

Strawberries: exploratory data analysis

Jiaxin Li(U77902942)

2023-10-23

Jiaxin's EDA for strawberry data

Initial questions:

Initial questions about strawberries, the data, and about the work:

- (1) What is the degree of harmfulness of each chemical, which chemicals are more harmful and which are relatively harmless?
- (2) How are strawberries sold in different states and in different years? What about the use of chemicals?
- (3) Whether there is a correlation between different variables, e.g., whether strawberry sales are related to the year or the state in which they are located, etc.

References

Here are references that helps me while I'm writing.

(Besides, thanks for Aidan's help about understanding variables, dealing with missing values, etc.)

[NASS help](#)

[Quick Stats Glossary](#)

[Quick Stats Column Definitions](#)

[stats by subject](#)

for EPA number lookup [epa numbers](#)

[Active Pesticide Product Registration Informational Listing](#)

[pc number input](#) [pesticide chemical search](#)

[toxic chemical dashboard](#)

[ACToR – Aggregated Computational Toxicology Resource](#)

[comptox dashboard](#)

[pubChem](#)

Investigating toxic pesticides

References that are helpful here:

[start here with chem PC code](#)

[step 2](#) to get label (with warnings) for products using the chemical

[International Chemical safety cards](#)

[Pesticide Product and Label System](#)

[Search by Chemical](#)

[CompTox Chemicals Dashboard](#)

[Active Pesticide Product Registration Informational Listing](#)

[OSHA chemical database](#)

[Pesticide Ingredients](#)

[NPIC Product Research Online \(NPRO\)](#)

[Databases for Chemical Information](#)

[Pesticide Active Ingredients](#)

[TSCA Chemical Substance Inventory](#)

[glyphosate](#)

Matching processes

First, I need to match different chemicals' PC codes, CAS numbers and their harm ranks.

To be more specific,

- (1) The numbers following the equals sign in these chemical substance records typically represent CAS numbers (Chemical Abstracts Service Registry Numbers). CAS numbers are a standardized numbering system used to uniquely identify chemical substances. Each CAS number is unique and serves the purpose of ensuring accurate identification and retrieval of information about a specific chemical substance. These numbers do not typically convey direct toxicity or hazard meanings; instead, they are used for the unique identification of chemicals in scientific and legal documents;
- (2) In the list of 171 chemical substance records, there are 89 unique CAS numbers. The reason there are not 171 unique CAS numbers is that some chemicals share the same CAS number. This can happen when multiple substances have similar chemical structures or active ingredients and are therefore assigned the same CAS number. CAS numbers are meant to uniquely identify chemical substances, but similar or equivalent substances may share the same CAS number in some cases.

Then, In order to match the subsequent PC codes with their corresponding CAS numbers, all the PC numbers in the “Data Items” column are first extracted into a new separate column variable using regular expressions.

```
library(dplyr)
library(writexl)

strwb_survey_chem_CAS <- strwb_survey_chem |>
  mutate(chemical_name = sub(".*\\((.*)\\s=.*", "\\1", temp43),
         PCcodes = sub(".*=\\s(.*)\\)", "\\1", temp43)) |>
  select(Year, State, temp23, chemical_name, PCcodes, Value)

# colnames(strwb_survey_chem_CAS)
# View(strwb_survey_chem_CAS)
# write_xlsx(strwb_survey_chem_CAS, path = "strwb_survey_chem_CAS.xlsx")
```

Data visualization about the usage of the four main groups of chemicals

Data visualization was used here to initially explore the use of the four main groups of chemicals (“FUNGICIDE”, “HERBICIDE”, “INSECTICIDE”, “OTHER”) in different states in different years:

- (1) Here I created line plots using ggplot2 to visualize the proportion of different chemicals usage over time (from 2016 to 2018 to 2019 to 2021) for four different states (CALIFORNIA, FLORIDA, OREGON, WASHINGTON). The proportion is defined as the count of a specific kind of chemical divided by the sum of the counts of the four types of chemicals: 'FUNGICIDE', 'HERBICIDE', 'INSECTICIDE' and 'OTHER';
- (2) 'OREGON' and 'WASHINGTON' have data for 2016 but not for other years, it is likely due to a lack of data availability for these states in the subsequent years (2018, 2019, and 2021) in the data set. In other words, the data set we are working with might not include information for those states in those specific years, resulting in no data points for those combinations."

```
unique(strwb_survey_chem_CAS$Year)
```

```
[1] 2021 2019 2018 2016
```

```
unique(strwb_survey_chem_CAS$State)
```

```
[1] "CALIFORNIA" "FLORIDA"      "OREGON"      "WASHINGTON"
```

```
unique(strwb_survey_chem_CAS$temp23)
```

```
[1] " FUNGICIDE"  " HERBICIDE"  " INSECTICIDE" " OTHER"
```

```
library(ggplot2)
```

```
library(dplyr)
```

```
states_of_interest <- c("CALIFORNIA", "FLORIDA", "OREGON", "WASHINGTON")
```

```
filtered_data <- strwb_survey_chem_CAS %>%
```

```
  filter(State %in% states_of_interest)
```

```
proportion_data <- filtered_data %>%
```

```
  group_by(Year, State) %>%
```

```
  summarise(
```

```
    fungicide_count = sum(temp23 == " FUNGICIDE"),
```

```
    total_count = sum(temp23 %in% c(" FUNGICIDE", " HERBICIDE", " INSECTICIDE", " OTHER"))
```

```
  ) %>%
```

```
  mutate(proportion = round(fungicide_count / total_count, 3))
```

