Part2

STREET TREES DISRIBUTION ANALYSIS

Running Code

```
library(tidyverse)
```

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

```
library(stringr)
library(ggplot2)
library(dplyr)
library(stringr)
```

Data Preparation

```
tree_data <- read_csv("/Users/youssoufdiombera/Downloads/Work2/TS3_Raw_tree_data.csv", sh
str(tree_data)
```

```
spc_tbl_ [14,487 × 41] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ DbaseID       : num [1:14487] 1 2 3 4 5 6 7 8 9 10 ...
 $ Region        : chr [1:14487] "InlVal" "InlVal" "InlVal" "InlVal" ...
 $ City          : chr [1:14487] "Modesto, CA" "Modesto, CA" "Modesto, CA" "Modesto, CA"
...
 $ Source        : chr [1:14487] "Motown2.xls: Completed Data" "Motown2.xls: Completed
Data" "Motown2.xls: Completed Data" "Motown2.xls: Completed Data" ...
 $ TreeID        : num [1:14487] 1 2 3 4 5 6 7 8 9 10 ...
 $ Zone          : chr [1:14487] "Nursery" "Nursery" "Nursery" "Nursery" ...
 $ Park/Street   : chr [1:14487] "Nursery" "Nursery" "Nursery" "Nursery" ...
 $ SpCode        : chr [1:14487] "ACSA1" "BEPE" "CESI4" "CICA" ...
 $ ScientificName: chr [1:14487] "Acer saccharinum" "Betula pendula" "Celtis sinensis"
"Cinnamomum camphora" ...
 $ CommonName    : chr [1:14487] "Silver maple" "European white birch" "Chinese
```

```
hackberry" "Camphor tree" ...
 $ TreeType      : chr [1:14487] "BDL" "BDM" "BDL" "BEM" ...
 $ address       : chr [1:14487] "-1" "-1" "-1" "-1" ...
 $ street        : chr [1:14487] "Nursery" "Nursery" "Nursery" "Nursery" ...
 $ side          : chr [1:14487] "-1" "-1" "-1" "-1" ...
 $ cell          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ OnStreet      : chr [1:14487] "-1" "-1" "-1" "-1" ...
 $ FromStreet    : chr [1:14487] "-1" "-1" "-1" "-1" ...
 $ ToStreet      : chr [1:14487] "-1" "-1" "-1" "-1" ...
 $ Age           : num [1:14487] 0 0 0 0 0 0 0 0 0 0 ...
 $ DBH (cm)      : num [1:14487] 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...
 $ TreeHt (m)    : num [1:14487] 2 1.5 1.8 2 2 2 2 2 2 1.6 ...
 $ CrnBase       : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ CrnHt (m)     : num [1:14487] 0.5 0.8 0.6 0.9 0.9 0.8 0.8 0.8 0.8 0.8 ...
 $ CdiaPar (m)   : num [1:14487] 1 0.6 0.7 1 1 0.8 0.8 0.8 1 0.7 ...
 $ CDiaPerp (m)  : num [1:14487] 1 0.6 0.7 1 1 0.8 0.8 0.8 1 0.7 ...
 $ AvgCdia (m)   : num [1:14487] 1 0.6 0.7 1 1 0.8 0.8 0.8 1 0.7 ...
 $ Leaf (m2)     : num [1:14487] 2.5 1.9 2.2 2 2.2 2.2 2.2 2.2 2.1 1.3 ...
 $ Setback       : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ TreeOr        : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ CarShade      : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ LandUse       : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ Shape         : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ WireConf      : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh1          : num [1:14487] 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 2.5 ...
 $ dbh2          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh3          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh4          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh5          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh6          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh7          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ dbh8          : num [1:14487] -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 - attr(*, "spec")=
  .. cols(
  ..   DbaseID = col_double(),
  ..   Region = col_character(),
  ..   City = col_character(),
  ..   Source = col_character(),
  ..   TreeID = col_number(),
  ..   Zone = col_character(),
  ..   `Park/Street` = col_character(),
  ..   SpCode = col_character(),
  ..   ScientificName = col_character(),
  ..   CommonName = col_character(),
  ..   TreeType = col_character(),
  ..   address = col_character(),
  ..   street = col_character(),
  ..   side = col_character(),
  ..   cell = col_double(),
  ..   OnStreet = col_character(),
  ..   FromStreet = col_character(),
```

```
  ..   ToStreet = col_character(),
  ..   Age = col_double(),
  ..   `DBH (cm)` = col_double(),
  ..   `TreeHt (m)` = col_double(),
  ..   CrnBase = col_double(),
  ..   `CrnHt (m)` = col_double(),
  ..   `CdiaPar (m)` = col_double(),
  ..   `CDiaPerp (m)` = col_double(),
  ..   `AvgCdia (m)` = col_double(),
  ..   `Leaf (m2)` = col_double(),
  ..   Setback = col_double(),
  ..   TreeOr = col_double(),
  ..   CarShade = col_double(),
  ..   LandUse = col_double(),
  ..   Shape = col_double(),
  ..   WireConf = col_double(),
  ..   dbh1 = col_double(),
  ..   dbh2 = col_double(),
  ..   dbh3 = col_double(),
