StrawberryDataCleaning

1 .Reading data and ditch the counties.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'

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

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.5
## v ggplot2   3.5.1     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()          masks stats::filter()
## x kableExtra::group_rows() masks dplyr::group_rows()
## x dplyr::lag()             masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
strawberry <- read.csv("strawberries25_v3.csv")
head(strawberry)
```

```
##   Program Year Period Week.Ending Geo.Level   State State.ANSI Ag.District
## 1  CENSUS 2022   YEAR          NA    COUNTY ALABAMA          1  BLACK BELT
## 2  CENSUS 2022   YEAR          NA    COUNTY ALABAMA          1  BLACK BELT
## 3  CENSUS 2022   YEAR          NA    COUNTY ALABAMA          1  BLACK BELT
## 4  CENSUS 2022   YEAR          NA    COUNTY ALABAMA          1  BLACK BELT
## 5  CENSUS 2022   YEAR          NA    COUNTY ALABAMA          1  BLACK BELT
## 6  CENSUS 2022   YEAR          NA    COUNTY ALABAMA          1  BLACK BELT
##   Ag.District.Code  County County.ANSI Zip.Code Region watershed_code Watershed
## 1               40 BULLOCK          11       NA     NA              0        NA
## 2               40 BULLOCK          11       NA     NA              0        NA
## 3               40 BULLOCK          11       NA     NA              0        NA
## 4               40 BULLOCK          11       NA     NA              0        NA
## 5               40 BULLOCK          11       NA     NA              0        NA
## 6               40 BULLOCK          11       NA     NA              0        NA
##      Commodity                                       Data.Item Domain
## 1 STRAWBERRIES                    STRAWBERRIES - ACRES BEARING  TOTAL
## 2 STRAWBERRIES                      STRAWBERRIES - ACRES GROWN  TOTAL
## 3 STRAWBERRIES                STRAWBERRIES - ACRES NON-BEARING  TOTAL
## 4 STRAWBERRIES     STRAWBERRIES - OPERATIONS WITH AREA BEARING  TOTAL
## 5 STRAWBERRIES       STRAWBERRIES - OPERATIONS WITH AREA GROWN  TOTAL
## 6 STRAWBERRIES STRAWBERRIES - OPERATIONS WITH AREA NON-BEARING  TOTAL
##   Domain.Category Value CV....
## 1   NOT SPECIFIED   (D)    (D)
## 2   NOT SPECIFIED     3   15.7
## 3   NOT SPECIFIED   (D)    (D)
## 4   NOT SPECIFIED     1    (L)
## 5   NOT SPECIFIED     6   52.7
## 6   NOT SPECIFIED     5   47.6
```

```
colnames(strawberry)
```

```
##  [1] "Program"          "Year"             "Period"           "Week.Ending"
##  [5] "Geo.Level"        "State"            "State.ANSI"       "Ag.District"
##  [9] "Ag.District.Code" "County"           "County.ANSI"      "Zip.Code"
## [13] "Region"           "watershed_code"   "Watershed"        "Commodity"
## [17] "Data.Item"        "Domain"           "Domain.Category"  "Value"
## [21] "CV...."
```

```
strawberry <- strawberry |>
  filter(`Geo.Level`== "NATIONAL" | `Geo.Level`== "STATE")
```

2. Here we will drop the column with most single value, since the columns are either all missing data, or cannnot be analyzed correctly.

```
drop_one_value_col <- function(df){
  df_id <- ensym(df)
  msg = paste("Looking for single value columns in data frame: ", as.character(df_id))
  print(msg)
  drop <- NULL
  val <- NULL

  for(i in 1:ncol(df)){
    if(length(unique(df[[i]])) == 1){
      drop <- c(drop, i)
      val <- c(val, df[1, i])
    }
  }

  if(is.null(drop)){
    print("No columns dropped")
    return(df)
  } else {
    print("Columns dropped:")
    print(unlist(val))
    df <- df[, -drop, drop = FALSE]
    return(df)
```

```r
  }
}

strawberry <- strawberry |> drop_one_value_col()
```

```
## [1] "Looking for single value columns in data frame:  strawberry"
## [1] "Columns dropped:"
##  [1] NA             ""             NA             ""             NA            
##  [6] NA             NA             "0"            NA             "STRAWBERRIES"
```

