

# RWorksheet\_PAMA#4B

Miko Pama

2023-11-08

#1

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

for (i in 1:5) {
  for (j in 1:5) {
    matrixA[i, j] <- abs(vectorA[i] - vectorA[j])
  }
}
```

matrixA

```
##      [,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

```
num_rows <- 5

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

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

#3

```
input.number <- as.numeric(readline("Enter a number to start the Fibonacci sequence: "))
```

```
## Enter a number to start the Fibonacci sequence:
```

```
assume.number <- 0
x <- 0
y <- 1
```

```
repeat {
  if (x > 500) {
    break
  }
  if (x >= assume.number) {
    cat(x, " ")
  }
  temp <- x + y
  x <- y
  y <- temp
}
```

```
## 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
```

```
cat("\n")
```

```
#4
```

```
library(readr)
library(readr)
shoe <- read_csv("shoe.csv")
```

```
## Rows: 28 Columns: 3
## -- Column specification -----
## Delimiter: ","
## chr (1): Gender
## dbl (2): Shoe Size, Height
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
shoe
```

```
## # A tibble: 28 x 3
##   `Shoe Size` Height Gender
##   <dbl> <dbl> <chr>
## 1      6.5    66    F
## 2      9     68    F
## 3      8.5   64.5  F
## 4      8.5   65    F
## 5     10.5   70    M
## 6      7     64    F
## 7      9.5   70    F
## 8      6     71    F
## 9     13     72    M
## 10     7.5   64    F
## # i 18 more rows
```

```
shoes <- read_csv("shoe.csv")
shoes
```

```
##   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      6.0  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     8.5  69.0    F
## 25    10.5  72.0    M
## 26    11.0  70.0    M
## 27     9.0  69.0    M
## 28    13.0  70.0    M
```

```
ssize <- shoes[c(1:6),]
ssize
```

```
##   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
```

```
male_subset <- shoes[shoes$Gender == "M", c("Shoe.Size", "Height")]
female_subset <- shoes[shoes$Gender == "F", c("Shoe.Size", "Height")]
male_subset
```

```
##   Shoe.Size Height
## 5      10.5   70.0
## 9      13.0   72.0
## 11     10.5   74.5
## 13     12.0   71.0
## 14     10.5   71.0
## 15     13.0   77.0
## 16     11.5   72.0
## 19     10.0   72.0
## 22      8.5   67.0
## 23     10.5   73.0
## 25     10.5   72.0
## 26     11.0   70.0
## 27      9.0   69.0
## 28     13.0   70.0
```

```
female_subset
```

```
##      Shoe.Size Height
## 1         6.5   66.0
## 2         9.0   68.0
## 3         8.5   64.5
## 4         8.5   65.0
## 6         7.0   64.0
## 7         9.5   70.0
## 8         6.0   71.0
## 10        7.5   64.0
## 12        8.5   67.0
## 17        8.5   59.0
## 18        5.0   62.0
## 20        6.5   66.0
## 21        7.5   64.0
## 24        8.5   69.0
```

```
GraphMF<- table(shoes$Gender)
barplot(GraphMF,
  main = "Number of Males and Females",
  xlab = "Gender",
  ylab = "Count",
  col = c("plum", "gray"),
  legend.text = c("Male", "Female"),
  beside = TRUE
)
```

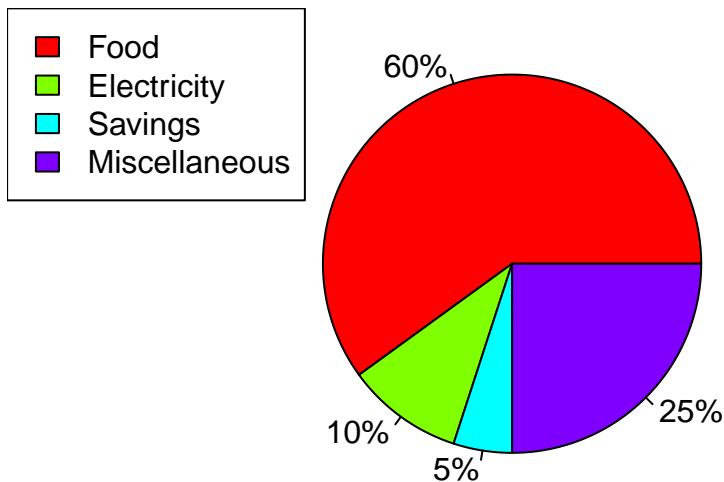


#5

```
pie_chart <- c(60, 10, 5, 25)
pie(pie_chart, labels = paste0(pie_chart, "%"),
    main = "Monthly Income of Dela Cruz family ", col = rainbow(length(pie_chart)))

legend("topleft", legend = c("Food", "Electricity", "Savings", "Miscellaneous"),
    fill = rainbow(length(pie_chart)))
```

## Monthly Income of Dela Cruz family



#6

```
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 ...
## It has 150 observations and 5 variables.

