MA615 strawberry

2024-10-21

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

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

```r
library(knitr)
library(kableExtra)
```

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

```r
library(stringr)
strawberry<-read.csv("strawberries25_v3.csv")
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)", "~
```

```
sum(strawberry$Domain == "TOTAL")
```

```
## [1] 8105
```

```
sum(strawberry$Domain == "TOTAL")
```

```
## [1] 8105
```

```
state_all <- strawberry |> distinct(State)
state_all1 <- strawberry |> group_by(State) |> count()
```

##Step 2: Remove columns containing only a single value. ##The rationale behind this step is that these columns display the same value across all entries and thus provide no unique insights for data analysis, modeling, or forecasting efforts. Such columns fail to offer any differentiation among observations.

```
drop1<- function(df){
drop <- NULL
for(i in 1:dim(df)[2]){
if((df |> distinct(df[,i]) |> count()) == 1){
drop = c(drop, i)
} }

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

   print("Columns dropped:")
   print(colnames(df)[drop])
   strawberry <- df[, -1*drop]
   }
}
strawberry <- drop1(strawberry)
```

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

```
drop1(strawberry)
```

```
## [1] "none"
```

###Step 3: Analyze the data sources to gain a deeper understanding of the data.

```
calif <- strawberry |> filter(State=="CALIFORNIA")
unique(calif$Program)
```

```
## [1] "CENSUS" "SURVEY"
```

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

The comparison reveals that the following variables in the survey data contain NA values: "Ag.District", "Ag.District.Code", "Country", "Country.ANSI", "CV…". This discrepancy may stem from the nature of surveys, which typically involve more frequent but smaller-scale data collection, as opposed to censuses that are conducted less frequently but encompass a broader data scope, resulting in more exhaustive datasets.

Step 4: Organize column variables.

The data consolidated under the same column (Data.Item) requires segmentation into separate columns, and the introduction of new variables is necessary.

```r
strawberry <- strawberry |>
  separate(
    col = `Data.Item`,
    into = c("Fruit", "Rest"),
    sep = " - ",
    remove = FALSE,
    extra = "merge",
    fill = "right"
  )

# Step 2: split 'Rest' into 'Measure' and 'Bearing_type'
strawberry <- strawberry |>
  separate(
    col = Rest,
    into = c("Measure", "Bearing_type"),
    sep = "(?=(ACRES|WITH))",
    remove = FALSE,
    extra = "merge",
    fill = "left"
  ) |>
  select(-Rest, -Fruit, -Data.Item)
```

Step 5: Convert any exceptional characters in 'VALUE' to NA.

```r
footnotes_v <- strawberry %>%
   filter(!is.na(Value) & !grepl("^[0-9]+(\\.[0-9]+)?(,[0-9]{1,3})*$", Value)) %>%
  distinct(Value)
strawberry <- strawberry %>% mutate(Value = na_if(Value, "(NA)"))
strawberry$Value<-as.numeric(str_replace(strawberry$Value,",",""))
```

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

```r
write.csv(strawberry, file = "cleaned_strawberry_data.csv", row.names = FALSE)
```

```r
library(tidyverse)
library(knitr)
library(kableExtra)
library(stringr)
strawberry<-read.csv("cleaned_strawberry_data.csv")
na_summary <- colSums(is.na(strawberry))
strawberry_clean <- strawberry %>% drop_na(Value)
summary(strawberry_clean$Value)
```

```
##     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       2       4    4526      18   895054
```

