615 Strawberry Final Project

Xiaohan Shi

2024-10-29

Step 1: Install packages & Read data

```
library(tidyverse)
```

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

```
library(knitr)
library(kableExtra)
```

```
##
## 载入程序包：'kableExtra'
##
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
library(stringr)
strawberry<-read.csv("strawberries25_v3.csv")
glimpse(strawberry)
```

```
## Rows: 12,669
## Columns: 21
## $ Program           <chr> "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CE…
## $ Year              <int> 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022,…
## $ Period            <chr> "YEAR", "YEAR", "YEAR", "YEAR", "YEAR", "YEAR", "YEAR…
## $ Week.Ending       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
## $ Geo.Level         <chr> "COUNTY", "COUNTY", "COUNTY", "COUNTY", "COUNTY", "CO…
## $ State             <chr> "ALABAMA", "ALABAMA", "ALABAMA", "ALABAMA", "ALABAMA"…
## $ State.ANSI        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,…
## $ Ag.District       <chr> "BLACK BELT", "BLACK BELT", "BLACK BELT", "BLACK BELT…
## $ Ag.District.Code <int> 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 4…
## $ County            <chr> "BULLOCK", "BULLOCK", "BULLOCK", "BULLOCK", "BULLOCK"…
## $ County.ANSI       <int> 11, 11, 11, 11, 11, 11, 101, 101, 101, 101, 119, 119,…
## $ Zip.Code          <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
## $ Region            <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
## $ watershed_code    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
## $ Watershed         <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
## $ Commodity         <chr> "STRAWBERRIES", "STRAWBERRIES", "STRAWBERRIES", "STRA…
## $ Data.Item         <chr> "STRAWBERRIES - ACRES BEARING", "STRAWBERRIES - ACRES…
## $ Domain            <chr> "TOTAL", "TOTAL", "TOTAL", "TOTAL", "TOTAL", "TOTAL",…
## $ Domain.Category   <chr> "NOT SPECIFIED", "NOT SPECIFIED", "NOT SPECIFIED", "N…
## $ Value             <chr> " (D)", "3", " (D)", "1", "6", "5", " (D)", " (D)", "…
## $ CV....            <chr> "(D)", "15.7", "(D)", "(L)", "52.7", "47.6", "(D)", "…
```

It shows a general view of data.

```
sum(strawberry$Domain == "TOTAL")
```

```
## [1] 8105
```

```
sum(strawberry$Domain == "TOTAL")
```

```
## [1] 8105
```

```
state_all <- strawberry |> distinct(State)
state_all1 <- strawberry |> group_by(State) |> count()
```

Step 2: Drop the columns that have a single value.

The reason is that the value of this column is the same in every row, so it won't contribute anything to data analysis, modeling, or forecasting. It does not help to distinguish between different observations.

```
drop_one_value_col <- function(df){
drop <- NULL
for(i in 1:dim(df)[2]){
if((df |> distinct(df[,i]) |> count()) == 1){
drop = c(drop, i)
} }

if(is.null(drop)){return("none")}else{

   print("Columns dropped:")
   print(colnames(df)[drop])
   strawberry <- df[, -1*drop]
   }
}
strawberry <- drop_one_value_col(strawberry)
```

```
## [1] "Columns dropped:"
## [1] "Week.Ending"    "Zip.Code"       "Region"         "watershed_code"
## [5] "Watershed"      "Commodity"
```

```
drop_one_value_col(strawberry)
```

```
## [1] "none"
```

By dropping the Columns with a single value, it makes the data easier to understand and process.

Step 3: understand the data by analysis the data sources.

```
calif <- strawberry |> filter(State=="CALIFORNIA")
unique(calif$Program)
```

```
## [1] "CENSUS" "SURVEY"
```

```
#It can be seen that there are two different kinds of data sources.
```

```
#Analysis the difference between two data sources.
calif_census <- calif |> filter(Program=="CENSUS")
calif_survey  <- calif |>  filter(Program=="SURVEY")
```

The comparison shows that the values of these variables in the survey data are NA: “Ag.District”, “Ag.District.Code”, “Country”, “Country.ANSI”, “CV…”. The reason might be that surveys are usually smaller, more frequent data collection activities, and censuses are usually collect large-scale data periodicly. Thus censuses data source may have more comprehensive data.

