Lab2

Priya(priku577) and Andreas(andch552)

19 September 2018

Assignment 1

1

scatterplot in Ggplot2 that shows dependence of Palmitic on Oleic in which observations are colored by Linolenic

```
library(ggplot2)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

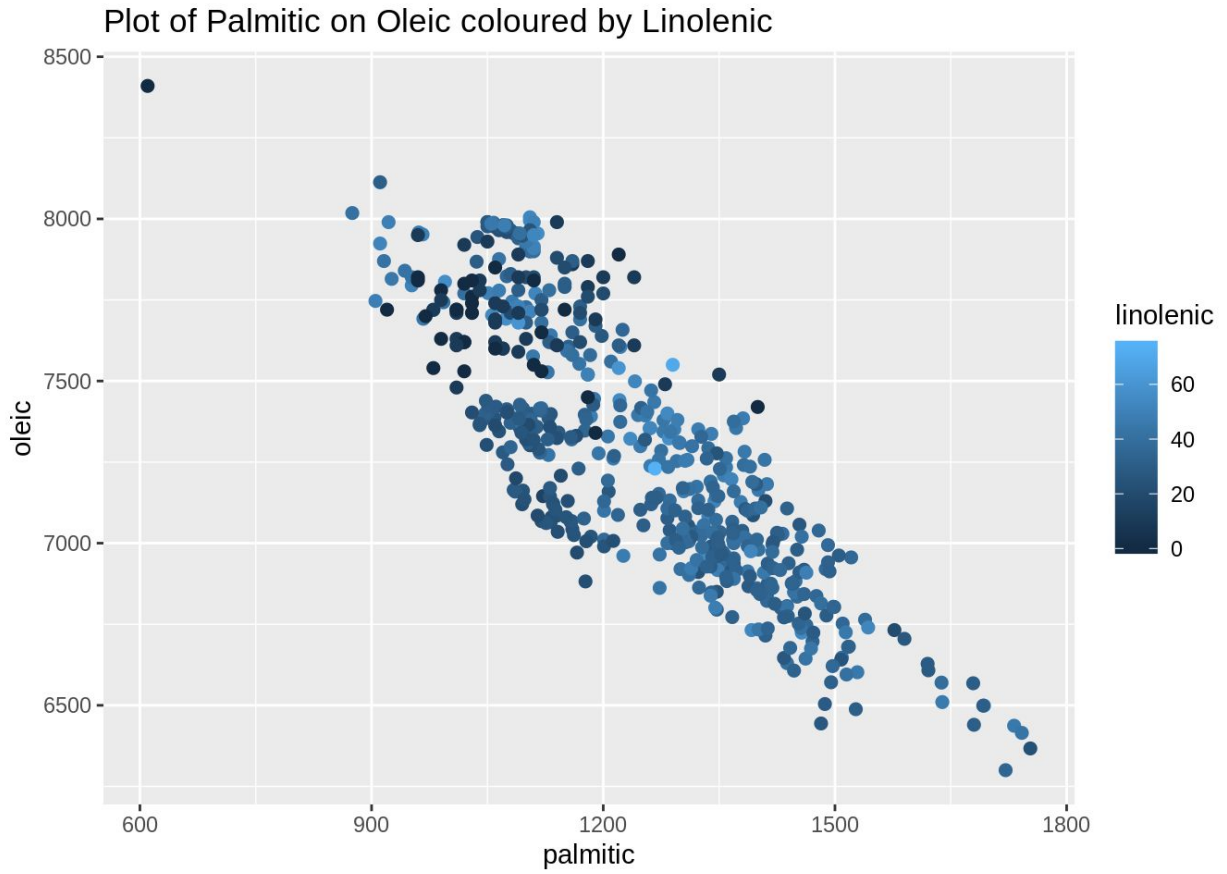
```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(MASS)
```

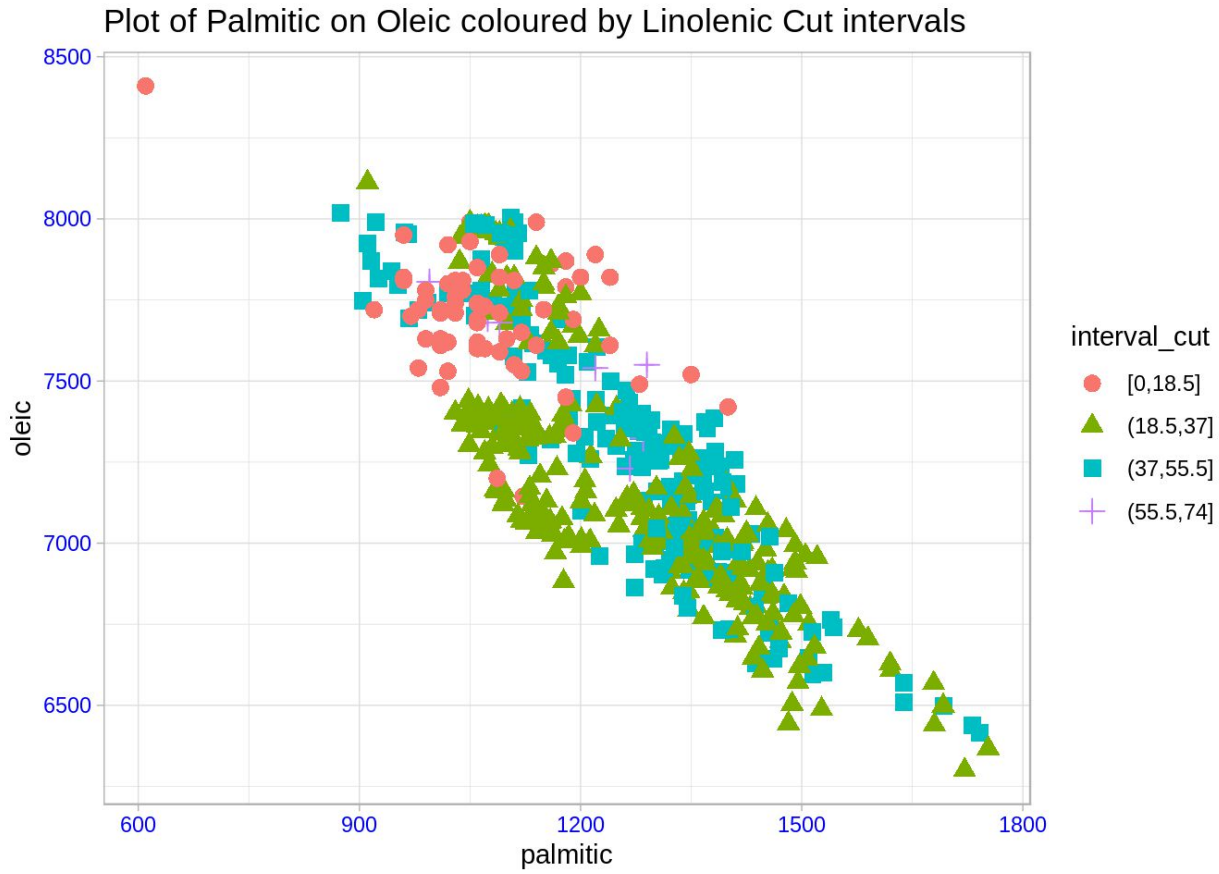
```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:plotly':
##
##     select
```

```
olive<-read.csv("olive.csv",header=TRUE, sep=",")



p1<-ggplot(olive,aes(palmitic,oleic,col=linolenic))+geom_point()+labs(x="palmitic",y="ole
ic")+geom_point(size=2)+ggtitle("Plot of Palmitic on Oleic coloured by Linolenic")
p1
```




