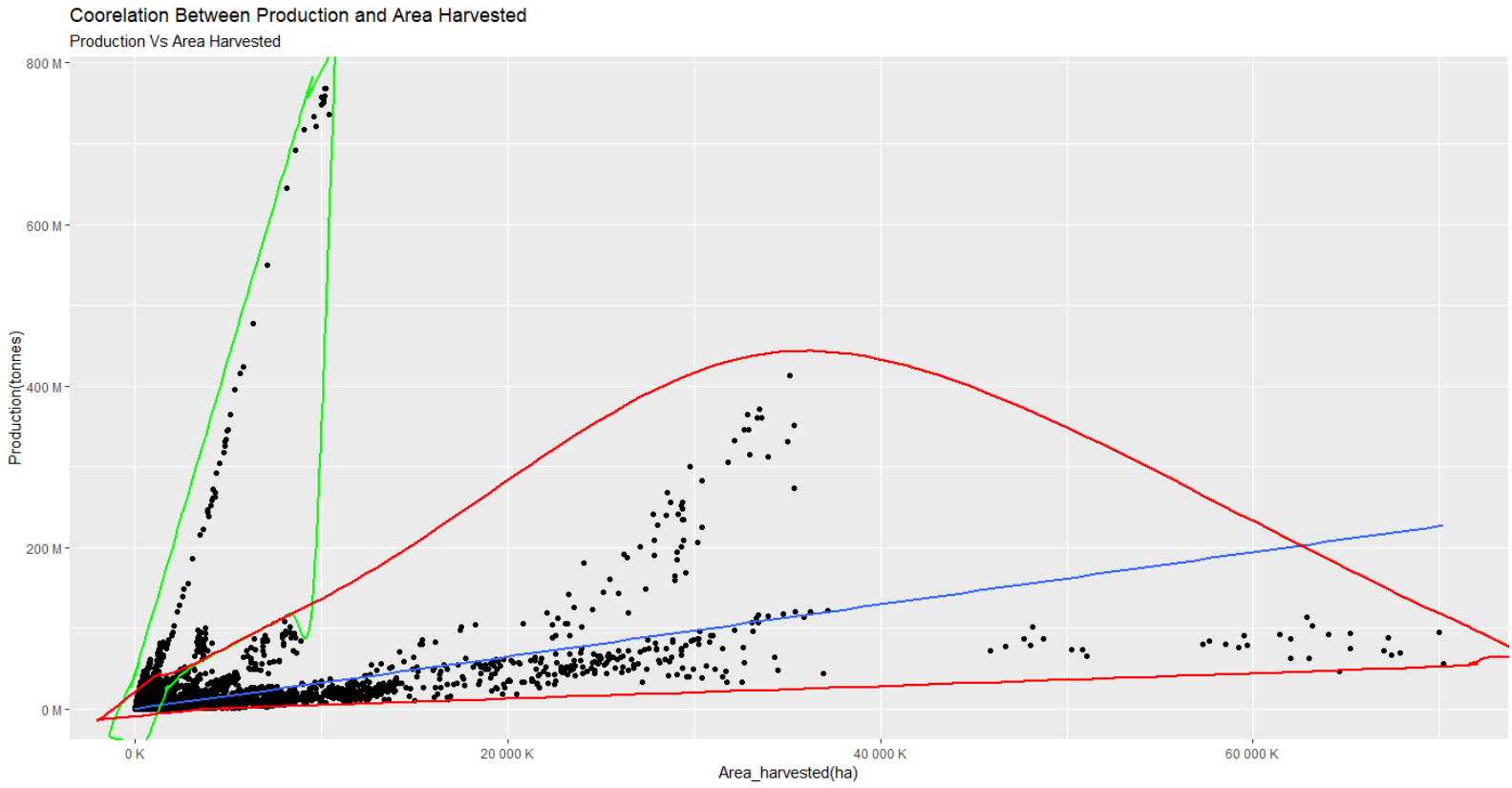
Vanilla has the lowest average yield with just **0.28 tonnes/ha**, the highest value on this chart is only **0.45 tonnes/ha** and it's for the **pyrethrum, dried** plant.




```
lowyield<-crop_organized %>%
  group_by(Item) %>%
  drop_na() %>%
  summarize(avg_yield=mean(`Yield(tonnes/ha)`)) %>%
  arrange(desc(avg_yield)) %>%
  top_n(avg_yield,n = -5)

ggplot(data = lowyield,aes(x=reorder(Item, avg_yield),y=avg_yield))+
  geom_bar(stat = "identity",fill="#f8474c")+
  theme(axis.text.x = element_text(face="bold",angle = 90, vjust = 0.5, hjust=1),
        axis.text.y = element_text(face="bold"))+
  ylab("AVG yield")+
  xlab("Crops")+
  labs(title = "Top 5 Lowest Crops By Average of Yield",
       subtitle = paste0("Date from: ",mindate," to ",maxdate),
       caption = "*Average Yield is in tonnes/ha")
```

WHAT IS THE RELATION BETWEEN PRODUCTION AND THE AREA OF HARVEST?

We can conclude from this scatter plot thanks to the correlation line that we have a **positive correlation** between the amount of production and area of harvest, the line is not steep meaning some crops have low yields and it's a big chunk of the overall crops, thanks to geom_encircle() function we can visually group the data into 2 groups ones with above average yield in **green**, and those with below-average yield in **red**.

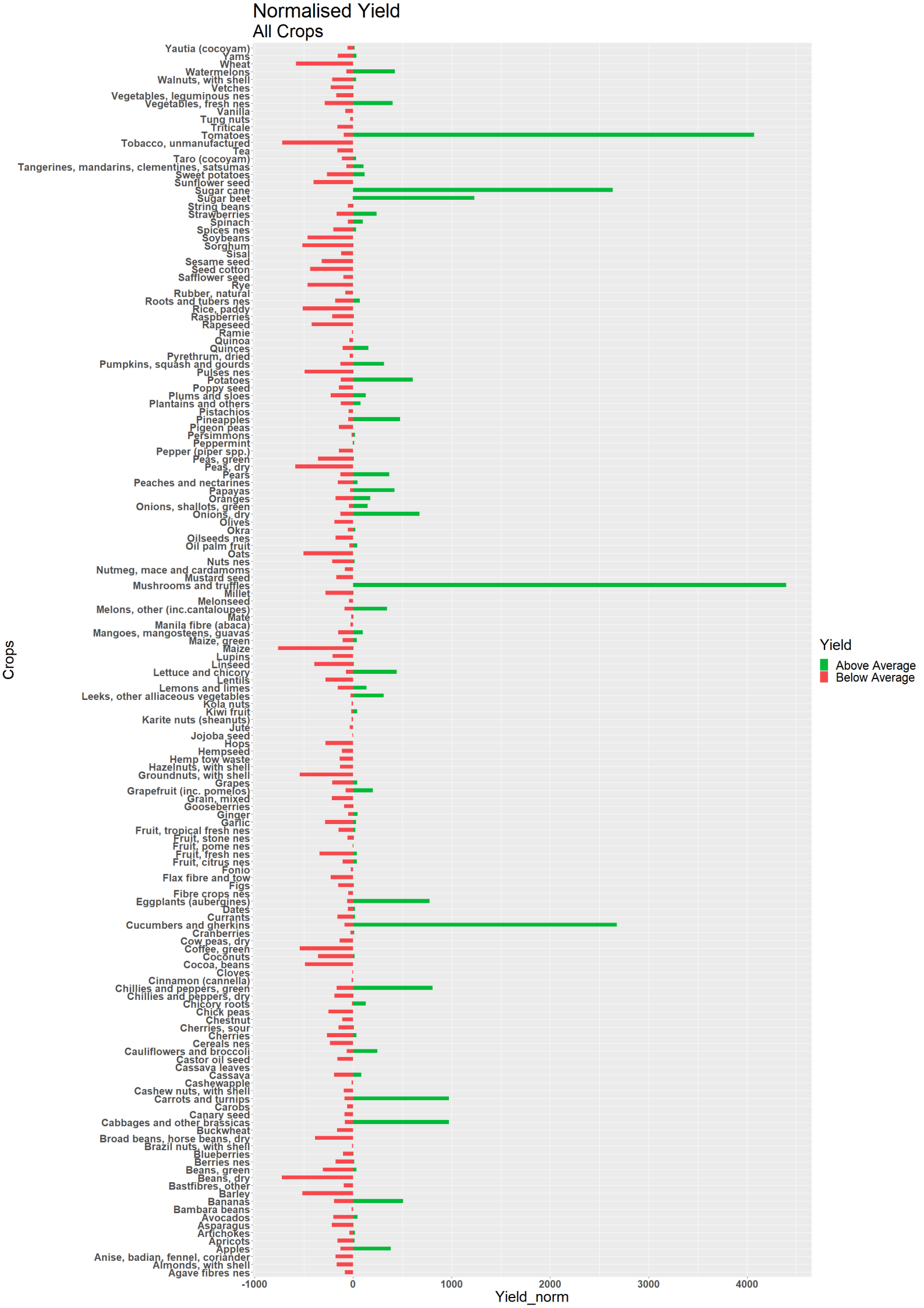



```
highyield<-crop_organized %>%
  filter(.,`Yield(tonnes/ha)`> 12.11)
lowyield<-crop_organized %>%
  filter(.,`Yield(tonnes/ha)`< 12.11)

ggplot(crop_organized,aes(x=`Area_harvested(ha)`, y =`Production(tonnes)`))+
  geom_point()+
  geom_smooth(method = "lm",se = FALSE)+
  geom_encircle(aes(x=`Area_harvested(ha)`, y=`Production(tonnes)`),
                data=highyield,
                color="green",
                size=2,
                expand=0.05) +
  geom_encircle(aes(x=`Area_harvested(ha)`, y=`Production(tonnes)`),
                data=lowyield,
                color="red",
                size=2,
                expand=0.05) +
  scale_y_continuous(labels = unit_format(unit = "M", scale = 1e-6))+
  scale_x_continuous(labels = unit_format(unit = "K", scale = 1e-3))+
  labs(title = "Coorelation Between Production and Area Harvested",subtitle =  "Production Vs Area Harvested")
```

WHAT IS THE NORMALIZED YIELD FOR EACH CROP ?