  ..   dbh4 = col_double(),
  ..   dbh5 = col_double(),
  ..   dbh6 = col_double(),
  ..   dbh7 = col_double(),
  ..   dbh8 = col_double()
  .. )
 - attr(*, "problems")=<externalptr>
```

```r
glimpse(tree_data)
```

```
Rows: 14,487
Columns: 41
$ DbaseID        <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, …
$ Region         <chr> "InlVal", "InlVal", "InlVal", "InlVal", "InlVal", "InlV…
$ City           <chr> "Modesto, CA", "Modesto, CA", "Modesto, CA", "Modesto, …
$ Source         <chr> "Motown2.xls: Completed Data", "Motown2.xls: Completed …
$ TreeID         <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, …
$ Zone           <chr> "Nursery", "Nursery", "Nursery", "Nursery", "Nursery", …
$ `Park/Street`  <chr> "Nursery", "Nursery", "Nursery", "Nursery", "Nursery", …
$ SpCode         <chr> "ACSA1", "BEPE", "CESI4", "CICA", "FRAN_R", "FREX_H", "…
$ ScientificName <chr> "Acer saccharinum", "Betula pendula", "Celtis sinensis"…
$ CommonName     <chr> "Silver maple", "European white birch", "Chinese hackbe…
$ TreeType       <chr> "BDL", "BDM", "BDL", "BEM", "BDM", "BDL", "BDM", "BDM",…
$ address        <chr> "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "…
$ street         <chr> "Nursery", "Nursery", "Nursery", "Nursery", "Nursery", …
$ side           <chr> "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "…
$ cell           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ OnStreet       <chr> "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "…
$ FromStreet     <chr> "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "…
$ ToStreet       <chr> "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "-1", "…
$ Age            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
```

```
$ `DBH (cm)`     <dbl> 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, …
$ `TreeHt (m)`   <dbl> 2.0, 1.5, 1.8, 2.0, 2.0, 2.0, 2.0, 2.0, 2.0, 1.6, 2.0, …
$ CrnBase        <dbl> -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -…
$ `CrnHt (m)`    <dbl> 0.5, 0.8, 0.6, 0.9, 0.9, 0.8, 0.8, 0.8, 0.8, 0.8, 0.7, …
$ `CdiaPar (m)`  <dbl> 1.0, 0.6, 0.7, 1.0, 1.0, 0.8, 0.8, 0.8, 1.0, 0.7, 1.1, …
$ `CDiaPerp (m)` <dbl> 1.0, 0.6, 0.7, 1.0, 1.0, 0.8, 0.8, 0.8, 1.0, 0.7, 1.1, …
$ `AvgCdia (m)`  <dbl> 1.0, 0.6, 0.7, 1.0, 1.0, 0.8, 0.8, 0.8, 1.0, 0.7, 1.1, …
$ `Leaf (m2)`    <dbl> 2.5, 1.9, 2.2, 2.0, 2.2, 2.2, 2.2, 2.2, 2.1, 1.3, 1.2, …
$ Setback        <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ TreeOr         <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ CarShade       <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ LandUse        <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ Shape          <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ WireConf       <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh1           <dbl> 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, 2.5, …
$ dbh2           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh3           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh4           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh5           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh6           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh7           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
$ dbh8           <dbl> -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1,…
```

```r
colSums(is.na(tree_data))
```

```
      DbaseID         Region           City         Source         TreeID 
            0              0              0              0              0 
         Zone    Park/Street         SpCode ScientificName     CommonName 
            0              0              0              0              0 
     TreeType        address         street           side           cell 
            0              0              0              0              0 
     OnStreet     FromStreet       ToStreet            Age       DBH (cm) 
            0              0              0              0              0 
   TreeHt (m)        CrnBase      CrnHt (m)    CdiaPar (m)   CDiaPerp (m) 
            0              0              0              0              0 
  AvgCdia (m)      Leaf (m2)        Setback         TreeOr       CarShade 
            0              0              0              0              0 
      LandUse          Shape       WireConf           dbh1           dbh2 
            0              0              0              0              0 
         dbh3           dbh4           dbh5           dbh6           dbh7 
            0              0              0              0              0 
         dbh8 
            0 
```

