````
---
title: "615strawberry"
author: "Yishun Zhang"
date: "2024-10-07"
output: pdf_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

Set-up

```{r}
#| label: load libraries and set options
#| warning: false
#| message: false

install.packages("kableExtra")

library(knitr)
library(kableExtra)
library(tidyverse)
strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv", col_names = TRUE)
glimpse(strawberry)


```
Read the data and take a first look

```{r}
#| label: read data - glimpse

library(tidyverse)

strawberry_data <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv")

glimpse(strawberry_data)

head(strawberry_data)

```


I have 12699 rows and 21 columns.
````

All I can see from the glimpse is I have date, location, values and coefficients of variation.

Examine the data. How is it organized?

```{r}
#| label: explore organization 1

library(tidyverse)

strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv", col_names = TRUE)

glimpse(strawberry)

summary(strawberry)

colSums(is.na(strawberry))
str(strawberry)

distinct_states <- strawberry |> distinct(State)
print(distinct_states)

state_counts <- strawberry |> group_by(State) |> count()
print(state_counts)

summary(strawberry$Value)
hist(as.numeric(strawberry$Value), main="Distribution of Value", xlab="Value", col="skyblue", breaks=50)
state_year_check <- strawberry |> group_by(State, Year) |> count()
print(state_year_check)

duplicates_check <- strawberry |> duplicated()
sum(duplicates_check)



```

remove columns with a single value in all rows

```{r}
#|label: function def - drop 1-item columns
library(readr)

strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv")
```

```
glimpse(strawberry)


drop_one_value_col <- function(df) {
  drop <- NULL
  for(i in 1:dim(df)[2]){
    if(n_distinct(df[[i]]) == 1){
      drop <- c(drop, i)
    }
  }
  if(!is.null(drop)){
    df <- df[,-drop]
  }
  return(df)
}

strawberry <- drop_one_value_col(strawberry)

```

To get better look at the data, look at California.

```{r}
#| label: explore California only
library(tidyverse)
library(readr)
strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv")
glimpse(strawberry)
strawberry <- strawberry |>
  mutate(`Data Item` = str_trim(`Data Item`, side = "both"))

strawberry <- strawberry |>
  separate(`Data Item`, into = c("Fruit", "Category", "Item", "Metric"), sep = ",",
fill = "right")
strawberry
california_data <- strawberry |> filter(State == "CALIFORNIA")
glimpse(california_data)
unique(california_data$Program)
unique(california_data$Year)
unique(california_data$Category)
california_census <- california_data |> filter(Program == "CENSUS")
california_survey <- california_data |> filter(Program == "SURVEY")
glimpse(california_census)
glimpse(california_survey)


