Data Analysis

Vatsala Tewari

08/01/2021

#Visualise data with R

For this first exercises we will use the same downloaded for the Course assignment 1 called `dataset_1.csv` in the visualization suborder

##Libraries in use.

In the console we can type the following codes to load the important libraries to complete this assignment. library(readr) library(tidyverse) install.packages("hexbin") library(hexbin)

##V1

Is there any correlation between the number of cells and the presence of clumps (grouped together)? Is this behavior different in different concentrations? Create a where you show the relation between cell_features$num_cellsandcell_features$clumps in each concentration. Can you draw a correlation line? Hint. see geom_smooth().

##AnswerV1

To answer this question let us first do some pre-work.

Lets us load the data set. [we can use the import_data set function, load via link]

Then we can view the data set via the function view().

We will see duplicates of a read for the same cell_id and same batch there are multiple readings.

First we need to create a table with only the variables we require to answer the question. The need for this table is it cleans the work space and helps the bio-informatician to have a systematic line of thought.

I have called this table **Clump_correlation**.

Here we will extract the num_cells, clumps from the dataset_1.csv file, but we also need to make sure the multiple readings for the same short_name , batch and concentration are merged, hence we use the group_by and summarise functions. This gives us a table with 5 variables, where multiplicates with the same short_name , batch and concentration have been merged and thier total cells and total clumps calculated.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(readr)
setwd("~/Desktop/DataAnalysis")
dataset_1 <- read_csv("dataset_1.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   cell_line_id = col_double(),
##   short_name = col_character(),
##   concentration = col_double(),
##   batch = col_character(),
##   num_cells = col_double(),
##   cell_area_mean = col_double(),
##   roundness_mean = col_double(),
##   nucleus_area_mean = col_double(),
##   nucleus_roundness_mean = col_double(),
##   proliferation_mean = col_double(),
##   clumps = col_double()
## )
```

```
View(dataset_1)
clump_correlation <- dataset_1 %>%
select(short_name, batch, concentration, num_cells, clumps)%>%
group_by(short_name, batch , concentration)%>%
summarise(total_cells = mean(num_cells), total_clumps = mean(clumps))
```

```
## `summarise()` regrouping output by 'short_name', 'batch' (override with `.groups` argument)
```

Now, we will plot the data into a scatter plot. Using ggplot, the data from which we want to plot the scatter plot is clump_correlation then the X axis and Y axis is defined, and we tell the program to plot a scatter plot by the function geom_point().But wait! We have a lot of observations! They are also quite small, we can tell just by looking that they are centered around 0. It would be better if we can see the density gradient.

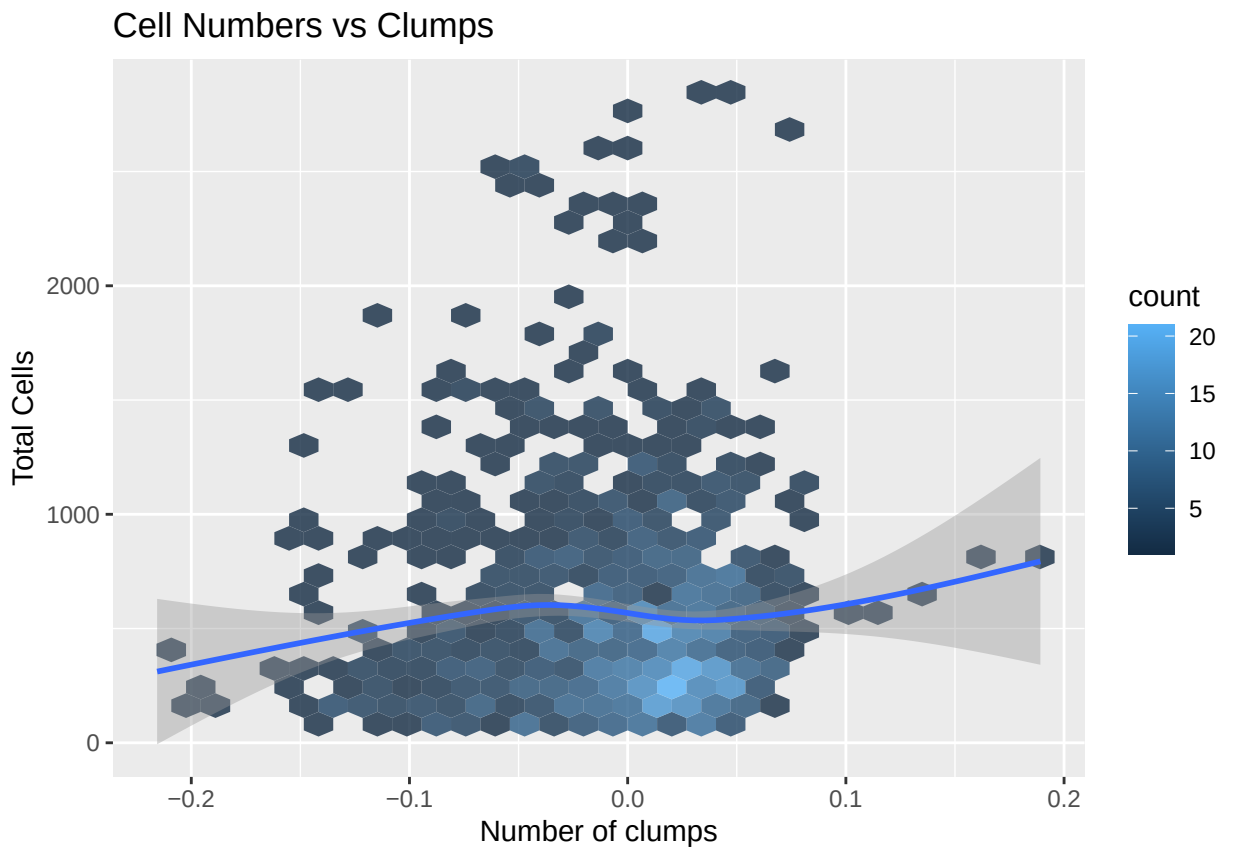
Hence we use geom_hex() function.

To plot the correlation line we use geom_smooth().

I use the labs function to make the data more informative to the viewer.

```
clump_plot <- ggplot(data = clump_correlation, mapping = aes(x = total_clumps , y = total_cells))
 clump_plot +
   geom_hex(alpha = 0.8)+
   geom_smooth()+
    labs(title = "Cell Numbers vs Clumps",
x = "Number of clumps",
y = "Total Cells")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Correlation

There is no correlation between cell number and clumps. This can be inferred from the some what horizontal line in the plot, plotted by the function geom_smooth(). The reason why we believe its horizontal is because we need to look carefully at the data, the variance is huge and the no. of observations very low (<5) Which means the trend is that with increasing cells there is no effect on the no. of clumps.

Another observation is most of the cells have about -0.1-0.1 clumps. Since there can't be negative clumps, more information about the data acquisition could help us understand why the data behaves like this.

A very interesting plot can be made if just add [color=batch] when we are forming the plot,this helps to get an idea if a batch was fault, if there are too many clumps.

We can play around with the data because its very well structured.

Another way to understand correlation is by using the line of best fit by abline() function.

Here I also show how we can also use the $ function to plot the X and Y axis.

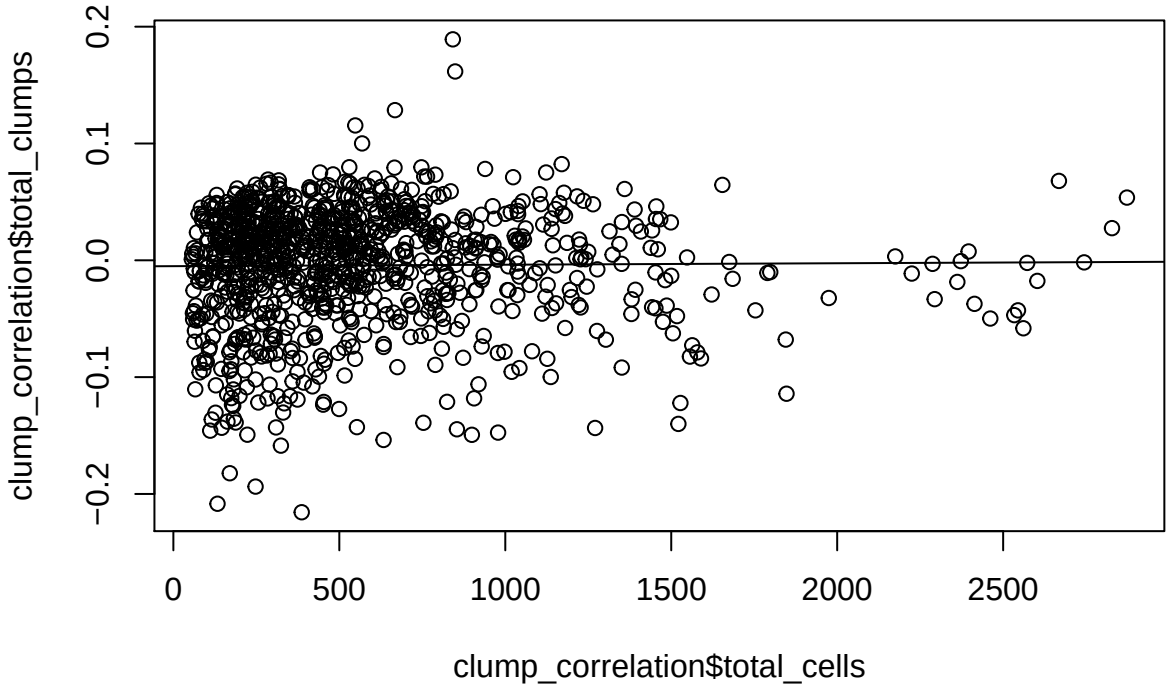
I also show how the line of best fit does not depend on what X and what Y axis you choose.

The horizontal line of best fit mean there is no correlation. It can be explained better, If the number of cells in the lower hundreds , no. of cells is the mean or if the number of cells is more than six thousand there can still be close to zero clumps.

