

RWorksheet_Calambro#4b

Christian Paul Calambro

2024-11-10

```
#1
vectorA <- c(1,2,3,4,5)
patmat <- matrix(0, nrow = 5, ncol = 5)

for (i in 1:5) {
  for (j in 1:5) {
    patmat[i, j] <- vectorA[(j + i - 2) %% 5 + 1]
  }
}
cat("Pattern Matrix:\n")
```

Pattern Matrix:

```
print(patmat)
```

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

```
zero_matrix <- matrix(0, nrow = 5, ncol = 5)
cat("\nZero Matrix:\n")
```

##

Zero Matrix:

```
print(zero_matrix)
```

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

#2

```
n <- 4
```

```
for (i in 1:n) {
  for (j in 1:i) {
    cat("*")
  }
}
```

```

    cat("\n")
}

## *
## **
## ***
## ****

#3
first <- as.integer(readline(prompt = "Enter starting number for fibonacci sequence(up to 500): "))

## Enter starting number for fibonacci sequence(up to 500):
fibonacci <- c(0,1)

repeat {
  next_fib <- sum(tail(fibonacci, 2))
  if (next_fib > 500) {
    break
  }
  fibonacci <- c(fibonacci, next_fib)
}

result <- fibonacci[fibonacci >= first]
cat("Fibonacci sequence from", first, "up to 500:", result, "\n")

## Fibonacci sequence from NA up to 500: NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
#4a
SHG <- read.csv("/cloud/project/Worksheet4/shoe_size_data.csv", header = TRUE, sep = ",")
SHG[1:6,]

##   Shoe.size Height Gender Shoe.size.1 Height.1 Gender.1
## 1      6.5  66.0      F      13.0      77          M
## 2      9.0  68.0      F      11.5      72          M
## 3      8.5  64.5      F       8.5      59          F
## 4      8.5  65.0      F       5.0      62          F
## 5     10.5  70.0      M      10.0      72          M
## 6       7.0  64.0      F       6.5      66          F

#4b
male_sub <- subset(SHG, Gender == "M")
female_sub <- subset(SHG, Gender == "F")

num_male <- nrow(male_sub)
num_female <- nrow(female_sub)

cat("Number of Male observations:", num_male, "\n")

## Number of Male observations: 5
cat("Number of Female observations:", num_female, "\n")

## Number of Female observations: 9
#4c

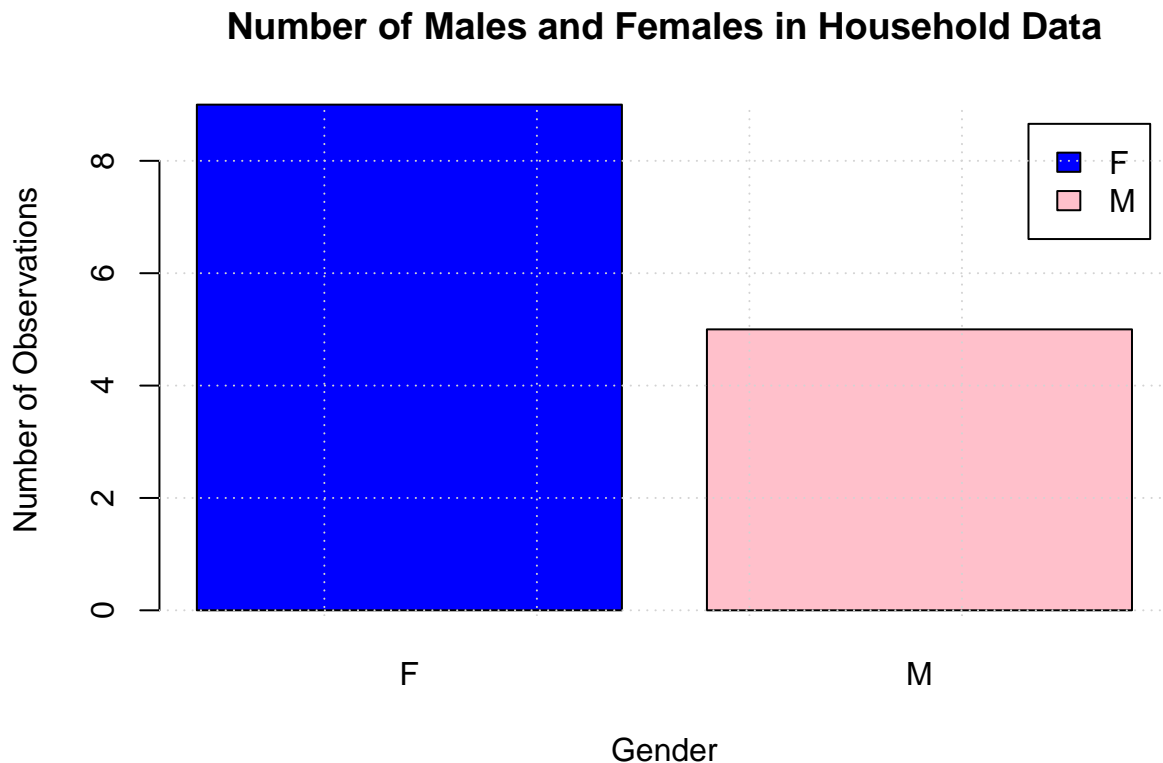
```

```

gender_count <- table(SHG$Gender)
bar_colors <- c("blue", "pink") # Colors for male and female
barplot(gender_count,
        main = "Number of Males and Females in Household Data",
        xlab = "Gender",
        ylab = "Number of Observations",
        col = bar_colors,
        legend = rownames(gender_count),
        beside = TRUE)

# Add grid lines
grid(nx = NULL, ny = NULL)

