## 

## Task 1

### 2.a. What is the medically relevant insight from the article?

The paper investigates the insulin sensitivity in white adipose tissue women with obesity before and 2 years after bariatric surgery. The primary insight reveals that human WAT displays a selective insulin response in obesity and genes are normalized by weight loss.

### 2.b Which genomics technology/ technologies were used?

RNA sequencing technology called as 5’ cap analysis of gene expression (CAGE) was for used for analysing gene transcripts. The sequencing was performed using Illumina Hi-Seq 2500 or 2000. The obtained FastQ files were split into 4 FastQ files using FastX toolkit and mapped to reference genomes with Bowtie. Bioinformatic analysis was performed followed by statistical analysis using student t-tests.

### 3.

I cannot meaningfully comment on this as I am not from a medical background. But in general the question/hypothesis I have is, can this study be extended/applied to men and if there are similar or different results compared to women. The other stupid question I have, given that the study analysed obese to post-obese patients (weight loss) patients, can this study be extended for work for non-obese to obese (weight gain)?

## 

## 

## 

## 

## Task 4

Writing the R code snippets for Task 4 to 8. The actual R script is available in the repo.

| *# Task 4* *## Q1* sqrt(10) *# ans:3.162278* *## Q2* log2(32) *# answer: 5* *## Q3* sum(1:1000) *# answer: 500500* *# Q4* sum(seq(2,1000,by=2)) *# answer:250500* *## Q5* choose(100,2) *# answer: 4950* *## Q6* choose(100, 3) \* factorial(3) *# answer: 970200* |
| --- |

## Task 5

| *# Q1* df=CO2 *# Q2* help(CO2) *# Description of dataset: The CO2 data frame has 84 rows and 5 columns of data from an experiment on the cold tolerance of the grass species Echinochloa crus-galli.* colnames(df)  *# [1] "Plant" "Type" "Treatment" "conc" "uptake"*  *# Q3* summary(df$uptake) *# answer: median = 28.3* *mean = 27.21* |
| --- |

## Task 6

| *## Q1* ratio\_mean\_median=function(vec){mean(vec)/median(vec)}  *## Q2*  mean\_skip\_minmax=function(vec){  *# using tidyverse discard\_at function to discard min and max values*  library(tidyverse)  vec=discard\_at(vec,which.max(vec))  vec=discard\_at(vec,which.min(vec))  return(mean(vec, na.rm = T)) } |
| --- |

## Task 7

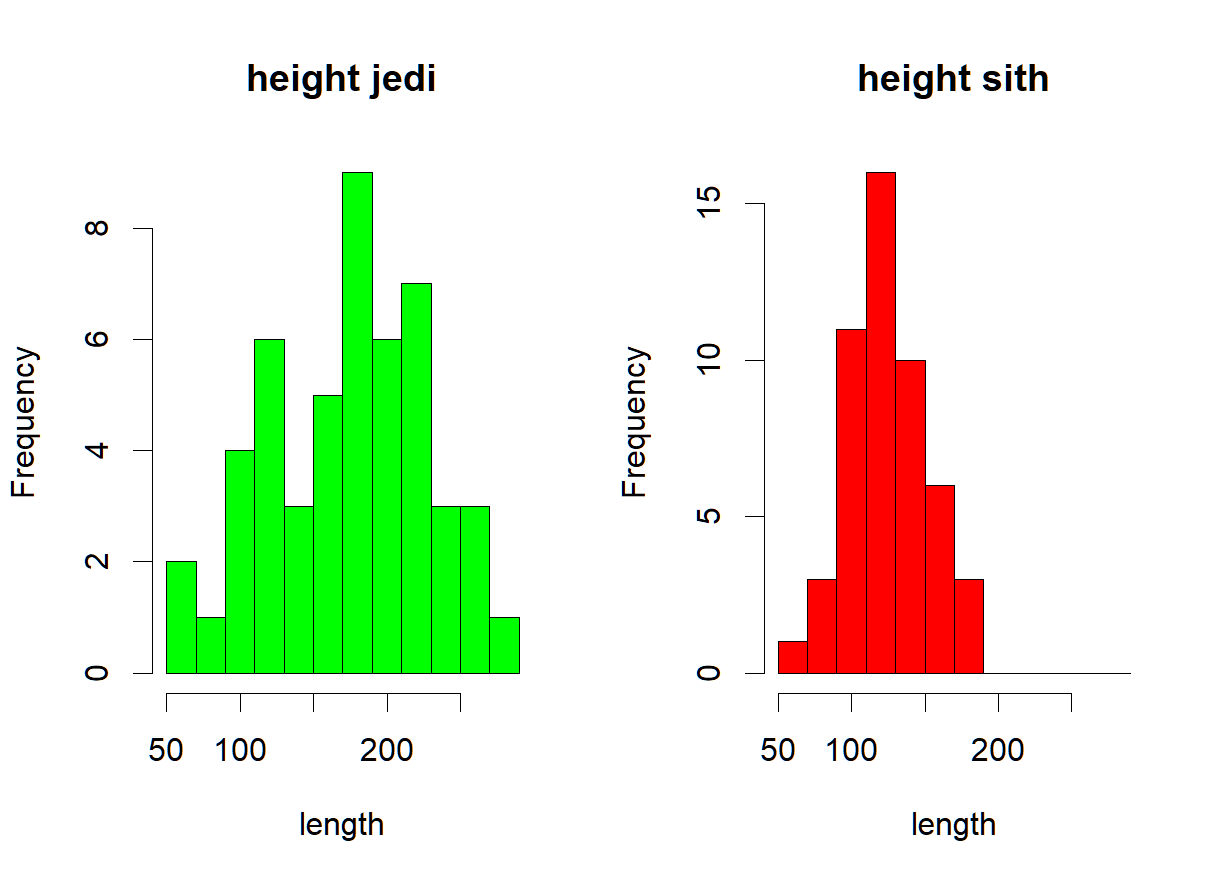
### Q1.

