Task 4

Abhishek Murthy

21BDS0064

2024-08-14

```
data <- c(5, 7, 8, 10, 12)

mean_value <- mean(data)
print(mean_value)
```

```
## [1] 8.4
```

```
median_value <- median(data)
print(median_value)
```

```
## [1] 8
```

```
find_mode <- function(x) {
  return(as.numeric(names(sort(table(x), decreasing = TRUE)[1])))
}
mode_value <- find_mode(data)
print(mode_value)
```

```
## [1] 5
```

```
variance_value <- var(data)
print(variance_value)
```

```
## [1] 7.3
```

```
sd_value <- sd(data)
print(sd_value)
```

```
## [1] 2.701851
```

```
library(modeest)
```

```
## Warning: package 'modeest' was built under R version 4.3.3
```

```
data <- c(2, 3, 3, 5, 7)
mode_value <- mfv(data)
print(mode_value)
```

```
## [1] 3
```

```r
# Bernoulli Distribution
n <- 10 # number of trials
p <- 0.5 # probability of success
bernoulli_data <- rbinom(n, 1, p)
print(bernoulli_data)
```

```
##  [1] 1 1 1 0 0 0 0 0 1 0
```

```r
# Chi-Square Distribution
n <- 10 # number of observations
df <- 5 # degrees of freedom
chi_square_data <- rchisq(n, df)
print(chi_square_data)
```

```
##  [1] 3.604329 2.510550 3.873842 2.888847 3.370809 4.007491 2.334854
6.413424
##  [9] 4.981080 2.497518
```

```r
# Binomial Distribution
n <- 20 # number of trials
p <- 0.3 # probability of success
size <- 10 # number of observations
# Generating random binomial data
binomial_data <- rbinom(size, n, p)
print(binomial_data)
```

```
##  [1] 4 7 6 7 6 4 5 7 3 7
```

```r
# Calculating probabilities
prob_success_5 <- dbinom(5, n, p)  # Probability of exactly 5 successes
print(prob_success_5)
```

```
## [1] 0.1788631
```

```r
# Normal Distribution
n <- 100 # number of observations
mean <- 50 # mean of the distribution
sd <- 10 # standard deviation of the distribution

# Generating random normal data
normal_data <- rnorm(n, mean, sd)
print(normal_data)
```

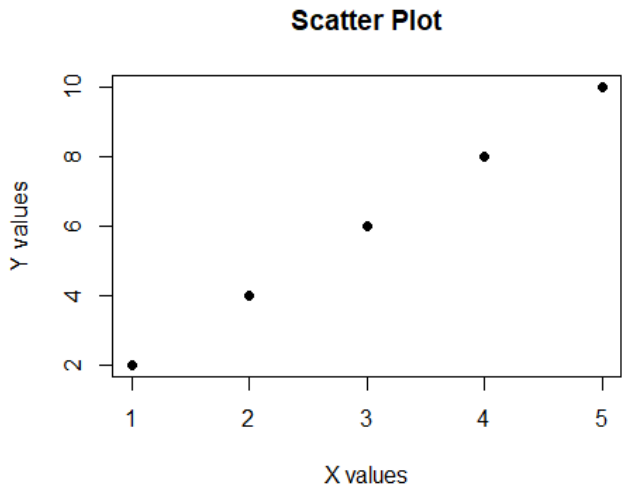
```
##   [1] 41.04134 44.28648 33.33913 54.35341 55.33323 47.85356 48.54451
43.46650
##   [9] 70.34658 49.48197 52.84672 58.24654 29.88986 66.75962 62.83022
51.39432
##  [17] 63.32550 57.08302 38.91776 56.93688 53.30677 60.53800 56.02227
62.33722
##  [25] 38.35490 32.98125 55.51420 60.32054 43.68406 27.40641 49.84891
56.63202
##  [33] 47.13464 40.00903 53.52463 52.15887 51.48148 41.10789 48.30426
```

```
51.13613
##  [41] 31.74304 55.62387 53.54622 40.19002 58.98417 52.64802 63.79210
58.86548
##  [49] 49.30362 52.32345 38.08167 55.68405 52.82474 72.21474 60.00929
56.03465
##  [57] 63.07406 21.93941 53.55849 58.68801 49.98481 41.14880 39.72658
34.97505
##  [65] 33.72657 37.37073 72.46617 49.18694 47.00906 60.37361 55.99406
61.79529
##  [73] 66.96532 64.96588 47.59423 44.92515 60.73434 62.32252 55.45546
46.26183
##  [81] 46.94594 36.42753 54.70871 42.73538 41.99468 50.64963 46.81584
51.49893
##  [89] 35.09125 50.33479 50.07440 40.68410 50.35551 53.59526 61.39830
60.99589
##  [97] 32.13765 50.97179 61.03977 41.84346
```

