Group2-PDF

Shuting Li, Lauren Marie, Zening Ye, Keliang Xu

10/25/2021

Abstrct

After almost ten days of teamwork, we completed our assigned midterm project group.

The main work is as follows: organize and analyze the three data files, pose questions, visualize the data and solve the problems, construct maps, draw conclusions, and finally summarize the completion of the project.

The order of layout of this article is data processing, data visualization, construct maps, pose questions and conclusions, summary, and references.

Data Processing

We have three data files – strawberries, insecticides, and herbicides-fungicides-other. The first two data files are used as data basis of our project, and the third one is the background information we need in the project.

import original data

The initial documents are as follows.

```
## # A tibble: 6 x 21
##   Program  Year Period `Week Ending` `Geo Level` State      `State ANSI`
##   <chr>   <dbl> <chr>  <lgl>         <chr>       <chr>             <dbl>
## 1 CENSUS   2019 YEAR   NA            STATE       CALIFORNIA            6
## 2 CENSUS   2019 YEAR   NA            STATE       CALIFORNIA            6
## 3 CENSUS   2019 YEAR   NA            STATE       CALIFORNIA            6
## 4 CENSUS   2019 YEAR   NA            STATE       CALIFORNIA            6
## 5 CENSUS   2019 YEAR   NA            STATE       CALIFORNIA            6
## 6 CENSUS   2019 YEAR   NA            STATE       CALIFORNIA            6
## # ... with 14 more variables: Ag District <lgl>, Ag District Code <lgl>,
## #   County <lgl>, County ANSI <lgl>, Zip Code <lgl>, Region <lgl>,
## #   watershed_code <dbl>, Watershed <lgl>, Commodity <chr>, Data Item <chr>,
## #   Domain <chr>, Domain Category <chr>, Value <chr>, CV (%) <chr>
```

```
## # A tibble: 6 x 6
##   Pesticide                    Carcinogen `Hormone Disrup~ Neurotoxins `Developmental ~
##   <chr>                        <chr>      <chr>            <chr>       <chr>
## 1 <NA>                         <NA>       <NA>             <NA>        <NA>
## 2 Tetrahydrophthalimide (THPI) <NA>       <NA>             <NA>        <NA>
## 3 <NA>                         <NA>       <NA>             <NA>        <NA>
## 4 Pyraclostrobin               <NA>       <NA>             <NA>        <NA>
## 5 <NA>                         <NA>       <NA>             <NA>        <NA>
## 6 Captan                       known      <NA>             <NA>        <NA>
## # ... with 1 more variable: Bee Toxins <chr>
```

data cleaning about strawb.csv

There may be missing data during data collection process. In this case, we delete the NA conditions to ensure the accuracy of subsequent operations. In addition, the first file – strawberries also contains some unnecessary and repetitive information, which also needs to be precessed at this stage.

drop all NA columns

```
## # A tibble: 6 x 10
##   Program  Year Period State      `State ANSI` `Data Item` Domain `Domain Categor~
##   <chr>   <dbl> <chr>  <chr>             <dbl> <chr>       <chr>  <chr>
## 1 CENSUS   2019 YEAR   CALIFORNIA            6 STRAWBERRI~ ORGAN~ ORGANIC STATUS:~
## 2 CENSUS   2019 YEAR   CALIFORNIA            6 STRAWBERRI~ ORGAN~ ORGANIC STATUS:~
## 3 CENSUS   2019 YEAR   CALIFORNIA            6 STRAWBERRI~ ORGAN~ ORGANIC STATUS:~
## 4 CENSUS   2019 YEAR   CALIFORNIA            6 STRAWBERRI~ ORGAN~ ORGANIC STATUS:~
## 5 CENSUS   2019 YEAR   CALIFORNIA            6 STRAWBERRI~ ORGAN~ ORGANIC STATUS:~
## 6 CENSUS   2019 YEAR   CALIFORNIA            6 STRAWBERRI~ ORGAN~ ORGANIC STATUS:~
## # ... with 2 more variables: Value <chr>, CV (%) <chr>
```

After droping NA columns, we get strawb1 with 10 columns.

```
##  [1] "Program"         "Year"            "Period"          "State"
##  [5] "State ANSI"      "Data Item"       "Domain"          "Domain Category"
##  [9] "Value"           "CV (%)"
```