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

```
library(ggplot2)
library(dplyr)

states_of_interest <- c("CALIFORNIA", "FLORIDA", "OREGON", "WASHINGTON")
filtered_data <- strwb_survey_chem_CAS %>%
  filter(State %in% states_of_interest)

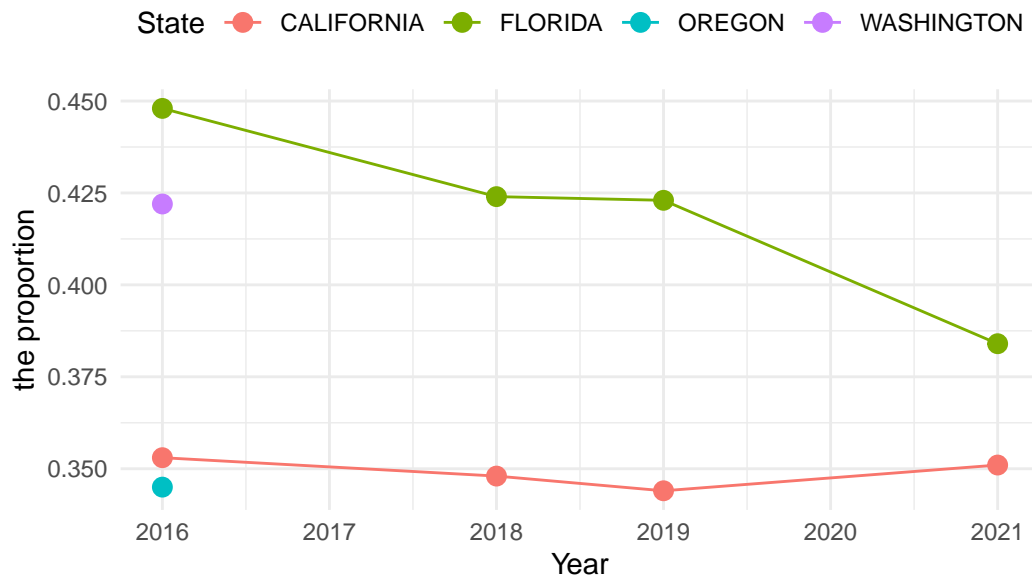
proportion_data <- filtered_data %>%
  group_by(Year, State) %>%
  summarise(
    fungicide_count = sum(temp23 == " FUNGICIDE"),
    total_count = sum(temp23 %in% c(" FUNGICIDE", " HERBICIDE", " INSECTICIDE", " OTHER"))
  ) %>%
  mutate(proportion = round(fungicide_count / total_count, 3))
```

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

```
ggplot(proportion_data, aes(x = Year, y = proportion, color = State)) +
  geom_line() +
  geom_point(aes(label = sprintf("%.3f", proportion)), size = 3) +
  labs(
    x = "Year",
    y = "the proportion",
    title = "The Proportion of the Utility of Fungicide"
  ) +
  theme_minimal() +
  theme(legend.position = "top")
```

Warning in geom_point(aes(label = sprintf("%.3f", proportion)), size = 3):
Ignoring unknown aesthetics: label

The Proportion of the Utility of Fungicide



```
states_of_interest <- c("CALIFORNIA", "FLORIDA", "OREGON", "WASHINGTON")
filtered_data <- strwb_survey_chem_CAS %>%
  filter(State %in% states_of_interest)

proportion_data <- filtered_data %>%
  group_by(Year, State) %>%
  summarise(
    fungicide_count = sum(temp23 == " HERBICIDE"),
    total_count = sum(temp23 %in% c(" FUNGICIDE", " HERBICIDE", " INSECTICIDE", " OTHER"))
  ) %>%
  mutate(proportion = round(fungicide_count / total_count, 3))
```

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

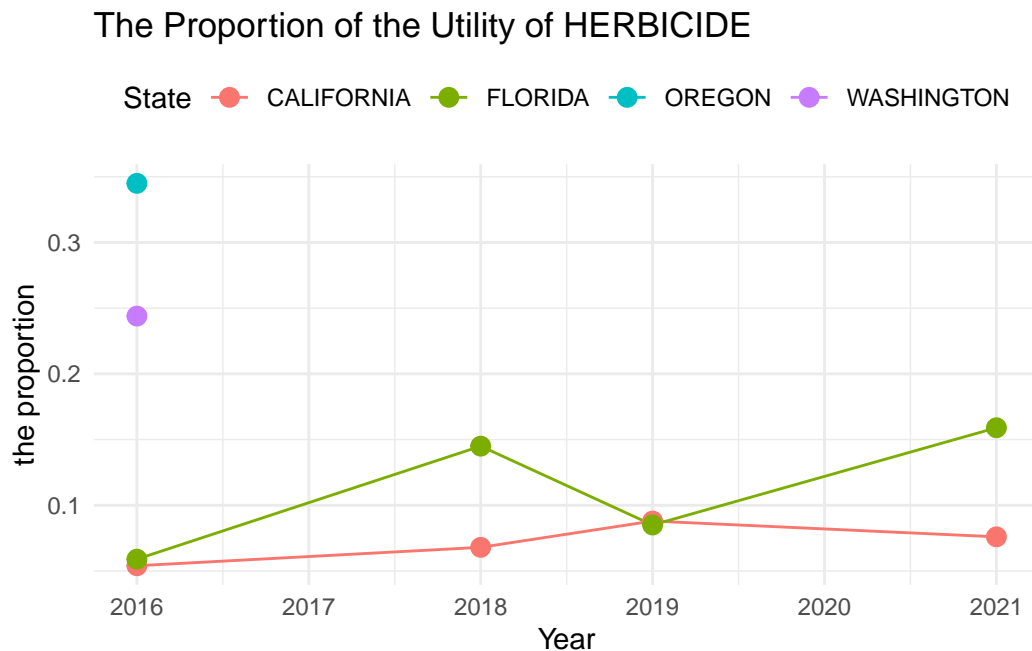
```
ggplot(proportion_data, aes(x = Year, y = proportion, color = State)) +
  geom_line() +
  geom_point(aes(label = sprintf("%.3f", proportion)), size = 3) +
  labs(
    x = "Year",
    y = "the proportion",
```

```

    title = "The Proportion of the Utility of HERBICIDE"
  ) +
  theme_minimal() +
  theme(legend.position = "top")