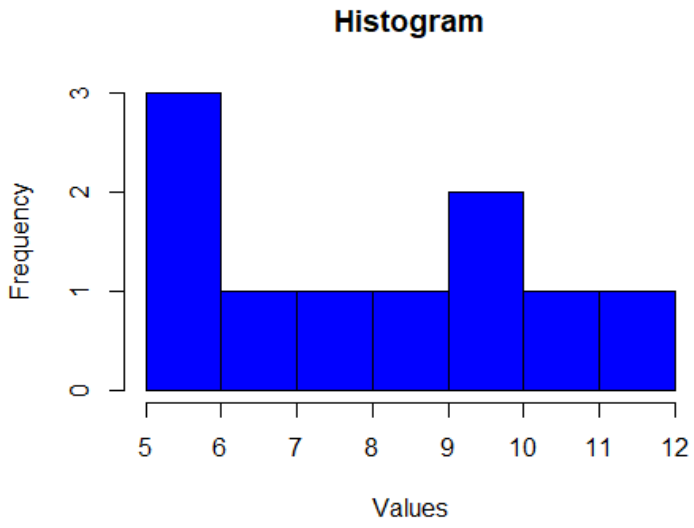
```
# Calculating probabilities
prob_less_than_60 <- pnorm(60, mean, sd)  # Probability of a value less than
60
print(prob_less_than_60)
```

```
## [1] 0.8413447
```

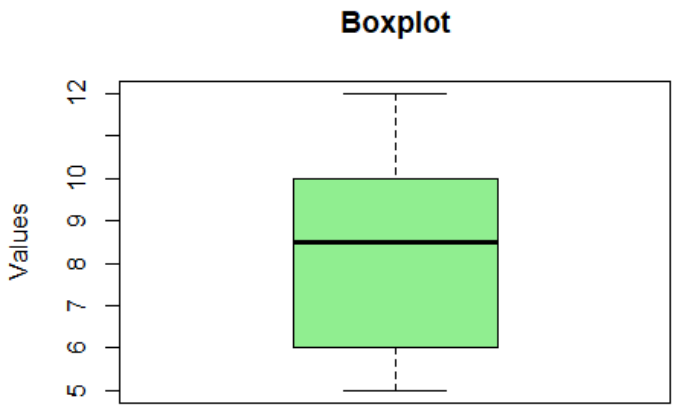
```
# Scatter Plot
x <- c(1, 2, 3, 4, 5)
y <- c(2, 4, 6, 8, 10)
plot(x, y, main = "Scatter Plot", xlab = "X values", ylab = "Y values", pch =
19)
```



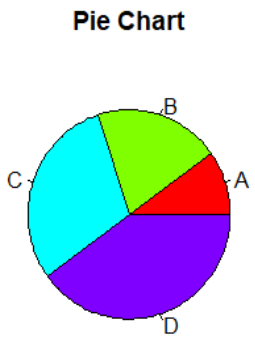

```
# Histogram
data <- c(5, 7, 8, 10, 12, 5, 6, 9, 10, 11)
hist(data, main = "Histogram", xlab = "Values", ylab = "Frequency", col =
"blue", breaks = 5)
```




```r
# Boxplot
data <- c(5, 7, 8, 10, 12, 5, 6, 9, 10, 11)
boxplot(data, main = "Boxplot", ylab = "Values", col = "lightgreen")
```




```r
# Pie Chart
data <- c(10, 20, 30, 40)
labels <- c("A", "B", "C", "D")
pie(data, labels = labels, main = "Pie Chart", col = rainbow(length(data)))
```


Pie Chart
B
C
A
D