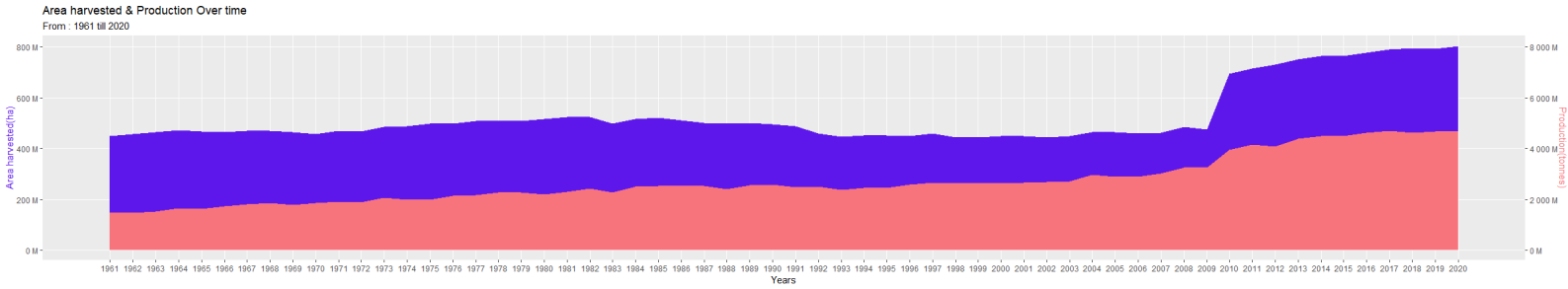
On this question I used diverging bars which can handle both positive and negative values, I wanted to make a full picture with all the crops in it thus making the chart a little big to work with, to see the full chart go to next page.

```
norm<-crop_organized
norm$yieldnorm<-
  round((norm$`Yield(tonnes/ha)`-mean(norm$`Yield(tonnes/ha)`,na.rm = TRUE))/sd(norm$`Yield(tonnes/ha)`,na.rm = TRUE),2)
norm$normtype<- ifelse(norm$yieldnorm < 0, "below", "above")
norm <- norm[order(norm$yieldnorm), ]

ggplot(norm, aes(x=Item, y=yieldnorm, label=yieldnorm)) +
  geom_bar(stat='identity', aes(fill=normtype), width=.5)  +
  scale_fill_manual(name="Yield",
                    labels = c("Above Average", "Below Average"),
                    values = c("above"="#00ba38", "below"="#f8474c")) +
  theme(plot.title = element_text(size=40),
        plot.subtitle = element_text(size = 35)
    ,axis.text.x = element_text(size = 20,face="bold")
    ,axis.text.y = element_text(size = 20,face="bold"))+
  xlab("Crops")+
  ylab("Yield_norm")+
  labs(subtitle="All Crops",
       title= "Normalised Yield") +
  coord_flip()
```

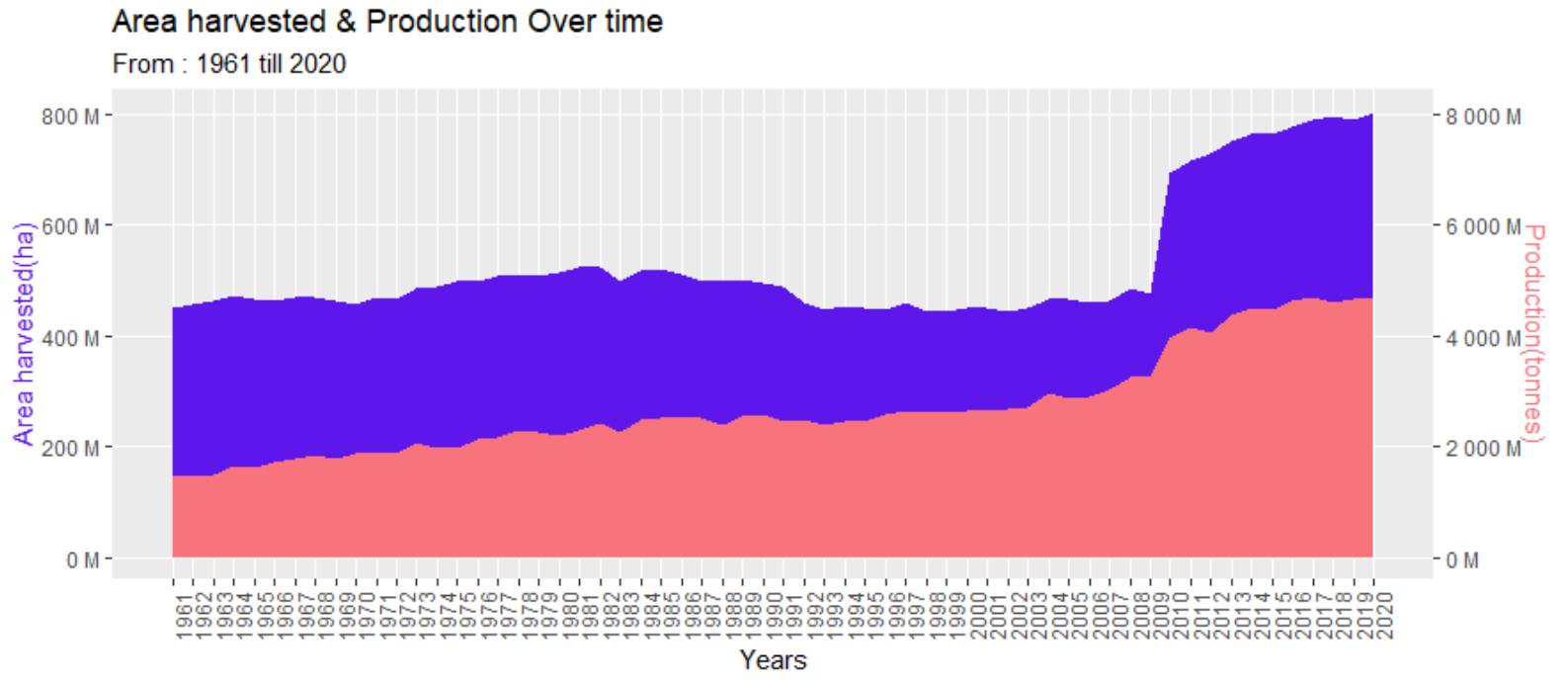

Normalised Yield
All Crops
Crops
Yautia (cocoyam)
Yams
Wheat
Watermelons
Walnuts, with shell
Vetches
Vegetables, leguminous nes
Vegetables, fresh nes
Vanilla
Tung nuts
Triticale
Tomatoes
Tobacco, unmanufactured
Tea
Taro (cocoyam)
Tangerines, mandarins, clementines, satsumas
Sweet potatoes
Sunflower seed
Sugar cane
Sugar beet
String beans
Strawberries
Spinach
Spices nes
Soybeans
Sorghum
Sisal
Sesame seed
Seed cotton
Safflower seed
Rye
Rubber, natural
Roots and tubers nes
Rice, paddy
Raspberries
Rapeseed
Ramie
Quinoa
Quinces
Pyrethrum, dried
Pumpkins, squash and gourds
Pulses nes
Potatoes
Poppy seed
Plums and sloes
Plantains and others
Pistachios
Pineapples
Pigeon peas
Persimmons
Peppermint
Pepper (piper spp.)
Peas, green
Peas, dry
Pears
Peaches and nectarines
Papayas
Oranges
Onions, shallots, green
Onions, dry
Olives
Okra
Oilseeds nes
Oil palm fruit
Oats
Nuts nes
Nutmeg, mace and cardamoms
Mustard seed
Mushrooms and truffles
Millet
Melonseed
Melons, other (inc.cantaloupes)
Mate
Manila fibre (abaca)
Mangoes, mangosteens, guavas
Maize, green
Maize
Lupins
Linseed
Lettuce and chicory
Lentils
Lemons and limes
Leeks, other alliaceous vegetables
Kola nuts
Kiwi fruit
Karite nuts (sheanuts)
Jute
Jojoba seed
Hops
Hempseed
Hemp tow waste
Hazelnuts, with shell
Groundnuts, with shell
Grapes
Grapefruit (inc. pomelos)
Grain, mixed
Gooseberries
Ginger
Garlic
Fruit, tropical fresh nes
Fruit, stone nes
Fruit, pome nes
Fruit, fresh nes
Fruit, citrus nes
Fonio
Flax fibre and tow
Figs
Fibre crops nes
Eggplants (aubergines)
Dates
Currants
Cucumbers and gherkins
Cranberries
Cow peas, dry
Coffee, green
Coconuts
Cocoa, beans
Cloves
Cinnamon (cannella)
Chillies and peppers, green
Chillies and peppers, dry
Chicory roots
Chick peas
Chestnut
Cherries, sour
Cherries
Cereals nes
Cauliflowers and broccoli
Castor oil seed
Cassava leaves
Cassava
Cashewapple
Cashew nuts, with shell
Carrots and turnips
Carobs
Canary seed
Cabbages and other brassicas
Buckwheat
Broad beans, horse beans, dry
Brazil nuts, with shell
Blueberries
Berries nes
Beans, green
Beans, dry
Bastfibres, other
Barley
Bananas
Bambara beans
Avocados
Asparagus
Artichokes
Apricots
Apples
Anise, badian, fennel, coriander
Almonds, with shell
Agave fibres nes
-1000
0
1000
2000
3000
4000
Yield_norm
Yield
Above Average
Below Average

WHAT ARE THE TREND OF AREA HARVESTED AND PRODUCTION OVER TIME?

This chart strengthens the same view as the scatter plot showing not only the relation between the area harvested and production but also the change over time where can see since the **year 2010** a **spike** in both area used and production.

