Stawberries: exploratory data analysis

Yueqi(Charlene) Jin

2023-10-15

Final Objective of Data Research

1. *Use Frequency of Top 15 Chemicals in various State*
2. *Visualize all chemicals and differentiate them by color based on their safety and toxicity*
3. *Toxic & Safe Chemical Frequency by State*
4. *Toxic Chemical Use Frequency in Various States from 2016 to 2021*
5. *Safe Chemical Use Frequency in Various States from 2016 to 2021*
6. *Average Value of Strawberries using toxic chemicals & safe chemicals*

Data acquisition and assessment

```
Rows: 4,314
Columns: 21
$ Program             <chr> "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CENSUS", "~
$ Year                <dbl> 2021, 2021, 2021, 2021, 2021, 2021, 2021, 2021, 202~
$ Period              <chr> "YEAR", "YEAR", "YEAR", "YEAR", "YEAR", "YEAR", "YE~
$ `Week Ending`       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Geo Level`         <chr> "STATE", "STATE", "STATE", "STATE", "STATE", "STATE~
$ State               <chr> "ALASKA", "ALASKA", "ALASKA", "ALASKA", "ALASKA", "~
$ `State ANSI`        <chr> "02", "02", "02", "02", "02", "02", "02", "06", "06~
$ `Ag District`       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Ag District Code`  <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ County              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `County ANSI`       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ `Zip Code`          <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ Region              <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
$ watershed_code      <chr> "00000000", "00000000", "00000000", "00000000", "00~
$ Watershed           <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
```

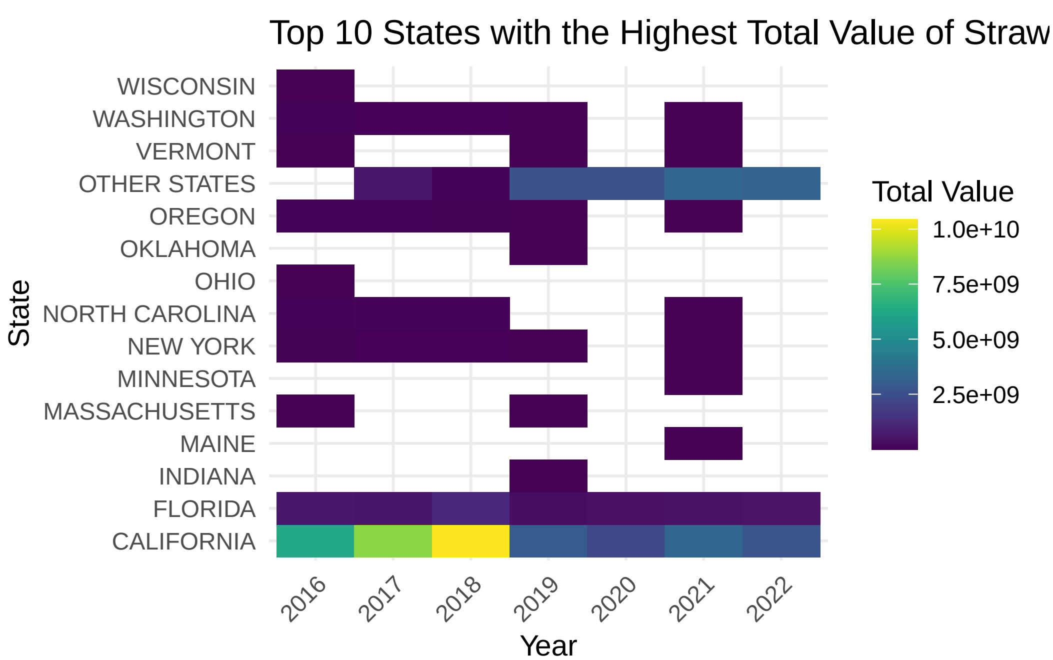
```
$ Commodity           <chr> "STRAWBERRIES", "STRAWBERRIES", "STRAWBERRIES", "ST~
$ `Data Item`         <chr> "STRAWBERRIES, ORGANIC - OPERATIONS WITH SALES", "S~
$ Domain              <chr> "ORGANIC STATUS", "ORGANIC STATUS", "ORGANIC STATUS~
$ `Domain Category`   <chr> "ORGANIC STATUS: (NOP USDA CERTIFIED)", "ORGANIC ST~
$ Value               <chr> "2", "(D)", "(D)", "(D)", "2", "(D)", "(D)", "142",~
$ `CV (%)`            <chr> "(H)", "(D)", "(D)", "(D)", "(H)", "(D)", "(D)", "1~
```

