Strawberry

Liwen Yin

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

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter()     masks stats::filter()
x dplyr::group_rows() masks kableExtra::group_rows()
x dplyr::lag()        masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
library(tidyr)
library(dplyr)
```

You can add options to executable code like this

```
#read data
strawberry <- read.csv("strawberries25_v3.csv", header = T)
#check data
glimpse(strawberry)
```

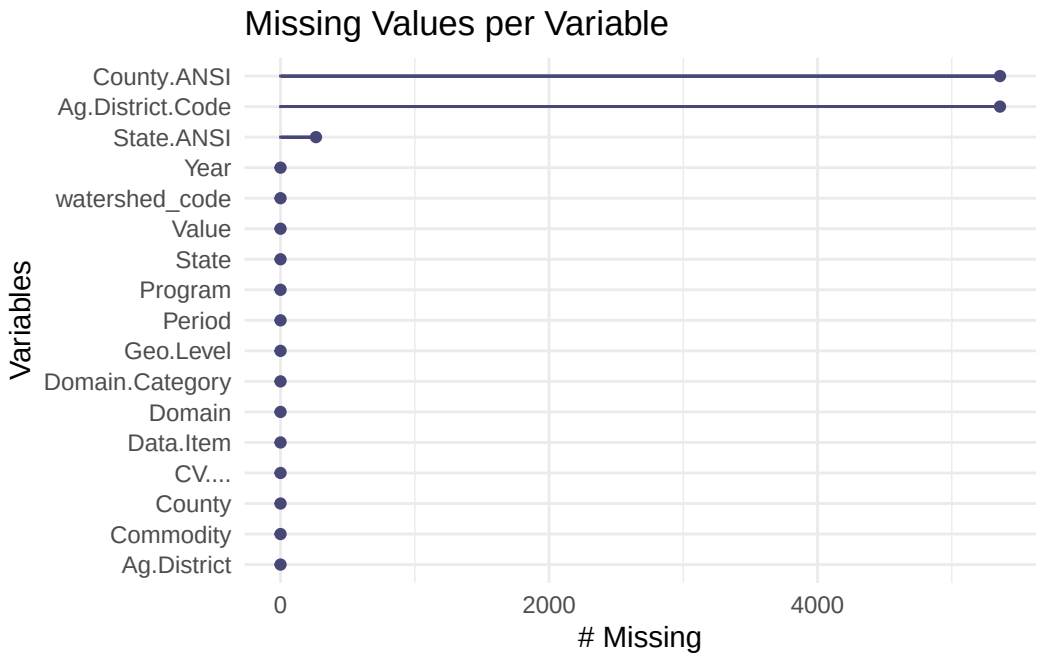
```
Rows: 12,669
Columns: 21
$ Program          <chr> "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CENSUS", "CE~
$ Year             <int> 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022,~
```

```
$ 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)", "~
```

##Data cleaning

```r
# Remove columns with excessive missing values
strawberry <- strawberry %>%
  filter(Geo.Level== "NATIONAL" | Geo.Level== "STATE")
strawberry <- strawberry %>%
  select(-Week.Ending, -Zip.Code, -Watershed, -Region)
# Fill missing values with 'Unknown'
strawberry <- strawberry %>%
  mutate(Ag.District = replace_na(Ag.District, "Unknown"),
         County = replace_na(County, "Unknown"))

# Convert 'Value' column to numeric and handle non-numeric entries
library(ggplot2)
library(naniar)
gg_miss_var(strawberry) +
  labs(title = "Missing Values per Variable")
```




```
strawberry$Value <- as.numeric(str_replace_all(strawberry$Value, ",",""))
```

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

```
strawberry$CV.... <- as.numeric(strawberry$CV....)
```

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

```
# Check the structure and summary of cleaned data
summary(strawberry)
```

```
   Program               Year         Period            Geo.Level
 Length:5359        Min.   :2018   Length:5359        Length:5359
 Class :character   1st Qu.:2019   Class :character   Class :character
 Mode  :character   Median :2021   Mode  :character   Mode  :character
                    Mean   :2020
                    3rd Qu.:2022
                    Max.   :2024

