Strawberry

You Been PArk

2024-10-31

Strawberries: Data

This is a project about acquiring strawberry data from the USDA-NASS system and then cleaning, organizing, and exploring the data in preparation for data analysis. To get started, I acquired the data from the USDA NASS system and downloaded them in a csv.

Data cleaning and organization references

"An introduction to data cleaning with R" by Edwin de Jonge and Mark van der Loo

"Problems, Methods, and Challenges in Comprehensive Data Cleansing" by Heiko Müller and Johann-Christoph Freytag

Questions about Strawberries

How are the chemicals classified (e.g., fungicides, insecticides), and which categories are most prevalent? Do certain chemical classes correlate with higher productivity or specific outcomes (e.g., fruit size or yield)?

##Data Cleaning for use

```
# Load libraries
library(dplyr)
```

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

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

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

```
library(tidyr)
library(stringr)
library(readr)

# Load the dataset
strawberries_data <- read_csv("Strawberries25_v.csv")
```

```
## Rows: 12669 Columns: 21

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (15): Program, Period, Geo Level, State, State ANSI, Ag District, County...
## dbl  (2): Year, Ag District Code
## lgl  (4): Week Ending, Zip Code, Region, Watershed
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Rename columns to more readable names if necessary
colnames(strawberries_data) <- str_replace_all(colnames(strawberries_data), "\\s+", "_")

# Step 1: Replace Empty Strings and Placeholders in All Character Columns
strawberries_data <- strawberries_data %>%
  mutate(across(where(is.character), ~na_if(.x, ""))) %>%       # Convert "" to NA
  mutate(across(where(is.character), ~na_if(.x, "(D)"))) %>%    # Convert "(D)" to NA
  mutate(across(where(is.character), ~na_if(.x, "(NA)"))) %>%   # Convert "(NA)" to NA
  mutate(across(where(is.character), ~na_if(.x, "(L)")))        # Convert "(L)" to NA

# Step 2: Specific Cleaning for 'Value' Column
# Convert 'Value' to numeric, replacing placeholders with NA
strawberries_data <- strawberries_data %>%
  mutate(Value = case_when(
    Value %in% c("(D)", "(NA)", "(L)") ~ NA_real_,  # Set placeholders as NA
    TRUE ~ as.numeric(Value)  # Convert remaining entries to numeric
  ))
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Value = case_when(...)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
# Step 3: Fill Missing Values
# Use median for numerical columns and "Unknown" for categorical columns
strawberries_data <- strawberries_data %>%
  mutate(across(where(is.numeric), ~ ifelse(is.na(.), median(., na.rm = TRUE), .))) %>%
  mutate(across(where(is.character), ~ ifelse(is.na(.), "Unknown", .)))

# Drop irrelevant columns for chemical analysis using backticks for special characters
strawberries_data <- strawberries_data %>%
    select(-c(`Ag_District`, `Ag_District_Code`, `County`, `County_ANSI`, `Zip_Code`, `watershed_code`,

# Step 5: Filter Out Rows Based on Specific Conditions
strawberries_data <- strawberries_data %>%
  filter(!Region %in% c("Irrelevant_Region1", "Irrelevant_Region2"))

# Step 6: Clean and Organize Columns for Analysis
# Extract 'Use', 'Name', and 'Code' from 'Domain' and 'Domain_Category'
strawberry_clean <- strawberries_data %>%
  mutate(
    Use = case_when(
```

```r
      str_detect(Domain, "FUNGICIDE") ~ "FUNGICIDE",
      str_detect(Domain, "INSECTICIDE") ~ "INSECTICIDE",
      str_detect(Domain, "HERBICIDE") ~ "HERBICIDE",
      TRUE ~ NA_character_
    ),
    Name = str_extract(Domain_Category, "\\((.*?)\\)"),
    Name = str_replace_all(Name, " = \\d+", ""),
    Name = str_replace_all(Name, "[()]", ""),
    Code = str_extract(Domain_Category, "\\d+"),
    Code = str_trim(Code)
  ) %>%
  drop_na(Use, Name, Code) %>%
  select(-Domain, -Domain_Category)

# Step 7: Detect and Remove Duplicates
strawberries_data <- strawberries_data %>%
  distinct()

# Save the cleaned dataset
write_csv(strawberries_data, "strawberries_cleaned.csv")

# Print a summary of the cleaned dataset
summary(strawberries_data)
```

```
##    Program               Year          Period          Week_Ending
##  Length:7467        Min.   :2018   Length:7467        Mode:logical
##  Class :character   1st Qu.:2019   Class :character   NA's:7467
##  Mode  :character   Median :2022   Mode  :character
##                     Mean   :2021
##                     3rd Qu.:2022
##                     Max.   :2024
##   Geo_Level            State            State_ANSI          Region
##  Length:7467        Length:7467        Length:7467        Mode:logical
##  Class :character   Class :character   Class :character   NA's:7467
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##   Commodity          Data_Item            Domain          Domain_Category
##  Length:7467        Length:7467        Length:7467        Length:7467
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      Value
##  Min.   :  0.00
##  1st Qu.:  4.00
##  Median :  4.00
##  Mean   : 29.32
##  3rd Qu.: 10.00
##  Max.   :963.00
```

