# FRUIT PRICES

# IN

# INTRODUCTION TO DATA SCIENCE

## Members Names and Role

| Name | ID | Role |
|---|---|---|
| 1. Ahmed Taha Mohamed Mohamed | 22010315 | Report |
| 2. Moaz Moustafa Abdelhamid Moustafa | 22010273 | GUI |
| 3.Ibrahim Gamil Abdelrahman Ahmed | 22010009 | GUI |
| 3. Yasser Ashraf Mohamed Gaber | 22010409 | Data Cleaning,K means ,Decision Tree |
| 5.Hamza Hussein Yousef omran | 22011501 | Visualization |
| 6.Karim Mohamed Samy Abo Shady | 22010378 | Report |

# 1. Introduction:

This dataset provides information on various fruits, their forms (such as fresh, canned, dried, frozen, or in juice form), retail prices, retail price units, yield, cup equivalent size, cup equivalent unit, and cup equivalent price. Let's break down the key components of the dataset:

1. **Fruit and Form:**
    - The dataset covers a variety of fruits, including apples, apricots, bananas, berries, blackberries, blueberries, cantaloupe, cherries, clementines, cranberries, dates, figs, grapes, honeydew, kiwi, mangoes, nectarines, oranges, papaya, peaches, pears, pineapple, plums, pomegranate, raspberries, strawberries, and watermelon.
    - Fruits are available in different forms such as fresh, canned, dried, frozen, or as juice.
2. **Retail Price and Unit:**
    - Retail prices are provided per pound for most fruits.
    - The retail price unit is specified for each fruit, indicating whether the price is per pound or per pint.
3. **Yield:**
    - The yield represents the proportion of the fruit that is usable or edible after any processing or preparation.
4. **Cup Equivalent Size and Unit:**
    - Cup equivalent size denotes the size of the edible portion of the fruit in terms of cups.
    - Cup equivalent unit specifies whether the size is in pounds or fluid ounces.
5. **Cup Equivalent Price:**
    - Cup equivalent price represents the cost of the edible portion of the fruit in terms of cups.

Idea and objective : given one fruit I need to analyze its data to compare it with the entire data in retail price and cup equivalent price etc.

Taking this data then clean it from duplicates and na values and zeros then use unsupervised technique

k means and supervised one decision tree then visualize it using plots to see which is most sold from this fruits etc.