Step 4: Tidy column variables.

a. Some data is collected in the same column (Data.Item), it needs to be split into different columns and also add new variables.

```
strawberry <- strawberry |>
  separate(
    col = `Data.Item`,
    into = c("Fruit", "Rest"),
    sep = " - ",
    remove = FALSE,
    extra = "merge",
    fill = "right"
  )

# split 'Rest' into 'Measure' and 'Bearing_type':
strawberry <- strawberry |>
  separate(
    col = Rest,
    into = c("Measure", "Bearing_type"),
    sep = "(?<=(ACRES|WITH))",     # separate by 'ACRES' AND 'WITH', and keep 'ACRES' in followi
ng columns.
    remove = FALSE,
    extra = "merge",
    fill = "left"
  ) |>
  select(-Rest, -Fruit, -Data.Item)
```

Step 5: Change the exception character in ‘VALUE’ to NA.

```
footnotes_v <- strawberry %>%
   filter(!is.na(Value) & !grepl("^[0-9]+(\\.[0-9]+)?(,[0-9]{1,3})*$", Value)) %>%
  distinct(Value)
strawberry <- strawberry %>% mutate(Value = na_if(Value, "(NA)"))
strawberry$Value<-as.numeric(str_replace(strawberry$Value,",",""))
```

```
## Warning: 强制改变过程中产生了NA
```

Export the cleared data.

```
write.csv(strawberry, file = "cleaned_strawberry_data.csv", row.names = FALSE)
```

Step 6: Tidy the chemical data.

```
library(tidyr)
library(dplyr)
library(stringr)
## Tidy'Domain.Category' column.
strawberry2<-read.csv("cleaned_strawberry_data.csv")
strawberry3 <- strawberry2 %>%
  extract(`Domain.Category`,
          into = c("Chemical_Type", "Specific_Chemical", "Chemical_Name", "Quantity"),
          regex = "([^,]+),?\\s*([^:]+)?:?\\s*\\(([^=]+)?\\s*=\\s*([0-9]+)?\\)",
          remove = FALSE) %>%
  mutate(Chemical_Type = ifelse(Chemical_Type == "NOT SPECIFIED", NA, Chemical_Type),
         Specific_Chemical = ifelse(is.na(Specific_Chemical) | Specific_Chemical == "", NA, Spe
cific_Chemical),
         Chemical_Name = ifelse(is.na(Chemical_Name) | Chemical_Name == "", NA, Chemical_Name),
         Quantity = ifelse(is.na(Quantity) | Quantity == "", NA, Quantity),
         Specific_Chemical = ifelse(grepl("FERTILIZER", Chemical_Type) & is.na(Specific_Chemica
l), "FERTILIZER", Specific_Chemical),
    Chemical_Name = ifelse(grepl("FERTILIZER", Chemical_Type) & is.na(Chemical_Name),
                           str_extract(Chemical_Type, "(?<=FERTILIZER:\\().+?(?=\\))"),
                           Chemical_Name)
  )
```

Handle the FERTILIZER situation in Domain.Category

```
strawberry3 <- strawberry3 %>%
  mutate(
    Specific_Chemical = ifelse(grepl("^FERTILIZER", strawberry2$`Domain.Category`),
                               "FERTILIZER",
                               Specific_Chemical),
    Chemical_Name = ifelse(grepl("^FERTILIZER", strawberry2$`Domain.Category`),
                           str_extract(strawberry2$`Domain.Category`, "\\((.*?)\\)"),
                           Chemical_Name)
  ) %>%
  mutate(
    Chemical_Name = str_replace_all(Chemical_Name, "[\\(\\)]", "")
  )
```

```
strawberry_update <- strawberry3 %>%
  mutate(
    Chemical_Type = ifelse(grepl("^CHEMICAL", strawberry2$`Domain.Category`),
                           "CHEMICAL",
                           Chemical_Type),
    Specific_Chemical = ifelse(grepl("^CHEMICAL", strawberry2$`Domain.Category`),
                               str_extract(strawberry2$`Domain.Category`, "(?<=CHEMICAL, ).+?(?
=: \\()"),
                               Specific_Chemical),
    Quantity = ifelse(grepl("^CHEMICAL", strawberry2$`Domain.Category`),
                      str_extract(strawberry2$`Domain.Category`, "(?<=\\().+?(?=\\))"),
                      Quantity)
  )
```

```
strawberry_update <- strawberry_update %>%
  mutate(
    Quantity = ifelse(
      Quantity == "TOTAL",
      Quantity,
      str_extract(Quantity, "\\d+")
    )
  )
```

