Week4_lab

Joseph

2024-06-29

```
# Descriptive Statistics:
 # Measures of Central Tendency:
 #     - mean(x): Calculate the mean of a numeric vector x.
 #     - median(x): Calculate the median of a numeric vector x.
 #     - mode(x): Calculate the mode of a vector x.
 #   - Measures of Dispersion:
 #     - range(x): Calculate the range of a numeric vector x.
 #     - var(x): Calculate the variance of a numeric vector x.
 #     - sd(x): Calculate the standard deviation of a numeric vector x.
# Installed package DescTools to run Mode function to get the most occurences of the value
# install.packages("DescTools")
# library(DescTools)
x <- c(3, 4, 7, 9, 3,10)
mean_value <- mean(x)
median_value <- median(x)
mode_value <- as.numeric(names(table(x))[table(x) == max(table(x))])
range_value <- range(x)
variance_value <- var(x)
sd_value <- sd(x)
print(mean_value)
```

```
## [1] 6
```

```
print(median_value)
```

```
## [1] 5.5
```

```
print(mode_value)
```

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

```
print(range_value)
```

```
## [1]  3 10
```

```
print(variance_value)
```

```
## [1] 9.6
```

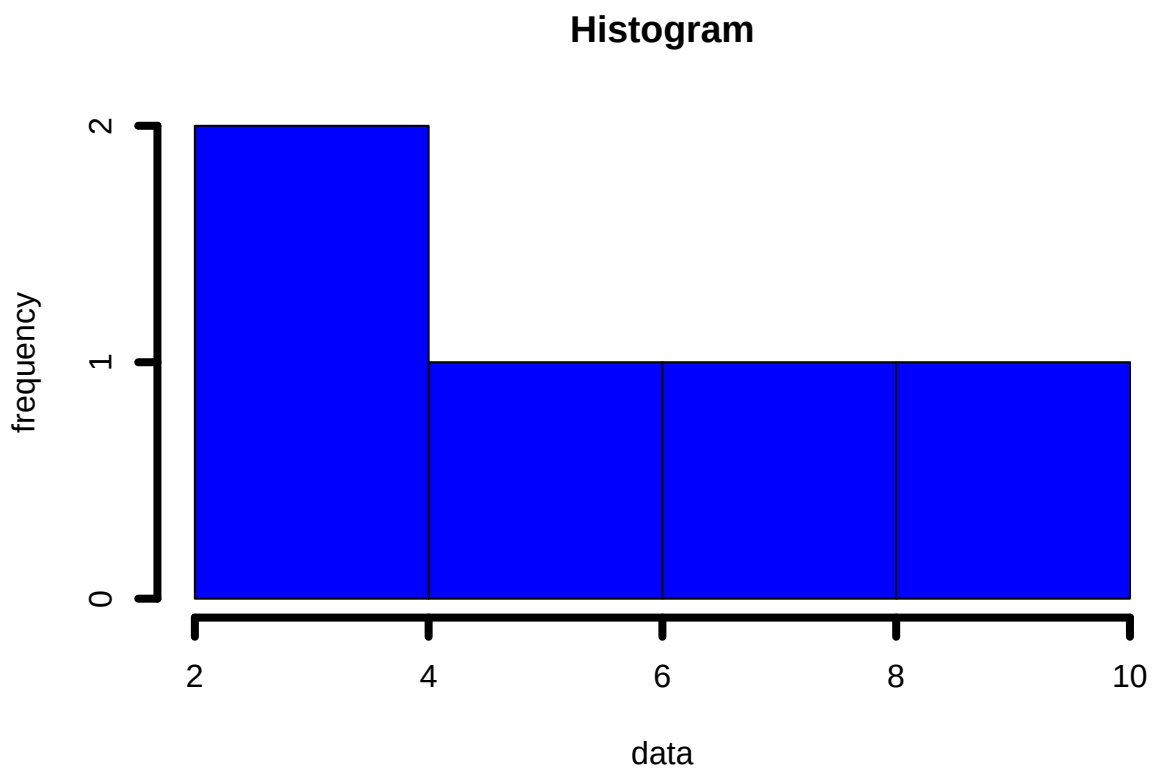
```r
print(sd_value)
```

```
## [1] 3.098387
```