# 2.Methodologies used:

```
#use readxl//install.packages("readxl")
#install.packages("readxl") # to install the package that read the data in excel file.xlsx
library(readxl)  to use it
#import the excel file to a data frame.
fruit_prices<read_excel("D:\\1.A.Semester3\\Intro_to_data_science\\project\\Fruit Prices 2020 .xlsx")
#print the data frame,using print method with n paramater to read all rows in the file.
print(fruit_prices,n=63)

#Data Cleaning
#1- the sum of duplicated rows are in your data
if(sum(duplicated(fruit_prices))==0){
  paste("No Dublicated Values ",sum(duplicated(fruit_prices)))
}else{
  paste("Number of duplicated rows :",sum(duplicated(fruit_prices)))
}

#2-which rows have duplicated values
duplicated(fruit_prices)

#3-Select only all distinct rows//install.packages("dplyr") to install the package that Select only all distinct rows
#install.packages("dplyr")
library(dplyr)
print(distinct(fruit_prices),n=63)
#4-sum of NA values
if(sum(is.na(fruit_prices)==0)){
  paste("There aren't NA Values",sum(is.na(fruit_prices)))
}else{
  paste("Number of NA values :",sum(is.na(fruit_prices)))
}
```

```
#unsupervised technique -> k-means we will use it on retailprice and
CupEquivalentPrice columns to identify the fair prices and expensive ones

fruit_prices_k<-fruit_prices[,c(3,8)]
#print(fruit_prices_k,n=63)
Kmean_clustering_fruits<-kmeans(fruit_prices_k,centers = 2)
Kmean_clustering_fruits

#Supervised learning technique ->Decision Tree used to identifyall probabilities that
form be with more than one type or it be one type.
#columns from 1 to 10
fruit_prices_tree<-fruit_prices[1:10,]
fruit_prices_tree
#use rpart to show the plot of decision tree
library(rpart)
tree<-rpart(Form ~ Fruit + Yield + RetailPriceUnit,
        data =fruit_prices_tree , minsplit=2)
tree



# this package that allow you to do UI
#install.packages("shiny")
library(shiny)
library(rpart.plot)

ui <- fluidPage(

  # App title ----
  titlePanel("Data Science Project"),

  # Sidebar layout with input and output definitions ----
  sidebarLayout(

    # Buttons that each shows a specific plot ----
    sidebarPanel(

      # Buttons to display specific graphs ----
```

```
    actionButton(inputId = "boxplotButton", label = "Boxplot"),
    actionButton(inputId = "barplotCupPricesButton", label = "Barplot - Cup Prices"),
    actionButton(inputId = "barplotCupSizesButton", label = "Barplot - Cup Sizes"),
    actionButton(inputId = "plotPricesVsYeildButton", label = "Plot - Prices vs
Yeild"),
    actionButton(inputId = "piePlotButton", label = "Pie Plot - Form Distribution"),
    actionButton(inputId = "barplotAveragePricesButton", label = "Barplot - Average
Retail Prices"),
    actionButton(inputId = "treePlotButton", label = "Decision Tree")

  ),

  # Main panel for displaying outputs ----
  mainPanel(

    # Output: Histogram ----
    plotOutput(outputId = "distPlot")

  )
 )
)

server <- function(input, output) {

 # Histogram of the Old Faithful Geyser Data ----
 # with requested number of bins

 observeEvent(input$boxplotButton, {
  output$distPlot <- renderPlot({
   boxplot_prices <- boxplot(x = fruit_prices$RetailPrice,
                             xlab = "Fruits",
                             main = "Compare Fruits Prices")
   boxplot_prices
  })
 })

 observeEvent(input$barplotCupPricesButton, {
```

```
  output$distPlot <- renderPlot({
 #this data compares cup equivalent size for each fruit
    barpolot_cupprices <- barplot(height = fruit_prices$CupEquivalentPrice,
                          names.arg = fruit_prices$Fruit,
                          xlab = "Fruits",
                          ylab = "Cup Price",
                          col = "cyan",
                          main = "Compare Fruits Cup Prices")
    barpolot_cupprices
  })
 })

 observeEvent(input$barplotCupSizesButton, {
  output$distPlot <- renderPlot({
#this compare cup equivalent size for each fruit
    barplot_cupsizes <- barplot(height = fruit_prices$CupEquivalentSize,
                        names.arg = fruit_prices$Fruit,
                        xlab = "Fruits",
                        ylab = "Cup Size",
                        col = "orange",
                        main = "Compare Fruits Cup Sizes")
    barplot_cupsizes
  })
 })

 observeEvent(input$plotPricesVsYeildButton, {
  output$distPlot <- renderPlot({
#   scatter plot compare prices vs yeild
  plot_prices_vs_yeild <- plot(x = fruit_prices$RetailPrice, y = fruit_prices$Yield,
                        xlab = "Price", ylab = "Yeild", main = "Price Vs Yeild", col = "red")
    plot_prices_vs_yeild
  })
 })

 observeEvent(input$piePlotButton, {
  output$distPlot <- renderPlot({
#this code take the form column
```

```
    form_counts <- table(fruit_prices$Form)
#this line defines the colors of the parts
    custom_colors <- c("cyan", "#3477eb", "#34eb77", "orange","#e05404")


 #the cex.main iStock change the size of font to 1.2 times
    pie_plot <- pie(form_counts, labels = names(form_counts), col = custom_colors,
                main = "Form Distribution of Fruits", cex.main = 1.2)
    pie_plot
   })
  })

  observeEvent(input$barplotAveragePricesButton, {
   output$distPlot <- renderPlot(
#this code take retail price column and fruit form column and calculate the mean
#price for each form
    average_prices <- tapply(fruit_prices$RetailPrice, fruit_prices$Form, mean)
 #define colors
    bar_colors <- c("lightblue", "skyblue", "darkcyan", "lightgreen",
"mediumseagreen")
# here the ylim takes the lower limit and the upper limit for y axis
    barplot_average_prices <- barplot(average_prices, col = bar_colors,
                              main = "Average Retail Prices by Form",
                              xlab = "Form", ylab = "Average Retail Price",
                              border = "black", ylim = c(0, max(average_prices) + 0.5))
    barplot_average_prices
   })
  })

  observeEvent(input$treePlotButton, {
   output$distPlot <- renderPlot({
    tree_plot <- rpart.plot(tree)
   })
  })
}

# Run the Shiny app
```

```
shinyApp(ui = ui, server = server)
```