```
interval_cut <- cut_interval(olive$linolenic, 4)
p2<-ggplot(olive,aes(palmitic,oleic,col=interval_cut,shape=interval_cut))+geom_point()+th
eme_light()+labs(x="palmitic",y="oleic")+geom_point(size=3)+ggtitle("Plot of Palmitic on
Oleic coloured by Linolenic Cut intervals")+theme(axis.text = element_text(colour = "blu
e"))
p2
```



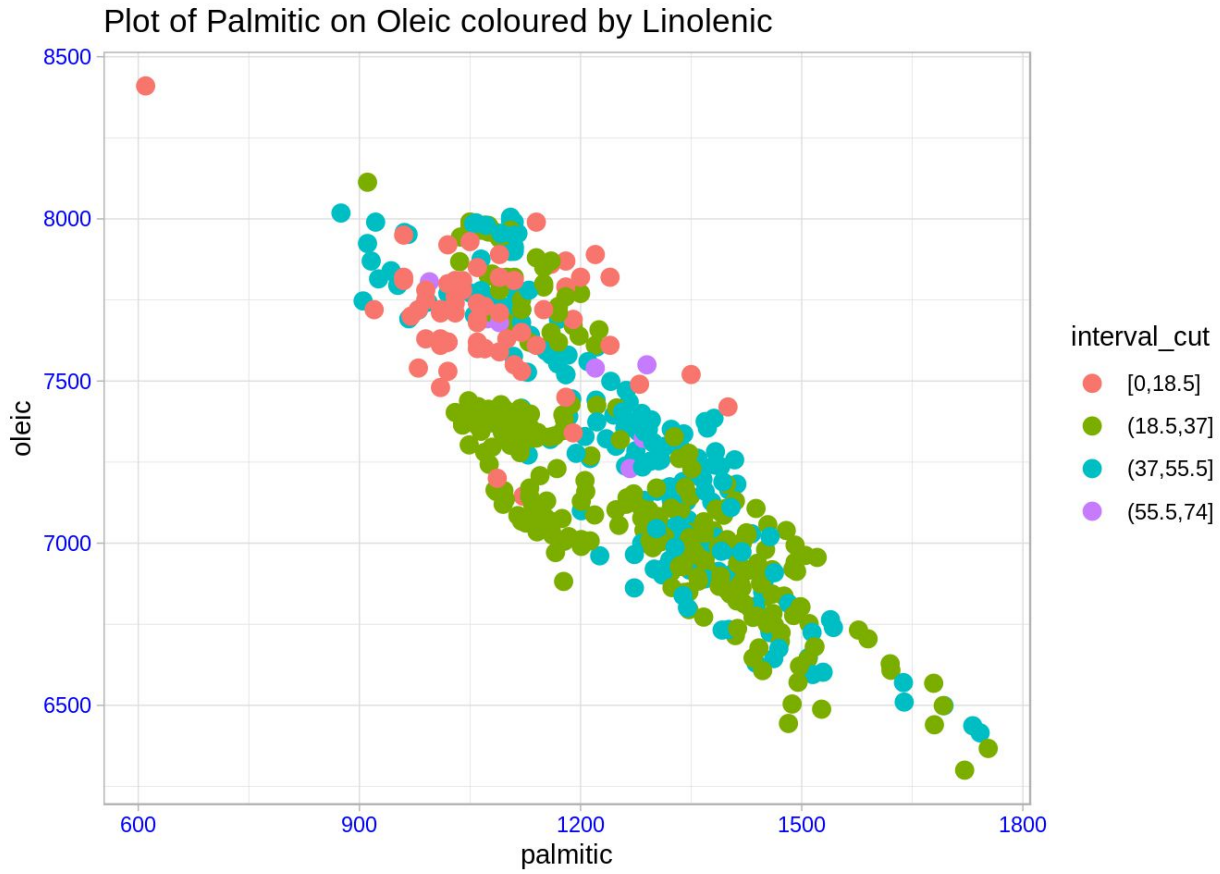
Colors help us to easiy understand and distinguish the patterns.The second plot is much better in visulisation.It is easier to analyze the plot that uses cut_interval. That is why the plot with intervalcut seems to better.

2

scatterplots of Palmitic vs Oleic in which you map the discretized Linolenic with four classes to:

a. Color

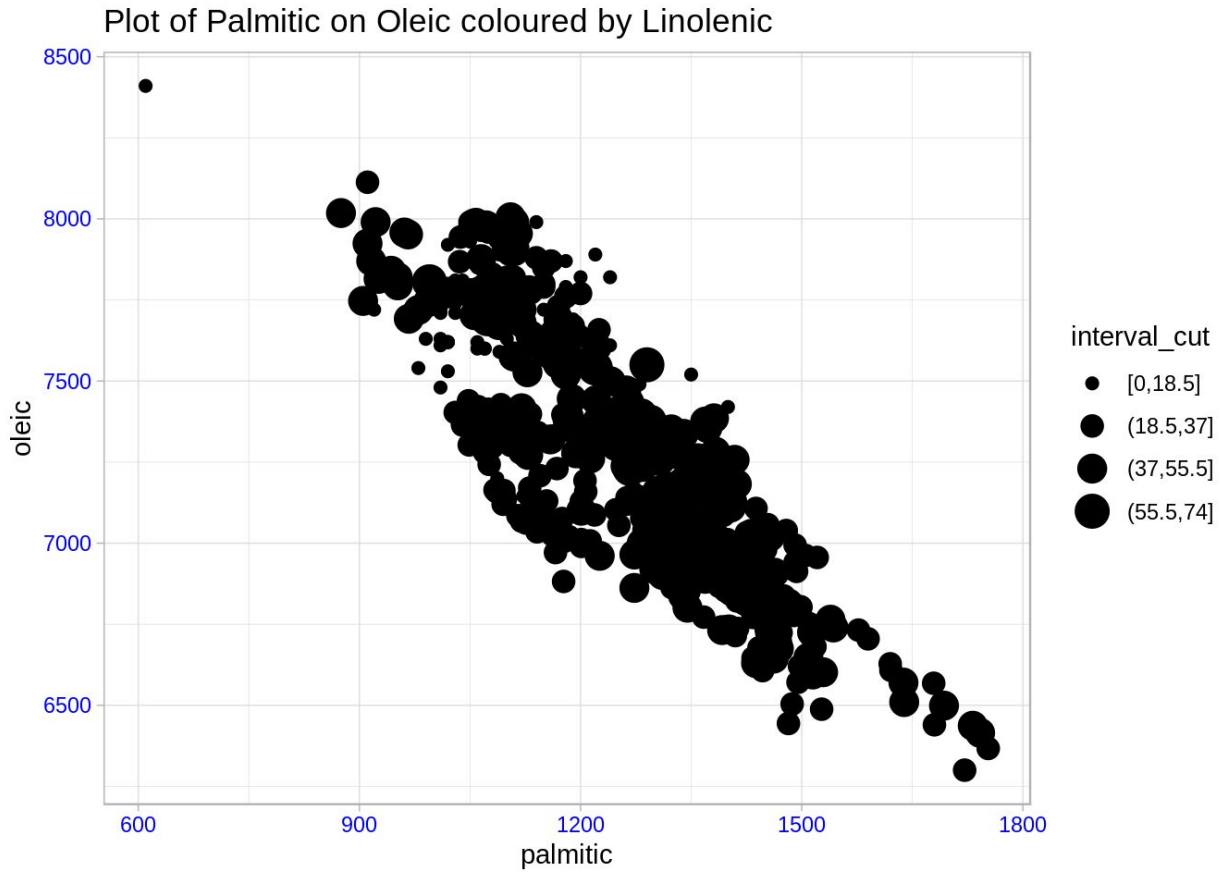
```
p3<-s2<-ggplot(olive,aes(palmitic,oleic,col=interval_cut))+geom_point()+theme_light()+lab
s(x="palmitic",y="oleic")+geom_point(size=3)+ggtitle("Plot of Palmitic on Oleic coloured
by Linolenic")+theme(axis.text = element_text(colour = "blue"))
p3
```



b. Size

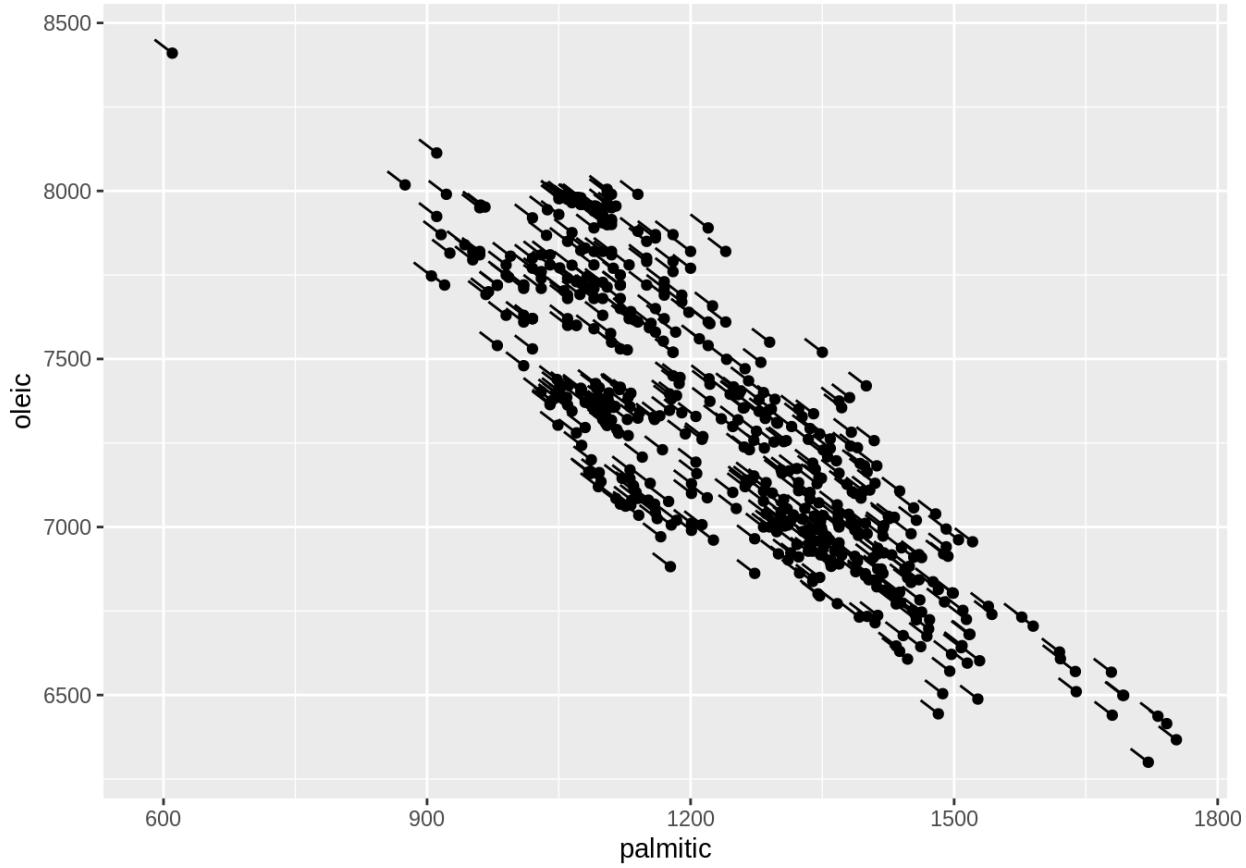
```
p4<-s2<-ggplot(olive,aes(palmitic,oleic))+geom_point(aes(size=interval_cut))+theme_light
()+labs(x="palmitic",y="oleic")+geom_point()+ggtitle("Plot of Palmitic on Oleic coloured
by Linolenic")+theme(axis.text = element_text(colour = "blue"))
p4
```