Export the final cleaned data.

```
write.csv(strawberry_update, file = "Final cleaned strawberry.csv", row.names = FALSE)
```

Seprate Census data and Survey data

```
library(knitr)
library(kableExtra)
library(tidyverse)
library(magrittr)
```

```
##
## 载入程序包：'magrittr'
```

```
## The following object is masked from 'package:purrr':
##
##     set_names
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
straw_cen <- strawberry_update|> filter(Program=="CENSUS")
straw_sur <- strawberry_update |> filter(Program=="SURVEY")
```

Analysis for chemical in survey data.

WHO list six deadly carcinogens, this report will look for the use of these chemicals between different growing regions.

6 Deadly carcinogens:

CAPTAFOL, ethylene dibromide, GLYPHOSATE, MALATHION, DIAZINON, Dichlorodiphenyltrichloroethane(DDT).

Searching each chemical in the data 'straw_sur', there are three carcinogens could be found. Therefore, I will analysis these three carcinogens: 'GLYPHOSATE ISO. SALT', 'MALATHION', 'DIAZINON'.

diazinon: an organophosphorus insecticide glyphosate: herbicide malathion: a man-made organophosphate insecticide

Step 1: Select rows containing three chemicals， list the names of different regions where the data come from.

```
# select 'GLYPHOSATE ISO. SALT', 'MALATHION', 'DIAZINON' rows.
filtered_data <- subset(straw_sur,
                        tolower(trimws(Chemical_Name)) %in% tolower(c("GLYPHOSATE ISO. SALT",
"MALATHION", "DIAZINON")), header= True)
```

```
## Warning: In subset.data.frame(straw_sur, tolower(trimws(Chemical_Name)) %in%
##     tolower(c("GLYPHOSATE ISO. SALT", "MALATHION", "DIAZINON")),
##     header = True) :
##  extra argument 'header' will be disregarded
```

```
# Remove unwanted columns
filtered_data <- filtered_data[, !(names(filtered_data) %in% c('Program', 'State.ANSI', 'Ag.Dis
trict', 'Ag.District.Code', 'Country', 'Country.ANSI'))]

# List the names of different regions where the data come from.
unique(filtered_data$State)
```