```



#5

```

spending <- c(Food = 60, Electricity = 10, Savings = 5, Miscellaneous = 25)

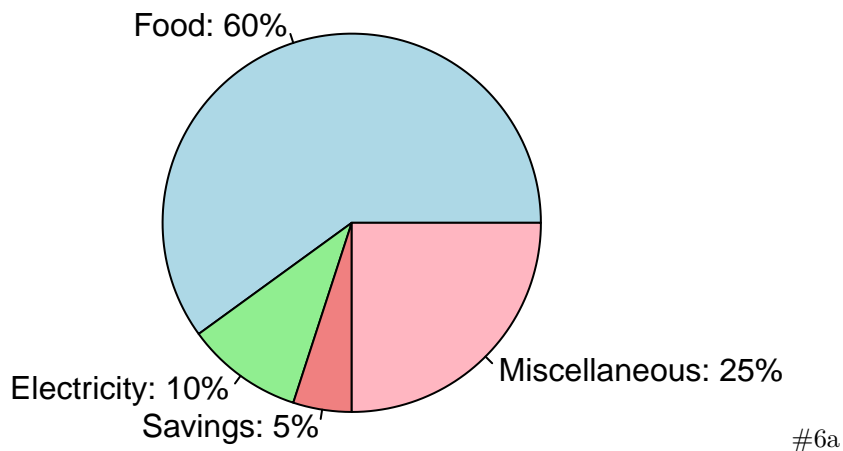
percent_labels <- paste0(names(spending), ": ", round(spending / sum(spending) * 100, 1), "%")

colors <- c("lightblue", "lightgreen", "lightcoral", "lightpink")

pie(spending,
    labels = percent_labels,
    col = colors,
    main = "Dela Cruz Family Monthly Income Distribution"
)