We see that the top 10 states with the highest total value of strawberries each year.

```r
# Convert the Value column from non-numeric format to numeric
strawberry$Value <- as.numeric(gsub("[^0-9.]", "", strawberry$Value))

# Group data by Year and State, then compute the total value for each state per year
annual_state_value <- strawberry %>%
  group_by(Year, State) %>%
  summarise(Total_Value = sum(Value, na.rm = TRUE), .groups = "drop") %>%
  ungroup()

# Sort data by Year and total value, then retrieve and format the top 10 states
# by value for each year

top_states_each_year <- annual_state_value %>%
  group_by(Year) %>%
  arrange(desc(Total_Value)) %>%
  slice_head(n = 10) %>%
  summarise(Top_10_States = paste(State, collapse=", "), .groups = "drop") %>%
  ungroup()

# Display the results
print(top_states_each_year)
```

```
# A tibble: 7 x 2
   Year Top_10_States
  <dbl> <chr>
1  2016 CALIFORNIA, FLORIDA, NORTH CAROLINA, OREGON, WASHINGTON, NEW YORK, VERM~
2  2017 CALIFORNIA, OTHER STATES, FLORIDA, NORTH CAROLINA, OREGON, WASHINGTON, ~
3  2018 CALIFORNIA, FLORIDA, NORTH CAROLINA, OTHER STATES, OREGON, WASHINGTON, ~
4  2019 CALIFORNIA, OTHER STATES, FLORIDA, OREGON, VERMONT, WASHINGTON, NEW YOR~
5  2020 OTHER STATES, CALIFORNIA, FLORIDA
6  2021 OTHER STATES, CALIFORNIA, FLORIDA, OREGON, WASHINGTON, VERMONT, NEW YOR~
7  2022 OTHER STATES, CALIFORNIA, FLORIDA
```

```r
# Filter to keep only the top 10 states for each year
top_states_each_year_data <- annual_state_value %>%
  group_by(Year) %>%
  top_n(10, Total_Value) %>%
  ungroup()

# Plot the heatmap to visualize top 10 States with the Highest Total Value
# of Strawberries Each Year
ggplot(top_states_each_year_data, aes(x = as.factor(Year), y = State, fill = Total_Value))
  geom_tile() +
  scale_fill_viridis(name = "Total Value") +
  labs(title = "Top 10 States with the Highest Total Value of Strawberries Each Year",
       x = "Year",
       y = "State") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Data cleaning and organization

```
Rows: 4,314
Columns: 10
```

```
$ Program            <chr> "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CENSUS", "C~
$ Year               <dbl> 2021, 2021, 2021, 2021, 2021, 2021, 2021, 2021, 2021~
$ Period             <chr> "YEAR", "YEAR", "YEAR", "YEAR", "YEAR", "YEAR", "YEA~
$ State              <chr> "ALASKA", "ALASKA", "ALASKA", "ALASKA", "ALASKA", "A~
$ `State ANSI`       <chr> "02", "02", "02", "02", "02", "02", "02", "06", "06"~
$ `Data Item`        <chr> "STRAWBERRIES, ORGANIC - OPERATIONS WITH SALES", "ST~
$ Domain             <chr> "ORGANIC STATUS", "ORGANIC STATUS", "ORGANIC STATUS"~
$ `Domain Category`  <chr> "ORGANIC STATUS: (NOP USDA CERTIFIED)", "ORGANIC STA~
$ Value              <dbl> 2, NA, NA, NA, 2, NA, NA, 142, 1413251, 311784980, 1~
$ `CV (%)`           <chr> "(H)", "(D)", "(D)", "(D)", "(H)", "(D)", "(D)", "19~
```

```
Warning in rm(calif_census, calif_survey, state_all): object 'state_all' not
found
```

footnotes

```
## finds single uppor case Character in parens in s2
## e.g. "(D)"
## To fine the location and value of the footnotes

v <- strwb_census$Value

## find the footnote locations
## fn_i: locations
fn_i <- v |> str_detect("^\\([:upper:]\\)$") ## returns

## dcomma returns numbers and NA's
v1 <- dcomma(v)

## locations of NA's
na_i <- is.na(v1)

dcomma <- function(c){
  suppressWarnings({
  xnew = as.numeric(gsub(",", "", c))
  fns = unique(c[is.na(xnew)])
  vtran = list("new_vec" = xnew, "footnotes" = fns)
  return(vtran)
  })
}

v_trns <- dcomma(v)
```

```r
a <- v_trns$new_vec
 # a[1:20]
 # v_trns$footnotes
```

EDA

Chemicals: Firstly we divide the chemical composition into two columns and eliminating irrelevant variables.

```r
stb_survey <- strwb_survey %>%
  filter(str_detect(`Data Item`, "MEASURED IN")) %>%
  mutate(`Data Item` = str_extract(`Data Item`, "(?<=MEASURED IN ).*"))
stb_survey <- stb_survey %>%
  mutate(
    Chemical = if_else(str_detect(`Domain Category`, "\\(.*=.*\\)"),
                       str_extract(`Domain Category`, "(?<=\\().*?(?=\\=)"),
                       NA_character_),
    Chemical_Code = if_else(str_detect(`Domain Category`, "\\(.*=.*\\)"),
                            str_extract(`Domain Category`, "(?<=\\=).*?(?=\\))"),
                            NA_character_)
  )


stb_survey <- subset(stb_survey, select = -Program)
stb_survey <- subset(stb_survey, select = -`Domain Category`)

stb_survey$Chemical_Code_num <- as.numeric(stb_survey$Chemical_Code)
stb_survey$Chemical_Code_str <- ifelse(is.na(stb_survey$Chemical_Code_num),NA,
                                       sprintf("%06d",
                                               stb_survey$Chemical_Code_num))
```

Handing Missing Values, Outliers, and Duplicates

