Lab - 5

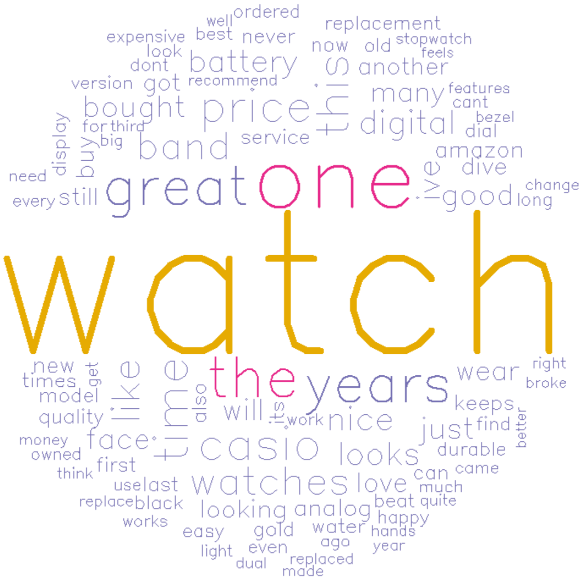
Group22 - tejma768, balra340

December 12, 2018

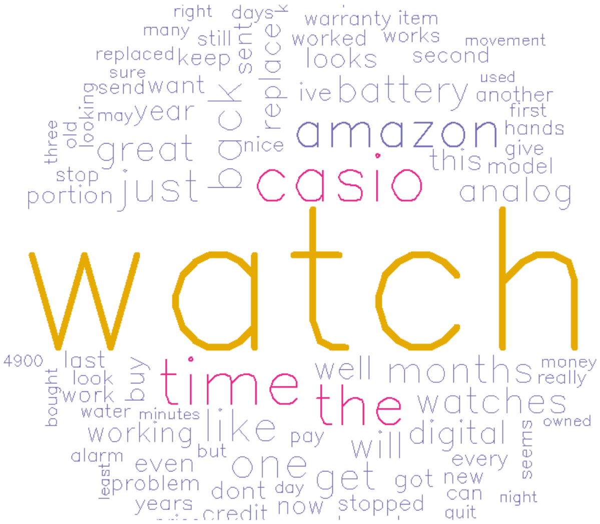
Assignment 1

1 Use R tools to create a word cloud corresponding to Five.txt and OneTwo.txt and adjust the colors in the way you like. Analyze the graphs.

```
library(tm)
library(tmap)
library(wordcloud)
library(RColorBrewer)

data1 <- read.table("Five.txt", header=F, sep='\n')
data1$doc_id=1:nrow(data1)
colnames(data1)[1]<-"text"

mycorpus <- Corpus(DataframeSource(data1)) #Creating corpus (collection of text data)
mycorpus <- tm_map(mycorpus, removePunctuation)
mycorpus <- tm_map(mycorpus, function(x) removeWords(x, stopwords("english")))
tdm <- TermDocumentMatrix(mycorpus) #Creating term-document matrix
M <- as.matrix(tdm)

A <- sort(rowSums(M),decreasing=TRUE) #Sum up the frequencies of each word
p <- data.frame(word = names(A),freq=A) #Create one column=names, second=frequences
pal <- brewer.pal(6,"Dark2")
pal <- pal[-(1:2)] #Create palette of colors
v<-wordcloud(p$word,p$freq, scale=c(8,.3),min.freq=2,max.words=100, random.order=F,
     rot.per=.15, colors=pal, vfont=c("sans serif","plain"))
```

