

RWorksheet_sayson#4b.Rmd

Adrian T. Sayson

2025-11-24

```
#1.  
vectorA <- c(1, 2, 3, 4, 5)  
matrix5x5 <- matrix(0, nrow = 5, ncol = 5)  
  
for(i in 1:5) {  
  for(j in 1:5) {  
    matrix5x5[i, j] <- abs(vectorA[j] - i)  
  }  
}  
print(matrix5x5)
```

```
##      [,1] [,2] [,3] [,4] [,5]  
## [1,]     0    1    2    3    4  
## [2,]     1    0    1    2    3  
## [3,]     2    1    0    1    2  
## [4,]     3    2    1    0    1  
## [5,]     4    3    2    1    0
```

```
#2.  
for(i in 1:5){  
  for(j in 1:i){  
    cat("*")  
  }  
  cat("\n")  
}
```

```
## *  
## **  
## ***  
## ****  
## *****
```

```
#3.  
a <- 0  
b <- 1  
  
cat(a, " ", sep = "")
```

```
## 0,
```

```

repeat {
  cat(b, ", ", sep = "") 
  next_val <- a + b
  a <- b
  b <- next_val

  if (b > 500) {
    break
  }
}

## 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377,

```

```

fib <- numeric()
a <- 0
b <- 1
while(b <= 500){
  fib <- c(fib, b)
  next_val <- a + b
  a <- b
  b <- next_val
}
cat("Fibonacci numbers up to 500:", paste(fib, collapse = ", "))

```

```
## Fibonacci numbers up to 500: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377
```

```
#4.
shoess <- read.csv("shoe_table.csv")
print(shoess)
```

	Shoe_size	Height	Gender
## 1	6.5	66.0	F
## 2	9.0	68.0	F
## 3	8.5	64.5	F
## 4	8.5	65.0	F
## 5	10.5	70.0	M
## 6	7.0	64.0	F
## 7	9.5	70.0	F
## 8	9.0	71.0	F
## 9	13.0	72.0	M
## 10	7.5	64.0	F
## 11	10.5	74.5	M
## 12	8.5	67.0	F
## 13	12.0	71.0	M
## 14	10.5	71.0	M
## 15	13.0	77.0	M
## 16	11.5	72.0	M
## 17	8.5	59.0	F
## 18	5.0	62.0	F
## 19	10.0	72.0	M
## 20	6.5	66.0	F
## 21	7.5	64.0	F

```
## 22      8.5  67.0    M
## 23     10.5  73.0    M
## 24     10.5  72.0    M
## 25     11.0  69.0    M
## 26      9.0  69.0    M
## 27     13.0  70.0    M
```

```
#A
head(shoess)
```

```
##   Shoe_size Height Gender
## 1      6.5    66.0      F
## 2      9.0    68.0      F
## 3      8.5    64.5      F
## 4      8.5    65.0      F
## 5     10.5    70.0      M
## 6      7.0    64.0      F
```

```
#b
female <- subset(shoess, Gender == "F")
male <- subset(shoess, Gender == "M")
nrow(male)
```

```
## [1] 14
```