##Consistency Check and Missing Value Interpretation

```
# Load necessary libraries
library(dplyr)
library(stringr)

# Step 1: Replace Placeholders and Empty Strings in Character Columns
strawberry_clean <- strawberry_clean %>%
  mutate(across(where(is.character), ~ na_if(.x, ""))) %>%
  mutate(across(where(is.character), ~ na_if(.x, "(D)"))) %>%
  mutate(across(where(is.character), ~ na_if(.x, "(NA)"))) %>%
  mutate(across(where(is.character), ~ na_if(.x, "(L)")))

# Step 2: Check Initial NA Counts in All Columns
initial_na_counts <- colSums(is.na(strawberry_clean))
cat("Initial NA counts per column:\n")
```

```
## Initial NA counts per column:
```

```
print(initial_na_counts)
```

```
##     Program        Year      Period Week_Ending   Geo_Level       State
##           0           0           0        2805           0           0
##  State_ANSI      Region   Commodity   Data_Item       Value         Use
##           0        2805           0           0           0           0
##        Name        Code
##           0           0
```

```
# Step 3: Imputation for All Columns

# Impute numeric columns using mean or median, grouped by `Use` and `State`
strawberry_clean <- strawberry_clean %>%
  group_by(Use) %>%
  mutate(across(where(is.numeric), ~ ifelse(is.na(.), mean(., na.rm = TRUE), .))) %>%
  ungroup() %>%
  group_by(Use, State) %>%
  mutate(across(where(is.numeric), ~ ifelse(is.na(.), mean(., na.rm = TRUE), .))) %>%
  ungroup()

# Impute character columns with "Unknown" if NA
strawberry_clean <- strawberry_clean %>%
  mutate(across(where(is.character), ~ ifelse(is.na(.), "Unknown", .)))

# Handle logical columns with default values or remove unnecessary ones
# Fill logical NA with FALSE where needed or drop these columns if appropriate
strawberry_clean <- strawberry_clean %>%
  mutate(across(where(is.logical), ~ ifelse(is.na(.), FALSE, .)))

# Step 4: Final NA Check
final_na_counts <- colSums(is.na(strawberry_clean))
cat("Final NA counts per column:\n")
```

```
## Final NA counts per column:
```

```r
print(final_na_counts)
```

```
##     Program        Year      Period Week_Ending   Geo_Level       State
##           0           0           0           0           0           0
##  State_ANSI      Region   Commodity   Data_Item       Value         Use
##           0           0           0           0           0           0
##        Name        Code
##           0           0
```

```r
# Final Summary
summary(strawberry_clean)
```

```
##    Program               Year         Period          Week_Ending
##  Length:2805        Min.   :2018   Length:2805        Mode :logical
##  Class :character   1st Qu.:2019   Class :character   FALSE:2805
##  Mode  :character   Median :2021   Mode  :character
##                     Mean   :2020
##                     3rd Qu.:2023
##                     Max.   :2023
##   Geo_Level            State            State_ANSI          Region
##  Length:2805        Length:2805        Length:2805        Mode :logical
##  Class :character   Class :character   Class :character   FALSE:2805
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##   Commodity          Data_Item             Value              Use
##  Length:2805        Length:2805        Min.   :  0.017   Length:2805
##  Class :character   Class :character   1st Qu.:  3.200   Class :character
##  Mode  :character   Mode  :character   Median :  4.000   Mode  :character
##                                        Mean   : 14.110
##                                        3rd Qu.:  4.000
##                                        Max.   :900.000
##      Name               Code
##  Length:2805        Length:2805
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

Answering Q1. I am going to be using bar charts to show which chemicals are used, as well as what chemicals are in which category, and frequency for each chemicals.

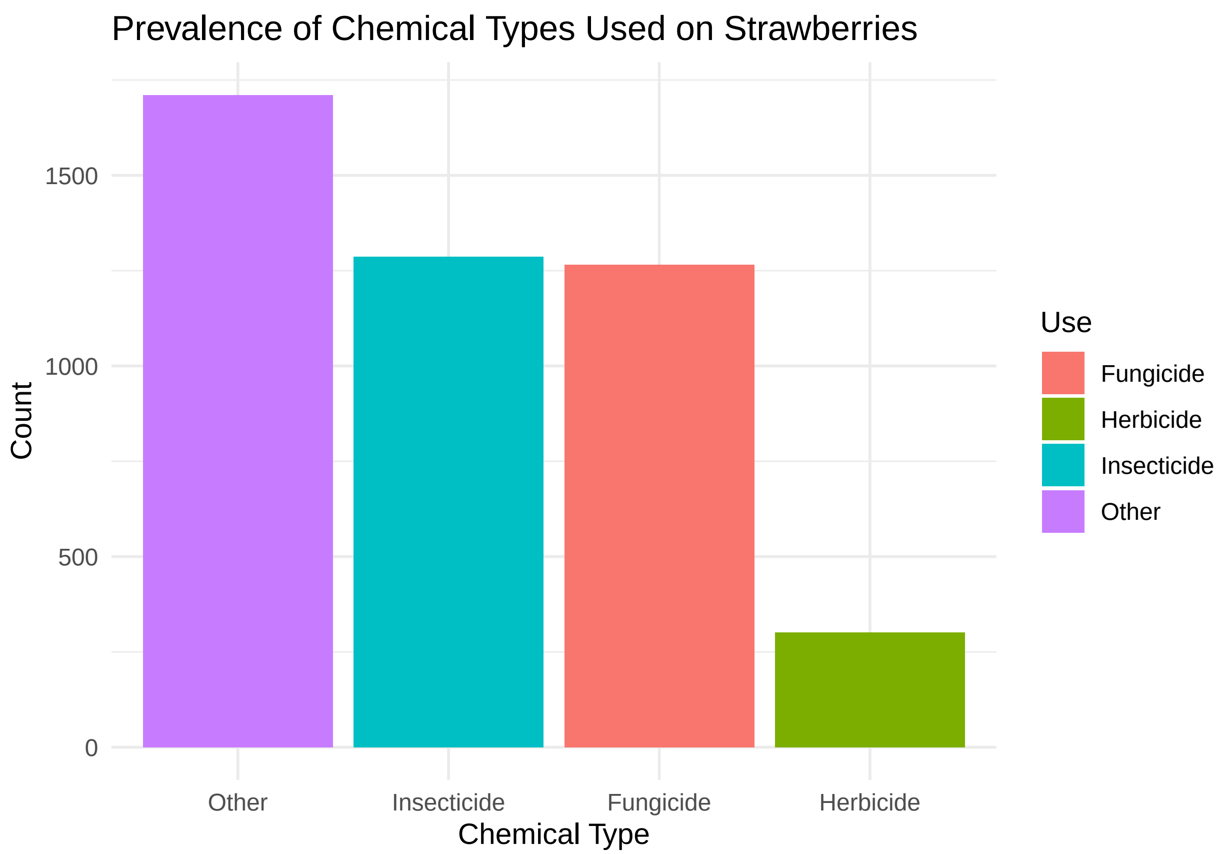
```r
# Load necessary libraries
library(dplyr)
library(ggplot2)
library(readr)
library(stringr)

# Load the cleaned dataset
strawberry_clean <- read_csv("strawberries_cleaned.csv")
```

```
## Rows: 7467 Columns: 13
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (9): Program, Period, Geo_Level, State, State_ANSI, Commodity, Data_Item...
## dbl (2): Year, Value
## lgl (2): Week_Ending, Region
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Extract chemical classifications (Use) from 'Domain' column
strawberry_clean <- strawberry_clean %>%
  mutate(
    Use = case_when(
      str_detect(Domain, "FUNGICIDE") ~ "Fungicide",
      str_detect(Domain, "INSECTICIDE") ~ "Insecticide",
      str_detect(Domain, "HERBICIDE") ~ "Herbicide",
      TRUE ~ "Other"
    ),
    # Extract specific chemical names from the 'Domain Category' column
    Chemical_Name = str_extract(`Domain_Category`, "\\((.*?)\\)"),
    Chemical_Name = str_replace_all(Chemical_Name, "[()]", "")  # Remove parentheses
  )

# Filter out rows where 'Use' or 'Chemical_Name' are NA
strawberry_clean <- strawberry_clean %>%
  filter(!is.na(Use) & !is.na(Chemical_Name))

# Count the prevalence of each chemical category
chemical_summary <- strawberry_clean %>%
  group_by(Use) %>%
  summarise(Count = n()) %>%
  arrange(desc(Count))

