Title – Quantitative Methods for Linguistics with R software

Author – Divya Rajan Kadav

Date – 23/10/2024

I started with **installing tidyverse, languageR,ggplot2** packages in RStudio.

Using the command

```
install.packages("languageR")

install.packages("tidyverse")

install.packages(“ggplot2”)
```

#For Setting directory

```
setwd("C:/Users/Acer/Downloads/Divya_2024")

> library(readr)
```

#To read night circus csv file

```
nightcircus <- read_csv("nightcircus.csv")

Rows: 120 Columns: 3
```

#To view night circus csv file

```
View(nightcircus)

a=(nightcircus)
```

OUTPUT

	words	word_length	word_type
1	the	3	f
2	circus	6	c
3	arrives	7	c
4	without	7	c
5	warning	7	c
6	no	2	f
7	announcements	13	c
8	precede	7	c
9	it	2	f
10	no	2	f
11	paper	5	c
12	notices	7	c
13	on	2	f
14	downtown	8	c
15	posts	5	c
16	and	3	f
17	billboards	10	c
18	no	2	f

Showing 1 to 18 of 120 entries, 3 total columns

#To know the average length

a<-nightcircus$word_length

> average_length=mean(a)

> print(average_length)

[1] 4.85

#middle value of word-length

> median_length=median(a)

> print(median_length)

[1] 4

#installation of the package “modeest”

> install.packages("modeest")

library(modeest)

#To find out the most common word-length

mode_length <- mlv(a, method = "mfv")

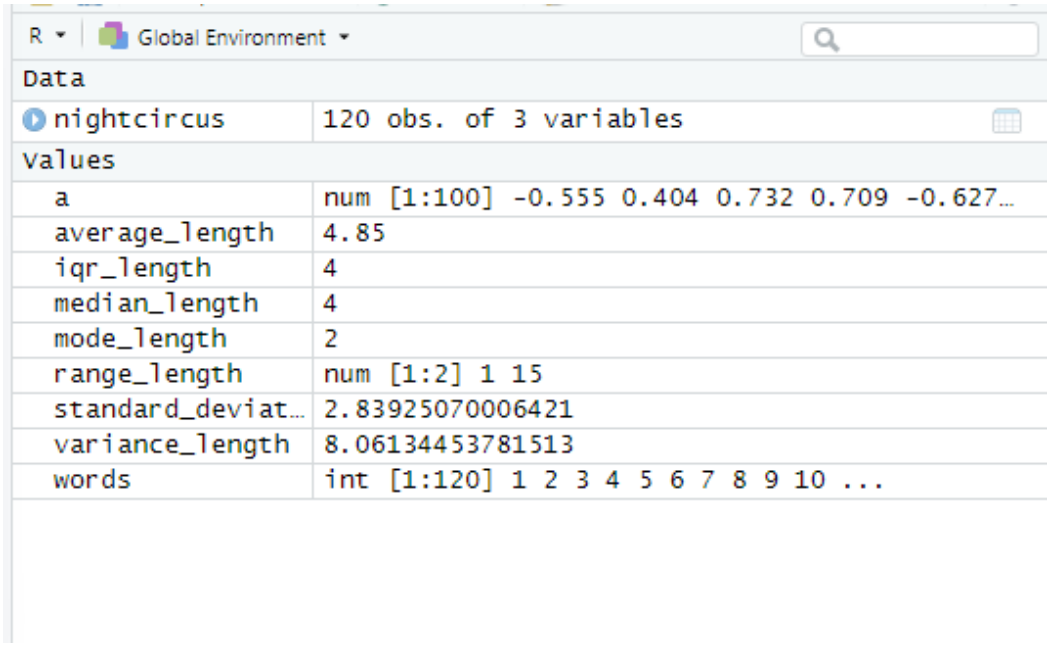
> range_length=range(a)

> iqr_length <- IQR(a)

> variance_length <- var(a)

> standard_deviation_length <- sd(a)