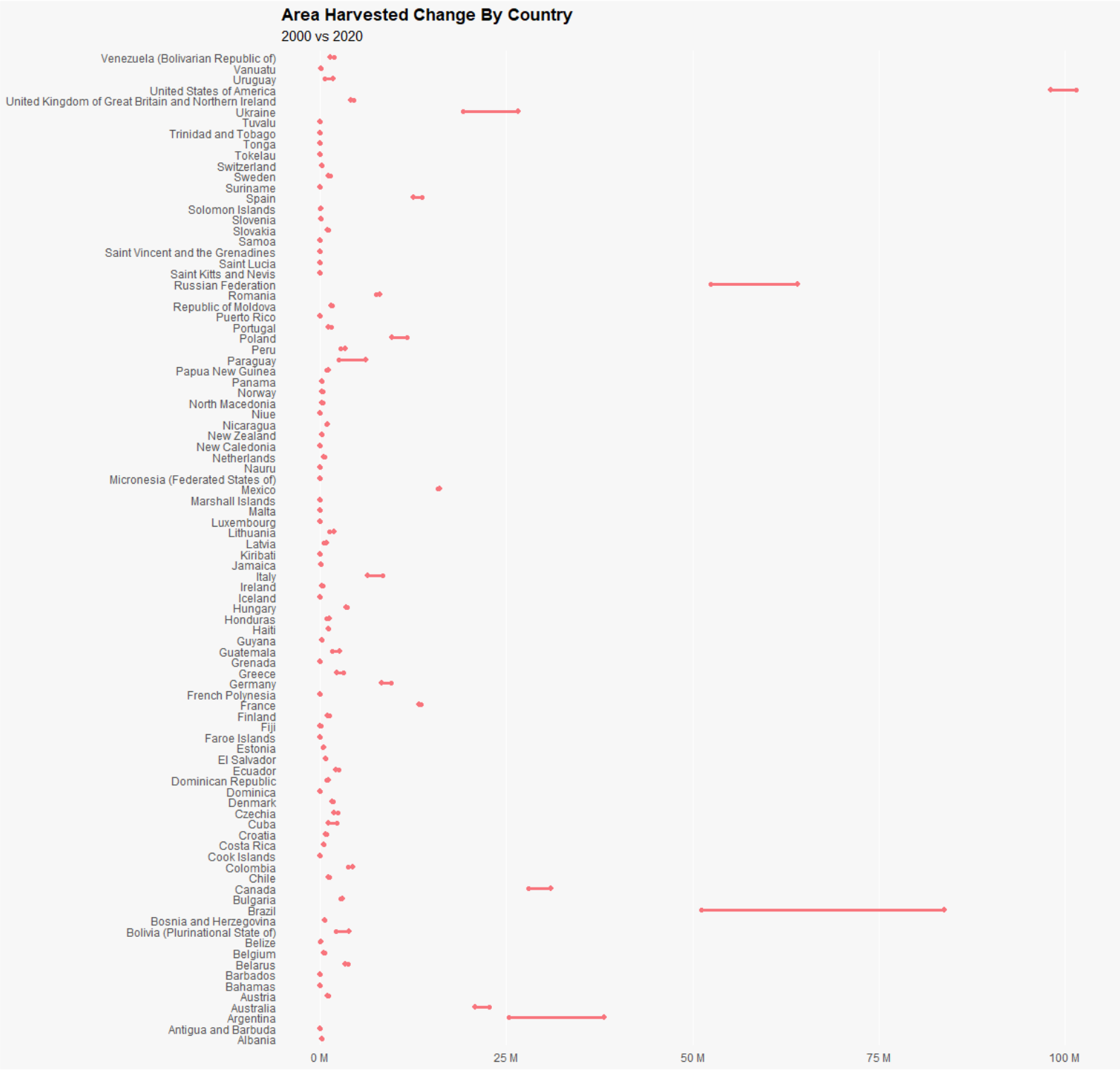


zoomed : changed the angle to 90




```
yersum<-crop_organized %>%
  group_by(.,year) %>%
  summarise(sumarea=sum(`Area_harvested(ha)`,na.rm = TRUE),sumprod=sum(`Production(tonnes)`,na.rm = TRUE))

coeff <- 10
bk<-1961:2020

ggplot(yersum,aes(x=year))+
  geom_area(aes(y=sumarea),fill="#5E17EB")+
  geom_area(aes(y=sumprod/coeff),fill="#F8747C")+
  scale_y_continuous(labels = unit_format(unit = "M", scale = 1e-6),
                     name = "Area harvested(ha)",
                     sec.axis = sec_axis(~.*coeff,name ="Production(tonnes)",
                                         labels = unit_format(unit = "M", scale = 1e-6)))+
  scale_x_continuous("Years", labels = as.character(yersum$year), breaks = bk)+
  theme(axis.title.y.left = element_text(colour = "#5E17EB"),
        axis.title.y.right = element_text(colour = "#F8747C"),
        axis.text.x = element_text(angle=0),panel.grid.minor = element_blank())+
  labs(title = "Area harvested & Production Over time",subtitle = paste0("From : ",mindate," till ",maxdate))
```

WHAT IS THE CHANGE IN AREA HARVESTED BETWEEN 2000 AND 2020?



code for last plot:

```
byareayear<-crop_organized %>%
  filter(.,year==2000 | year==2020) %>%
  group_by(.,Area,year) %>%
  summarise(sumarea=sum(`Area_harvested(ha)`,na.rm = TRUE)) %>%
  pivot_wider(.,names_from = year,
              values_from = sumarea,
              values_fn = function(x) mean(x,na.rm = TRUE),
              names_prefix = "year_" ) %>%
  mutate(Area= fct_reorder(Area, year_2020)) %>%
  drop_na()


byareayear$Area <-factor(byareayear$Area,
                         levels= as.character(byareayear$Area))

ggplot(byareayear, aes(x=year_2000, xend=year_2020, y=Area, group=Area)) +
  geom_dumbbell(color="#F8747C",
                size=1.3,
                point.colour.l="#0e668b") +
  scale_x_continuous(labels = unit_format(unit = "M", scale = 1e-6)) +
  labs(x=NULL,
       y=NULL,
       title="Area Harvested Change By Country",
       subtitle="2000 vs 2020") +
  theme(plot.title = element_text(face="bold"),
        plot.background=element_rect(fill="#f7f7f7"),
        panel.background=element_rect(fill="#f7f7f7"),
        panel.grid.minor=element_blank(),
        panel.grid.major.y=element_blank(),
        panel.grid.major.x=element_line(),
        axis.ticks=element_blank(),
        legend.position="top",
        panel.border=element_blank())
```

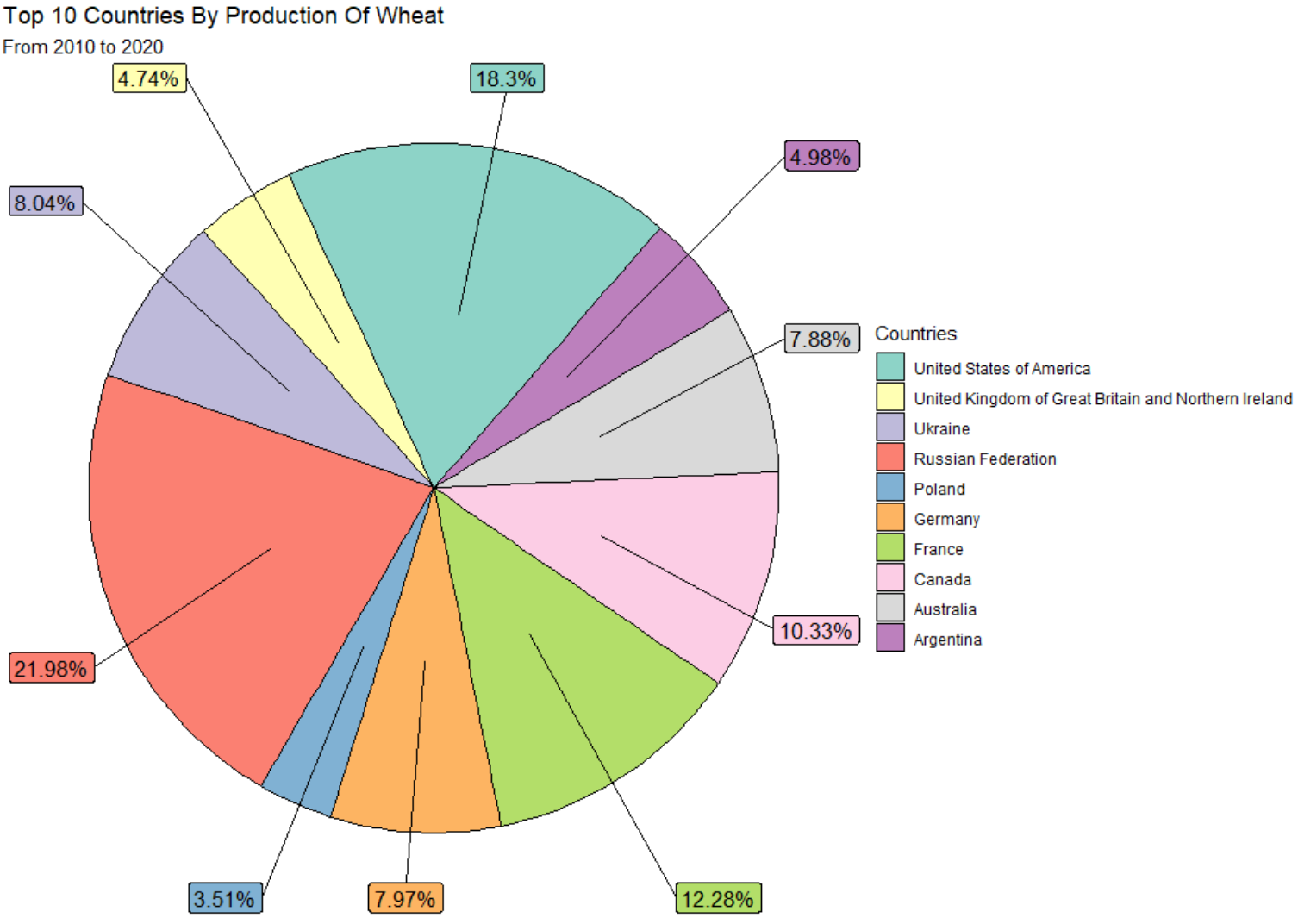
WHAT WE LEARNED SO FAR

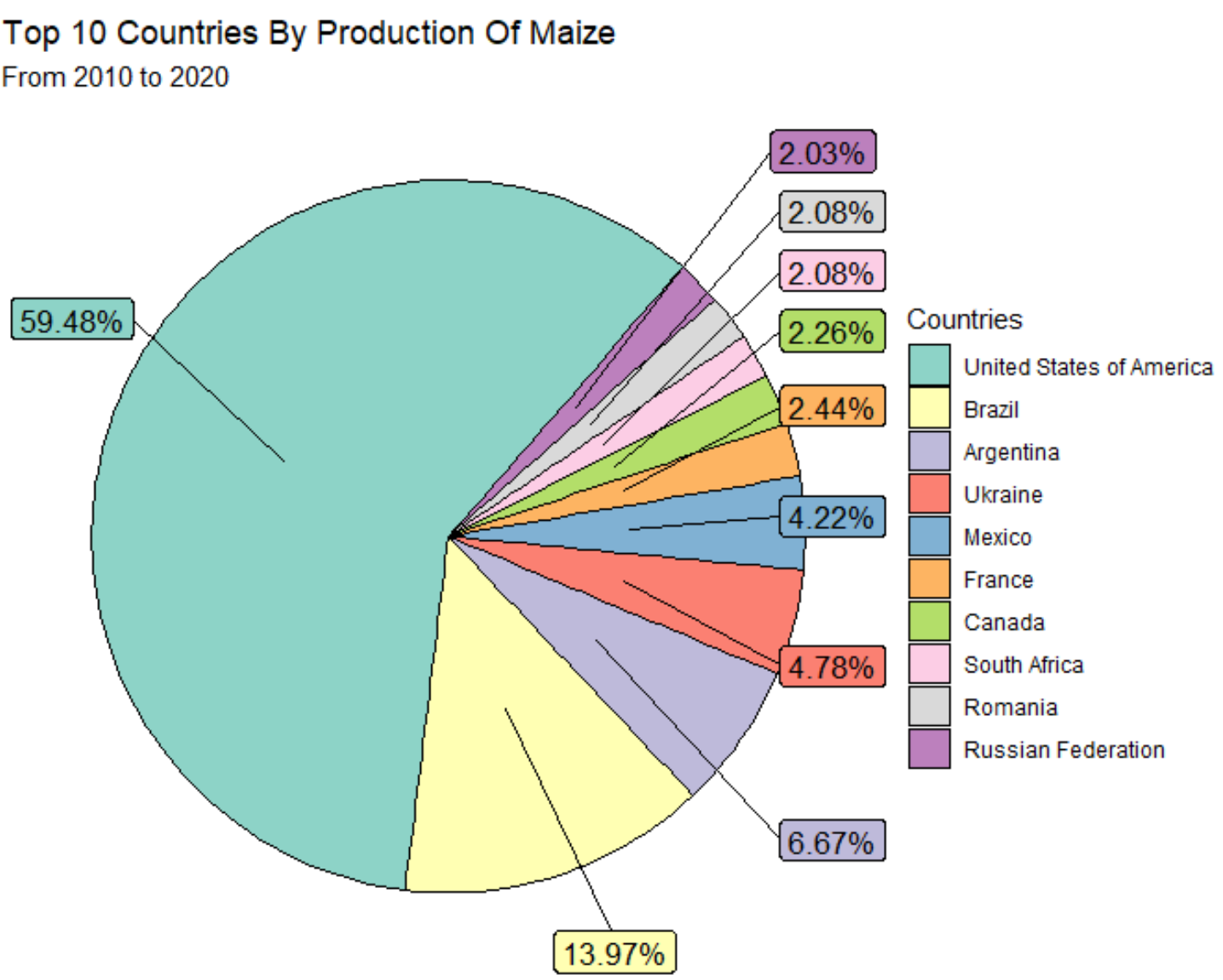
We can conclude that even with breakthroughs in the agriculture sector the efficiency is still nearly the same for over 60 years, the production did rise but also the area usage meaning the average yield is still the same we are only depleting resources at a faster rate, in just 60 years the area harvested doubled from 400 million hectares to 800 million hectares. And from our previous analysis, a big chunk of crops have a very low yield thus they need more resources like water, land, and of course fertilizers. While high-yield products are nowhere to be seen in high-produced crops, the problem may not only be how we make crops but also what we consume and demand to be put on the table.