# 3.Challenges in the dataset:

While the dataset provides valuable information about various fruits and their characteristics, there are some challenges and limitations that should be considered:

1. **Inconsistencies in Units:**
    - The dataset contains information on retail prices, yields, cup equivalents, etc., with varying units such as pounds and fluid ounces. Ensuring consistent units across the dataset is crucial for accurate analysis and comparisons.
2. **Limited Scope:**
    - The dataset may not cover all possible forms or varieties of fruits, potentially limiting its representativeness. For a more comprehensive analysis, additional fruits or forms could be included.
3. **Processing Methods:**
    - The dataset mentions forms like fresh, canned, dried, and frozen, but it lacks detailed information on specific processing methods. Understanding how the fruits are processed could be important for nutritional analysis and comparisons.
4. **Categorical Information:**
    - The dataset primarily focuses on numerical data, such as prices and quantities, but lacks detailed categorical information about the characteristics of each fruit (e.g., organic, conventional, variety).
5. **Limited Geographical Representation:**

- The dataset doesn't specify the geographical origin of the fruits. Prices and availability can vary significantly based on the region, climate, and season, which could impact the analysis.

6. **Temporal Aspect:**
    - The dataset is static and doesn't provide information about the temporal aspect of the data. Prices and availability of fruits can fluctuate over time due to seasonal variations and market conditions.
7. **Normalization of Prices:**
    - The retail prices are given in various units (e.g., per pound or per pint). Normalizing the prices to a consistent unit could be necessary for accurate comparisons.
8. **Limited Nutritional Information:**
    - While the dataset includes some nutritional information indirectly through yield and cup equivalents, a more detailed nutritional breakdown (e.g., calories, vitamins) would enhance its usefulness.
9. **No Consumer Demand Data:**
    - The dataset does not include information on consumer demand or preferences, which could be valuable for understanding market dynamics and trends.

# 4.Interpretations of the results

- "No Dublicated Values  0" : NO row is repeated in the data
- [1] "There aren't NA Values 0" : no NA values in the data
- K-means clustering with 2 clusters of sizes 13, 49