OUTPUT



#Ggplot using Histogram

ggplot(nightcircus, aes(x = word_type)) +

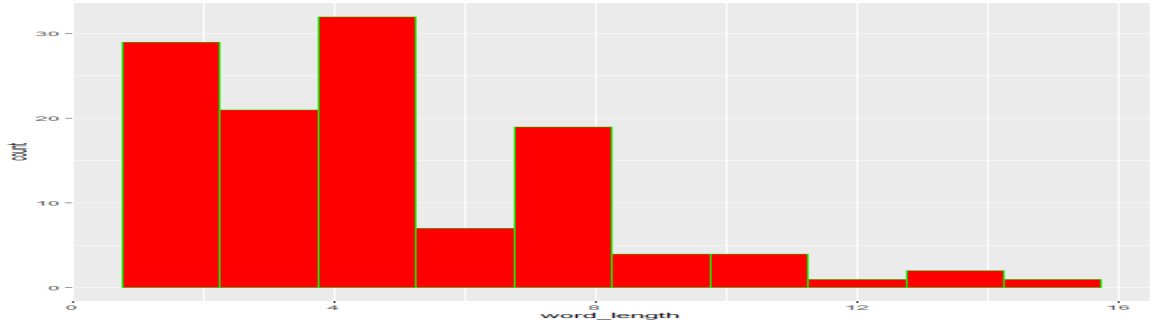
+ geom_bar(fill = "red", color = "green")

> ggplot(nightcircus, aes(x = word_type)) +

+ geom_bar(fill = "red", color = "green",binwidth=1)

ggplot(nightcircus,aes(x=word_length))+geom_histogram(fill="red",color="green",binwidth=1.5)

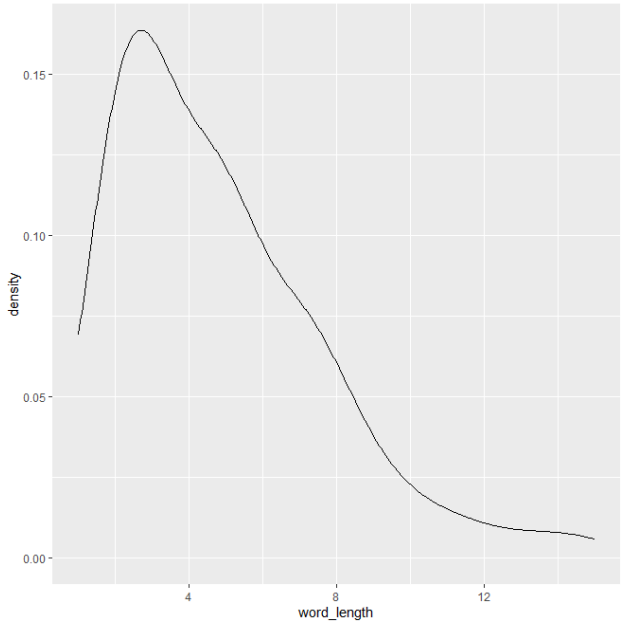
OUTPUT



#ggplot using Density Chart

ggplot(nightcircus,aes(x=word_length))+geom_density()