To be sure of our analysis let's investigate further.

WHO ARE THE TOP 10 COUNTRIES THAT PRODUCE THE MOST PROMINENT CROPS BY AVERAGE PRODUCTION IN THE LAST 10 YEARS?

We will base our calculations on 2 types of crops, maize and wheat not only since they both are in our top 5 crops by average production and top 5 by average usage of harvesting area, but also because both of them are used by SEDAC as variables to analyze the 21th-century crops thanks to being an important crop for the majority of the world(except for Asia it is rice paddies) .






```
wheat10<-crop_organized %>%
  filter(.,crop_organized$Item =="Wheat", crop_organized$year>2010) %>%
  group_by(Area) %>%
  drop_na() %>%
  summarize(sumprod=sum(`Production(tonnes)`)) %>%
  arrange(desc(sumprod)) %>%
  top_n(sumprod,n = 10)

colnames(wheat10)<-c("Country","Total")
wheat10$percent_Total<-round(100*(wheat10$Total/sum(wheat10$Total)),2)
View(wheat10)
wheat11 <- wheat10 %>%
  mutate(csum = rev(cumsum(rev(percent_Total))),
         pos = percent_Total/2 + lead(csum, 1),
         pos = if_else(is.na(pos), percent_Total/2, pos))



ggplot(wheat10, aes(x = "" , y = percent_Total, fill = fct_inorder(Country))) +
  geom_col(width = 1, color = 1) +
  coord_polar(theta = "y",start = 7) +
  scale_fill_brewer(palette = "Set3") +
  geom_label_repel(data = wheat11,
                   aes(y = pos, label = paste0(percent_Total, "%")),
                   size = 4.5, nudge_x = 1, show.legend = FALSE) +
  guides(fill = guide_legend(title = "Countries")) +
  theme_void()+
  labs(title = "Top 10 Countries By Production Of Wheat",subtitle = "From 2010 to 2020")
```

FERTILIZERS & POPULATION :

In this part, we will tackle both the Fertilizers and Population datasets, then their relation to crop production.

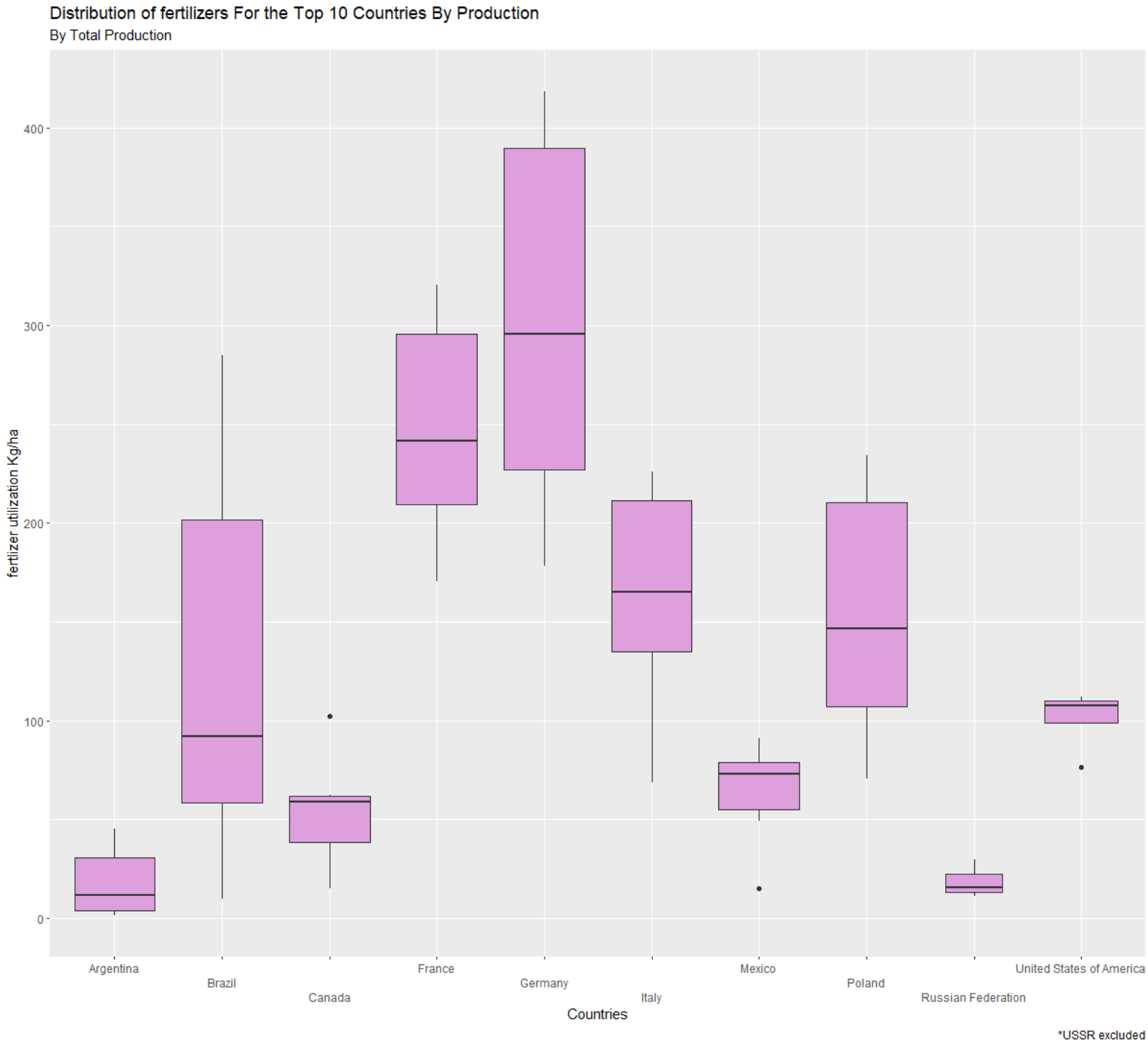
```
round(max(fert_organized$fertilizer_utilization,na.rm = TRUE))
min(fert_organized$fertilizer_utilization,na.rm = TRUE)
round(mean(fert_organized$fertilizer_utilization,na.rm = TRUE))
round(sd(fert_organized$fertilizer_utilization,na.rm = TRUE))
```

- The average fertilizers usage is **129 Kg/ha.**
- The max fertilizers usage is **3101 Kg/ha.**
- The min fertilizers usage is **0 Kg/ha.**
- The standard deviation for fertilizers usage is **204 Kg/ha.**

```
max(pop_organized$Population,na.rm = TRUE)
min(pop_organized$Population,na.rm = TRUE)
mean(pop_organized$Population,na.rm = TRUE)
sd(pop_organized$Population,na.rm = TRUE)
```

- The average population is **31,340,297** which is **thirty-one million three hundred forty thousand two hundred ninety-seven.**
- The max population is **1,877,902,324** which is **one billion eight hundred seventy-seven million nine hundred two thousand three hundred twenty-four.**
- The min population usage is **4377.**
- The standard deviation for population is **143,059,403** which is **one hundred forty-three million fifty-nine thousand four hundred three.**

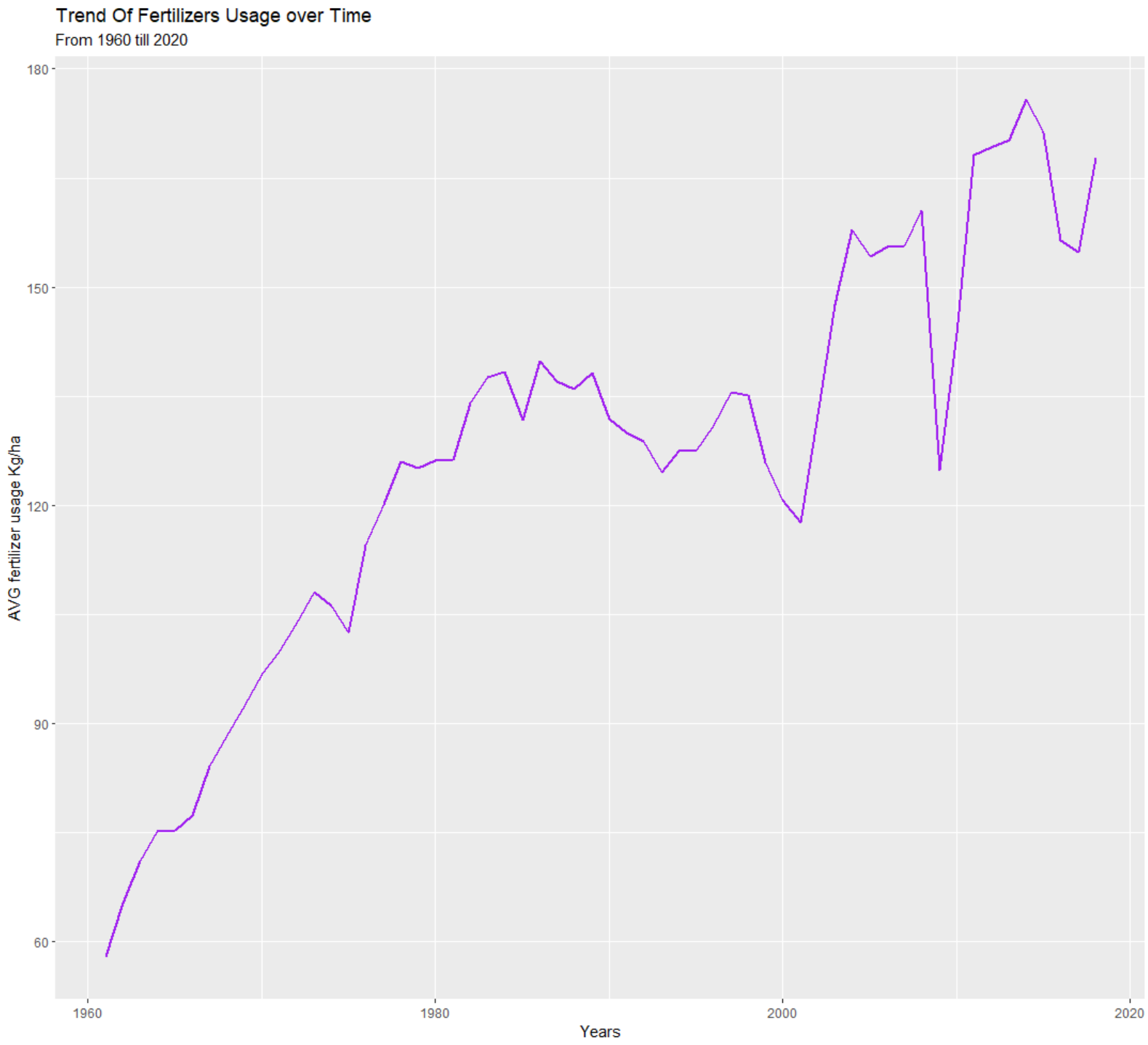
DISTRIBUTION OF FERTILIZERS FOR THE TOP 10 COUNTRIES BY PRODUCTION



code for last plot:

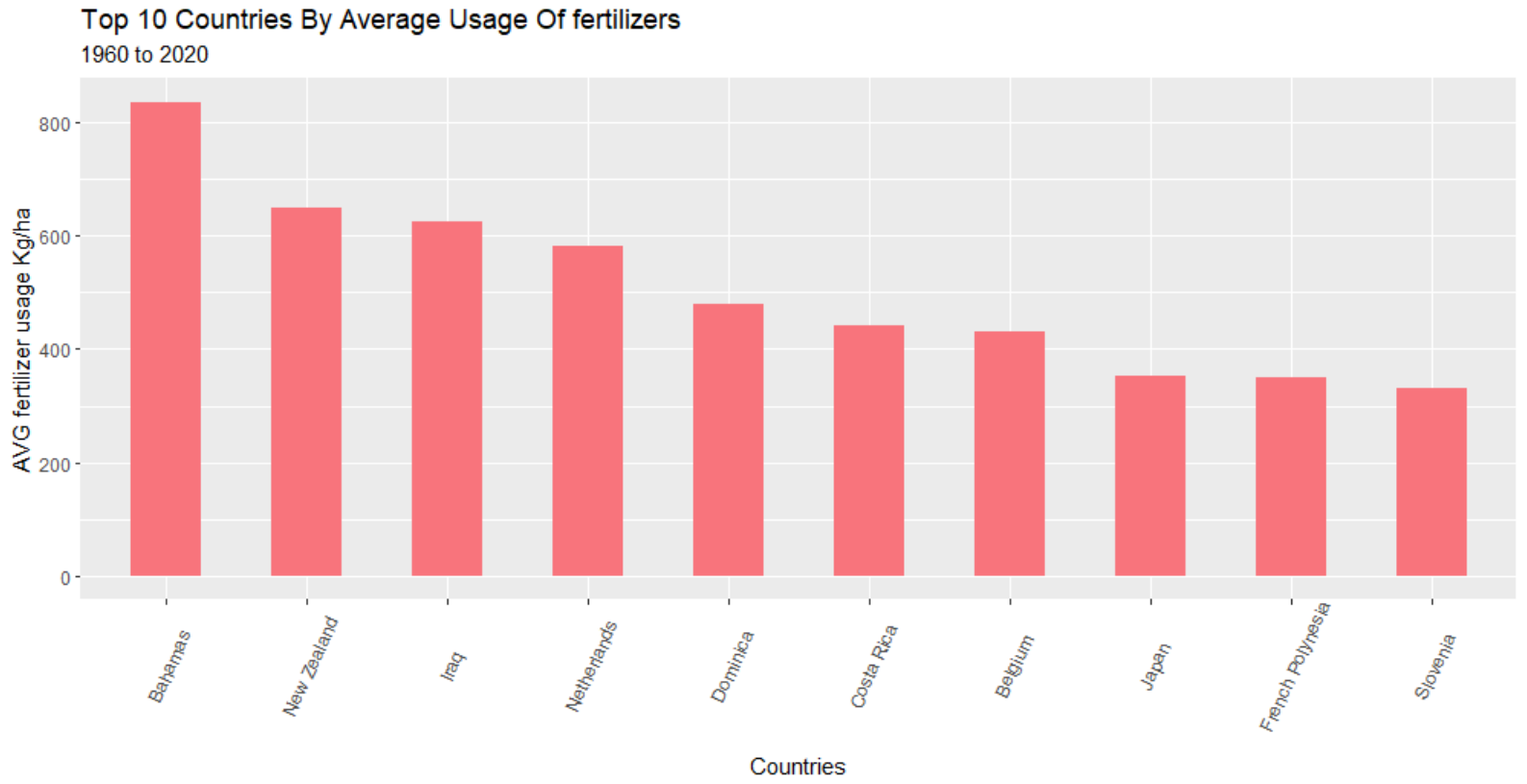
```
topprod<-crop_organized %>%
  filter(.,Area != "USSR") %>%
  group_by(.,Area) %>%
  summarise(.,sumprod=sum(`Production(tonnes)`,na.rm = TRUE)) %>%
  arrange(desc(sumprod)) %>%
  top_n(sumprod,n=10)

fer10<- fert_organized %>%
  filter(.,Country.Name == topprod$Area)
unique(fer10$Country.Name)


ggplot(fer10, aes(Country.Name, fer10$fertilizer_utilization))+
  geom_boxplot(varwidth=T, fill="plum") +
  labs(title="Distribution of fertilizers For the Top 10 Countries By Production ",
       subtitle="By Total Production",
       caption="*USSR excluded",
       x="Countries",
       y="fertlizer utilization Kg/ha")+
  scale_x_discrete(guide = guide_axis(n.dodge=3))
```

TREND OF FERTILIZERS USAGE OVER TIME




```
trefert<-fert_organized %>%
  group_by(.,year) %>%
  summarise(., avgfert=mean(fertilizer_utilization,na.rm=TRUE))

ggplot(trefert, aes(x=year)) +
  geom_line(aes(y=avgfert),size = 1,color= "purple") +
  labs(title="Trend Of Fertilizers Usage over Time",
       subtitle="From 1960 till 2020",
       y="AVG fertilizer usage Kg/ha",
       x= "Years")
```

TOP 10 COUNTRIES BY AVERAGE USAGE OF FERTILIZERS




```
topfert<-fert_organized %>%
  group_by(.,Country.Name) %>%
  summarise(.,avgfert=mean(fertilizer_utilization,na.rm=TRUE)) %>%
  arrange(desc(avgfert)) %>%
  top_n(avgfert,n = 10)

ggplot(topfert,aes(x=reorder(Country.Name,-avgfert),y=avgfert))+
  geom_bar(stat="identity",width=.5,fill="#F8747C")+
  labs(title = "Top 10 Countries By Average Usage Of fertilizers",
       subtitle = "1960 to 2020",x="Countries", y="AVG fertilizer usage Kg/ha")+
  theme(axis.text.x = element_text(angle=65,vjust = 0.6))
```

We can furthermore investigate these countries and see why they use high quantities of fertilizers.
To do that we will need to see if these countries have low or above-average production and yield.

- By average production all the top 10 countries have a poor production when compared to the international average

	Area	avgprod	above_avg
1	Bahamas	6767.958	FALSE
2	Belgium	293185.596	FALSE
3	Costa Rica	174936.907	FALSE
4	Dominica	3853.405	FALSE
5	French Polynesia	5132.064	FALSE
6	Netherlands	387734.217	FALSE
7	New Zealand	46454.835	FALSE
8	Slovenia	20698.963	FALSE

- By average yield, the story changes where only 1 out of 10 have a low average yield while other countries all have high average yield.

	Area	avgyield	above_avg
1	Bahamas	12.928822	TRUE
2	Belgium	57.300273	TRUE
3	Costa Rica	13.034149	TRUE
4	Dominica	7.449666	FALSE
5	French Polynesia	13.614388	TRUE
6	Netherlands	86.694407	TRUE
7	New Zealand	19.429654	TRUE
8	Slovenia	13.068499	TRUE

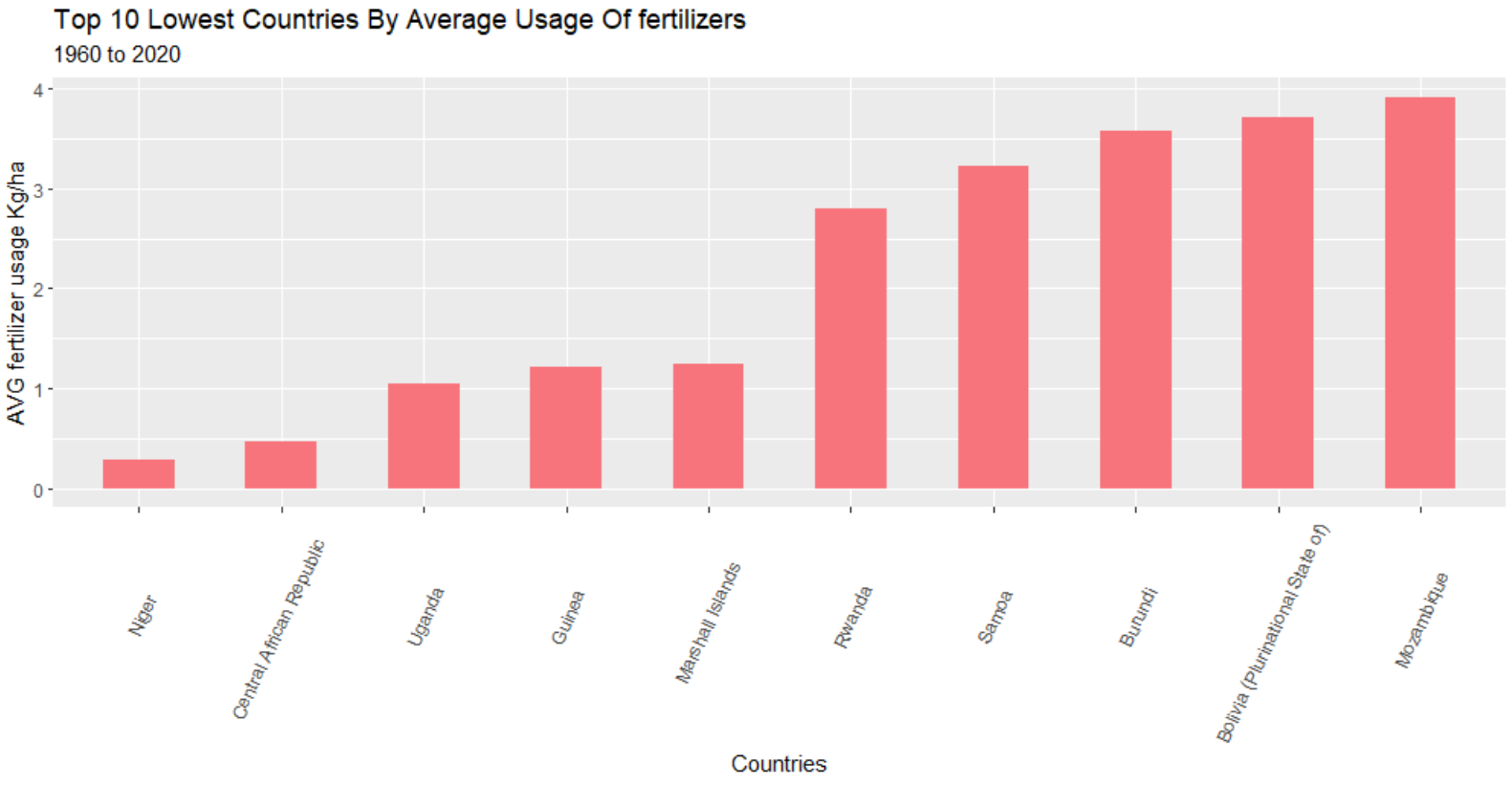
This can be explained by countries with low production are trying to optimize and get the most out of the small area of harvest available to them by using more fertilizers leading to high yields but still low production.
Let's see if these countries do have a low average harvested area.

- And indeed all theses countries have below average area of harvest.