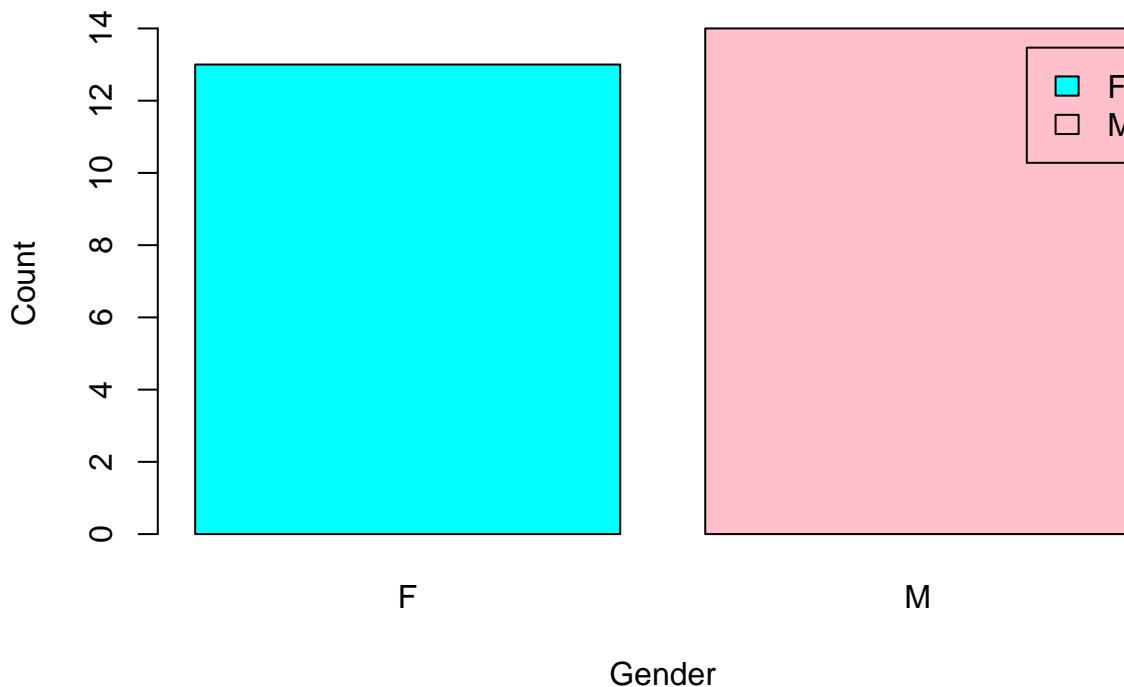
```
nrow(female)
```

```
## [1] 13
```

```
#c
gender_ct <- table(shoess$Gender)

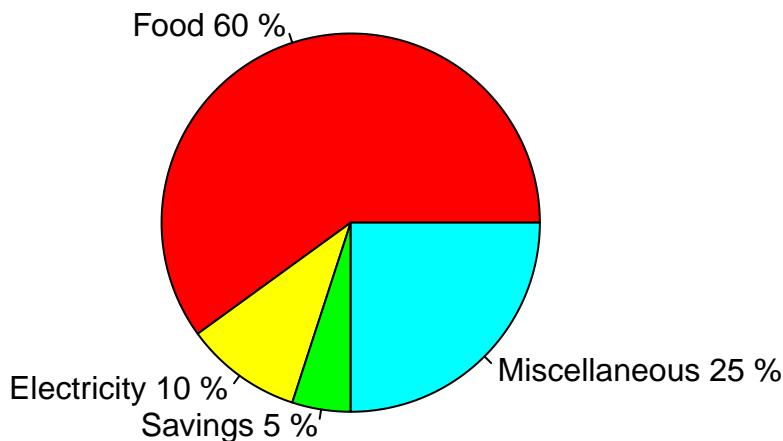
barplot(gender_ct,
         main = "Number of Males and Females",
         xlab = "Gender",
         ylab = "Count",
         col = c("cyan", "pink"),
         legend.text = TRUE)
```

Number of Males and Females



```
#5
expenses <- c(60, 10, 5, 25)
categories <- c("Food", "Electricity", "Savings", "Miscellaneous")
percent <- round(expenses / sum(expenses) * 100)
labels <- paste(categories, percent, "%")
pie(expenses,
    labels = labels,
    col = c("red", "yellow", "green", "cyan"),
    main = "Monthly Income Distribution of Dela Cruz Family")
```

Monthly Income Distribution of Dela Cruz Family



```
#a
```

```
str(iris)
```

```
## 'data.frame': 150 obs. of 5 variables:  
## $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...  
## $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...  
## $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...  
## $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...  
## $ Species     : Factor w/ 3 levels "setosa","versicolor",...: 1 1 1 1 1 1 1 1 1 1 ...
```

#There is 150 rows and 5 columns. The dataset has 3 species of the "Iris" flowers which are then each c

```
#b
```

```
means_iris <- colMeans(iris[, 1:4])
```

```
means_iris
```