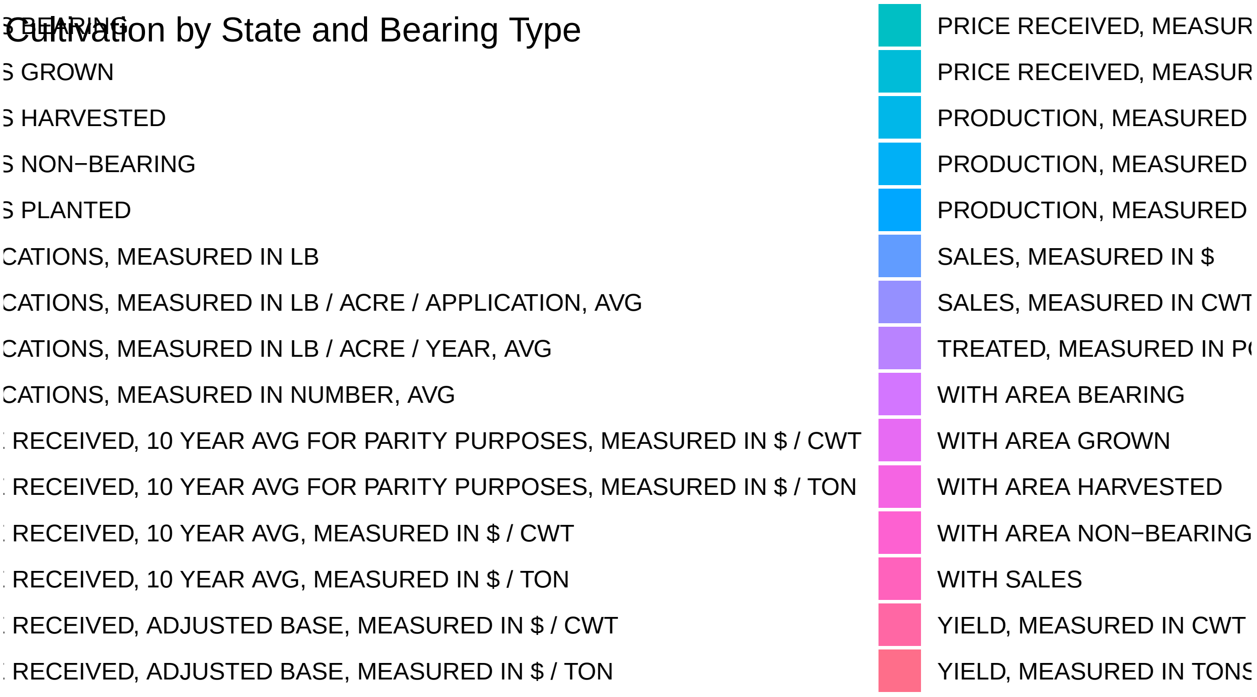
```
state_measure_summary <- strawberry_clean %>%
  group_by(State, Measure, Bearing_type) %>%
  summarise(Total_Value = sum(Value, na.rm = TRUE)) %>%
  arrange(desc(Total_Value))
```

```
## `summarise()` has grouped output by 'State', 'Measure'. You can override using
## the `.groups` argument.
```

```
head(state_measure_summary)
```

