Berries Project

Jiachen Feng

```r
library(knitr)
library(tidyverse)
```

```
## -- Attaching packages -------------------------------------------------------------------------------

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ---------------------------------------------------------------------------------- tid
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(magrittr)
```

```
##
## Attaching package: 'magrittr'

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

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

```r
library(kableExtra)
```

```
## Warning: package 'kableExtra' was built under R version 4.0.3

##
## Attaching package: 'kableExtra'

## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```r
## read the data

ag_data <- read_csv("berries.csv", col_names = TRUE)
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   Year = col_double(),
##   `Week Ending` = col_logical(),
##   `Ag District` = col_logical(),
##   `Ag District Code` = col_logical(),
```

```
##   County = col_logical(),
##   `County ANSI` = col_logical(),
##   `Zip Code` = col_logical(),
##   Region = col_logical(),
##   Watershed = col_logical(),
##   `CV (%)` = col_logical()
## )
```

```
## See spec(...) for full column specifications.
```

Get R environment ready and read the "Berries" data set.

```r
## look at number of unique values in each column
ag_data %>% summarize_all(n_distinct) -> aa


## make a list of the columns with only one unique value
bb <- which(aa[1,]==1)

## list the 1-unique valu column names
colnames(ag_data)[bb]
```

```
##  [1] "Program"          "Week Ending"      "Geo Level"        "Ag District"
##  [5] "Ag District Code" "County"           "County ANSI"      "Zip Code"
##  [9] "Region"           "watershed_code"   "Watershed"        "CV (%)"
```

```r
## list the 1-unique single values.
## Consider if they should be used for labels

single_values <- ag_data[1,bb]


## remove the 1-unique columns from the dataset
ag_data %<>% select(-all_of(bb))

## look at number of unique values in each column
ag_data %>% summarize_all(n_distinct) -> aa


## make a list of the columns with only one unique value
bb <- which(aa[1,]==1)

## list the 1-unique valu column names
colnames(ag_data)[bb]
```

```
## character(0)
```

```r
## list the 1-unique single values.
## Consider if they should be used for labels

single_values <- ag_data[1,bb]


## remove the 1-unique columns from the dataset
ag_data %<>% select(-all_of(bb))

## Make a table of the number of unique values in each column.
```

```r
aa %<>% select(-all_of(bb))

## State name and the State ANSI code are (sort of) redundant


ag_data %<>% select(-4)
aa %<>% select(-4)


ag_data$Year %>%  unique()
```

```
## [1] 2019 2018 2017 2016 2015
```

```r
## [1] 2019 2018 2017 2016 2015

ag_data$Period %>% unique()
```

```
## [1] "MARKETING YEAR"      "YEAR"                 "YEAR - AUG FORECAST"
```

```r
## "MARKETING YEAR"      "YEAR"                 "YEAR - AUG FORECAST"

## Year:
## Generally refers to calendar year.
## For Prices Received data, refers to
##an unweighted average (by month) for the calendar year.

## Marketing year:
## Definition varies by commodity;
## see Agricultural Prices publications
## for definitions by commodity.
## For Prices Received data, refers to a
## weighted average for the marketing year.
```

This process identifies columns without any data or with a single repeated Values, then remove those columns from the initial dataset.

```r
### let's focus on: period = "Year" and Commodity = "BLUEBERRIES"

## blueberry data
ag_data_bb <- ag_data %>% filter((Commodity=="BLUEBERRIES") & (Period=="YEAR"))

ag_data_bb %<>% separate(`Data Item`, c("berry", "type", "data_item", "unit"), ",")
```

```
## Warning: Expected 4 pieces. Missing pieces filled with `NA` in 1537 rows [1, 2,
## 3, 11, 12, 13, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, ...].
```

```r
ag_data_bb %<>% select(-c(Period,Commodity,berry))

kable(head(ag_data_bb)) %>% kable_styling(font_size=12)
```

Year	State	type	data_item	unit
2019	CALIFORNIA	TAME - ACRES HARVESTED	NA	NA
2019	CALIFORNIA	TAME - PRODUCTION	MEASURED IN LB	NA
2019	CALIFORNIA	TAME - YIELD	MEASURED IN LB / ACRE	NA
2019	CALIFORNIA	TAME	FRESH MARKET - PRODUCTION	MEAS
2019	CALIFORNIA	TAME	FRESH MARKET - PRODUCTION	MEAS
2019	CALIFORNIA	TAME	NOT SOLD - PRODUCTION	MEAS