##Question 1 #How many records are there in each state (include a table or bar plot)?

```r
city_state_data <- tree_data %>%
  mutate(
    State = str_extract(City, "[A-Z]{2}$")
  ) %>%
```

```
  select(City, State) %>%
  distinct()

print(city_state_data, n = Inf)
```

```
# A tibble: 17 × 2
   City              State
   <chr>             <chr>
 1 Modesto, CA       CA
 2 Santa Monica, CA CA
 3 Claremont, CA     CA
 4 Berkeley, CA      CA
 5 Glendale, AZ      AZ
 6 Fort Collins, CO CO
 7 Minneapolis, MN   MN
 8 Indianapolis, IN IN
 9 Queens, NY        NY
10 Boise, ID         ID
11 Albuquerque, NM   NM
12 Honolulu, HI      HI
13 Charleston, SC    SC
14 Charlotte, NC     NC
15 Orlando, FL       FL
16 Longview, WA      WA
17 Sacramento, CA    CA
```

```
colnames(tree_data)
```

```
 [1] "DbaseID"        "Region"         "City"           "Source"
 [5] "TreeID"         "Zone"           "Park/Street"    "SpCode"
 [9] "ScientificName" "CommonName"     "TreeType"       "address"
[13] "street"         "side"           "cell"           "OnStreet"
[17] "FromStreet"     "ToStreet"       "Age"            "DBH (cm)"
[21] "TreeHt (m)"     "CrnBase"        "CrnHt (m)"      "CdiaPar (m)"
[25] "CDiaPerp (m)"   "AvgCdia (m)"    "Leaf (m2)"      "Setback"
[29] "TreeOr"         "CarShade"       "LandUse"        "Shape"
[33] "WireConf"       "dbh1"           "dbh2"           "dbh3"
[37] "dbh4"           "dbh5"           "dbh6"           "dbh7"
[41] "dbh8"
```

```
merged_data <- tree_data %>%
  left_join(city_state_data, by = "City")

state_counts <- merged_data %>%
  filter(!is.na(State)) %>% # Ensure valid State entries
  group_by(State) %>%
  summarise(RecordCount = n(), .groups = "drop") %>%
  arrange(desc(RecordCount))
```