```r
stb_survey <- stb_survey[, !sapply(stb_survey, function(col) all(is.na(col)))]
stb_survey <- stb_survey[!is.na(stb_survey$Value), ]
stb_survey <- stb_survey[stb_survey$State != "OTHER STATES", ]
```

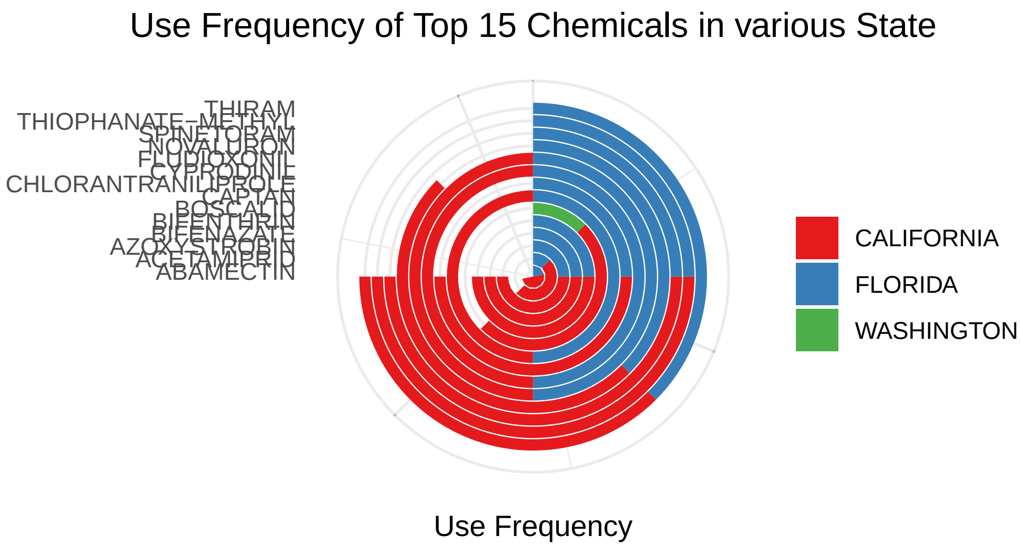
```r
strwb_census$`CV (%)`<- as.numeric(strwb_census$`CV (%)`)
```

```
Warning: NAs introduced by coercion
```

```r
strwb_census <- strwb_census %>%
  select(-Program,-`Period`,-Fruit,-crop_type,-Domain,-`Domain Category`)
```

Create a detailed visualization that illustrates the varying frequencies at which each individual chemical substance is utilized across the various states.

```r
stb_survey$Domain <- gsub("CHEMICAL,", "", stb_survey$Domain)
stb_survey$Domain <- trimws(stb_survey$Domain)
write.csv(stb_survey, file = "stb_survey.csv", row.names = FALSE)

top_chemicals_data <- stb_survey %>%
  count(Chemical) %>%
  arrange(-n) %>%
  head(15)

# Extract data for these top 15 chemicals from the original dataset.
subset_stb_survey <- stb_survey %>%
  filter(!is.na(Chemical)) %>%
  filter(Chemical %in% top_chemicals_data$Chemical)