```r
##########################################################

### Then focus on: period = "Year" and Commodity = "strawberries"

## Strawberry data
ag_data_sb <- ag_data %>% filter((Commodity=="STRAWBERRIES") & (Period=="YEAR"))

ag_data_sb %<>% separate(`Data Item`, c("berry", "type", "data_item", "unit"), ",")
```

```
## Warning: Expected 4 pieces. Missing pieces filled with `NA` in 890 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```r
ag_data_sb %<>% select(-c(Period,Commodity,berry))

kable(head(ag_data_sb)) %>% kable_styling(font_size=12)
```

Year	State	type	data_item	unit	Domain
2019	CALIFORNIA	NA	NA	NA	TOTAL
2019	CALIFORNIA	NA	NA	NA	TOTAL
2019	CALIFORNIA	MEASURED IN $	NA	NA	TOTAL
2019	CALIFORNIA	MEASURED IN CWT	NA	NA	TOTAL
2019	CALIFORNIA	MEASURED IN CWT / ACRE	NA	NA	TOTAL
2019	CALIFORNIA	BEARING - APPLICATIONS	MEASURED IN LB	NA	CHEMICAL, FU

```r
##########################################################

### Also,period = "Year" and Commodity = "raspberries"

## Raspberry data
ag_data_rb <- ag_data %>% filter((Commodity=="RASPBERRIES") & (Period=="YEAR"))

ag_data_rb %<>% separate(`Data Item`, c("berry", "type", "data_item", "unit"), ",")
```

```
## Warning: Expected 4 pieces. Missing pieces filled with `NA` in 539 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```r
ag_data_rb %<>% select(-c(Period,Commodity,berry))