```

Dela Cruz Family Monthly Income Distribution



```
data(iris)
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 ...

#6b

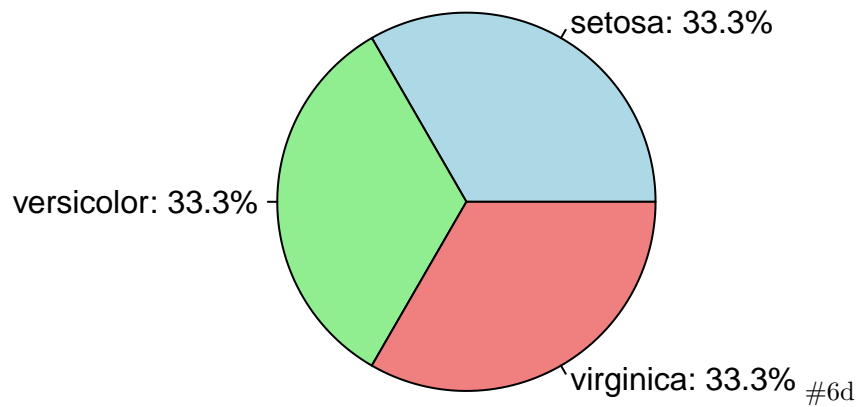
mean_values <- colMeans(iris[, c("Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width")])
mean_values

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

#6c

species_counts <- table(iris$Species)
pie(species_counts,
    labels = paste(names(species_counts), ": ", round(species_counts / sum(species_counts) * 100, 1), "%"),
    col = c("lightblue", "lightgreen", "lightcoral"),
    main = "Species Distribution in Iris Dataset"
)
```

Species Distribution in Iris Dataset



```
setosa_subset <- tail(subset(iris, Species == "setosa"), 6)
versicolor_subset <- tail(subset(iris, Species == "versicolor"), 6)
virginica_subset <- tail(subset(iris, Species == "virginica"), 6)
```

setosa_subset

##	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

versicolor_subset

##	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

virginica_subset

##	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

#6e

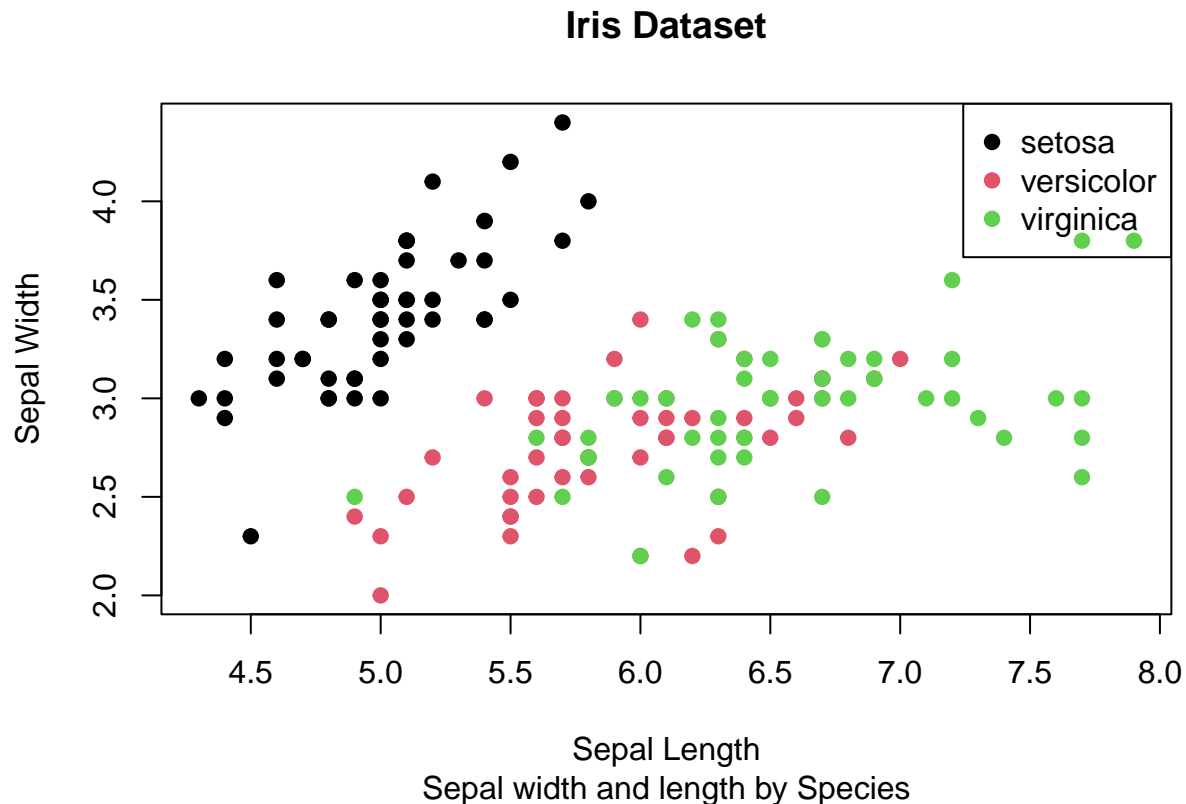
```
plot(iris$Sepal.Length, iris$Sepal.Width,
     col = iris$Species,
     pch = 19,
```

```

xlab = "Sepal Length",
ylab = "Sepal Width",
main = "Iris Dataset",
sub = "Sepal width and length by Species"
)

legend("topright", legend = levels(iris$Species),
      col = 1:3, pch = 19)