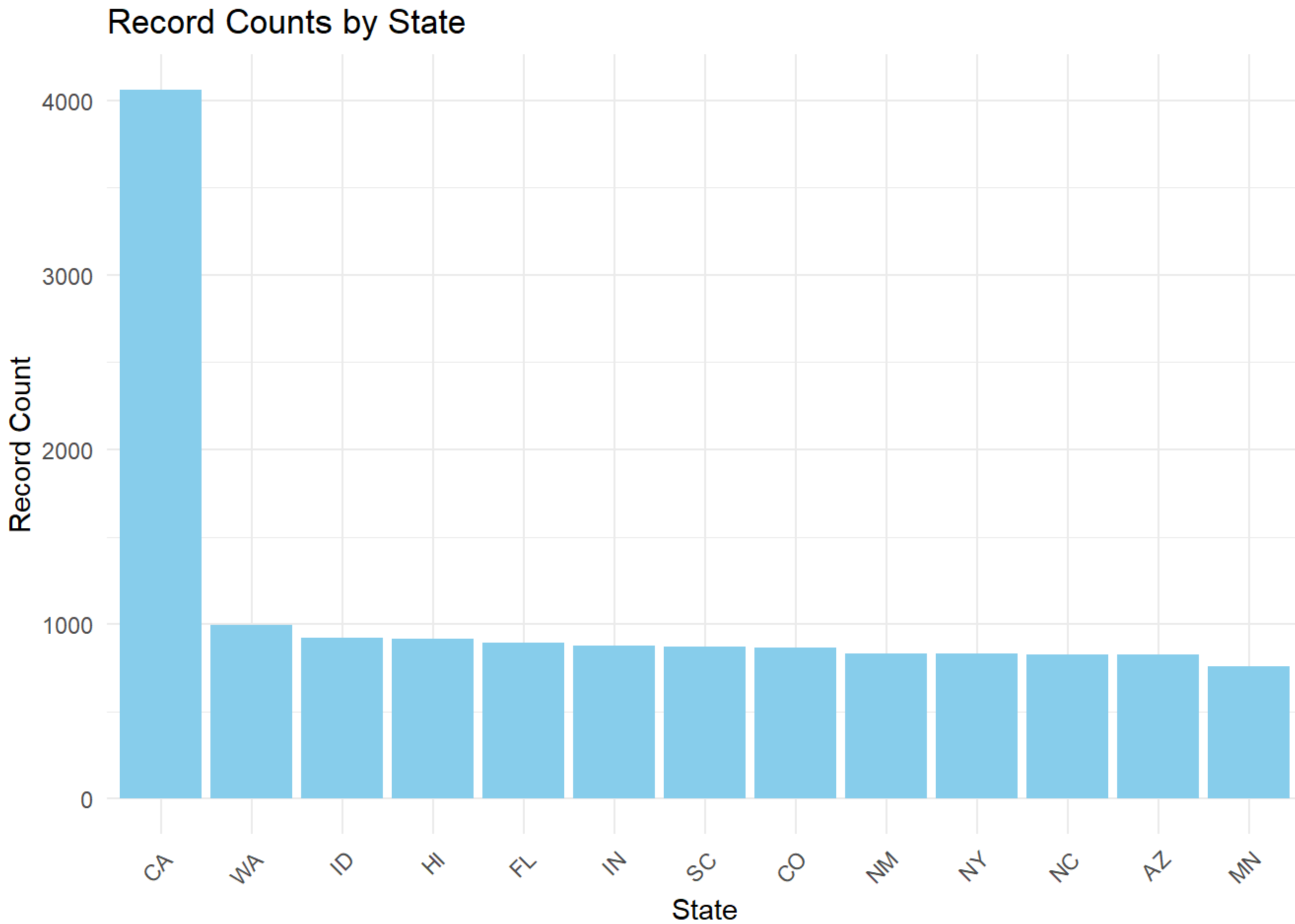
```r
# Print the results
print("Record Counts by State:")
```

```
[1] "Record Counts by State:"
```

```r
print(state_counts, n = Inf)
```

```
# A tibble: 13 × 2
   State RecordCount
   <chr>       <int>
 1 CA           4062
 2 WA            994
 3 ID            923
 4 HI            918
 5 FL            895
 6 IN            877
 7 SC            872
 8 CO            867
 9 NM            833
10 NY            831
11 NC            828
12 AZ            827
13 MN            760
```

```r
ggplot(state_counts, aes(x = reorder(State, -RecordCount), y = RecordCount)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  theme_minimal() +
  labs(
    title = "Record Counts by State",
    x = "State",
    y = "Record Count"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Question 2: What cities did they collect data from in North and South Carolina? [3 points]

```
nc_sc_data <- merged_data %>%
  filter(State %in% c("NC", "SC"))


cities <- nc_sc_data %>%
  distinct(City) %>%
  pull(City)