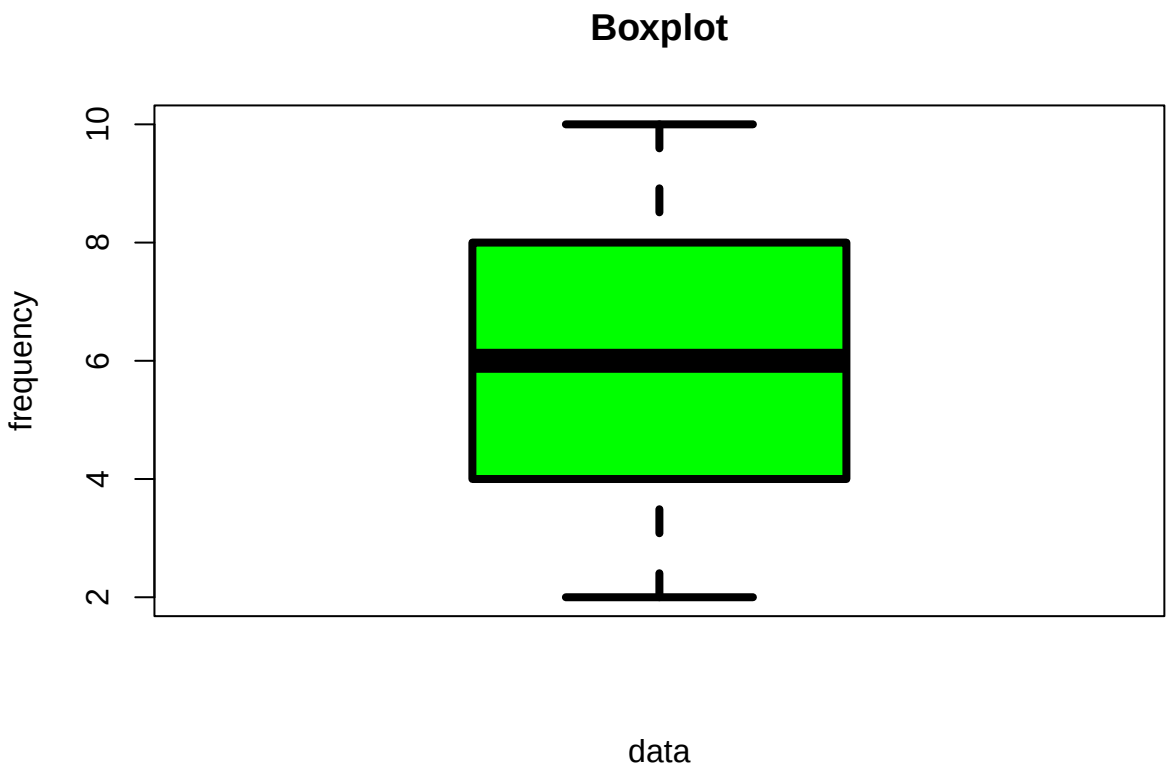
```r
# Example 2: Graphical Representation

   # - hist(x): Create a histogram of a numeric vector x.
   # - boxplot(x): Create a box plot of a numeric vector x.
   # - plot(x, y): Create a scatter plot of two numeric vectors x and y.
x <- c(2, 4, 6, 8, 10)

hist(x, main="Histogram", xlab = "data", ylab = "frequency", col="blue", lwd=4)
```