p <- ggplot(subset_stb_survey, aes(x = Chemical, fill = State)) +
  geom_bar(position = "stack", stat = "count") +
  scale_fill_brewer(palette = "Set1") +
  labs(title = "Use Frequency of Top 15 Chemicals in various State",
       x = NULL, y = "Use Frequency") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = 1),
    axis.ticks = element_blank(),
    legend.title = element_blank(),
    legend.position = "right",
    plot.title = element_text(hjust = 0.5)
  ) +
  coord_polar(theta = "y")
print(p)
```



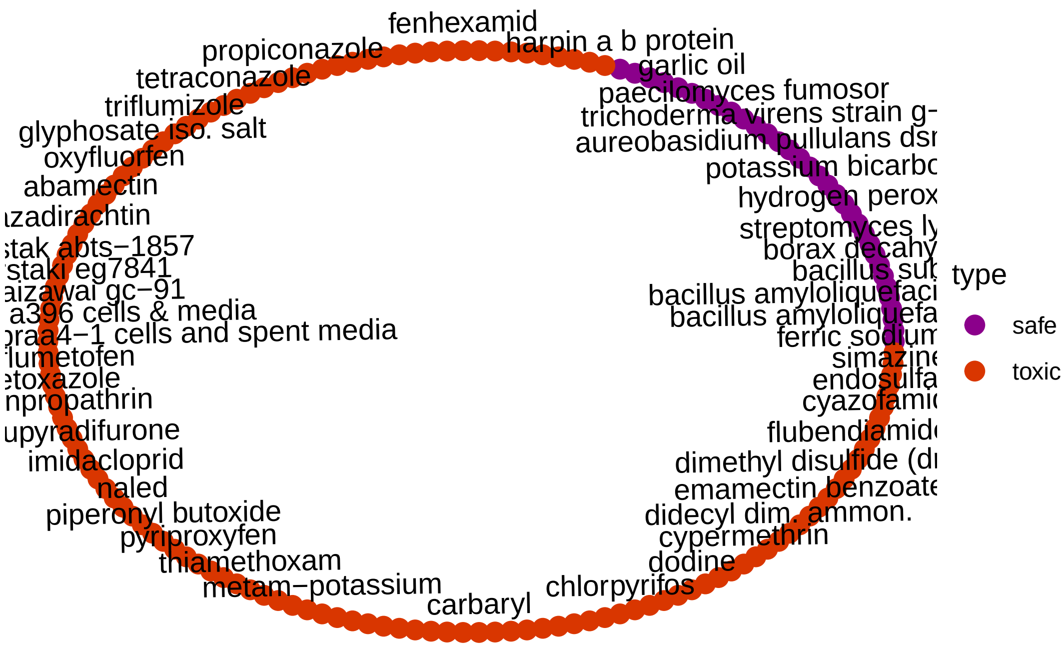
The chart visualizes the use frequency of the top 15 chemicals across three states: California, Florida, and Washington. California, represented by the red color, seems to have the highest use frequency for most of the chemicals listed, as its bands are more pronounced and extend further outward on the chart. Florida, depicted in blue, also has significant use for many of these chemicals, but in general, its usage seems to be less than California for most of the chemicals. Washington, represented by the green color, has the least use frequency among the three states for almost all chemicals, as its bands are the narrowest. Specific chemicals like THIRAM, THIOPHANATE-METHYL, and SPINETORAM show notably high usage in California compared to the other two states. For some chemicals, such as AZOXYSTROBIN and ACETAMIPRID, the usage frequency appears to be relatively close between California and Florida, with both states showing notable use. ABAMECTIN has a discernible usage in Florida, even though it's much less in comparison to California.

Then Create two subsets, poisons_chem and safe_chem, by filtering based on the values in the 'Chemical' column from the stb_survey data table.Visualize all chemicals and differentiate them by color based on their safety.

```
#Data Cleaning: Remove blanks and standardize to lowercase, extract unique
#chemical names and create a new dataframe to store these distinct chemical names

stb_survey$Chemical <- tolower(trimws(stb_survey$Chemical))
unique_chemicals <- unique(stb_survey$Chemical)
```

```r
unique_chemicals_df <- data.frame(Chemical = unique_chemicals)
write.csv(unique_chemicals_df, file = "unique_chemicals.csv", row.names = FALSE)

poisonous_chemicals <- c("azoxystrobin", "blad", "boscalid",
                         "bt subsp kurstaki evb-113-19", "captan",
                         "copper octanoate", "cyflufenamid", "cyprodinil",
                         "difenoconazole", "fenhexamid", "fludioxonil",
                         "fluopyram", "fluxapyroxad", "fosetyl-al", "isofetamid",
                         "mefenoxam", "mono-potassium salt", "myclobutanil",
                         "penthiopyrad", "polyoxin d zinc salt", "propiconazole",
                         "pyraclostrobin", "pyrimethanil", "quinoline", "sulfur",
                         "tetraconazole", "thiophanate-methyl", "thiram",
                         "trifloxystrobin",
                         "triflumizole", "carfentrazone-ethyl", "flumioxazin",
                         "glyphosate iso. salt", "glyphosate pot. salt",
                         "napropamide", "oxyfluorfen", "paraquat", "pendimethalin",
                         "abamectin", "acequinocyl", "acetamiprid", "azadirachtin",
                         "bifenazate", "bifenthrin", "bt kurstak abts-1857",
                         "bt kurstaki abts-351", "bt kurstaki eg7841",
                         "bt kurstaki sa-11", "bt sub aizawai gc-91",
                         "buprofezin", "burkholderia a396 cells & media",
                         "chlorantraniliprole",
                         "chromobac subtsugae praa4-1 cells and spent media",
                         "cyantraniliprole", "cyflumetofen", "diazinon", "etoxazole",
                         "fenbutatin-oxide", "fenpropathrin", "fenpyroximate",
                         "flonicamid", "flupyradifurone", "helicoverpa zea npv",
                         "hexythiazox", "imidacloprid", "malathion",
                         "methoxyfenozide", "naled", "novaluron",
                         "petroleum distillate", "piperonyl butoxide",
                         "pyrethrins", "pyridaben", "pyriproxyfen",
                         "spinetoram", "spinosad", "spiromesifen",
                         "thiamethoxam", "acibenzolar-s-methyl",
                         "chloropicrin", "dichloropropene", "flutriafol",
                         "metam-potassium", "peroxyacetic acid", "potassium silicate",
                         "pydiflumetofen", "clethodim", "copper ethanolamine",
                         "dimethenamid", "fluroxypyr 1-mhe", "halosulfuron-methyl",
                         "kantor", "carbaryl", "fenazaquin", "sulfoxaflor",
                         "cytokinins", "ethephon", "indolebutyric acid",
                         "copper hydroxide", "glufosinate-ammonium",
                         "sulfentrazone", "chlorpyrifos", "zeta-cypermethrin",
                         "metaldehyde", "metam-sodium", "copper chloride hyd.",
```

