Stawberries 3

AUTHOR
MA615

PUBLISHED
September 30, 2024

Version 5

We ditch the counties

Preparing data for analysis

Acquire, explore, clean & structure, EDA

Data cleaning and organization

"An introduction to data cleaning with R" by Edwin de Jonge and Mark van der Loo

"Problems, Methods, and Challenges in Comprehensive Data Cleansing" by Heiko Müller and Johann-Christoph Freytag

Strawberries

Questions

- Where they are grown? By whom?
- Are they really loaded with carcinogenic poisons?
- Are they really good for your health? Bad for your health?
- Are organic strawberries carriers of deadly diseases?
- When I go to the market should I buy conventional or organic strawberries?
- Do Strawberry farmers make money?
- How do the strawberries I buy get to my market?

The data

The data set for this assignment has been selected from:

[USDA_NASS_strawb_2024SEP25 The data have been stored on NASS here: USDA_NASS_strawb_2024SEP25

and has been stored on the blackboard as strawberries25_v3.csv.

read and explore the data

Set-up

Read the data and take a first look

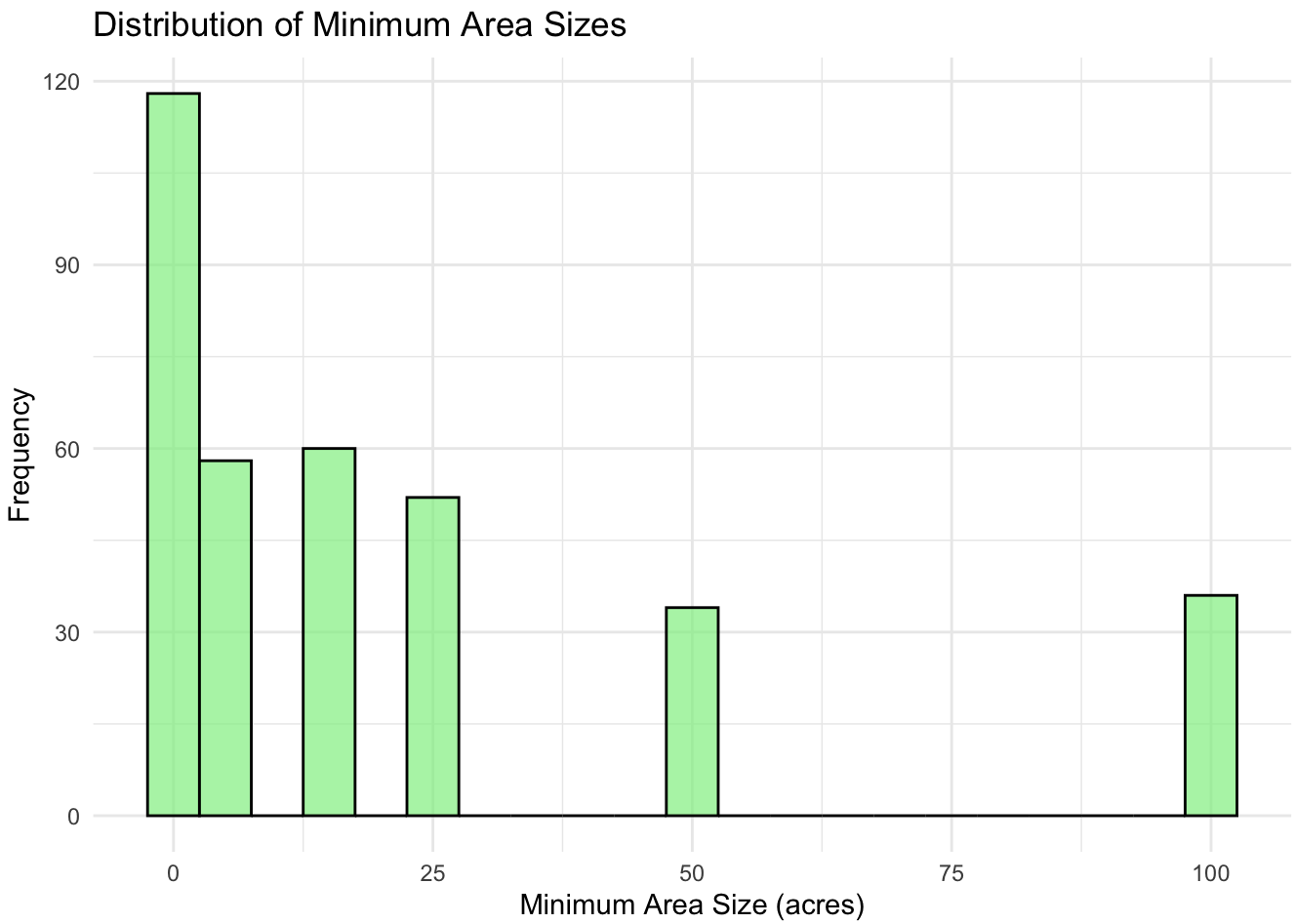
```
#install.packages("stringr")
#install.packages("dplyr")
library(dplyr)
library(stringr)
library(readr)
library(ggplot2)

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