kable(head(ag_data_rb)) %>% kable_styling(font_size=12)
```

Year	State	type	data_item	unit	Domain
2019	CALIFORNIA	NA	NA	NA	TOTAL
2019	CALIFORNIA	MEASURED IN LB	NA	NA	TOTAL
2019	CALIFORNIA	MEASURED IN LB / ACRE	NA	NA	TOTAL
2019	CALIFORNIA	BEARING - APPLICATIONS	MEASURED IN LB	NA	CHEMICAL, FU
2019	CALIFORNIA	BEARING - APPLICATIONS	MEASURED IN LB	NA	CHEMICAL, FU
2019	CALIFORNIA	BEARING - APPLICATIONS	MEASURED IN LB	NA	CHEMICAL, FU

This process divides the initial data set into three subsets to facilitate our classification research.

The first part of this project mentioned above is about data cleaning. At first, I want to thank professor Wright for his help in completing this part of the project. Data cleaning, or data preparation is an essential part of statistical analysis. Based on personal experience of the data cleaning process, I find that it is more time-consuming than the statistical analysis itself. But this process is indispensable, for the reason that it ensures data can be deemed technically correct.

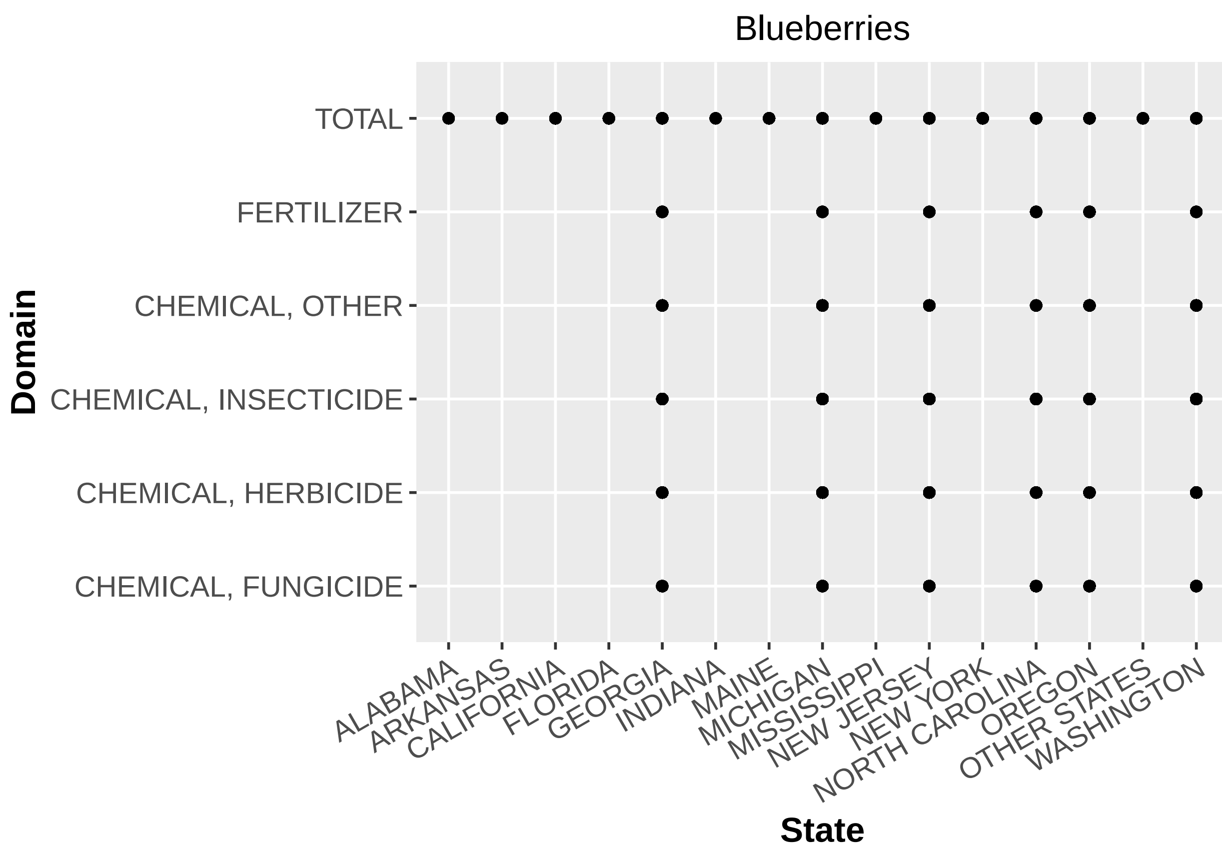
To sum up, I believe data cleaning is a meaningful part of the whole statistical analysis process.