It is not like if the number of cells increase the number of clumps would also increase.

there is no correlation

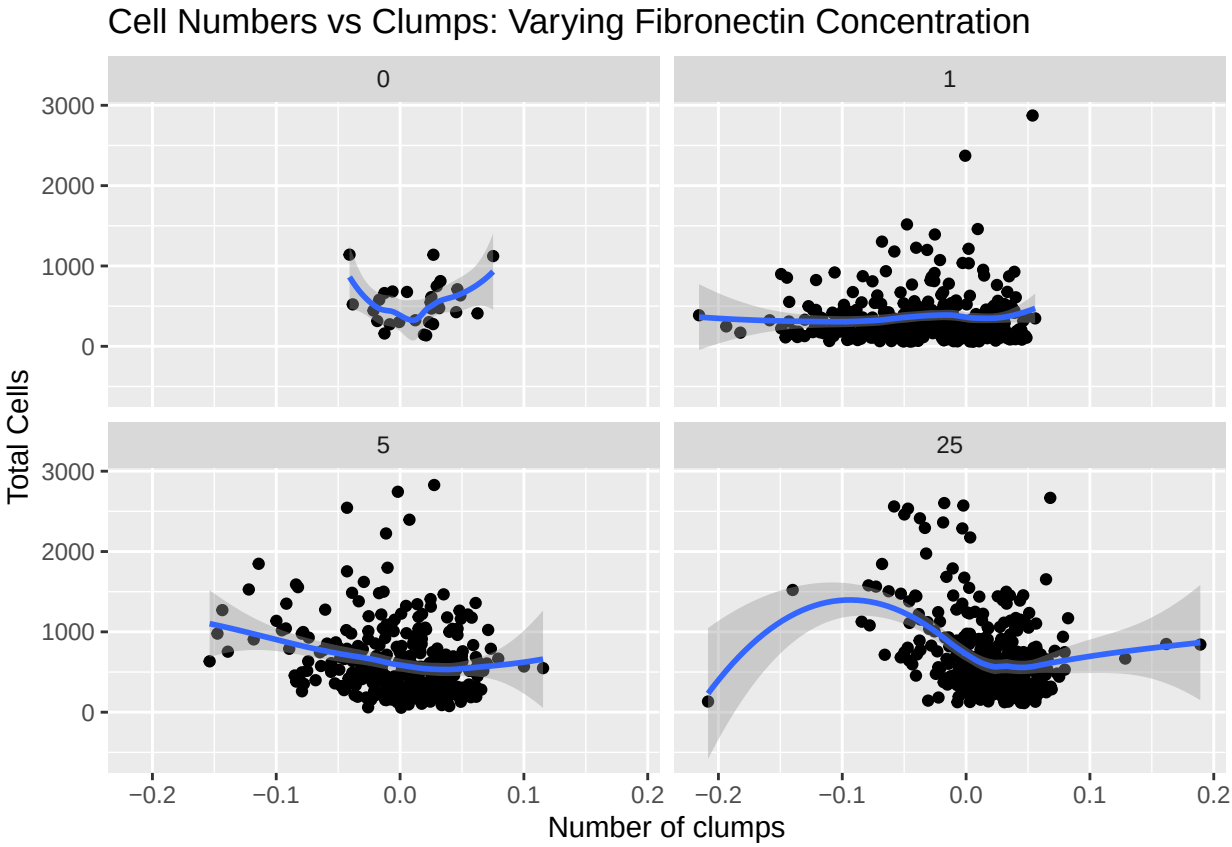
```
plot( clump_correlation$total_clumps ~ clump_correlation$total_cells)
abline(lm(clump_correlation$total_clumps~clump_correlation$total_cells))
```



Do Different Concentration mean Different Behavior? We can plot the different concentrations from the clump_plot [plot we plotted above] so that each concentration has a different plot. We can also use the same function to build the correlation line.

```
clump_plot+
  geom_point()+
  geom_smooth()+
  facet_wrap(facets = vars(concentration))+
labs(title = "Cell Numbers vs Clumps: Varying Fibronectin Concentration",
x = "Number of clumps",
y = "Total Cells")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



We can again observe that there is not much of a correlation in the plots, and the places where the plots show a negative and positive slope is accompanied by increased variance and decreased sample size hence this shows that possibly it happened by chance.

Hence, there is no correlation what so ever.

##V2 Show the distribution of cell area and nucleus area in the different fibronectin concentrations. Each will be displayed in a separate plot. How can you create a figure with the two plots side by side?

##Answer2 We use the same technique to plot the area correlation plot.

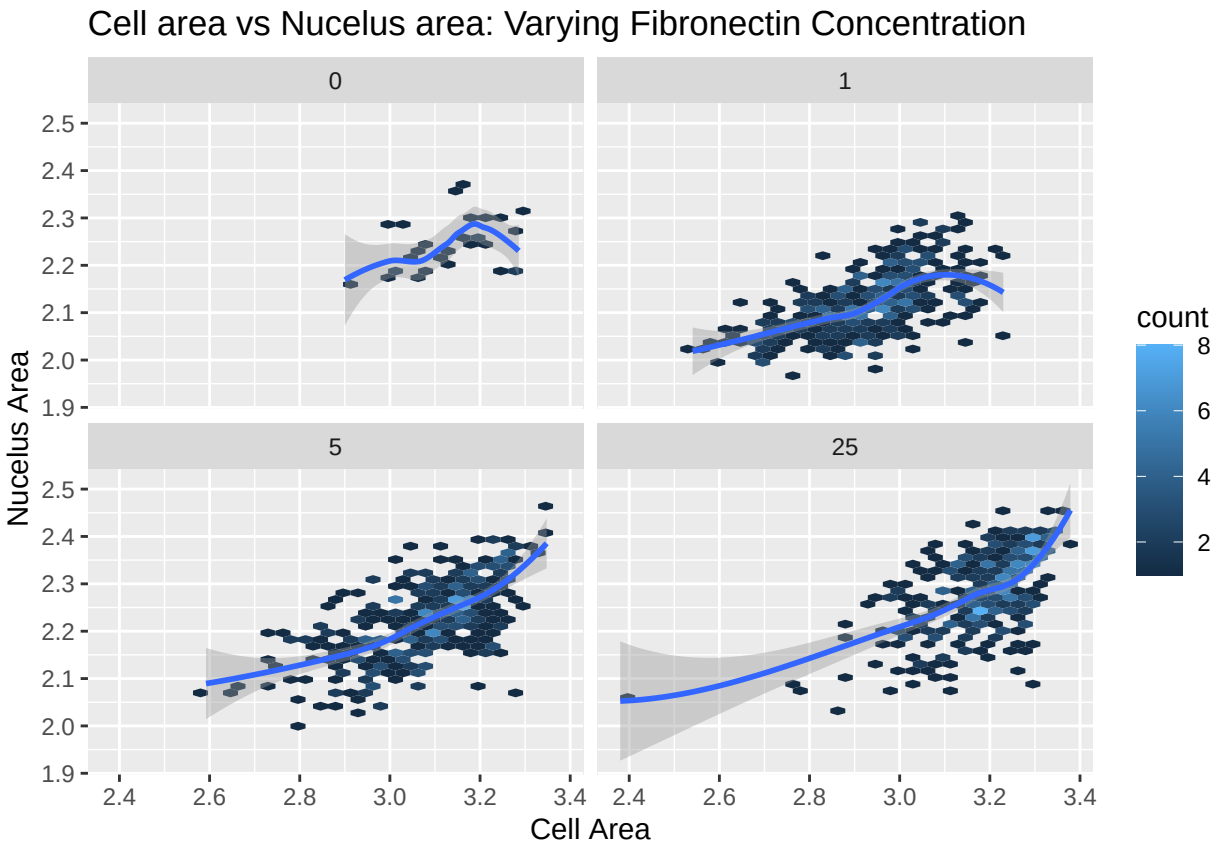
The plot we made above[clump_plot], and the plot we made now[area_plot] can be shown together by

```
area_correlation <- dataset_1%>%
select(short_name, batch, concentration, cell_area_mean, nucleus_area_mean)%>%
group_by(short_name, batch , concentration)%>%
summarise(cell_area = mean(cell_area_mean), nucleus_area = mean(nucleus_area_mean))
```

```
## `summarise()` regrouping output by 'short_name', 'batch' (override with `.groups` argument)
```

```
area_plot <- ggplot(data = area_correlation, mapping = aes(x = cell_area , y = nucleus_area))
 area_plot +
   geom_hex()+
   geom_smooth()+
    facet_wrap(facets = vars(concentration))+
    labs(title = "Cell area vs Nucelus area: Varying Fibronectin Concentration",
x = "Cell Area",
y = "Nucelus Area")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