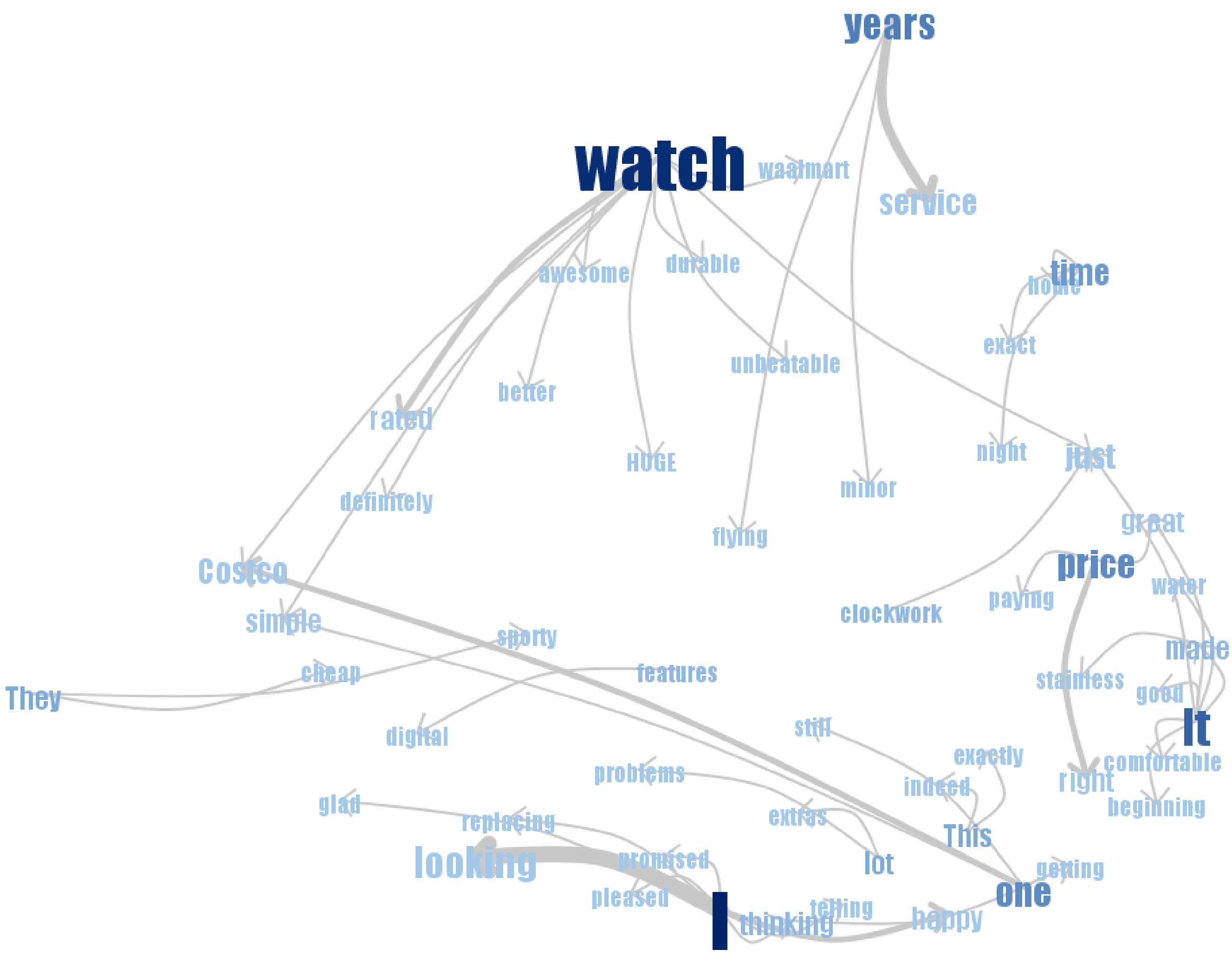


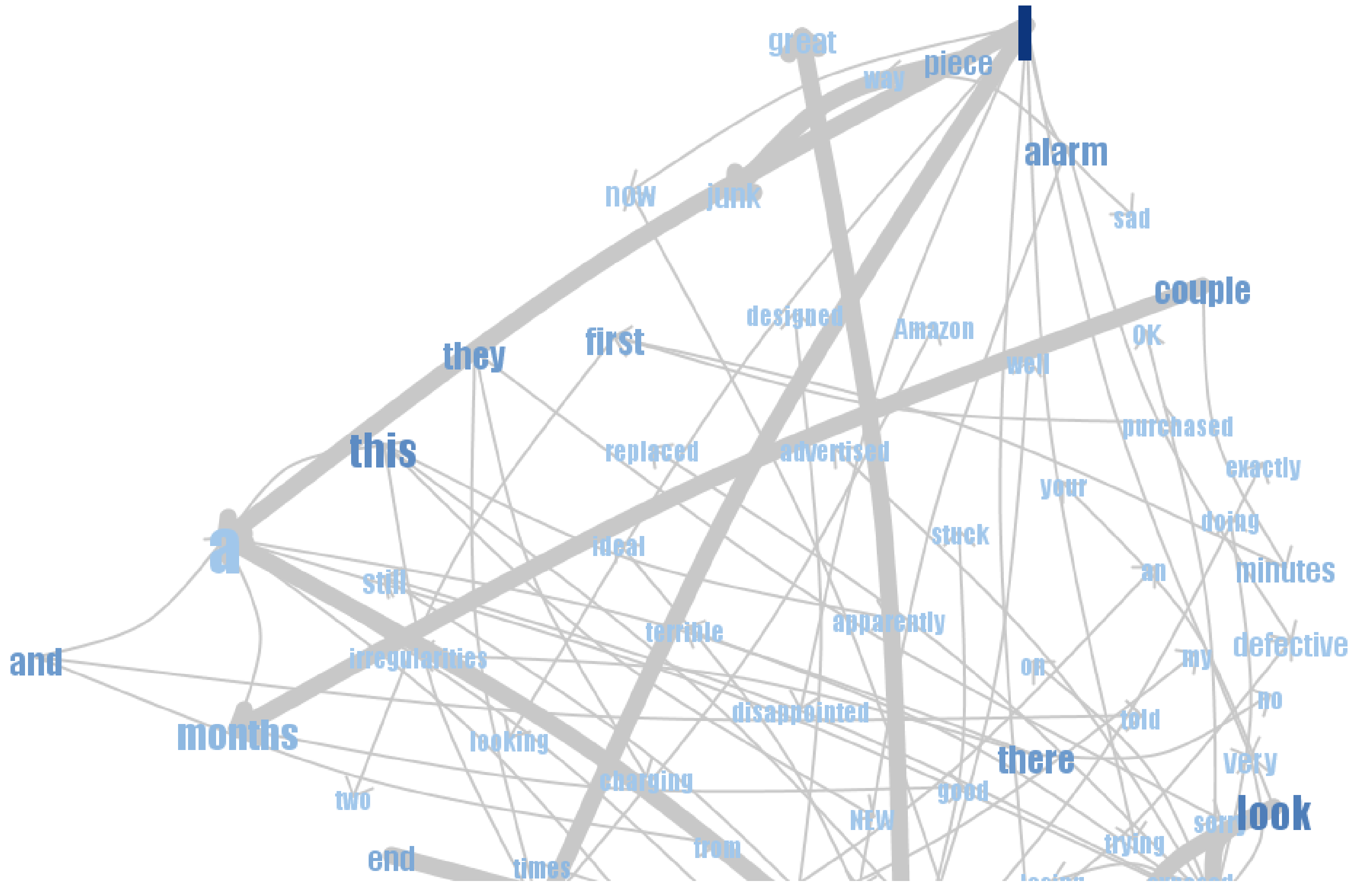
```
data2 <- read.table("OneTwo.txt", header=F, sep='\n')
data2$doc_id=1:nrow(data2)
colnames(data2)[1]<-"text"

mycorpus <- Corpus(DataframeSource(data2)) #Creating corpus (collection of text data)
mycorpus <- tm_map(mycorpus, removePunctuation)
mycorpus <- tm_map(mycorpus, function(x) removeWords(x, stopwords("english")))
tdm <- TermDocumentMatrix(mycorpus) #Creating term-document matrix
M1 <- as.matrix(tdm)

A1 <- sort(rowSums(M1),decreasing=TRUE) #Sum up the frequencies of each word
p1 <- data.frame(word = names(A1),freq=A1) #Create one column=names, second=frequences
colors1 <- brewer.pal(6,"Dark2")
colors1 <- colors1[-(1:2)] #Create palette of colors
v1<-wordcloud(p1$word,p1$freq, scale=c(8,.3),min.freq=2,max.words=100, random.order=F,
     rot.per=.15, colors=colors1, vfont=c("sans serif","plain"))
```



From the word cloud of Five.txt we can observe that every positive feature of the watch is being displayed, words like great, love, good, happy. And from the word cloud of OneTwo.txt we observe that negative feedback of the customer is being displayed, words like stopped, problem, return, never. And the font of the word being displayed is proportional to the frequency of the occurence of the word in the text file, like the word watch has appeared many times in the text file so it is displayed in the center with a bigger font.

2 Create the phrase nets for Five.Txt and One.Txt with connector words


years
watch
walmart
service
awesome
durable
time
home
exact
unbeatable
better
rated
HUGE
minor
night
just
definitely
flying
great
price
Costco
clockwork
paying
water
simple
sporty
made
cheap
features
stainless
good
They
It
digital
still
exactly
comfortable
problems
indeed
right
beginning
glad
replacing
extras
This
looking
promised
lot
getting
pleased
telling
one
I
thinking
happy


I
great
way
piece
alarm
now
junk
sad
couple
they
first
designed
Amazon
OK
well
purchased
this
replaced
advertised
exactly
your
doing
a
ideal
stuck
an
minutes
still
terrible
apparently
and
irregularities
on
my
defective
no
disappointed
told
months
looking
there
very
two
charging
good
NEW
trying
look
end
times
from

3 3.1 Which properties of this watch are mentioned mostly often? Ans: Properties of the watch that are being mentioned are that it is waterproof, made up of stainless material, digital features, sporty look, low price.

3.2 What are satisfied customers talking about? Ans: Satisfied customers are talking about the low price, good looks, and feelings of the people, like how happy they are that they braught this watch.

3.3 What are unsatisfied customers talking about? Ans: Unsatisfied customers are talking about disappointment, irregularities, defective alarm.