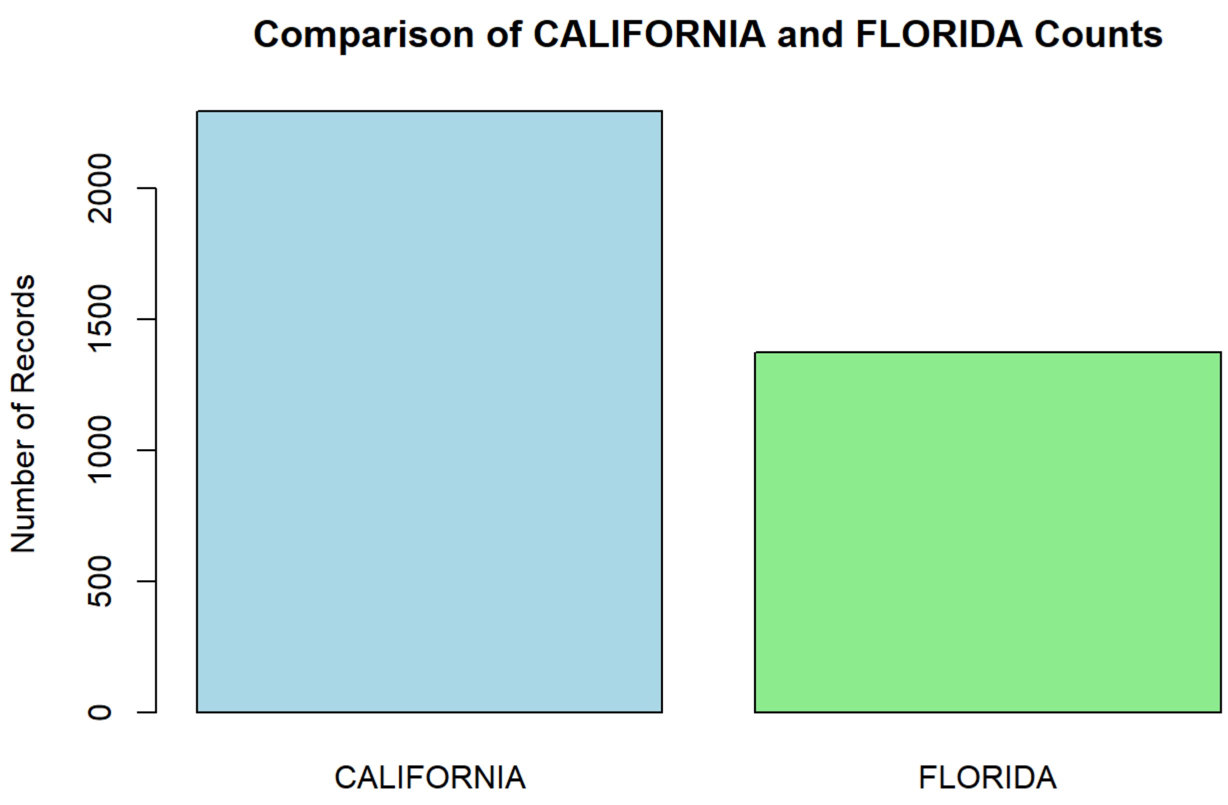
```
## [1] "CALIFORNIA" "FLORIDA"
```

It shows that strawberries which used any kind of these three chemicals come from California and Florida.

Compare the difference of survey data volume

between the two regions.

```
state_counts <- table(straw_sur$State[straw_sur$State %in% c("CALIFORNIA", "FLORIDA")])
barplot(state_counts,
        main = "Comparison of CALIFORNIA and FLORIDA Counts",
        ylab = "Number of Records",
        col = c("lightblue", "lightgreen"),
        names.arg = c("CALIFORNIA", "FLORIDA"))
```



The bar plot shows that California has more survey data records than Florida.

Step 2: Sepreat the two regions' data.

```
# Tidy the 'Quantity' cloumn
filtered_data$Quantity <- as.numeric(gsub("[^0-9]", "", filtered_data$Quantity))

# I find that there is a mistake on variable's name. 'Quantity' column should be the code of ch
emical, not the quantity. Thus change the column name.
names(filtered_data)[names(filtered_data) == "Quantity"] <- "Code"
```

```
# Split into two separate data boxes by the State column
library(dplyr)
cali <- filtered_data %>% filter(State == "CALIFORNIA")
flor <- filtered_data %>% filter(State == "FLORIDA")
```

Q1: What about the use of these three chemicals in California?