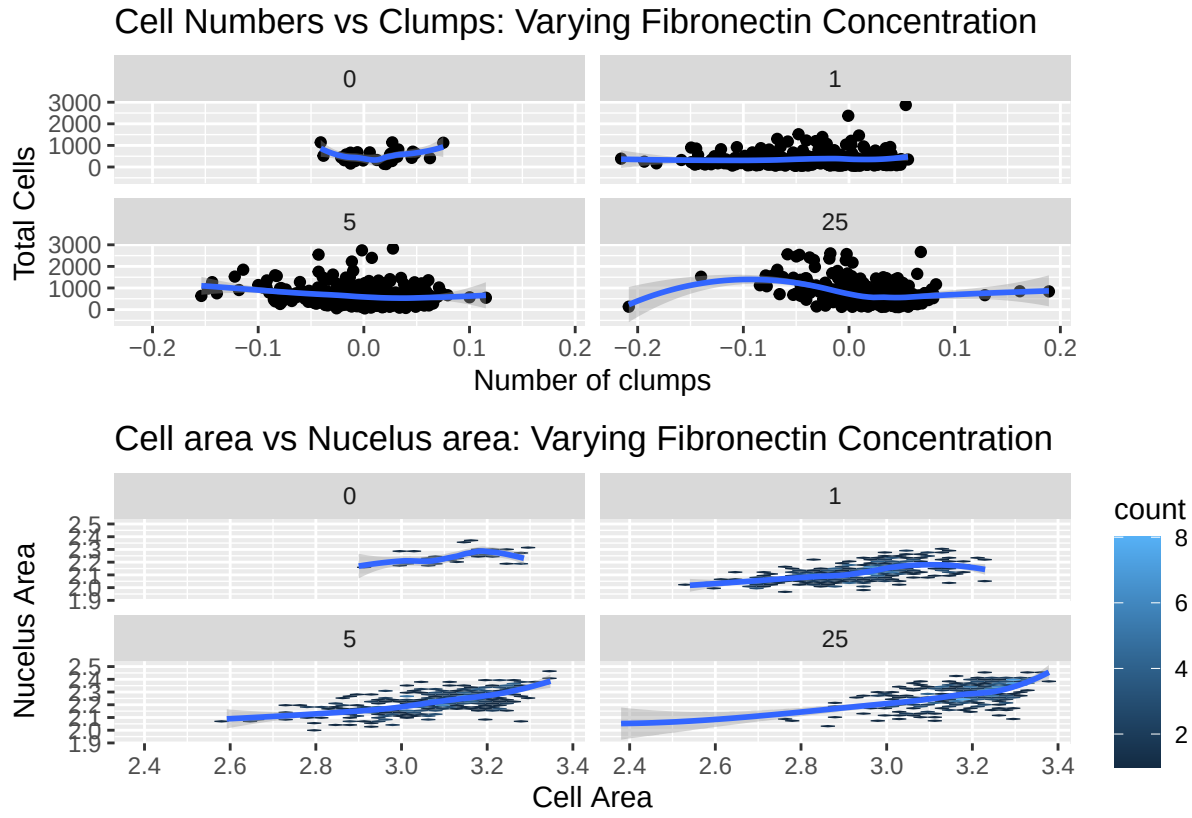
Plotting the two plots together

```
library(patchwork)
plot1<-clump_plot+
  geom_point()+
  geom_smooth()+
  facet_wrap(facets = vars(concentration))+
labs(title = "Cell Numbers vs Clumps: Varying Fibronectin Concentration",
x = "Number of clumps",
y = "Total Cells")

plot2<- area_plot +
   geom_hex()+
   geom_smooth()+
    facet_wrap(facets = vars(concentration))+
    labs(title = "Cell area vs Nucelus area: Varying Fibronectin Concentration",
x = "Cell Area",
y = "Nucelus Area")

plot1 / plot2
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



##V3

Create a heatmap showing the cell roundness in all the cell lines in the fibronectin concentrations 1, 5 and 25.

Create the same heatmap showing the different behavior in Batch A, Batch B and Batch C.

Change the default filling color (scale of blue) with a two or three color scale. Hint. use the RColorBrewer package and create a cols vector containing the color palette you like such as `cols <- rev(brewer.pal(11, 'RdYlBu'))`

##AnswerV3

First we will create a object in R with the values we need.

First we remove all zero concentration data points.

We then select only for cell_line_id , concentration and roundness mean.

group the multiplicities and average there data points.

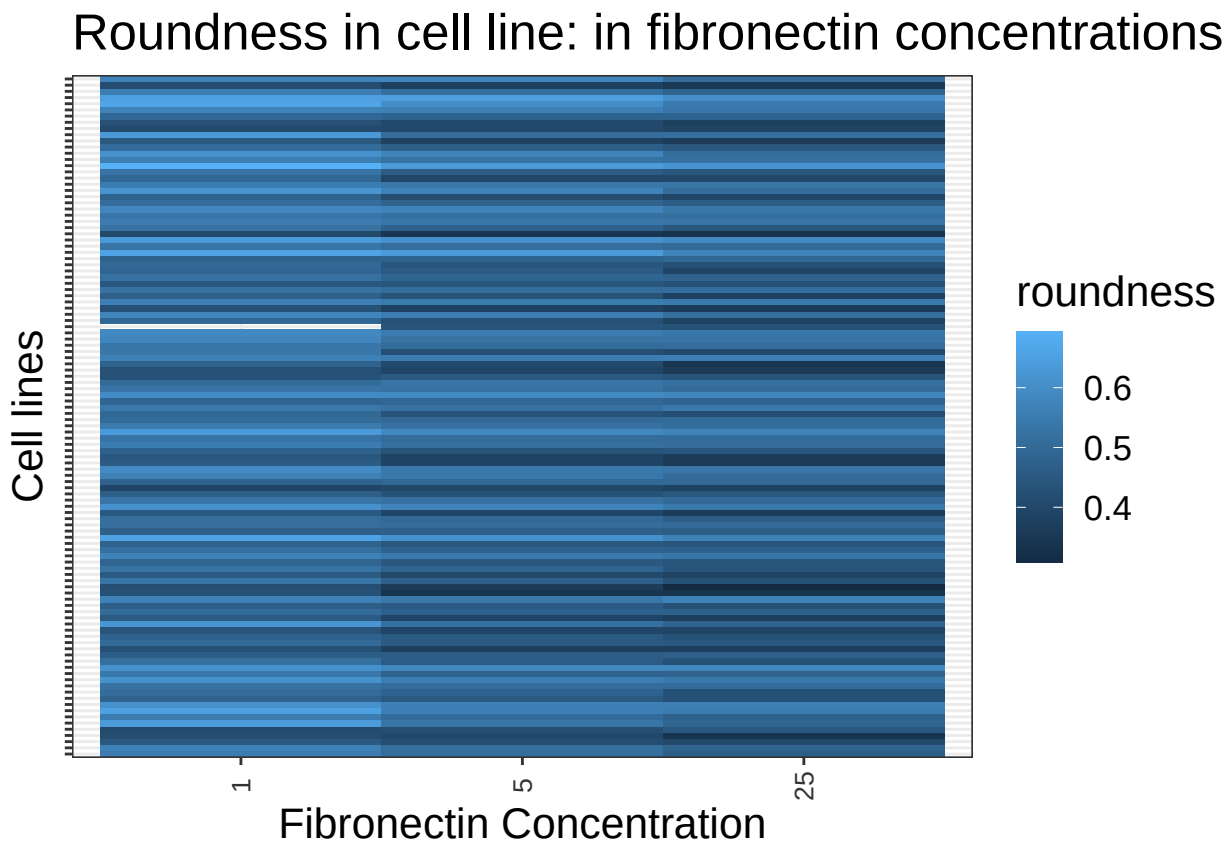
```
roundness_correlation <- dataset_1%>%
filter(concentration > 0 ) %>%
select(cell_line_id, concentration,roundness_mean)%>%
group_by(cell_line_id, concentration)%>%
summarise(roundness = mean(roundness_mean))
```

```
## `summarise()` regrouping output by 'cell_line_id' (override with `.groups` argument)
```

We then plot the heatmap using the geom_title() function where we describe the X axis Y axis and the values that need to fill the heatmap. Since the concentration and cell_line_id are numbers we need R to read them as factors.

We can further label the plot using lab function and theme functions to make the heat map understandable, we can further add themes and change labels formats of the X,Y axis.

```
Concentration_map<-ggplot(roundness_correlation, aes(factor(concentration) ,factor(cell_line_id), fill =
Concentration_map+
geom_tile()+
   labs(title = "Roundness in cell line: in fibronectin concentrations",
x = "Fibronectin Concentration",
y = "Cell lines")+
   theme_bw() +
theme(axis.text.x = element_text(colour = "grey20", size = 10, angle = 90, hjust = 0.5, vjust = 0.5),
axis.text.y = element_text(colour = "grey20", size = 0.1),
strip.text = element_text(face = "italic"),
text = element_text(size = 16))
```



To plot the same plot [variation of roundness in cell lines in different concentrations] now will be plotted for those where they only belong to Batch A/Batch B/Batch C.

We first save the data we require to a new table.We then create separate objects for each concentration.

If there are any NA we need to change them to 0, to make the map more continuous.

```
#Creating Table
batch_correlation <- dataset_1%>%
filter((concentration > 0),batch == c('Batch A','Batch B','Batch C')) %>%
select(cell_line_id,batch,concentration,roundness_mean)%>%
group_by(cell_line_id,concentration, batch)%>%
summarise(roundness = mean(roundness_mean))
```

```
## Warning in batch == c("Batch A", "Batch B", "Batch C"): longer object length is
## not a multiple of shorter object length
```

```
## `summarise()` regrouping output by 'cell_line_id', 'concentration' (override with `.groups` argument)
```

```
#Creating the objects and converting NA to 0

Concentration_1_mapdata <- batch_correlation%>%
   filter(concentration ==1 )%>%
  spread(key = batch, value = roundness)

Concentration_1_mapdata[is.na(Concentration_1_mapdata)]=0

Concentration_5_mapdata <-batch_correlation%>%
   filter(concentration ==5 )%>%
  spread(key = batch, value = roundness)

Concentration_5_mapdata[is.na(Concentration_5_mapdata)]=0

Concentration_25_mapdata <-batch_correlation%>%
   filter(concentration ==25 )%>%
  spread(key = batch, value = roundness)

Concentration_25_mapdata[is.na(Concentration_25_mapdata)]=0

# We can remove the concentration column since its not needed now
Concentration_25_mapdata$concentration<- NULL
Concentration_5_mapdata$concentration<- NULL
Concentration_1_mapdata$concentration<- NULL
```

We can add color to our heat map by loading the right library and then saving custom made changes to a variable by pallets in the RColorBrewer library.

We then form the Heatmap. We first save the needed columns in a object as a matrix. we then use the heatmap() function to plot the heatmap. We can remove the clustering by setting Rowv and Colv to NA. We normalize Our data by the scale() function.

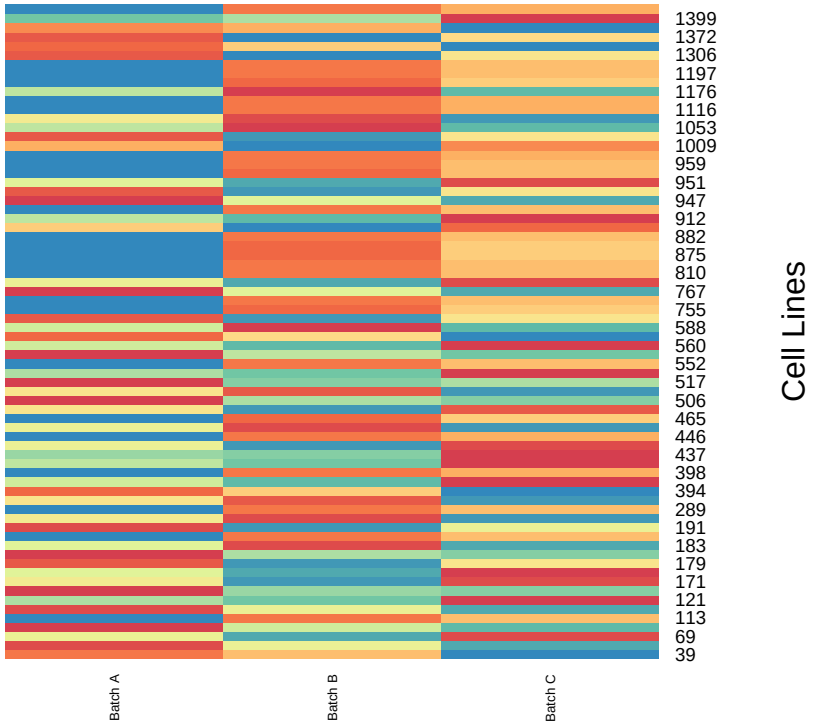
```
#Now lets plot our data as a heatmap.
# to add colour
library(RColorBrewer)
mycolor <- colorRampPalette(brewer.pal(8, "Spectral"))(25)