OUPUT

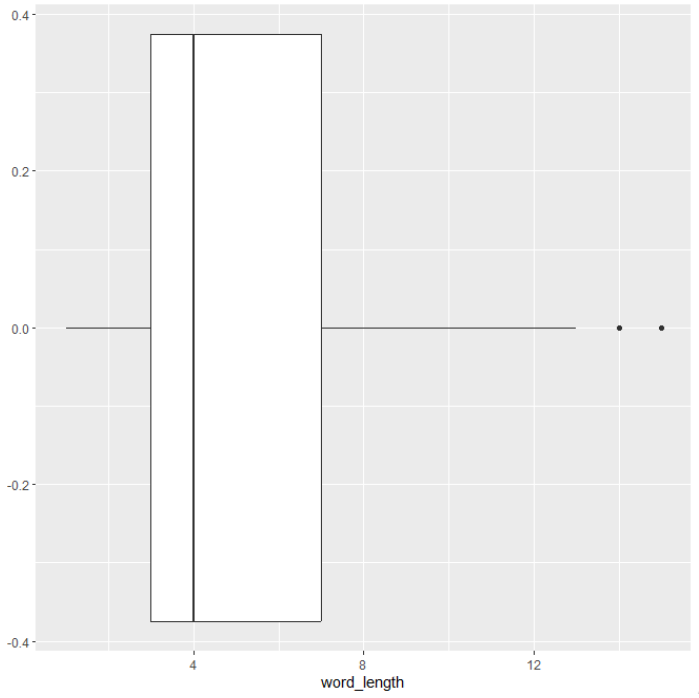


This data is positively skewed.

#To create a Box plot

ggplot(nightcircus,aes(x=word_length))+geom_boxplot()

OUTPUT



table(nightcircus$word_length)

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

1 28 21 12 20 7 12 7 4 2 2 1 1 1 1

#How many content words and function words are there in the data

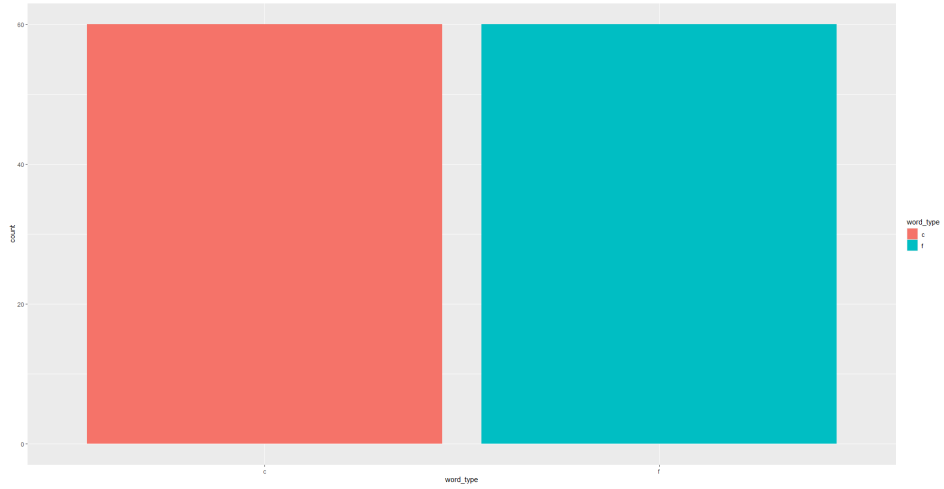
```
table(nightcircus$word_type)
```

```
 c  f
60 60
```

#To get the Bar Chart Diagram

```
ggplot(nightcircus,aes(x=word_type,fill=word_type))+geom_bar(position="dodge")
```