print(cities)
```

```
[1] "Charleston, SC" "Charlotte, NC"
```

Therefore the cities that they collected data from are Charleston and Charlotte.

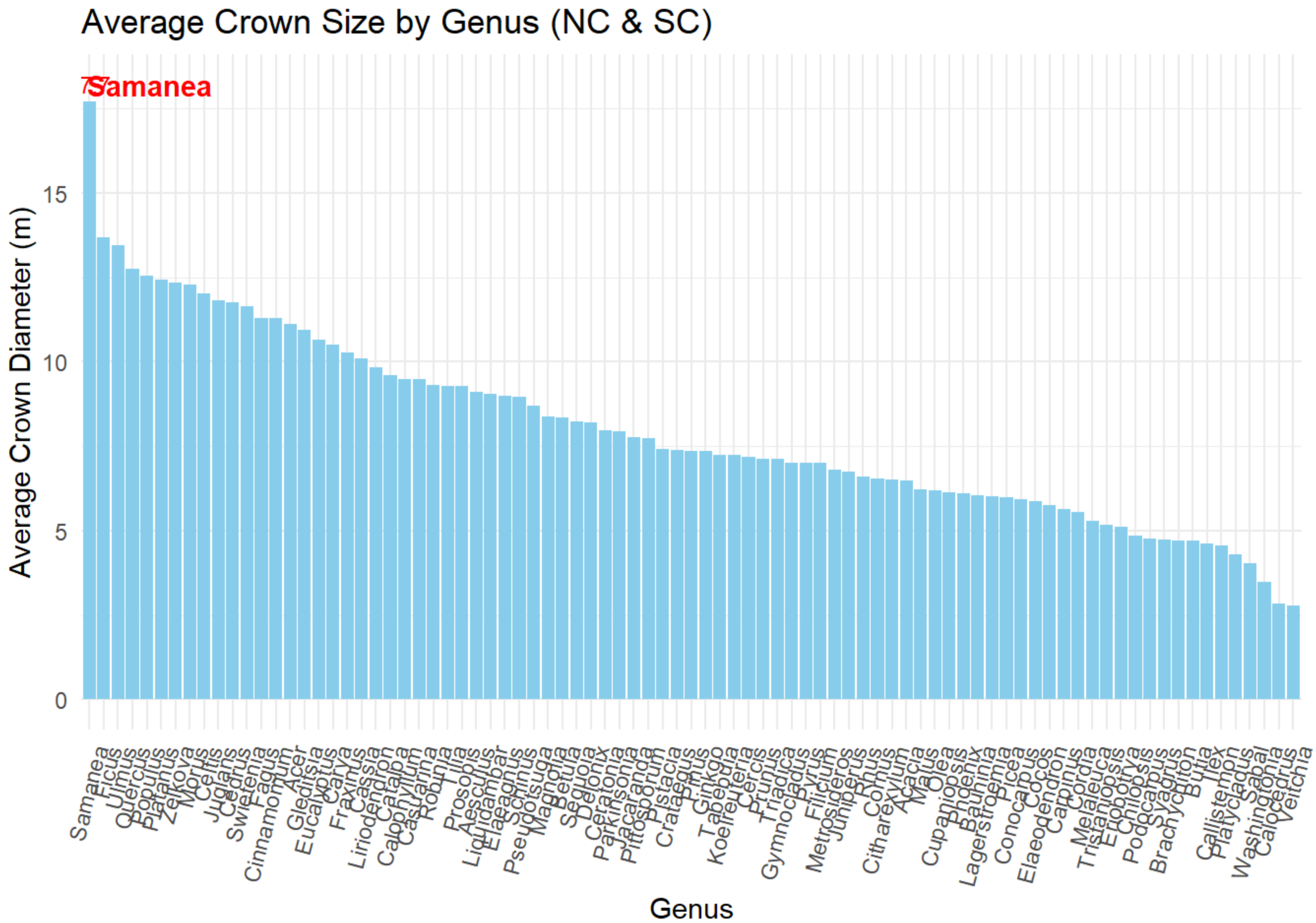
##Question 3: What genus of trees has the largest crown diameter in North and South Carolina? [3 points]

```
nc_sc_data <- merged_data %>%
  mutate(Genus = str_extract(ScientificName, "^[^ ]+"))

genus_crown <- nc_sc_data %>%
  group_by(Genus) %>%
  summarise(AverageCrown = mean(`AvgCdia (m)`, na.rm = TRUE)) %>%
  arrange(desc(AverageCrown))
largest_crown_genus <- genus_crown %>%
  slice_max(AverageCrown, n = 1)
print(largest_crown_genus)
```

```
# A tibble: 1 × 2
  Genus   AverageCrown
  <chr>          <dbl>
1 Samanea         17.7
```