```

Explore California to understand the census and survey

```{r}
#| label: explore Calif census and survey
library(tidyverse)
strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv", col_names = TRUE)



california_data <- strawberry |> filter(State == "CALIFORNIA")

calif_census <- california_data |> filter(Program == "CENSUS")
calif_survey <- california_data |> filter(Program == "SURVEY")

glimpse(calif_census)
glimpse(calif_survey)
summary(calif_census)
summary(calif_survey)
drop_one_value_col <- function(df) {
  drop <- NULL
  for(i in 1:dim(df)[2]){
    if(n_distinct(df[[i]]) == 1){
      drop <- c(drop, i)
    }
  }
  if(!is.null(drop)){
    df <- df[,-drop]
  }
  return(df)
}
strawberry <- strawberry |>
mutate(`Data Item` = str_replace_all(`Data Item`, " - ", ","))
#Split 'Data Item' into 4 columns
strawberry <- strawberry |>
separate_wider_delim( cols = `Data Item`,
delim = ",",
names = c("Fruit",
"Category",
"Item",
"Metric"),
too_many = "merge",
too_few = "align_start"
)
#Remove 'measured in' to metric columns
```

```
strawberry <- strawberry |>
mutate(Metric = ifelse(grepl("MEASURED IN", Item), Item, Metric), # Move the 'Item' value to 'Metric' if it contains 'MEASURED IN'
Item = ifelse(grepl("MEASURED IN", Item), NA, Item) # Set 'Item' to NA where we moved the value
)
#Remove 'production' to its correct way.
strawberry <- strawberry |>
mutate(
Item = ifelse(grepl("PRODUCTION", Metric), "PRODUCTION", Item), # Move 'PRODUCTION' to 'Item'
Metric = ifelse(grepl("PRODUCTION", Metric), sub("PRODUCTION, ", "", Metric), Metric) # Remove 'PRODUCTION' from 'Metric'
)
#Remove 'utilized' from category to Item
3
strawberry <- strawberry |>
mutate(
Item = ifelse(grepl("UTILIZED", Category, ignore.case = TRUE),
paste("UTILIZED", Item, sep = " "), # Combine 'Item' with 'Utilized'
Item), # Keep 'Item' unchanged if 'Utilized' not found
Category = ifelse(grepl("UTILIZED", Category, ignore.case = TRUE), NA, Category)# Set 'Category' to NA where 'Utilized' is moved
)
#Consider a better waty to move items in one step.
movingitem<- c("ACRES BEARING", "ACRES NON-BEARING", "ACRES GROWN", "OPERATIONS WITH AREA BEARING", "YIELD", "ACRES HARVESTED", "ACRES PLANTED", "OPERATIONS WITH AREA GROWN", "OPERATIONS WITH AREA NON-BEARING","PRODUCTION")
# Move terms from 'Metric' or 'Category' to 'Item' without replacing 'Metric' data
strawberry <- strawberry |>
mutate(Item = ifelse(grepl(paste(movingitem, collapse = "|"), Category,
ignore.case = TRUE) & is.na(Item), Category,
ifelse(grepl(paste(movingitem, collapse = "|"), Category, ignore.case = TRUE),
paste(Item, Category, sep = ", "), Item)
),
Category = ifelse(grepl(paste(movingitem, collapse = "|"), Category,
ignore.case = TRUE),
NA, Category)
)
```

`Data Item` into (fruit, category, item)

```{r}
#|label: split Data Item
library(tidyverse)

strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv", col_names = TRUE)

census_data <- strawberry |> filter(Program == "CENSUS")
survey_data <- strawberry |> filter(Program == "SURVEY")
drop_one_value_col <- function(df) {
  drop <- NULL
  for(i in 1:dim(df)[2]){
    if(n_distinct(df[[i]]) == 1){
      drop <- c(drop, i)
    }
  }
  if(!is.null(drop)){
    df <- df[,-drop]
  }
  return(df)
}
census_data <- drop_one_value_col(census_data)
survey_data <- drop_one_value_col(survey_data)

```

There is a problem you have to fix -- a leading space.

```{r}
#|label: fix the leading space
clean_data <- function(df) {
  df <- df %>%
    mutate(across(c(Category, Item, Metric), str_trim))
  return(df)
}

census_data <- clean_data(census_data)
survey_data <- clean_data(survey_data)

cleaned_strawberry <- bind_rows(census_data, survey_data)

cleaned_strawberry