	Area	avgarea	above_avg
1	Bahamas	487.3909	FALSE
2	Belgium	11963.8232	FALSE
3	Costa Rica	11058.3379	FALSE
4	Dominica	650.8190	FALSE
5	French Polynesia	983.6906	FALSE
6	Netherlands	15160.2501	FALSE
7	New Zealand	4813.8600	FALSE
8	Slovenia	2550.9793	FALSE

```
###### Compare avg production to top 10 countries by fert
fertavgprod<-crop_organized %>%
  filter(.,Area %in% topfert$Country.Name) %>%
  group_by(.,Area) %>%
  summarise(.,avgprod=mean(`Production(tonnes)`,na.rm=TRUE))

fertavgprod<- fertavgprod %>%
  mutate(above_avg=ifelse(avgprod>=656195.6,T,F))
View(fertavgprod)

###### Compare avg yield to top 10 countries by fert
fertavgyield<-crop_organized %>%
  filter(.,Area %in% topfert$Country.Name) %>%
  group_by(.,Area) %>%
  summarise(.,avgyield=mean(`Yield(tonnes/ha)`,na.rm=TRUE))

fertavgyield<- fertavgyield %>%
  mutate(above_avg=ifelse(avgyield>=12.11,T,F))
View(fertavgyield)

###### Compare avg yield to top 10 countries by fert
fertavgarea<-crop_organized %>%
  filter(.,Area %in% topfert$Country.Name) %>%
  group_by(.,Area) %>%
  summarise(.,avgarea=mean(`Area_harvested(ha)`,na.rm=TRUE))

fertavgarea<- fertavgarea %>%
  mutate(above_avg=ifelse(avgarea>=135912.5,T,F))
View(fertavgarea)
```

TOP 10 LOWEST COUNTRIES BY AVERAGE USAGE OF FERTILIZERS




```r
lowfert<-fert_organized %>%
  group_by(.,Country.Name) %>%
  summarise(.,avgfert=mean(fertilizer_utilization,na.rm=TRUE)) %>%
  arrange(desc(avgfert)) %>%
  top_n(avgfert,n = -10)

ggplot(lowfert,aes(x=reorder(Country.Name,avgfert),y=avgfert))+
  geom_bar(stat="identity",width=.5,fill="#F8747C")+
  labs(title = "Top 10 Lowest Countries By Average Usage Of fertilizers",
       subtitle = "1960 to 2020",x="Countries", y="AVG fertilizer usage Kg/ha")+
  theme(axis.text.x = element_text(angle=65,vjust = 0.6,hjust = 0.6))
```

Lets see how do the top 10 lowest countries by average fertilizers usage do.

We can definitely see that these countries don't go against common sense with low fertilizers usage they have both low low production and low yield, putting in mind that 40% of countries on this list have above the average area of harvest. what we can conclude is the following high fertilizer usage in a low area of harvest won't improve production overall but only the yield.

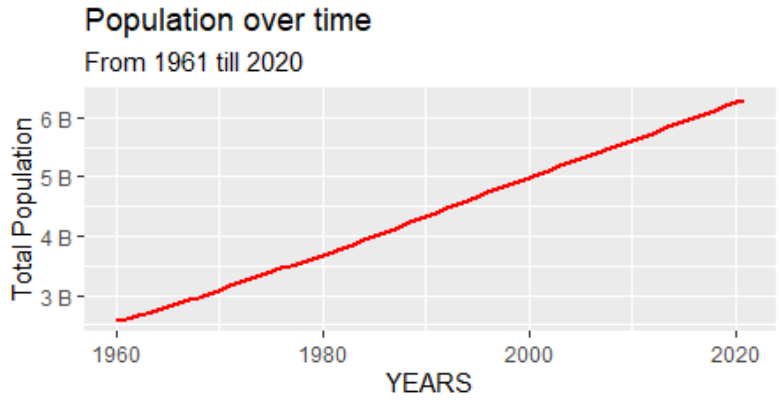
	Area	avgyield	above_avg
1	Bolivia (Plurinational State of)	5.198118	FALSE
2	Burundi	6.780256	FALSE
3	Central African Republic	3.047717	FALSE
4	Guinea	4.936235	FALSE
5	Marshall Islands	2.787413	FALSE
6	Mozambique	5.794603	FALSE
7	Niger	9.323728	FALSE
8	Rwanda	6.211061	FALSE
9	Samoa	8.728968	FALSE
10	Uganda	4.157190	FALSE

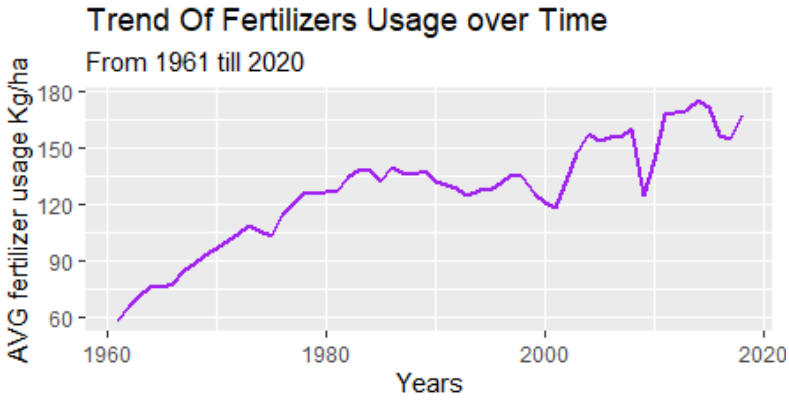
	Area	avgyield	above_avg
1	Bolivia (Plurinational State of)	5.198118	FALSE
2	Burundi	6.780256	FALSE
3	Central African Republic	3.047717	FALSE
4	Guinea	4.936235	FALSE
5	Marshall Islands	2.787413	FALSE
6	Mozambique	5.794603	FALSE
7	Niger	9.323728	FALSE
8	Rwanda	6.211061	FALSE
9	Samoa	8.728968	FALSE
10	Uganda	4.157190	FALSE

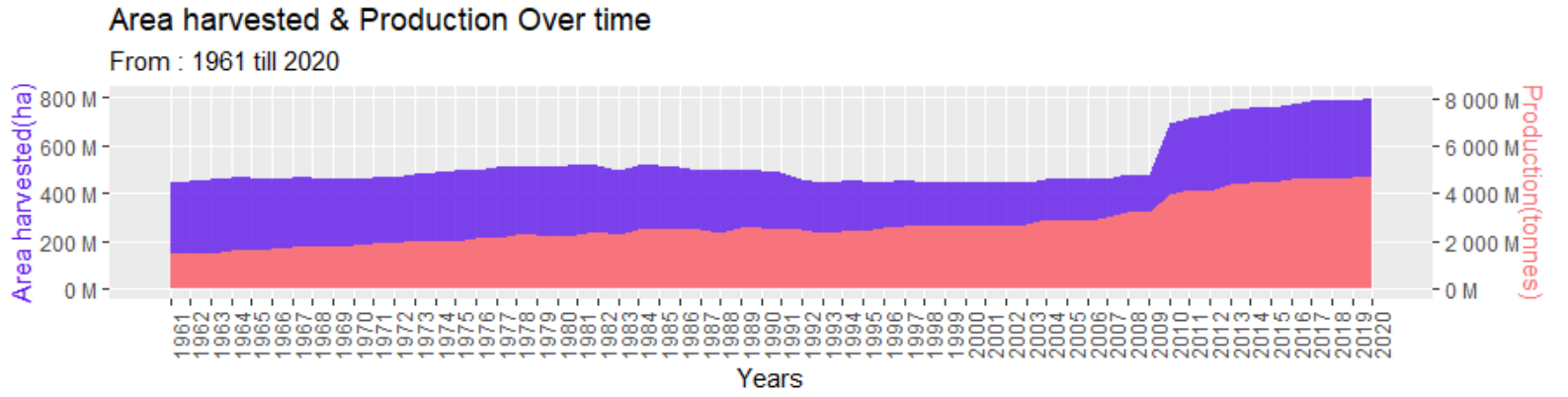
	Area	avgarea	above_avg
1	Bolivia (Plurinational State of)	29928.556	FALSE
2	Burundi	60175.582	FALSE
3	Central African Republic	29159.747	FALSE
4	Guinea	150711.050	TRUE
5	Marshall Islands	9043.267	FALSE
6	Mozambique	155592.800	TRUE
7	Niger	377133.769	TRUE
8	Rwanda	49445.155	FALSE
9	Samoa	2272.708	FALSE
10	Uganda	189313.437	TRUE

WHAT IS THE RELATIONSHIP BETWEEN FERTILIZER USAGE AND POPULATION?

With the high rise in population, there is a higher demand for food pushing countries to use more area in order to produce more and to keep up with the demand fertilizers seem to be a must in the way we are doing agriculture today or yesterday since the rates between these 4 variables give a notion of causation rather than correlation.








```r
popsum<-pop_organized %>%
  group_by(.,year) %>%
  summarise(.,sumpop=sum(Population,na.rm = TRUE))

po<-ggplot(popsum,aes(x=year))+
  geom_line(aes(y=sumpop),size= 1,color= "red")+
  scale_y_continuous(labels = unit_format(unit = "B", scale = 1e-9))+
  labs(title = "Population over time",subtitle = "From 1961 till 2020",x="YEARS",y="Total Population")

grid.arrange(arrangeGrob(po,fer, ncol=2, nrow=1),
             arrangeGrob(cr, ncol=1, nrow=1), heights=c(4,4))
```

5.SHARE:

In the ASK phase, we asked many questions for the purpose to understand and know the state of agriculture and where it's heading by measuring and aggregating different variables, visualizing a long frame or frames of data into graphs that give us factual statements but most importantly it gave us the answer to how do fertilizers play in the overall crop production and what is the most important variable in this equation.

Here is a statement of facts :