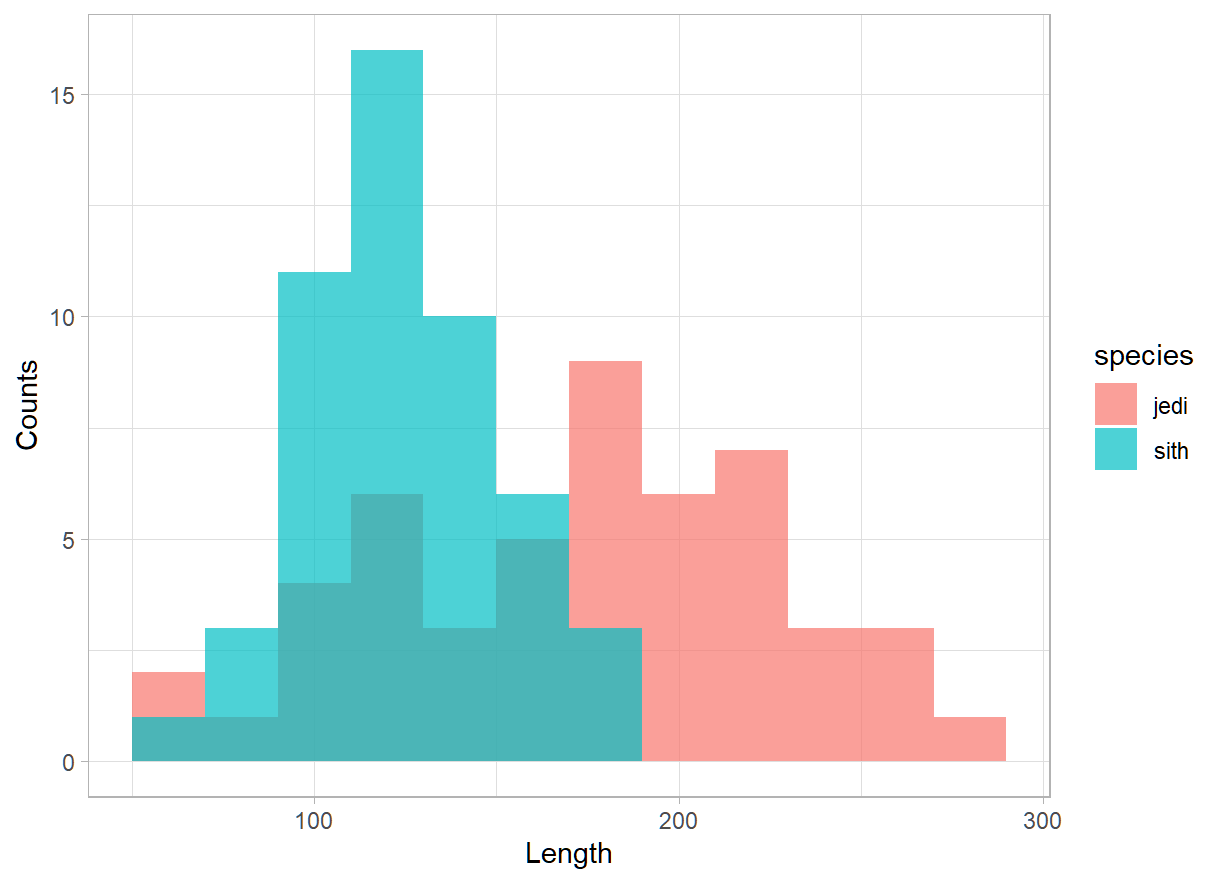
Installing packages.