```
## Warning: Using size for a discrete variable is not advised.
```



c. Orientation angle

```
angleplot<-ggplot(olive,aes(x=palmitic,y=oleic)) +
  geom_point() +
  geom_spoke(aes(angle = 90, radius = 45))
angleplot
```

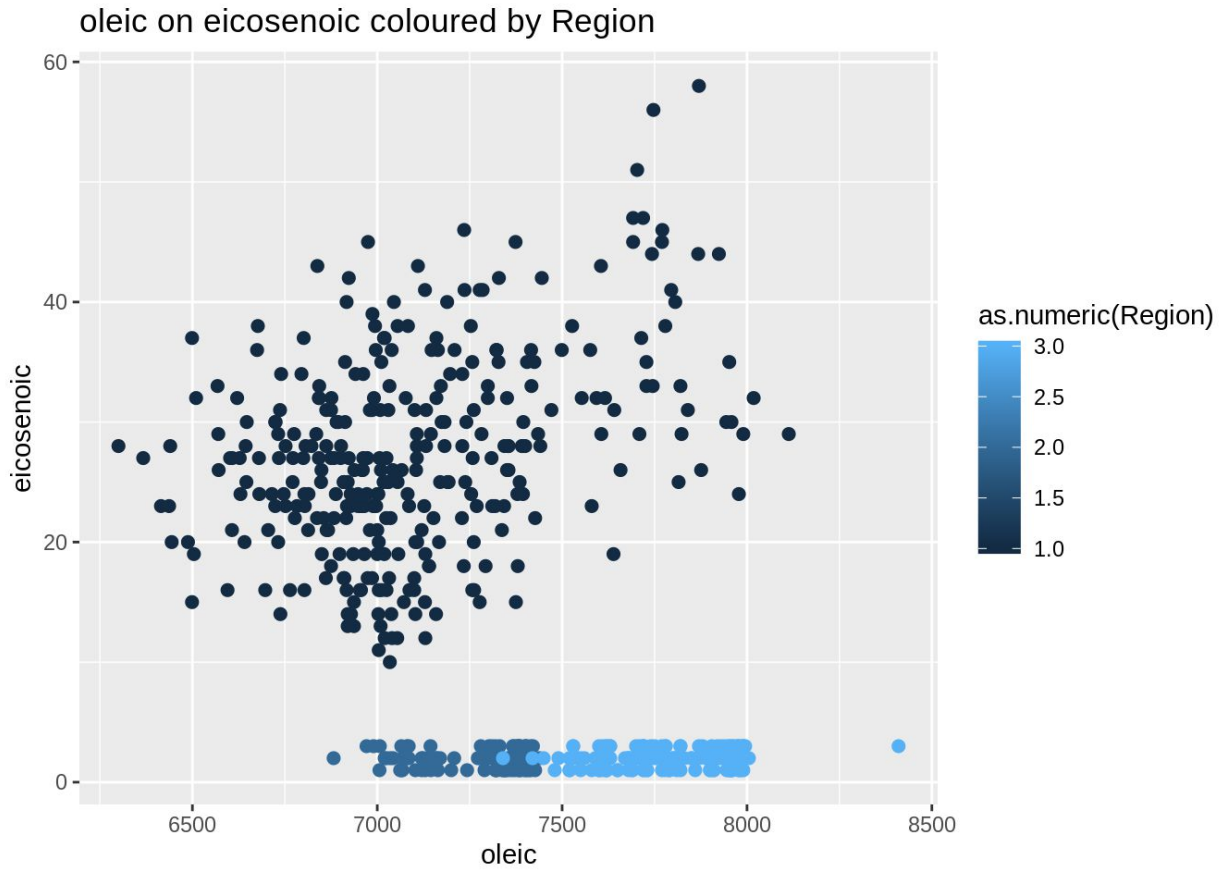


We feel it is very difficult to differenciate between categories in the second plot with size (b.size).The third plot is not good either.But a bit better than the second.

3

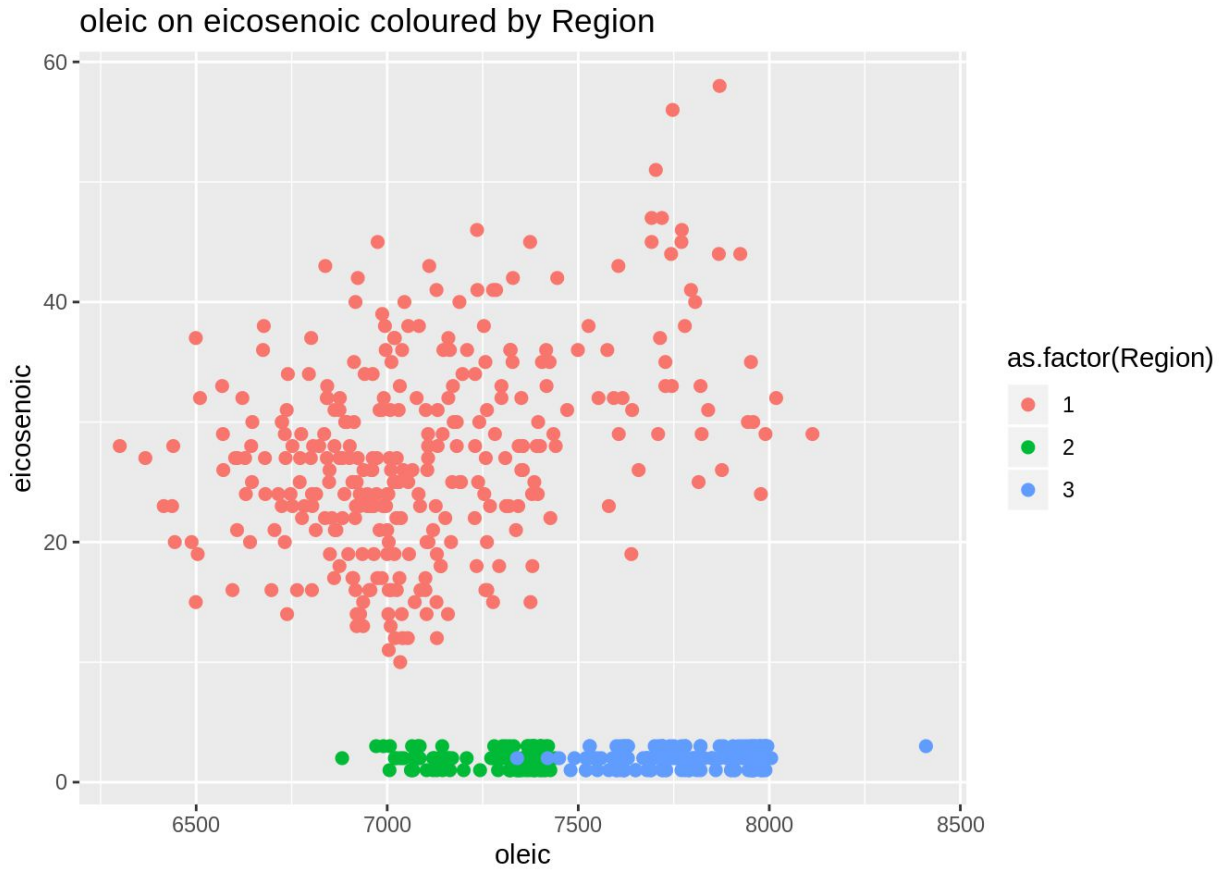
Create a scatterplot of Oleic vs Eicosenoic in which color is defined by numeric values of Region.