```r
head(strawberry)
```

```
##   Program Year Period Geo.Level    State State.ANSI
## 1  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 2  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 3  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 4  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 5  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 6  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
##                        Data.Item     Domain                  Domain.Category
## 1 STRAWBERRIES - ACRES BEARING AREA GROWN   AREA GROWN: (0.1 TO 0.9 ACRES)
## 2 STRAWBERRIES - ACRES BEARING AREA GROWN   AREA GROWN: (1.0 TO 4.9 ACRES)
## 3 STRAWBERRIES - ACRES BEARING AREA GROWN  AREA GROWN: (100 OR MORE ACRES)
## 4 STRAWBERRIES - ACRES BEARING AREA GROWN AREA GROWN: (15.0 TO 24.9 ACRES)
## 5 STRAWBERRIES - ACRES BEARING AREA GROWN AREA GROWN: (25.0 TO 49.9 ACRES)
## 6 STRAWBERRIES - ACRES BEARING AREA GROWN  AREA GROWN: (5.0 TO 14.9 ACRES)
##    Value CV....
## 1    963    5.6
## 2  3,195    5.9
## 3 46,265   25.3
## 4  2,514   20.0
## 5  4,231   13.0
## 6  3,396    8.6
```

3.We will separate the data into census and survey, since I find the we can separate different ways may cause some overall effect on the variable of data.item and domain-categroy.

```r
straw_cen <- strawberry |> filter(Program=="CENSUS")
straw_sur <- strawberry |> filter(Program=="SURVEY")

head(straw_cen)
```

```
##   Program Year Period Geo.Level    State State.ANSI
## 1  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 2  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 3  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 4  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 5  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
## 6  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA
##                        Data.Item     Domain                  Domain.Category
## 1 STRAWBERRIES - ACRES BEARING AREA GROWN   AREA GROWN: (0.1 TO 0.9 ACRES)
## 2 STRAWBERRIES - ACRES BEARING AREA GROWN   AREA GROWN: (1.0 TO 4.9 ACRES)
## 3 STRAWBERRIES - ACRES BEARING AREA GROWN  AREA GROWN: (100 OR MORE ACRES)
## 4 STRAWBERRIES - ACRES BEARING AREA GROWN AREA GROWN: (15.0 TO 24.9 ACRES)
## 5 STRAWBERRIES - ACRES BEARING AREA GROWN AREA GROWN: (25.0 TO 49.9 ACRES)
## 6 STRAWBERRIES - ACRES BEARING AREA GROWN  AREA GROWN: (5.0 TO 14.9 ACRES)
##    Value CV....
## 1    963    5.6
## 2  3,195    5.9
## 3 46,265   25.3
## 4  2,514   20.0
## 5  4,231   13.0
## 6  3,396    8.6
```

```r
head(straw_sur)
```

```
##   Program Year          Period Geo.Level      State State.ANSI
## 1  SURVEY 2024           YEAR  NATIONAL   US TOTAL          NA
## 2  SURVEY 2024           YEAR  NATIONAL   US TOTAL          NA
## 3  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL          NA
## 4  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL          NA
## 5  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL          NA
## 6  SURVEY 2023 MARKETING YEAR     STATE CALIFORNIA           6
##                                                                           Data.Item
## 1 STRAWBERRIES, FRESH MARKET - PRICE RECEIVED, ADJUSTED BASE, MEASURED IN $ / CWT
## 2   STRAWBERRIES, PROCESSING - PRICE RECEIVED, ADJUSTED BASE, MEASURED IN $ / TON
## 3                               STRAWBERRIES - PRICE RECEIVED, MEASURED IN $ / CWT
## 4                 STRAWBERRIES, FRESH MARKET - PRICE RECEIVED, MEASURED IN $ / CWT
## 5                   STRAWBERRIES, PROCESSING - PRICE RECEIVED, MEASURED IN $ / CWT
## 6                               STRAWBERRIES - PRICE RECEIVED, MEASURED IN $ / CWT
##   Domain Domain.Category Value CV....
## 1  TOTAL   NOT SPECIFIED  10.9
## 2  TOTAL   NOT SPECIFIED  4.04
## 3  TOTAL   NOT SPECIFIED   123
## 4  TOTAL   NOT SPECIFIED   142
## 5  TOTAL   NOT SPECIFIED  43.8
## 6  TOTAL   NOT SPECIFIED   121
```

```r
unique_values1 <- unique(straw_sur$Data.Item)
unique_values2 <- unique(straw_cen$Data.Item)
print(unique_values1)
```