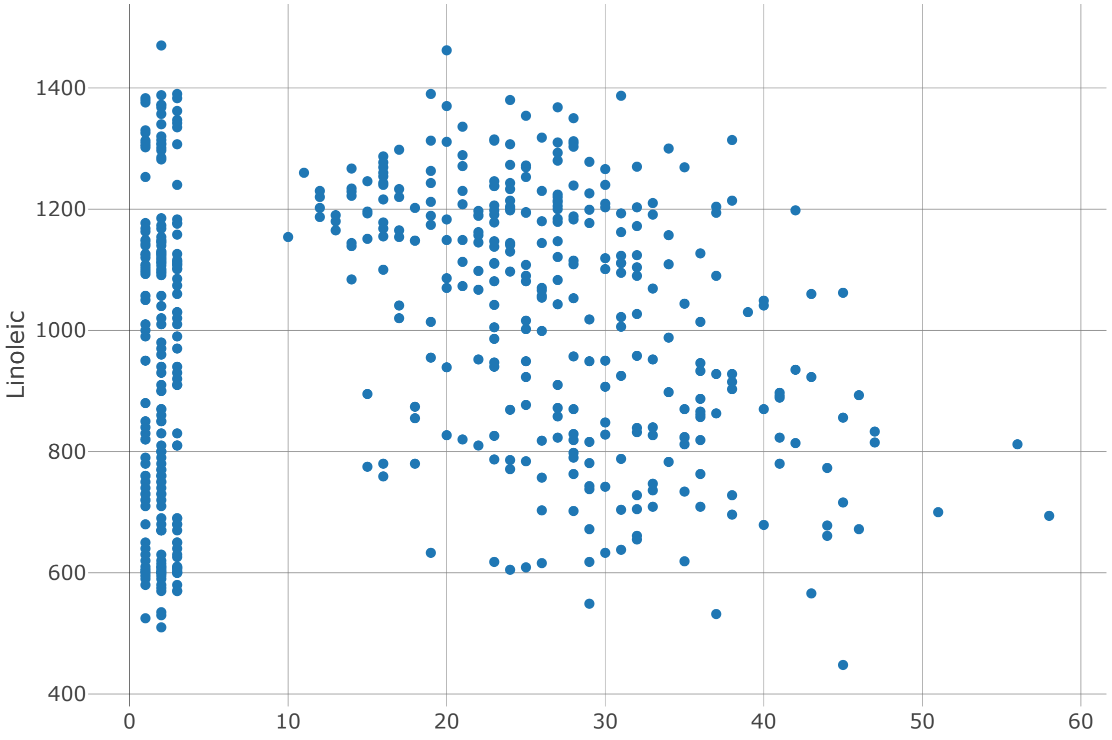
3.4 What are good and bad properties of the watch mentioned by both groups? Ans: Good Properties: Waterproof, good looks, durable, sporty, cheap, comfortable. Bad Properties: Defective Alarm, huge design.

3.5 Can you understand watch characteristics (like type of display, features of the watches) by observing these graphs? Ans: Yes by looking at the word clouds we can understand a few characteristics of the watch.

Assignment 2

1 Create an interactive scatter plot of the eicosenoic against linoleic. You have probably found a group of observations having unusually low values of eicosenoic. Hover on these observations to find out the exact values of eicosenoic for these observations.

```
library(plotly)
library(tidyr)
library(crosstalk)
library(GGally)

my_data <- read.csv("olive.csv")

d <- SharedData$new(my_data)

p <- plot_ly(d, x = ~eicosenoic, y = ~linoleic) %>%
   add_markers() %>% layout(xaxis = list(title="Eicosenoic"), yaxis = list(title="Linoleic"))
p
```

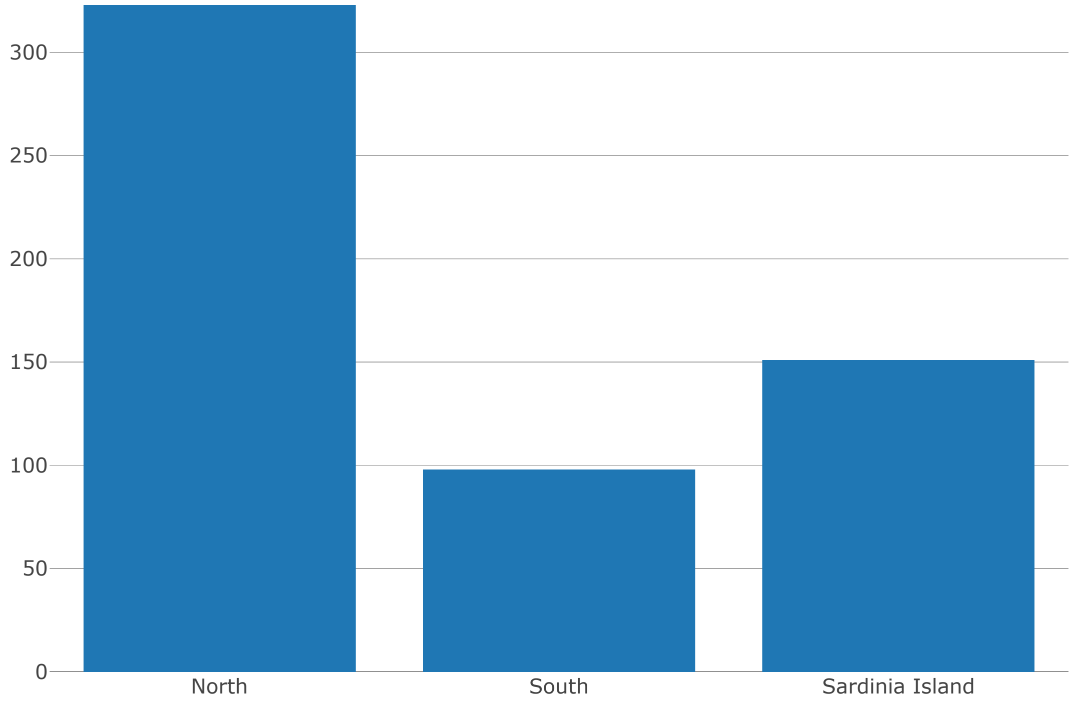


Eicosenoic

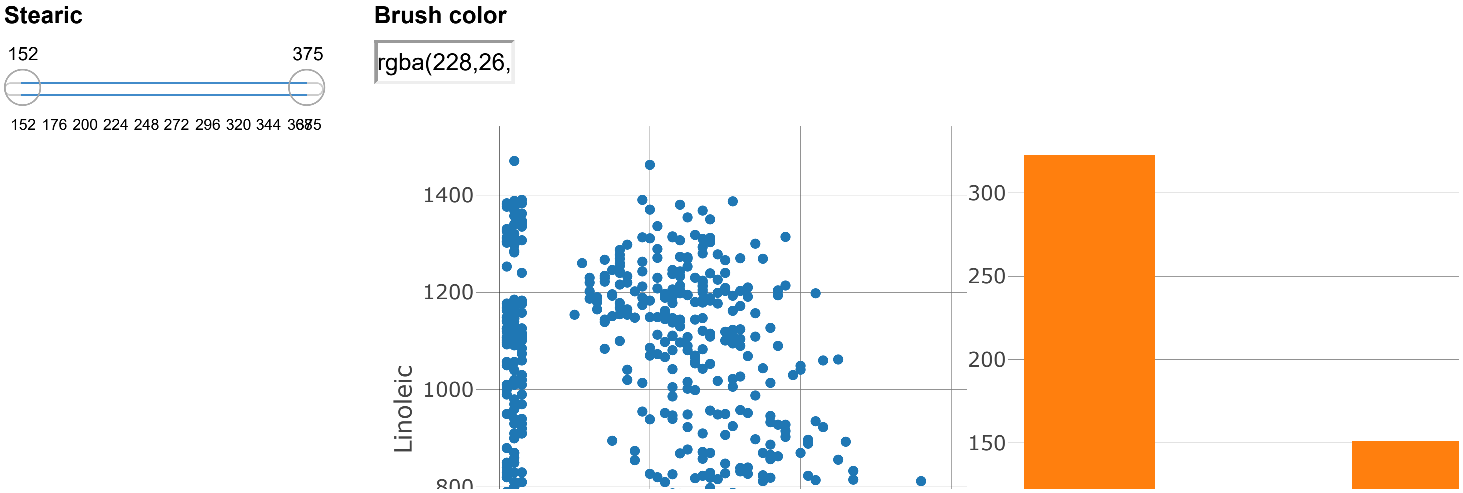
The lowest values of eicosenoic found by hovering on the obsrvations are 1,2, and 3