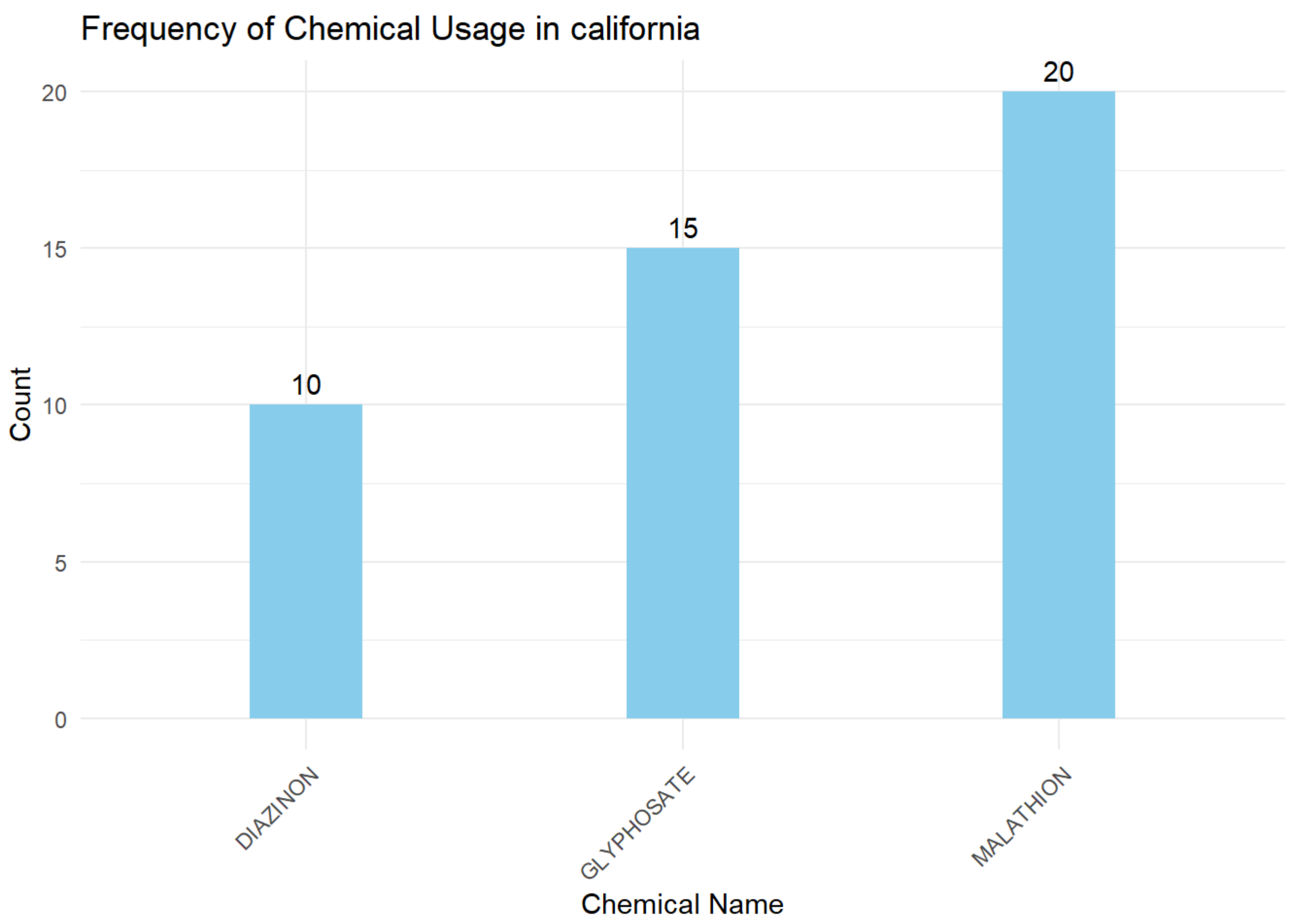
```
# Check out the column
unique(cali$Chemical_Name)
```

```
## [1] "MALATHION "            "GLYPHOSATE ISO. SALT " "DIAZINON "
```

```
diazinon_count <- sum(grepl("DIAZINON", cali$Chemical_Name))
malathion_count <- sum(grepl("MALATHION", cali$Chemical_Name))
glyphosate_count <- sum(grepl("GLYPHOSATE", cali$Chemical_Name))

chemical_data <- data.frame(
  Chemical = c("DIAZINON", "MALATHION", "GLYPHOSATE"),
  Count = c(diazinon_count, malathion_count, glyphosate_count)