```



#6f
 #For part f, after making the scatterplot, you can explain what you see by looking at how the points are
 #grouped for each species. Notice if certain species have bigger or smaller sepals (the lengths and widths of
 #the flower parts). See if each type of flower is grouped together or spread out differently, which might tell
 you how their sepal sizes vary from one species to another.

#7a

```

library(readxl)
alexa_data <- read_excel("/cloud/project/Worksheet4/alexa_file.xlsx")

alexa_data$variation <- gsub("Black\\s+Dot", "Black Dot", alexa_data$variation)
alexa_data$variation <- gsub("Black\\s+Plus", "Black Plus", alexa_data$variation)
alexa_data$variation <- gsub("Black\\s+Show", "Black Show", alexa_data$variation)
alexa_data$variation <- gsub("Black\\s+Spot", "Black Spot", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Dot", "White Dot", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Plus", "White Plus", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Show", "White Show", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Spot", "White Spot", alexa_data$variation)

```

```
table(alexa_data$variation)
```

```
##
##           Black           Black Dot
##           261           516
##       Black Plus       Black Show
##           270           265
##       Black Spot       Charcoal Fabric
##           241           430
## Configuration: Fire TV Stick       Heather Gray Fabric
##           350           157
##           Oak Finish       Sandstone Fabric
##           14             90
##       Walnut Finish       White
##           9             91
##           White Dot       White Plus
##           184           78
##           White Show       White Spot
##           85           109
```

```
#7b
```

```
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_count <- alexa_data %>%
  count(variation, name = "Total")
```

```
save(variations_count, file = "variations.RData")
```

```
print(variations_count)
```

```
## # A tibble: 16 x 2
##   variation      Total
##   <chr>         <int>
## 1 Black         261
## 2 Black Dot     516
## 3 Black Plus    270
## 4 Black Show    265
## 5 Black Spot    241
## 6 Charcoal Fabric 430
## 7 Configuration: Fire TV Stick 350
## 8 Heather Gray Fabric 157
## 9 Oak Finish     14
```

```
## 10 Sandstone Fabric          90
## 11 Walnut Finish             9
## 12 White                    91
## 13 White Dot                184
## 14 White Plus               78
## 15 White Show               85
## 16 White Spot              109
```