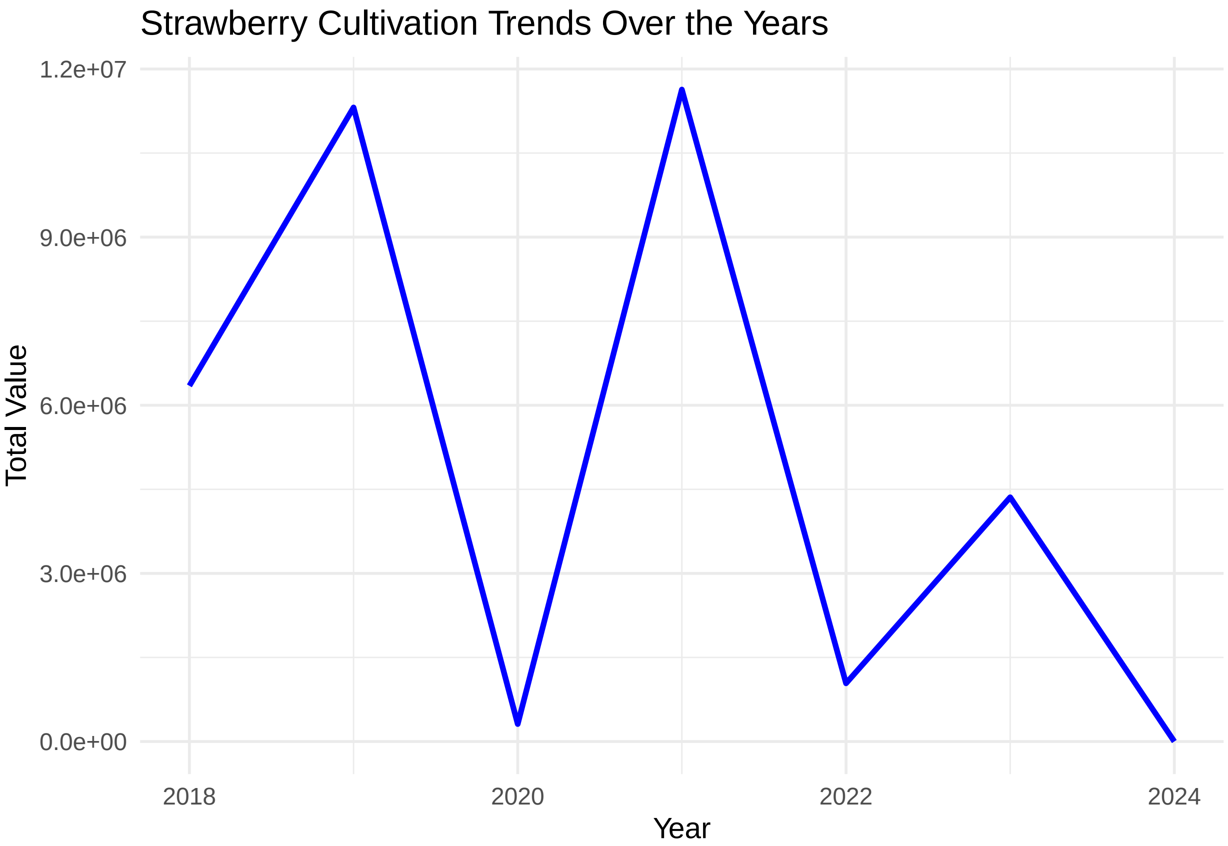
```
## # A tibble: 6 x 4
## # Groups:   State, Measure [6]
##   State      Measure Bearing_type                 Total_Value
##   <chr>      <chr>   <chr>                              <dbl>
## 1 CALIFORNIA <NA>    APPLICATIONS, MEASURED IN LB    10433500
## 2 FLORIDA    <NA>    APPLICATIONS, MEASURED IN LB     4231300
## 3 WASHINGTON <NA>    SALES, MEASURED IN $             2485043
## 4 OREGON     <NA>    SALES, MEASURED IN $             2295766
## 5 VERMONT    <NA>    SALES, MEASURED IN $             1934348
## 6 NEW YORK   <NA>    SALES, MEASURED IN $             1277266
```

```
library(ggplot2)
ggplot(state_measure_summary, aes(x = reorder(State, -Total_Value), y = Total_Value, fill = Bearing_typ
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Strawberry Cultivation by State and Bearing Type", x = "State", y = "Total Value") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