```
ggplot(genus_crown, aes(x = reorder(Genus, -AverageCrown), y = AverageCrown)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  geom_text(
    aes(label = ifelse(Genus == largest_crown_genus$Genus, round(AverageCrown, 2), "")),
    vjust = -0.5,
    color = "red",
    size = 3
  ) +
  theme_minimal() +
  labs(
    title = "Average Crown Size by Genus (NC & SC)",
    x = "Genus",
    y = "Average Crown Diameter (m)"
  ) +
  theme(axis.text.x = element_text(angle = 75, hjust = 1)) +
  annotate(
    "text",
    x = largest_crown_genus$Genus,
    y = largest_crown_genus$AverageCrown + 0.5,
    label = paste("Largest:", largest_crown_genus$Genus),
    color = "red",
    fontface = "bold"
  )
```



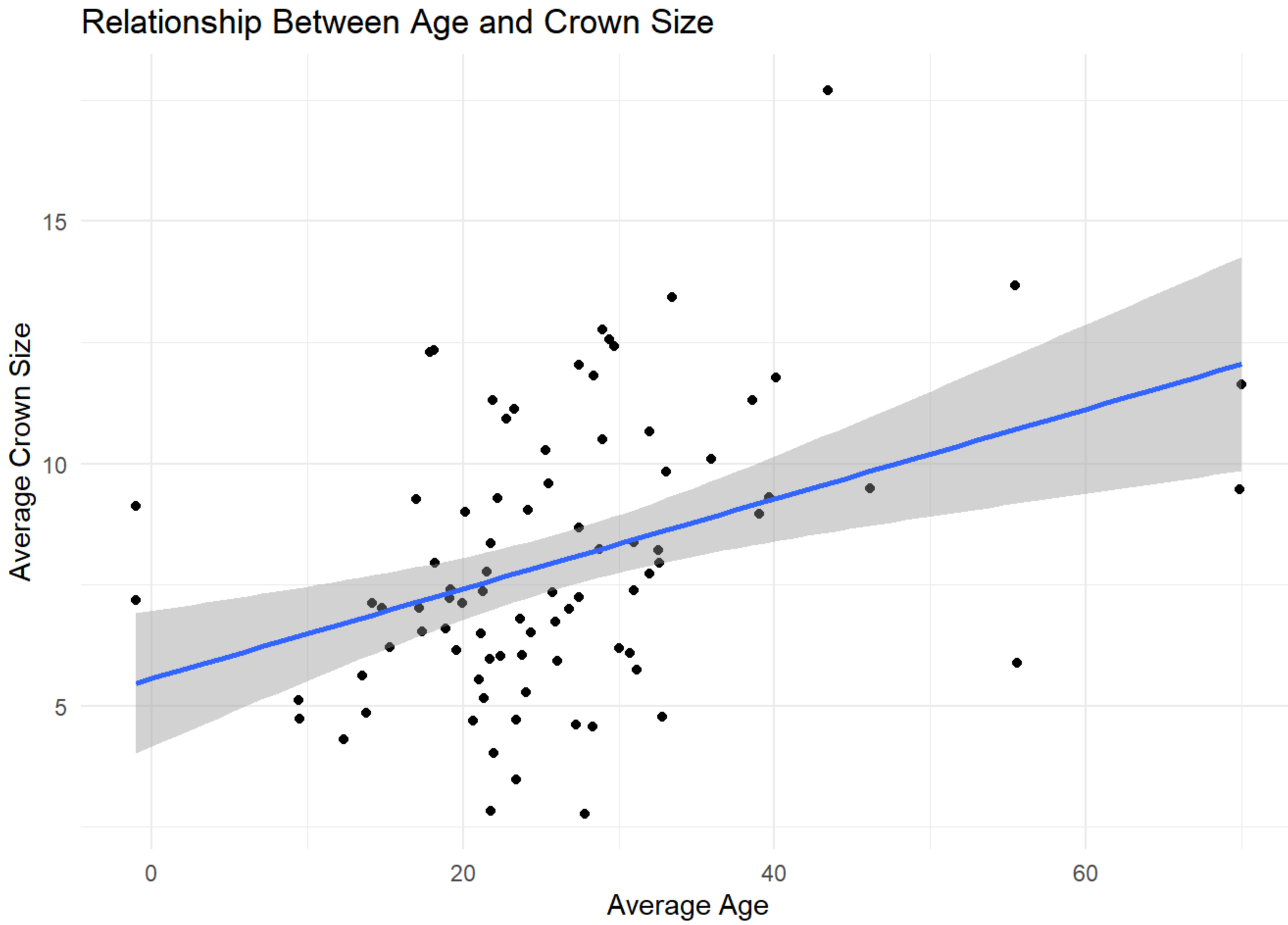
From the above analysis its evident that the Samanea Species has the largest crown diameter of 17.704

##Extra Credit #1 Older trees, of course, have larger crowns. Are there differences in the average age of the different genera of trees in the dataset? Might this explain the results of the previous question? [1 point]