| install.packages("remotes") library(remotes) install\_url("http://emotion.utu.fi/wp-content/uploads/2019/11/nummenmaa\_1.0.tar.gz",dependencies=TRUE) library(nummenmaa) |
| --- |

* 1a

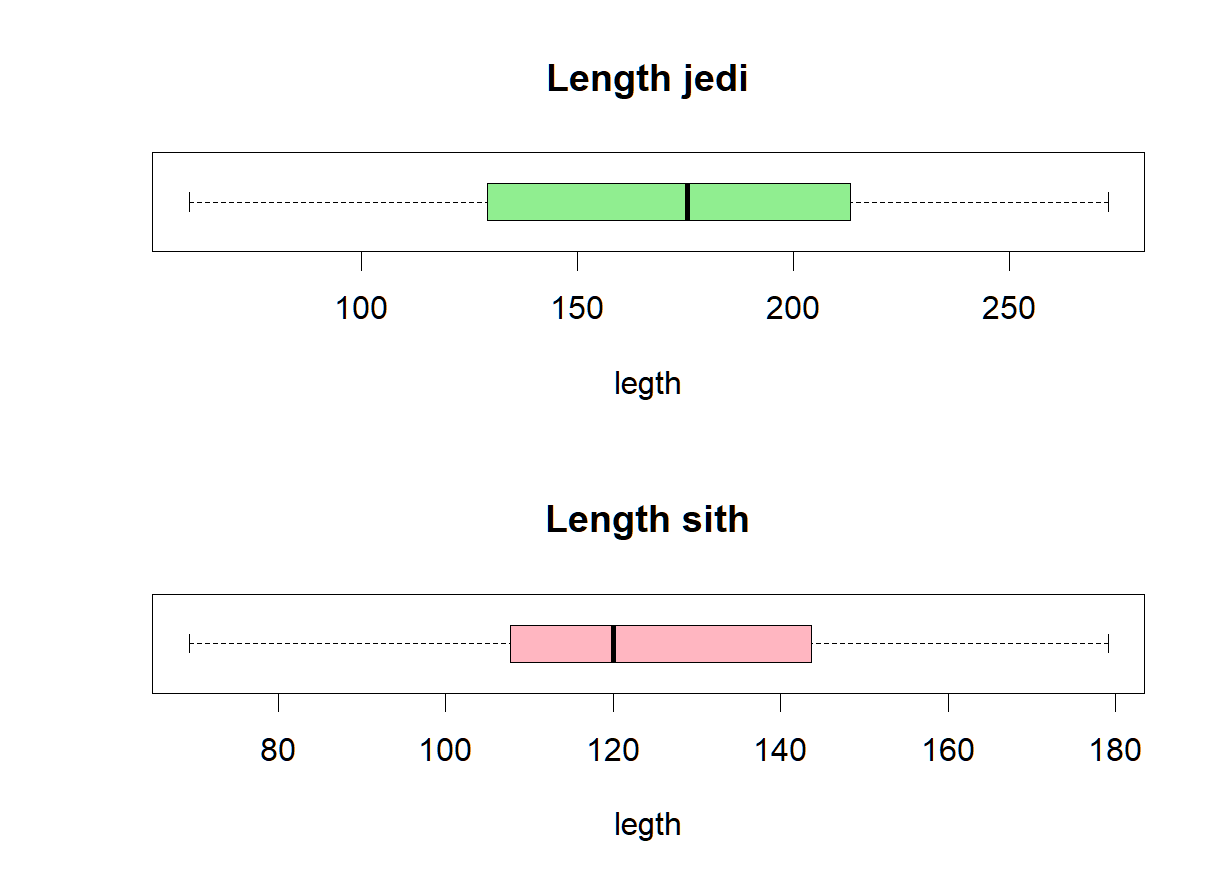
| *# 1.a* df=read.csv2('magic\_guys.csv', sep = ',') par(mfrow=c(1,2)) *# creating panels* *# histogram selecting length for species jedi* p1=hist(as.numeric(df$length[df$species=='jedi']),  breaks = seq(50,300,by=20),  main='height jedi',xlab='legth',col='green') *# histogram selecting length for species sith* p2=hist(as.numeric(df$length[df$species=='sith']),   main='height sith',xlab='legth',col='red',  breaks = seq(50,300,by=20)) dev.off() *# using ggplot* library(ggplot2) *# plot using geom\_histogram and specifying breaks* ggplot(df, aes(x=as.numeric(length),fill=species)) +   geom\_histogram(alpha = 0.7, breaks=seq(50,300,by=20),   position = 'identity',linetype = 1)+  theme\_light()+  scale\_fill\_discrete()+  labs(x="Length", y="Counts") |
| --- |