```
##  [1] "STRAWBERRIES, FRESH MARKET - PRICE RECEIVED, ADJUSTED BASE, MEASURED IN $ / CWT"
##  [2] "STRAWBERRIES, PROCESSING - PRICE RECEIVED, ADJUSTED BASE, MEASURED IN $ / TON"
##  [3] "STRAWBERRIES - PRICE RECEIVED, MEASURED IN $ / CWT"
##  [4] "STRAWBERRIES, FRESH MARKET - PRICE RECEIVED, MEASURED IN $ / CWT"
##  [5] "STRAWBERRIES, PROCESSING - PRICE RECEIVED, MEASURED IN $ / CWT"
##  [6] "STRAWBERRIES - ACRES HARVESTED"
##  [7] "STRAWBERRIES - ACRES PLANTED"
##  [8] "STRAWBERRIES - PRODUCTION, MEASURED IN $"
##  [9] "STRAWBERRIES - PRODUCTION, MEASURED IN CWT"
## [10] "STRAWBERRIES - PRODUCTION, MEASURED IN TONS"
## [11] "STRAWBERRIES - YIELD, MEASURED IN CWT / ACRE"
## [12] "STRAWBERRIES - YIELD, MEASURED IN TONS / ACRE"
## [13] "STRAWBERRIES, FRESH MARKET - PRICE RECEIVED, 10 YEAR AVG FOR PARITY PURPOSES, MEASURED IN $ / C
## [14] "STRAWBERRIES, FRESH MARKET - PRICE RECEIVED, 10 YEAR AVG, MEASURED IN $ / CWT"
## [15] "STRAWBERRIES, FRESH MARKET - PRODUCTION, MEASURED IN $"
## [16] "STRAWBERRIES, FRESH MARKET, UTILIZED - PRODUCTION, MEASURED IN CWT"
## [17] "STRAWBERRIES, NOT SOLD - PRODUCTION, MEASURED IN CWT"
## [18] "STRAWBERRIES, PROCESSING - PRICE RECEIVED, 10 YEAR AVG FOR PARITY PURPOSES, MEASURED IN $ / TON
## [19] "STRAWBERRIES, PROCESSING - PRICE RECEIVED, 10 YEAR AVG, MEASURED IN $ / TON"
## [20] "STRAWBERRIES, PROCESSING - PRODUCTION, MEASURED IN $"
## [21] "STRAWBERRIES, PROCESSING, UTILIZED - PRODUCTION, MEASURED IN CWT"
## [22] "STRAWBERRIES, UTILIZED - PRODUCTION, MEASURED IN CWT"
## [23] "STRAWBERRIES, UTILIZED - PRODUCTION, MEASURED IN TONS"
## [24] "STRAWBERRIES - APPLICATIONS, MEASURED IN LB"
## [25] "STRAWBERRIES - APPLICATIONS, MEASURED IN LB / ACRE / APPLICATION, AVG"
## [26] "STRAWBERRIES - APPLICATIONS, MEASURED IN LB / ACRE / YEAR, AVG"
## [27] "STRAWBERRIES - APPLICATIONS, MEASURED IN NUMBER, AVG"
## [28] "STRAWBERRIES - TREATED, MEASURED IN PCT OF AREA BEARING, AVG"
## [29] "STRAWBERRIES, BEARING - APPLICATIONS, MEASURED IN LB"
## [30] "STRAWBERRIES, BEARING - APPLICATIONS, MEASURED IN LB / ACRE / APPLICATION, AVG"
## [31] "STRAWBERRIES, BEARING - APPLICATIONS, MEASURED IN LB / ACRE / YEAR, AVG"
## [32] "STRAWBERRIES, BEARING - APPLICATIONS, MEASURED IN NUMBER, AVG"
## [33] "STRAWBERRIES, BEARING - TREATED, MEASURED IN PCT OF AREA BEARING, AVG"
## [34] "STRAWBERRIES, PROCESSING - PRICE RECEIVED, MEASURED IN $ / TON"
## [35] "STRAWBERRIES, PROCESSING, UTILIZED - PRODUCTION, MEASURED IN TONS"
```

‘ 4.In this step, I want to separate two interaction columns which are the domain’s category which contains both domain and its category.

```r
straw_cen_cleaned <- straw_cen %>%
  separate(`Data.Item`, into = c("Commodity_Type", "Operation_Measure"), sep = " - ", extra = "merge",
  separate(Commodity_Type, into = c("Commodity", "Type"), sep = ", ", extra = "merge", fill = "right")
  mutate(
    Commodity = str_trim(Commodity),
    Type = ifelse(is.na(Type), "OTHER", str_trim(Type)),
    Operation_Measure = str_trim(Operation_Measure)
  )%>%
  dplyr::select(-Commodity)

head(straw_cen_cleaned)
```

```
##   Program Year Period Geo.Level    State State.ANSI  Type Operation_Measure
## 1  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 2  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 3  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 4  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 5  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 6  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
##       Domain                 Domain.Category  Value CV....
## 1 AREA GROWN   AREA GROWN: (0.1 TO 0.9 ACRES)    963    5.6
## 2 AREA GROWN   AREA GROWN: (1.0 TO 4.9 ACRES)  3,195    5.9
## 3 AREA GROWN  AREA GROWN: (100 OR MORE ACRES) 46,265   25.3
## 4 AREA GROWN AREA GROWN: (15.0 TO 24.9 ACRES)  2,514   20.0
## 5 AREA GROWN AREA GROWN: (25.0 TO 49.9 ACRES)  4,231   13.0
## 6 AREA GROWN  AREA GROWN: (5.0 TO 14.9 ACRES)  3,396    8.6
```