```
                              "dodine", "flutolanil", "2,4-d, dimeth. salt", "2,4-d,
                              triiso. salt",
                              "cypermethrin", "alkyl. dim. benz. am", "decyldimethyloctyl",
                              "didecyl dim. ammon.", "dimethyldioctyl", "iprodione",
                              "cyflumetofen", "emamectin benzoate", "lambda-cyhalothrin",
                              "spirotetramat", "dimethyl disulfide (dmds)",
                              "copper oxide", "ammonium pelargonate", "flubendiamide",
                              "methyl bromide", "chlorothalonil", "cyazofamid",
                              "mancozeb", "endosulfan", "clopyralid mono salt",
                              "simazine", "terbacil", "ferric sodium edta", "clomazone")

safe_chemicals <- c("bacillus amyloliquefaciens mbi 600", "bacillus amyloliquefac f727",
                    "bacillus amyloliquefaciens strain d747", "bacillus pumilus",
                    "bacillus subtilis", "beauveria bassiana", "borax decahydrate",
                    "canola oil", "streptomyces lydicus", "neem oil",
                    "neem oil, clar. hyd.", "hydrogen peroxide", "iron phosphate",
                    "mineral oil", "potassium bicarbon.", "potassium salts",
                    "soybean oil", "aureobasidium pullulans dsm 14940",
                    "aureobasidium pullulans dsm 14941", "trichoderma harz.",
                    "trichoderma virens strain g-41", "gliocladium virens",
                    "bacillus subt. gb03", "paecilomyces fumosor",
                    "reynoutria sachaline", "pseudomonas chlororaphis strain afs009",
                    "capsicum oleoresin extract", "garlic oil", "capric acid",
                    "caprylic acid", "mustard oil", "capsaicin", "harpin a b protein")

all_chemicals <- c(safe_chemicals, poisonous_chemicals)
unique_chemicals <- unique(all_chemicals)

df <- data.frame(
  name = unique_chemicals,
  type = ifelse(unique_chemicals %in% safe_chemicals, "safe", "toxic")
)
graph <- graph.empty(n = 0, directed = FALSE)
graph <- add_vertices(graph, nrow(df), name = df$name, type = df$type)

ggraph(graph, layout = 'circle') +
  geom_node_point(aes(color = type), size = 3) +
  geom_node_text(aes(label = name), nudge_y = 0.1, check_overlap = TRUE, angle = TRUE) +
  scale_color_manual(values = c("safe" = "darkmagenta", "toxic" = "#D93600")) +
  theme_void() +
  theme(legend.position = "right")
```

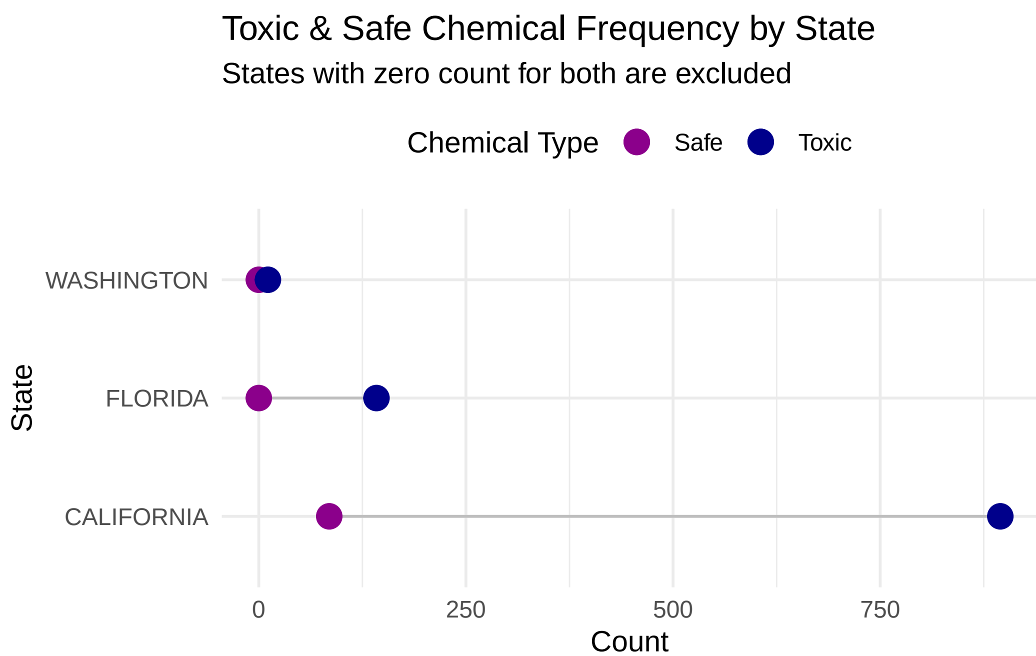



```
stb_survey$Chemical <- trimws(stb_survey$Chemical)

poisons_chem <- subset(stb_survey, Chemical %in% poisonous_chemicals)
safe_chem <- subset(stb_survey, Chemical %in% safe_chemicals)
```

Deal with preprocessing data, calculate the count of toxic and non-toxic chemicals for each state.And Exclude states where both toxic and non-toxic chemical counts are zero.