# Print the summary of chemical types
print(chemical_summary)
```

```
## # A tibble: 4 x 2
##   Use         Count
##   <chr>       <int>
## 1 Other        1711
## 2 Insecticide  1286
## 3 Fungicide    1266
## 4 Herbicide     301
```

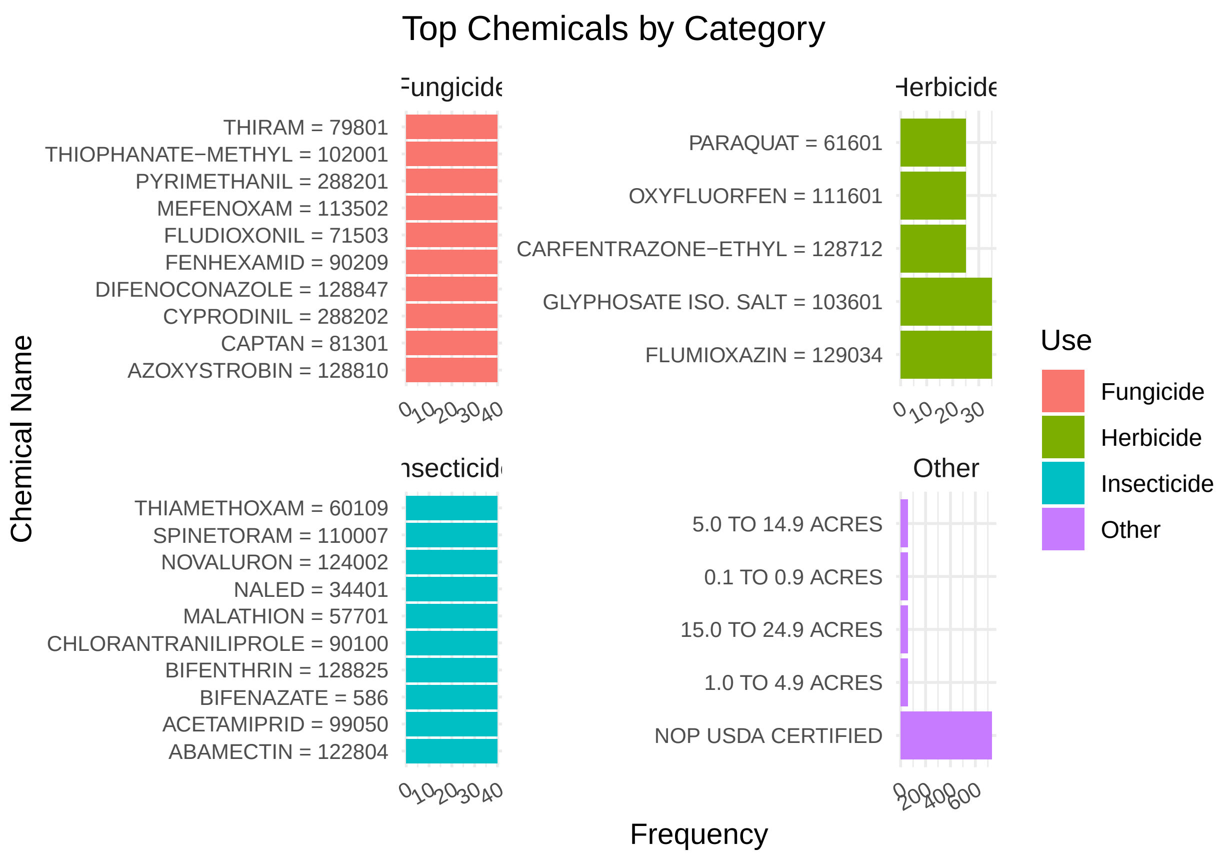
```
# Visualization: Bar chart of chemical types
ggplot(chemical_summary, aes(x = reorder(Use, -Count), y = Count, fill = Use)) +
  geom_bar(stat = "identity") +
  labs(title = "Prevalence of Chemical Types Used on Strawberries",
       x = "Chemical Type",
       y = "Count") +
  theme_minimal()
```




```
# Visualization: Top chemicals within each category
top_chemicals <- strawberry_clean %>%
  group_by(Use, Chemical_Name) %>%
  summarise(Frequency = n()) %>%
  arrange(desc(Frequency)) %>%
  slice_max(Frequency, n = 5)  # Top 5 chemicals per category
```

```
## `summarise()` has grouped output by 'Use'. You can override using the `.groups`
## argument.
```

```
ggplot(top_chemicals, aes(x = reorder(Chemical_Name, -Frequency), y = Frequency, fill = Use)) +
  geom_bar(stat = "identity") +
  facet_wrap(~ Use, scales = "free", nrow = 2) +  # Arrange categories in 2 rows for better spacing
  labs(title = "Top Chemicals by Category",
       x = "Chemical Name",
       y = "Frequency") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 30, hjust = 1, size = 8),  # Adjust text angle and size
        axis.text.y = element_text(size = 8),
        strip.text = element_text(size = 10)) +  # Increase facet label size for clarity
  coord_flip()  # Flip coordinates for horizontal bars
```



As shown, not including other chemicals, Insecticides are the most commonly used chemical type, fungicide being similar bbut a bit smaller and herbicide being used the least.

https://www.cambridge.org/core/journals/weed-technology/article/weed-control-with-and-strawberry-tolerance-to-herbicides-applied-through-drip-irrigation/77FBD1F590F3401C449ACAD43FE1B1DD

This website gives me reasons why herbicides are used the least. Strawberries are sensitive to herbicides, leading to less use of herbicides. For example, oxyfluorfen should be very carefully applied, or else, this could eventually harm the crop.

I would like to look deeper into how other chemicals are preferred for growing strawberries.

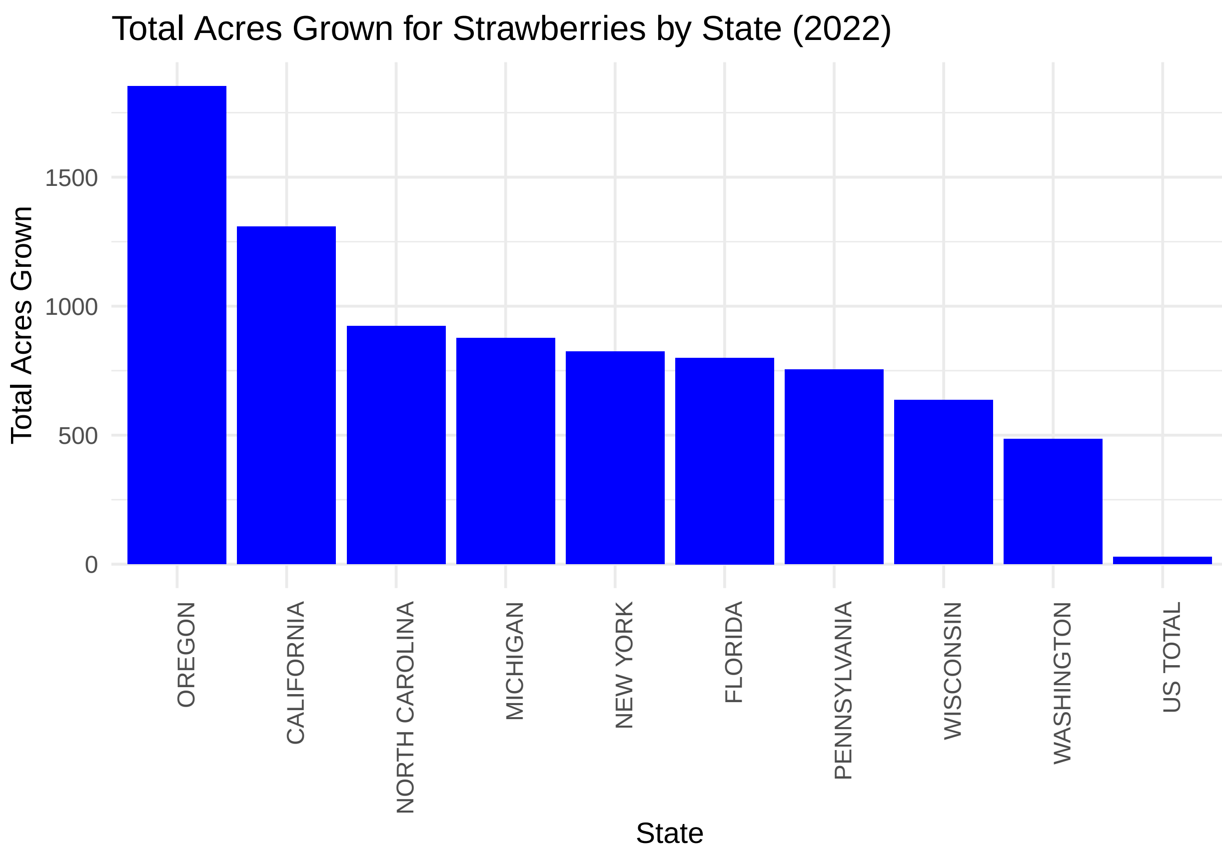
##Total Acres Grown by state We will now look at the total Acre of production in Strawberries.