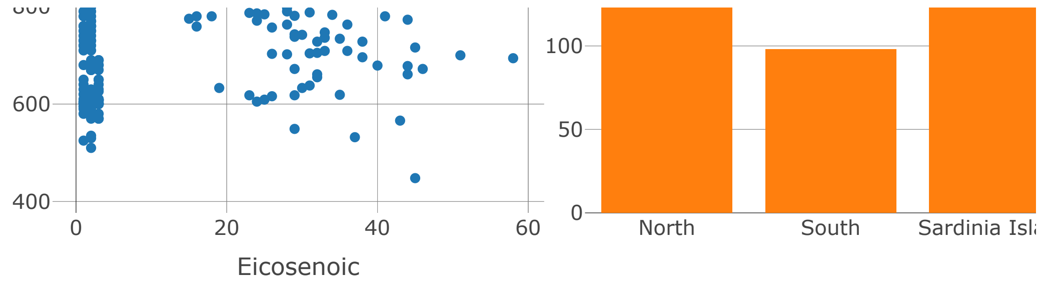
2 Link the scatterplot of (eicosenoic, linoleic) to a bar chart showing Region and a slider that allows to filter the data by the values of stearic. Use persistent brushing to identify the regions that correspond unusually low values of eicosenoic. Use the slider and describe what additional relationships in the data can be found by using it.

```
my_data$Region <- as.factor(my_data$Region)
levels(my_data$Region) <- c("North","South","Sardinia Island")

p1 <- plot_ly(d, x= my_data$Region)%>%add_histogram()%>%layout(barmode="overlay")

p1
```



```
bscols(widths=c(3, NA),filter_slider("stearic", "Stearic", d, ~stearic)
       ,subplot(p, p1,titleX = TRUE, titleY = TRUE)%>%
         highlight(on="plotly_select", dynamic=T, persistent = T, opacityDim = I(1))%>%hide_legend())
```

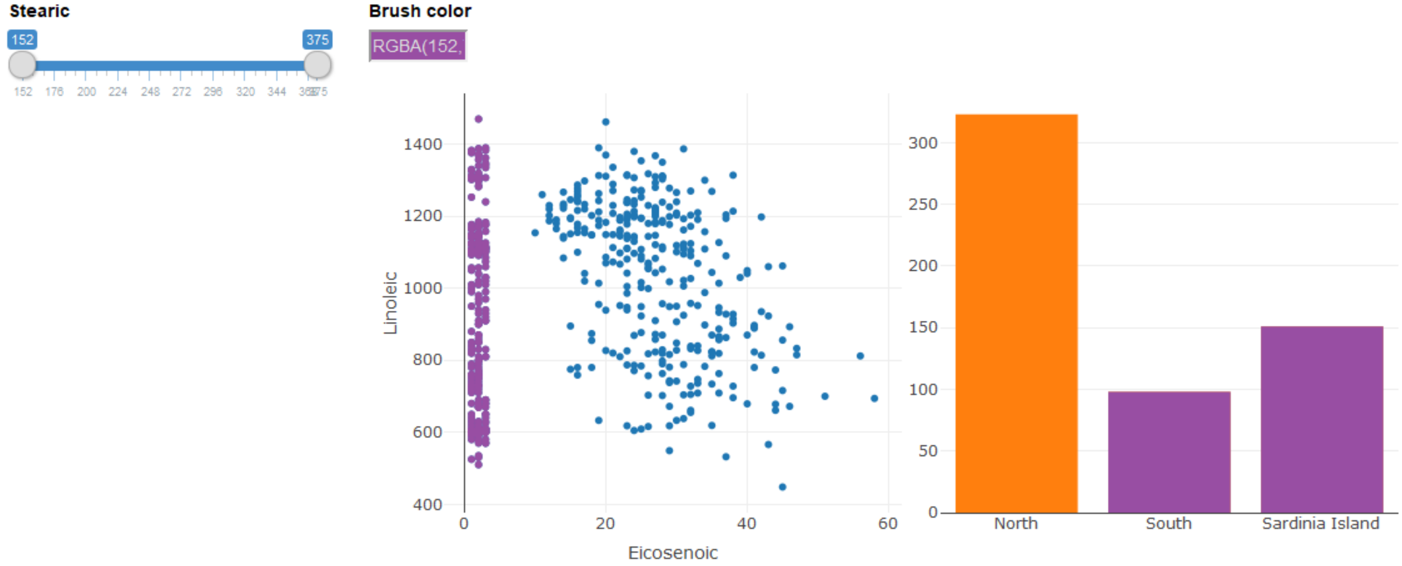




With persistent brushing found that the regions that correspond to unusually low values of Eicosenoic are South and Sardinia Island. Having highlighted the regions with different colour we can clearly distinguish the unusual low values of Eicosenoic. Values less than or equal to 1050 belong to Sardinia Island and values greater than 1050 belong to South region.

By using slider we see that South region disappears when Stearic value is less than 199 and more than 273, thus the Stearic range of South region is 199 - 273.

And when the slider is set less than or even more than this range of south region, most of the Linoleic values that are less than 1000 do not appear which can be seen from the figures attached.