Separate 'Data Item' into 4 columns

Through the observation of the data, we found that 'Data Item' column covers a lot of data we need. There are four types of data in this column and make it messy and long, we need to separate this column into four columns.

```
## # A tibble: 2 x 1
##   Strawberries
##   <chr>
## 1 STRAWBERRIES
## 2 STRAWBERRIES - YIELD
```

```
## # A tibble: 7 x 1
##   Items
##   <chr>
## 1 " ORGANIC - OPERATIONS WITH SALES"
## 2 " ORGANIC - SALES"
## 3 " ORGANIC"
## 4 " MEASURED IN CWT / ACRE"
## 5 " MEASURED IN TONS / ACRE"
## 6 " BEARING - APPLICATIONS"
## 7 " BEARING - TREATED"
```

```
## # A tibble: 12 x 1
##    Discription
##    <chr>
##  1  <NA>
##  2 " MEASURED IN $"
##  3 " MEASURED IN CWT"
##  4 " FRESH MARKET - OPERATIONS WITH SALES"
##  5 " FRESH MARKET - SALES"
##  6 " PROCESSING - OPERATIONS WITH SALES"
```

```
##  7 " PROCESSING - SALES"
##  8 " MEASURED IN LB"
##  9 " MEASURED IN LB / ACRE / APPLICATION"
## 10 " MEASURED IN LB / ACRE / YEAR"
## 11 " MEASURED IN NUMBER"
## 12 " MEASURED IN PCT OF AREA BEARING"
```

```
## # A tibble: 4 x 1
##   Units
##   <chr>
## 1  <NA>
## 2 " MEASURED IN $"
## 3 " MEASURED IN CWT"
## 4 " AVG"
```

Separate 'Data Item' into "Strawberries", "Items", "Discription", "Units".

Separate 'Domain' into 2 columns

The same as 'Data Item' column, 'Domain' column also contains two sets of data information.

```
## # A tibble: 4 x 1
##   dname
##   <chr>
## 1 ORGANIC STATUS
## 2 TOTAL
## 3 CHEMICAL
## 4 FERTILIZER
```

```
## # A tibble: 5 x 1
##   type
##   <chr>
## 1  <NA>
## 2 " FUNGICIDE"
## 3 " HERBICIDE"
## 4 " INSECTICIDE"
## 5 " OTHER"
```

Separate 'Domain' into "dname", "type".

Separate 'Domain Category' into 2 columns

There are a lot of information in 'Domain Category' column that needs attention. Firstly, part of the information is repeated with previous column. Secondly, the information contains many unnecessary symbols such as "(",",",")".The main work of the following program is to isolate a small part of the information needed.

```
## # A tibble: 159 x 1
##    Details
##    <chr>
##  1 " (NOP USDA CERTIFIED)"
##  2  <NA>
##  3 " (AZOXYSTROBIN = 128810)"
##  4 " (BACILLUS AMYLOLIQUEFACIENS MBI 600 = 129082)"
##  5 " (BACILLUS AMYLOLIQUEFACIENS STRAIN D747 = 16482)"
##  6 " (BACILLUS PUMILUS = 6485)"
##  7 " (BACILLUS SUBT. GB03 = 129068)"
##  8 " (BACILLUS SUBTILIS = 6479)"
##  9 " (BLAD = 30006)"
```

```
## 10 " (BORAX DECAHYDRATE = 11102)"
## # ... with 149 more rows
```

```
## # A tibble: 159 x 1
##    `Chemical Name`
##    <chr>
##  1 " NOP USDA CERTIFIED"
##  2  <NA>
##  3 " AZOXYSTROBIN "
##  4 " BACILLUS AMYLOLIQUEFACIENS MBI 600 "
##  5 " BACILLUS AMYLOLIQUEFACIENS STRAIN D747 "
##  6 " BACILLUS PUMILUS "
##  7 " BACILLUS SUBT. GB03 "
##  8 " BACILLUS SUBTILIS "
##  9 " BLAD "
## 10 " BORAX DECAHYDRATE "
## # ... with 149 more rows
```

```
## # A tibble: 153 x 1
##    Number
##    <chr>
##  1  <NA>
##  2 " 128810"
##  3 " 129082"
##  4 " 16482"
##  5 " 6485"
##  6 " 129068"
##  7 " 6479"
##  8 " 30006"
##  9 " 11102"
## 10 " 128008"
## # ... with 143 more rows
```

Replicate 'Domain.Category' to new variable and separate this new variable into 'Title', 'Details'.

And then clean 'Details' and separate it into 'Chemical Name', "Number". Also captalize all words in 'Chemical Name'.

drop useless columns

After the data cleaning step, the data set looks clearly and we start to try data analysis in R. In the analysis, we found that some columns is not needed. So we dropped these useless columns in this step.

```
## # A tibble: 6 x 15
##   Program  Year Period State  `State ANSI` Items  Discription  Units dname type
##   <chr>   <dbl> <chr>  <chr>         <dbl> <chr>  <chr>        <chr> <chr> <chr>
## 1 CENSUS   2019 YEAR   CALIF~            6 " ORG~  <NA>         <NA> ORGA~ <NA>
## 2 CENSUS   2019 YEAR   CALIF~            6 " ORG~ " MEASURED ~  <NA> ORGA~ <NA>
## 3 CENSUS   2019 YEAR   CALIF~            6 " ORG~ " MEASURED ~  <NA> ORGA~ <NA>
## 4 CENSUS   2019 YEAR   CALIF~            6 " ORG~ " FRESH MAR~  <NA> ORGA~ <NA>
## 5 CENSUS   2019 YEAR   CALIF~            6 " ORG~ " FRESH MAR~ " ME~ ORGA~ <NA>
## 6 CENSUS   2019 YEAR   CALIF~            6 " ORG~ " FRESH MAR~ " ME~ ORGA~ <NA>
## # ... with 5 more variables: Title <chr>, Chemical Name <chr>, Number <chr>,
## #   Value <chr>, CV (%) <chr>
```