Con_1 <- as.matrix(Concentration_1_mapdata[,2:4])
rownames(Con_1) <- Concentration_1_mapdata$cell_line_id
heat_1<-heatmap(scale(Con_1),Rowv = NA, Colv = NA ,col = mycolor, xlab="Batch", ylab="Cell Lines", main=
```

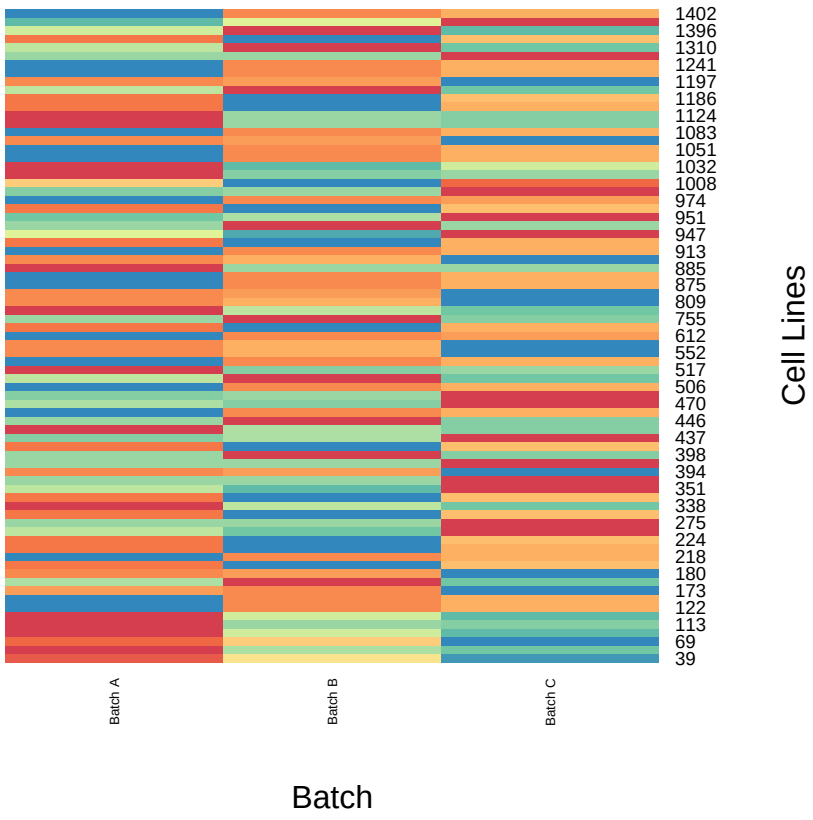
Cell roundness for Fibronectin Concentration 1


```
Con_5 <- as.matrix(Concentration_5_mapdata[,2:4])
rownames(Con_5) <- Concentration_5_mapdata$cell_line_id
heat_5<-heatmap(scale(Con_5),Rowv = NA, Colv = NA ,col = mycolor, xlab="Batch", ylab="Cell Lines", main=
```

Cell roundness for Fibronectin Concentration 5




```
Con_25 <- as.matrix(Concentration_25_mapdata[,2:4])
rownames(Con_25) <- Concentration_25_mapdata$cell_line_id
heat_25<-heatmap(scale(Con_25), Rowv = NA, Colv = NA ,col = mycolor, xlab="Batch", ylab="Cell Lines", ma
```




```
#If you want you can print these together like shown above in an example or use pfam.
```

##V4

Create an R object containing only info for cell and nucleus roundness (keeping the first 4 columns as well)

Reshape the data (by using either the melt function from the reshape package or the gather function from the tidyverse package)

Create two boxplots (one for nucleus roundness and one for cell roundness) showing the roundness value for each short name.

```
#Creating the R object using select function and using pipes
newrounddata <- dataset_1%>%
select(cell_line_id,short_name,concentration,batch,roundness_mean,nucleus_roundness_mean)%>%
group_by(cell_line_id,short_name,concentration, batch)%>%
summarise(cellroundness = mean(roundness_mean), nucleus_roundness = mean(nucleus_roundness_mean))
```

```
## `summarise()` regrouping output by 'cell_line_id', 'short_name', 'concentration' (override with `.gr
```

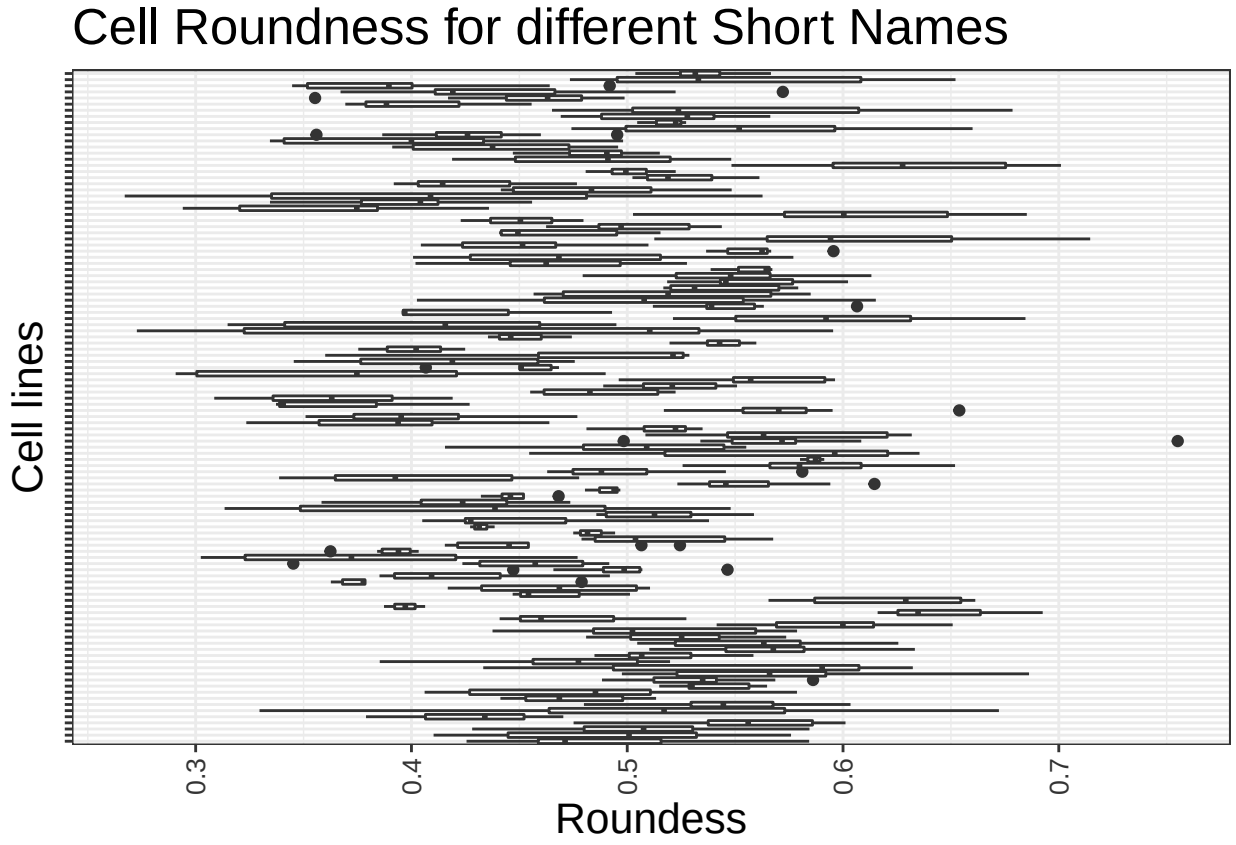
```
#Reshaping using gather
round_gather <- newrounddata %>%
gather(key = "Roundness_Type", value = "Roundness",-cell_line_id,-batch,-concentration,-short_name)

#Creating Box Plots for cell roundness and different short names
namecell_round <- round_gather %>%
```

```
   select(short_name,Roundness_Type,Roundness)%>%
   filter(Roundness_Type == 'cellroundness')
```

```
## Adding missing grouping variables: `cell_line_id`, `concentration`
```

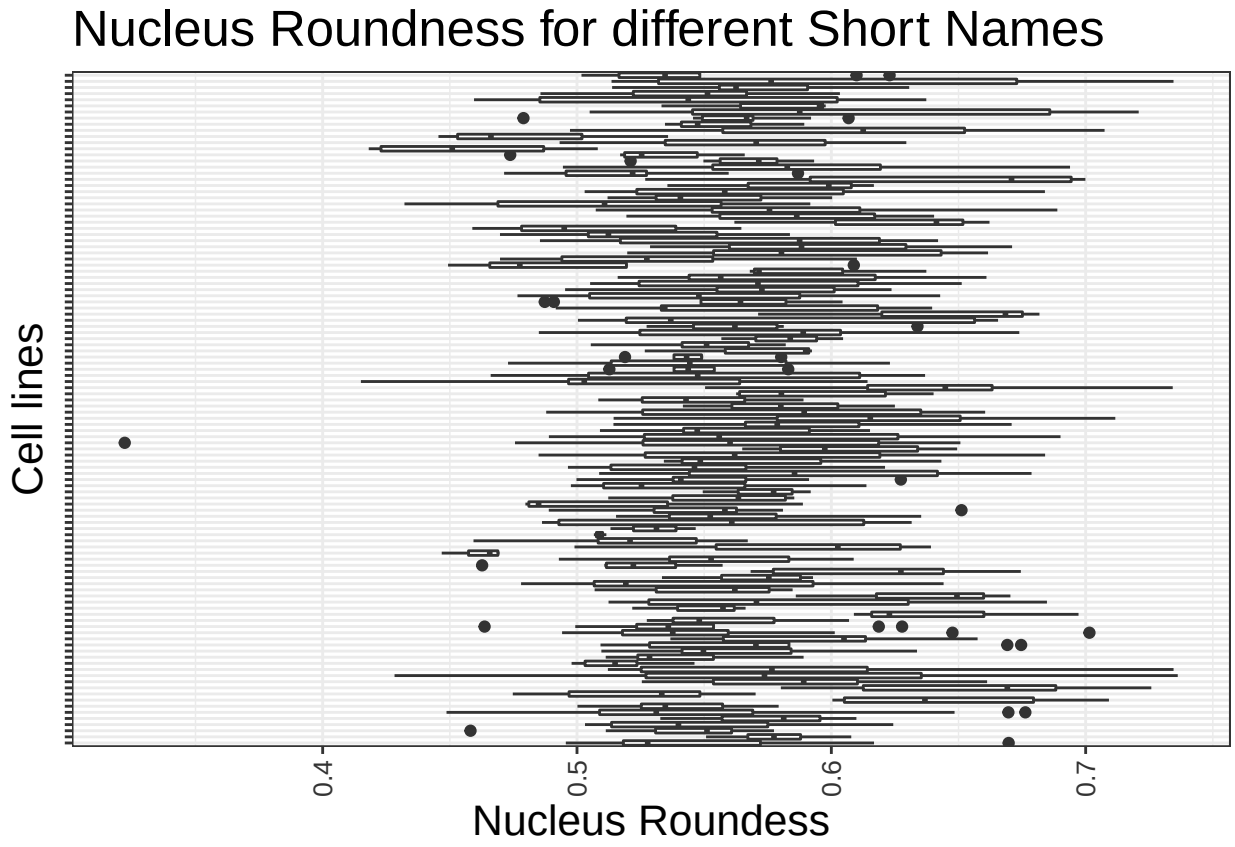
```
ggplot(data = namecell_round, mapping = aes(x =namecell_round$Roundness , y =namecell_round$short_name ))+
geom_boxplot()+
   labs(title = "Cell Roundness for different Short Names",
x = "Roundess",
y = "Cell lines")+
   theme_bw() +
theme(axis.text.x = element_text(colour = "grey20", size = 10, angle = 90, hjust = 0.5, vjust = 0.5),
axis.text.y = element_text(colour = "grey20", size = 0.1),
strip.text = element_text(face = "italic"),
text = element_text(size = 16))
```




```
#Creating Box Plots for Nucleus roundness and different short names
namenucleus_round <- round_gather %>%
   select(short_name,Roundness_Type,Roundness)%>%
   filter(Roundness_Type == 'nucleus_roundness')
```

