Strawberries Assignment

Jin Wen Lin

```
library(knitr)
library(kableExtra)
library(tidyverse)
library(stringr)
library(tidyverse)
```

First, read the data and see the overall summary.

```
strawberry <- read_csv("strawberries25_v3.csv")
```

```
Rows: 12669 Columns: 21
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (15): Program, Period, Geo Level, State, State ANSI, Ag District, County...
dbl  (2): Year, Ag District Code
lgl  (4): Week Ending, Zip Code, Region, Watershed

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse(strawberry)
```

```
Rows: 12,669
Columns: 21
$ Program            <chr> "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CENSUS", "~
$ Year               <dbl> 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 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> "COUNTY", "COUNTY", "COUNTY", "COUNTY", "COUNTY", "~
$ State              <chr> "ALABAMA", "ALABAMA", "ALABAMA", "ALABAMA", "ALABAM~
```

```
$ `State ANSI`       <chr> "01", "01", "01", "01", "01", "01", "01", "01", "01~
$ `Ag District`      <chr> "BLACK BELT", "BLACK BELT", "BLACK BELT", "BLACK BE~
$ `Ag District Code` <dbl> 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40, 40,~
$ County             <chr> "BULLOCK", "BULLOCK", "BULLOCK", "BULLOCK", "BULLOC~
$ `County ANSI`      <chr> "011", "011", "011", "011", "011", "011", "101", "1~
$ `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 - ACRES BEARING", "STRAWBERRIES - ACR~
$ Domain             <chr> "TOTAL", "TOTAL", "TOTAL", "TOTAL", "TOTAL", "TOTAL~
$ `Domain Category`  <chr> "NOT SPECIFIED", "NOT SPECIFIED", "NOT SPECIFIED", ~
$ Value              <chr> "(D)", "3", "(D)", "1", "6", "5", "(D)", "(D)", "2"~
$ `CV (%)`           <chr> "(D)", "15.7", "(D)", "(L)", "52.7", "47.6", "(D)",~
```

```r
#|label: function def - drop 1-item columns (code from lecture)

drop_one_value_col <- function(df){   ## takes whole dataframe
drop <- NULL

## test each column for a single value
for(i in 1:dim(df)[2]){
if((df |> distinct(df[,i]) |> count()) == 1){
drop = c(drop, i)
} }

## report the result -- names of columns dropped
## consider using the column content for labels
## or headers

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

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


## drop the columns with NAs
```

```
strawberry <- drop_one_value_col(strawberry)
```

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

Separate the data item column into columns of strawberries, Measure, and Bearing type.

```
# strawberry census
strawberry_cens<- strawberry %>% filter(Program == "CENSUS")
data_item <- strawberry_cens %>% distinct(`Data Item`)
data_item
```

```
# A tibble: 18 x 1
   `Data Item`
   <chr>
 1 STRAWBERRIES - ACRES BEARING
 2 STRAWBERRIES - ACRES GROWN
 3 STRAWBERRIES - ACRES NON-BEARING
 4 STRAWBERRIES - OPERATIONS WITH AREA BEARING
 5 STRAWBERRIES - OPERATIONS WITH AREA GROWN
 6 STRAWBERRIES - OPERATIONS WITH AREA NON-BEARING
 7 STRAWBERRIES, ORGANIC - ACRES HARVESTED
 8 STRAWBERRIES, ORGANIC - OPERATIONS WITH AREA HARVESTED
 9 STRAWBERRIES, ORGANIC - OPERATIONS WITH SALES
10 STRAWBERRIES, ORGANIC - PRODUCTION, MEASURED IN CWT
11 STRAWBERRIES, ORGANIC - SALES, MEASURED IN $
12 STRAWBERRIES, ORGANIC - SALES, MEASURED IN CWT
13 STRAWBERRIES, ORGANIC, FRESH MARKET - OPERATIONS WITH SALES
14 STRAWBERRIES, ORGANIC, FRESH MARKET - SALES, MEASURED IN $
15 STRAWBERRIES, ORGANIC, FRESH MARKET - SALES, MEASURED IN CWT
16 STRAWBERRIES, ORGANIC, PROCESSING - OPERATIONS WITH SALES
17 STRAWBERRIES, ORGANIC, PROCESSING - SALES, MEASURED IN $
18 STRAWBERRIES, ORGANIC, PROCESSING - SALES, MEASURED IN CWT
```

```
## Separate the Data Item column
strawberry_cens <- strawberry_cens %>%
  separate_wider_delim(  cols = `Data Item`,
                         delim = " - ",
                         names = c("strawberries",
```

```
                                "Category"),
                        too_many = "error",
                        too_few = "align_start"
                      )

strawberry_cens <- strawberry_cens %>%
  separate_wider_delim(  cols = `Category`,
                        delim = " ",
                        names = c("Measure",
                                "Bearing_type"),
                        too_many = "merge",
                        too_few = "align_start"
                      )
# remove commas in the Measure Column
strawberry_cens <- strawberry_cens %>%
  mutate(Measure = gsub("\\,", "", Measure))

# organic census data
organic_cens <- strawberry_cens %>% filter(str_detect(strawberries, "ORGANIC")==TRUE)

# non-organic census data
non_organic_cens <- strawberry_cens %>% filter(str_detect(strawberries, "ORGANIC")==FALSE)
```