```
##EDA

summary(ag_data_bb)
```

```
##       Year          State               type             data_item        
##  Min.   :2015   Length:7419        Length:7419        Length:7419       
##  1st Qu.:2015   Class :character   Class :character   Class :character  
##  Median :2017   Mode  :character   Mode  :character   Mode  :character  
##  Mean   :2017                                                           
##  3rd Qu.:2019                                                           
##  Max.   :2019                                                           
##      unit              Domain          Domain Category       Value          
##  Length:7419        Length:7419        Length:7419        Length:7419       
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
## 
## 
## 
```

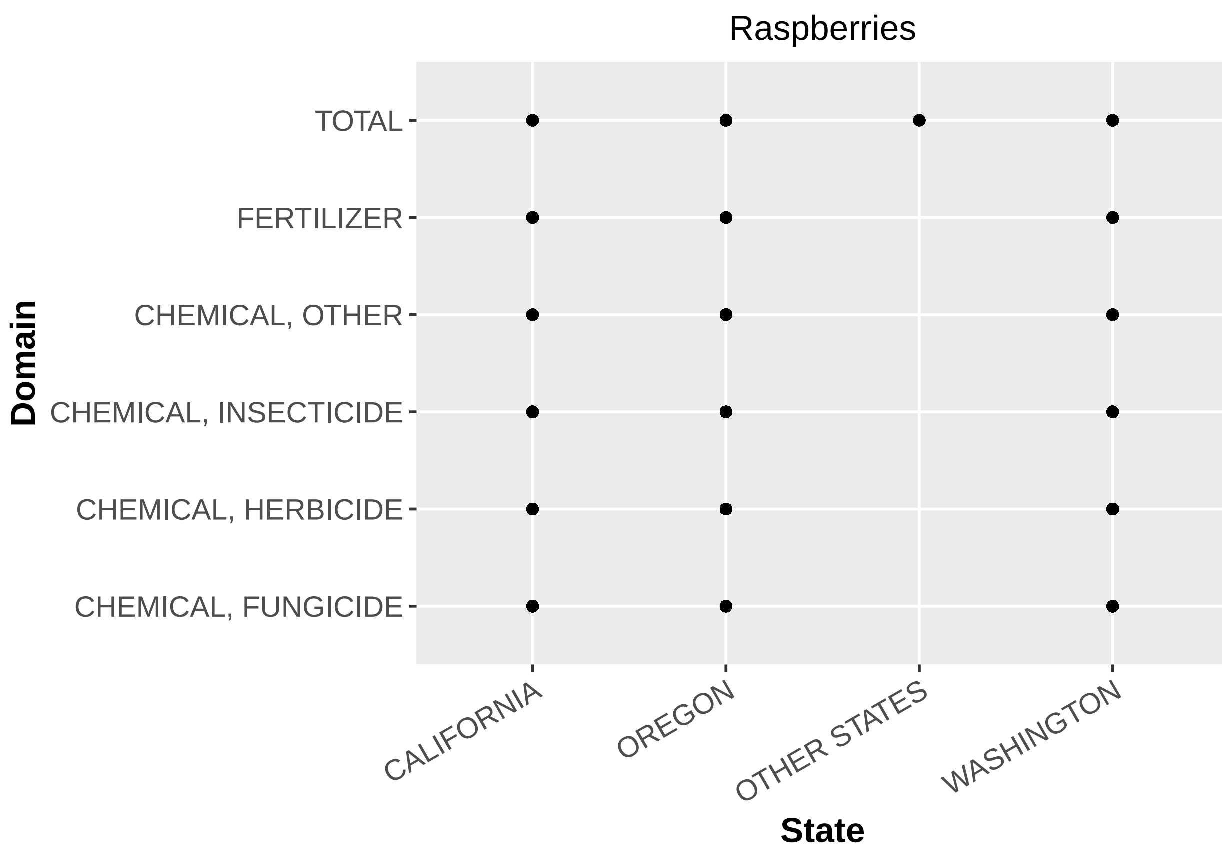
```
bbplot1 <- ggplot(ag_data_bb, aes(x = State, y = Domain))
bbplot1 <- bbplot1 + geom_point() +
  theme(axis.text.x = element_text(angle = 30, hjust = 1),
        axis.text = element_text(size = 11),
        axis.title = element_text(size = 13, face = "bold")) +
  labs(x = "State",y="Domain",title = "Blueberries")+
  theme(plot.title = element_text(hjust = 0.5))
bbplot1
```




```
summary(ag_data_rb)
```

```
##       Year         State               type             data_item
##  Min.   :2015   Length:2068        Length:2068        Length:2068
##  1st Qu.:2015   Class :character   Class :character   Class :character
##  Median :2017   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2017
##  3rd Qu.:2019
##  Max.   :2019
##      unit              Domain          Domain Category      Value
##  Length:2068        Length:2068        Length:2068        Length:2068
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