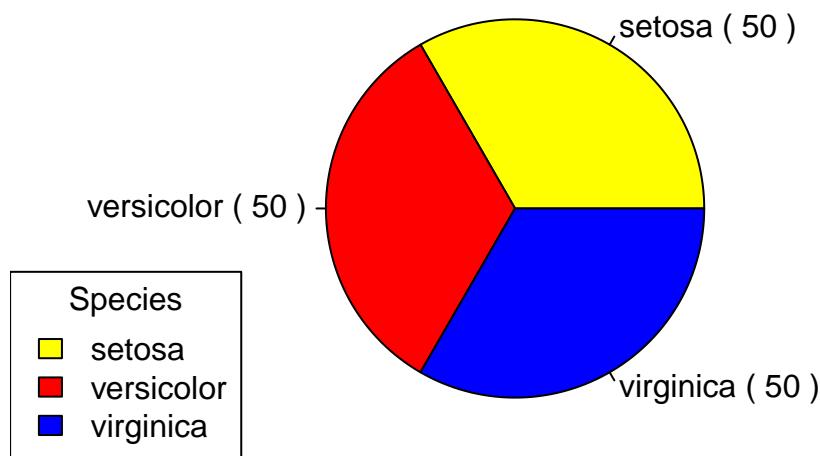
```
## Sepal.Length Sepal.Width Petal.Length Petal.Width  
##      5.843333     3.057333     3.758000     1.199333
```

```
#c
```

```
species_count <- table(iris$Species)
```

```
pie(species_count,  
    main = "Distribution of Iris Species",  
    col = c("yellow", "red", "blue"),  
    labels = paste(names(species_count), "(", species_count, ")")  
)  
legend("bottomleft",  
      legend = names(species_count),  
      fill = c("yellow", "red", "blue"),  
      title = "Species")
```

Distribution of Iris Species



```

#d
setosa <- subset(iris, Species == "setosa")
versicolor <- subset(iris, Species == "versicolor")
virginica <- subset(iris, Species == "virginica")

tail(setosa)

##      Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 45          5.1       3.8        1.9       0.4   setosa
## 46          4.8       3.0        1.4       0.3   setosa
## 47          5.1       3.8        1.6       0.2   setosa
## 48          4.6       3.2        1.4       0.2   setosa
## 49          5.3       3.7        1.5       0.2   setosa
## 50          5.0       3.3        1.4       0.2   setosa

tail(versicolor)

##      Sepal.Length Sepal.Width Petal.Length Petal.Width     Species
## 95          5.6       2.7        4.2       1.3 versicolor
## 96          5.7       3.0        4.2       1.2 versicolor
## 97          5.7       2.9        4.2       1.3 versicolor
## 98          6.2       2.9        4.3       1.3 versicolor
## 99          5.1       2.5        3.0       1.1 versicolor
## 100         5.7       2.8        4.1       1.3 versicolor

tail(virginica)

##      Sepal.Length Sepal.Width Petal.Length Petal.Width     Species
## 145         6.7       3.3        5.7       2.5 virginica
## 146         6.7       3.0        5.2       2.3 virginica
## 147         6.3       2.5        5.0       1.9 virginica
## 148         6.5       3.0        5.2       2.0 virginica
## 149         6.2       3.4        5.4       2.3 virginica
## 150         5.9       3.0        5.1       1.8 virginica

#e
iris$Species <- as.factor(iris$Species)

colors <- c("setosa" = "orange",
           "versicolor" = "purple",
           "virginica" = "green")

symbols <- c("setosa" = 16,
            "versicolor" = 17,
            "virginica" = 15)

plot(iris$Sepal.Length, iris$Sepal.Width,
      col = colors[iris$Species],
      pch = symbols[iris$Species],
      main = "Iris Dataset",
      sub = "Sepal Width and Length",