```

Warning in `geom_point(aes(label = sprintf("%.3f", proportion)), size = 3)`:
 Ignoring unknown aesthetics: label



```

states_of_interest <- c("CALIFORNIA", "FLORIDA", "OREGON", "WASHINGTON")
filtered_data <- strwb_survey_chem_CAS %>%
  filter(State %in% states_of_interest)

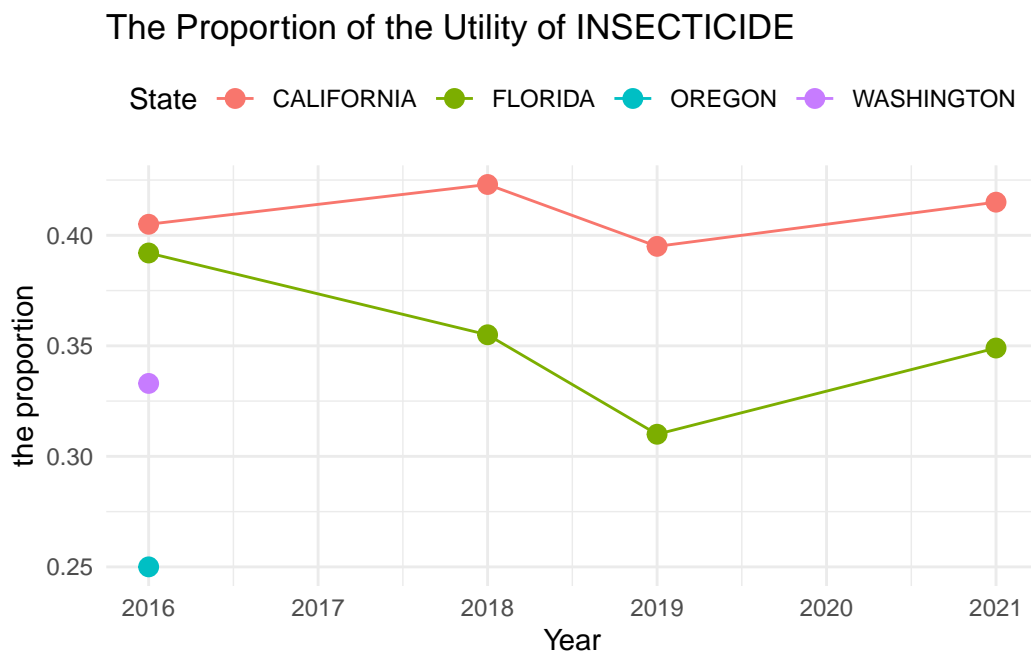
proportion_data <- filtered_data %>%
  group_by(Year, State) %>%
  summarise(
    fungicide_count = sum(temp23 == " INSECTICIDE"),
    total_count = sum(temp23 %in% c(" FUNGICIDE", " HERBICIDE", " INSECTICIDE", " OTHER"))
  ) %>%
  mutate(proportion = round(fungicide_count / total_count, 3))

```

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

```
ggplot(proportion_data, aes(x = Year, y = proportion, color = State)) +  
  geom_line() +  
  geom_point(aes(label = sprintf("%.3f", proportion)), size = 3) +  
  labs(  
    x = "Year",  
    y = "the proportion",  
    title = "The Proportion of the Utility of INSECTICIDE"  
  ) +  
  theme_minimal() +  
  theme(legend.position = "top")
```

Warning in geom_point(aes(label = sprintf("%.3f", proportion)), size = 3):
Ignoring unknown aesthetics: label



```
states_of_interest <- c("CALIFORNIA", "FLORIDA", "OREGON", "WASHINGTON")  
filtered_data <- strwb_survey_chem_CAS %>%  
  filter(State %in% states_of_interest)
```



```

proportion_data <- filtered_data %>%
  group_by(Year, State) %>%
  summarise(
    fungicide_count = sum(temp23 == " OTHER"),
    total_count = sum(temp23 %in% c(" FUNGICIDE", " HERBICIDE", " INSECTICIDE", " OTHER"))
  ) %>%
  mutate(proportion = round(fungicide_count / total_count, 3))

```

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

```

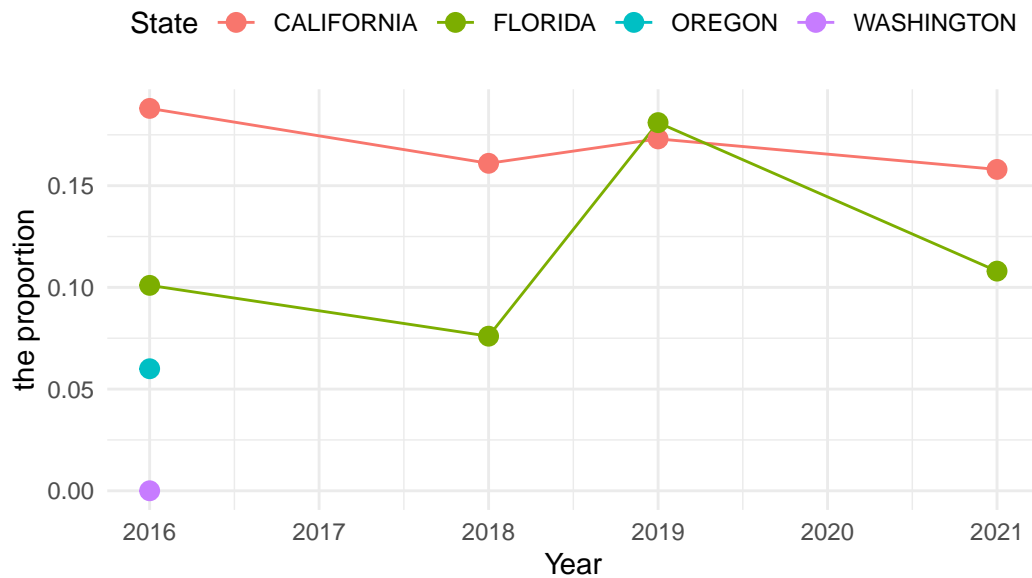
lineplot4 <- ggplot(proportion_data, aes(x = Year, y = proportion, color = State)) +
  geom_line() +
  geom_point(aes(label = sprintf("%.3f", proportion)), size = 3) +
  labs(
    x = "Year",
    y = "the proportion",
    title = "The Proportion of the Utility of OTHER"
  ) +
  theme_minimal() +
  theme(legend.position = "top")