```
Rows: 12669 Columns: 21
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (12): Program, Period, Geo Level, State, Ag District, County, Commodity,...
dbl  (5): Year, State ANSI, Ag District Code, County ANSI, watershed_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.
```

```
split_chemical_data <- function(domain_category) {
  # Check if 'domain_category' is NA or does not contain the word 'CHEMICAL'
  if (is.na(domain_category) || !grepl("CHEMICAL", domain_category)) {
    return(c(NA, NA, NA))
  }

  # Extract chemical info with a code (e.g., "CHEMICAL: (USE = CODE)")
  match_with_code <- regmatches(domain_category, regexec("([A-Z ]+): \\(([^=]+) = (\\d+)\\)

  # Extract chemical info without a code (e.g., "CHEMICAL: (USE)")
  match_without_code <- regmatches(domain_category, regexec("([A-Z ]+): \\(([^)]+)\\)", dom

  # If 'match_with_code' succeeds, return chemical name, use, and code
  if (length(match_with_code[[1]]) == 4) {
    return(c(trimws(match_with_code[[1]][2]), trimws(match_with_code[[1]][3]), match_with_c

  # If 'match_without_code' succeeds, return chemical name, use, and NA
  } else if (length(match_without_code[[1]]) == 3) {
    return(c(trimws(match_without_code[[1]][2]), trimws(match_without_code[[1]][3]), NA))

  # Return NAs if neither pattern matches
  } else {
    return(c(NA, NA, NA))
  }
}

# Apply the function to the Domain Category column
split_data <- t(sapply(strawberry_data$`Domain Category`, split_chemical_data))
strawberry_data <- cbind(strawberry_data, split_data)
colnames(strawberry_data)[(ncol(strawberry_data)-2):ncol(strawberry_data)] <- c("use", "nam
```

```r
# Modify 'strawberry_data' to update 'use' and 'name' columns based on 'Domain Category'
strawberry_data <- strawberry_data %>%
  # Set 'use' to "ORGANIC STATUS" if 'Domain Category' contains "ORGANIC STATUS: (NOP USDA
  mutate(
    use = ifelse(grepl("ORGANIC STATUS: \\(NOP USDA CERTIFIED\\)", `Domain Category`),
                 "ORGANIC STATUS", use),
    # Set 'name' to "NOP USDA CERTIFIED" if 'Domain Category' contains "ORGANIC STATUS: (NO
    name = ifelse(grepl("ORGANIC STATUS: \\(NOP USDA CERTIFIED\\)", `Domain Category`),
                  "NOP USDA CERTIFIED", name)
  )

# Modify 'strawberry_data' to update 'use' and 'name' columns based on 'Domain Category' co
strawberry_data <- strawberry_data %>%
  # Set 'use' to "FERTILIZER" where 'Domain Category' contains "FERTILIZER", if 'use' is cu
  mutate(
    use = coalesce(use, if_else(str_detect(`Domain Category`, "FERTILIZER"), "FERTILIZER",
    # Set 'name' to text within parentheses after "FERTILIZER" if 'name' is currently NA
    name = coalesce(name, if_else(str_detect(`Domain Category`, "FERTILIZER"), str_extract(
  )

# Filter 'strawberry_data' to create 'strawberry_AREA' with rows containing "AREA GROWN" or
strawberry_AREA <- strawberry_data %>% filter(grepl('AREA GROWN|ORGANIC STATUS',Domain))
# Filter 'strawberry_data' to create 'strawberry_Chemical' with rows containing "CHEMICAL"
strawberry_Chemical<- strawberry_data %>% filter(grepl('CHEMICAL|FERTILIZER',Domain))

# Modify 'strawberry_AREA' to add 'Min' and 'Max' columns based on 'Domain Category'
strawberry_AREA <- strawberry_AREA %>%
  # Set 'Min' to the value before "OR MORE" or extract the first number in parentheses
  mutate(
    Min = case_when(
      str_detect(`Domain Category`, "OR MORE") ~ str_extract(`Domain Category`, "\\d+(?= OR
      TRUE ~ str_extract(`Domain Category`, "(?<=\\()\\d+\\.\\d+|(?<=\\()\\d+")
    ),
    # Set 'Max' to "More" if "OR MORE" is present, otherwise extract the number after "TO"
    Max = case_when(
      str_detect(`Domain Category`, "OR MORE") ~ "More",
      TRUE ~ str_extract(`Domain Category`, "(?<=TO )\\d+\\.\\d+|(?<=TO )\\d+")
    )
  )

strawberry_combined <- bind_rows(strawberry_AREA, strawberry_Chemical)
write.csv(strawberry_combined, "strawberry_combined.csv", row.names = FALSE)

ggplot(strawberry_combined, aes(x = as.numeric(Min))) +
  geom_histogram(binwidth = 5, fill = "lightgreen", color = "black", alpha = 0.7) +
  labs(title = "Distribution of Minimum Area Sizes", x = "Minimum Area Size (acres)", y = '
  theme_minimal()
```

```
Warning: Removed 4206 rows containing non-finite outside the scale range
(`stat_bin()`).
```



Most strawberry acreage is small, with the largest number of plots being between 0 and 5 acres in size. The frequency of occurrence decreases as the size of the smallest plots increases.