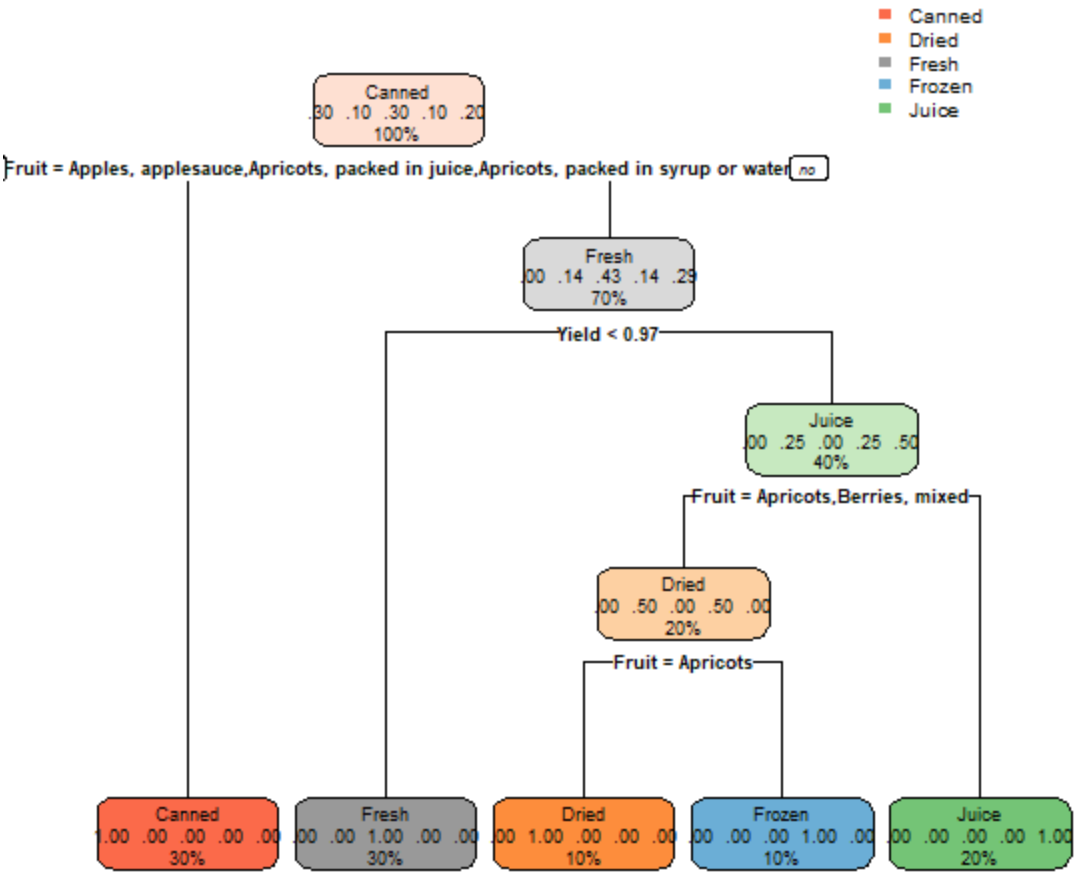
```
Cluster means:
RetailPrice CupEquivalentPrice
1    5.972085           1.381623
2    1.725673           0.797200

Clustering vector:
[1] 2 2 2 2 2 2 2 1 2 2 1 2 1 2 2 2 1 2 1 1 1 2 2 2 2 2 2 2 2
2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2
[54] 1 2 2 2 1 1 2 2 2

Within cluster sum of squares by cluster:
```

```
   [1] 39.28580 47.61082
   (between_SS / total_SS =  68.5 %)

   Available components:

   [1] "cluster"      "centers"      "totss"        "withinss"
   "tot.withinss" "betweenss"    "size"
   [8] "iter"         "ifault"
```

: this is k means clustering to two specific columns retail pric
e , cup equivalent price