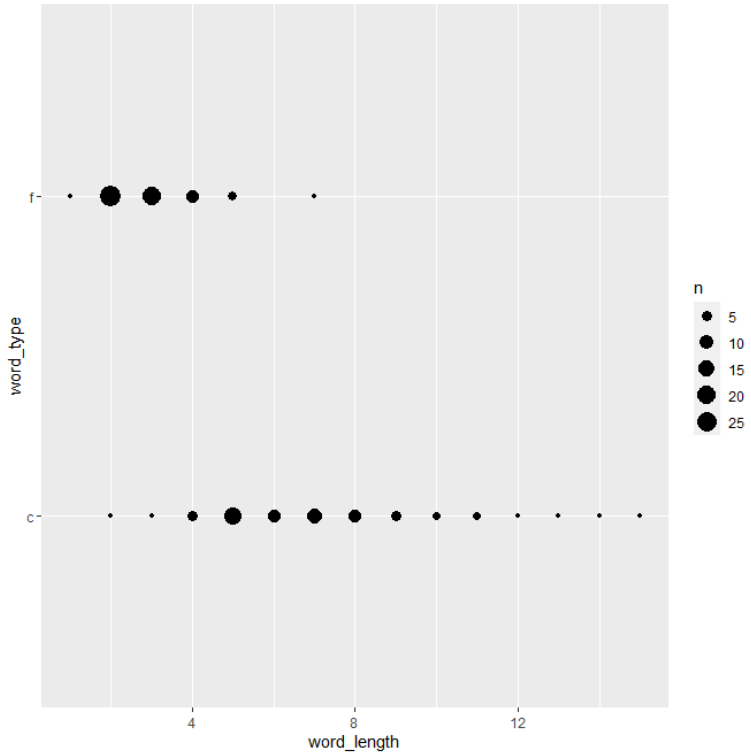
OUTPUT



#To get the Point-chart

```
ggplot(nightcircus, aes(x = word_length, y = word_type)) +
+     geom_count()
```

OUTPUT



Task 2: More on the Data "nightcircus"

1. From the given data, create two subset dataframes. One must contain ONLY the content words, and the other ONLY the function words. Save them as two separate datasheets in your folder.

```
> df1 <- nightcircus%>%filter(word_type=="f")
> View(df1)
```

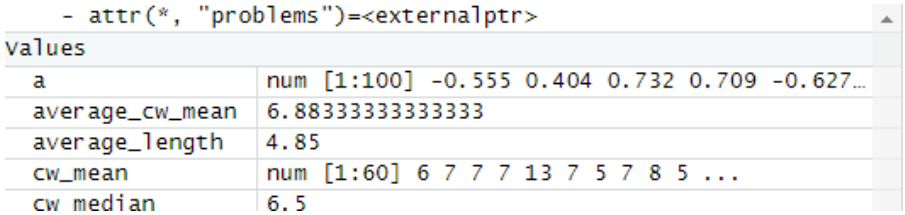
	words	word_length	word_type
1	the	3	f
2	no	2	f
3	it	2	f
4	no	2	f
5	on	2	f
6	and	3	f
7	no	2	f
8	or	2	f
9	in	2	f
10	it	2	f
11	is	2	f
12	there	5	f
13	when	4	f
14	it	2	f
15	was	3	f
16	not	3	f
17	the	3	f
18	are	3	f

df2 <- nightcircus%>%filter(word_type=="c")
> View(df2)

	words	word_length	word_type
1	circus	6	c
2	arrives	7	c
3	without	7	c
4	warning	7	c
5	announcements	13	c
6	precede	7	c
7	paper	5	c
8	notices	7	c
9	downtown	8	c
10	posts	5	c
11	billboards	10	c
12	mentions	8	c
13	advertisements	14	c
14	local	5	c
15	newspapers	10	c
16	simply	6	c
17	yesterday	9	c
18	towering	8	c

2. Find out the three measures of central tendency for the word-length of content words.

```
cw_mean <- df2$word_length
> average_cw_mean <- mean(cw_mean)
>
> #for median
> cw_median<-median(cw_mean)
> mode_cw_length <- mlv(cw_mean,method="mfv")
```



```
   - attr(*, "problems")=<externalptr>
```

Values	
a	num [1:100] -0.555 0.404 0.732 0.709 -0.627…
average_cw_mean	6.88333333333333
average_length	4.85
cw_mean	num [1:60] 6 7 7 7 13 7 5 7 8 5 ...
cw median	6.5

3. Find out the three measures of central tendency for the word-length of function words.


```
fw_mean <- df1$word_length
> average_fw_mean <- mean(fw_mean)
> fw_median <- median(fw_mean)
> mode_fw_mean <- mlv(fw_mean,method="mfv")
```