```r
unique_values3 <- unique(straw_cen_cleaned$Type)
unique_values4 <- unique(straw_cen_cleaned$Operation_Measure)
print(unique_values3)
```

```
## [1] "OTHER"                 "ORGANIC"               "ORGANIC, FRESH MARKET"
## [4] "ORGANIC, PROCESSING"
```

```r
print(unique_values4)
```

```
##  [1] "ACRES BEARING"                    "ACRES GROWN"
##  [3] "ACRES NON-BEARING"                "OPERATIONS WITH AREA BEARING"
##  [5] "OPERATIONS WITH AREA GROWN"       "OPERATIONS WITH AREA NON-BEARING"
##  [7] "ACRES HARVESTED"                  "OPERATIONS WITH AREA HARVESTED"
##  [9] "OPERATIONS WITH SALES"            "PRODUCTION, MEASURED IN CWT"
## [11] "SALES, MEASURED IN $"             "SALES, MEASURED IN CWT"
```

5. In this step we will focus on the survey data set. Since it is more complex, we will separate them in more columns.

```
straw_sur_cleaned <- straw_sur %>%
  separate(`Data.Item`, into = c("Commodity_Market", "Details"), sep = " - ", extra = "merge", fill = "
  separate(Commodity_Market, into = c("Commodity", "Market_Type"), sep = ", ", extra = "merge", fill = 
  separate(Details, into = c("Measure_Operation", "Unit_of_Measure"), sep = ", MEASURED IN ", extra = "
  mutate(
    Commodity = str_trim(Commodity),
    Market_Type = ifelse(is.na(Market_Type), "OTHER", str_trim(Market_Type)),
    Measure_Operation = str_trim(Measure_Operation),
    Unit_of_Measure = str_trim(Unit_of_Measure)
  )%>%
  dplyr::select(-Commodity)

head(straw_sur_cleaned)
```

```
##   Program Year         Period Geo.Level      State State.ANSI  Market_Type
## 1  SURVEY 2024           YEAR  NATIONAL   US TOTAL         NA FRESH MARKET
## 2  SURVEY 2024           YEAR  NATIONAL   US TOTAL         NA   PROCESSING
## 3  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         NA        OTHER
## 4  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         NA FRESH MARKET
## 5  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         NA   PROCESSING
## 6  SURVEY 2023 MARKETING YEAR     STATE CALIFORNIA          6        OTHER
##                Measure_Operation Unit_of_Measure Domain Domain.Category Value
## 1 PRICE RECEIVED, ADJUSTED BASE         $ / CWT  TOTAL   NOT SPECIFIED  10.9
## 2 PRICE RECEIVED, ADJUSTED BASE         $ / TON  TOTAL   NOT SPECIFIED  4.04
## 3                PRICE RECEIVED         $ / CWT  TOTAL   NOT SPECIFIED   123
## 4                PRICE RECEIVED         $ / CWT  TOTAL   NOT SPECIFIED   142
## 5                PRICE RECEIVED         $ / CWT  TOTAL   NOT SPECIFIED  43.8
## 6                PRICE RECEIVED         $ / CWT  TOTAL   NOT SPECIFIED   121
##   CV....
## 1
## 2
## 3
## 4
## 5
## 6
```

```
unique_values5 <- unique(straw_sur_cleaned$Market_Type)
unique_values6 <- unique(straw_sur_cleaned$Measure_Operation)
print(unique_values5)
```

```
## [1] "FRESH MARKET"            "PROCESSING"              "OTHER"
## [4] "FRESH MARKET, UTILIZED" "NOT SOLD"               "PROCESSING, UTILIZED"
## [7] "UTILIZED"               "BEARING"
```

```
print(unique_values6)
```

```
##  [1] "PRICE RECEIVED, ADJUSTED BASE"
##  [2] "PRICE RECEIVED"
##  [3] "ACRES HARVESTED"
##  [4] "ACRES PLANTED"
##  [5] "PRODUCTION"
##  [6] "YIELD"
##  [7] "PRICE RECEIVED, 10 YEAR AVG FOR PARITY PURPOSES"
##  [8] "PRICE RECEIVED, 10 YEAR AVG"
##  [9] "APPLICATIONS"
## [10] "TREATED"
```

6: In this step I want to o the same thing on Domain.Category that separates it into two columns. I found in the survey graph, the this column has two cases, the not specified annd chemical details as mentioned in the assignment instructions.