```
## Adding missing grouping variables: `cell_line_id`, `concentration`
```

```
ggplot(data = namenucleus_round, mapping = aes(x =namenucleus_round$Roundness , y =namenucleus_round$sh
geom_boxplot()+
   labs(title = "Nucleus Roundness for different Short Names",
x = "Nucleus Roundess",
y = "Cell lines")+
   theme_bw() +
theme(axis.text.x = element_text(colour = "grey20", size = 10, angle = 90, hjust = 0.5, vjust = 0.5),
axis.text.y = element_text(colour = "grey20", size = 0.1),
strip.text = element_text(face = "italic"),
text = element_text(size = 16))
```



##V5

Interrogate the and create a plot. Describe which biological question you are trying to address.

##AnswerV5 I wanted to see the relation between: Nucleus area and Cell roundness for different concentrations of fibronectin.

The Biological question stems from whether cells with a larger nucleus have rounder surface?

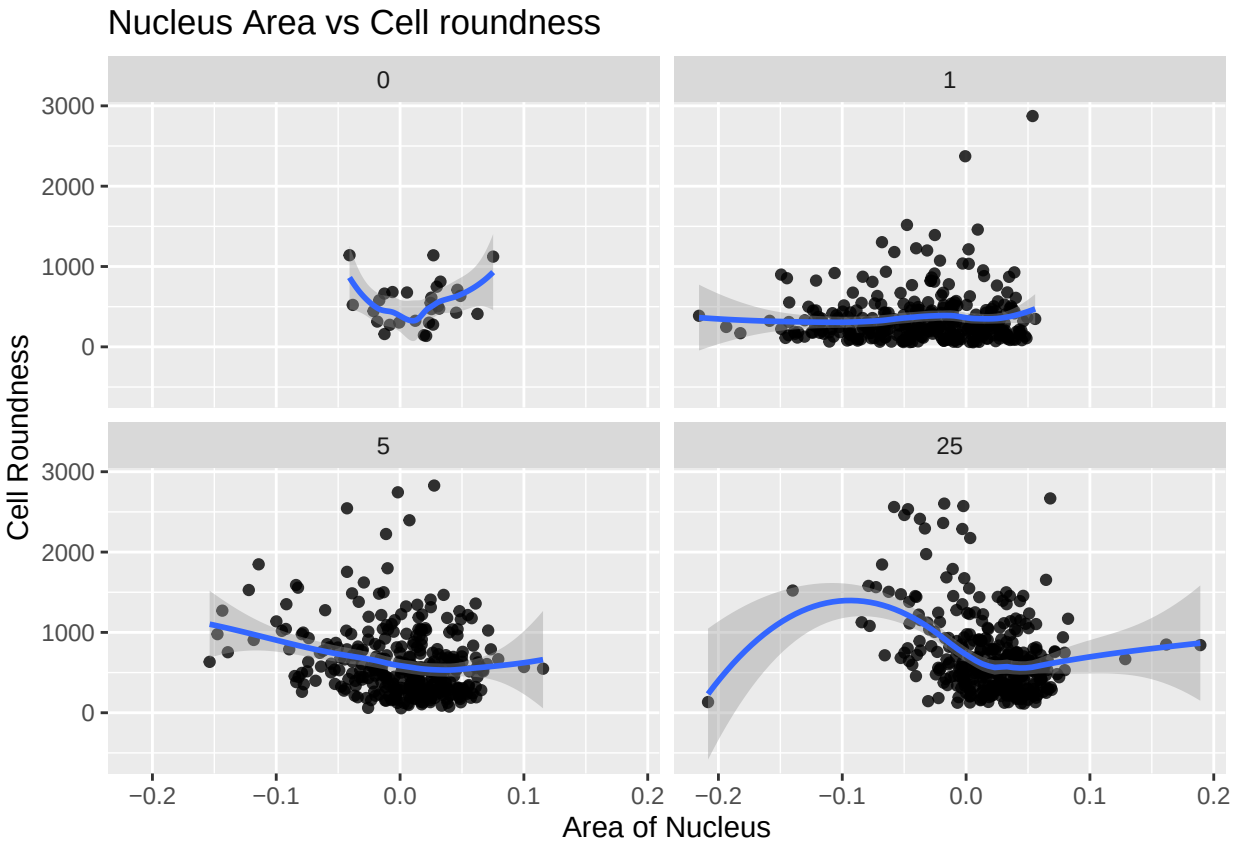
Hypothesis Large Nucleus —-> maybe more nucleus activity —> more cell roundness

```
abnormal <- dataset_1 %>%
select(short_name, batch, concentration, nucleus_area_mean,roundness_mean)%>%
group_by(short_name, batch , concentration)%>%
summarise(Area_of_Nucleus= mean(nucleus_area_mean), Cell_roundness = mean(roundness_mean))
```

```
## `summarise()` regrouping output by 'short_name', 'batch' (override with `.groups` argument)
```

```
my_plot <- ggplot(data = abnormal, mapping = aes(x =abnormal$Area_of_nucleus , y = abnormal$Cell_roundn
 clump_plot +
   geom_point(alpha = 0.8)+
   geom_smooth()+
    facet_wrap(facets = vars(concentration))+
    labs(title = "Nucleus Area vs Cell roundness",
x = "Area of Nucleus",
y = "Cell Roundness")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



My answer is that there is no correlation since the correlation line is horizontal.

Data Proves the hypothesis is wrong.

Doesn't mean my hypothesis is wrong! We just have to make a better experiment and read more literature!

Assignment2Stats

VatsalaTewari

1/5/2021

#Stats with R

For this set of exercises, please use the files in the stats subfolder

SR 1

For this SR you will use the `car.data.csv` file. This dataset represents the sales of different models of car in the year 1993.For this SR only two colums are selected, one selected the airbags type and one the car type. Here we aim to find out if there is a statistical significant relationship between the types of car sold and the type of Air bags it has.

SR 1.1

Are these data qualitative or quantitative?

Answer 1.1

The data is Qualitative, since it's type of airbag and car type which is discriptive and is a characteristic feature.

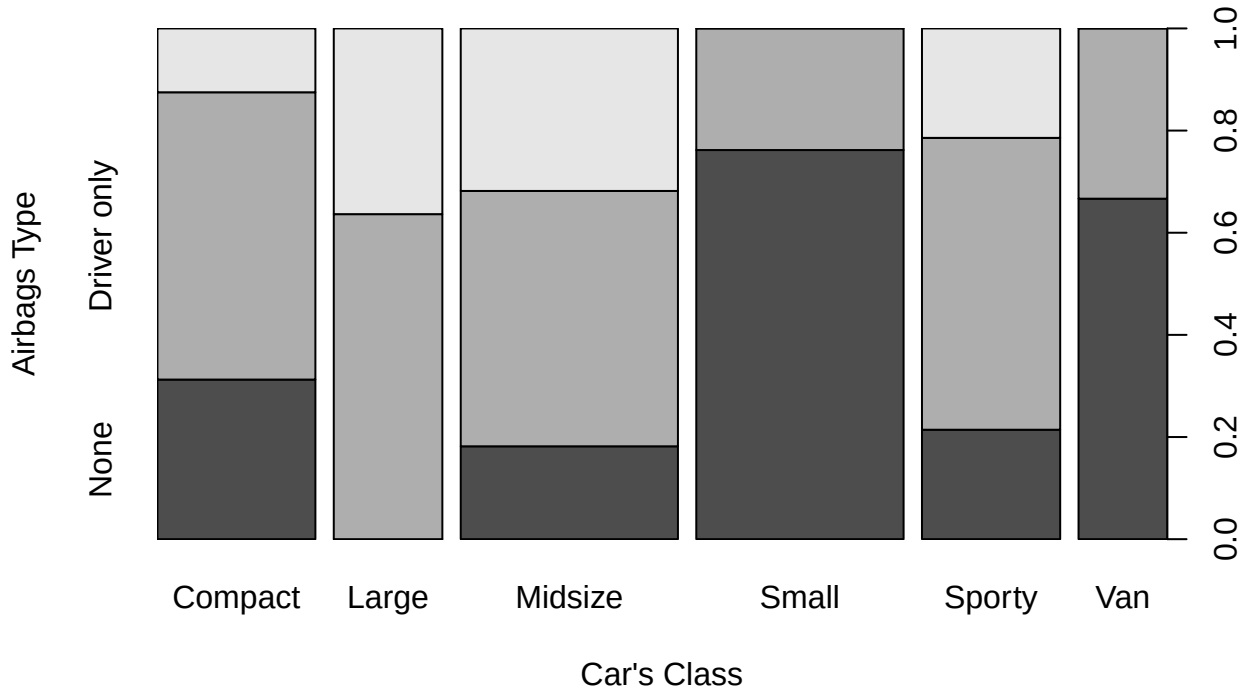
SR 1.2

Plot the data, using the `plot()` function

```
#reading the file to cars.data
cars.data <- read.csv("car.data.csv")
head(cars.data)
```

```
##              AirBags    Type
## 1               None   Small
## 2 Driver & Passenger Midsize
## 3        Driver only Compact
## 4 Driver & Passenger Midsize
## 5        Driver only Midsize
## 6        Driver only Midsize
```

```
plot(factor(cars.data$Type), factor(cars.data$AirBags), xlab = "Car's Class", ylab = "Airbags Type")
```




```
table(cars.data)
```

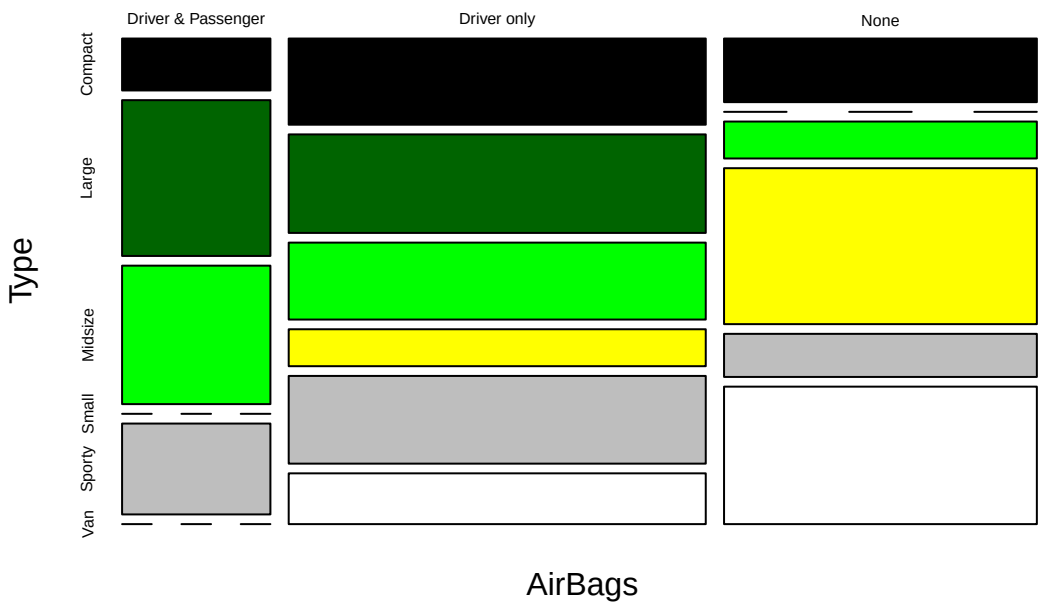
```
##                     Type
## AirBags              Compact Large Midsize Small Sporty Van
##   Driver & Passenger       2     4       7     0      3   0
##   Driver only              9     7      11     5      8   3
##   None                     5     0       4    16      3   6
```

```
#proportion of sample collected for each car if made similar then a better plot can be made.
contingency.table <- table(cars.data)
contingency.table100<-prop.table(contingency.table,2)
contingency.table100<-round(contingency.table100*100)
contingency.table100
```

```
##                     Type
## AirBags              Compact Large Midsize Small Sporty Van
##   Driver & Passenger      12    36      32     0     21   0
##   Driver only             56    64      50    24     57  33
##   None                    31     0      18    76     21  67
```

```
#now every car type has same proportion of sample collected.


#Plotting
plot(contingency.table100, col=c("black","darkgreen","green","yellow","grey","white"),cex.axis=0.50)
```