)

library(ggplot2)
ggplot(chemical_data, aes(x = Chemical, y = Count)) +
  geom_bar(stat = "identity", fill = "skyblue",  width = 0.3) +
  geom_text(aes(label = Count), vjust = -0.5) +
  theme_minimal() +
  labs(title = "Frequency of Chemical Usage in california",
       x = "Chemical Name",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



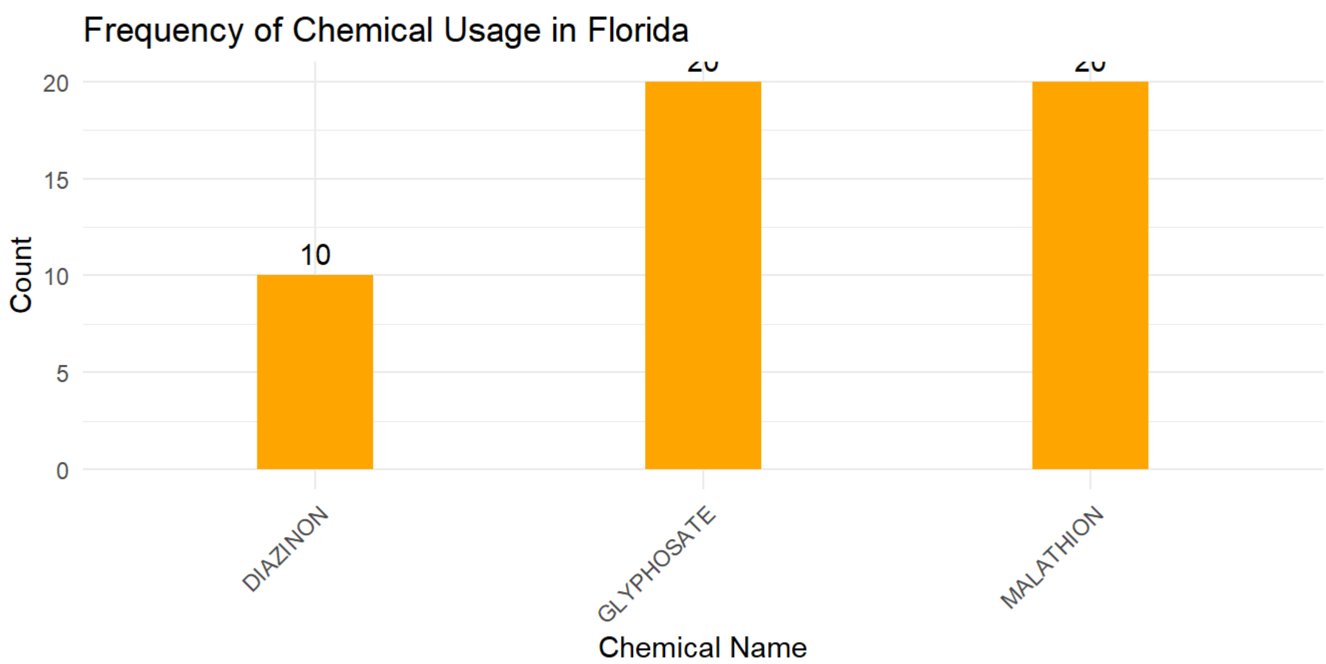
As the bar plot shows, California uses ‘MALATHION’ the most.

Q2: What about the use of these three chemicals in florida?