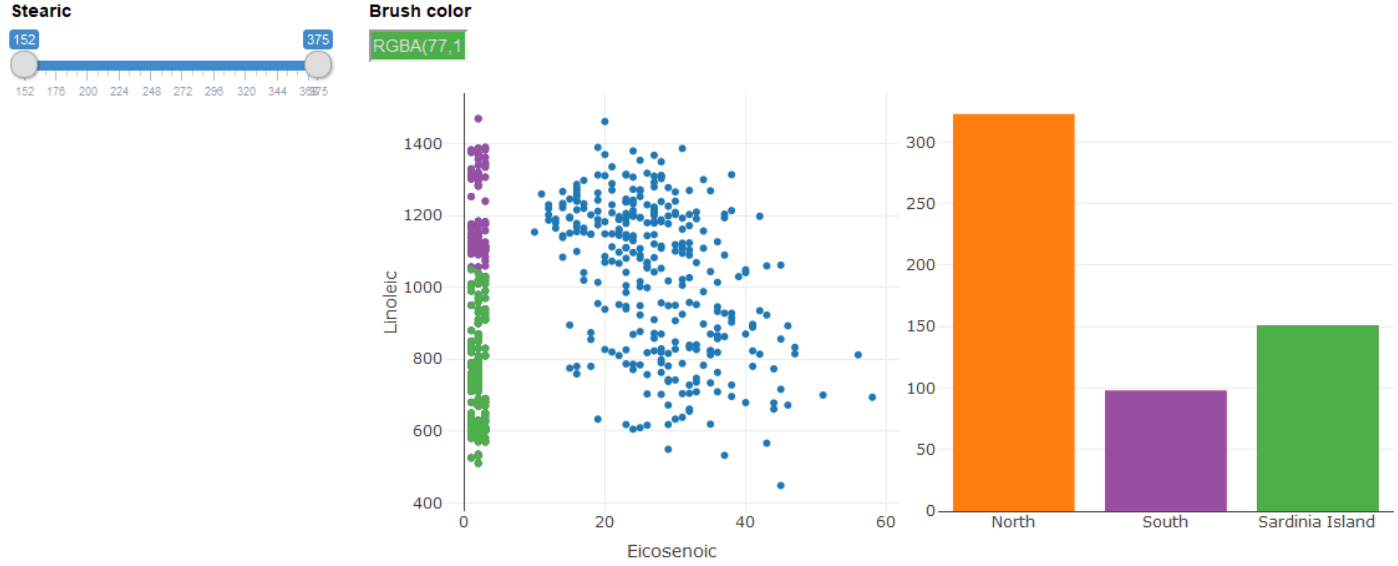


Stearic

152 375

Brush color

RGBA(152,

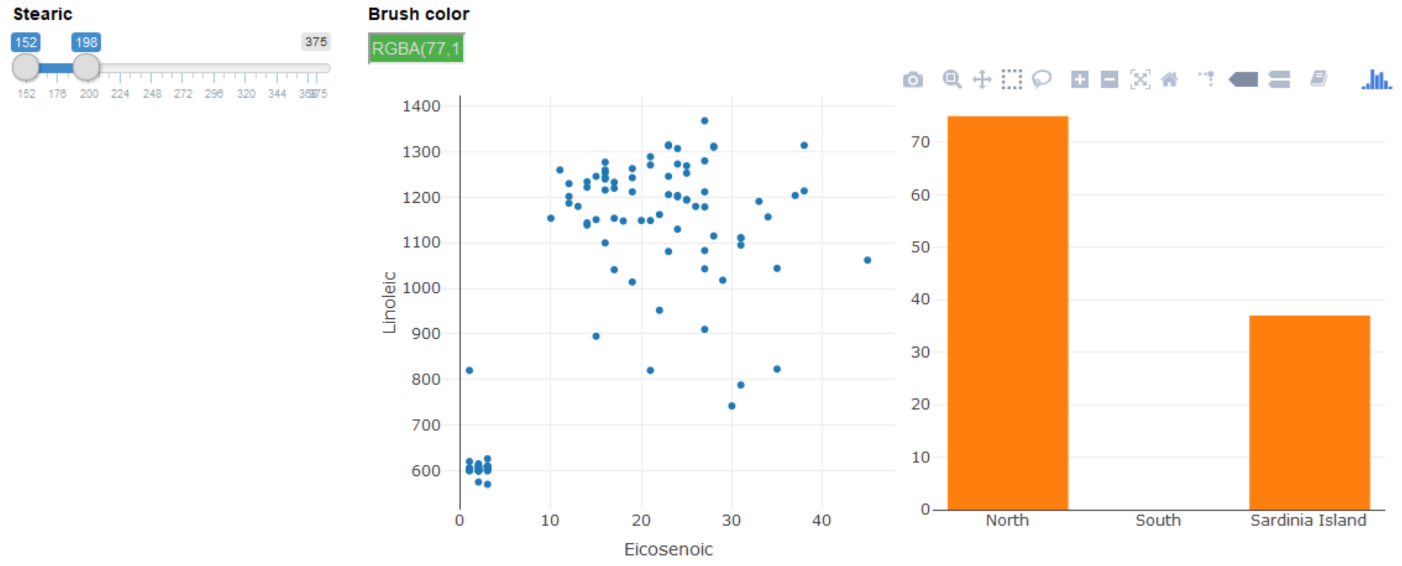


Stearic

152 375

Brush color

RGBA(77,1

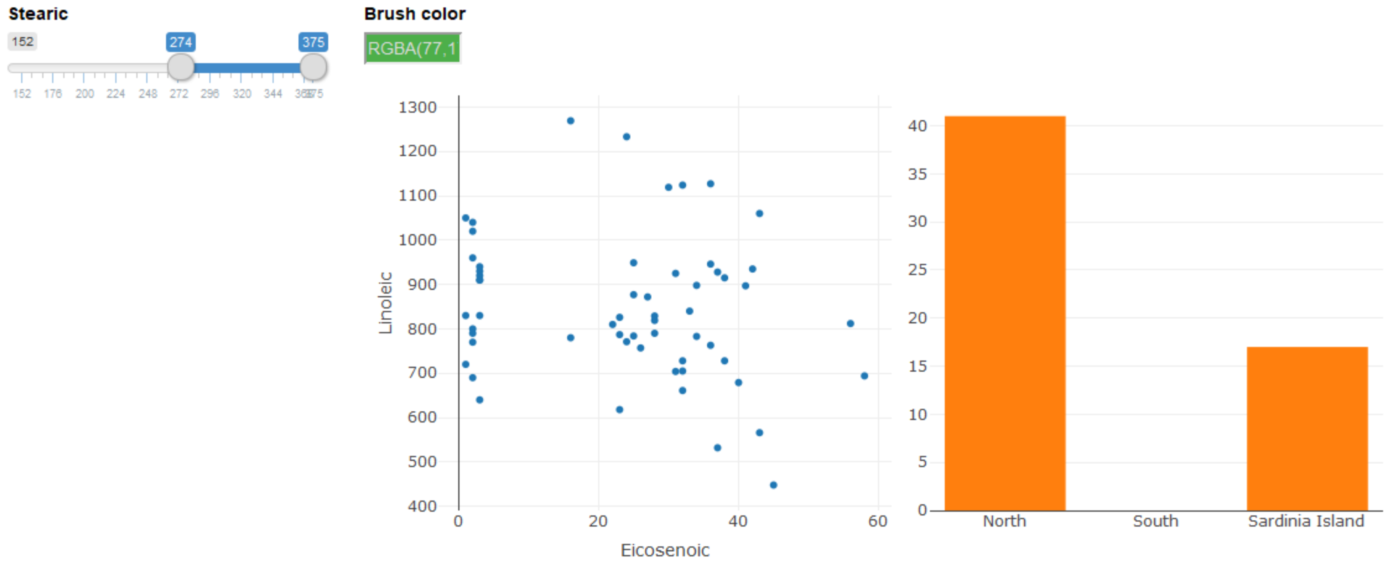


Stearic

152 198 375

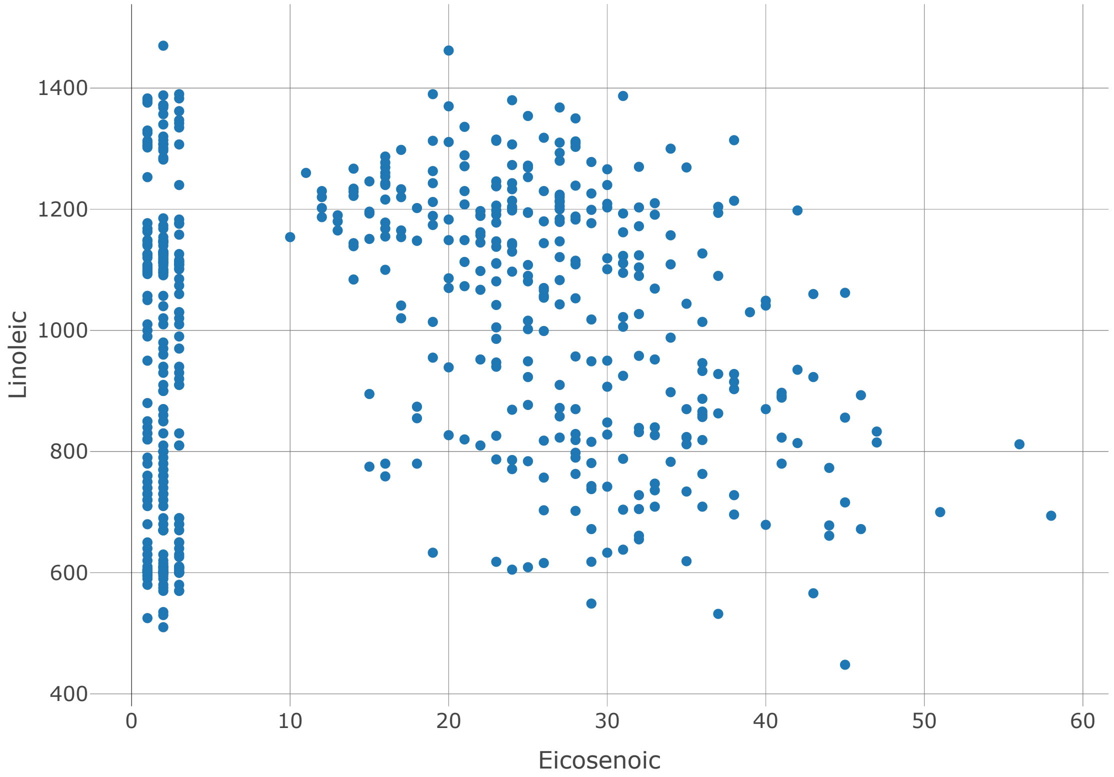
Brush color

RGBA(77,1

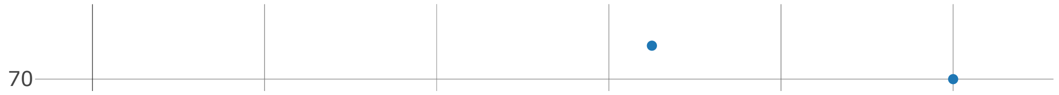


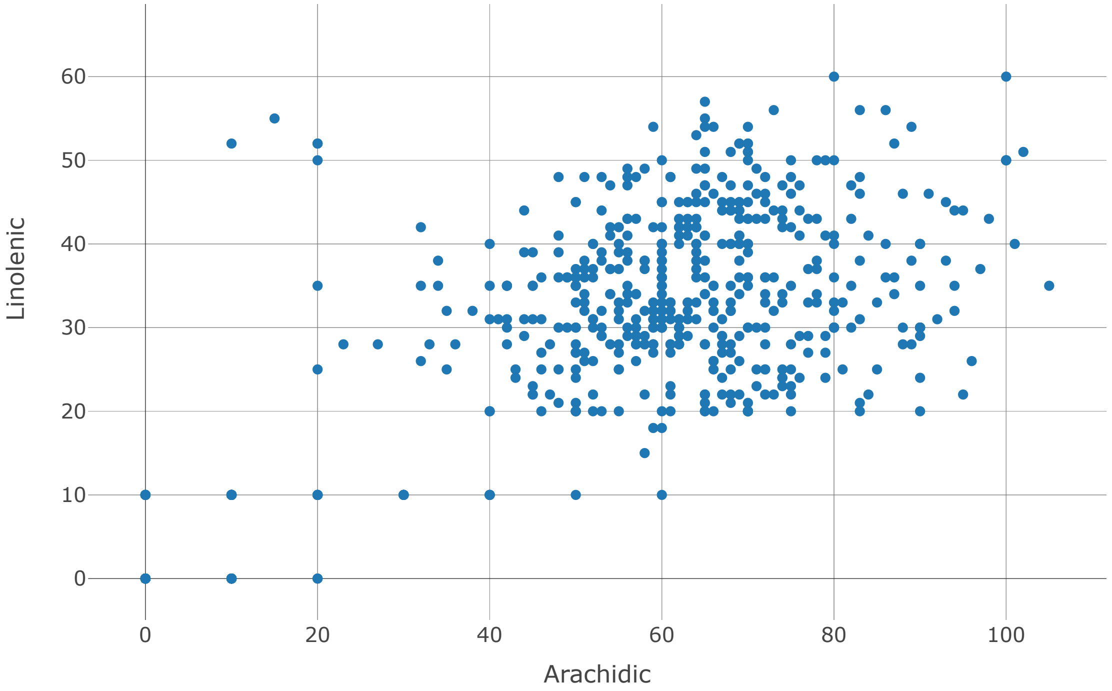
3 Create linked scatter plots eicosenoic against linoleic and arachidic against linolenic. Which outliers in (arachidic, linolenic) are also outliers in (eicosenoic, linoleic)? Are outliers grouped in some way? Use brushing to demonstrate your findings.