```
yearly_summary <- strawberry_clean %>%
  group_by(Year) %>%
  summarise(Total_Value = sum(Value, na.rm = TRUE))
ggplot(yearly_summary, aes(x = Year, y = Total_Value)) +
  geom_line(color = "blue", size = 1) +
  theme_minimal() +
  labs(title = "Strawberry Cultivation Trends Over the Years", x = "Year", y = "Total Value")
```

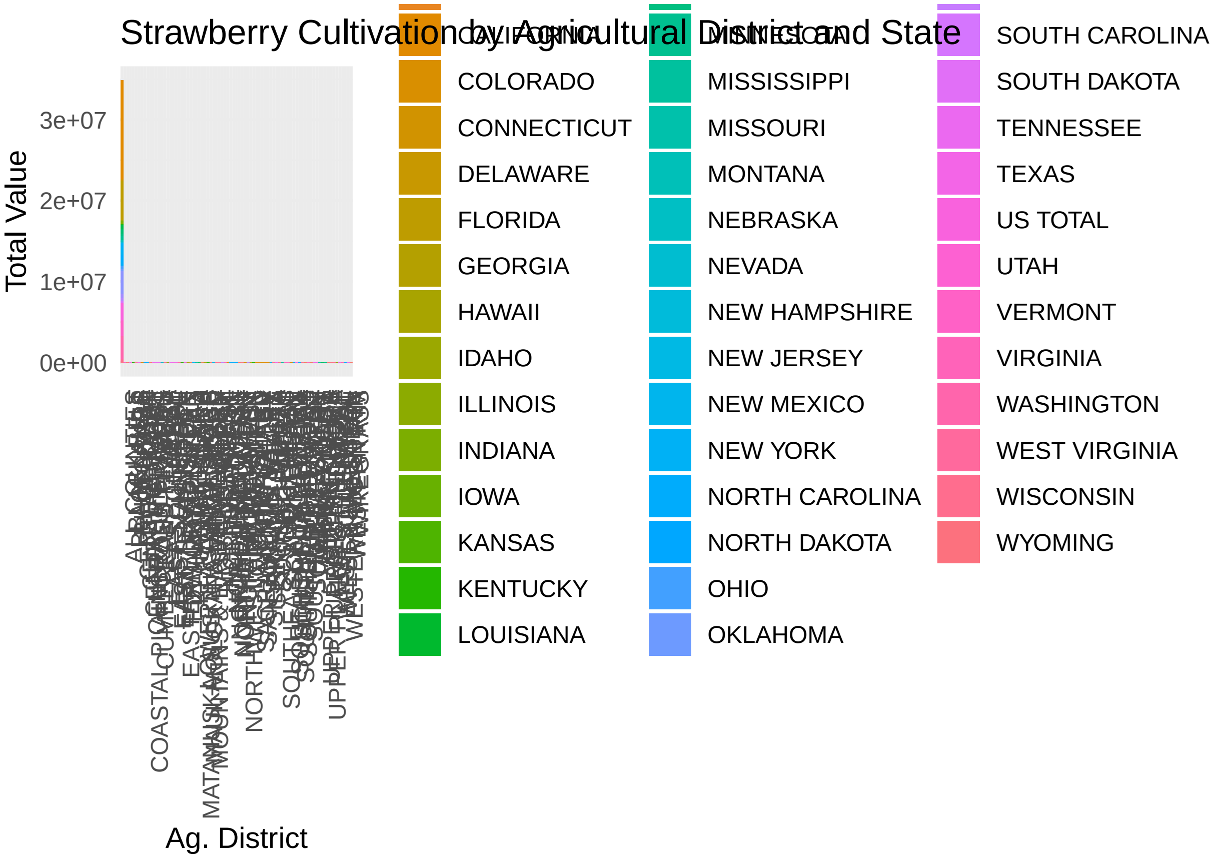
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
district_summary <- strawberry_clean %>%
  group_by(State, Ag.District) %>%
  summarise(Total_Value = sum(Value, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'State'. You can override using the
## `.groups` argument.
```

```
ggplot(district_summary, aes(x = Ag.District, y = Total_Value, fill = State)) +