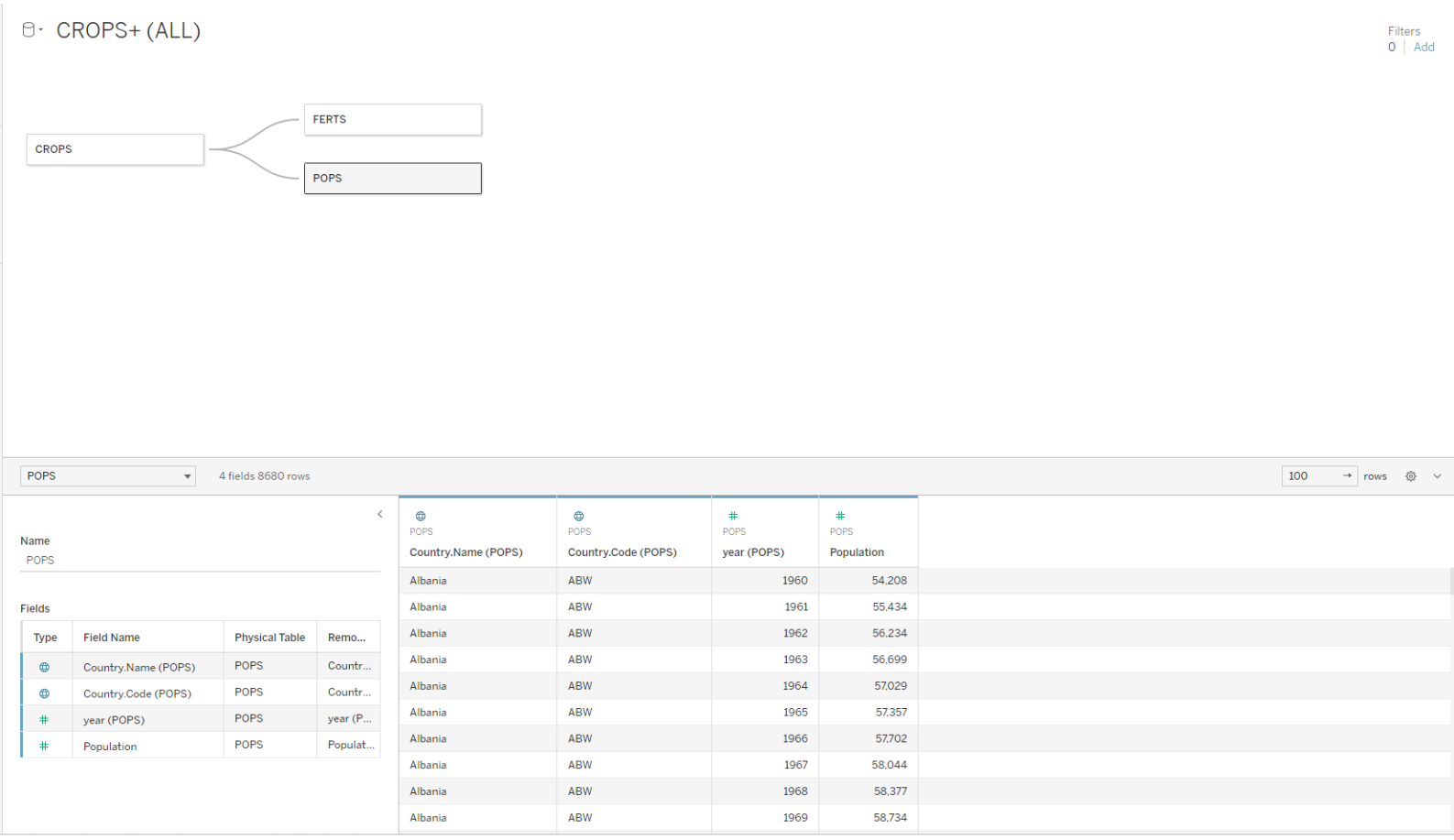
- **High production does not equal high yield the best example of this is the USSR in 1965 having the highest production yet a yield of 0.799 where the average is 12.11.**
- **High production crops usually use the highest amount of land while mostly having below-average yield**
- **Sugar cane and sugar beats are the only crops that have high production and above-average yield**
- **Low-yielding crops are usually low-calorie crops like vanilla or cacao.**
- **The bigger the area of harvest the higher the production**
- **The highest yielding crop is mushrooms and truffles yet still below average production rate**
- **Most of the crops are low yield**
- **The rate of production over the years is on a positive growth path but that also means the area of harvest is growing at nearly the same exact rate.**
- **Brazil had the largest growth in area usage between 2000 and 2020 the area used for harvest went from 51 Million hectares to 85 Million hectares that's 50% more area used in only 20 years**
- **The USA is the leader in the total area used for harvest as in 2020 about 105 Million hectares are being used that's about 7.92% from 2000 to 2020**
- **Yet although the USA uses the largest amount of land it still produces the highest out of the most 2 demanded or prominent crops maize and wheat while using around the same amount of fertilizers as the international average.**
- **In the top 10 countries by production Germany is the highest user of fertilizers well above the average still it's nowhere near the top 10 countries by average usage of fertilizers**
- **These top 10 countries with the highest fertilizer usage all have in common below-average production and below-average area harvest yet above-average yields, while the top 10 lowest countries by average fertilizers usage have below-average yield and below-average production with 60% of them having below-average area harvested.**
- **Outside other variables, we did not tackle on this EDA all of this leads to understanding that high usage of fertilizers doesn't automatically give you high production but rather the most important variable in this equation is the area as for fertilizer with right amounts are used as a catalyst for efficiency based on the crop trying to grow.**
- **The trend of fertilizers amount used is on the rise and it does not look to be going down any soon, especially with the steady rise of the population at a high rate which demands more food thus more area is going to be used for harvest**

I will use Tableau to create charts and a dashboard for more check my website or LinkedIn.

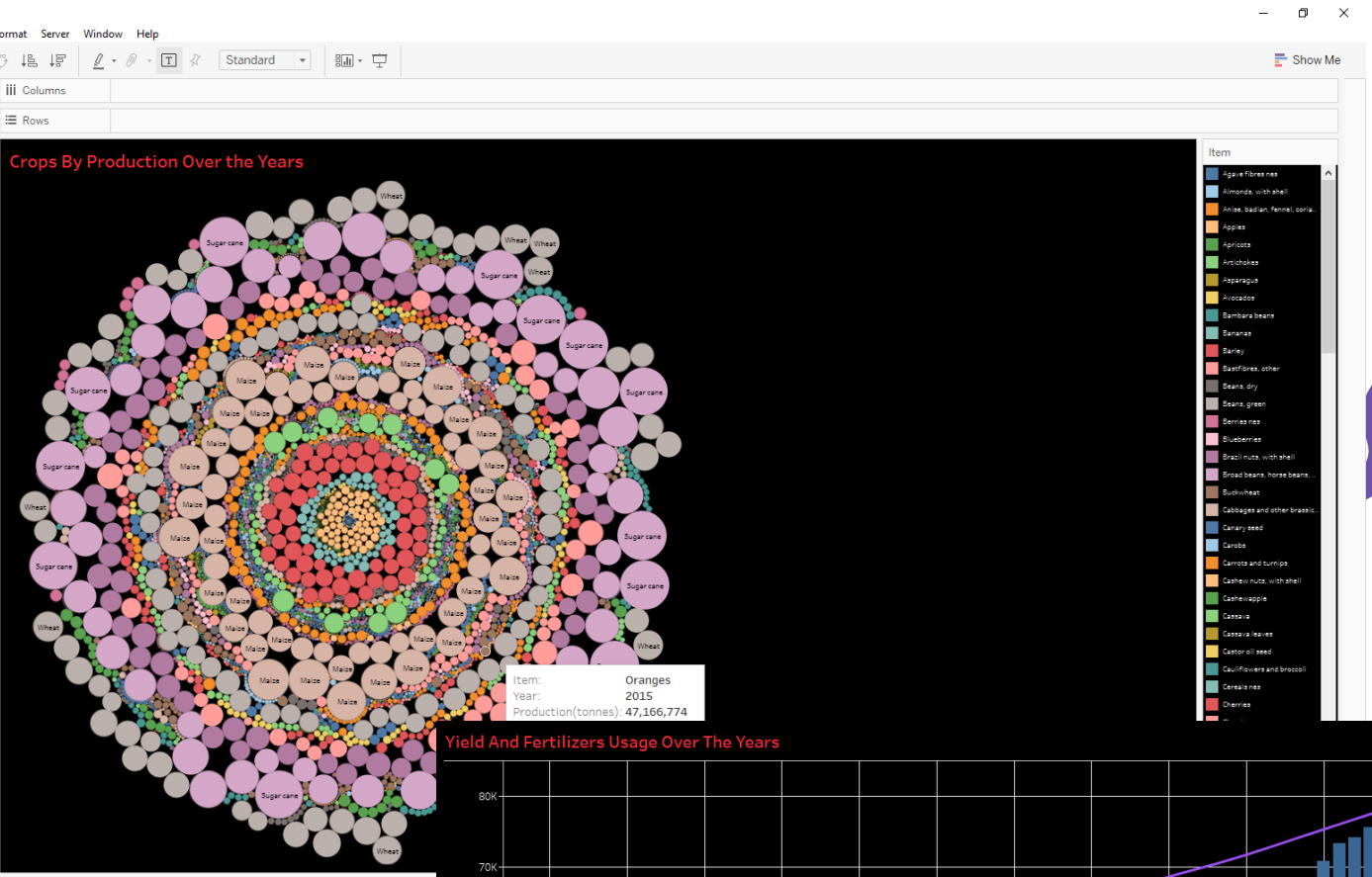
First we need to export our cleaned data frames as xlsx for that I used write_xlsx() function from the **writexl** library

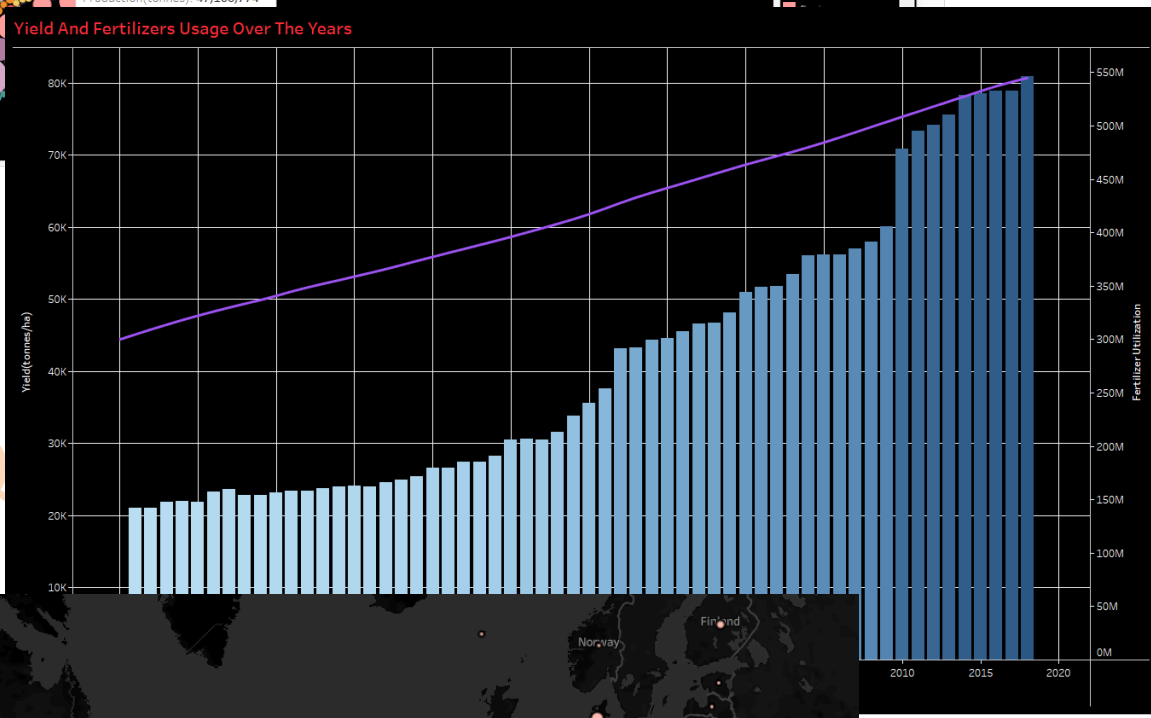
```
write_xlsx(crop_organized,"C://Users//Wadie//Desktop//cleaned_r//crops_r.xlsx")
write_xlsx(fert_organized,"C://Users//Wadie//Desktop//cleaned_r//ferts_r.xlsx")
write_xlsx(pop_organized,"C://Users//Wadie//Desktop//cleaned_r//pops_r.xlsx")
```

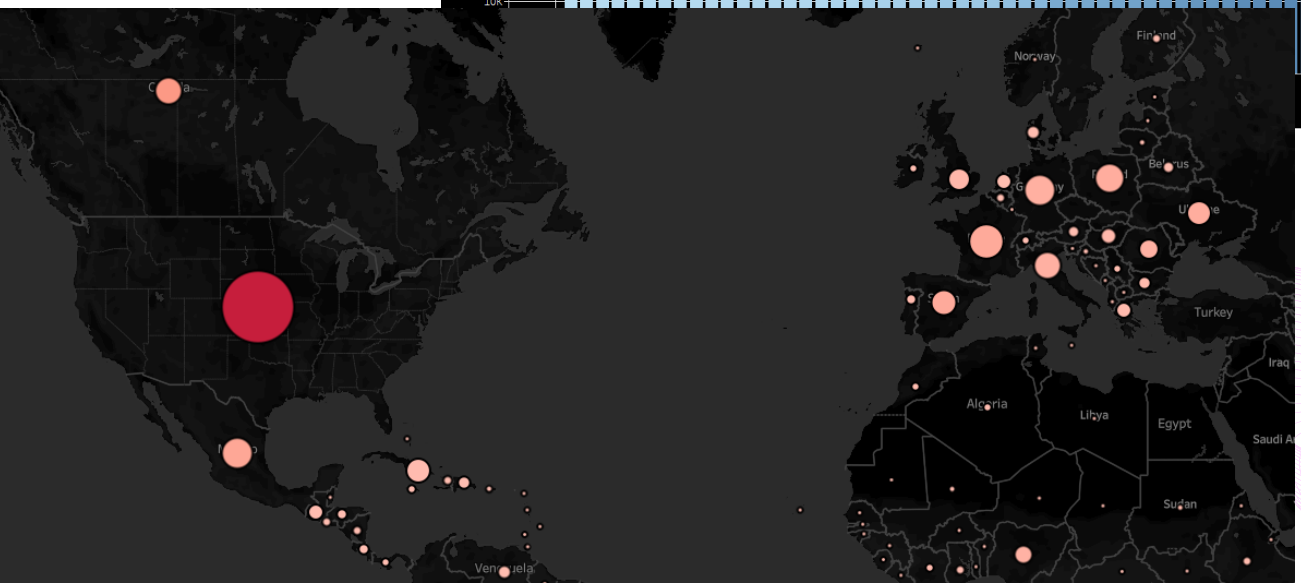
After that we open Tableau and link the 3 data frames on the common fields in Tableau that's called Field Name and Remote Field, it's like a SQL JOIN on a Primary key