Drop "Strawberries", "Domain Category", because it is useless in data analysis.

data cleaning about pesti.csv

After processing the strawberries file, we begin to deal with pesti.csv. This file has less data, so the data cleaning required is relatively simple.

drop NA rows in pesti.csv and clean it

```
## # A tibble: 6 x 6
##   `Chemical Name`    Carcinogen `Hormone Disrup~ Neurotoxins `Developmental or ~
##   <chr>              <chr>      <chr>            <chr>       <chr>
## 1 TETRAHYDROPHTHALI~ <NA>       <NA>             <NA>        <NA>
## 2 PYRACLOSTROBIN     <NA>       <NA>             <NA>        <NA>
## 3 CAPTAN             known      <NA>             <NA>        <NA>
## 4 FENHEXAMID         <NA>       <NA>             <NA>        <NA>
## 5 PYRIMETHANIL       possible   suspected        <NA>        <NA>
## 6 BOSCALID           possible   <NA>             <NA>        <NA>
## # ... with 1 more variable: Bee Toxins <chr>
```

Drop NA rows in pesti.csv, finally we get pesti1 with 45 rows.

And then rename 'Pesticide' to 'Chemical Name' and captalize all words in 'Chemical Name'.

Define Human Toxins level

Based on the existing information, we hope to construct a variable to represent human toxin levels.

Use columns 'carcinogen', 'Neurotoxins', 'Developmental or Reproductive Toxins' to define human toxins level.

High toxic for human: carcinogen = known or Neurotoxins = present or `Developmental or Reproductive Toxins` = present.

Moderate toxic for human: carcinogen = probable/possible and `Hormone Disruptor` = suspect.

Slight toxic for human: carcinogen= possible/possible or `Hormone Disruptor` = suspect, only one happens.

wrangling two datasets

These two data sets use the same chemical name column(pesticide name) as the join key.

```
## # A tibble: 6 x 21
##   Program  Year Period State  `State ANSI` Items   Discription Units dname type
##   <chr>   <dbl> <chr>  <chr>         <dbl> <chr>   <chr>       <chr> <chr> <chr>
## 1 SURVEY   2019 YEAR   CALIF~            6 " BEAR~ " MEASURED~ <NA>  CHEM~ " FU~
## 2 SURVEY   2019 YEAR   CALIF~            6 " BEAR~ " MEASURED~ <NA>  CHEM~ " FU~
## 3 SURVEY   2019 YEAR   CALIF~            6 " BEAR~ " MEASURED~ <NA>  CHEM~ " FU~
## 4 SURVEY   2019 YEAR   CALIF~            6 " BEAR~ " MEASURED~ <NA>  CHEM~ " FU~
## 5 SURVEY   2019 YEAR   CALIF~            6 " BEAR~ " MEASURED~ <NA>  CHEM~ " FU~
## 6 SURVEY   2019 YEAR   CALIF~            6 " BEAR~ " MEASURED~ <NA>  CHEM~ " FU~
## # ... with 11 more variables: Title <chr>, Chemical Name <chr>, Number <chr>,
## #   Value <chr>, CV (%) <chr>, Carcinogen <chr>, Hormone Disruptor <chr>,
## #   Neurotoxins <chr>, Developmental or Reproductive Toxins <chr>,
## #   Bee Toxins <chr>, Human Toxins <chr>
```

Combine two dataset into strawbPesti.csv by key column 'Chemical Name'. Only keep rows with known pesticides.

Data Visualization

clean strawbPesti.csv

Only choose columns we will use: 'Year', 'State', 'Discription', 'Chemical Name', 'Value', 'Human Toxins'. Which is dataset "strawbPesti2".

We can see from strawbPesti2, values are different depend on measured methods for each chemical type in each state, so we can choose one method to go deep, finally we chose "MEASURED IN LB". Then we get dataset "strawbPesti3".

Then we drop no meaning rows of value. And change value into numeric variable. Finally we get dataset "strawbPesti4".