```

Warning in geom_point(aes(label = sprintf("%.3f", proportion)), size = 3):
 Ignoring unknown aesthetics: label

```
lineplot4
```

The Proportion of the Utility of OTHER



Here, we could quickly check that there is data missing, definitely. And take the 2018 year for an example:

```
desired_year <- 2018
desired_state <- "OREGON"

subset_data <- subset(strwb_survey_chem_CAS, Year == desired_year & State == desired_state)
print(subset_data)
```

```
# A tibble: 0 x 6
# i 6 variables: Year <dbl>, State <chr>, temp23 <chr>, chemical_name <chr>,
#   PCcodes <chr>, Value <chr>
```

```
desired_year <- 2018
desired_state <- "WASHINGTON"

subset_data <- subset(strwb_survey_chem_CAS, Year == desired_year & State == desired_state)
# print(subset_data)
```

```
strwb_survey_mkt <- strwb_survey_mkt[!grepl("\\(D\\)", strwb_survey_mkt$Value), ]
strwb_survey_mkt$Value <- as.numeric(strwb_survey_mkt$Value)
```

Warning: NA

```
strwb_survey_mkt_grouped <- strwb_survey_mkt |>
  group_by(Year, State, temp1b) |>
  summarise(
    average_value = mean(Value)
  )
```

`summarise()` has grouped output by 'Year', 'State'. You can override using the `groups` argument.

```
# View(strwb_survey_mkt_grouped)
```

Exploring different chemicals' harm ranks

In the following, I will match the PC codes of the chemical substances in the existing dataset to their corresponding CAS numbers, and then match them to the hazard classes of the corresponding chemical substances, with the help of the EXCEL datasheet provided by the professor, which contains CAS numbers and hazard classes.

And about the harms of different chemicals with different CAS numbers, we can learn from the file that prof shared that “INDEX. CLASS IFICATION OF PESTICIDE AC TIVE INGREDIENTS: Ia = Extremely hazardous; Ib= Highly hazardous; II=Moderately hazardous; III=Slightly hazardous; U = Unlikely to present acute hazard in normal use; FM =Fumigant, not classified; O = Obsolete as pesticide, not classified.”

```
cas_and_harm <- readxl::read_xlsx("CAS.xlsx")
# View(cas_and_harm)
```

```
cas <- as.vector(unique(strwb_survey_chem_CAS$PCcodes))
```

```
PCcodes <- cas[nchar(cas) == 6 & grepl("[0-9]+$", cas)]
```

```
PCcodes <- as.data.frame(PCcodes)
```

```
PCcodes$cas_valid <- c("23564-05-8", "15299-99-7", "38641-94-0", "70901-12-1", "13356-08-6")
# View(PCcodes)
```

```
library(dplyr)
```

```
strwb_survey_chem_CAS <- merge(strwb_survey_chem_CAS, PCcodes, by = "PCcodes")
```

```
strwb_survey_chem_CAS <- merge(strwb_survey_chem_CAS, cas_and_harm, by = "cas_valid")
```

```
# View(strwb_survey_chem_CAS)
# write_xlsx(strwb_survey_chem_CAS, path = "data_with_cas_and_harms.xlsx")
```

Dealing with the missing values

Handling of missing values in the “Value” variable: Calculate the mean value of the “Value” variable from valid values other than “(D)” and assign this calculated mean value to “(D)”, “(NA)” and “(Z)”. (Thanks for Aidan’s inspiration here!)

```
non_numeric_values <- c("(D)", "(NA)", "(Z)")
strwb_survey_chem_CAS$Value[strwb_survey_chem_CAS$Value %in% non_numeric_values] <- NA
mean_value <- mean(as.numeric(strwb_survey_chem_CAS$Value), na.rm = TRUE)
```

```
Warning in mean(as.numeric(strwb_survey_chem_CAS$Value), na.rm = TRUE):
NA
```

```
strwb_survey_chem_CAS$Value[is.na(strwb_survey_chem_CAS$Value)] <- mean_value
strwb_survey_chem_CAS$Value <- as.numeric(strwb_survey_chem_CAS$Value)
```

```
Warning:      NA
```