```
p <- plot_ly(d, x = ~eicosenoic, y = ~linoleic) %>%
   add_markers() %>% layout(xaxis = list(title="Eicosenoic"), yaxis = list(title="Linoleic"))
p
```



```
p2 <- plot_ly(data = d, x = ~arachidic, y = ~linolenic) %>% add_markers() %>%
      layout(xaxis = list(title="Arachidic"), yaxis = list(title="Linolenic"))
p2
```

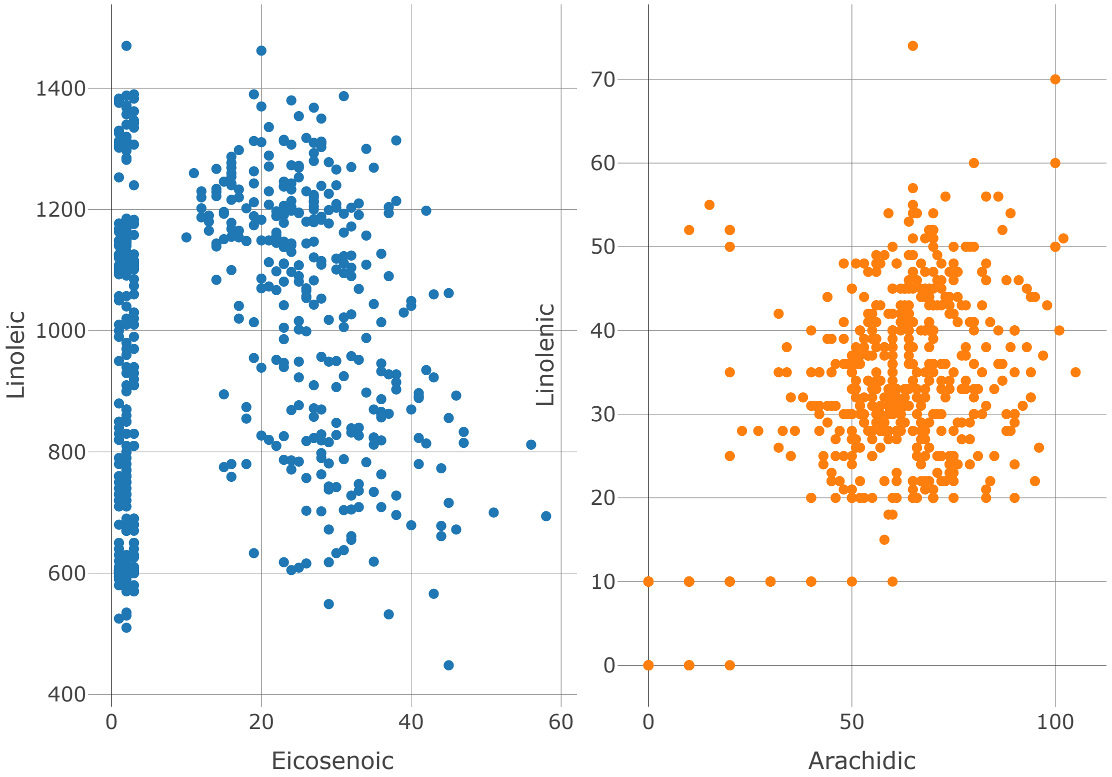




```
subplot(p, p2, titleX = TRUE, titleY = TRUE) %>%
highlight(on="plotly_select", dynamic=T, persistent=T, opacityDim = I(1)) %>% hide_legend()
```