```
p6<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=as.numeric(Region)))+labs(x="oleic",y="ei
cosenoic")+geom_point(size=2)+ggtitle("oleic on eicosenoic coloured by Region")
p6
```



The decision boundaries are hard to distinguish as there seems to be a lot of classes with small variation in color.

```
p7<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=as.factor(Region)))+labs(x="oleic",y="eic
osenoic")+geom_point(size=2)+ggtitle("oleic on eicosenoic coloured by Region")
p7
```



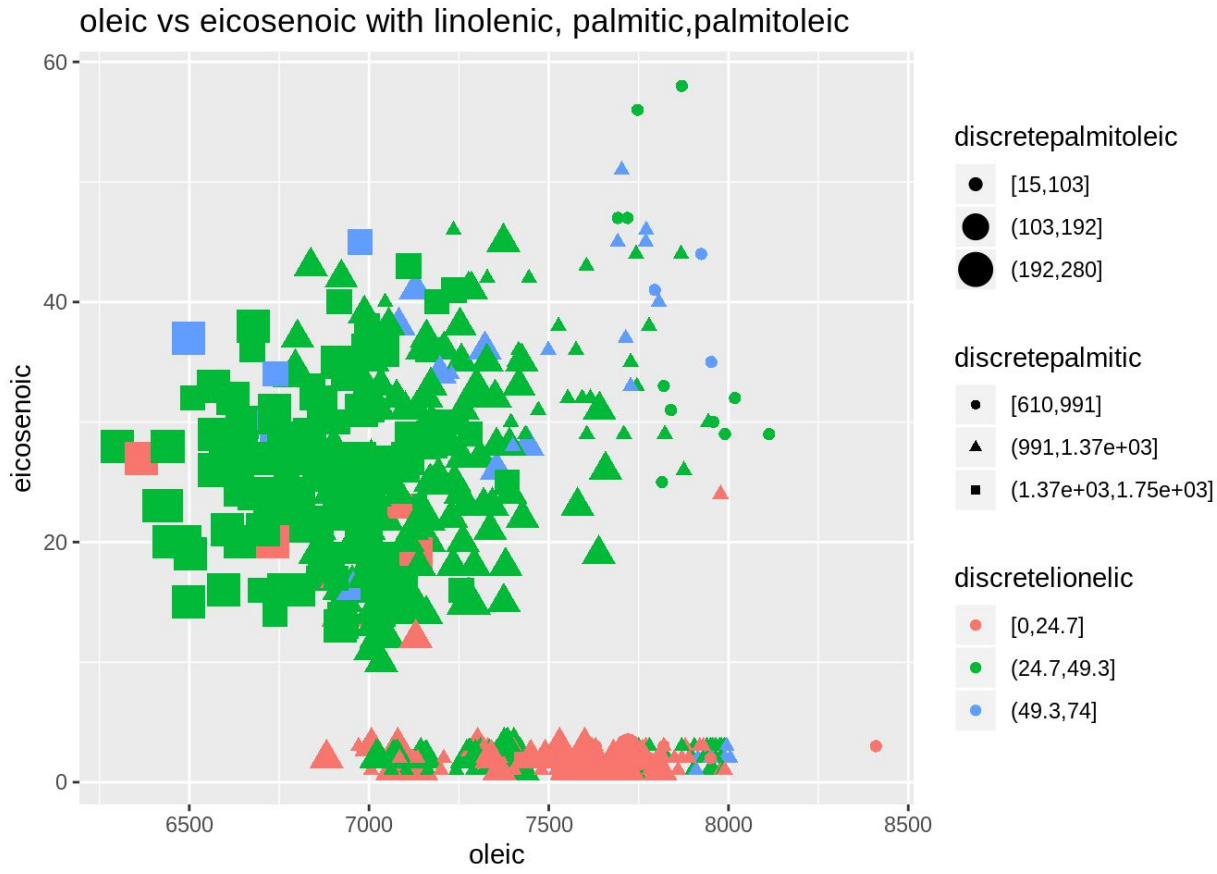
In the above plot it is very easy to find the decision boundaries .The three facors of region can be easily understood.Preattendive mechanism make it possible

4

scatterplot of Oleic vs Eicosenoic in which color is defined by a discretized Linoleic (3 classes), shape is defined by a discretized Palmitic (3 classes) and size is defined by a discretized Palmitoleic (3 classes)

```
discretelionelic<-cut_interval(olive$linolenic, 3)
discretepalmitic<-cut_interval(olive$palmitic, 3)
discretepalmitoleic<-cut_interval(olive$palmitoleic, 3)

p8<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=discretelionelic,shape=discretepalmitic,s
ize=discretepalmitoleic))+geom_point()+ggtitle("oleic vs eicosenoic with linolenic, palmi
tic,palmitoleic")
p8
```

```
## Warning: Using size for a discrete variable is not advised.
```



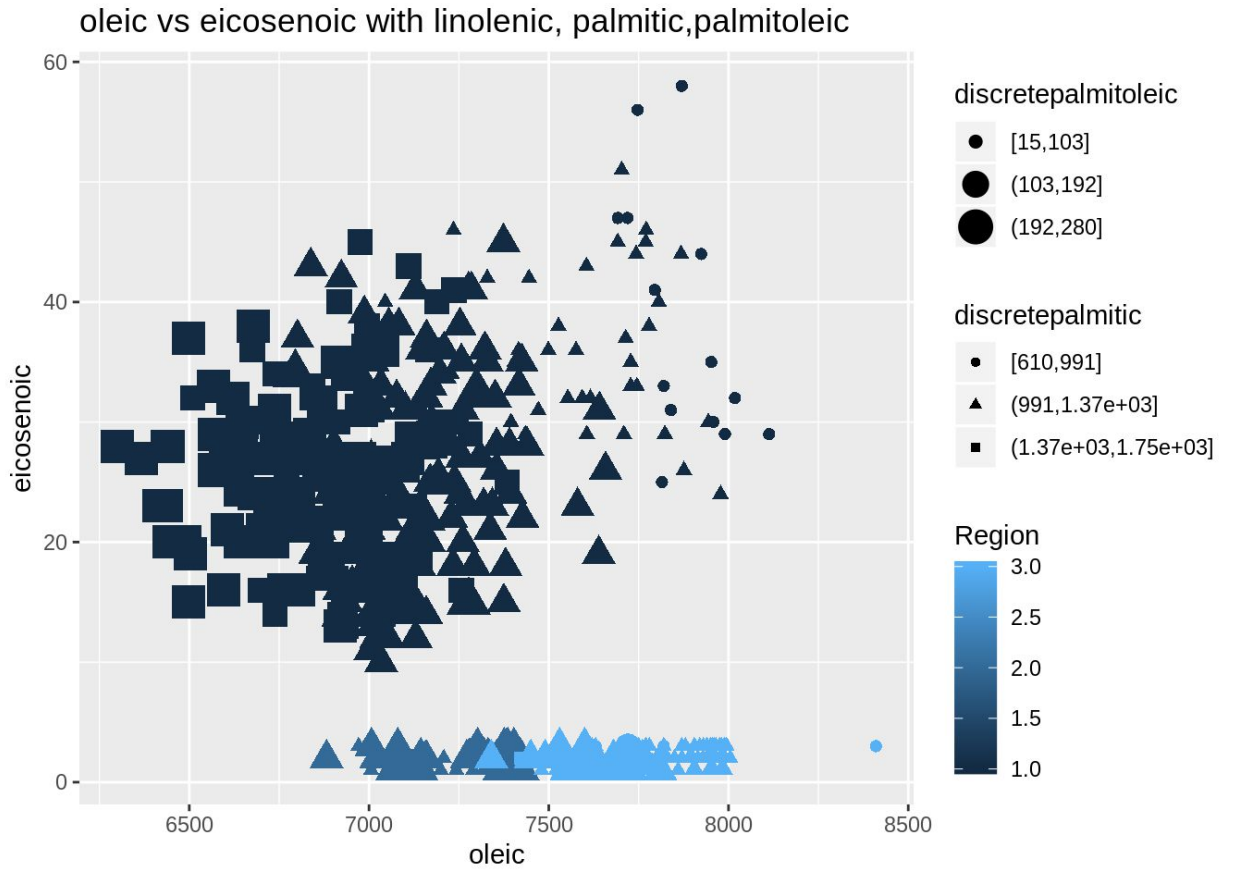
It is really difficult to differentiate between 27 different types of observations.Feels like too much information is fed into a single graph.It is above the channel capacity.SO the data looks clumpsy and difficult to interpret.

5

scatterplot of Oleic vs Eicosenoic in which color is defined by Region,shape is defined by a discretized Palmitic (3 classes) and size is defined by a discretized Palmitoleic (3 classes)