```r
# Convert 'Value' column to numeric, replacing '(D)' or other placeholders with NA
strawberry_clean <- strawberry_clean %>%
  mutate(Value = ifelse(Value %in% c("(D)", "(NA)"), NA, as.numeric(Value)))

acres_data <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES GROWN"))

# Display the aggregated data to verify its content
print(acres_data)
```

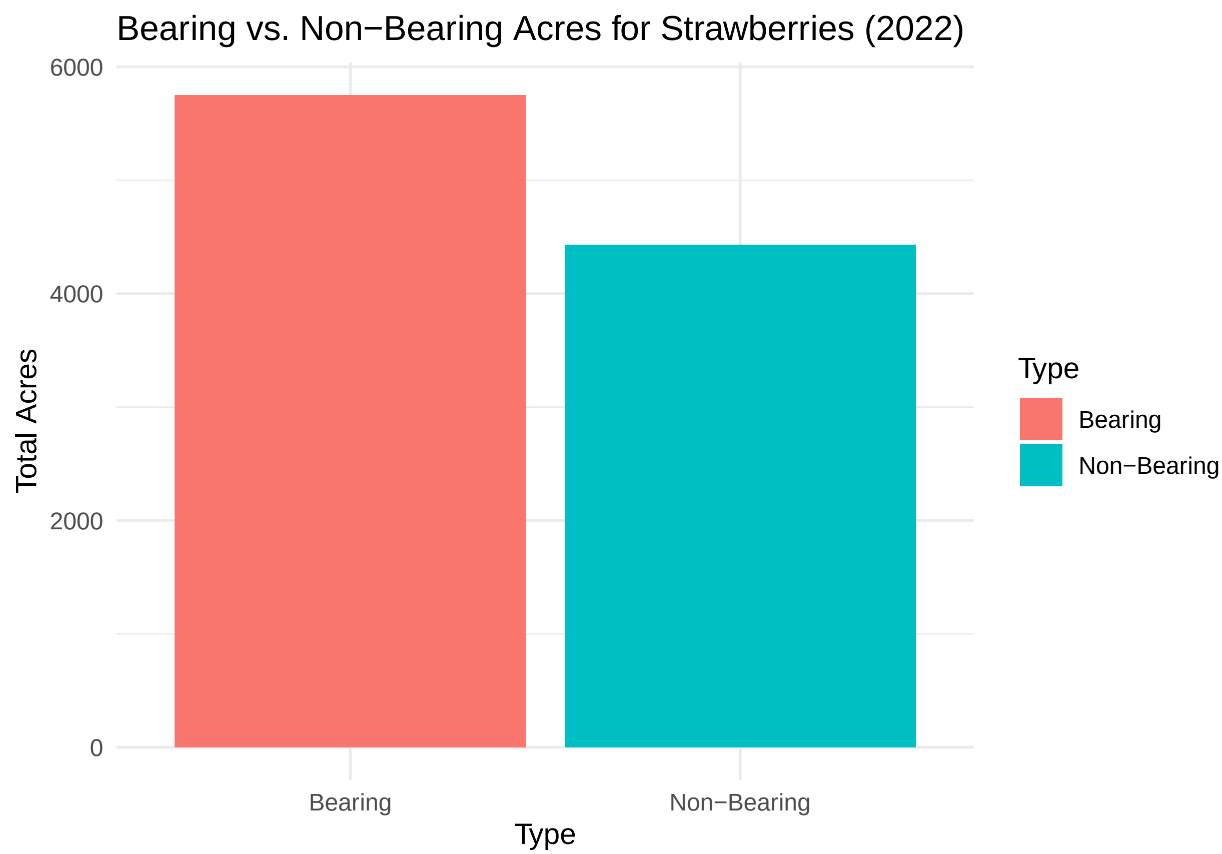
```
## # A tibble: 61 x 15
##    Program  Year Period Week_Ending Geo_Level State  State_ANSI Region Commodity
##    <chr>   <dbl> <chr>  <lgl>       <chr>     <chr>  <chr>      <lgl>  <chr>
##  1 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA     STRAWBER~
```

```
##   2 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA      STRAWBER~
##   3 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA      STRAWBER~
##   4 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA      STRAWBER~
##   5 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA      STRAWBER~
##   6 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA      STRAWBER~
##   7 CENSUS   2022 YEAR   NA          NATIONAL  US TO~ Unknown    NA      STRAWBER~
##   8 CENSUS   2022 YEAR   NA          STATE     CALIF~ 06         NA      STRAWBER~
##   9 CENSUS   2022 YEAR   NA          STATE     CALIF~ 06         NA      STRAWBER~
##  10 CENSUS   2022 YEAR   NA          STATE     CALIF~ 06         NA      STRAWBER~
## # i 51 more rows
## # i 6 more variables: Data_Item <chr>, Domain <chr>, Domain_Category <chr>,
## #   Value <dbl>, Use <chr>, Chemical_Name <chr>
```

```r
# Filter data for acres grown in 2022 and group by state
acres_by_state <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES GROWN"), Year == 2022) %>%
  group_by(State) %>%
  summarise(Total_Acres = sum(Value, na.rm = TRUE))

# Plot total acres grown by state
ggplot(acres_by_state, aes(x = reorder(State, -Total_Acres), y = Total_Acres)) +
  geom_bar(stat = "identity", fill = "blue") +
  labs(title = "Total Acres Grown for Strawberries by State (2022)",
       x = "State",
       y = "Total Acres Grown") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```




```r
# Filter data for bearing and non-bearing acres in 2022
bearing_acres <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES BEARING"), Year == 2022) %>%
  summarise(Total_Bearing_Acres = sum(Value, na.rm = TRUE))

non_bearing_acres <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES NON-BEARING"), Year == 2022) %>%
  summarise(Total_Non_Bearing_Acres = sum(Value, na.rm = TRUE))

# Combine the two into a single data frame
acres_type <- data.frame(
  Type = c("Bearing", "Non-Bearing"),
  Acres = c(bearing_acres$Total_Bearing_Acres, non_bearing_acres$Total_Non_Bearing_Acres)
)

# Plot bearing vs. non-bearing acres
ggplot(acres_type, aes(x = Type, y = Acres, fill = Type)) +
  geom_bar(stat = "identity") +
  labs(title = "Bearing vs. Non-Bearing Acres for Strawberries (2022)",
       x = "Type",
       y = "Total Acres") +
  theme_minimal()
```



From the Acres Data, we see that Oregon is the state with the biggest Acres of land to grow strawberries. Nationally, there is a bigger proportion of bearing acres than that of non-bearing, showing a good sign of eco-friendly farming, saving the soil.