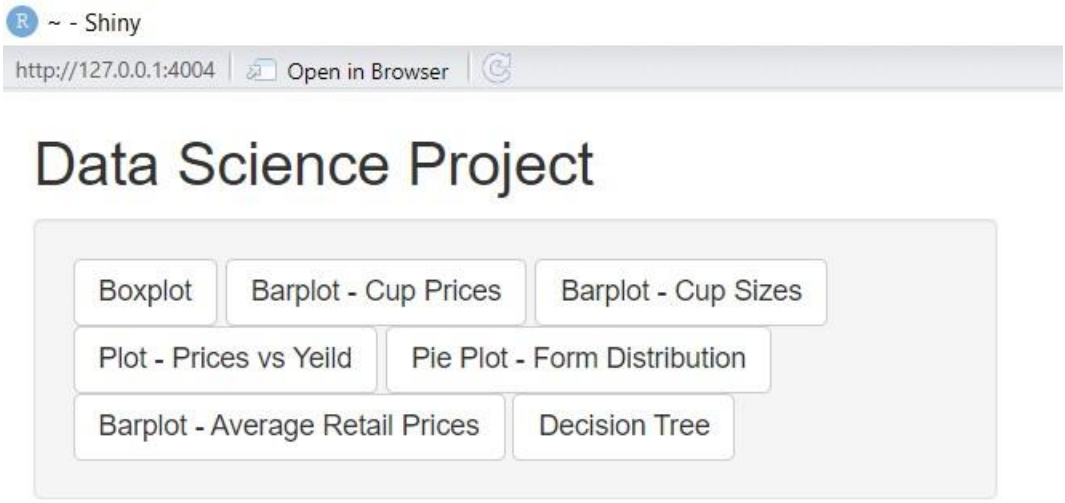
- n= 10

```
node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 10 7 Canned (0.3 0.1 0.3 0.1 0.2)
   2) Fruit=Apples, applesauce,Apricots, packed in
juice,Apricots, packed in syrup or water 3 0 Canned (1 0 0 0 0)
*
   3) Fruit=Apples,Apples, frozen concentrate,Apples, ready-to-
drink,Apricots,Bananas,Berries, mixed 7 4 Fresh (0 0.14 0.43
0.14 0.29)
     6) Yield< 0.965 3 0 Fresh (0 0 1 0 0) *
     7) Yield>=0.965 4 2 Juice (0 0.25 0 0.25 0.5)
      14) Fruit=Apricots,Berries, mixed 2 1 Dried (0 0.5 0 0.5
0)
        28) Fruit=Apricots 1 0 Dried (0 1 0 0 0) *
        29) Fruit=Berries, mixed 1 0 Frozen (0 0 0 1 0) *
      15) Fruit=Apples, frozen concentrate,Apples, ready-to-
drink 2 0 Juice (0 0 0 0 1) *
```

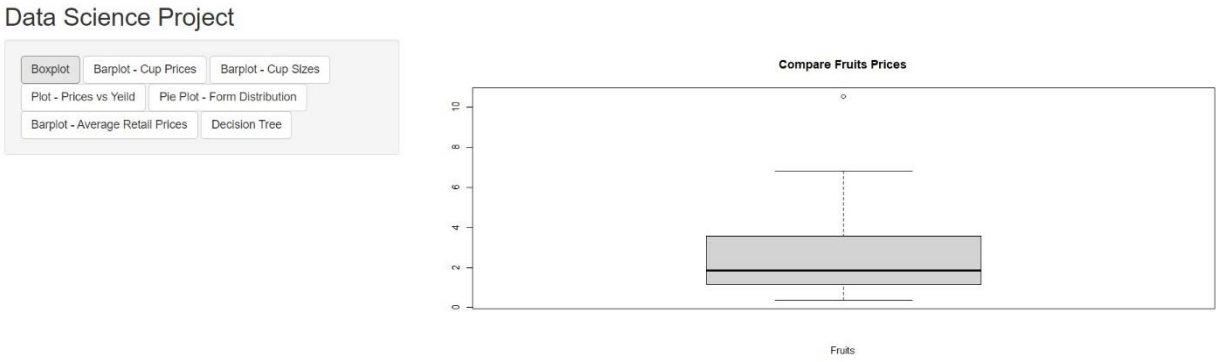



: this is the decision tree for the first 10 rows from which we can conclude that canned fruit is the most sold form (make the most yield) and the plot shows the probability that if it is canned then any other types of fruits at the same time based on the conditions in the above figure.