```

now exam the Fruit column -- find hidden sub-columns

```{r}

library(tidyverse)

strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv", col_names = TRUE)

strawberry <- strawberry |>
separate_wider_delim(
cols = Domain,
delim = " , ",
names = c("Area Grown", "Fertilize", "Organic", "Chemical"),
too_many = "merge",
too_few = "align_start"
)
#Loading variables to each column
strawberry <- strawberry |>
mutate(
Chemical = ifelse(grepl("CHEMICAL", `Area Grown`, ignore.case = TRUE), `Area Grown`, NA),
Organic = ifelse(grepl("ORGANIC", `Area Grown`, ignore.case = TRUE), `Area Grown`, NA),
Fertilize = ifelse(grepl("FERTILIZER", `Area Grown`, ignore.case = TRUE), `Area Grown`, NA),
`Area Grown` = ifelse(grepl("CHEMICAL|ORGANIC|FERTILIZER", `Area Grown`, ignore.case = TRUE), NA, `Area Grown`)
)
#Dealing with 'Domain Category' column
strawberry <- strawberry |>
mutate(
Chemical = ifelse(grepl("CHEMICAL", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
Chemical),
Organic = ifelse(grepl("ORGANIC", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
Organic),
Fertilize = ifelse(grepl("FERTILIZER", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
Fertilize),
`Area Grown` = ifelse(grepl("AREA", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
`Area Grown`),
`Domain Category` = ifelse(grepl("CHEMICAL|ORGANIC|FERTILIZER|AREA", `Domain Category`, ignore.case = TRUE), NA, `Domain Category`)
)
```

```
#Move 'Total' to its best place
strawberry <- strawberry |>
  mutate(`Data Item` = str_trim(`Data Item`, side = "both"))

strawberry <- strawberry |>
  separate(`Data Item`, into = c("Fruit", "Category", "Item", "Metric"), sep = ",",
fill = "right")
strawberry
strawberry <- strawberry |>
mutate(Item = ifelse(grepl("Total", `Area Grown`, ignore.case = TRUE),
paste("Total", Item, sep = " "),
Item),
`Area Grown` = ifelse(grepl("Total", `Area Grown`, ignore.case = TRUE), NA, `Area
Grown`)
)




```

```{r}


library(tidyverse)

strawberry <- read_csv("C:/Users/17756/Downloads/strawberries25_v3.csv", col_names =
TRUE)
strawberry <- strawberry |>
separate_wider_delim(
cols = Domain,
delim = " , ",
names = c("Area Grown", "Fertilize", "Organic", "Chemical"),
too_many = "merge",
too_few = "align_start"
)
#Loading variables to each column
strawberry <- strawberry |>
mutate(
Chemical = ifelse(grepl("CHEMICAL", `Area Grown`, ignore.case = TRUE), `Area Grown`,
NA),
Organic = ifelse(grepl("ORGANIC", `Area Grown`, ignore.case = TRUE), `Area Grown`, NA),
Fertilize = ifelse(grepl("FERTILIZER", `Area Grown`, ignore.case = TRUE), `Area Grown`,
NA),
`Area Grown` = ifelse(grepl("CHEMICAL|ORGANIC|FERTILIZER", `Area Grown`, ignore.case
= TRUE), NA, `Area Grown`)
)
```

```
#Dealing with 'Domain Category' column
strawberry <- strawberry |>
mutate(
Chemical = ifelse(grepl("CHEMICAL", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
Chemical),
Organic = ifelse(grepl("ORGANIC", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
Organic),
Fertilize = ifelse(grepl("FERTILIZER", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
Fertilize),
`Area Grown` = ifelse(grepl("AREA", `Domain Category`, ignore.case = TRUE),
`Domain Category`,
`Area Grown`),
`Domain Category` = ifelse(grepl("CHEMICAL|ORGANIC|FERTILIZER|AREA", `Domain
Category`, ignore.case = TRUE), NA, `Domain Category`)
)
strawberry <- strawberry |>
mutate(Chemical = str_replace_all(Chemical, "[,:=()]", ","))
#Split it into three columns
strawberryc<- strawberry |>
separate_wider_delim(
cols = Chemical,
delim = ",",
names = c("Type", "Ingredient", "Code"), #Separate Chemical into type, ingredient, and
code.
too_many = "merge",
too_few = "align_start"
)
#Filling in the columns
strawberryc<- strawberryc |>
mutate(
Type = ifelse(Type == "CHEMICAL" | is.na(Type), Ingredient, Type),Ingredient =
ifelse(!is.na(Ingredient), str_extract(Code, "\\b[A-Za-z\\-\\.\\s]+\\b"),
Ingredient), #"\\b[A-Za-z0\\-\\.\\s]+\\b" are regular expressions, which are used to
extract specific numbers or words
Code = str_replace(Code, "\\b[A-Za-z\\-\\.\\s]+\\b", "")
)
#Clean 'Code' Column
strawberryc <- strawberryc |>
mutate(
Code = str_replace_all(Code, "^\\s*,+|,+\\s*$|\\s*,\\s*,+", ""),
Code = str_trim(Code)
)
head(strawberryc)
```

```
```