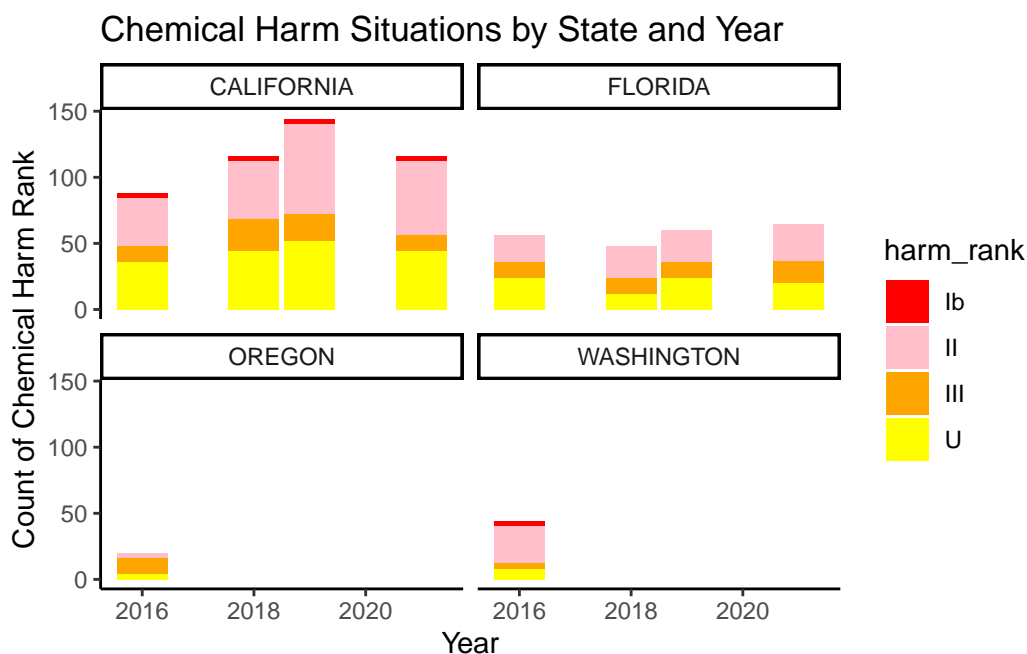
```
strwb_survey_chem_CAS$Value <- round(strwb_survey_chem_CAS$Value, digits = 3)

head(strwb_survey_chem_CAS)
```

	cas_valid	PCcodes	Year	State	temp23	chemical_name	Value	harm_rank
1	103361-09-7	121011	2021	FLORIDA	HERBICIDE	CLETHODIM	38.188	III
2	103361-09-7	121011	2016	OREGON	HERBICIDE	CLETHODIM	38.188	III
3	103361-09-7	121011	2016	OREGON	HERBICIDE	CLETHODIM	38.188	III
4	103361-09-7	121011	2021	FLORIDA	HERBICIDE	CLETHODIM	38.188	III
5	103361-09-7	121011	2021	FLORIDA	HERBICIDE	CLETHODIM	38.188	III
6	103361-09-7	121011	2016	OREGON	HERBICIDE	CLETHODIM	38.188	III

```
#colnames(strwb_survey_chem_CAS)
#unique(strwb_survey_chem_CAS$harm_rank)
#unique(strwb_survey_chem_CAS$Year)
#unique(strwb_survey_chem_CAS$State)
```

```
library(ggplot2)
ggplot(data = strwb_survey_chem_CAS, aes(x = Year, fill = harm_rank)) +
  geom_bar(position = "stack") +
  facet_wrap(~ State) +
  labs(title = "Chemical Harm Situations by State and Year",
       x = "Year",
       y = "Count of Chemical Harm Rank") +
  scale_fill_manual(values = c("Ib" = "red", "II" = "pink", "III" = "orange", "U" = "yellow")) +
  theme_classic()
```



From this, it can be seen that the harm ranks of chemicals used in different states in different years are predominantly II (Moderately hazardous) and U (Unlikely to present acute hazard in normal use.) different years is predominantly level 2 and level 5, which means that the hazards of the chemicals used between 2016 and 2021 are relatively acceptable.

Data distributions and regression models

exploring different states' situations about the harm ranks of chemicals

In addition, although the above analysis of the hazards of chemical substances has been carried out on a granular basis for each chemical substance, there are other factors that may have an

impact on the harmlessness of different chemical substances in practice, such as the dosage used, whether strawberries are washed carefully, and so on.

explore data in data frame “strwb_survey_chem” and “strwb_survey_mkt”

Here I want to explore data in data frame “strwb_survey_chem”, especially I’m curious about the relationships among variables like “State”, “Year”, “Value”. So I deal with the missing values in column “Value” first, then I could try to create some visualization plots and fit some models as below. This means that

```
non_numeric_values <- c("(D)", "(NA)", "(Z)")
strwb_survey_chem$Value[strwb_survey_chem$Value %in% non_numeric_values] <- NA
mean_value <- mean(as.numeric(strwb_survey_chem$Value), na.rm = TRUE)
```

Warning in mean(as.numeric(strwb_survey_chem\$Value), na.rm = TRUE):
NA

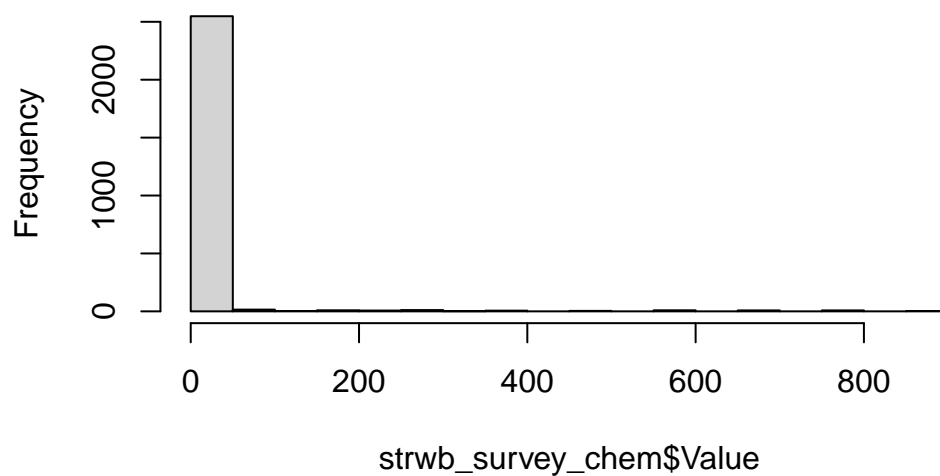
```
strwb_survey_chem$Value[is.na(strwb_survey_chem$Value)] <- mean_value
strwb_survey_chem$Value <- as.numeric(strwb_survey_chem$Value)
```