We will now look at how it differs by state.

##Analysis on Strawberries grown.

```
# Load necessary libraries
library(dplyr)
library(ggplot2)
library(readr)

# Convert 'Value' column to numeric, replacing '(D)' or other placeholders with NA
strawberry_clean <- strawberry_clean %>%
  mutate(Value = ifelse(Value %in% c("(D)", "(NA)"), NA, as.numeric(Value)))

# Check the structure of the cleaned dataset
str(strawberry_clean)
```

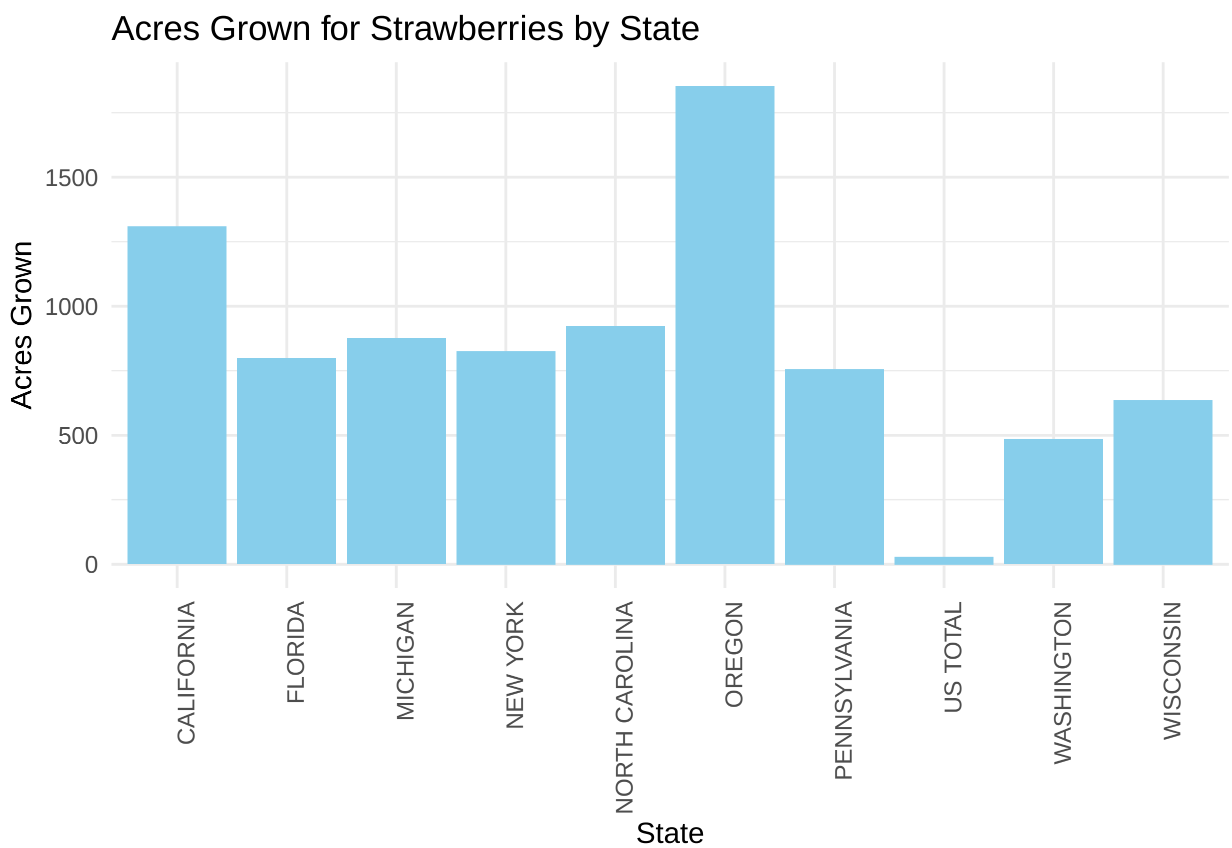
```
## tibble [4,564 x 15] (S3: tbl_df/tbl/data.frame)
##  $ Program         : chr [1:4564] "CENSUS" "CENSUS" "CENSUS" "CENSUS" ...
##  $ Year            : num [1:4564] 2022 2022 2022 2022 2022 ...
##  $ Period          : chr [1:4564] "YEAR" "YEAR" "YEAR" "YEAR" ...
##  $ Week_Ending     : logi [1:4564] NA NA NA NA NA NA ...
##  $ Geo_Level       : chr [1:4564] "NATIONAL" "NATIONAL" "NATIONAL" "NATIONAL" ...
##  $ State           : chr [1:4564] "US TOTAL" "US TOTAL" "US TOTAL" "US TOTAL" ...
```

```
##  $ State_ANSI     : chr [1:4564] "Unknown" "Unknown" "Unknown" "Unknown" ...
##  $ Region         : logi [1:4564] NA NA NA NA NA NA ...
##  $ Commodity      : chr [1:4564] "STRAWBERRIES" "STRAWBERRIES" "STRAWBERRIES" "STRAWBERRIES" ...
##  $ Data_Item      : chr [1:4564] "STRAWBERRIES - ACRES BEARING" "STRAWBERRIES - ACRES BEARING" "STRAW
##  $ Domain         : chr [1:4564] "AREA GROWN" "AREA GROWN" "AREA GROWN" "AREA GROWN" ...
##  $ Domain_Category: chr [1:4564] "AREA GROWN: (0.1 TO 0.9 ACRES)" "AREA GROWN: (1.0 TO 4.9 ACRES)" "A
##  $ Value          : num [1:4564] 963 4 4 4 4 4 4 4 4 4 ...
##  $ Use            : chr [1:4564] "Other" "Other" "Other" "Other" ...
##  $ Chemical_Name  : chr [1:4564] "0.1 TO 0.9 ACRES" "1.0 TO 4.9 ACRES" "100 OR MORE ACRES" "15.0 TO 1
```