Environment | History | Connections | Tutorial

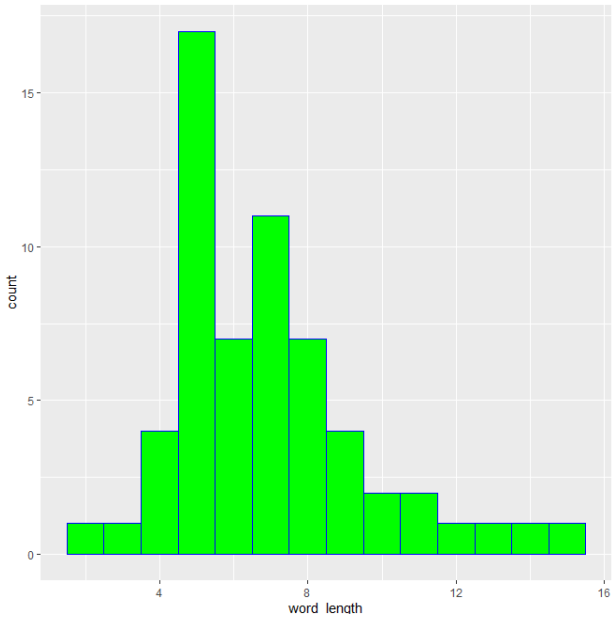
Import Dataset | 390 MiB | List

R | Global Environment

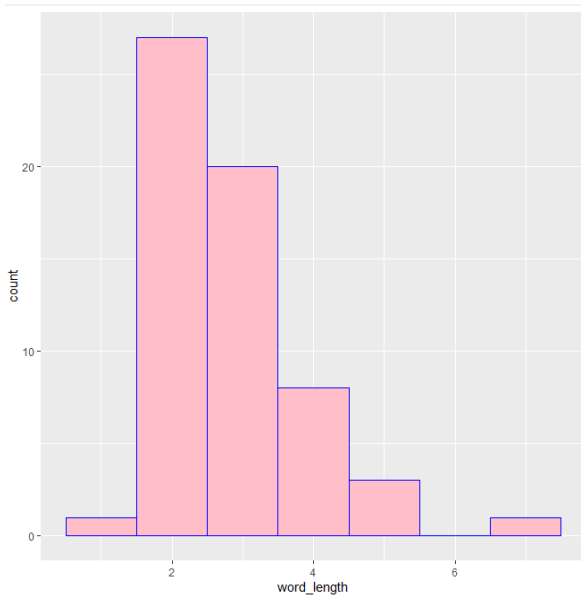
average_fw_mean	2.81666666666667
average_length	4.85
cw_mean	num [1:60] 6 7 7 7 13 7 5 7 8 5 ...
cw_median	6.5
fw_mean	num [1:60] 3 2 2 2 2 3 2 2 2 2 ...
fw_median	3
iqr_length	4
mean_cw	NA_real_
mean_data	NA_real_
median_length	4
mode_cw_length	5
mode_fw_mean	2
mode_length	2
range_length	num [1:2] 1 15
standard_deviat...	2.83925070006421
variance_length	8.06134453781513
words	int [1:120] 1 2 3 4 5 6 7 8 9 10 ...

4. How are each of these two data of word-length distributed? Draw histograms and/or line plots and respond.

ggplot(df2,aes(x=word_length))+geom_histogram(fill="green",color="blue",binwidth=1)



ggplot(df1,aes(x=word_length))+geom_histogram(fill="pink",color="blue",binwidth=1)



5. What are the standard deviations of the word-lengths in each of the datasets? Which of the two data you think exhibits more variation? How do you determine this?