Then we start creating and customizing charts. like these :



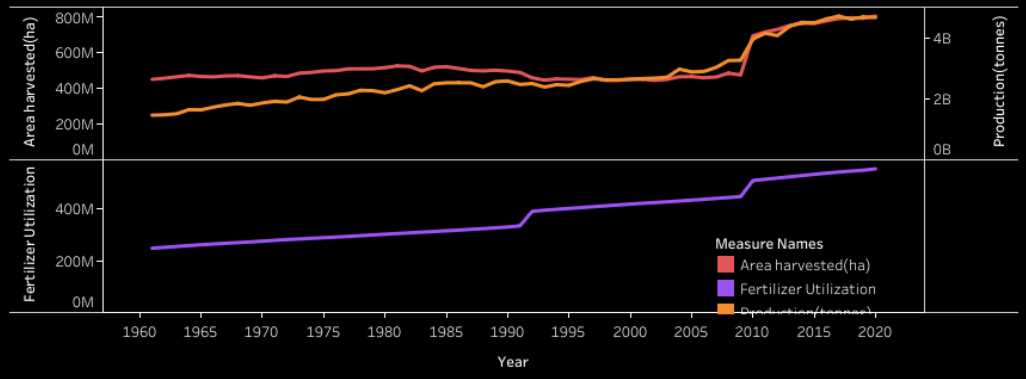


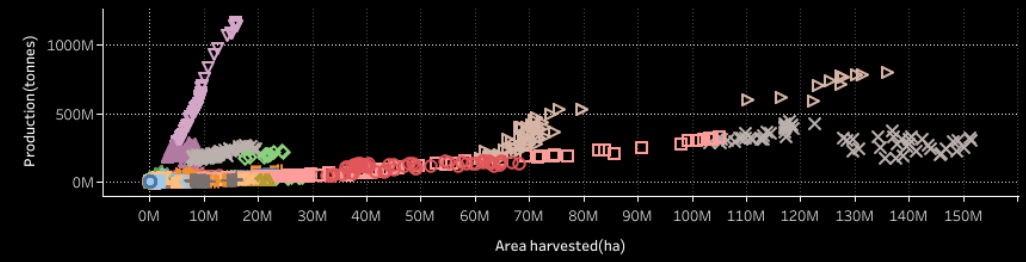


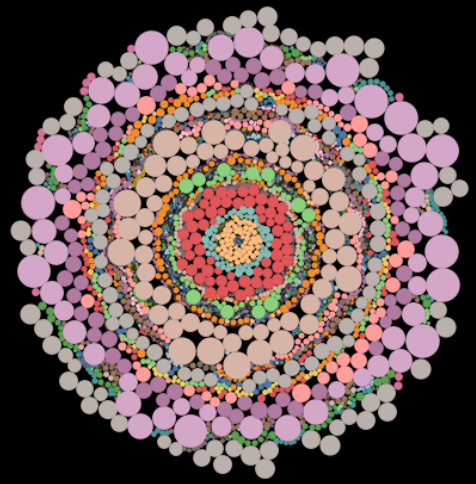
CROPS,FERTILIZERS, AND POPULATION

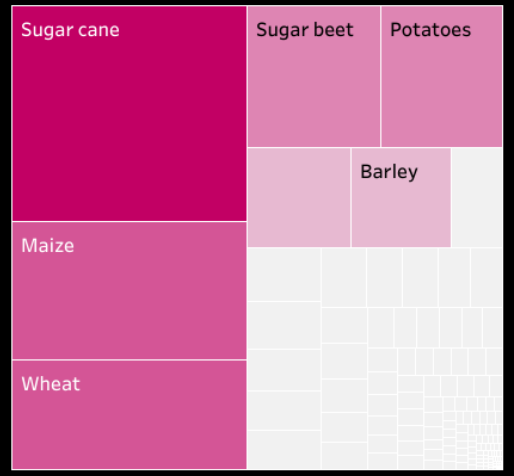
TABLEAU DASHBOARD EXAMPLE

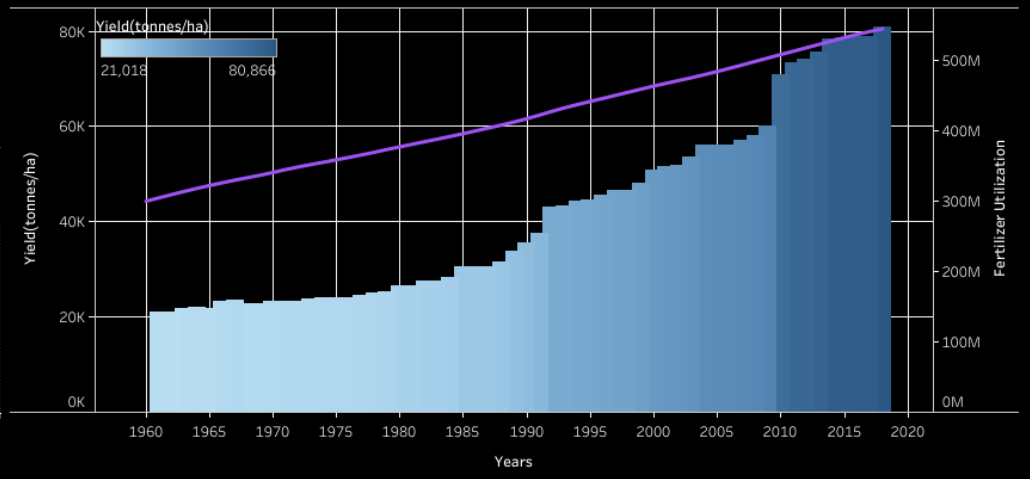
THE RATE OF PRODUCTION OVER THE YEARS IS ON A POSITIVE GROWING PATH BUT THAT ALSO MEANS THE AREA OF HARVEST IS GROWING AT NEARLY THE SAME EXACT RATE.











THE TREND OF FERTILIZERS AMOUNT USED ARE ON THE RISE AND IT DOES NOT LOOK TO BE GOING DOWN ANY SOON, ESPECIALLY WITH THE STEADY RISE OF THE POPULATION AT HIGH RATE WHICH DEMAND MORE FOOD THUS MORE AREA IS GOING TO BE USED FOR HARVEST

6.ACT:

My conclusion after this EDA is that there is vicious cycle where each variable is feeding and pushing the other, all of these variables are limited, there is only enough land on this planet thus a limited cap on how much we can produce fertilizers are not a magic powder that make food grow from thin air, it only allows the area of harvest to be more capable of containing more crops thus leading to a higher than average yield yet not higher than average production. The amount of land dictate how much food can be grown, but the only variable that isn't limited here is population yet it's the only variable that should not be controlled but I believe there is hope by :

- **Countries with small area of harvest should try and focus on high yield crops more.**
- **Neighboring countries can implement a crop rotation together where each year a country grow a different crop diminishing the usage of fertilizers.**
- **Marketing more high yield crops and food diversity.**
- **Calculating the right amounts of fertilizers based on the area available rather on the wanted production margin.**

These were some actions that can be taken not actually to break the cycle but to slow it until we improve on the arcane ways we grow food with, with new discoveries and inventions like hydroponics or even aquaponics where the area used is not only measured horizontally but vertically also double or quadrupling the area of harvest on the same patch of lands, but the best way for an individual to help with this is not to invent a way to plant food on the moon but to just not to waste it, in the United States, food waste is estimated at between 30-40 percent of the food supply. This estimate, based on estimates from USDA's Economic Research Service of 31 percent food loss at the retail and consumer levels, corresponded to approximately 133 billion pounds and $161 billion worth of food in 2010.

7.REFERENCES:

MAIN REFERENCES :

[1]. “FAO.[Crops and livestock products]. License: CC BY-NC-SA 3.0 IGO. Extracted from: [https://www.fao.org/faostat/en/#data/QCL]. Data of Access: [14-08-22].”

[2]. “FAO.[Fertilizer consumption (kilograms per hectare of arable land)]. License: CC BY-NC-SA 3.0 IGO. Extracted from: [https://www.fao.org/faostat/en/#data/QCL]. Data of Access: [14-08-22].”

[3]. United Nations Population Division. World Population Prospects: 2019 Revision. (2) Census reports and other statistical publications from national statistical offices, (3) Eurostat: Demographic Statistics, (4) United Nations Statistical Division. Population and Vital Statistics Reprot (various years), (5) U.S. Census Bureau: International Database, and (6) Secretariat of the Pacific Community: Statistics and Demography Programme.

INSPIRATION :

Originator: Anderson, W., W. Baethgen, F. Capitanio, P. Ciais, G. Rocca da Cunha, L. Goddard, B. Schauberger , K. Sonder, G. Podesta, M. van der Velde, L. You, and Y. Ru
Publication Date: 20220819
Publication Time:
Title:
Twentieth Century Crop Statistics, 1900-2017

RESOURCES USED:

Google BigQuery

stackoverflow

RStudio®

+ableau

THE COMPREHENSIVE R ARCHIVE NETWORK

READ MORE

DATA ANALYSIS

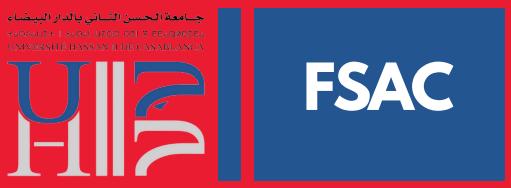
IF YOU ENJOYED THIS READING, THERE IS MORE !
READ HOW YOU CAN USE MULTIPLE TOOLS FOR
DATA ANALYSIS

WWW.WADIANORRI.COM

WADIA NORRI

CRM/ DATABASES AND NETWORK/
DATA ANALYST

This project was made for the purpose of sharing my technical skills as well as another important aspect that is essential to the path am growing on to become a Data Scientist which is communicating data into ideas that anyone can understand.



WADIA NORRI

EVERY PART OF THIS PROJECT WAS MADE BY WADIA NORRI, FROM THE MAKING OF THIS TEMPLATE TO THE IDEA OF THIS PROJECT TO DESIGNING IT AND LAST IMPLEMENTING IT, NO CODE WAS COPIED NOR THE TEXT IT'S ALL ORIGINAL WORK BY ME.

Creativity is intelligence
having fun.

ALBERT EINSTEIN

AN R EXPLORATORY DATA ANALYSIS PROJECT

27-08-2022