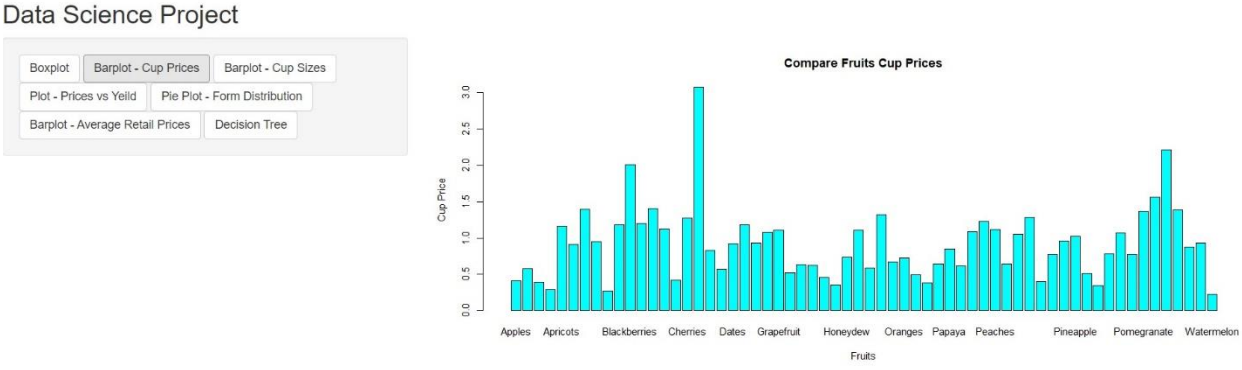
- 



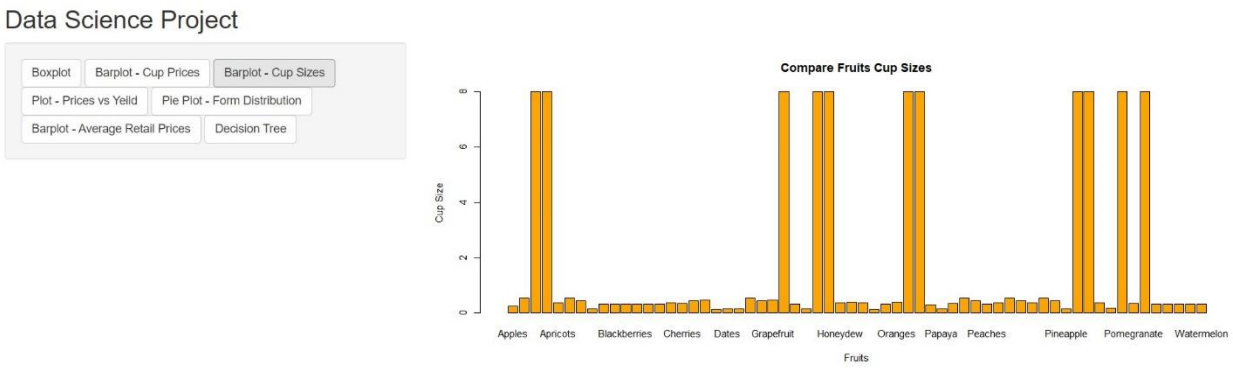

These are UI buttons used for Displaying various plots for specific cloumns in the data set.



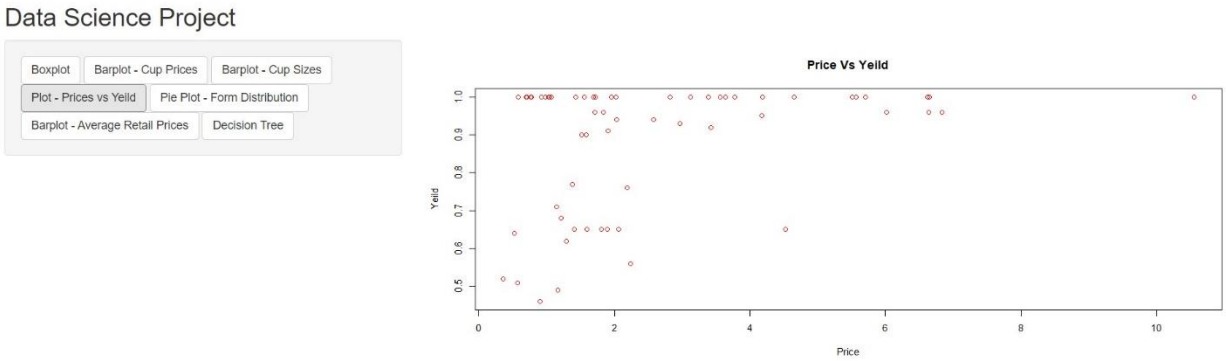

This is Box plot which compares different fruits prices we see that the fruits prices are nearly between 1 and 3 max is near to 7 and min is less than 0.5 and First Quartile (Q1) is above one and second quartile (Q2)(mean) is near to 2 and third quartile (Q3) is near to 4.