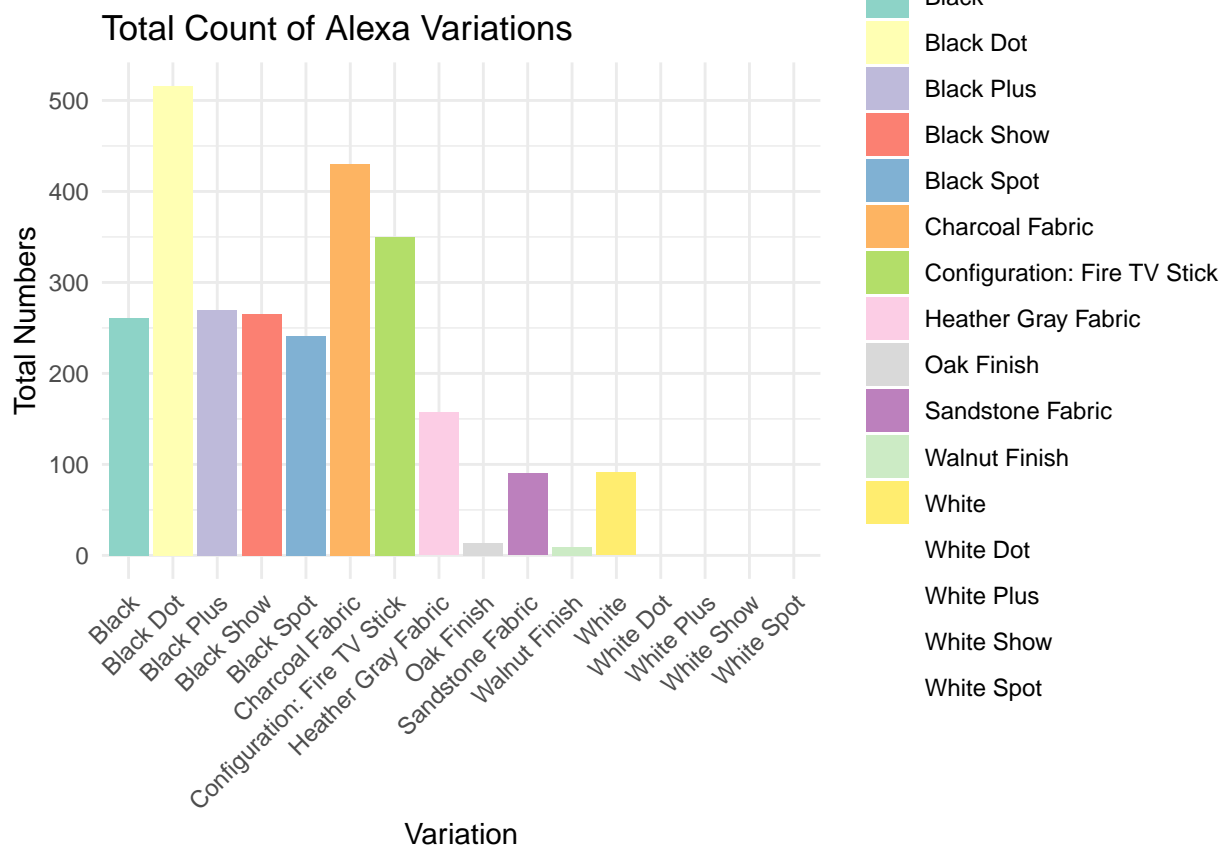
#7c

```
library(ggplot2)

load("variations.RData")

ggplot(variations_count, aes(x = variation, y = Total, fill = variation)) +
  geom_bar(stat = "identity") +
  ggtitle("Total Count of Alexa Variations") +
  xlab("Variation") +
  ylab("Total Numbers") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Set3")
```

```
## Warning in RColorBrewer::brewer.pal(n, pal): n too large, allowed maximum for palette Set3 is 12
## Returning the palette you asked for with that many colors
```



#7d


```

variations_count$Category <- ifelse(grepl("Black", variations_count$variation), "Black Variants",
                                     ifelse(grepl("White", variations_count$variation), "White Variants",
                                              NA))

black_white_variants <- variations_count %>% filter(!is.na(Category))

ggplot(black_white_variants, aes(x = variation, y = Total, fill = variation)) +
  geom_bar(stat = "identity") +
  facet_wrap(~ Category, scales = "free_x") +
  ggtitle("Counts of Alexa Black and White Variants") +
  xlab("Variation") +
  ylab("Total Numbers") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Set2")

```

```

## Warning in RColorBrewer::brewer.pal(n, pal): n too large, allowed maximum for palette Set2 is 8
## Returning the palette you asked for with that many colors

```