```
straw_sur_cleaned1 <- straw_sur_cleaned %>%
  separate(`Domain.Category`, into = c("Chemical_Use", "Chemical_Details"), sep = ": ", extra = "merge"
  mutate(
    Chemical_Use = str_trim(str_replace(Chemical_Use, "CHEMICAL, ", "")),
    Chemical_Details = ifelse(Chemical_Use == "NOT SPECIFIED", "NOT SPECIFIED", Chemical_Details)
  ) %>%
  separate(Chemical_Details, into = c("Chemical_Name", "Chemical_Code"), sep = " = ", extra = "merge", 
  mutate(
    Chemical_Name = str_trim(str_replace_all(Chemical_Name, "[()]", "")),
    Chemical_Code = str_trim(str_replace(Chemical_Code, "[)]$", ""))
  )
straw_sur_cleaned1 <- straw_sur_cleaned1 %>%
  mutate(
    Chemical_Code = as.numeric(Chemical_Code)
  )

head(straw_sur_cleaned1)
```

```
##   Program Year         Period Geo.Level      State State.ANSI  Market_Type
## 1  SURVEY 2024           YEAR  NATIONAL   US TOTAL         NA FRESH MARKET
## 2  SURVEY 2024           YEAR  NATIONAL   US TOTAL         NA   PROCESSING
## 3  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         NA        OTHER
## 4  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         NA FRESH MARKET
## 5  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         NA   PROCESSING
## 6  SURVEY 2023 MARKETING YEAR     STATE CALIFORNIA          6        OTHER
##               Measure_Operation Unit_of_Measure Domain  Chemical_Use
## 1 PRICE RECEIVED, ADJUSTED BASE         $ / CWT  TOTAL NOT SPECIFIED
## 2 PRICE RECEIVED, ADJUSTED BASE         $ / TON  TOTAL NOT SPECIFIED
## 3                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 4                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 5                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 6                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
##   Chemical_Name Chemical_Code Value CV....
## 1 NOT SPECIFIED            NA  10.9
## 2 NOT SPECIFIED            NA  4.04
## 3 NOT SPECIFIED            NA   123
## 4 NOT SPECIFIED            NA   142
## 5 NOT SPECIFIED            NA  43.8
## 6 NOT SPECIFIED            NA   121
```

7. Then we will focus on the census data just like last step.

```
unique_values8 <- unique(straw_cen_cleaned$Domain.Category)
print(unique_values8)
```

```
## [1] "AREA GROWN: (0.1 TO 0.9 ACRES)"
## [2] "AREA GROWN: (1.0 TO 4.9 ACRES)"
## [3] "AREA GROWN: (100 OR MORE ACRES)"
## [4] "AREA GROWN: (15.0 TO 24.9 ACRES)"
## [5] "AREA GROWN: (25.0 TO 49.9 ACRES)"
## [6] "AREA GROWN: (5.0 TO 14.9 ACRES)"
## [7] "AREA GROWN: (50.0 TO 99.9 ACRES)"
## [8] "NOT SPECIFIED"
## [9] "ORGANIC STATUS: (NOP USDA CERTIFIED)"
```

```
straw_cen_cleaned1 <- straw_cen_cleaned %>%
  separate(`Domain.Category`, into = c("Category_Type", "Details"), sep = ": ", extra = "merge", fill =
  mutate(
    Category_Type = str_trim(Category_Type),
    Details = ifelse(Category_Type == "NOT SPECIFIED", "NOT SPECIFIED", str_trim(str_replace_all(Details
  )

head(straw_cen_cleaned1)
```

```
##   Program Year Period Geo.Level    State State.ANSI  Type Operation_Measure
## 1  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 2  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 3  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 4  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 5  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
## 6  CENSUS 2022   YEAR  NATIONAL US TOTAL         NA OTHER     ACRES BEARING
##       Domain Category_Type            Details  Value CV....
## 1 AREA GROWN    AREA GROWN   0.1 TO 0.9 ACRES    963    5.6
## 2 AREA GROWN    AREA GROWN   1.0 TO 4.9 ACRES  3,195    5.9
## 3 AREA GROWN    AREA GROWN  100 OR MORE ACRES 46,265   25.3
## 4 AREA GROWN    AREA GROWN 15.0 TO 24.9 ACRES  2,514   20.0
## 5 AREA GROWN    AREA GROWN 25.0 TO 49.9 ACRES  4,231   13.0
## 6 AREA GROWN    AREA GROWN  5.0 TO 14.9 ACRES  3,396    8.6
```

8: In this step we will use deal with the N.A data in both data set First: we change the NAs in State.ANSI to -1, since when we want all the variables to be numeric, which is eaiser when comparision.