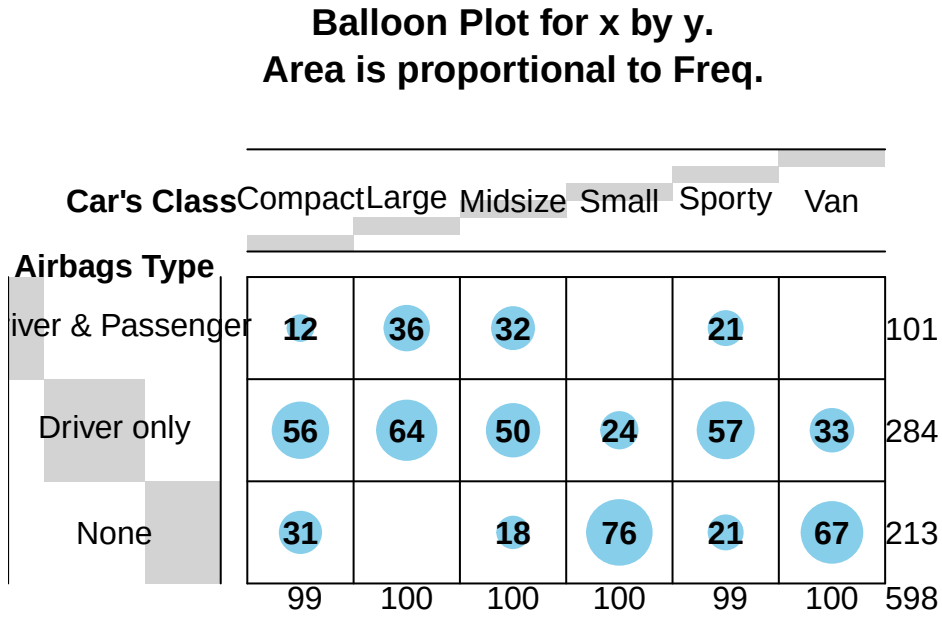



```
#Another Information presentation.
library(gplots)
```

```
##
## Attaching package: 'gplots'
```

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

```
balloonplot(t(contingency.table100) , xlab = "Car's Class", ylab = "Airbags Type", cex.axis=0.75)
```



SR 1.3

Which test is more appropriate for this dataset?

Answer

We can say there are three variable { type of airbags} and six groups {type of car} Chi Square test : as these are independent groups and data is categorical

SR 1.4

Perform the chosen statistical test and describe the obtained results.

Answer 1.4

Since the P value is significant we can reject the null hypothesis. Hence there is statistcal significance enough to point towards that these numbers don't come together by chance. But using the data we cant show any causation.

```
chisq <- chisq.test(contingency.table)
```

```
## Warning in chisq.test(contingency.table): Chi-squared approximation may be
## incorrect
```

```
chisq
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency.table
## X-squared = 33.001, df = 10, p-value = 0.0002723
```

SR 2

A teacher wants to understand if the 2 different nutrition education programs he teaches affect the mean sodium intake of the students. To do that he needs to ask students to keep diaries of what they eat for a week, and then calculate the daily sodium intake in milligrams. First he needs to calculate how many students he needs in his cohort to have a 80% power and a statistica significance of 5%, so he runs a power analysis based on the following preliminary data:

mean1= 1287.5 mean2=1246.25 sd= 170

Perform the power analysis. What's the minimum sample size? Write the code and describe the results.

Answer

```
mean1<- 1287.5
mean2<-1246.25
delta<-1287.5-1246.25
sd<- 170
power.t.test(delta=1287.5-1246.25, sd = 170, sig.level = 0.05, power = 0.8)
```

```
##
##      Two-sample t test power calculation
##
##               n = 267.5806
##           delta = 41.25
##              sd = 170
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

We would need a sample size of 536. Since n is rounded to 268 and the test being a two sided test would require double the number of samples.

SR 3

For this SR you will use the `sodium_intake.csv` file. You now have the measurements.

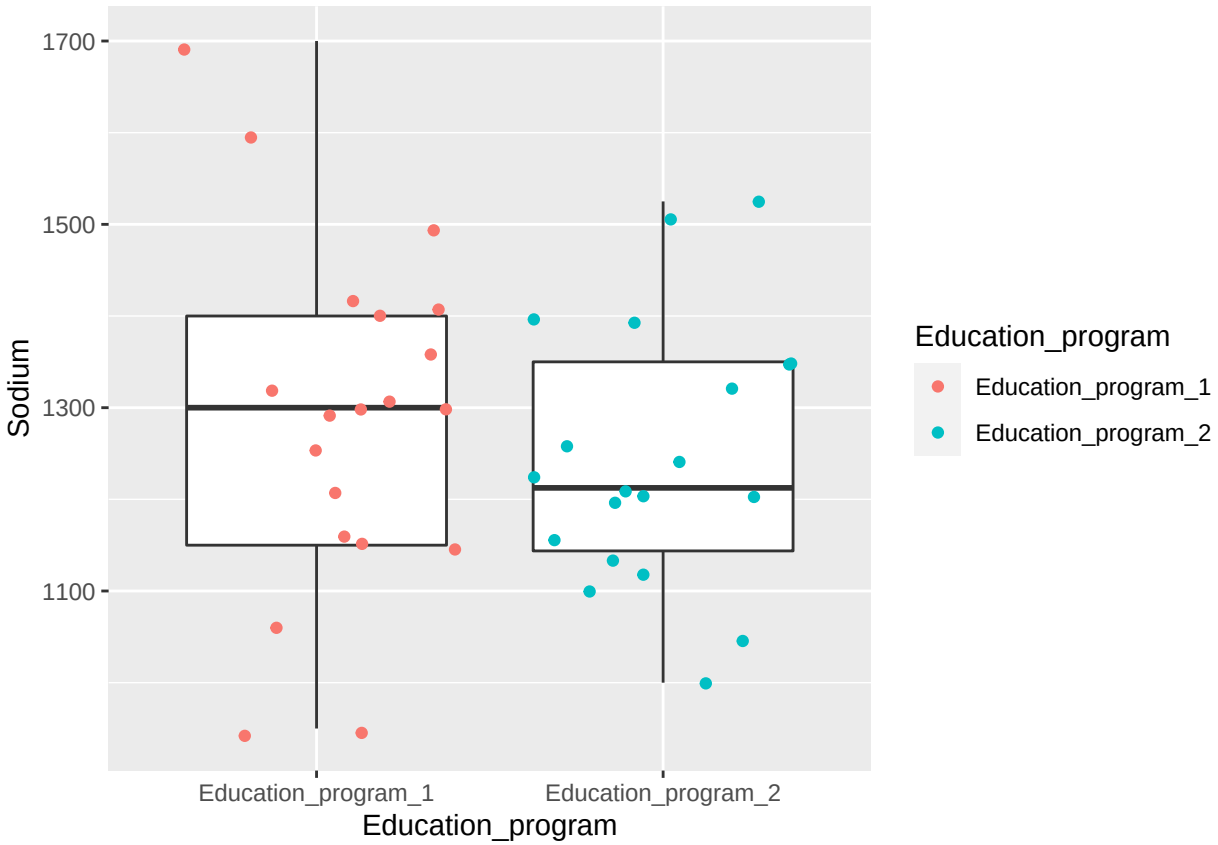
```
library(readr)
sodium.intake <- read_csv("sodium_intake.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Education_program = col_character(),
##   Student = col_character(),
##   Sodium = col_double()
## )
```

SR 3.1

Plot the data. Choose at least one of the functions we used in class. Show the code and describe what you see from the plot.

```
library(ggplot2)
ggplot(sodium.intake, aes(Education_program, Sodium))+geom_boxplot()+geom_jitter(aes(color=Education_pr
```



SR 3.2

Are the data parametric? Check that the sata a normally distributed.

Answer

The Data could be parametric since the box plots and the data str fits the four assumptions.

The data seems to be normal data. There seems to be homogenius Variance. The data is not skewed or biased [Interval data] The data is from indepent samples or seperate student. [Student a only gave one data sample, or is only part of education program 1]

```
library(pastecs)
tapply(sodium.intake$Sodium,sodium.intake$Education_program, stat.desc, basic=F, desc=F, norm=T)
```

```
## $Education_program_1
##    skewness    skew.2SE    kurtosis    kurt.2SE normtest.W normtest.p
##  0.1186154   0.1158120 -0.4626583 -0.2331046   0.9721232   0.7989385
##
## $Education_program_2
##    skewness    skew.2SE    kurtosis    kurt.2SE normtest.W normtest.p
##  0.3030858   0.2959225 -0.8453012 -0.4258944   0.9677228   0.7062500
```

Since the normaltest p value is always non significant we dont need to reject the null hypothesis hence data is normally distributed.

SR 3.3

Perform the appropriate test. Write the code and describe the results.

Answer 3.3

Appropriate test :- Independent T test

```
res <- t.test(sodium.intake$Sodium~sodium.intake$Education_program, var.equal=T)
res
```

```
##
##  Two Sample t-test
##
## data:  sodium.intake$Sodium by sodium.intake$Education_program
## t = 0.76722, df = 38, p-value = 0.4477
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -67.59215 150.09215
## sample estimates:
## mean in group Education_program_1 mean in group Education_program_2
##                           1287.50                           1246.25
```