  geom_bar(stat = "identity") +
  theme_minimal() +
  labs(title = "Strawberry Cultivation by Agricultural District and State", x = "Ag. District", y = "To
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

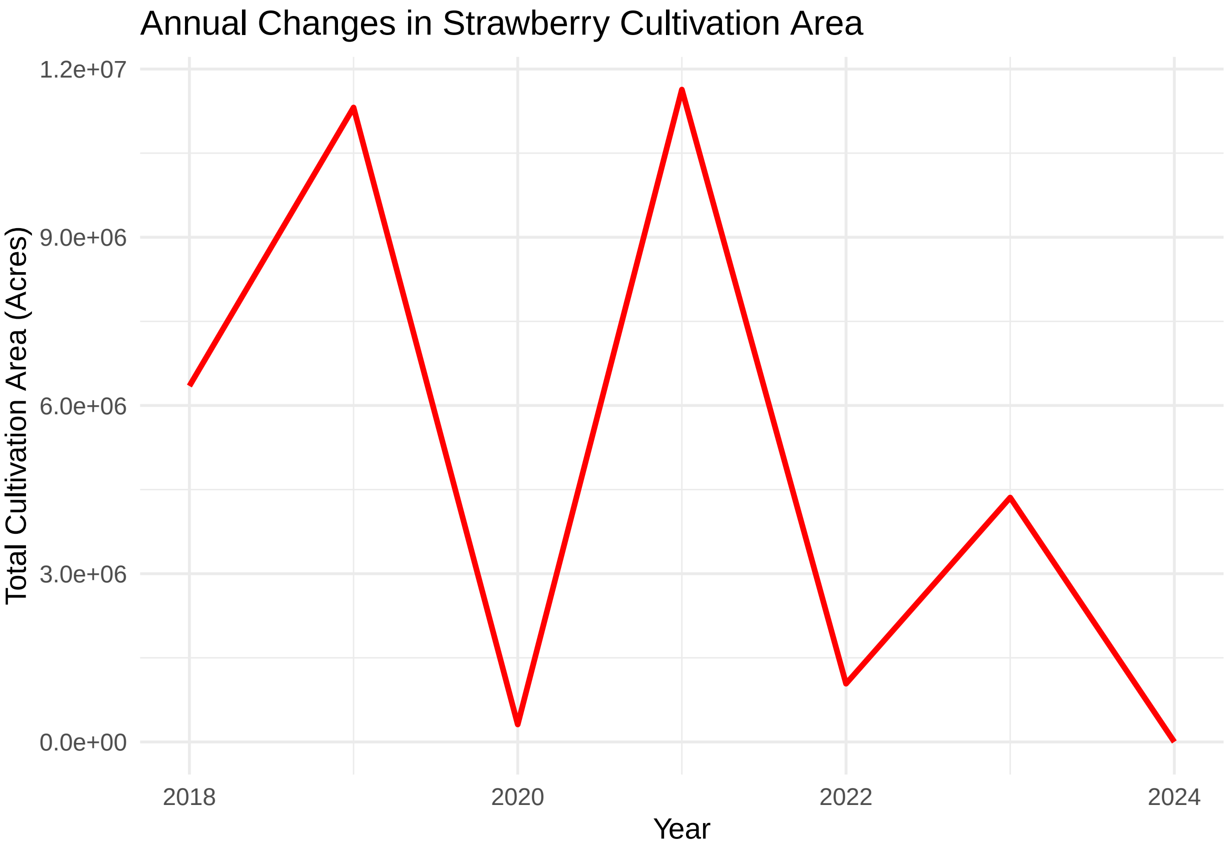


#Conclusion #1. Regional distribution of strawberry planting: As can be seen from the bar chart, there are obvious differences in strawberry planting among different states. Some states have particularly large strawberry planting areas, and the planting characteristics and policy support of these states can be further studied in the future. #2. Changes in planting trends: Strawberry planting area has fluctuated over the past few years. Using the time series graph, we can identify whether there is