```
straw_sur_cleaned1 <- straw_sur_cleaned1 %>%
  mutate(
    `State.ANSI` = ifelse(is.na(`State.ANSI`), -1, `State.ANSI`)
  )
straw_cen_cleaned1 <- straw_cen_cleaned1 %>%
  mutate(
    `State.ANSI` = ifelse(is.na(`State.ANSI`), -1, `State.ANSI`)
  )
head(straw_sur_cleaned1)
```

```
##   Program Year          Period Geo.Level      State State.ANSI  Market_Type
## 1  SURVEY 2024            YEAR  NATIONAL   US TOTAL         -1 FRESH MARKET
## 2  SURVEY 2024            YEAR  NATIONAL   US TOTAL         -1   PROCESSING
## 3  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         -1        OTHER
## 4  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         -1 FRESH MARKET
## 5  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         -1   PROCESSING
## 6  SURVEY 2023 MARKETING YEAR     STATE CALIFORNIA          6        OTHER
##                Measure_Operation Unit_of_Measure Domain  Chemical_Use
## 1 PRICE RECEIVED, ADJUSTED BASE         $ / CWT  TOTAL NOT SPECIFIED
## 2 PRICE RECEIVED, ADJUSTED BASE         $ / TON  TOTAL NOT SPECIFIED
## 3                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 4                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 5                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 6                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
##   Chemical_Name Chemical_Code Value CV....
## 1 NOT SPECIFIED            NA  10.9
## 2 NOT SPECIFIED            NA  4.04
## 3 NOT SPECIFIED            NA   123
## 4 NOT SPECIFIED            NA   142
## 5 NOT SPECIFIED            NA  43.8
## 6 NOT SPECIFIED            NA   121
```

```r
head(straw_cen_cleaned1)
```

```
##   Program Year Period Geo.Level    State State.ANSI  Type Operation_Measure
## 1  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
## 2  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
## 3  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
## 4  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
## 5  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
## 6  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
##       Domain Category_Type            Details  Value CV....
## 1 AREA GROWN    AREA GROWN   0.1 TO 0.9 ACRES    963    5.6
## 2 AREA GROWN    AREA GROWN   1.0 TO 4.9 ACRES  3,195    5.9
## 3 AREA GROWN    AREA GROWN  100 OR MORE ACRES 46,265   25.3
## 4 AREA GROWN    AREA GROWN 15.0 TO 24.9 ACRES  2,514   20.0
## 5 AREA GROWN    AREA GROWN 25.0 TO 49.9 ACRES  4,231   13.0
## 6 AREA GROWN    AREA GROWN  5.0 TO 14.9 ACRES  3,396    8.6
```

9:The second step for this is that I want to fill the missing data in the value. I will create a linear regression with Year, Category_Type, Details, State.ANSI of the census data.

```r
straw_cen_cleaned1 <- straw_cen_cleaned1 %>%
  mutate(Value = as.numeric(Value)) %>%
  drop_na(Value, Year, Category_Type, Details, State.ANSI)
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Value = as.numeric(Value)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```r
straw_cen_cleaned1 <- straw_cen_cleaned1 %>%
  mutate(
    Category_Type = as.factor(Category_Type),
    Details = as.factor(Details)
  )

modelcen <- lm(Value ~ Year + Category_Type + Details + State.ANSI, data = straw_cen_cleaned1)

summary(modelcen)
```

```
##
## Call:
```

```
## lm(formula = Value ~ Year + Category_Type + Details + State.ANSI,
##     data = straw_cen_cleaned1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -152.25  -94.23  -65.46   30.19  797.67
##
## Coefficients: (2 not defined because of singularities)
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -1.591e+04  1.702e+04  -0.935   0.3502
## Year                         7.950e+00  8.417e+00   0.944   0.3452
## Category_TypeNOT SPECIFIED  -1.117e+01  2.813e+01  -0.397   0.6914
## Category_TypeORGANIC STATUS -3.976e+01  3.265e+01  -1.218   0.2237
## Details1.0 TO 4.9 ACRES     -2.953e+01  3.476e+01  -0.849   0.3959
## Details100 OR MORE ACRES    -2.478e+01  4.446e+01  -0.557   0.5775
## Details15.0 TO 24.9 ACRES   -7.209e+01  3.535e+01  -2.039   0.0417 *
## Details25.0 TO 49.9 ACRES   -9.377e+01  3.866e+01  -2.426   0.0155 *
## Details5.0 TO 14.9 ACRES    -2.593e+01  3.449e+01  -0.752   0.4523
## Details50.0 TO 99.9 ACRES   -8.935e+01  4.519e+01  -1.977   0.0483 *
## DetailsNOP USDA CERTIFIED           NA         NA      NA       NA
## DetailsNOT SPECIFIED                NA         NA      NA       NA
## State.ANSI                  -5.113e-01  3.430e-01  -1.491   0.1364
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 167.8 on 942 degrees of freedom
## Multiple R-squared:  0.02574,    Adjusted R-squared:  0.0154
## F-statistic: 2.489 on 10 and 942 DF,  p-value: 0.006007
```