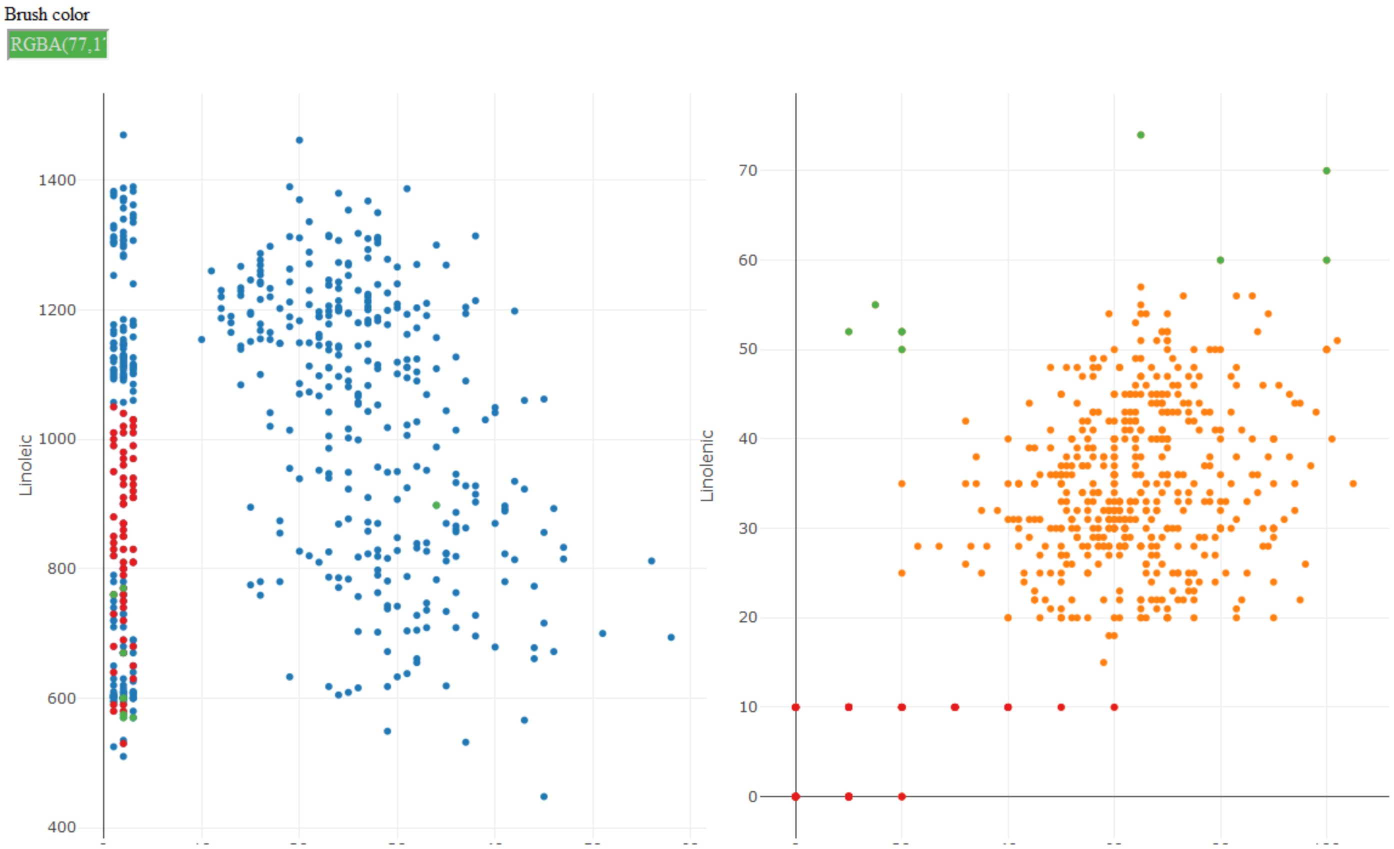
Brush color

rgba(228,26,



The observations that are highlighted in green are common in both the plots hence those are the outliers. Whereas the observations that are highlighted in red are also common in both the plots, but there is overplotting in Eicosenoic vs Linoleic plot hence they cannot be considerede as outliers.

Yes the outliers are grouped by having lower values of Eicosenoic and higher values of Linolenic, and low and high Arachidic values.



Brush color

RGBA(77,1

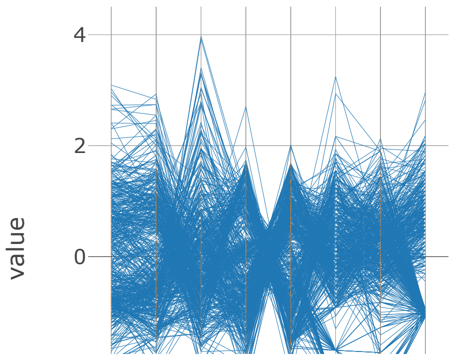
Result 2.2

4 Create a parallel coordinate plot for the available eight acids, a linked 3d-scatter plot in which variables are selected by three additional drop boxes and a linked bar chart showing Regions.

```
my_data <- read.csv("olive.csv")
d <- SharedData$new(my_data)
p<-ggparcoord(my_data, columns = c(4:11))
d<-plotly_data(ggplotly(p))%>%group_by(.ID)
d1<-SharedData$new(d, ~.ID, group="my_data")
p1<-plot_ly(d1, x=~variable, y=~value)%>%add_lines(line=list(width=0.3))%>%
  add_markers(marker=list(size=0.3),text=~.ID, hoverinfo="text")

olive2=my_data
olive2$.ID=1:nrow(my_data)
d2<-SharedData$new(olive2, ~.ID, group="my_data")
p2<-plot_ly(d2, x=~factor(Region) )%>%add_histogram()%>%layout(barmode="overlay")

ButtonsX=list()
for (i in 4:11){
  ButtonsX[[i-3]]= list(method = "restyle",
                        args = list( "x", list(my_data[[i]])),
                        label = colnames(my_data)[i])
}
ButtonsY=list()
for (i in 4:11){
  ButtonsY[[i-3]]= list(method = "restyle",
                        args = list( "y", list(my_data[[i]])),
                        label = colnames(my_data)[i])
}
ButtonsZ=list()
for (i in 4:11){
  ButtonsZ[[i-3]]= list(method = "restyle",
                        args = list( "z", list(my_data[[i]])),
                        label = colnames(my_data)[i])
}


p3 <- plot_ly(d2, x=~palmitic, y=~stearic, z=~oleic, alpha = 0.8) %>%
  add_markers() %>%
  layout(xaxis=list(title=""), yaxis=list(title=""), zaxis=list(title=""),
         title = "Select variable:",
         updatemenus = list(
           list(y=1.00, buttons = ButtonsX),
           list(y=0.85, buttons = ButtonsY),
           list(y=0.70, buttons = ButtonsZ)
         )  )

bscols(p1%>%highlight(on="plotly_select", dynamic=T, persistent = T, opacityDim = I(1))%>%
         hide_legend(),
       p3%>%highlight(on="plotly_click", dynamic=T, persistent = T)%>%hide_legend(),
       p2%>%highlight(on="plotly_click", dynamic=T, persistent = T)%>%hide_legend())
```