```
p9<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=Region,shape=discretepalmitic,size=discre
tepalmitoleic))+geom_point()+ggtitle("oleic vs eicosenoic with linolenic, palmitic,palmit
oleic")
p9
```

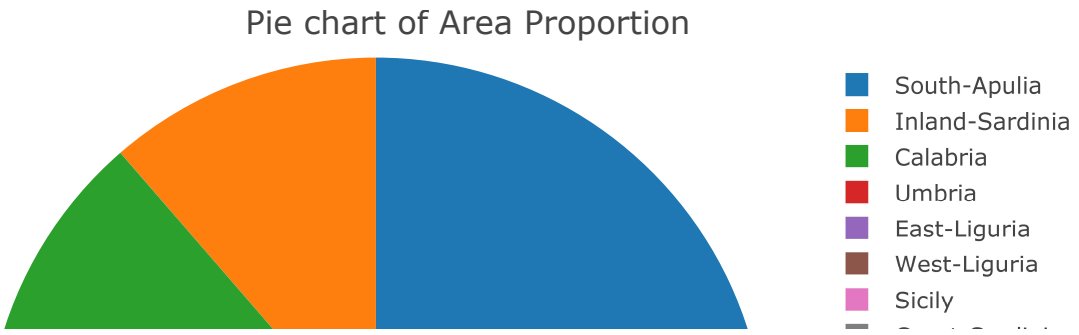
```
## Warning: Using size for a discrete variable is not advised.
```

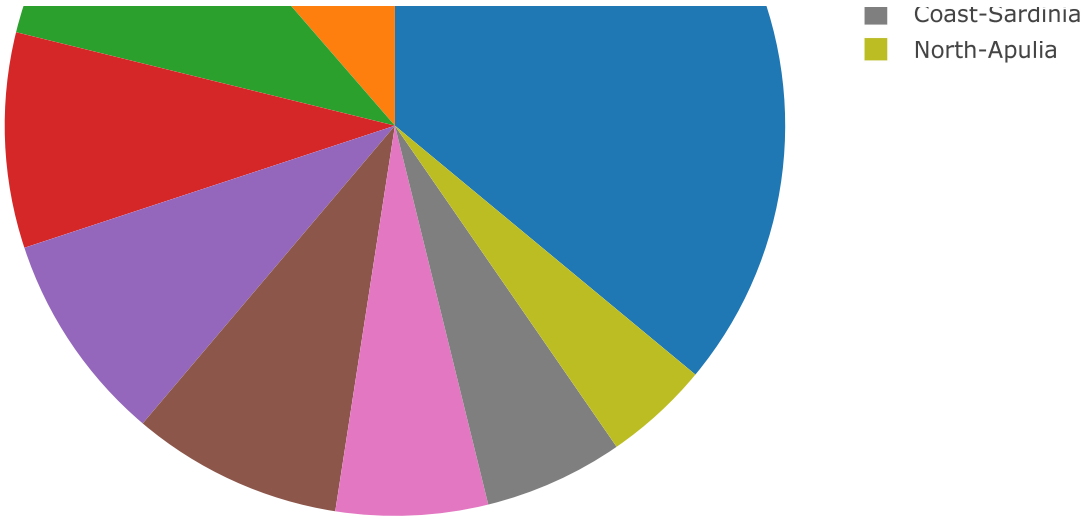


The figure is processed by checking individual feature maps.Each feature map will be going through a specific preattentive task and hence we are able to see clear decision boundaries. ##6

create a pie chart that shows the proportions of oils coming from different Areas using plotly

```
library(plotrix)
library(plotly)

mytable <- table(olive$Area)
dframe<-as.data.frame(mytable)

#pie3D(dframe$Freq,labels=dframe$Var1,explode=0.1,main="Pie Chart of Area Proportion ")

p10 <- plot_ly(dframe, labels = ~Var1, values = ~Freq, type = 'pie',textinfo = "none", ho
verinfo = "text",text=~Var1) %>%
  layout(title = 'Pie chart of Area Proportion',
         xaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE),
         yaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE))

p10
```

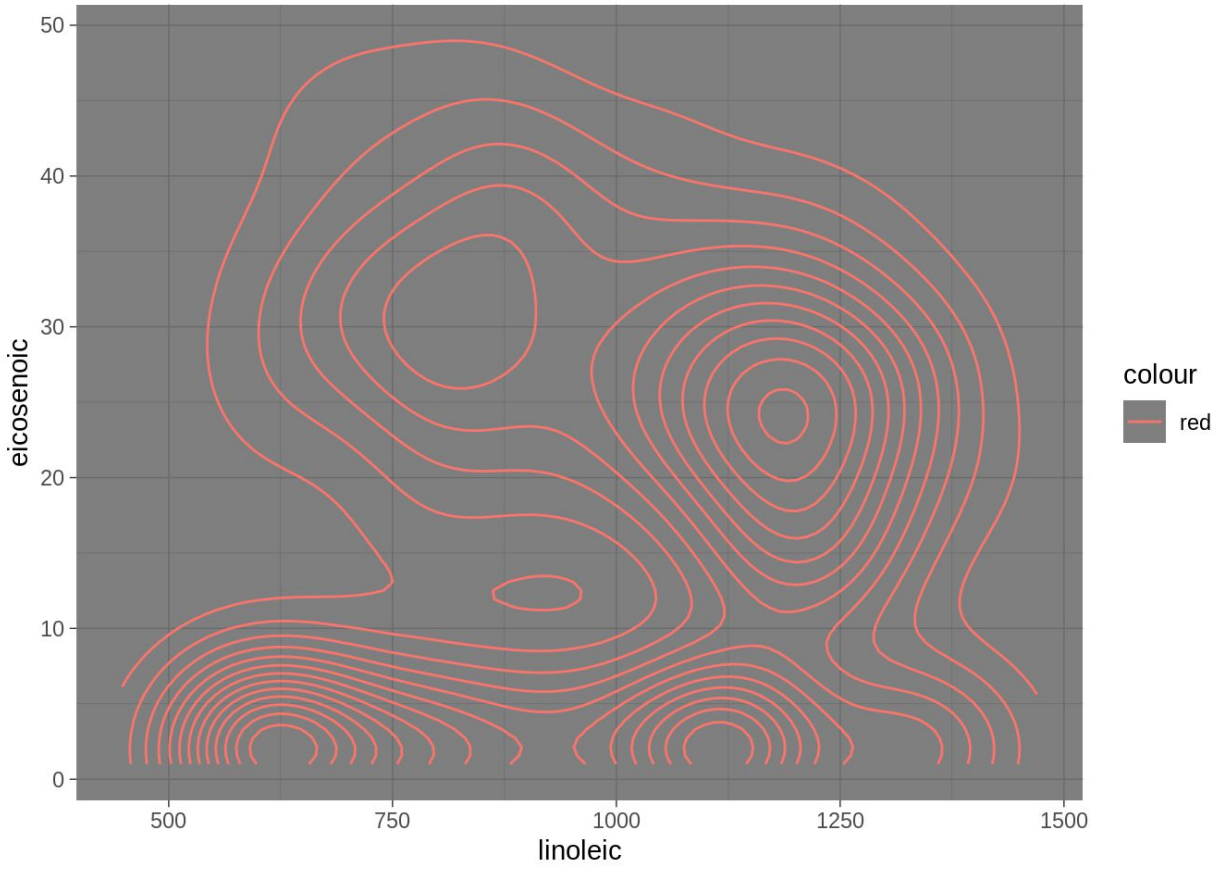




THe pie chart is less effective than the bar chart.When the values differ by small number it is diffcult to predict which area's frequency was more without hovering and finding for the exact value.In barchart just looking at the lenght we can find this. ##7

2d-density contour plot