    State             State.ANSI     Ag.District        Ag.District.Code
```

```
 Length:5359        Min.   : 1.00   Length:5359        Min.   : NA
 Class :character   1st Qu.: 6.00   Class :character   1st Qu.: NA
 Mode  :character   Median :12.00   Mode  :character   Median : NA
                    Mean   :14.73                      Mean   :NaN
                    3rd Qu.:12.00                      3rd Qu.: NA
                    Max.   :56.00                      Max.   : NA
                    NA's   :264                        NA's   :5359
    County           County.ANSI   watershed_code  Commodity
 Length:5359        Min.   : NA    Min.   :0      Length:5359
 Class :character   1st Qu.: NA    1st Qu.:0      Class :character
 Mode  :character   Median : NA    Median :0      Mode  :character
                    Mean   :NaN    Mean   :0
                    3rd Qu.: NA    3rd Qu.:0
                    Max.   : NA    Max.   :0
                    NA's   :5359
  Data.Item            Domain          Domain.Category        Value
 Length:5359        Length:5359        Length:5359        Min.   :0.000e+00
 Class :character   Class :character   Class :character   1st Qu.:2.000e+00
 Mode  :character   Mode  :character   Mode  :character   Median :3.100e+01
                                                          Mean   :2.877e+07
                                                          3rd Qu.:5.460e+02
                                                          Max.   :3.584e+09
                                                          NA's   :2266
     CV....
 Min.   : 2.70
 1st Qu.:19.57
 Median :34.95
 Mean   :40.01
 3rd Qu.:56.20
 Max.   :99.80
 NA's   :4355
```

##census vs survey

```
options(scipen = 999)
straw_cen <- strawberry %>% filter(Program=="CENSUS")
straw_sur <- strawberry %>% filter(Program=="SURVEY")
straw_cen <- straw_cen %>%
  filter(!grepl("COUNTY", Geo.Level))
```

##Census data

```
#non_organic
straw_organic <- straw_cen %>%
  filter(grepl("ORGANIC", Data.Item))
straw_non_organic <- straw_cen %>%
  filter(!grepl("ORGANIC", Data.Item))
straw_organic <- straw_organic %>%
  separate_wider_delim(  cols = Data.Item,
                         delim = ", ",
                         names = c("strawberries",
                                 "ORGANIC",
                                 "organic_detail"),
                         too_many = "merge",
                         too_few = "align_start"
                       )
straw_non_organic <- straw_non_organic %>%
  separate_wider_delim(  cols = `Data.Item`,
                         delim = "-",
                         names = c("Fruit",
                                 "Category"),
                         too_many = "merge",
                         too_few = "align_start"
                       )
straw_non_organic$Category <- straw_non_organic$Category %>%
  str_remove_all("OPERATIONS WITH ")
```

```
## remove AREA GROWN and parens
## change NOT SPECIFIEC TO TOTAL
straw_cen <- straw_cen |> rename(size_bracket = `Domain.Category`)

straw_cen$size_bracket <- str_replace(straw_cen$size_bracket, "NOT SPECIFIED", "TOTAL")

straw_cen$size_bracket <- str_replace(straw_cen$size_bracket, "AREA GROWN: ", "")
```

##Survey data

```
straw_sur1 <- straw_sur |>  separate_wider_delim(cols = `Data.Item`,
                                                 delim = ", ",
                                                 names = c("straw",
                                                           "mkt",
                                                           "measure",
                                                           "other"
                                                           ),
```