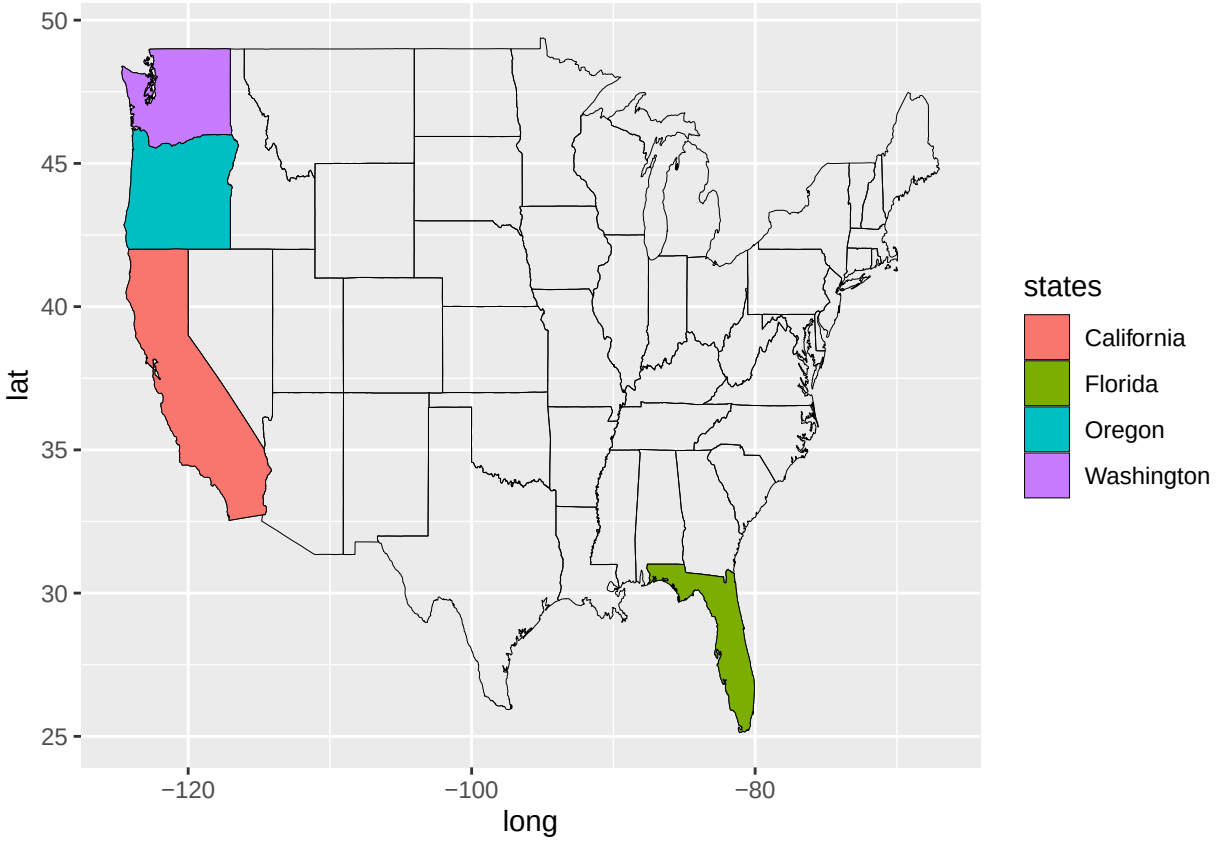
To compare pesticides' toxin level, we can take average value across 2016-1019. We build dataset "strawbPestiAVG".

Construct maps

Map of states in USA

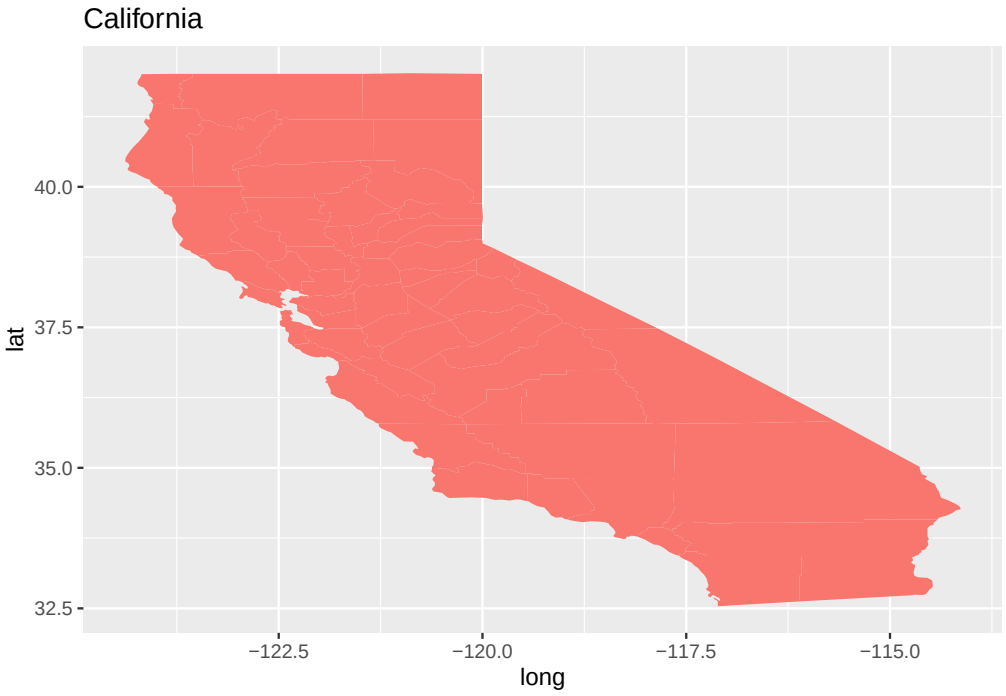
There are only four states used in data sets, which are shown in colors on the map of the United States.

```
## # A tibble: 4 x 1
##   State
##   <chr>
## 1 CALIFORNIA
## 2 FLORIDA
## 3 WASHINGTON
## 4 OREGON
```



Map of selected states

Take California as an example, we use a custom function to display a map of California.