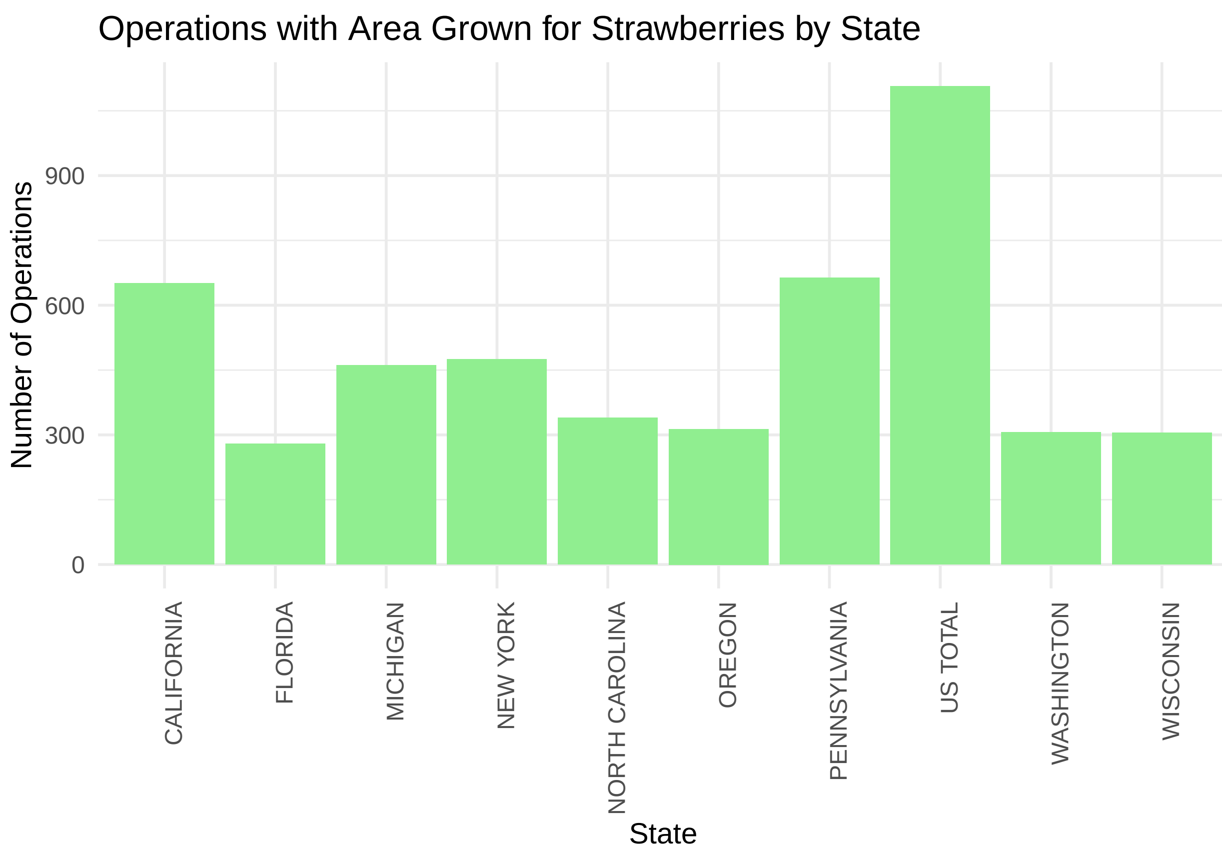
```r
summary(strawberry_clean)
```

```
##    Program               Year         Period           Week_Ending
##  Length:4564        Min.   :2018   Length:4564        Mode:logical
##  Class :character   1st Qu.:2019   Class :character   NA's:4564
##  Mode  :character   Median :2021   Mode  :character
##                     Mean   :2020
##                     3rd Qu.:2022
##                     Max.   :2023
##   Geo_Level            State            State_ANSI         Region
##  Length:4564        Length:4564        Length:4564        Mode:logical
##  Class :character   Class :character   Class :character   NA's:4564
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##   Commodity          Data_Item            Domain          Domain_Category
##  Length:4564        Length:4564        Length:4564        Length:4564
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      Value              Use            Chemical_Name
##  Min.   :  0.017   Length:4564        Length:4564
##  1st Qu.:  4.000   Class :character   Class :character
##  Median :  4.000   Mode  :character   Mode  :character
##  Mean   : 26.879
##  3rd Qu.:  4.000
##  Max.   :963.000
```

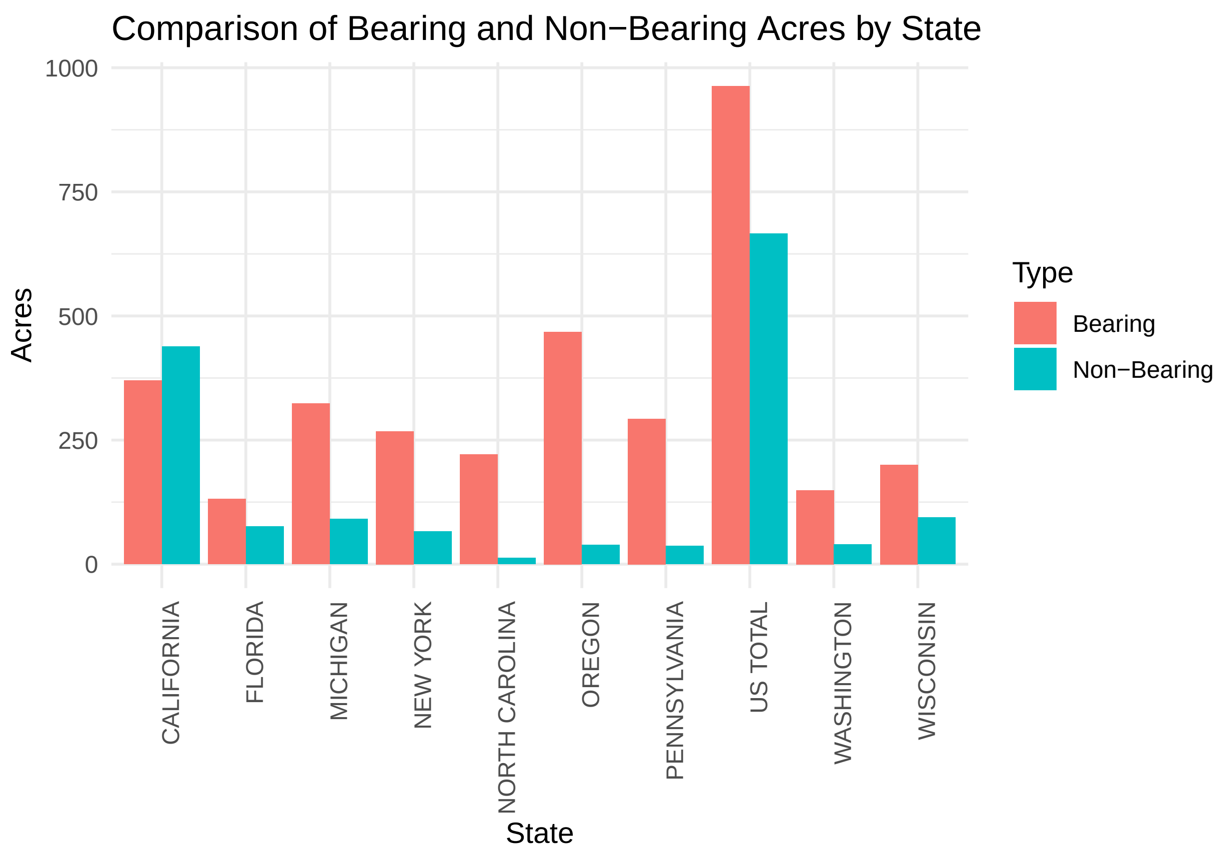
```r
### Visualization 1: Total Acres Grown for Strawberries by State
acres_data <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES GROWN"))

ggplot(acres_data, aes(x = State, y = Value)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(title = "Acres Grown for Strawberries by State",
       x = "State",
       y = "Acres Grown") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```