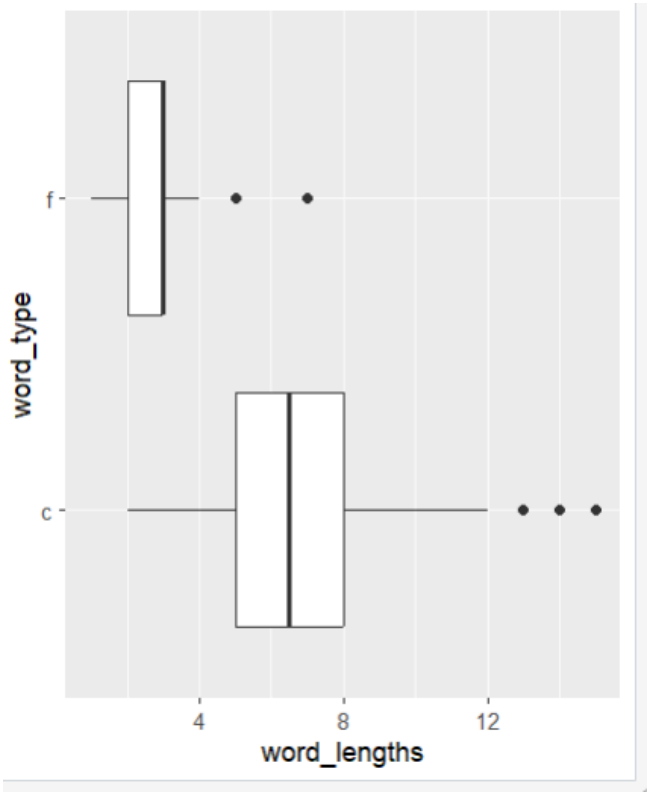
sd2 <- sd(cw_mean)
> sd1 <- sd(fw_mean)

mode_length	2
range_length	num [1:2] 1 15
sd1	1.04948198144538
sd2	2.59785874452349
standard_deviat…	2.83925070006421

6. Execute the following code and describe what output you get. What inferences can you draw from the output about the data?

```
> word_lengths <- data.frame(word_lengths = c(content-word-data$word_length,
function-word-data$word_length), word_type = c(rep("c", total-count), rep("f", total-count)))
>ggplot(word_lengths, aes(x = word_lengths, y = word_type)) + geom_boxplot()

word_lengths <- data.frame(word_lengths = c(df2$word_length, df1$word_length), word_type =
c(rep("c", 60), rep("f", 60)))
View(word_lengths)
ggplot(word_lengths, aes(x = word_lengths, y = word_type)) + geom_boxplot()
str(dativeSimplified)
```



(Replace the expressions in the light font with expressions/items that you have. For example, total count is the no. of observations in your respective dataset, and the expressions before $ are the names you assigned to the datasets of content and function words.)

str(dativeSimplified)

install.packages(languageR)

library(languageR)

str(dativeSimplified)

dativedata <- dativeSimplified %>% count(RealizationOfRec, Animacy)

npcount <- dativeSimplified%>%

filter(n="NP")%>%

count(animacy)

npcount <- dativeSimplified%>%

+ filter(n=="NP")%>%

```
+ count(animacy)

view(dativeSimplified)

str(dativeSimplified)

npcount <- dativeSimplified %>%

filter(RealizationOfRec == "NP") %>%

count(AnimacyOfRec)

View(npcount)

pp_counts <- dativeSimplified %>%

filter(RealizationOfRec == "PP") %>%

count(AnimacyOfRec)

View(pp_counts)

np_total <- sum(np_counts$n)

np_total <- sum(npcount$n)

pp_total <- sum(pp_counts$n)

np_animate_prop <- np_counts$n[np_counts$AnimacyOfRec == "animate"]

np_animate_prop <- np_counts$n[npcount$AnimacyOfRec == "animate"]

np_animate_prop <- npcount$n[npcount$AnimacyOfRec == "animate"]

pp_animate_prop <- pp_counts$n[pp_counts$AnimacyOfRec == "animate"]

ggplot(dativeSimplified, aes(x = RealizationOfRec, fill = AnimacyOfRec)) + geom_bar(position =

"fill")

ggplot(dativeSimplified, aes(x = RealizationOfRec, fill =
AnimacyOfRec))+geom_bar(position="fill")+labs(y="proportion",fill="animacy")
```