Separate the Domain Category column into three new columns with chemical name, chemical type, and chemical code.

```
#|label: Strawberry Survey Data

strawberry_survey <- strawberry %>% filter(Program == "SURVEY")
strawberry_survey %>% distinct(strawberry_survey$`Domain Category`)
```

```
# A tibble: 183 x 1
   `strawberry_survey$\`Domain Category\``
   <chr>
 1 NOT SPECIFIED
 2 CHEMICAL, FUNGICIDE: (OXATHIAPIPROLIN = 128111)
 3 CHEMICAL, INSECTICIDE: (CYCLANILIPROLE = 26202)
 4 CHEMICAL, INSECTICIDE: (PERMETHRIN = 109701)
 5 CHEMICAL, OTHER: (ISARIA FUMOSOROSEA STRAIN FE 9901 = 115003)
 6 CHEMICAL, FUNGICIDE: (AZOXYSTROBIN = 128810)
 7 CHEMICAL, FUNGICIDE: (BACILLUS AMYLOLIQUEFACIENS STRAIN D747 = 16482)
```

```
 8 CHEMICAL, FUNGICIDE: (BACILLUS SUBTILIS = 6479)
 9 CHEMICAL, FUNGICIDE: (BLAD = 30006)
10 CHEMICAL, FUNGICIDE: (BORAX DECAHYDRATE = 11102)
# i 173 more rows
```

```
# remove CHEMICAL in the Domain Category
strawberry_survey <- strawberry_survey %>%
  mutate(`Domain Category` = gsub("CHEMICAL, ", "", `Domain Category`))
# separate them into chemical name, chemical type, and chemical code
strawberry_survey <- strawberry_survey %>%
  separate_wider_delim(  cols = `Domain Category`,
                         delim = ": ",
                         names = c("Chemical Name",
                                   "Others"),
                         too_many = "error",
                         too_few = "align_start"
                       )

strawberry_survey <- strawberry_survey %>%
  separate_wider_delim(  cols = Others,
                         delim = " = ",
                         names = c("Chemical Type",
                                   "Chemical Code"),
                         too_many = "error",
                         too_few = "align_start"
                       )
# remove brackets
strawberry_survey <- strawberry_survey %>%
  mutate(`Chemical Type` = gsub("\\)", "", `Chemical Type`))

strawberry_survey <- strawberry_survey %>%
  mutate(`Chemical Type` = gsub("\\(", "", `Chemical Type`))

strawberry_survey <- strawberry_survey %>%
  mutate(`Chemical Code` = gsub("\\)", "", `Chemical Code`))
```

```
#|label: Check the Value Column

# organic census data Value
value_1 <- organic_cens %>% distinct(Value)
value_1
```

```
# A tibble: 182 x 1
   Value
   <chr>
 1 5,301
 2 546
 3 1,495,299
 4 335,964,420
 5 1,494,673
 6 540
 7 (D)
 8 1,483,234
 9 18
10 11,440
# i 172 more rows
```

```
# change Value into numbers
organic_cens$Value <- as.numeric(gsub(",", "", organic_cens$Value))
```

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

```
# non-organic census data Value
value_2 <- non_organic_cens %>% distinct(Value)
value_2
```

```
# A tibble: 316 x 1
   Value
   <chr>
 1 (D)
 2 3
 3 1
 4 6
 5 5
 6 2
 7 4
 8 8
 9 9
10 10
# i 306 more rows
```

```
# change Value into numbers
non_organic_cens$Value <- as.numeric(gsub(",", "", non_organic_cens$Value))
```

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

```
# survey data
# change Value into numbers
strawberry_survey$Value <- as.numeric(gsub(",", "", strawberry_survey$Value))
```

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

While turning the value column into numbers or doubles instead of characters, there are some values automatically turned into NA such as notations like (D) and (Z) etc. The different notations do have different meanings behind them but now they are all in NAs.

```
# write the survey data into csv
write.csv(strawberry_survey, "strawberry_survey.csv", row.names = FALSE)
```