```
numeric_data <- straw_cen_cleaned1 %>%
  filter(!is.na(Value) & grepl("^[0-9.]+$", Value)) %>%
  mutate(Value = as.numeric(Value))

non_numeric_data <- straw_cen_cleaned1 %>%
  filter(is.na(Value) | !grepl("^[0-9.]+$", Value))

predicted_values <- predict(modelcen, newdata = non_numeric_data)

non_numeric_data <- non_numeric_data %>%
  mutate(Value = predicted_values)

straw_cen_cleaned2 <- bind_rows(numeric_data, non_numeric_data)
head(straw_cen_cleaned2)
```

```
##   Program Year Period Geo.Level    State State.ANSI  Type Operation_Measure
## 1  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER     ACRES BEARING
## 2  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER ACRES NON-BEARING
## 3  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER ACRES NON-BEARING
## 4  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER ACRES NON-BEARING
## 5  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER ACRES NON-BEARING
## 6  CENSUS 2022   YEAR  NATIONAL US TOTAL         -1 OTHER ACRES NON-BEARING
##       Domain Category_Type            Details Value CV....
## 1 AREA GROWN    AREA GROWN   0.1 TO 0.9 ACRES   963    5.6
## 2 AREA GROWN    AREA GROWN   0.1 TO 0.9 ACRES   236   13.2
## 3 AREA GROWN    AREA GROWN   1.0 TO 4.9 ACRES   535    5.1
## 4 AREA GROWN    AREA GROWN  100 OR MORE ACRES   666    4.2
## 5 AREA GROWN    AREA GROWN 15.0 TO 24.9 ACRES   244   34.1
## 6 AREA GROWN    AREA GROWN 25.0 TO 49.9 ACRES   210   37.3
```

10. We will do the same thing to survey data. We found a really nice model with a R^2 over 0.58. Contrasting to the model for census data, it is more than 10 times better .

```r
straw_sur_cleaned1 <- straw_sur_cleaned1 %>%
  mutate(Value = as.numeric(Value)) %>%
  drop_na(Value, Year, Market_Type, Unit_of_Measure, State.ANSI)
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Value = as.numeric(Value)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```r
straw_sur_cleaned1 <- straw_sur_cleaned1 %>%
  mutate(
    Market_Type = as.factor(Market_Type),
    Unit_of_Measure = as.factor(Unit_of_Measure)
  )

modelsur <- lm(Value ~ Year + Unit_of_Measure + Market_Type + State.ANSI, data = straw_sur_cleaned1)
summary(modelsur)
```