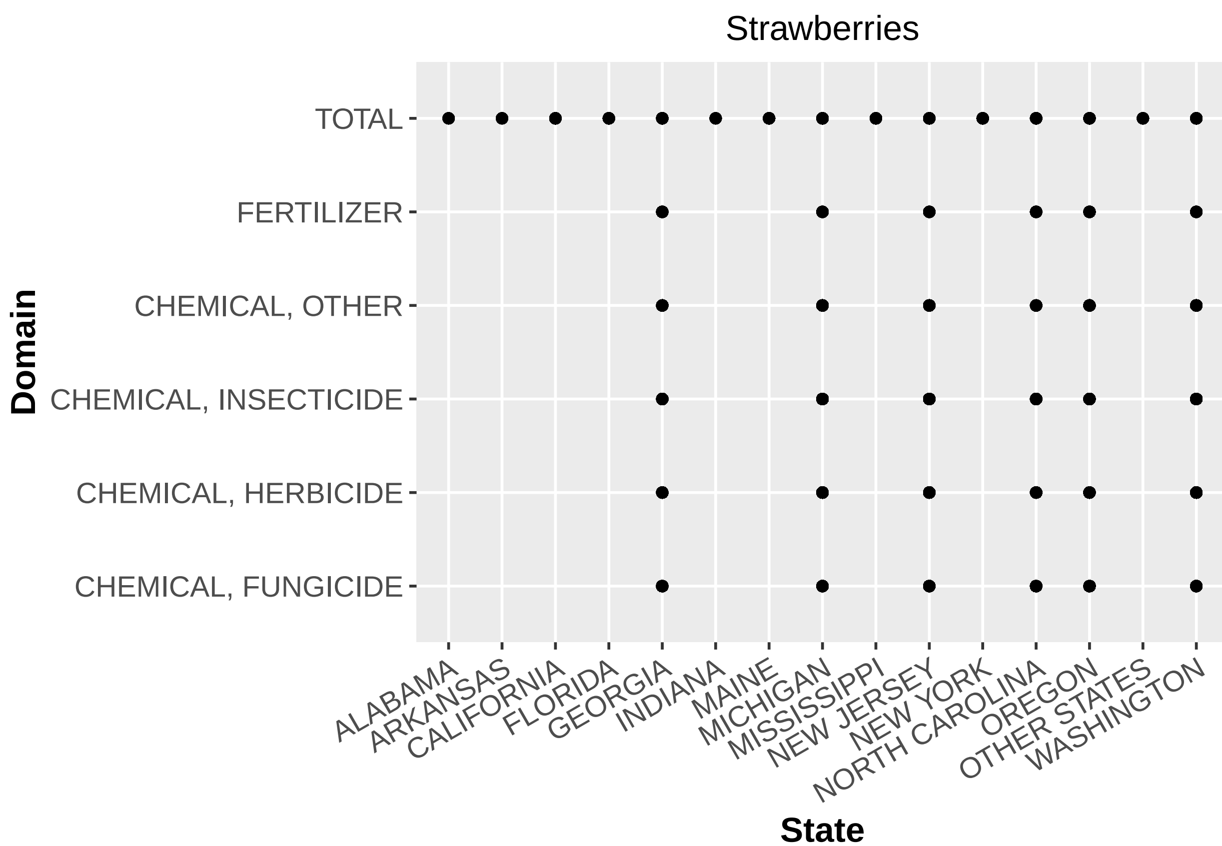
```
rbplot1 <- ggplot(ag_data_rb, aes(x = State, y = Domain))
rbplot1 <- rbplot1 + geom_point() +
  theme(axis.text.x = element_text(angle = 30, hjust = 1),
        axis.text = element_text(size = 11),
        axis.title = element_text(size = 13, face = "bold")) +
  labs(x = "State",y="Domain",title = "Raspberries")+
  theme(plot.title = element_text(hjust = 0.5))
rbplot1
```




```
summary(ag_data_sb)
```

```
##       Year         State               type            data_item
##  Min.   :2015   Length:3220        Length:3220        Length:3220
##  1st Qu.:2016   Class :character   Class :character   Class :character
##  Median :2018   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2017
##  3rd Qu.:2019
##  Max.   :2019
##      unit              Domain          Domain Category       Value
##  Length:3220        Length:3220        Length:3220        Length:3220
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

```
sbplot1 <- ggplot(ag_data_bb, aes(x = State, y = Domain))
sbplot1 <- sbplot1 + geom_point() +
  theme(axis.text.x = element_text(angle = 30, hjust = 1),
        axis.text = element_text(size = 11),
        axis.title = element_text(size = 13, face = "bold")) +
  labs(x = "State",y="Domain",title = "Strawberries")+
  theme(plot.title = element_text(hjust = 0.5))
sbplot1
```



The second part of this project is EDA. We can easily learn the basic information about the three datasets which are generated from the data cleaning part. Also, We can find from the plots which planting methods different states tend to adopt. Different varieties of berries may adopt a different domain category in planting.

Reference:

[1] Edwin de Jonge, Mark van der Loo. An introduction in data cleaning with R, 2013.

[2] Hadley Wickham and Garrett Grolemund. R for Data Science. Import, Tidy, Transform, Visualize and Model Data, 2016.