```
p11<-ggplot(olive, aes(x=linoleic, y=eicosenoic,col="red") ) +geom_density_2d(position="identity")+theme_dark()
p11
```



Outliers are missed in contour plots.

Assignment 2

Load the baseball data in r

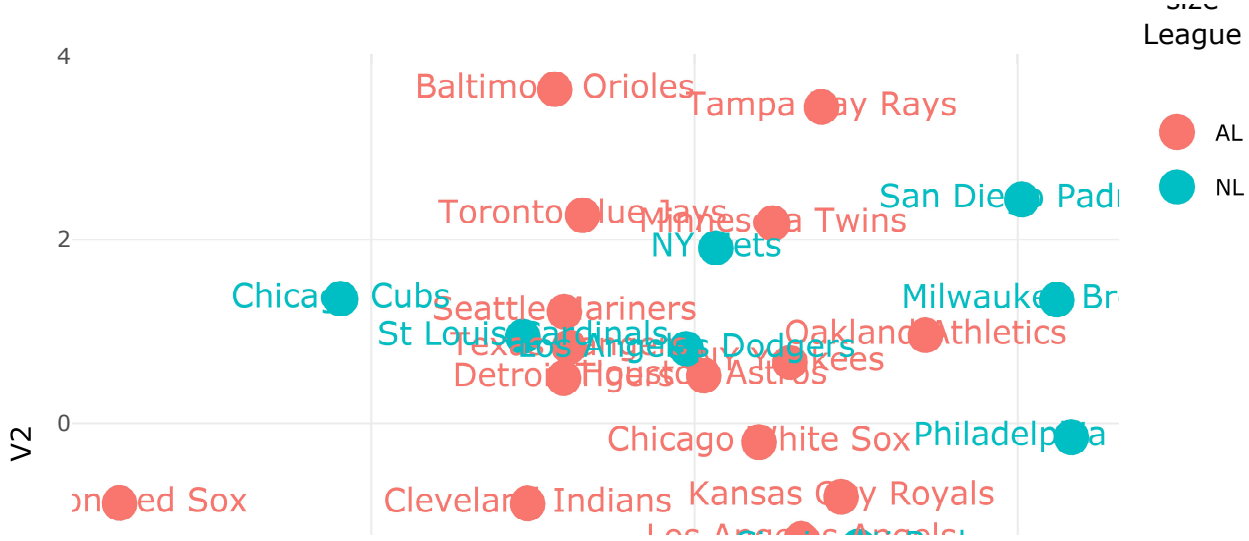
```
library(readxl)
library(ggplot2)
library(plotly)
library(MASS)


df<-read_xlsx("baseball-2016.xlsx",col_names=TRUE)
```

It is reasonable to scale the data because we have different range of features in the data.

```
df<-read_xlsx("baseball-2016.xlsx",col_names=TRUE)

scaled_df<-scale(df[,-c(1,2)])
d<-dist(scaled_df)
mds<-isoMDS(d,k=2)
```

```
## initial  value 19.856833
## iter   5 value 16.319153
## iter  10 value 16.046215
## final  value 15.935476
## converged
```

```
dt<-(mds$points)
dt_new<-as.data.frame(dt)
dt_new$League<-df$League
rownames(dt_new)<-df$Team



#plot_ly(dt, x = ~V1, y= ~V2,type ="scatter", color = ~League,labels=rownames(dt))

p <- ggplot(data=dt_new,aes(x=V1,y=V2,col=League,size=5))+geom_point()+theme_minimal()+ge
om_text(aes(label=rownames(dt_new),size=5))
ggplotly(p)
```



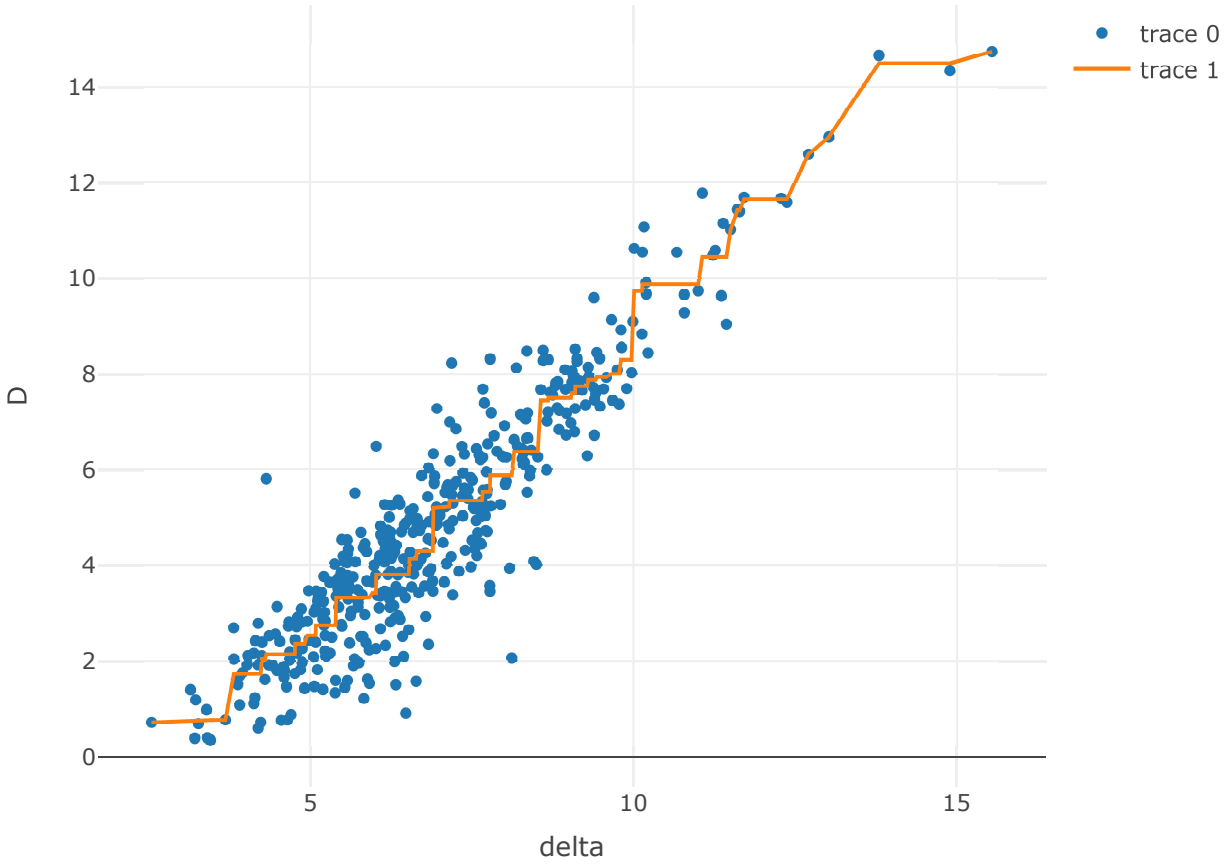
Form the graph we can first observe that On Red Sox seems to be an outlier.Also we can assume that the V1 coordinate seems to provide better differentiation between the 2 Leagues.Finally,we can say that theres seems to be some kind of diffrence between the Leagues.

3