Warning: NA

```
strwb_survey_chem$Value <- round(strwb_survey_chem$Value, digits = 3)
# View(strwb_survey_chem)

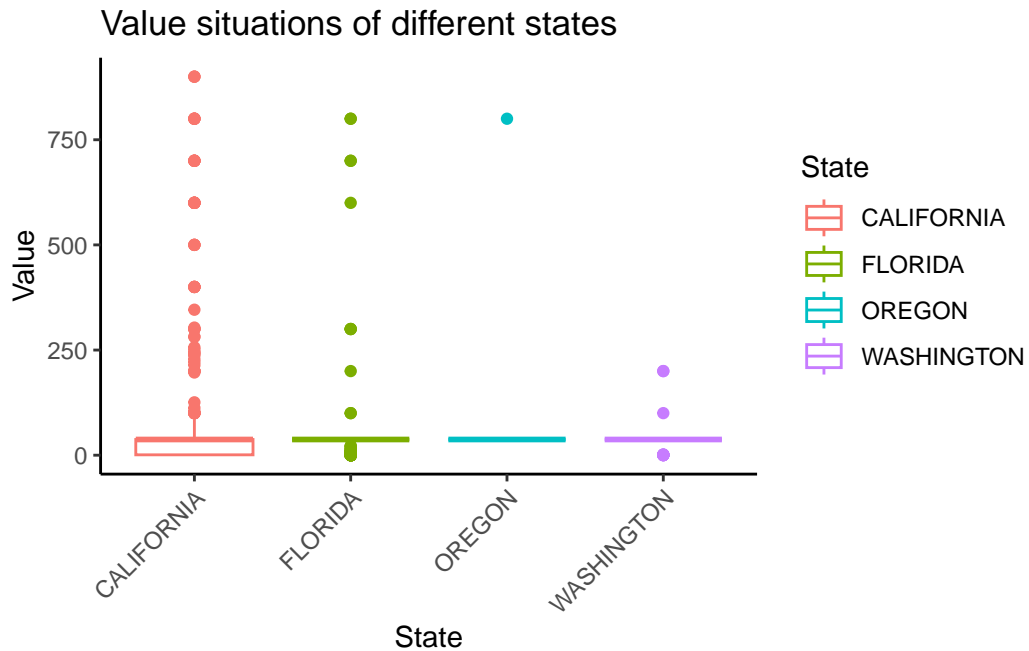
hist(strwb_survey_chem$Value)
```

Histogram of strwb_survey_chem\$Value



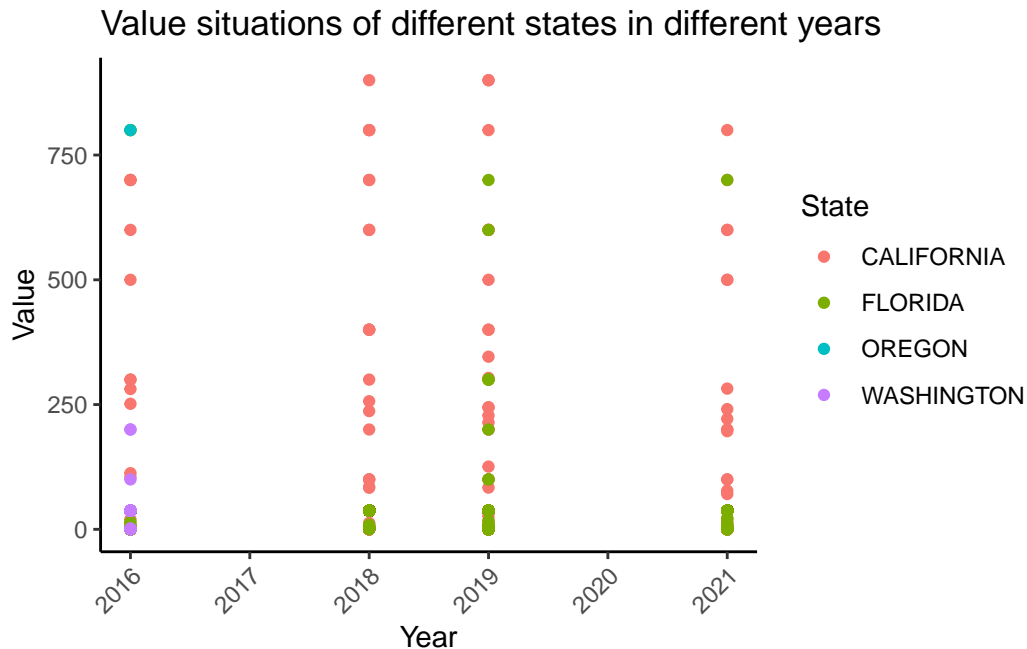
```
ggplot(data = strwb_survey_chem, aes(x=State, y=Value, color = State)) +  
  geom_boxplot() +  
  theme_classic() +  
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +  
  labs(  
    title = "Value situations of different states"  
  )
```

Warning: Removed 246 rows containing non-finite values (`stat_boxplot()`).



```
ggplot(data = strwb_survey_chem, aes(x=Year, y=Value, color = State)) +
  geom_point() +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(
    title = "Value situations of different states in different years"
  )
```

Warning: Removed 246 rows containing missing values (`geom_point()`).



Then I try to explore whether the factors like state, year, chemical types could influence the Value, so I fit a generalized linear model(model1) and linear model(model2) below.

But in both outputs, many coefficients has relatively large standard deviation, and the t-stat indicates I fail to reject the Null Hypothesis(All the coefficients are equal to zero), so these two model don't perform well. In the future, I will try more model to explore which model could fit well.