a cyclical change and further analyze the possible causes, such as climate, market demand, etc. #3. The use of chemical substances: For the use of toxic chemicals, we can see whether the carcinogens listed by WHO are frequently used in strawberry cultivation, which has an important impact on health and the environment.

#New question #Is the trend of strawberry planting area related to climate and policy changes? #Are the differences between different agricultural areas due to natural conditions or differences in growing techniques? #Can climate data or economic data be combined to further analyze factors affecting strawberry cultivation in the future?

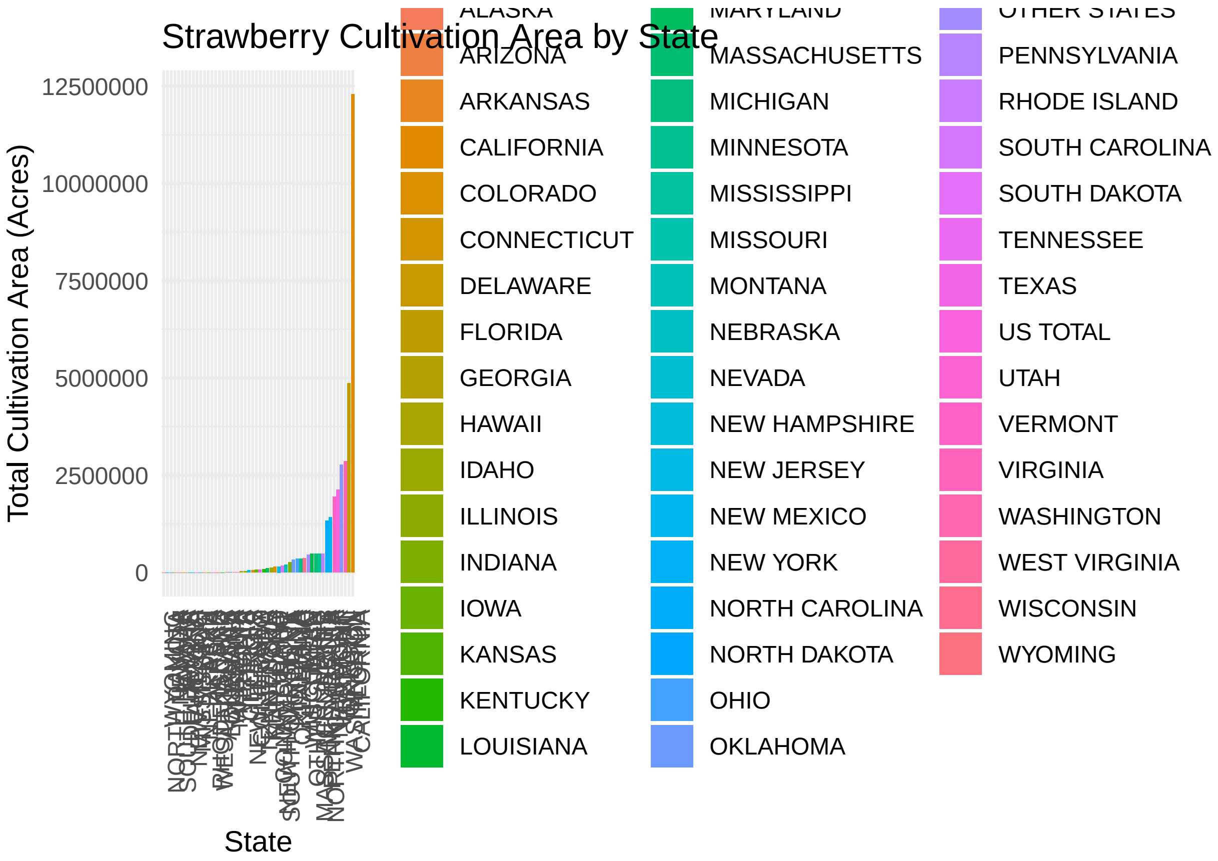
##Annual trend of strawberry planting area