```
stb_grouped <- stb_survey %>%
  group_by(State) %>%
  summarise(
    toxic_count = sum(Chemical %in% poisonous_chemicals),
    safe_count = sum(Chemical %in% safe_chemicals)
  ) %>%
  filter(toxic_count > 0 | safe_count > 0)
options(repr.plot.width=10, repr.plot.height=8)

ggplot(stb_grouped, aes(y = State)) +
  geom_segment(aes(x = safe_count, xend = toxic_count, yend = State), color = 'grey') +
  geom_point(aes(x = safe_count, color = "Safe"), size = 4) +
  geom_point(aes(x = toxic_count, color = "Toxic"), size = 4) +
```

```
  scale_color_manual(values = c("Toxic" = "darkblue", "Safe" = "darkmagenta")) +
  labs(x="Count", y="State", title="Toxic & Safe Chemical Frequency by State",
       subtitle = "States with zero count for both are excluded",
       color="Chemical Type") +
  theme_minimal() +
  theme(legend.position = "top")
```

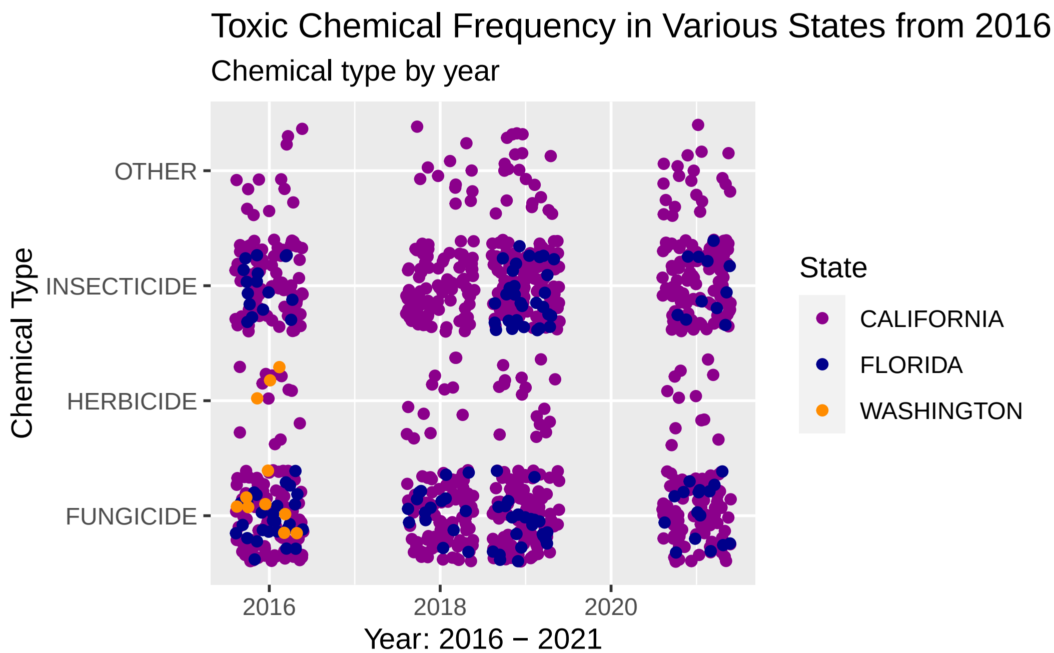


Among the three listed states, California has the highest frequency of both safe and toxic chemical usage.For safe chemicals, there is no safe chemical in Florida and Washington.When considering toxic chemicals, Florida's frequency is noticeably higher than that of Washington.

Toxic Chemical Frequency in Various States from 2016 to 2021

```
# Filter the data to include only the specified domains
selected_domains <- c("INSECTICIDE", "FUNGICIDE", "HERBICIDE", "OTHER")
poisons_chem_selected <- poisons_chem %>%
  filter(Domain %in% selected_domains)

# Create the plot with a custom color scale
toxic_plot1 <- ggplot(data = poisons_chem_selected,
           aes(x = Year, y = Domain, col = State)) +
```