Pose Questions and Conclusions

What pesticides do each state use and how toxic are they?

This issue is what our team members intend to start studying. Because the main content of the pesticide file is the toxins of pesticides, we began to study the degree of toxins to humans.

Preparation for data visualization

clean strawbPesti.csv

Only choose columns we will use: 'Year', 'State', 'Discription', 'Chemical Name', 'Value', 'Human Toxins'. Which is dataset "strawbPesti2".

We can see from strawbPesti2, values are different depend on measured methods for each chemical type in each state, so we can choose one method to go deep, finally we chose "MEASURED IN LB". Then we get dataset "strawbPesti3".

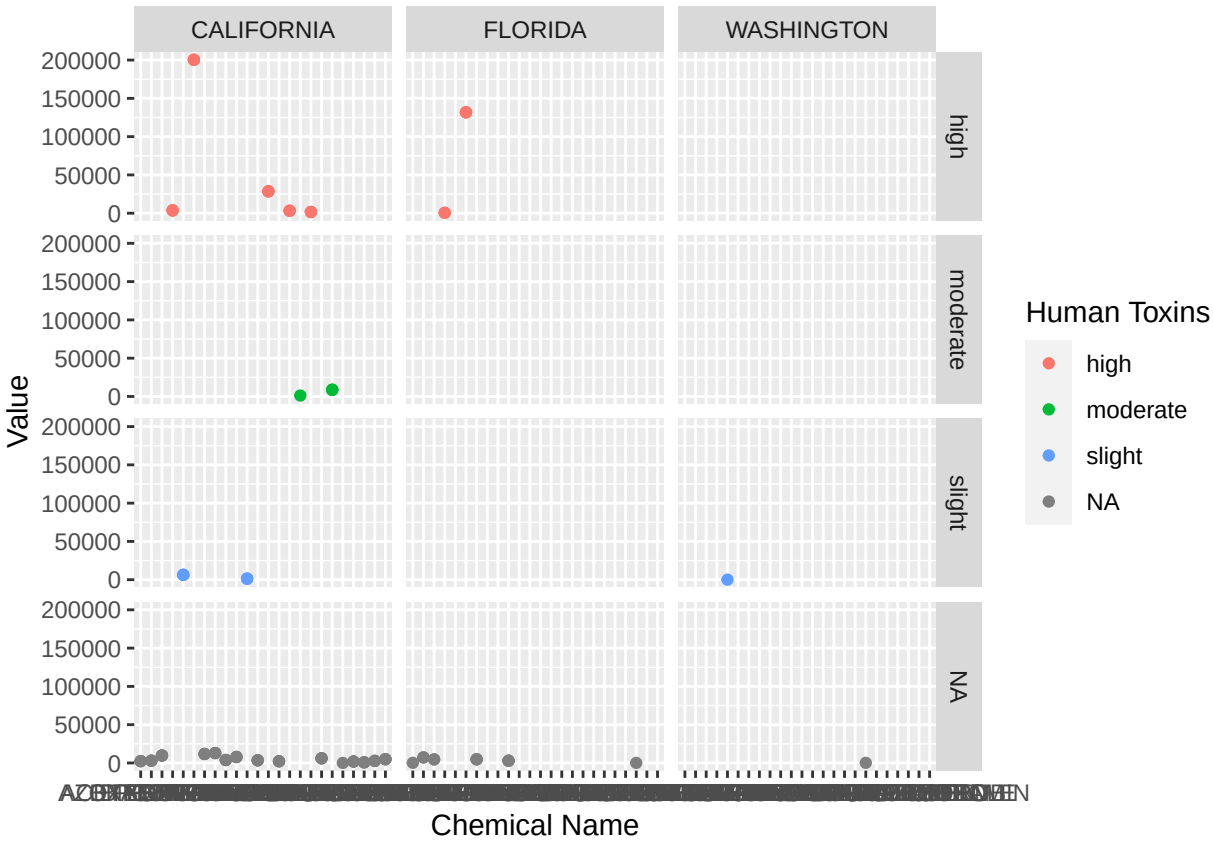
Then we drop no meaning rows of value. And change value into numeric variable. Finally we get dataset "strawbPesti4".