	AnimacyOfRec	n
1	animate	301
2	inanimate	47

All the codes (Appendix)

```
install.packages("languageR")

setwd("C:/Users/HP/Downloads/Divya_2024")

library(readr)

nightcircus <- read_csv("nightcircus.csv")

View(nightcircus)

a=(nightcircus)

a<-nightcircus$word_length

average_length=mean(a)

median_length=median(a)

install.packages("modeest")

library(modeest)

mode_length <- mlv(a, method = "mfv")

range_length=range(a)
```

```
iqr_length <- IQR(a)

variance_length <- var(a)

standard_deviation_length <- sd(a)

#ggplot using Histogram

ggplot(nightcircus, aes(x = word_type)) +

+     geom_bar(fill = "red", color = "green")

install.packages("ggplot2")

library(ggplot2)

ggplot(nightcircus, aes(x = word_type)) +

+     geom_bar(fill = "red", color = "green")

ggplot(nightcircus,aes(x=word_type))+geom_bar(fill="red",color="green",binwidth=1)

ggplot(nightcircus,aes(x=word_length))+geom_density()

ggplot(nightcircus,aes(x=word_length))+geom_boxplot()

table(nightcircus$word_type)

ggplot(nightcircus,aes(x=word_type,fill=word_type))+geom_bar(position="dodge")

ggplot(nightcircus, aes(x = word_length, y = word_type))+geom_count

ggplot(nightcircus,aes(x=word_length,y=word_type))+geom_count()

df1 <- nightcircus%>%filter(word_type=="f")

library(dplyr)

df1 <- nightcircus%>%filter(word_type=="f")

View(df1)

df2 <- nightcircus%>%filter(word_type=="c")

View(df2)

cw_mean <- df2$word_length

average_cw_mean <- mean(cw_mean)

cw_median<-median(cw_median)

cw_median<-median(cw_mean)

mode_cw_length <- mlv(cw_mean,method="mfv")
```

```
fw_mean <- df1$word_length

average_fw_mean <- mean(fw_mean)

fw_median <- median(fw_mean)

mode_fw_mean <- mlv(fw_mean,method="mfv")

ggplot(df2,aes(x=word_length))+geom_histogram(fill="green",color="blue",binwidth=1)

ggplot(df1,aes(x=word_length))+geom_histogram(fill="pink",color="blue",binwidth=1)

sd2 <- sd(cw_mean)

sd1 <- sd(fw_mean)

word_lengths <- data.frame(word_lengths = c(contentdata$word_length,
functiondata$word_length), word_type = c(rep("c", 60), rep("f", 60)))

word_lengths <- data.frame(word_lengths = c(df2$word_length, df1$word_length),
word_type = c(rep("c", 60), rep("f", 60)))

View(word_lengths)

ggplot(word_lengths, aes(x = word_lengths, y = word_type)) + geom_boxplot()

str(dativeSimplified)

install.packages(languageR)

library(languageR)

str(dativeSimplified)

dativedata <- dativeSimplified %>% count(RealizationOfRec, Animacy)

npcount <- dativeSimplified%>%

filter(n="NP")%>%

count(animacy)

npcount <- dativeSimplified%>%

+ filter(n=="NP")%>%

+ count(animacy)

view(dativeSimplified)

str(dativeSimplified)

npcount <- dativeSimplified %>%

filter(RealizationOfRec == "NP") %>%
```

```
count(AnimacyOfRec)

View(npcount)

pp_counts <- dativeSimplified %>%

filter(RealizationOfRec == "PP") %>%

count(AnimacyOfRec)

View(pp_counts)

np_total <- sum(np_counts$n)

np_total <- sum(npcount$n)

pp_total <- sum(pp_counts$n)

np_animate_prop <- np_counts$n[np_counts$AnimacyOfRec == "animate"]

np_animate_prop <- np_counts$n[npcount$AnimacyOfRec == "animate"]

np_animate_prop <- npcount$n[npcount$AnimacyOfRec == "animate"]

pp_animate_prop <- pp_counts$n[pp_counts$AnimacyOfRec == "animate"]

ggplot(dativeSimplified, aes(x = RealizationOfRec, fill = AnimacyOfRec)) +
geom_bar(position =

"fill")

ggplot(dativeSimplified, aes(x = RealizationOfRec, fill =
AnimacyOfRec))+geom_bar(position="fill")+labs(y="proportion",fill="animacy")

q()
```

References

- Class Notes and Discussion

To access all the files like R history, scan the QR.