```

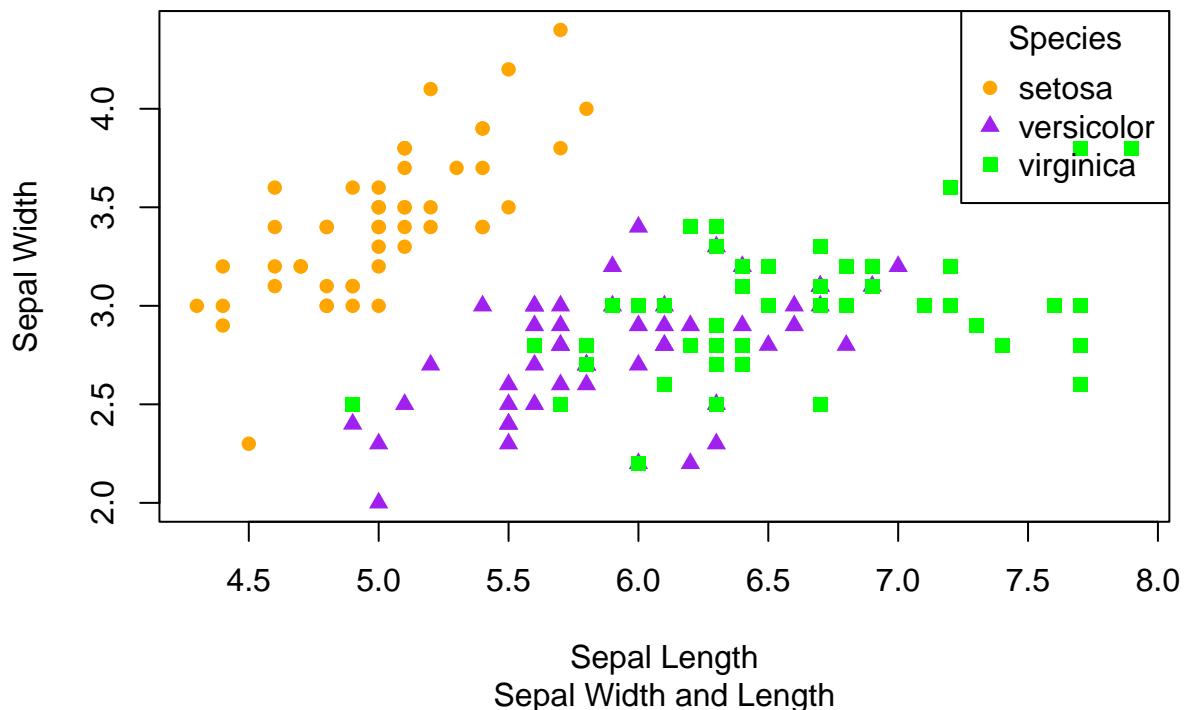
```

xlab = "Sepal Length",
ylab = "Sepal Width")

legend("topright",
       legend = levels(iris$Species),
       col = colors,
       pch = symbols,
       title = "Species")

```

Iris Dataset



```

#f
#The scatterplot shows that the virginica iris has the overall longest sepal and the setosa iris has th

```

```

#7
library(readxl)

alexa <- read_excel("alexa-file.xlsx")

print(alexa)

## # A tibble: 5 x 5
##   rating date      variation  verified_reviews feedback
##   <dbl> <chr>     <chr>        <chr>           <dbl>
## 1      5 2018-07-30 Black Dot    Good             1
## 2      5 2018-07-30 Black Dot  Pretty OK          1
## 3      5 2018-07-30 Black Dot    Great            1
## 4      5 2018-07-30 White Dot  Lackluster        1
## 5      5 2018-07-30 White Plus NA              1