```
res$p.value
```

```
## [1] 0.4476896
```

With a P value of 0.4476896 and not much difference in means we can say that the two education programs and thier sodium levels in the students on those programs show no significance.

SR 4

The teacher added another nutrition education program to his research. The file with the additional measurements is the `sodium_intake_3_groups.csv`. Be careful in The question is the same but with the addition of another program you cannot use the t-test anymore. Which test can you use instead?

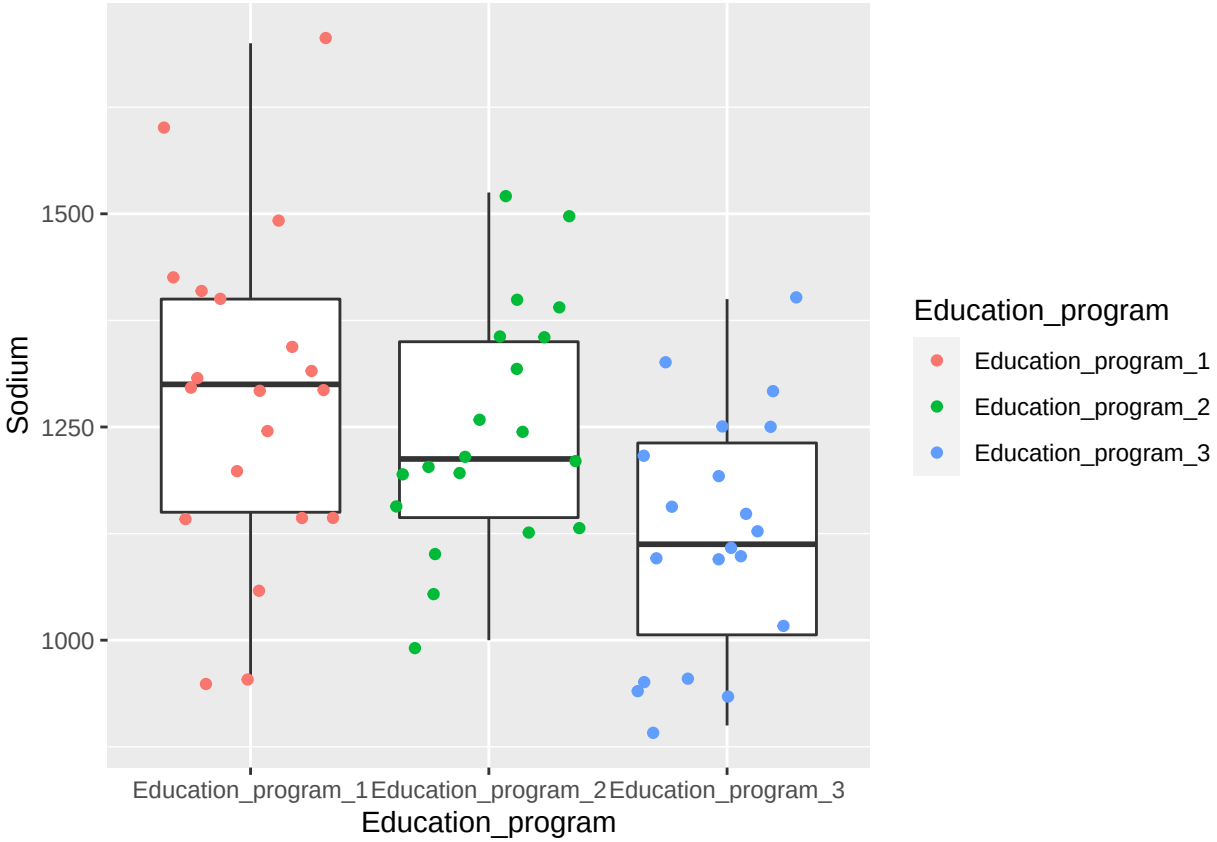
SR 4.1

Plot the data and describe.

```
#reading the file to sodium.data
sodium.data <- read.csv("sodium_intake_3_groups.csv")
head(sodium.data)
```

```
##     Education_program Student Sodium
## 1 Education_program_1       a   1200
## 2 Education_program_1       b   1400
## 3 Education_program_1       c   1350
## 4 Education_program_1       d    950
## 5 Education_program_1       e   1400
## 6 Education_program_1       f   1150
```

```
library(ggplot2)
ggplot(sodium.data, aes(Education_program, Sodium))+geom_boxplot()+geom_jitter(aes(color=Education_prog
```



Students in Education program 1 have average higher sodium then those in education program 2 which have higher sodium than those in education program 3.

There is approximately equal variance as made a guess just by looking at the graph ### SR 4.2

Perform the appropriate test. Write the code and describe the results.

Answer 4.2

The Data is parametric since the blox plots and the data str fits the four assumptions.

The data seems to be normal data. There seems to be homogenius Variance. The data is not skewed or biased [Interval data] The data is from indepent samples or seperate student. [Student a only gave one data sample, or is only part of education program 1]

```
library(pastecs)
tapply(sodium.data$Sodium,sodium.data$Education_program, stat.desc, basic=F, desc=F, norm=T)
```

```
## $Education_program_1
##    skewness    skew.2SE    kurtosis    kurt.2SE normtest.W normtest.p
##   0.1186154   0.1158120 -0.4626583 -0.2331046   0.9721232   0.7989385
##
## $Education_program_2
##    skewness    skew.2SE    kurtosis    kurt.2SE normtest.W normtest.p
##   0.3030858   0.2959225 -0.8453012 -0.4258944   0.9677228   0.7062500
##
## $Education_program_3
##     skewness     skew.2SE     kurtosis     kurt.2SE   normtest.W   normtest.p
##   0.07575633   0.07396586 -1.09788749 -0.55315680   0.95834680   0.51137535
```

Since the normaltest p value is always non significant we dont need to reject the null hypothesis hence data is normally distributed.

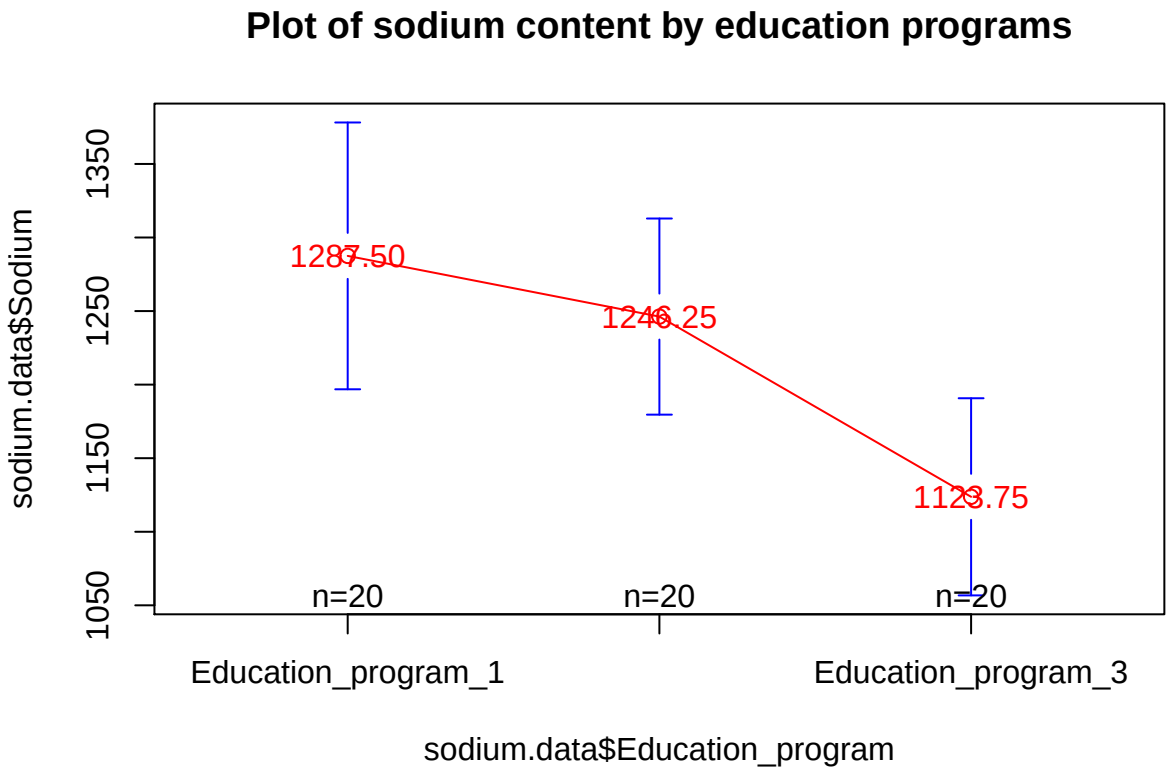
Similary using the levene test we can also find out the variance p value.

We can use the ANOVA test

ANOVA test since data is parametric and has more than 2 groups[3 educational programs]

```
means<- round(tapply(sodium.data$Sodium,
                     sodium.data$Education_program,mean), digits=2)
library(gplots)

plotmeans(sodium.data$Sodium~          sodium.data$Education_program,
digits=2, col="red", mean.labels=T, main="Plot of sodium content by education programs")
```




```
aov_cont<- aov(sodium.data$Sodium~          sodium.data$Education_program)

summary(aov_cont)
```

```
##                               Df  Sum Sq Mean Sq F value  Pr(>F)
## sodium.data$Education_program  2  290146  145073   5.558 0.00624 **
## Residuals                     57 1487812   26102
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F value is large, and at significance level 0.001 the data is significant since its less than 0.05.

Hence there is a significance difference is education program students were on and the sodium intake.

SR 4.3

Perform a post-hoc test. Write the code and describe the results.

```
tuk<- TukeyHSD(aov_cont)
#To view
tuk
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = sodium.data$Sodium ~ sodium.data$Education_program)
##
## $`sodium.data$Education_program`
##                                           diff       lwr         upr     p adj
## Education_program_2-Education_program_1  -41.25 -164.1941  81.6941219 0.7000083
## Education_program_3-Education_program_1 -163.75 -286.6941 -40.8058781 0.0061739
## Education_program_3-Education_program_2 -122.50 -245.4441   0.4441219 0.0510271
```

The only groups with stastical significance is between Educational_program_1 and Educational_program_3 since only its p adjusted value is significant.