```
library(readxl)
library(ggplot2)
library(plotly)
library(MASS)


df<-read_xlsx("baseball-2016.xlsx",col_names=TRUE)
scaled_df<-scale(df[,-c(1,2)])
d<-dist(scaled_df)
mds<-isoMDS(d,k=2)
```

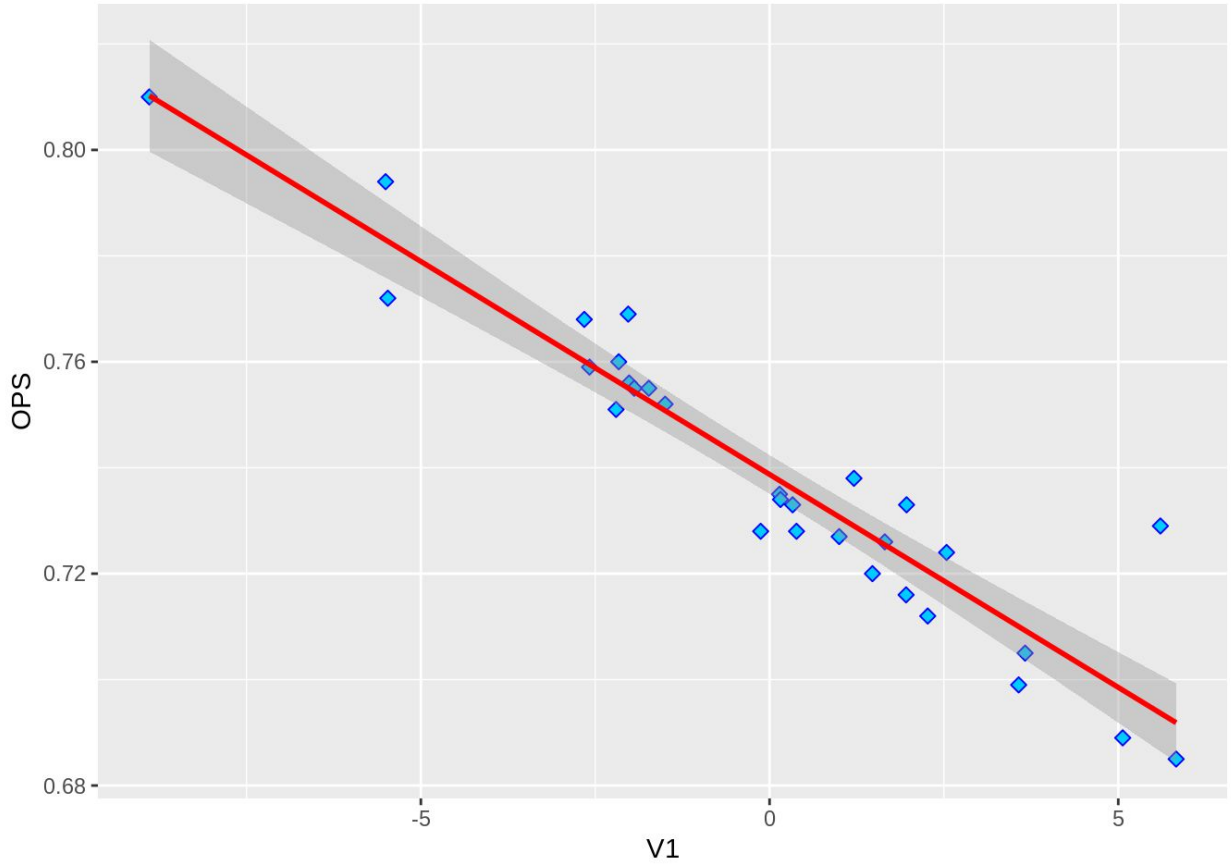
```
## initial  value 19.856833
## iter   5 value 16.319153
## iter  10 value 16.046215
## final  value 15.935476
## converged
```

```
dt<-(mds$points)
dt_new<-as.data.frame(dt)
dt_new$League<-df$League
rownames(dt_new)<-df$Team


sh <- Shepard(d,dt)
delta <-as.numeric(d)
D<- as.numeric(dist(dt))

n=nrow(dt)
index=matrix(1:n, nrow=n, ncol=n)
index1=as.numeric(index[lower.tri(index)])

n=nrow(dt)
index=matrix(1:n, nrow=n, ncol=n, byrow = T)
index2=as.numeric(index[lower.tri(index)])



plot_ly()%>%
  add_markers(x=~delta, y=~D, hoverinfo = 'text')%>%add_lines(x=~sh$x, y=~sh$yf)
```

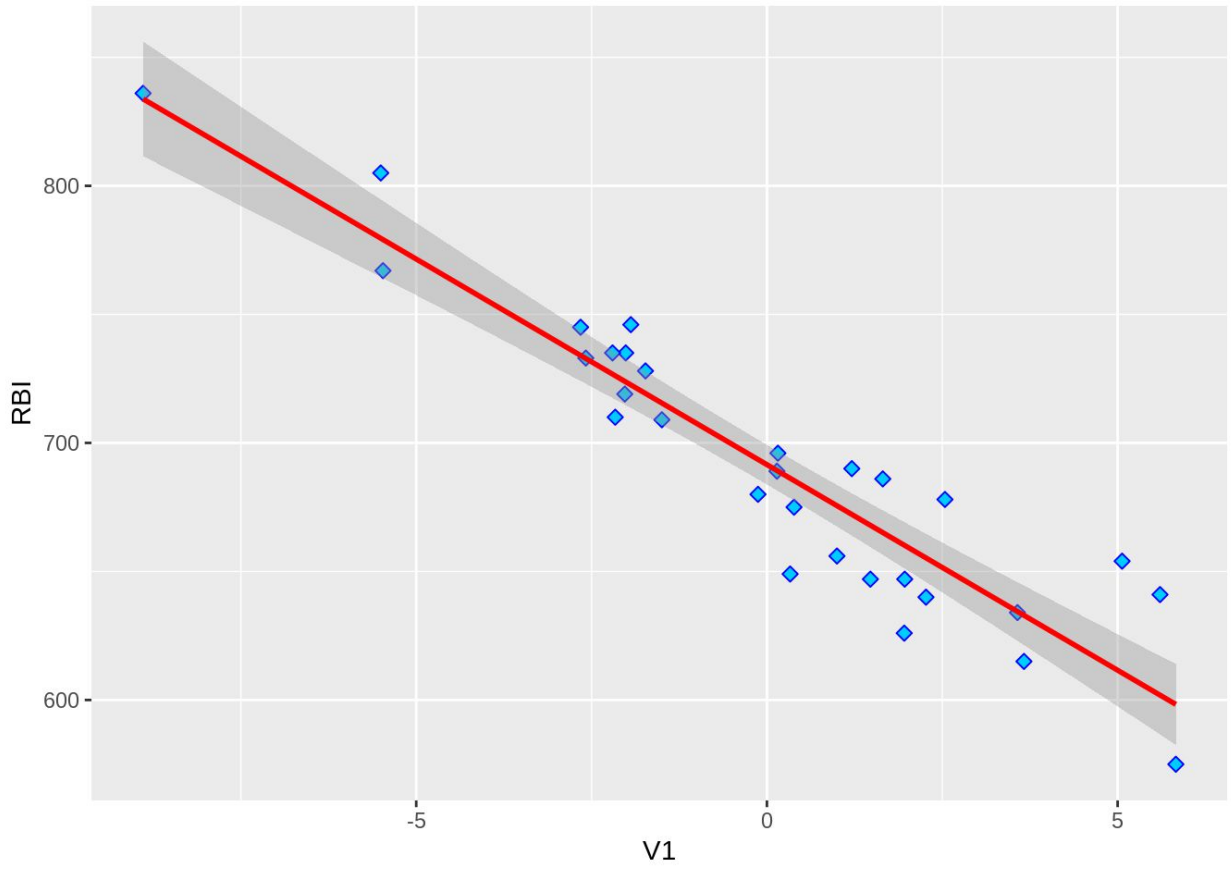


From the Shepard Plot we can derive the fact that 2 point on the upper right of the graph (coordinates (14.88924,14.50608) and(15.53976,14.7172)) where harder for the MDS to map.