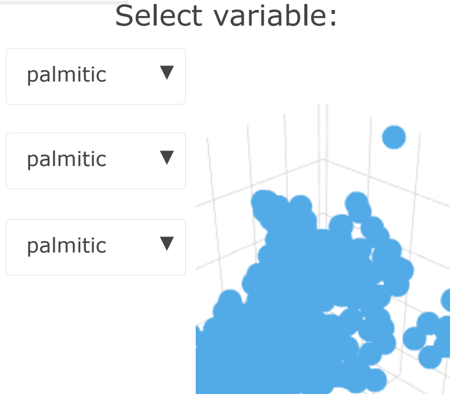
Brush color

rgba(228,26,



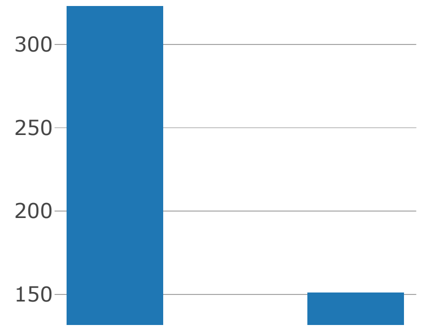
Brush color

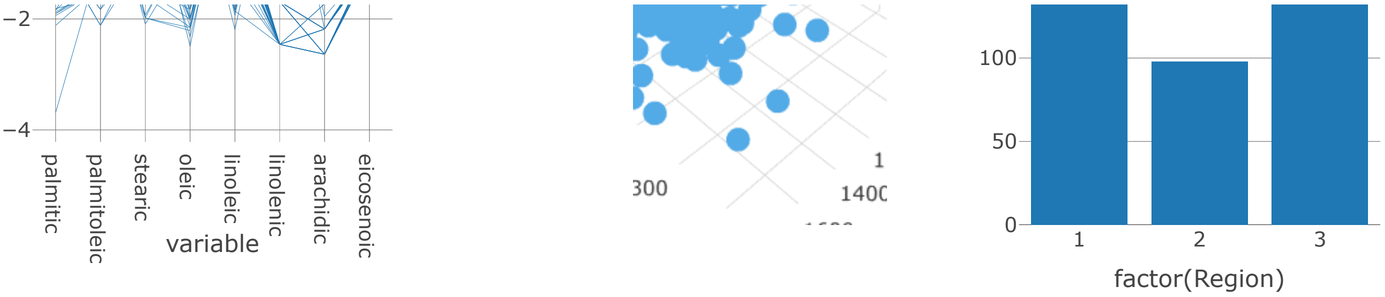
rgba(228,26,



Brush color

rgba(228,26,

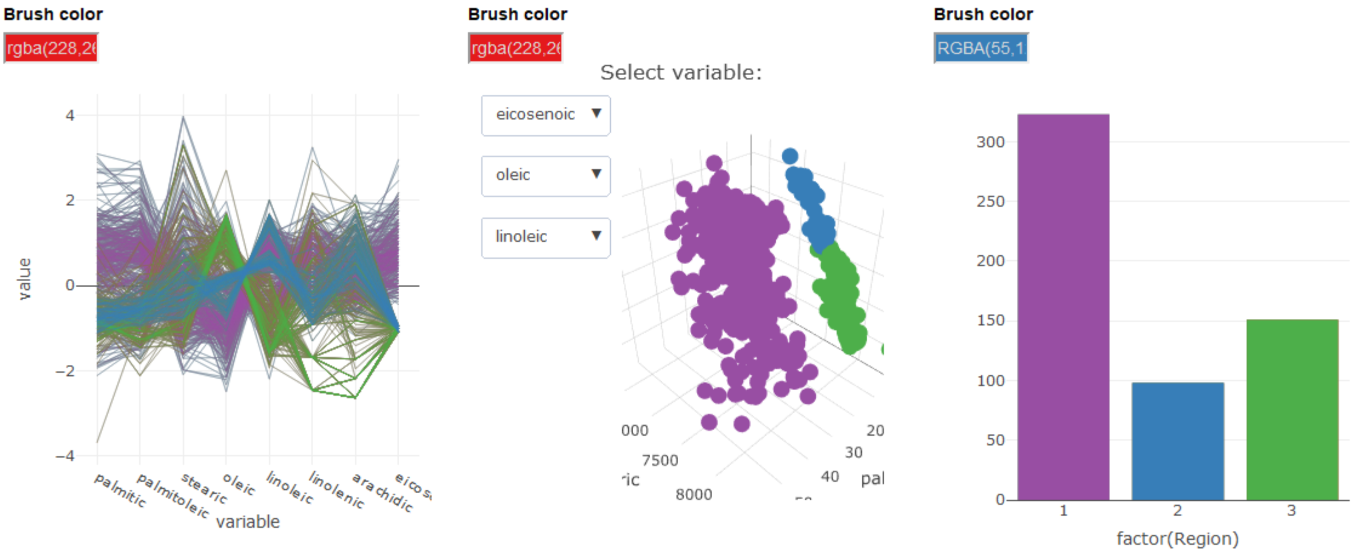




Each region has been marked by a different color by using persistent brushing. From the previous plots we got to know that eicosenoic values are the ones being used to differentiate between regions. And from the parallel coordinate plot we can see that oleic and linoleic follow a similar pattern, there seems to be a break or distance between their values. Hence Eicosenoic, Oleic and Linoleic can be called as the influential variables.

Yes, the parallele coordinate plot demonstrates that there are clusters among the observations that belong to the same region and those clusters are found in the south region, and the clusters are formed by the disjoint or break in the values of Oleic and Linoleic.

Each region does correspond to one cluster and that is seen by persistent brushing and by selecting three influential variables from the drop down.



Result 2.2

5 Think about which interaction operators are available in step 4 and what interaction operands they are be applied to. Which additional interaction operators can be added to the visualization in step 4 to make it even more efficient/flexible? Based on the analysis in the previous steps, try to suggest a strategy (or, maybe, several strategies) that would use information about the level of acids to discover which regions different oils comes from.

Interaction operators that are available here are:- -Navigation (panning, rotation, zooming) -Selection (highlighting) -Connecting (linked views) -Reconfiguring (changing Aesthetics) -Encoding (changing highlight colors)

A filter or a slider interation operator can be added to the visualization is step 4 to make it more efficient or flexible. Using which we could select values of influential variables.

We can use the following strategies,

1. If the ecosenic values are above 10 ,then it means it is from the north region.
2. We can also filter it the other way. Values of of ecosenice less than 10 and linolenic values less than 1000 might belong to sardinia island.