To compare pesticides' toxin level, we can take average value across 2016-1019. We build dataset "strawbPestiAVG".

show all states chemical usage value

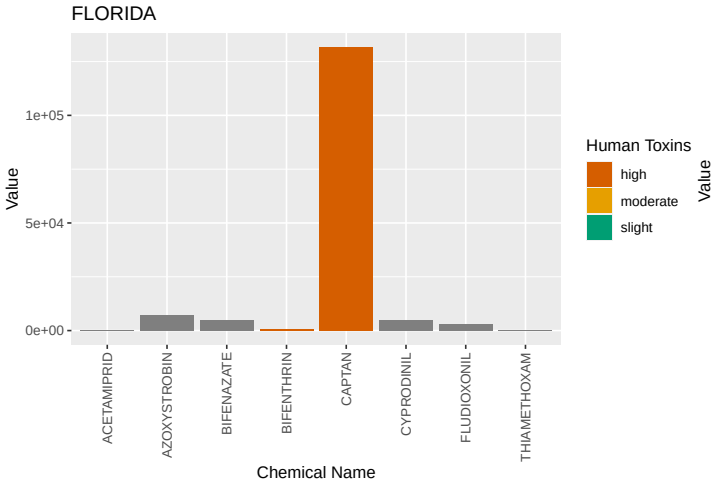
From the following plot, we can see California and Florida have relatively more points, that means California and Florida used relatively more kinds of pesticides in 2016-2019, and we can also see the values of these pesticides from this plot. At the same time, Washington state used only two kinds of pesticides through 2016-2019.

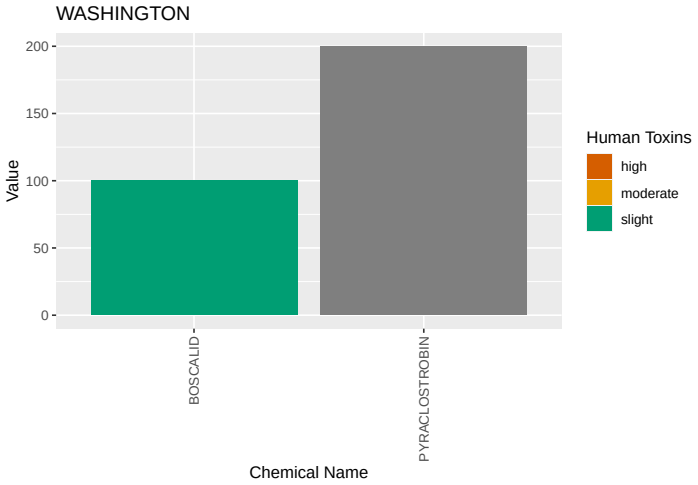
It also can be seen from the figure is that the pesticides are harmful to humans are only used in California and Florida.

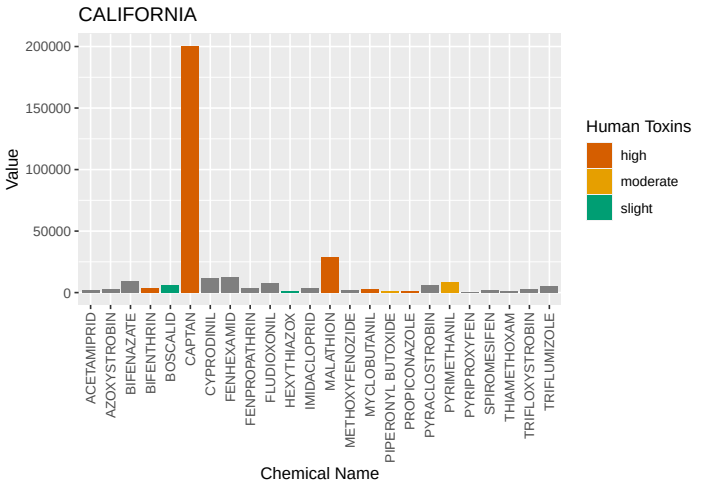

CALIFORNIA
FLORIDA
WASHINGTON
high
moderate
slight
NA
Value
200000
150000
100000
50000
0
Chemical Name
Human Toxins
high
moderate
slight
NA

Chemical usage in different states

To see the details of pesticides usage in specific state, we set a function to show different State's pesticides usage and how toxin they are.