It turns out that the variable data are over dispersed so there are many outliers and the usual linear models don't perform well.

```
library(rstanarm)
```

Rcpp

This is rstanarm version 2.21.4

- See <https://mc-stan.org/rstanarm/articles/priors> for changes to default priors!
- Default priors may change, so it's safest to specify priors, even if equivalent to the default
- For execution on a local, multicore CPU with excess RAM we recommend calling

```
options(mc.cores = parallel::detectCores())
```

```
model1 <- stan_glm(Value~Year + State + temp23, data=strwb_survey_chem, refresh=0)  
summary(model1)
```

Model Info:

```
function:      stan_glm  
family:        gaussian [identity]  
formula:       Value ~ Year + State + temp23  
algorithm:     sampling  
sample:        4000 (posterior sample size)  
priors:        see help('prior_summary')  
observations:  2633  
predictors:    8
```

Estimates:

	mean	sd	10%	50%	90%
(Intercept)	261.9	1851.2	-2122.6	243.9	2645.7
Year	-0.1	0.9	-1.3	-0.1	1.1
StateFLORIDA	2.8	3.5	-1.7	2.8	7.3
StateOREGON	10.9	9.6	-1.4	10.8	23.2
StateWASHINGTON	3.4	7.7	-6.3	3.3	13.1
temp23 HERBICIDE	8.6	5.6	1.6	8.5	15.8
temp23 INSECTICIDE	10.9	3.6	6.2	10.9	15.7
temp23 OTHER	15.4	4.9	9.0	15.3	21.6
sigma	81.3	1.1	79.8	81.3	82.7

Fit Diagnostics:

	mean	sd	10%	50%	90%
mean_PPD	37.3	2.2	34.5	37.3	40.1

The mean_ppd is the sample average posterior predictive distribution of the outcome variable

MCMC diagnostics

	mcse	Rhat	n_eff
(Intercept)	28.4	1.0	4243
Year	0.0	1.0	4244
StateFLORIDA	0.1	1.0	4186
StateOREGON	0.1	1.0	4270
StateWASHINGTON	0.1	1.0	4145

```
temp23 HERBICIDE    0.1  1.0 3753
temp23 INSECTICIDE  0.1  1.0 3939
temp23 OTHER        0.1  1.0 3827
sigma               0.0  1.0 4192
mean_PPD            0.0  1.0 4280
log-posterior       0.0  1.0 2005
```

For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective

```
model2 <- lm(Value~Year + State + temp23, data=strwb_survey_chem)
summary(model2)
```

Call:

```
lm(formula = Value ~ Year + State + temp23, data = strwb_survey_chem)
```

Residuals:

```
      Min       1Q   Median       3Q      Max
-43.97 -27.76  -5.31   0.34  871.60
```

Coefficients:

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   280.8700   1849.5516   0.152  0.87931
Year          -0.1250     0.9163  -0.136  0.89146
StateFLORIDA    2.9301     3.4411   0.852  0.39456
StateOREGON    11.0538     9.6052   1.151  0.24991
StateWASHINGTON 3.2225     7.8243   0.412  0.68047
temp23 HERBICIDE  8.5870     5.6532   1.519  0.12889
temp23 INSECTICIDE 10.9387     3.6665   2.983  0.00288 **
temp23 OTHER    15.3162     4.9430   3.099  0.00197 **
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 81.24 on 2625 degrees of freedom
( 246 )
```

```
Multiple R-squared:  0.005629, Adjusted R-squared:  0.002977
```

```
F-statistic: 2.123 on 7 and 2625 DF, p-value: 0.03821
```

The last part of my EDA is about exploring data in the “strwb_survey_mkt”, to explore the relationships among “state”, “year”, “Value”, etc.

```

mean_value <- mean(as.numeric(strwb_survey_mkt$Value), na.rm = TRUE)
strwb_survey_mkt$Value[is.na(strwb_survey_mkt$Value)] <- mean_value
strwb_survey_mkt$Value <-round(strwb_survey_mkt$Value, digits = 3)
# View(strwb_survey_mkt)

model3 <- stan_glm(Value~Year + State, data=strwb_survey_mkt, refresh=0)
summary(model1)

```

Model Info:

```

function:      stan_glm
family:        gaussian [identity]
formula:       Value ~ Year + State + temp23
algorithm:     sampling
sample:        4000 (posterior sample size)
priors:        see help('prior_summary')
observations:  2633
predictors:    8

```

Estimates:

	mean	sd	10%	50%	90%
(Intercept)	261.9	1851.2	-2122.6	243.9	2645.7
Year	-0.1	0.9	-1.3	-0.1	1.1
StateFLORIDA	2.8	3.5	-1.7	2.8	7.3
StateOREGON	10.9	9.6	-1.4	10.8	23.2
StateWASHINGTON	3.4	7.7	-6.3	3.3	13.1
temp23 HERBICIDE	8.6	5.6	1.6	8.5	15.8
temp23 INSECTICIDE	10.9	3.6	6.2	10.9	15.7
temp23 OTHER	15.4	4.9	9.0	15.3	21.6
sigma	81.3	1.1	79.8	81.3	82.7

Fit Diagnostics:

	mean	sd	10%	50%	90%
mean_PPD	37.3	2.2	34.5	37.3	40.1

The mean_ppd is the sample average posterior predictive distribution of the outcome variable

MCMC diagnostics

	mcse	Rhat	n_eff
(Intercept)	28.4	1.0	4243
Year	0.0	1.0	4244

StateFLORIDA	0.1	1.0	4186
StateOREGON	0.1	1.0	4270
StateWASHINGTON	0.1	1.0	4145
temp23 HERBICIDE	0.1	1.0	3753
temp23 INSECTICIDE	0.1	1.0	3939
temp23 OTHER	0.1	1.0	3827
sigma	0.0	1.0	4192
mean_PPD	0.0	1.0	4280
log-posterior	0.0	1.0	2005

For each parameter, mcse is Monte Carlo standard error, n_eff is a crude measure of effective