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

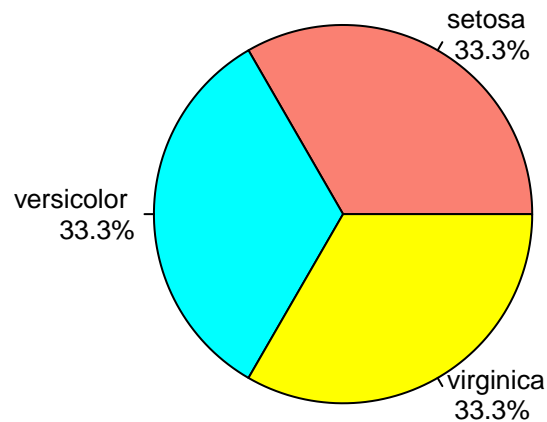
## Sepal.Length Sepal.Width Petal.Length Petal.Width
## 5.843333 3.057333 3.758000 1.199333

specs <- table(iris$Species)

clors <- c("salmon", "cyan", "yellow")

pie(specs,
    labels = paste(names(specs), "\n", sprintf("%.1f%%", prop.table(specs) * 100)),
    col = clors,
    main = "Species Distribution",
    cex.main = 1.5,
    cex = 0.8
)
```

# Species Distribution



```
SetSub <- subset(iris, Species == "setosa")
VersiSub <- subset(iris, Species == "versicolor")
VirgiSub <- subset(iris, Species == "virginica")
```

```
# Display the last six rows of each species
cat("Last six rows of Setosa:")
```

```
## Last six rows of Setosa:
```

```
print(tail(SetSub))
```

```
##      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
```

```
cat("Last six rows of Versicolor:")
```

```
## Last six rows of Versicolor:
```

```
print(tail(VersiSub))
```

```
##      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
```

```
cat("Last six rows of Virginica:")
```

```
## Last six rows of Virginica:
```

```
print(tail(VirgiSub))
```

```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
```

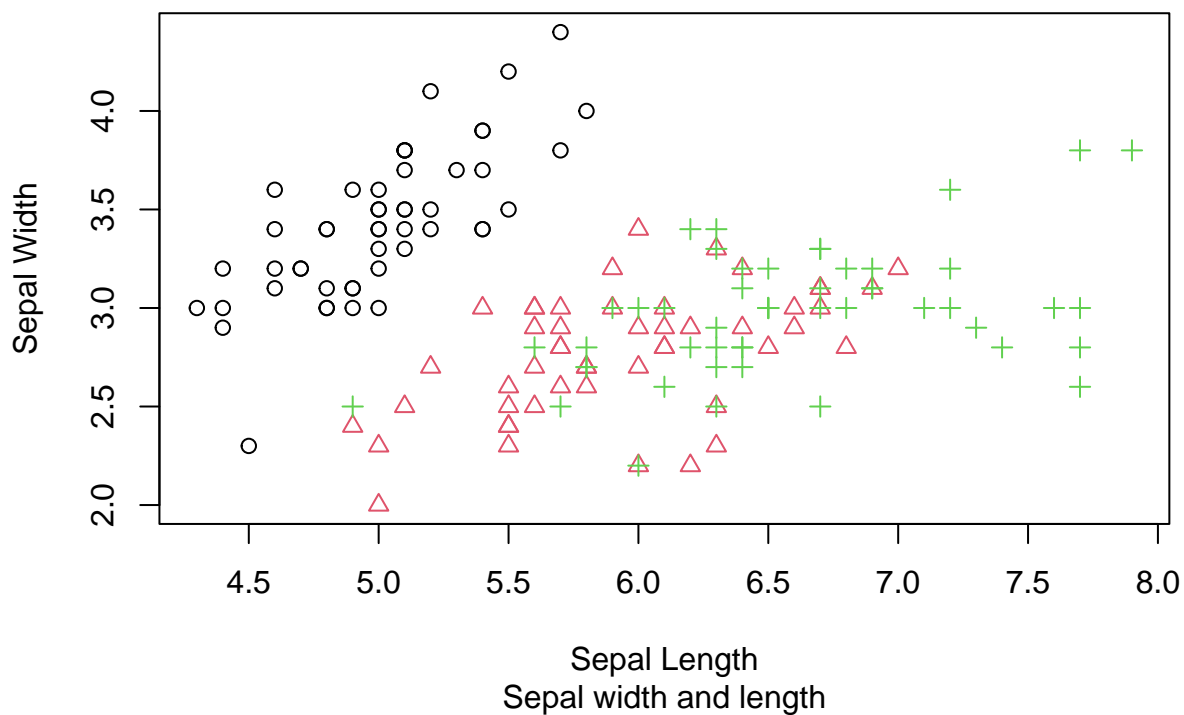
```
## 145      6.7      3.3      5.7      2.5 virginica
## 146      6.7      3.0      5.2      2.3 virginica
## 147      6.3      2.5      5.0      1.9 virginica
## 148      6.5      3.0      5.2      2.0 virginica
## 149      6.2      3.4      5.4      2.3 virginica
## 150      5.9      3.0      5.1      1.8 virginica
```