```
diazinon_count <- sum(grepl("DIAZINON", flor$Chemical_Name))
malathion_count <- sum(grepl("MALATHION", flor$Chemical_Name))
glyphosate_count <- sum(grepl("GLYPHOSATE", flor$Chemical_Name))

chemical_data_flor <- data.frame(
 Chemical = c("DIAZINON", "MALATHION", "GLYPHOSATE"),
 Count = c(diazinon_count, malathion_count, glyphosate_count)
)

ggplot(chemical_data_flor, aes(x = Chemical, y = Count)) +
 geom_bar(stat = "identity",
          fill = "orange",
          width = 0.3) +
 geom_text(aes(label = Count),
           vjust = -0.5) +
 theme_minimal() +
 labs(title = "Frequency of Chemical Usage in Florida",
      x = "Chemical Name",
      y = "Count") +
 theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
 coord_fixed(ratio = 0.05)
```



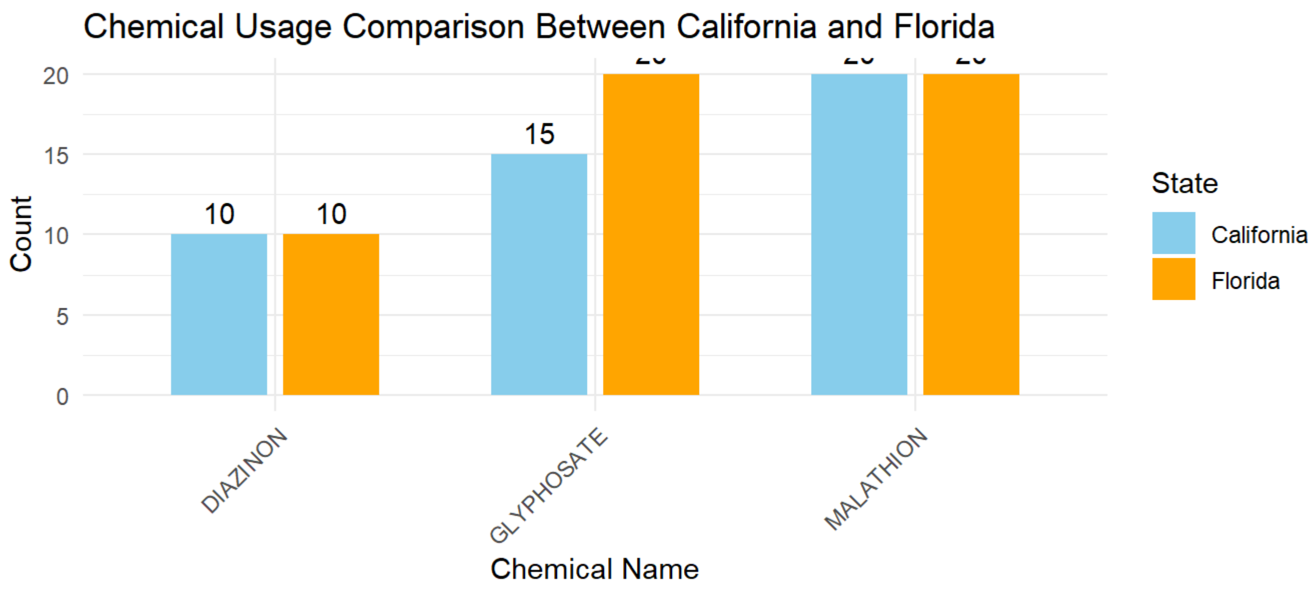
Q3: What are the similarities and differences in the use of chemical substances in the two places?

```
# California data
cali_diazinon <- sum(grepl("DIAZINON", cali$Chemical_Name))
cali_malathion <- sum(grepl("MALATHION", cali$Chemical_Name))
cali_glyphosate <- sum(grepl("GLYPHOSATE", cali$Chemical_Name))

# Florida data
flor_diazinon <- sum(grepl("DIAZINON", flor$Chemical_Name))
flor_malathion <- sum(grepl("MALATHION", flor$Chemical_Name))
flor_glyphosate <- sum(grepl("GLYPHOSATE", flor$Chemical_Name))

# combined data
combined_data <- data.frame(
  Chemical = rep(c("DIAZINON", "MALATHION", "GLYPHOSATE"), each = 2),
  Count = c(cali_diazinon, flor_diazinon,
            cali_malathion, flor_malathion,
            cali_glyphosate, flor_glyphosate),
  State = rep(c("California", "Florida"), 3)
)

ggplot(combined_data, aes(x = Chemical, y = Count, fill = State)) +
  geom_bar(stat = "identity",
           position = position_dodge(width = 0.7),
           width = 0.6) +
  geom_text(aes(label = Count),
            position = position_dodge(width = 0.7),
            vjust = -0.5) +
  theme_minimal() +
  labs(title = "Chemical Usage Comparison Between California and Florida",
       x = "Chemical Name",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values = c("skyblue", "orange")) +
  coord_fixed(ratio = 0.05)
```



As the figure shows, two region have same level of diazinon and malathion usage. The usage of malathion is twice as much as the usage of diaziono. For glyphosate, a widely used herbicide, florida use it more than california.

Some possible inference:

As diazinon and malathion are both insecticide, two pesticides were used very closely in both states, which may mean that the two pesticides are common pest control methods in agriculture in these areas.

California uses more herbicide, thus it may be a higher demand for herbicides for strawberry growing in california rather than floridia.

Acknowledgement: Use chatgpt to resolve code errors, chart exceptions, etc.