```r
### Visualization 2: Operations with Area Grown by State
operations_data <- strawberry_clean %>%
  filter(str_detect(Data_Item, "OPERATIONS WITH AREA GROWN"))

ggplot(operations_data, aes(x = State, y = Value)) +
  geom_bar(stat = "identity", fill = "lightgreen") +
  labs(title = "Operations with Area Grown for Strawberries by State",
       x = "State",
       y = "Number of Operations") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



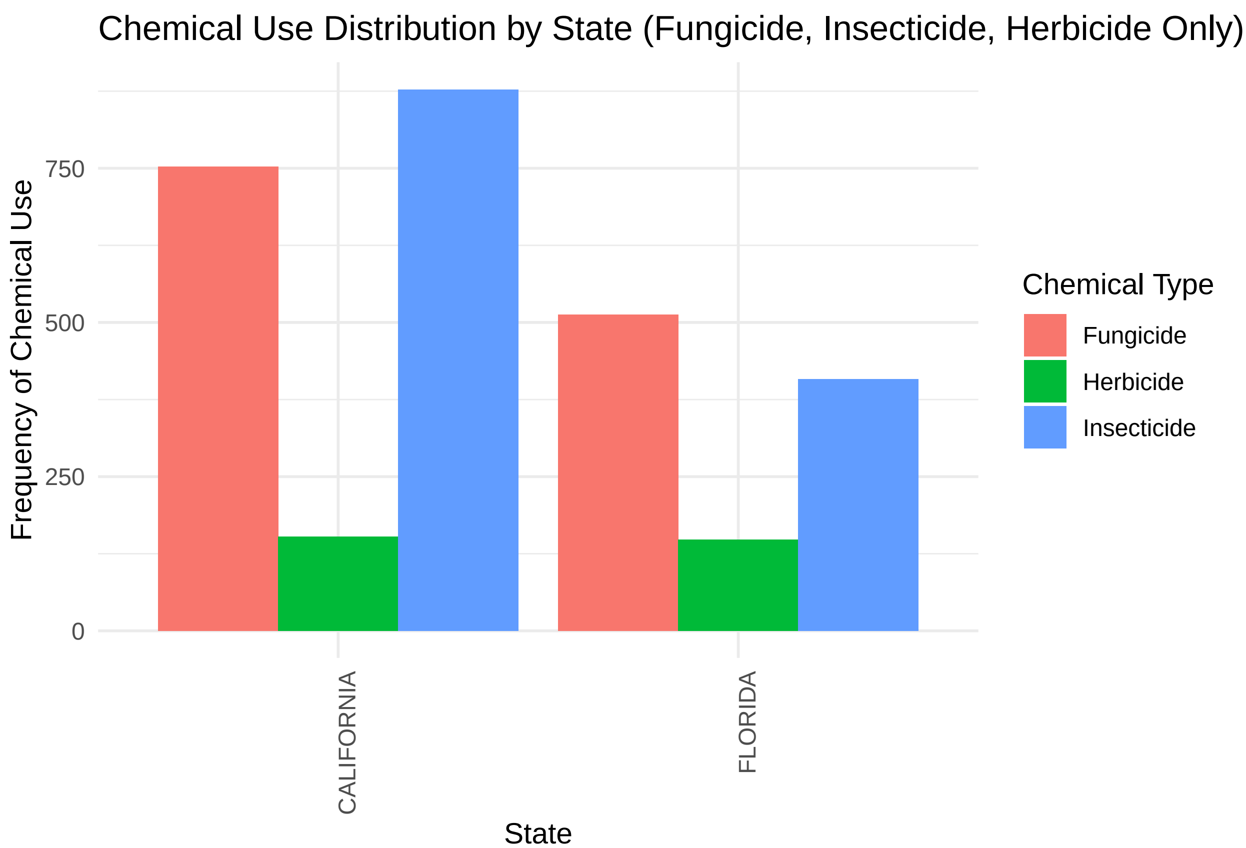

```r
### Visualization 3: Comparison of Bearing vs. Non-Bearing Acres by State
bearing_data <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES BEARING"))

non_bearing_data <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES NON-BEARING"))

combined_acres <- rbind(
  bearing_data %>% mutate(Type = "Bearing"),
  non_bearing_data %>% mutate(Type = "Non-Bearing")
)

ggplot(combined_acres, aes(x = State, y = Value, fill = Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Comparison of Bearing and Non-Bearing Acres by State",
       x = "State",
       y = "Acres",
       fill = "Type") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



California: .Across all graphs, California stands out as the leading state in terms of strawberry acreage and operations. This could lead to an understanding of importance in the US strawberry market. Showing a higher number of non-bearing acres suggesting that the state is investing in future production and crop rotation practices Oregon and North Carolina: Also showing significance in portion of non-bearing acres, indicating similar practices to maintain soil health and prepare for future production cycles. US Total: showing a balanced comparison between bearing and non-bearing acres. It represents the nationwide trend of substantial portion of land is kept in non-bearing status to sustain long-term productivity.

We could look deeper into how strawberries farming could actually be a eco-friendly farming in the future.

```r
strawberry_clean_filtered <- strawberry_clean %>%
  filter(Use %in% c("Fungicide", "Insecticide", "Herbicide"))

# Group by State and Chemical Use
chemicals_by_state <- strawberry_clean_filtered %>%
  group_by(State, Use) %>%
  summarise(Frequency = n(), .groups = 'drop')

# Plot the distribution of chemical types by state excluding "Other"
ggplot(chemicals_by_state, aes(x = reorder(State, -Frequency), y = Frequency, fill = Use)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Chemical Use Distribution by State (Fungicide, Insecticide, Herbicide Only)",
       x = "State",
       y = "Frequency of Chemical Use",
       fill = "Chemical Type") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



As you can see, California uses inseciticdes the most, and fungicides as shown, while herbicide is low From here, I am questioning why this is the case, with California being the state with the most operation going on .