```

```
#a

alexa$variation <- gsub("\\s+", " ", trimws(alexa$variation))

alexa$variation <- gsub("Black Dot", "Black", alexa$variation)
alexa$variation <- gsub("White Dot", "White", alexa$variation)
alexa$variation <- gsub("White Plus", "White Plus", alexa$variation)

knitr::include_graphics("Iris-PNG-Pic.png")
```



```

#b
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

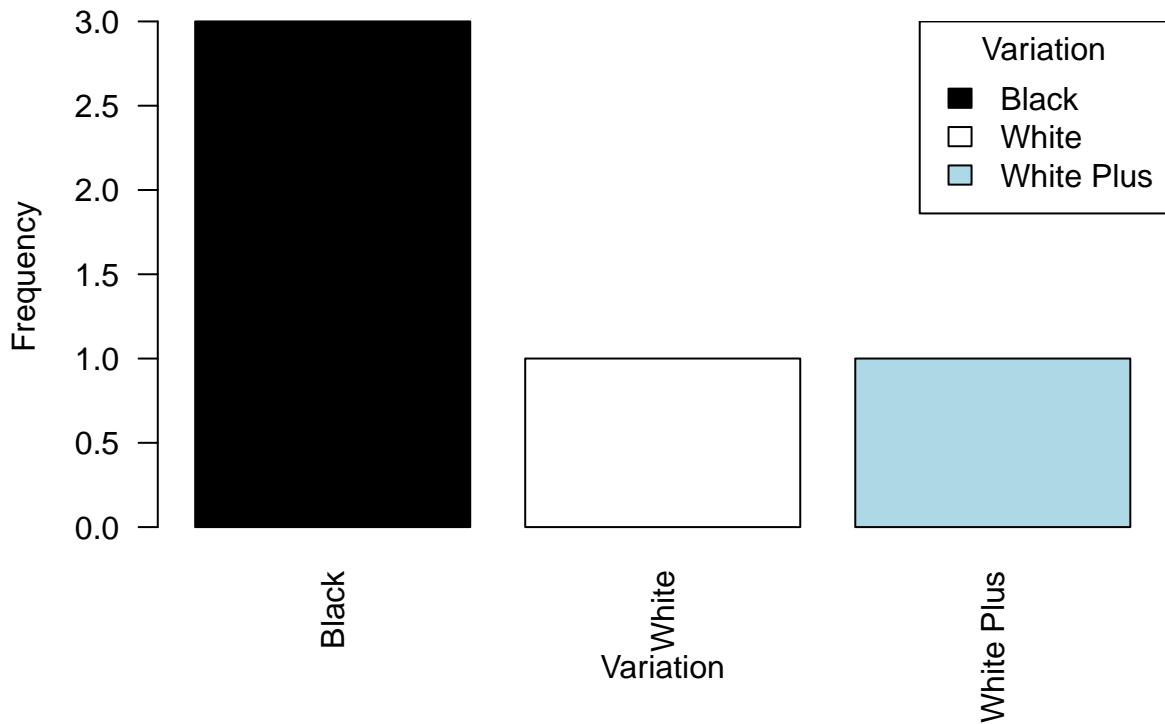
variations <- alexa %>%
  count(variation)
save(variations, file = "variations.RData")
variations

## # A tibble: 3 x 2
##   variation      n
##   <chr>        <int>
## 1 Black          3
## 2 White          1
## 3 White Plus    1

#c
colors <- c("Black" = "black", "White" = "white", "White Plus" = "lightblue")
barplot(
  height = variations$n,
  names.arg = variations$variation,
  main = "Count of Alexa Device Variations",
  xlab = "Variation",
  ylab = "Frequency",
  col = colors[variations$variation],
  las = 2
)
legend(
  "topright",
  legend = variations$variation,
  fill = colors[variations$variation],
  title = "Variation"
)

```

Count of Alexa Device Variations



```
#7  
  
load("variations.RData")  
  
bw <- variations[variations$variation %in% c("Black", "White"), ]  
  
barplot(  
  height = bw$n,  
  names.arg = bw$variation,  
  main = "Comparison of Black and White Variations",  
  xlab = "Variation",  
  ylab = "Count",  
  col = c("black", "white"),  
  beside = TRUE  
)  
legend(  
  "topright",  
  legend = bw$variation,  
  fill = c("black", "white"),  
  title = "Variation"  
)
```

Comparison of Black and White Variations