```
age_analysis <- nc_sc_data %>%
  group_by(Genus) %>%
  summarise(AverageAge = mean(Age, na.rm = TRUE), AverageCrown = mean(`AvgCdia (m)`, na.r
  arrange(desc(AverageCrown))

ggplot(age_analysis, aes(x = AverageAge, y = AverageCrown)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(title = "Relationship Between Age and Crown Size", x = "Average Age", y = "Average
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```



Explanation: From the above analysis it can be noted that the regression plot slopes upward which indicates that older trees tend to have larger crowns on average. This also indicates that there has been a large variance on the crown size and this might be due to factors such as growing conditions, and environmental stresses. #Recommending the Genus that produces the largest Crown Quickly

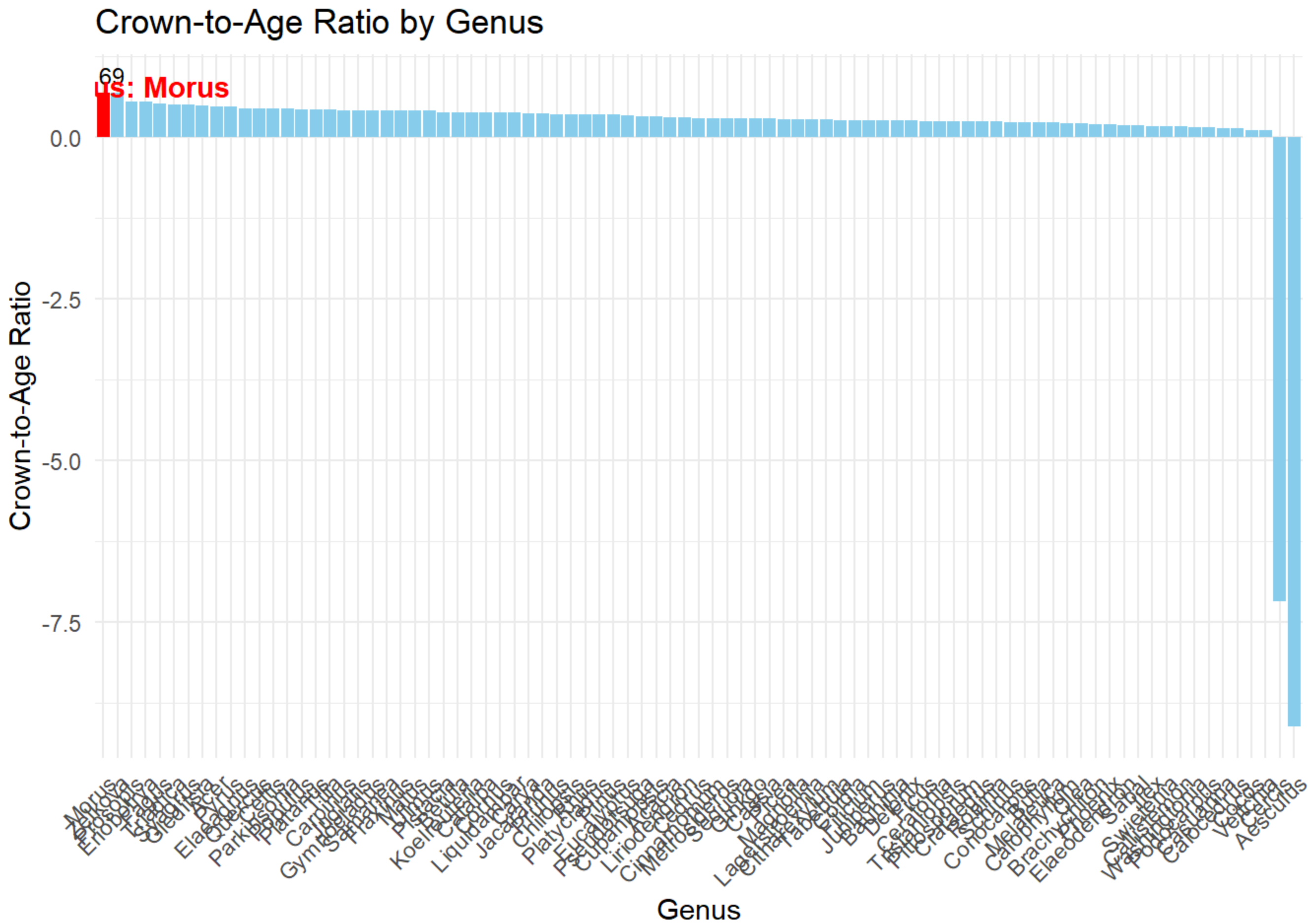
```
merged_data <- merged_data %>%
  mutate(Genus = str_extract(ScientificName, "^[^ ]+"))
genus_analysis <- merged_data %>%
  group_by(Genus) %>%
  summarise(
    AverageCrown = mean(`AvgCdia (m)`, na.rm = TRUE),
    AverageAge = mean(Age, na.rm = TRUE)
  ) %>%
  filter(!is.na(AverageCrown), !is.na(AverageAge))
genus_analysis <- genus_analysis %>%
  mutate(CrownToAgeRatio = AverageCrown / AverageAge) %>%
  arrange(desc(CrownToAgeRatio))

# View the top candidate
top_genus <- genus_analysis %>%
  slice_max(CrownToAgeRatio, n = 1)

print(top_genus)
```

```
# A tibble: 1 × 4
  Genus AverageCrown AverageAge CrownToAgeRatio
  <chr>        <dbl>      <dbl>           <dbl>
1 Morus         12.3       17.9           0.688
```

```
ggplot(genus_analysis, aes(x = reorder(Genus, -CrownToAgeRatio), y = CrownToAgeRatio)) +
  geom_bar(stat = "identity", fill = ifelse(genus_analysis$Genus == top_genus$Genus, "red
  theme_minimal() +
  labs(
    title = "Crown-to-Age Ratio by Genus",
    x = "Genus",
    y = "Crown-to-Age Ratio"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  geom_text(
    aes(label = ifelse(Genus == top_genus$Genus, round(CrownToAgeRatio, 2), "")),
    vjust = -0.5,
    color = "black",
    size = 3
  ) +
  annotate(
    "text",
    x = which(genus_analysis$Genus == top_genus$Genus),
    y = top_genus$CrownToAgeRatio + 0.1,
    label = paste("Top Genus:", top_genus$Genus),
    color = "red",
    fontface = "bold"
  )
```