```
  geom_jitter() +
  xlab('Year: 2016 - 2021') +
  ylab('Chemical Type') +
  labs(title = 'Toxic Chemical Frequency in Various States from 2016 to 2021',
       subtitle = 'Chemical type by year') +
  scale_color_manual(values = c("darkmagenta", "darkblue","darkorange"))
toxic_plot1
```

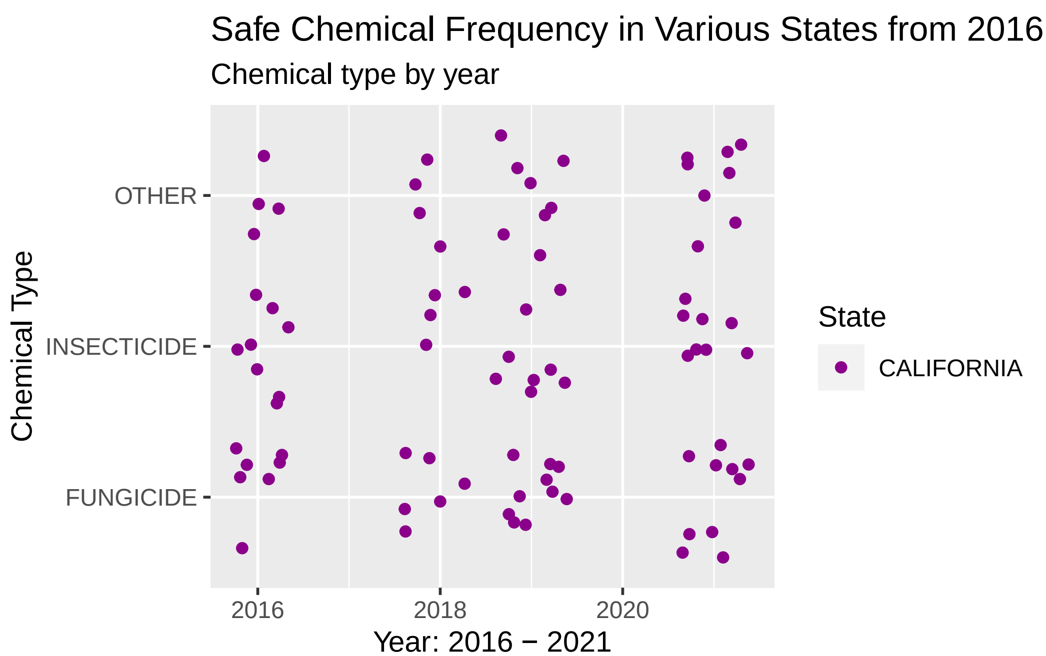


From the plot regarding toxic chemicals, we can find that insecticide and fungicide widely used in strawberry farming in California and Florida. Strawberry farming in Washington only used insecticides, herbicides, and fungicides in 2016, but did not use them in subsequent years.

Safe Chemical Frequency in Various States from 2016 to 2021

```
# Filter the data to include only the specified domains
selected_domains <- c("INSECTICIDE", "FUNGICIDE", "HERBICIDE", "OTHER")
safe_chem_selected <- safe_chem %>%
  filter(Domain %in% selected_domains)

# Create the plot with a custom color scale
toxic_plot1 <- ggplot(data = safe_chem_selected,
```

```
            aes(x = Year, y = Domain, col = State)) +
  geom_jitter() +
  xlab('Year: 2016 - 2021') +
  ylab('Chemical Type') +
  labs(title = 'Safe Chemical Frequency in Various States from 2016 to 2021',
       subtitle = 'Chemical type by year') +
  scale_color_manual(values = c("darkmagenta", "darkblue", "darkgreen", "darkorange"))
toxic_plot1
```

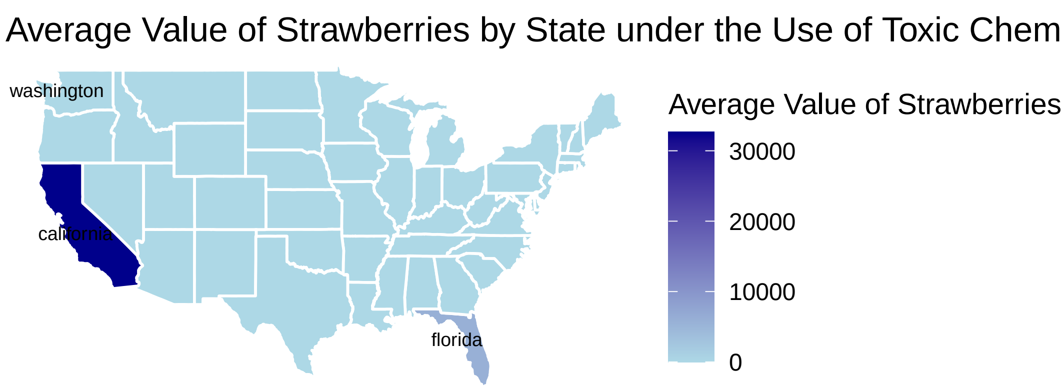


From the plot regarding safe chemicals, we can find that insecticide and fungicide widely only used in strawberry farming in California in all years.

We observe the distribution of the average value of strawberries in each state when using toxic chemicals.