```r
yearly_summary <- strawberry %>%
  group_by(Year) %>%
  summarise(Total_Acres = sum(Value, na.rm = TRUE))
ggplot(yearly_summary, aes(x = Year, y = Total_Acres)) +
  geom_line(color = "red", size = 1) +
  labs(title = "Annual Changes in Strawberry Cultivation Area",
       x = "Year", y = "Total Cultivation Area (Acres)") +
  theme_minimal()
```



##Comparison of strawberry acreage in different states

```r
state_summary <- strawberry %>%
  group_by(State) %>%
  summarise(Total_Acres = sum(Value, na.rm = TRUE))
ggplot(state_summary, aes(x = reorder(State, Total_Acres), y = Total_Acres, fill = State)) +
  geom_bar(stat = "identity") +
  labs(title = "Strawberry Cultivation Area by State",
       x = "State", y = "Total Cultivation Area (Acres)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

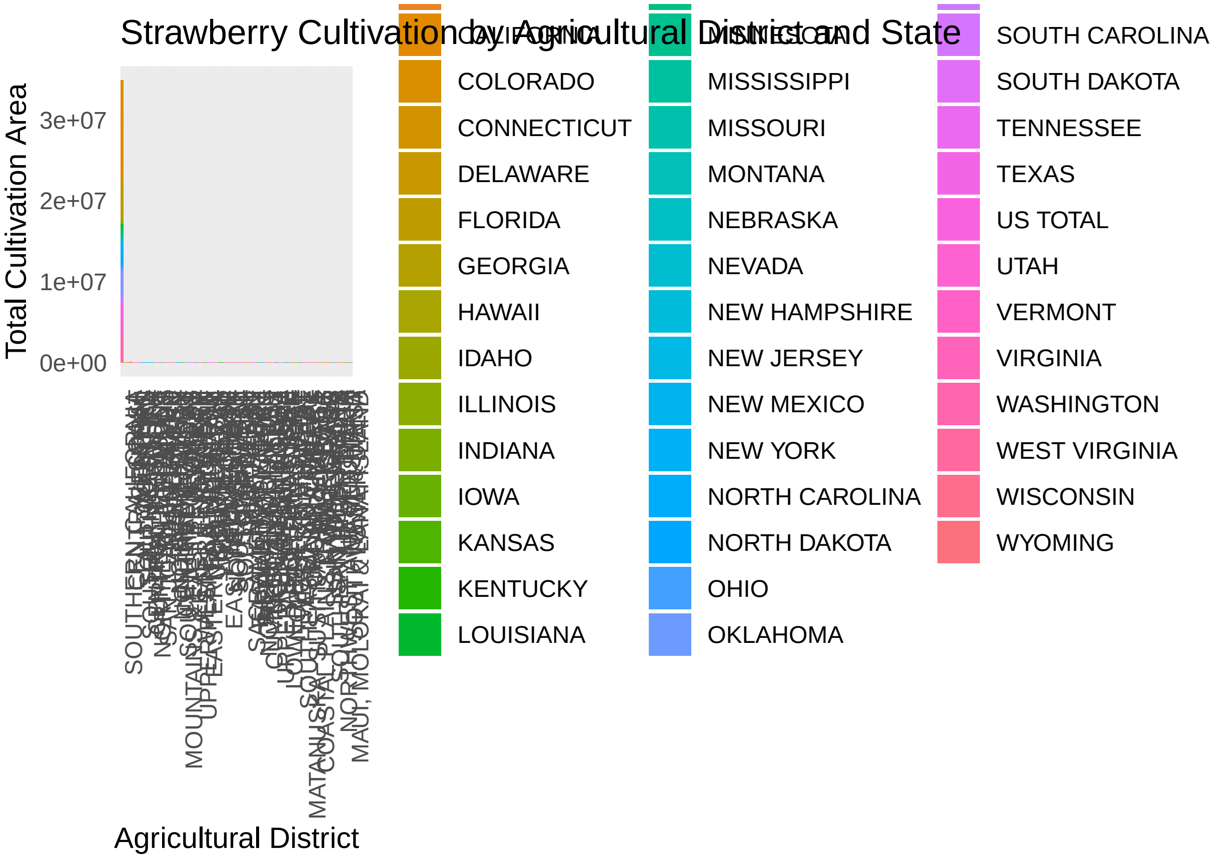


#Relationship between strawberry planting area and specific agricultural area

```
district_summary <- strawberry %>%
  group_by(State, Ag.District) %>%
  summarise(Total_Acres = sum(Value, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'State'. You can override using the
## `.groups` argument.
```

```
ggplot(district_summary, aes(x = reorder(Ag.District, -Total_Acres), y = Total_Acres, fill = State)) +
  geom_bar(stat = "identity") +
  labs(title = "Strawberry Cultivation by Agricultural District and State",
       x = "Agricultural District", y = "Total Cultivation Area") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.