SR 5

Five teachers recorded several measurements for students in their classes related to their nutrition education program: Grade, Weight in kilograms, intake of Calories per day, daily Sodium intake in milligrams, and Score on the assessment of knowledge gain. The measurements are collected in the file `assessment_of_knowledge_gain.csv` Is there a correlation between sodium intake and calories gained?

```
library(readr)
assessment_of_knowledge_gain <- read_csv("assessment_of_knowledge_gain.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Teacher = col_character(),
##   Grade = col_double(),
##   Weight = col_double(),
##   Calories = col_double(),
##   Sodium = col_double(),
##   Score = col_double()
## )
```

SR 5.1

Perform a linear regression analysis. What's the relationship between the two variables? Write the code and describe.

```
fit.correlation<-lm(assessment_of_knowledge_gain$Calories~assessment_of_knowledge_gain$Sodium)


fit.correlation
```

```
##
## Call:
```

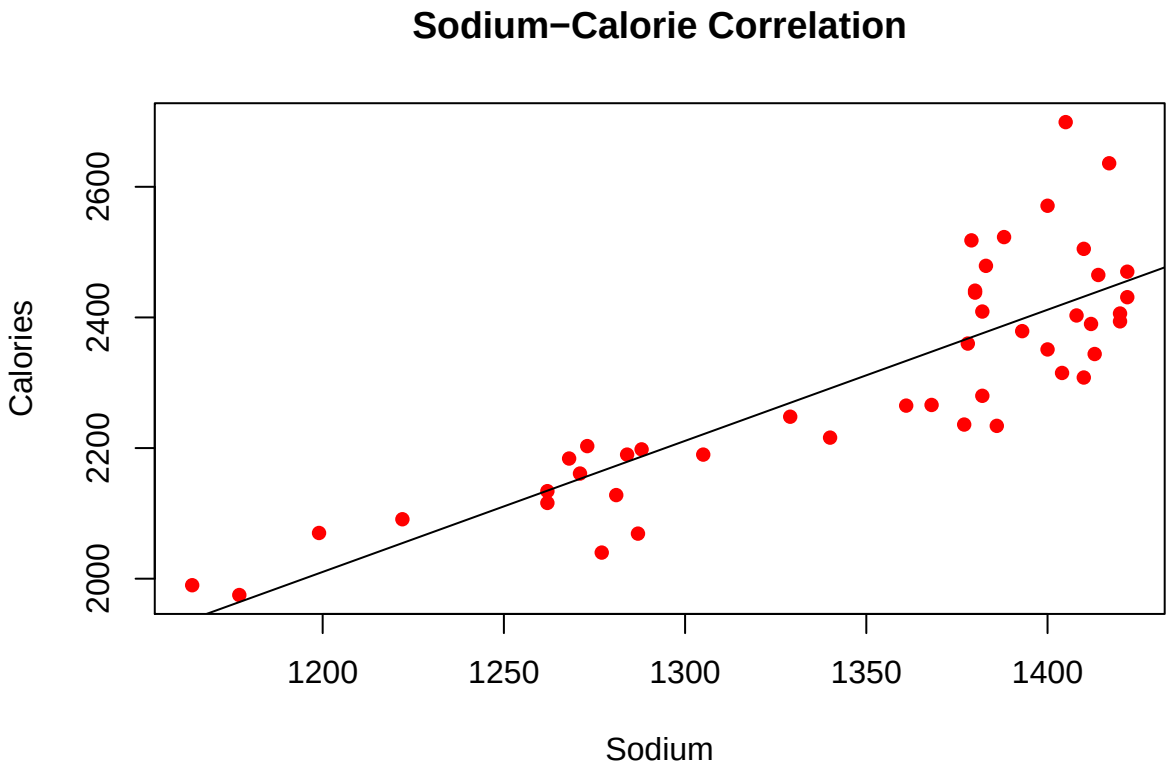
```
## lm(formula = assessment_of_knowledge_gain$Calories ~ assessment_of_knowledge_gain$Sodium)
##
## Coefficients:
##                             (Intercept)  assessment_of_knowledge_gain$Sodium
##                                -398.131                                2.007
```

Here the Intercept doesnt mean anything since never can the sodium intake be 0. The slope however shows a strong possitive correlation [2.007] .

SR 5.2

Plot the data

```
plot(assessment_of_knowledge_gain$Calories~assessment_of_knowledge_gain$Sodium,pch=16,main="Sodium-Calo
abline(fit.correlation)
```



RNAseq

Vatsala Tewari

11/01/2021

#RNAseq analysis

For this set of exercises please download the files in the RNAseq subfolder

#RSQ.1

Order the following steps to make an appropriate workflow for detecting differential expression using RNA-seq:

1. Library preparation
2. Sequencing
3. Differential expression analysis
4. Read mapping
5. Quantification

#Aswer

1. Library type/preperation.
2. Sequence
3. Read Mapping
4. Qantification
5. Differential Expression analysis

#RSQ.2

In a RNA-seq experiment, raw sequence data are in the following file:

1. CRAM file
2. FASTQ file
3. SAM file
4. BAM file

#Answer 2. FASTQ File

#RSQ.3

Describe the difference between classical alignment and pseudo alignment. Give some examples of the most used tools

#Answer Classical Allignment: Trying to all of the read data to the refrence genome. Intron in the sequence cause incorrect mapping. Presense of pseudogenes cause problems by mapping of read to them rather than the correct position. Requires heavy Computation.

Tools: TOPHAT , STAR.

Pseudo Allignemnt: Maps read to chunks (kmer) of the genome stored in the library. Binding to only inofrmative and perfect matches. Pseudo Allignment like Kallisto take very fast compared to classical alignment tools for the analysis. Tools: KALLISTO

#RSQ.4 Use the fastq file and perform a quality contol of the reads using FastQC. Show and describe some of the results from the report.

#Answer The Basic Stastics tell Us the File name, File type, Encoding - type of platform to create FastQ file : illumina 1.9/sanger, Total sequences 10873108 and sequence lenght which varies from 21-100,%GC content being 44%.

Base Quality:Quality control graph on the per base quality function tells us the PHRED score for the base position. This tells us that since most box plots are in the green zone of the graph the sequence quality is good. Even as the position of the base pair is further down the read the quality still maintains itself in the green area in the plot.

Sequence Quality: X axsis has mean sequence quality PHRED score and Y axsis is no. of sequences. Average quality of the sequences were around 38 which is a good PHRED score. Negligible amount of sequences had bad quality score since the plot of the graph starts its slope at 24 on the X axis. 9.197% of the sequences have a phred score below 35.

Sequence Content across the bases: A-T and G-C should be same, and we can see the graph the plots on the X axsis the read lenght and on the Y axis the content of base. that there is excellent overlap between the pairs A-T after 15bp till 95bp, but the overlap between C-G the over lap is good, but not as good as the A-T pair. #RSQ.5 Download the count matrix and the related annotation present in the downloaded table.

```
setwd("~/Desktop/DataAnalysis")
library(DESeq2)
```

```
## Loading required package: S4Vectors
```

```
## Loading required package: stats4
```

```
## Loading required package: BiocGenerics
```

```
## Loading required package: parallel
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:parallel':
##
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
```

```
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which.max, which.min

##
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:base':
##
##     expand.grid

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following object is masked from 'package:grDevices':
##
##     windows

## Loading required package: GenomicRanges

## Loading required package: GenomeInfoDb

## Loading required package: SummarizedExperiment

## Loading required package: MatrixGenerics

## Loading required package: matrixStats

##
## Attaching package: 'MatrixGenerics'

## The following objects are masked from 'package:matrixStats':
##
##     colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse,
##     colCounts, colCummaxs, colCummins, colCumprods, colCumsums,
##     colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs,
##     colMads, colMaxs, colMeans2, colMedians, colMins, colOrderStats,
##     colProds, colQuantiles, colRanges, colRanks, colSdDiffs, colSds,
##     colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
##     colWeightedMeans, colWeightedMedians, colWeightedSds,
##     colWeightedVars, rowAlls, rowAnyNAs, rowAnys, rowAvgsPerColSet,
##     rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
##     rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps,
##     rowMadDiffs, rowMads, rowMaxs, rowMeans2, rowMedians, rowMins,
##     rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
##     rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars,
##     rowWeightedMads, rowWeightedMeans, rowWeightedMedians,
##     rowWeightedSds, rowWeightedVars
```

```
## Loading required package: Biobase

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.

##
## Attaching package: 'Biobase'

## The following object is masked from 'package:MatrixGenerics':
##
##     rowMedians

## The following objects are masked from 'package:matrixStats':
##
##     anyMissing, rowMedians
```

```
library(readr)
SRP049988_raw_counts <- read_csv("SRP049988.raw_counts.csv")
```

```
## Warning: Missing column names filled in: 'X1' [1]

##
## -- Column specification --------------------------------------------------------
## cols(
##   X1 = col_character(),
##   CASE_1 = col_double(),
##   CASE_2 = col_double(),
##   CASE_3 = col_double(),
##   CASE_4 = col_double(),
##   CASE_5 = col_double(),
##   CTRL_1 = col_double(),
##   CTRL_2 = col_double(),
##   CTRL_3 = col_double(),
##   CTRL_4 = col_double(),
##   CTRL_5 = col_double(),
##   width = col_double()
## )
```

```
SRP049988.colData <- read.delim("~/Desktop/DataAnalysis/SRP049988.colData.tsv")

#Setting the data as matrix and adding the first column as row names

Data <- as.matrix(subset(SRP049988_raw_counts[-1], select = c(-width)))
rownames(Data) <- SRP049988_raw_counts$X1

designFormula <- "~group"

dds <- DESeqDataSetFromMatrix(countData = Data,
colData = SRP049988.colData,
design = as.formula(designFormula))
```

```
## converting counts to integer mode

## Warning in DESeqDataSet(se, design = design, ignoreRank): 2 duplicate rownames
## were renamed by adding numbers

## Warning in DESeqDataSet(se, design = design, ignoreRank): some variables in
## design formula are characters, converting to factors
```

```
#Removing the genes which arent expressed
dds <- dds[ rowSums(DESeq2::counts(dds)) > 1, ]

dds <- DESeq(dds)
```

```
## estimating size factors

## estimating dispersions

## gene-wise dispersion estimates

## mean-dispersion relationship

## final dispersion estimates

## fitting model and testing
```

```
DEresults = results(dds, contrast = c("group", 'CASE', 'CTRL'))
#Ordering them by P value.
DEresults <- DEresults[order(DEresults$pvalue),]
#Differential analysis complete

#Most differential expressed genes are
head(DEresults,10)
```

```
## log2 fold change (MLE): group CASE vs CTRL
## Wald test p-value: group CASE vs CTRL
## DataFrame with 10 rows and 6 columns
##         baseMean log2FoldChange     lfcSE      stat       pvalue         padj
##        <numeric>      <numeric> <numeric> <numeric>    <numeric>    <numeric>
## CES1    127785.8        4.92947 0.1722360   28.6205 3.73938e-180 6.52259e-176
## HHIP     15143.0       -2.41985 0.0998128  -24.2439 7.67682e-130 6.69534e-126
## MYO1D    34380.1        2.58027 0.1092321   23.6219 2.29709e-123 1.33561e-119
## TJP2     15139.6        2.09252 0.0888465   23.5521 1.19394e-122 5.20649e-119
## KDM5D    29970.7        5.55159 0.2450029   22.6593 1.13036e-113 3.94338e-110
## EIF1AY   21738.2        4.80951 0.2130724   22.5722 8.13239e-113 2.36422e-109
## USP9Y    29712.5        4.95670 0.2252645   22.0039 2.64166e-107 6.58264e-104
## CDH2     49440.2        1.34800 0.0630307   21.3864 1.78880e-101  3.90027e-98
## TMEM98   20897.9        2.66667 0.1310356   20.3507  4.57281e-92  8.86262e-89
## DDX3Y    64568.4        5.06864 0.2527466   20.0542  1.85432e-89  3.23449e-86
```

Perform a differential expression analysis following the tutorial from the last lesson.

What are the top 10 differential expressed genes?

#Answer

Top 10 differential Expressed Genes after Analysis are, CES1 HHIP MYO1D TJP2 KDM5D EIF1AY USP9Y CDH2 TMEM98 DDX3Y