```
poisons_chem$Value <- as.numeric(gsub("[^0-9.]", "", poisons_chem$Value))
us_states <- map_data("state")

# Ensure state names match. Convert both to lowercase to avoid discrepancies due to case.
poisons_chem$State <- tolower(poisons_chem$State)
us_states$region <- tolower(us_states$region)
```

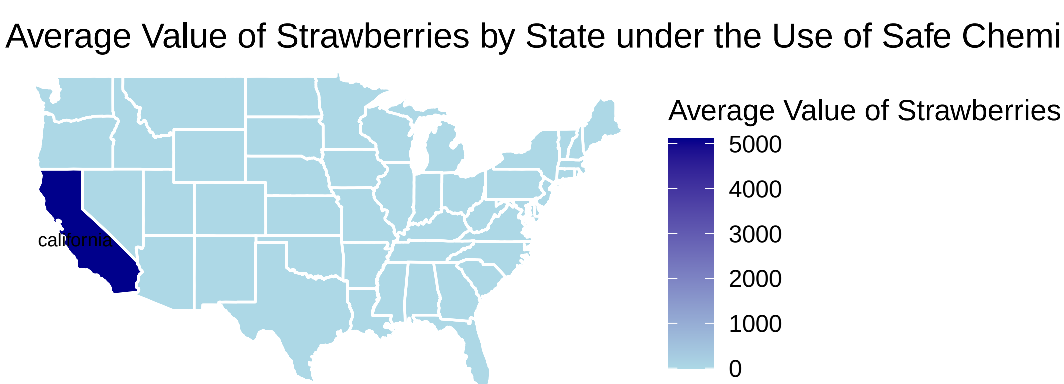
```
# Calculate the average value for each state
state_avg_values <- poisons_chem %>%
  group_by(State) %>%
  summarise(Avg_Value = mean(Value, na.rm = TRUE), .groups='drop')

# Merge the map data and the average values
map_data <- left_join(us_states, state_avg_values, by = c("region" = "State"))

# If there are NA values in the merged data, you can replace them with zeros.
map_data$Avg_Value[is.na(map_data$Avg_Value)] <- 0

#Find the centroid of each state for placing the state label
state_centroids <- map_data %>%
  group_by(region) %>%
  summarise(cent_x = mean(long, na.rm = TRUE), cent_y = mean(lat, na.rm = TRUE),
            Avg_Value = first(Avg_Value))

#Filter centroids to keep only those states that have a non-zero Avg_Value
state_centroids <- filter(state_centroids, Avg_Value > 0)

#Create a choropleth map with shades of blue and label states with non-zero average values
ggplot(data = map_data, aes(x = long, y = lat, fill = Avg_Value)) +
  geom_polygon(aes(group = group), color = "white") +
  geom_text(data = state_centroids, aes(x = cent_x, y = cent_y, label = region),
            size = 2.5, check_overlap = TRUE) +
  scale_fill_gradient(name = "Average Value of Strawberries", low = "lightblue",
                      high = "darkblue") +
  labs(title = "Average Value of Strawberries by State under the Use of Toxic Chemicals")
  coord_fixed(1.3) +
  theme_void()
```



We observe the distribution of the average value of strawberries in each state when using safe chemicals.

```
safe_chem$Value <- as.numeric(gsub("[^0-9.]", "", safe_chem$Value))
us_states <- map_data("state")

# Ensure state names match. Convert both to lowercase to avoid discrepancies due to case.
safe_chem$State <- tolower(safe_chem$State)
us_states$region <- tolower(us_states$region)

# Calculate the average value for each state
state_avg_values <- safe_chem %>%
  group_by(State) %>%
  summarise(Avg_Value = mean(Value, na.rm = TRUE), .groups='drop')

# Merge the map data and the average values
map_data <- left_join(us_states, state_avg_values, by = c("region" = "State"))

# If there are NA values in the merged data, you can replace them with zeros.
map_data$Avg_Value[is.na(map_data$Avg_Value)] <- 0

# Find the centroid of each state for placing the state label
```

```r
state_centroids <- map_data %>%
  group_by(region) %>%
  summarise(cent_x = mean(long, na.rm = TRUE), cent_y = mean(lat, na.rm = TRUE),
            Avg_Value = first(Avg_Value))

# Filter centroids to keep only those states that have a non-zero Avg_Value
state_centroids <- filter(state_centroids, Avg_Value > 0)

# Create a choropleth map with shades of blue and label states
# with non-zero average values
ggplot(data = map_data, aes(x = long, y = lat, fill = Avg_Value)) +
  geom_polygon(aes(group = group), color = "white") +
  geom_text(data = state_centroids, aes(x = cent_x, y = cent_y, label = region),
            size = 2.5, check_overlap = TRUE) +
  scale_fill_gradient(name = "Average Value of Strawberries", low = "lightblue",
                      high = "darkblue") +
  labs(title="Average Value of Strawberries by State under the Use of Safe Chemicals") +
  coord_fixed(1.3) +
  theme_void()
```



Based on the distribution from the two maps, we conclude that under both the use of toxic and non-toxic chemicals, the average value of strawberries in California is the highest. Furthermore, *when using toxic chemicals, the average value of strawberries in Washington is the lowest.*

References

https://www.osha.gov/chemicaldata

https://ordspub.epa.gov/ords/pesticides/f?p=113:17::::::

https://comptox.epa.gov/dashboard/

https://quickstats.nass.usda.gov/src/glossary.pdf

https://quickstats.nass.usda.gov/param_define