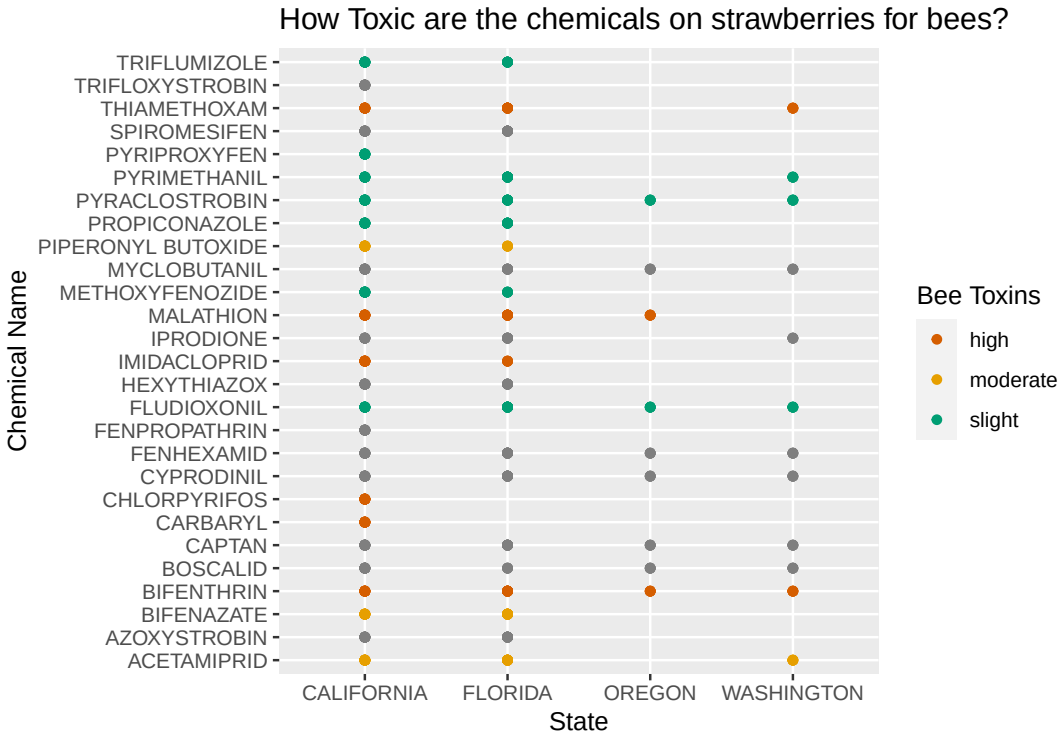




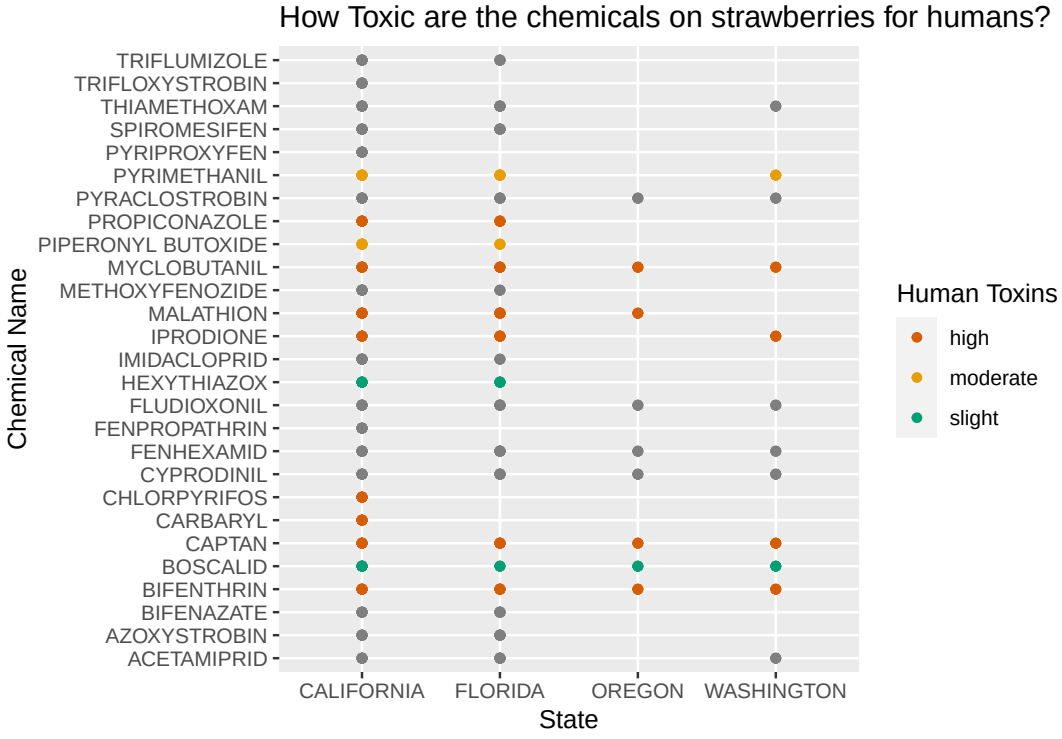


How Toxic are the chemicals on strawberries for bees?

Then let's look at more specific issues. According to the colors shown in the pictures, the hazards of various chemicals can be seen.



How Toxic are the chemicals on strawberries for humans?