#Explanation From the above analysis the data was prepared by filtering only the NC and SC states and we extracted the genus from the ScientificName Column. From the date we were able to calculate th Age and the AvgCdia (m). From here we were able to get the genera that produces the large crowns quickly. Therefore the recommended genus was Morus.

#2. Species Analysis

```
nc_sc_data <- nc_sc_data %>%
  mutate(Species = str_extract(ScientificName, "(?<= )[a-z]+"))

species_count <- nc_sc_data %>%
  group_by(Genus) %>%
  summarise(SpeciesCount = n_distinct(Species)) %>%
  arrange(desc(SpeciesCount))
print(species_count)
```

```
# A tibble: 85 × 2
   Genus      SpeciesCount
   <chr>             <int>
 1 Pinus                15
 2 Quercus              12
 3 Fraxinus              8
 4 Acer                  7
 5 Populus               5
```

```
 6 Prunus              5
 7 Eucalyptus          4
 8 Ulmus               4
 9 Acacia              3
10 Celtis              3
# ℹ 75 more rows
```

```r
ggplot(species_count, aes(x = reorder(Genus, -SpeciesCount), y = SpeciesCount)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  theme_minimal() +
  labs(
    title = "Species Count by Genus (NC & SC)",
    x = "Genus",
    y = "Number of Distinct Species"
  ) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_text(aes(label = SpeciesCount), vjust = -0.5, size = 3)
```