```
data(iris)

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

plot(iris$Sepal.Length, iris$Sepal.Width,
     pch = as.integer(iris$Species),
     col = iris$Species,
     main = "Iris Dataset",
     sub = "Sepal width and length",
     xlab = "Sepal Length",
     ylab = "Sepal Width"
)
```

**Iris Dataset**



*#The scatterplot shows similarities between the sepal width and length ranging from 5.5 to 7.0.*

#7

```
library(readxl)
alexa_file <- read_excel("alexa_file.xlsx")
alexa_file
```

```
## # A tibble: 3,150 x 5
```

```
##   rating date      variation      verified_reviews      feedback
```

```
##      <dbl> <dtm>          <chr>          <chr>          <dbl>
## 1      5 2018-07-31 00:00:00 Charcoal Fabric Love my Echo!      1
## 2      5 2018-07-31 00:00:00 Charcoal Fabric Loved it!          1
## 3      4 2018-07-31 00:00:00 Walnut Finish   Sometimes while play~    1
## 4      5 2018-07-31 00:00:00 Charcoal Fabric I have had a lot of ~    1
## 5      5 2018-07-31 00:00:00 Charcoal Fabric Music                1
## 6      5 2018-07-31 00:00:00 Heather Gray Fabric I received the echo ~ 1
## 7      3 2018-07-31 00:00:00 Sandstone Fabric Without having a cel~ 1
## 8      5 2018-07-31 00:00:00 Charcoal Fabric I think this is the ~    1
## 9      5 2018-07-30 00:00:00 Heather Gray Fabric looks great        1
## 10     5 2018-07-30 00:00:00 Heather Gray Fabric Love it! I've listen~ 1
## # i 3,140 more rows
```

```
# Remove extra whitespaces in black variants
alexa_file$variation <- gsub("\\s+", " ", alexa_file$variation)
alexa_file$variation <- gsub("Black ", "Black", alexa_file$variation)

# Remove extra whitespaces in white variants
alexa_file$variation <- gsub("\\s+", " ", alexa_file$variation)
alexa_file$variation <- gsub("White ", "White", alexa_file$variation)
```

#7a

```
# Remove extra whitespaces in black variants
alexa_file$variation <- gsub("\\s+", " ", alexa_file$variation)
alexa_file$variation <- gsub("Black ", "Black", alexa_file$variation)

# Remove extra whitespaces in white variants
alexa_file$variation <- gsub("\\s+", " ", alexa_file$variation)
alexa_file$variation <- gsub("White ", "White", alexa_file$variation)
```

#7b

```
# Install and load the dplyr package
if (!require(dplyr)) {
  install.packages("dplyr")
}
```

```
## Loading required package: 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
```

```
library(dplyr)

# Group by Variation and calculate the total count
variation_counts <- alexa_file %>%
  group_by(variation) %>%
```



```

summarise(Count = n())

# Save the object as variations.RData
save(variation_counts, file = "variations.RData")

variation_counts

## # A tibble: 16 x 2
##   variation          Count
##   <chr>          <int>
## 1 Black          261
## 2 BlackDot       516
## 3 BlackPlus      270
## 4 BlackShow      265
## 5 BlackSpot      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 WhiteDot       184
## 14 WhitePlus       78
## 15 WhiteShow      85
## 16 WhiteSpot     109

#7c
# Load the variations.RData file
load("variations.RData")

# Increase the size of the plot
par(mar = c(5, 5, 4, 2) + 0.1) # Adjust the margins

# Create a barplot with rotated x-axis labels
barplot(variation_counts$Count,
        names.arg = variation_counts$variation,
        col = rainbow(length(variation_counts$variation)),
        main = "Variation Counts",
        xlab = "Variation",
        ylab = "Count",
        las = 2, # Rotate x-axis labels 90 degrees
        cex.names = 0.8, # Adjust the size of the x-axis labels
        width = 0.8) # Adjust the width of the bars

