

Class Activity 14

Your name here

March 19 2024

Group Activity 1

a. Variance and Skewness

The variance of a random variable x is defined as:

$$\text{Var}(x) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

where $x_i = (\sum_i^n x_i)/n$ is the sample mean. Also, the skewness of the random variable x is defined as:

$$\text{Skew}(x) = \frac{\frac{1}{n-2} \left(\sum_{i=1}^n (x_i - \bar{x})^3 \right)}{\text{Var}(x)^{3/2}}$$

Please write functions to calculate the variance and skewness of $\{12, 45, 54, 34, 56, 30, 67, \text{NA}\}$.

```
x <- c(12, 45, 54, 34, 56, 30, 67, NA)
```

```
# function to calculate the variance of a vector
var <- function(x){
  x <- na.omit(x) # omit NA values
  sum((x - mean(x)) ^ 2) / (length(x) - 1)
}
```

```
var(x)
[1] 346.619
```

```
# function to calculate the skewness of a vector
skewness <- function(x) {
  x <- na.omit(x) # omit NA values
  sum((x - mean(x)) ^ 3) / ((length(x) - 2) * var(x) ^ (3 / 2))
}
```

```
skewness(x)
[1] -0.3930586
```

b. Conditions and breaks

Create a function that iterates through a numeric vector and stops when it encounters the first negative number, returning the position of that negative number. If there are no negative numbers in the vector, the function should return a message stating that there are no negative numbers.

```
find_first_negative <- function(x) {  
  for (i in seq_along(x)) {  
    if (x[i] < 0) {  
      return(paste("The first negative number is at position", i))  
      break  
    }  
  }  
  return("There are no negative numbers in the vector")  
}
```

```
test_vector <- c(5, 12, -7, 20, 15)  
find_first_negative(test_vector)  
[1] "The first negative number is at position 3"
```

Group Activity 2

```
musical_instr_reviews <- read_csv("https://raw.githubusercontent.com/deepbas/statdatasets/main/musicreviews.csv")  
rename(musical_instr_reviews, ratingOverall = overall)  
glimpse(musical_instr_reviews)  
Rows: 10,261  
Columns: 3  
$ reviewerName <chr> "cassandra tu \"Yeah, well, that's just like, u...\", \"Ja~  
$ reviewText <chr> "not much to write about here but it does exactly what i~  
$ ratingOverall <dbl> 5, 5, 5, 5, 5, 5, 5, 3, 5, 5, 5, 4, 3, 5, 5, 2, 4, 5, 5, ~
```

a. Write a function to filter the dataset based on the provided rating:

```
filter_reviews_by_rating <- function(data, rating) {  
  data %>% filter(ratingOverall == rating)  
}
```

b. Write a function to process the text and remove stop words:

```
process_text <- function(data) {  
  data %>%  
    select(reviewText) %>%  
    unnest_tokens(output = word, input = reviewText) %>%  
    anti_join(stop_words)  
}
```

c. Write a function to join the processed text with sentiment data and create a word count table.

```
create_word_count_table <- function(data) {  
  data %>%  
    inner_join(get_sentiments("bing")) %>%
```

```

count(word, sentiment, sort = TRUE) %>%
reshape2::acast(word ~ sentiment, value.var = "n", fill = 0)
}

```

d. Create the final function that takes the rating and number of words as input arguments and returns a word cloud plot.

```

library(wordcloud)
library(reshape2)

word_cloud <- function(rating, num.words) {
  rating <- as.numeric(rating)
  num.words <- as.numeric(num.words)
  if (rating >= 1 & rating <= 5){
    my_cloud <- musical_instr_reviews %>%
      filter(ratingOverall == rating) %>%
      select(reviewText) %>%
      unnest_tokens(output=word,
                    input=reviewText) %>%
      anti_join(stop_words) %>%
      inner_join(get_sentiments("bing")) %>%
      count(word, sentiment, sort = TRUE) %>%
      reshape2::acast(word ~ sentiment, value.var = "n", fill = 0) %>%
      wordcloud::comparison.cloud(colors = c("blue","purple"),
                                scale = c(2,0.5),
                                max.words = num.words,
                                title.size = 2)
  }else warning(" Please enter a rating from 1 to 5")
}

word_cloud(rating = "4", num.words = 300)

```

negative



positive