```
##
## Call:
## lm(formula = Value ~ Year + Unit_of_Measure + Market_Type + State.ANSI,
##     data = straw_sur_cleaned1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -344.74  -15.40   -6.11    2.93  593.57
##
## Coefficients: (1 not defined because of singularities)
##                                               Estimate Std. Error t value
## (Intercept)                                  3.822e+03  2.334e+03   1.638
## Year                                        -1.869e+00  1.155e+00  -1.619
## Unit_of_Measure$ / CWT                       6.928e+01  3.305e+01   2.096
## Unit_of_Measure$ / TON                       1.776e+02  4.000e+01   4.439
## Unit_of_MeasureCWT                          -4.589e+01  5.700e+01  -0.805
## Unit_of_MeasureCWT / ACRE                    2.970e+02  3.590e+01   8.272
## Unit_of_MeasureLB                            3.874e+02  7.371e+01   5.255
## Unit_of_MeasureLB / ACRE / APPLICATION, AVG -3.260e+01  7.312e+01  -0.446
## Unit_of_MeasureLB / ACRE / YEAR, AVG        -2.373e+01  7.312e+01  -0.325
## Unit_of_MeasureNUMBER, AVG                  -3.759e+01  7.311e+01  -0.514
## Unit_of_MeasurePCT OF AREA BEARING, AVG     -4.118e+00  7.309e+01  -0.056
## Unit_of_MeasureTONS                         -4.589e+01  8.009e+01  -0.573
## Unit_of_MeasureTONS / ACRE                  -1.994e+01  4.904e+01  -0.407
## Market_TypeFRESH MARKET                      6.653e+00  6.897e+01   0.096
## Market_TypeNOT SOLD                          1.540e+02  5.087e+01   3.026
## Market_TypeOTHER                             3.797e-12  6.497e+01   0.000
## Market_TypePROCESSING                       -6.263e+01  6.924e+01  -0.905
## Market_TypePROCESSING, UTILIZED              1.767e+01  6.178e+01   0.286
## Market_TypeUTILIZED                                 NA         NA      NA
## State.ANSI                                  -9.127e-01  4.291e-01  -2.127
##                                              Pr(>|t|)
## (Intercept)                                   0.10173
## Year                                          0.10578
## Unit_of_Measure$ / CWT                        0.03624 *
## Unit_of_Measure$ / TON                       9.75e-06 ***
## Unit_of_MeasureCWT                            0.42094
## Unit_of_MeasureCWT / ACRE                    3.01e-16 ***
## Unit_of_MeasureLB                            1.71e-07 ***
## Unit_of_MeasureLB / ACRE / APPLICATION, AVG  0.65578
## Unit_of_MeasureLB / ACRE / YEAR, AVG         0.74553
## Unit_of_MeasureNUMBER, AVG                    0.60717
## Unit_of_MeasurePCT OF AREA BEARING, AVG       0.95507
## Unit_of_MeasureTONS                           0.56678
## Unit_of_MeasureTONS / ACRE                    0.68431
## Market_TypeFRESH MARKET                       0.92316
## Market_TypeNOT SOLD                           0.00252 **
## Market_TypeOTHER                              1.00000
## Market_TypePROCESSING                         0.36581
## Market_TypePROCESSING, UTILIZED               0.77493
## Market_TypeUTILIZED                                NA
## State.ANSI                                    0.03360 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 79.57 on 1413 degrees of freedom
## Multiple R-squared:  0.5885, Adjusted R-squared:  0.5832
## F-statistic: 112.3 on 18 and 1413 DF,  p-value: < 2.2e-16
```

```
numeric_data <- straw_sur_cleaned1 %>%
  filter(!is.na(Value) & grepl("^[0-9.]+$", Value)) %>%
  mutate(Value = as.numeric(Value))

non_numeric_data <- straw_sur_cleaned1 %>%
  filter(is.na(Value) | !grepl("^[0-9.]+$", Value))

predicted_values <- predict(modelsur, newdata = non_numeric_data)

non_numeric_data <- non_numeric_data %>%
  mutate(Value = predicted_values)

straw_sur_cleaned2 <- bind_rows(numeric_data, non_numeric_data)
head(straw_sur_cleaned2)
```

```
##   Program Year         Period Geo.Level      State State.ANSI  Market_Type
## 1  SURVEY 2024           YEAR  NATIONAL   US TOTAL         -1 FRESH MARKET
## 2  SURVEY 2024           YEAR  NATIONAL   US TOTAL         -1   PROCESSING
## 3  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         -1        OTHER
## 4  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         -1 FRESH MARKET
## 5  SURVEY 2023 MARKETING YEAR  NATIONAL   US TOTAL         -1   PROCESSING
## 6  SURVEY 2023 MARKETING YEAR     STATE CALIFORNIA          6        OTHER
##               Measure_Operation Unit_of_Measure Domain  Chemical_Use
## 1 PRICE RECEIVED, ADJUSTED BASE         $ / CWT  TOTAL NOT SPECIFIED
## 2 PRICE RECEIVED, ADJUSTED BASE         $ / TON  TOTAL NOT SPECIFIED
## 3                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 4                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 5                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
## 6                PRICE RECEIVED         $ / CWT  TOTAL NOT SPECIFIED
##   Chemical_Name Chemical_Code  Value CV....
## 1 NOT SPECIFIED            NA  10.90
## 2 NOT SPECIFIED            NA   4.04
## 3 NOT SPECIFIED            NA 123.00
## 4 NOT SPECIFIED            NA 142.00
## 5 NOT SPECIFIED            NA  43.80
## 6 NOT SPECIFIED            NA 121.00
```

10.We will output those two dataset.

```
write.csv(straw_sur_cleaned2, "straw_sur_cleaned2.csv", row.names = FALSE)
write.csv(straw_cen_cleaned2, "straw_cen_cleaned2.csv", row.names = FALSE)
```