```
model4 <- lm(Value~Year + State, data=strwb_survey_mkt)
summary(model2)
```

Call:

```
lm(formula = Value ~ Year + State + temp23, data = strwb_survey_chem)
```

Residuals:

Min	1Q	Median	3Q	Max
-43.97	-27.76	-5.31	0.34	871.60

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	280.8700	1849.5516	0.152	0.87931
Year	-0.1250	0.9163	-0.136	0.89146
StateFLORIDA	2.9301	3.4411	0.852	0.39456
StateOREGON	11.0538	9.6052	1.151	0.24991
StateWASHINGTON	3.2225	7.8243	0.412	0.68047
temp23 HERBICIDE	8.5870	5.6532	1.519	0.12889
temp23 INSECTICIDE	10.9387	3.6665	2.983	0.00288 **
temp23 OTHER	15.3162	4.9430	3.099	0.00197 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

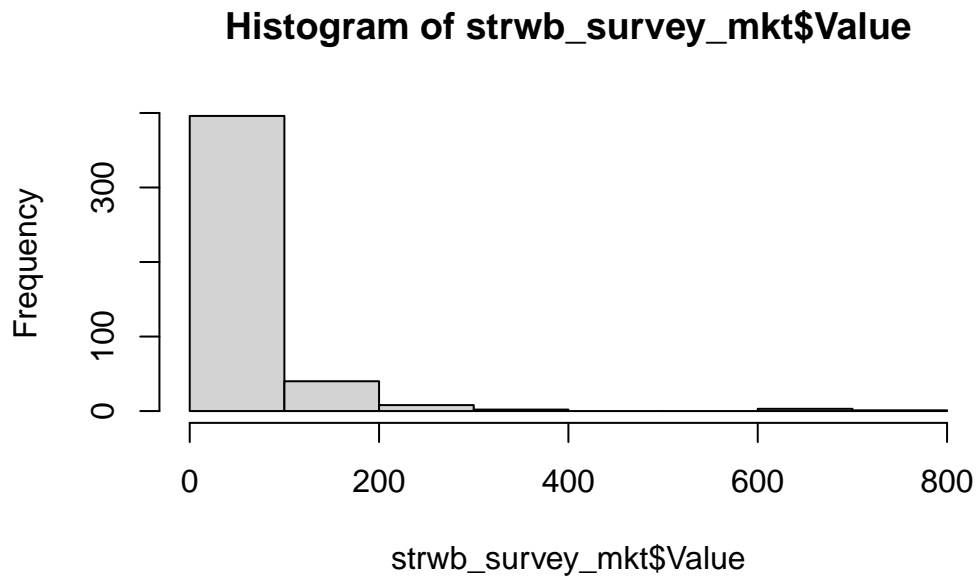
Residual standard error: 81.24 on 2625 degrees of freedom

(246)

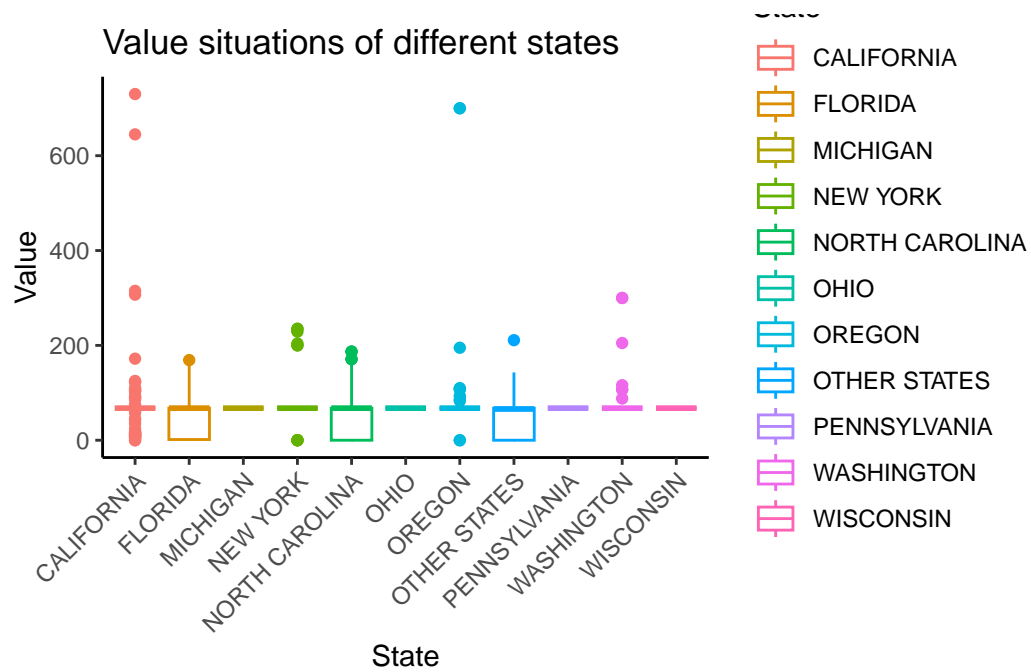
Multiple R-squared: 0.005629, Adjusted R-squared: 0.002977

F-statistic: 2.123 on 7 and 2625 DF, p-value: 0.03821

```
hist(strwb_survey_mkt$Value)
```



```
ggplot(data = strwb_survey_mkt, aes(x=State, y=Value, color = State)) +  
  geom_boxplot() +  
  theme_classic() +  
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +  
  labs(  
    title = "Value situations of different states"  
  )
```



```
ggplot(data = strwb_survey_mkt, aes(x=Year, y=Value, color = State)) +
  geom_point() +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(
    title = "Value situations of different states in different years"
  )
```