# Manually add legend in topright
legend_labels <- variation_counts$variation
legend_colors <- rainbow(length(legend_labels))

for (i in seq_along(legend_labels)) {
  rect(max(par("usr")[1]) + 0.1,
        max(par("usr")[3]) - i * 0.5,
        max(par("usr")[1]) + 0.3,

```

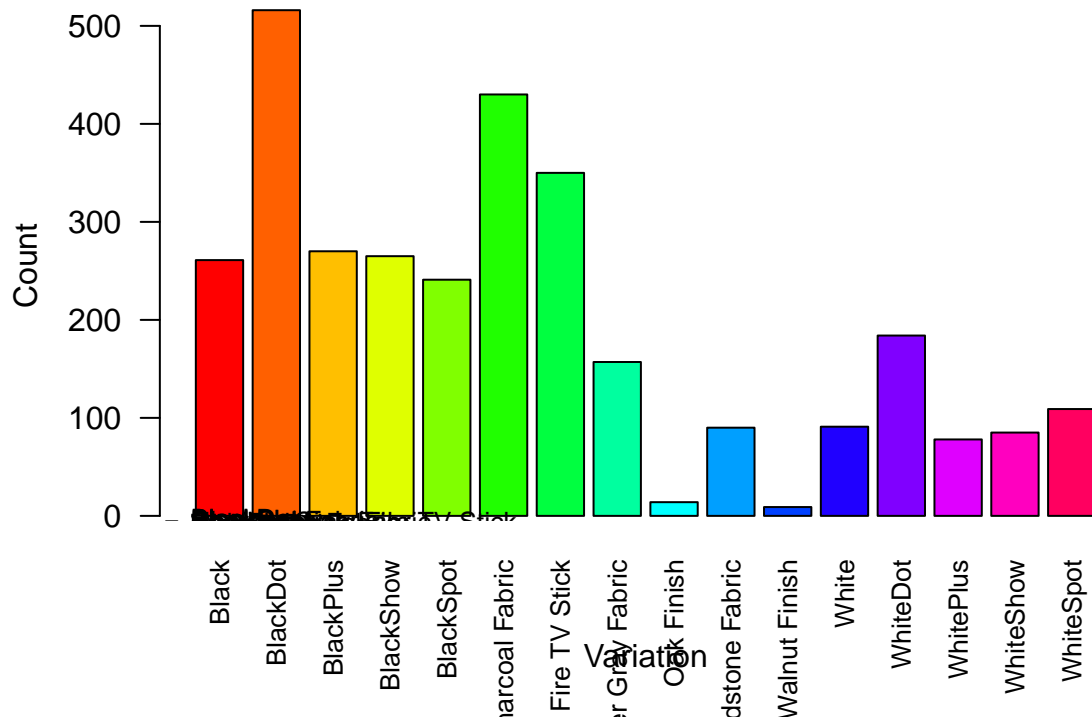
```

max(par("usr")[3]) - (i + 1) * 0.5,
col = legend_colors[i])

text(max(par("usr")[1]) + 0.4,
max(par("usr")[3]) - i * 0.5,
labels = legend_labels[i],
pos = 4,
offset = 0.2,
cex = 0.8)
}

```

## Variation Counts



```

#7d
# Load the variations.RData file
load("variations.RData")

# Extract data for black and white variations
black_variations <- variation_counts[variation_counts$variation %in% c("Black", "BlackDot", "BlackPlus")
white_variations <- variation_counts[variation_counts$variation %in% c("White", "WhiteDot", "WhitePlus")]

# Set up the plotting area
par(mfrow = c(1, 2)) # 1 row, 2 columns

# Barplot for black variations
barplot(black_variations$Count,
names.arg = black_variations$variation,
col = rainbow(length(black_variations$variation)),
main = "Black Variations",
xlab = "Variation",

```

```

ylab = "Count",
las = 2, # Rotate x-axis labels 90 degrees
cex.names = 0.8, # Adjust the size of the x-axis labels
width = 0.8) # Adjust the width of the bars

# Barplot for white variations
barplot(white_variations$Count,
names.arg = white_variations$variation,
col = rainbow(length(white_variations$variation)),
main = "White Variations",
xlab = "Variation",
ylab = "Count",
las = 2, # Rotate x-axis labels 90 degrees
cex.names = 0.8, # Adjust the size of the x-axis labels
width = 0.8) # Adjust the width of the bars

```