```
library(gridExtra)
num_df<-df[,-c(1,2)]
num_df$V1<-dt_new$V1
num_df<-as.data.frame(num_df)
names(num_df)[8]<-"Doubles"
names(num_df)[9]<-"Triples"


ggplot(num_df,aes(x=dt_new$V1,y=OPS))+geom_point(size=2,color="blue",pch=23,fill="#00CCF
F")+stat_smooth(method = "lm", col = "red")+xlab("V1")
```




```
ggplot(num_df,aes(x=dt_new$V1,y=RBI))+geom_point(size=2,color="blue",pch=23,fill="#00CCF
F")+stat_smooth(method = "lm", col = "red")+xlab("V1")
```



We choose to plot the OPS and RBI against V1 coordinate we choose previously according to the first plot.We can easily see that in both plots theres a negative correlation. The variable RBI is a statistic in baseball that measure the total number of runs a hitter generates off of their at-bats with exception to runs scored due to errors by the fielding team. A batter is awarded an RBI when their at bat results in a safe hit, including home runs, sacrifice bunts and flys, infield out or fielder's choice, and leads to runners scoring. And the variable OPS is a statistic calculated as the sum of a player's on-base percentage and slugging average.The ability of a player both to get on base and to hit for power, two important offensive skills, are represented. Both of these variables are considered important in the baseball scoring because OPS represents offensive skills of the players and RBI measures a players ability compared to errors by the fielding team.

Appendix

```
library(tidyverse)
library(MASS)

olive<-read.csv("olive.csv",header=TRUE, sep=",")

###############Assignment-1_1
#Create a scatterplot in Ggplot2 that shows dependence of Palmitic on Oleic in which obse
rvations are colored by Linolenic

p1<-ggplot(olive,aes(palmitic,oleic,col=linolenic))+geom_point()+labs(x="palmitic",y="ole
ic")+geom_point(size=2)+ggtitle("Plot of Palmitic on Oleic coloured by Linolenic")
p1

interval_cut <- cut_interval(olive$linolenic, 4)
p2<-ggplot(olive,aes(palmitic,oleic,col=interval_cut,shape=interval_cut))+geom_point()+th
eme_light()+labs(x="palmitic",y="oleic")+geom_point(size=3)+ggtitle("Plot of Palmitic on
Oleic coloured by Linolenic")+theme(axis.text = element_text(colour = "blue"))
p2

###############Assignment-1_2
#Create scatterplots of Palmitic vs Oleic in which you map the discretized Linolenic wit
h four classes to:
#a. Color b. Size c. Orientation angle (use geom_spoke())

p3<-s2<-ggplot(olive,aes(palmitic,oleic,col=interval_cut))+geom_point()+theme_light()+lab
s(x="palmitic",y="oleic")+geom_point(size=3)+ggtitle("Plot of Palmitic on Oleic coloured
by Linolenic")+theme(axis.text = element_text(colour = "blue"))
p3

p4<-s2<-ggplot(olive,aes(palmitic,oleic))+geom_point(aes(size=interval_cut))+theme_light
()+labs(x="palmitic",y="oleic")+geom_point(size=3)+ggtitle("Plot of Palmitic on Oleic col
oured by Linolenic")+theme(axis.text = element_text(colour = "blue"))
p4

p5<-ggplot(olive,aes(x=palmitic,y=oleic)) +geom_point() +geom_spoke(aes(angle = 90, radiu
s = 45))
p5

###############Assignment-1_3
#Create a scatterplot of Oleic vs Eicosenoic in which color is defined by numeric values
of Region.

p6<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=as.numeric(Region)))+labs(x="oleic",y="ei
cosenoic")+geom_point(size=2)+ggtitle("oleic on eicosenoic coloured by Region")
p6

p7<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=as.factor(Region)))+labs(x="oleic",y="eic
osenoic")+geom_point(size=2)+ggtitle("oleic on eicosenoic coloured by Region")
p7

###############Assignment-1_4

#Create a scatterplot of Oleic vs Eicosenoic in which color is defined by a discretized L
inoleic (3 classes),
#shape is defined by a discretized Palmitic (3 classes) and size is defined by a discreti
```

```
zed Palmitoleic (3 classes)

discretelionelic<-cut_interval(olive$linolenic, 3)
discretepalmitic<-cut_interval(olive$palmitic, 3)
discretepalmitoleic<-cut_interval(olive$palmitoleic, 3)

p8<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=discretelionelic,shape=discretepalmitic,s
ize=discretepalmitoleic))+geom_point()+ggtitle("oleic vs eicosenoic with linolenic, palmi
tic,palmitoleic")
p8
###############Assignment-1_5
##Create a scatterplot of Oleic vs Eicosenoic in which color is defined by Region,
#shape is defined by a discretized Palmitic (3 classes) and size is defined by a discreti
zed Palmitoleic (3 classes)

p9<-ggplot(olive,aes(x=oleic,y=eicosenoic,color=Region,shape=discretepalmitic,size=discre
tepalmitoleic))+geom_point()+ggtitle("oleic vs eicosenoic with linolenic, palmitic,palmit
oleic")
p9

###############Assignment-1_6
#Use Plotly to create a pie chart that shows the proportions of oils coming from differen
t Areas
library(plotrix)

mytable <- table(olive$Area)
dframe<-as.data.frame(mytable)

#pie3D(dframe$Freq,labels=dframe$Var1,explode=0.1,main="Pie Chart of Area Proportion ")

p10 <- plot_ly(dframe, labels = ~Var1, values = ~Freq, type = 'pie',textinfo = "none", ho
verinfo = "text",text=~Var1) %>%
  layout(title = 'Pie chart of Area Proportion',
         xaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE),
         yaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE))

p10



mytable <- table(olive$Area)
dframe<-as.data.frame(mytable)

#pie3D(dframe$Freq,labels=dframe$Var1,explode=0.1,main="Pie Chart of Area Proportion ")

p10 <- plot_ly(dframe, labels = ~Var1, values = ~Freq, type = 'pie') %>%
  layout(title = 'Pie chart of Area Proportion',
         xaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE),
         yaxis = list(showgrid = FALSE, zeroline = FALSE, showticklabels = FALSE))

p10

###############Assignment-1_7
#Create a 2d-density contour plot with Ggplot2 in which you show dependence of Linoleic v
s Eicosenoic
```

```
p11<-ggplot(olive, aes(x=linoleic, y=eicosenoic,col="red") ) +geom_density_2d(position="i
dentity")+theme_dark()
p11


library(readxl)
library(ggplot2)
library(plotly)
library(MASS)


df<-read_xlsx("baseball-2016.xlsx",col_names=TRUE)

scaled_df<-scale(df[,-c(1,2)])
d<-dist(scaled_df)
mds<-isoMDS(d,k=2)

dt<-(mds$points)
dt_new<-as.data.frame(dt)
dt_new$League<-df$League
rownames(dt_new)<-df$Team



#plot_ly(dt, x = ~V1, y= ~V2,type ="scatter", color = ~League,labels=rownames(dt))

p <- ggplot(data=dt_new,aes(x=V1,y=V2,col=League,size=5))+geom_point()+theme_minimal()+ge
om_text(aes(label=rownames(dt_new),size=5))
ggplotly(p)

library(readxl)
library(ggplot2)
library(plotly)
library(MASS)


df<-read_xlsx("baseball-2016.xlsx",col_names=TRUE)
scaled_df<-scale(df[,-c(1,2)])
d<-dist(scaled_df)
mds<-isoMDS(d,k=2)

dt<-(mds$points)
dt_new<-as.data.frame(dt)
dt_new$League<-df$League
rownames(dt_new)<-df$Team


sh <- Shepard(d,dt)
delta <-as.numeric(d)
D<- as.numeric(dist(dt))

n=nrow(dt)
index=matrix(1:n, nrow=n, ncol=n)
index1=as.numeric(index[lower.tri(index)])
```

```
n=nrow(dt)
index=matrix(1:n, nrow=n, ncol=n, byrow = T)
index2=as.numeric(index[lower.tri(index)])



plot_ly()%>%
  add_markers(x=~delta, y=~D, hoverinfo = 'text')%>%add_lines(x=~sh$x, y=~sh$yf)

library(gridExtra)
num_df<-df[,-c(1,2)]
num_df$V1<-dt_new$V1
num_df<-as.data.frame(num_df)
names(num_df)[8]<-"Doubles"
names(num_df)[9]<-"Triples"


ggplot(num_df,aes(x=dt_new$V1,y=OPS))+geom_point(size=2,color="blue",pch=23,fill="#00CCF
F")+stat_smooth(method = "lm", col = "red")+xlab("V1")

library(gridExtra)
num_df<-df[,-c(1,2)]
num_df$V1<-dt_new$V1
num_df<-as.data.frame(num_df)
names(num_df)[8]<-"Doubles"
names(num_df)[9]<-"Triples"


ggplot(num_df,aes(x=dt_new$V1,y=OPS))+geom_point(size=2,color="blue",pch=23,fill="#00CCF
F")+stat_smooth(method = "lm", col = "red")+xlab("V1")
```