```
                                                  too_many = "merge",
                                                  too_few = "align_start")


straw_sur2 <- straw_sur1 |> separate_wider_delim(cols = "straw",
                                                  delim = " - ",
                                                  names = c("straw",
                                                            "more"),
                                                  too_many = "merge",
                                                  too_few = "align_start"
                                                  )

rm(straw_sur, straw_sur1)
shift_loc <- function(df, col_name, dat_name, num_col, num_shift){
 # browser()
  col_num = which(colnames(df) == col_name)
  row_num = which(df[,col_num] == dat_name)  ## calcs a vector of rows

  for(k in 1:length(row_num)){
  d = rep(0,num_col) ## storage for items to be moved
  for(i in 1:num_col){
    d[i] = df[row_num[k], col_num + i - 1]
  }
  for(i in 1:num_col){
    ra = row_num[k]
    cb = col_num + i - 1
    df[ra, cb] <-  NA
  }
  for(j in 1:num_col){
    rc = row_num[k]
    cd = col_num + j - 1 + num_shift
    df[rc, cd] = d[j]
  }
  }
 # sprintf("Rows adjusted:")
  # print("%d",row_num)
  return(df)
}
```

```
straw_sur2 <- straw_sur2 |> shift_loc("more", "PRICE RECEIVED", 2, 1 )


straw_sur2 <- straw_sur2 |> shift_loc("more", "ACRES HARVESTED", 1, 1 )

straw_sur2 <- straw_sur2 |> shift_loc("more", "ACRES PLANTED", 1, 1 )

straw_sur2 <- straw_sur2 |> shift_loc("more", "PRODUCTION", 2, 1 )

straw_sur2 <- straw_sur2 |> shift_loc("more", "YIELD", 2, 1 )

straw_sur2 <- straw_sur2 |> shift_loc("more", "APPLICATIONS", 3, 1 )

straw_sur2 <- straw_sur2 |> shift_loc("more", "TREATED", 3, 1 )


# split chemical data
straw_sur2 <- straw_sur2 %>%
  extract(
    col = Domain.Category,
    into = c("Type", "Name", "Chemical.Code"),
    regex = ".*?, (\\w+): \\((\\w+) = (\\d+)\\)",
    remove = TRUE
  )
straw_sur2 <- straw_sur2 |>  separate_wider_delim(cols = Domain,
                                                  delim = ", ",
                                                  names = c("col1",
                                                            "col2"),
                                                  too_many = "merge",
                                                  too_few = "align_start")
```

##look at totals

```
survey_d_total <- straw_sur2 |>  filter(col1 == "TOTAL")
survey_d_chem <- straw_sur2 |>  filter(col1 == "CHEMICAL")
survey_d_fert <- straw_sur2 |>  filter(col1 == "FERTILIZER")
### align terms
survey_d_total <- survey_d_total |> shift_loc("measure", "MEASURED IN $ / CWT", 1, 1 )
survey_d_total <- survey_d_total |> shift_loc("measure", "MEASURED IN $", 1, 1 )
survey_d_total <- survey_d_total |> shift_loc("measure", "MEASURED IN CWT", 1, 1 )

survey_d_total <- survey_d_total |> shift_loc("measure", "MEASURED IN TONS", 1, 1 )
```

```
survey_d_total <- survey_d_total |> shift_loc("measure", "MEASURED IN CWT / ACRE", 1, 1 )

survey_d_total <- survey_d_total |> shift_loc("measure", "MEASURED IN TONS / ACRE", 1, 1 )


#### split the mkt column


survey_d_total <- survey_d_total |>
  separate_wider_delim(cols = mkt,
                       delim = " - ",
                       names = c("col3",
                                 "col4"),
                    too_many = "merge",
                     too_few = "align_start")
```

#Group by state

```
straw_sur_calnif <- straw_sur2 %>%
  filter(straw_sur2$State=="CALIFORNIA")
straw_sur_florida <- straw_sur2 %>%
  filter(straw_sur2$State=="FLORIDA")
straw_nono_calnif <- straw_non_organic %>%
  filter(straw_non_organic$State=="CALIFORNIA")
straw_nono_florida <- straw_non_organic %>%
  filter(straw_non_organic$State=="FLORIDA")
```