This is bar plot which compares fruits cup prices and we can conclude that cherries is the most expensive cup while watermelon is the cheapest one and the most cup fruits prices is less than 1.5.



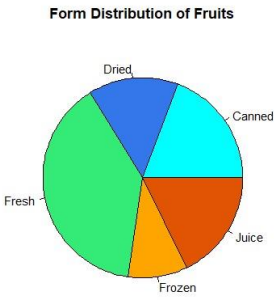

This is bar plot which compares fruits cup sizes(the size of juice inside a cup from one fruit piece) we can conclude that some of them have the biggest size 8 and most of them have low size that is smaller than 1.




This is plot which compare between price and yield that told us that the lower the price the higher the sell which mean that higher in the yield and we see that yield become most when price is between 1 and two and most of the data more than 2 dollars it is yeild is between 0.9 and 1.

Data Science Project

Boxplot | Barplot - Cup Prices | Barplot - Cup Sizes | Plot - Prices vs Yeild | Pie Plot - Form Distribution | Barplot - Average Retail Prices | Decision Tree

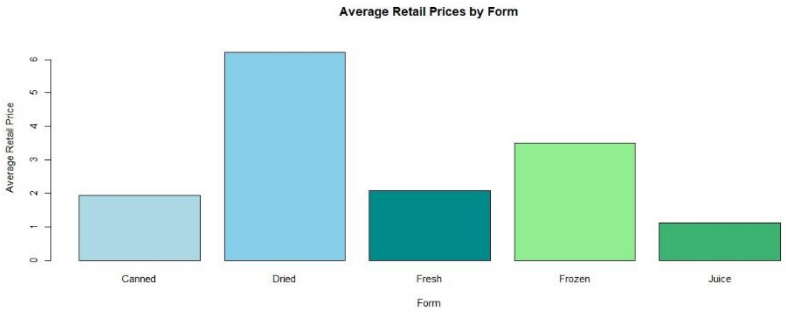



Looking at this pie plot we see that is a form distribution of fruits and we can conclude that fresh fruits are the most one in the market followed by canned and juice as they are nearly the same amount

Then dried and finally frozen which is the lowest amount in the market.

Data Science Project

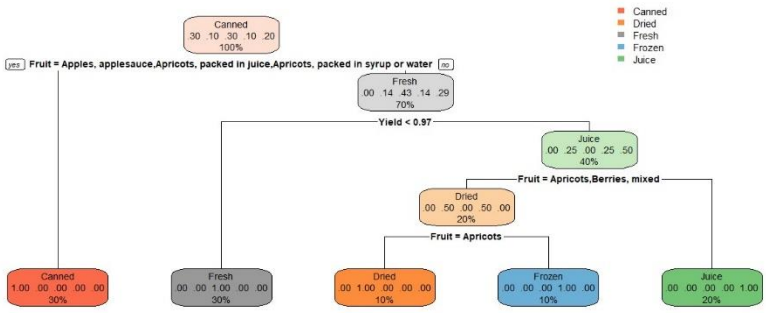
Boxplot | Barplot - Cup Prices | Barplot - Cup Sizes | Plot - Prices vs Yeild | Pie Plot - Form Distribution | Barplot - Average Retail Prices | Decision Tree




This is a bar plot that clarifies the average retail prices by form and we can conclude that dried fruits have the most average retail price then frozen have a medium price then canned and fresh have the same retail price and finally we reached juice which has the lowest retail price.

Data Science Project

Boxplot | Barplot - Cup Prices | Barplot - Cup Sizes | Plot - Prices vs Yeild | Pie Plot - Form Distribution | Barplot - Average Retail Prices | Decision Tree



Finally it is the decision tree which we have already explained in other pages.

## 5.Conclusion:

- To conclude we can say that :
- fresh fruits is the most existing form .
- canned fruits is the most sold form (make the highest yield) so it should be the most delicious one.
- Dried is the most expensive form.
- Juice has the lowest average retail price .
- Frozen is the smallest amount .
- cherries is the most expensive cup while watermelon is the cheapest one.
- Fruit cup Sizes vary too much.