The high use of insecticides in California's strawberry fields is due to the state's specific pest challenges. One of the major pests is the lygus bug (Lygus hesperus), which causes significant damage to strawberry crops. The lygus bug is particularly difficult to control due to its mobility and its tendency to migrate into strawberry fields from nearby vegetation. As a result, farmers often resort to using insecticides like malathion, acetamiprid, and novaluron to manage these pests effectively

https://croplifefoundation.wordpress.com/wp-content/uploads/2012/07/combined_document_strawberries.pdf

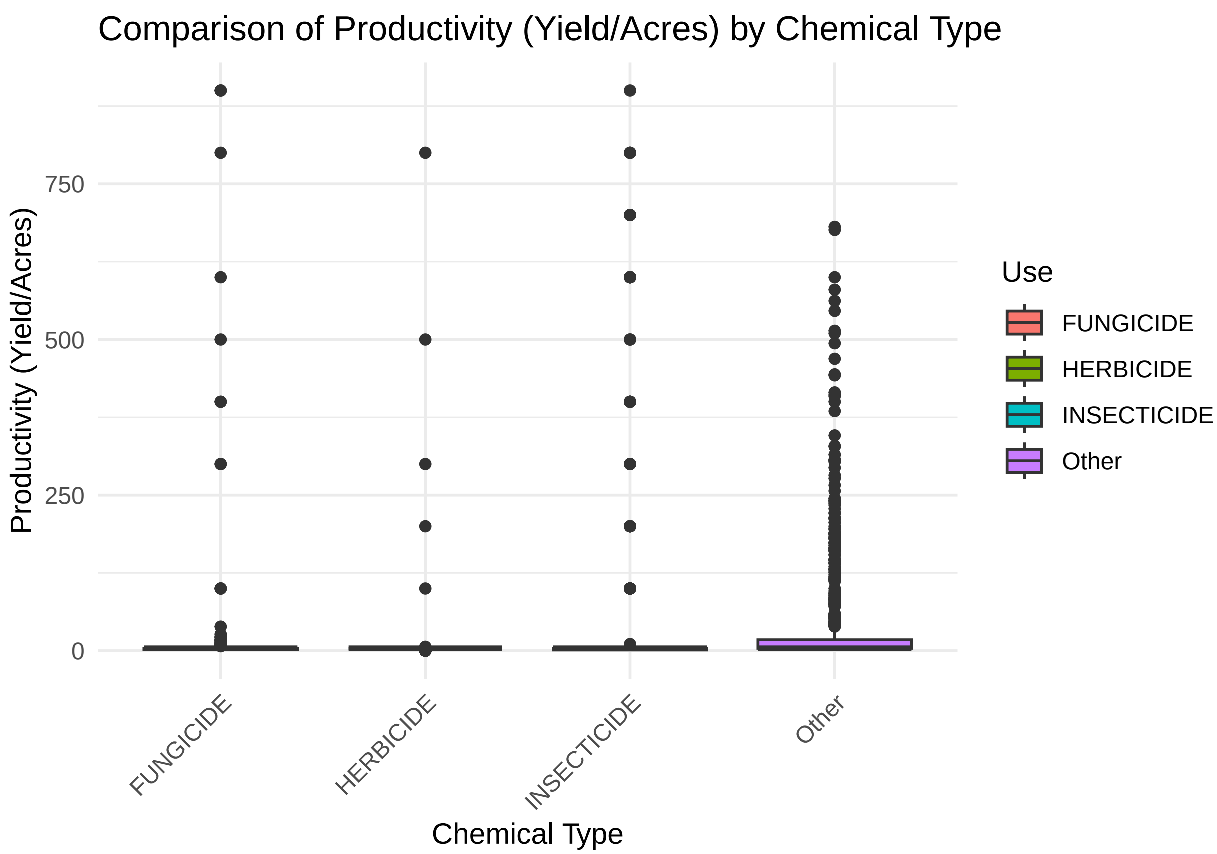
#Producticity Conparison by chemicals

```
# Step 1: Classify Chemical Types and Extract Chemical Names
strawberry_clean <- strawberry_clean %>%
  mutate(
    Use = case_when(
      str_detect(Domain, regex("FUNGICIDE", ignore_case = TRUE)) ~ "FUNGICIDE",
      str_detect(Domain, regex("INSECTICIDE", ignore_case = TRUE)) ~ "INSECTICIDE",
      str_detect(Domain, regex("HERBICIDE", ignore_case = TRUE)) ~ "HERBICIDE",
      TRUE ~ "Other"
    ),
    # Extract specific chemical names from 'Domain_Category' column
    Chemical_Name = str_extract(Domain_Category, "\\((.*?)\\)"),
    Chemical_Name = str_replace_all(Chemical_Name, "[()]", "")  # Remove parentheses
  )
```

```r
# Step 2: Filter Out Rows Where 'Use' or 'Chemical_Name' Are NA
strawberry_clean <- strawberry_clean %>%
  filter(!is.na(Use) & !is.na(Chemical_Name))

# Step 3: Analyze Productivity by Chemical Use Type
# Filter dataset for relevant productivity data (e.g., "ACRES GROWN", "OPERATIONS WITH AREA", "APPLICAT
productivity_data <- strawberry_clean %>%
  filter(str_detect(Data_Item, "ACRES GROWN") |
         str_detect(Data_Item, "OPERATIONS WITH AREA") |
         str_detect(Data_Item, "APPLICATIONS") |
         str_detect(Data_Item, "CHEMICAL"))

# Calculate average productivity (yield/acres) per chemical use category
average_productivity <- productivity_data %>%
  group_by(Use) %>%
  summarise(Average_Productivity = mean(Value, na.rm = TRUE)) %>%
  arrange(desc(Average_Productivity))

# Print summary of average productivity by chemical use category
print(average_productivity)
```

```
## # A tibble: 4 x 2
##   Use         Average_Productivity
##   <chr>                      <dbl>
## 1 Other                      38.7
## 2 INSECTICIDE                18.4
## 3 HERBICIDE                  11.5
## 4 FUNGICIDE                   8.63
```

```r
# Visualization: Boxplot of productivity (yield/acres) by chemical type
ggplot(productivity_data, aes(x = Use, y = Value, fill = Use)) +
  geom_boxplot() +
  labs(title = "Comparison of Productivity (Yield/Acres) by Chemical Type",
       x = "Chemical Type",
       y = "Productivity (Yield/Acres)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Other: Showing the highest average productivity (38.66), which indicates that non-classified chemicals might be associated with higher acre productivity in strawberry farming. Insecticide: Second highest average productivity (18.38), suggesting that insecticides are relatively efficient. Herbicide: Showing a lower average productivity (11.55), which implies that herbicides may not contribute significantly to productivity in terms of yield or acres in this data set. Fungicide: Lowest average productivity (8.63) out of all variables, suggesting that fungicides are less associated with productivity in terms of yield or acres compared to the other chemical categories.

Disregarding the Other element, we can see that the yield increases by how it impacts human. Insecticides are known to be the most harmful for humans, because it is a chemical that kills insects, it could also be very toxic to humans depending on the chemical.

Some of Herbicides are known to be very dangerous for humans, while some aren't, but could be the second most harmful for humans. And fungicides are less toxic to humans, and has minor impacts to humans even if it does have an impact.

Then, why would farmers choose Fungicides with the lowest efficiency, if they could just increase the use of insecticides for the increasing yield?

We will have to take a further look into how education has led to people being attracted more to the chemicals that will be less harmful for humans. Also, scientists will have to develop chemicals that are both efficient, while being safe for humans.

Will there be major changes into the farming world where there will be chemical that impacts humans to the least while increasing the yield of product to its highest?