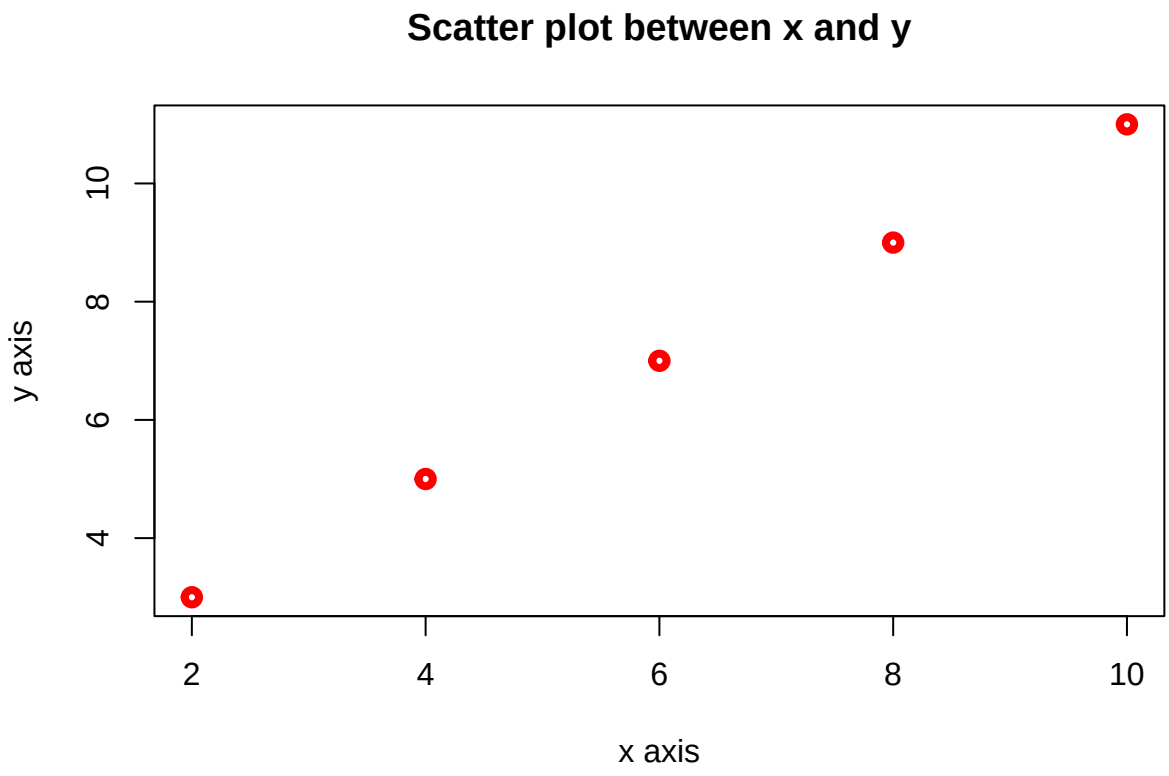



```r
boxplot(x, main="Boxplot", xlab = "data", ylab = "frequency", col="green", lwd=4)
```




```r
y <- c(3, 5, 7, 9, 11)

plot(x,y , main="Scatter plot between x and y", xlab = "x axis", ylab = "y axis", col="red", lwd=4)
```




```r
# 3. Inferential Statistics:
#    - Hypothesis Testing:
#      - t.test(x, y): Perform a t-test to compare the means of two samples x and y.
#      - chisq.test(x, y): Perform a chi-squared test for independence between two categorical variable
#
#    - Confidence Intervals:
#      - t.test(x)$conf.int: Calculate the confidence interval for a t-test on a sample x.
#      - prop.test(x, n)$conf.int: Calculate the confidence interval for a proportion test on a sample
#    - p-value:
#      - t.test(x)$p.value: Get the p-value for a t-test on a sample x.
#      - prop.test(x, n)$p.value: Get the p-value for a proportion test on a sample x out of n.

x <- c(2, 4, 6, 8, 10)
y <- c(1, 3, 5, 7, 9)
# Assuming a sample size of 20 for each success count
n <- c(10, 10, 10, 10, 10)
t_test_result <- t.test(x, y)
chisq_test_result <- chisq.test(x, y)
```

```
## Warning in chisq.test(x, y): Chi-squared approximation may be incorrect
```

```r
conf_interval_t_test <- t.test(x)$conf.int
conf_interval_prop_test <- prop.test(x, n)$conf.int
```

```
## Warning in prop.test(x, n): Chi-squared approximation may be incorrect
```

```r
p_value_t_test <- t.test(x)$p.value

p_value_prop_test <- prop.test(x, n)$p.value
```

```
## Warning in prop.test(x, n): Chi-squared approximation may be incorrect
```

```r
print(t_test_result)
```

```
##
##  Welch Two Sample t-test
##
## data:  x and y
## t = 0.5, df = 8, p-value = 0.6305
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.612008  5.612008
## sample estimates:
## mean of x mean of y
##         6         5
```

```r
print(chisq_test_result)
```

```
##
##  Pearson's Chi-squared test
##
## data:  x and y
## X-squared = 20, df = 16, p-value = 0.2202
```

```r
print(conf_interval_t_test)
```

```
## [1] 2.073514 9.926486
## attr(,"conf.level")
## [1] 0.95
```

```r
print(conf_interval_prop_test)
```

```
## NULL
```

```r
print(p_value_t_test)
```

```
## [1] 0.0132356
```

```r
print(p_value_prop_test)
```

```
## [1] 0.002243448
```