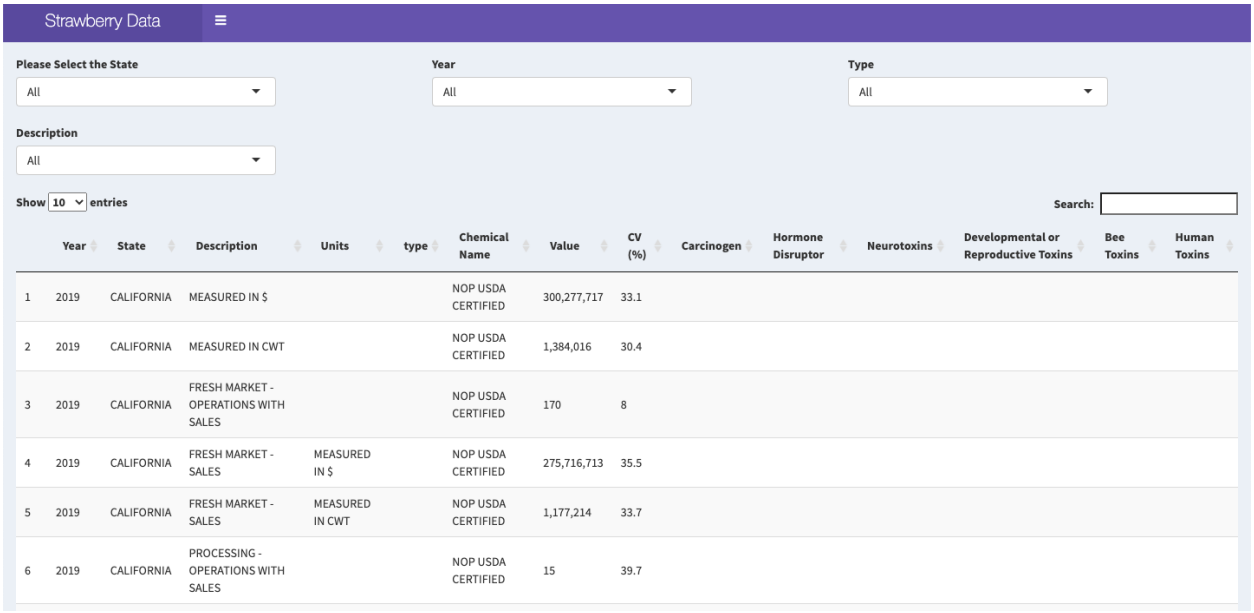
For both two questions, we can find the answer in this plot directly.

Summary

Shiny

Shiny is an R package that makes it easy to build interactive web apps straight from R.

In the shiny we made, click the button in the upper left corner to switch to view the data table, data icon, and map. In the data table, you can select the required data according to the state, year, and type. This interactive shiny file provides visual data content. Bring a more convenient and in-depth experience to those who view the data.



	Year	State	Description	Units	type	Chemical Name	Value	CV (%)	Carcinogen	Hormone Disruptor	Neurotoxins	Developmental or Reproductive Toxins	Bee Toxins	Human Toxins
1	2019	CALIFORNIA	MEASURED IN $			NOP USDA CERTIFIED	300,277,717	33.1						
2	2019	CALIFORNIA	MEASURED IN CWT			NOP USDA CERTIFIED	1,384,016	30.4						
3	2019	CALIFORNIA	FRESH MARKET - OPERATIONS WITH SALES			NOP USDA CERTIFIED	170	8						
4	2019	CALIFORNIA	FRESH MARKET - SALES	MEASURED IN $		NOP USDA CERTIFIED	275,716,713	35.5						
5	2019	CALIFORNIA	FRESH MARKET - SALES	MEASURED IN CWT		NOP USDA CERTIFIED	1,177,214	33.7						
6	2019	CALIFORNIA	PROCESSING - OPERATIONS WITH SALES			NOP USDA CERTIFIED	15	39.7						

Figure 1: A shiny App.

Pros and Cons

Because we completed this project in a short time, we learned a lot of knowledge from the team members, and there are also some areas for improvement.

What we learn?

We are more familiar with R's data processing program, and apply the knowledge in class to practical problems. At the same time, data visualization is also our main focus this time. The data is reflected in tables, charts, maps and other aspects. The realization of shiny files is another brand new attempt. The use of shiny files can realize interactive data visualization.

At the same time, the experience of group members working together to complete mid-term assignments also allows us to learn from the group members and get the experience of cooperation. We use GitHub to complete the work together. Our group communicated through the team, and the news to each other was very quick. In terms of task allocation, we are all able to complete our respective tasks well and cooperate with each other.

What needs to be improved?

In terms of time allocation, we still need better arrangements. Our program is relatively tight to complete this time. Because the time was not planned well at the beginning of the task, the work piled up before due time.

In terms of data processing, we have omitted a lot of NA data. If we can find a better way to fill in the blank data than directly delete it, we can further optimize it.

In terms of raising and solving the problem this time, the problem we raised is relatively simple, and the conclusion is also based on data visualization, and there is no specific comparison data. At this point, we can also optimize.

Reference

1.Mastering Shiny https://mastering-shiny.org/

2.R for Data Science https://r4ds.had.co.nz/index.html

3.Exploratory Data Analysis in R https://blog.datascienceheroes.com/exploratory-data-analysis-in-r-intro/