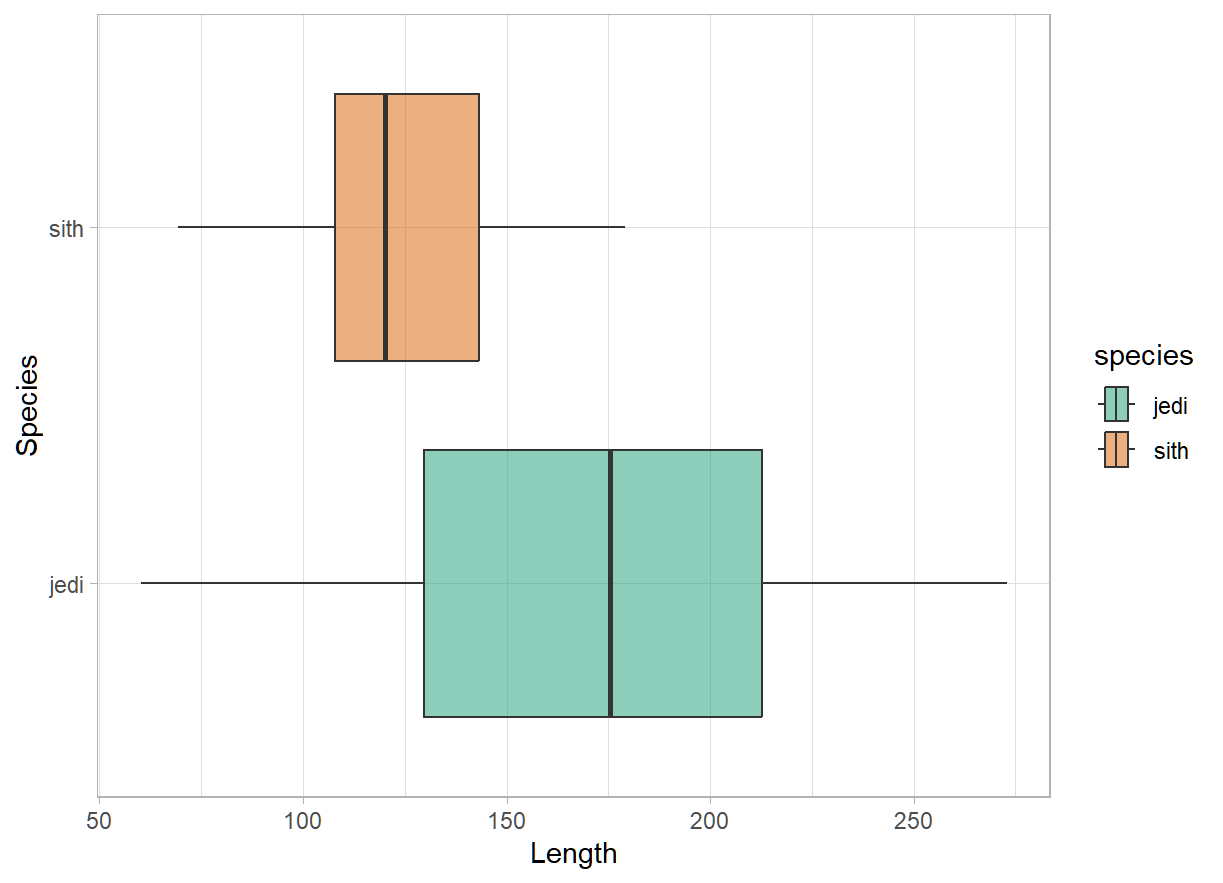




* 1b

| *# 1.b* par(mfrow=c(2,1)) *# boxplots* boxplot(as.numeric(df$length[df$species=='jedi']),  main='Length jedi',xlab='length',col='lightgreen',horizontal=T, alpha=0.1) boxplot(as.numeric(df$length[df$species=='sith']),  main='Length sith',xlab='length',col='lightpink', horizontal = T) dev.off() *# using geom\_boxplot* ggplot(df, aes(x=as.numeric(length),species) ) +   geom\_boxplot(alpha=0.5,aes(fill=species))+  theme\_light()+  scale\_fill\_brewer(palette = 'Dark2')+  labs(x="Length", y="Species", main='Distribution of length for species') |
| --- |





* 1.c

Plot saved below each question respectively. I think it depends on how the plot is intended to be shared. For web or presentations png or svg can be used. For reports pdf or svg are better. Png may loose image quality upon scaling where as pdf/svg can maintain the quality.

### Q2.

| df=read.csv2('microarray\_data.tab',sep = '\t', na.strings = "") *# convert strings to numbers as NA* df=as.data.frame(sapply(df, function(x) as.numeric(x))) View(df) |
| --- |

| *# a* dim(df)  # answer: rows 553 columns 1000 |
| --- |

| *# b (not sure)* nullrows=sapply(df, function(x) sum(is.na(x))) boxplot(nullrows, main='distribution of null values in data', col='cyan',  alpha=0.5, ylab='null rows', xlab='genes') |
| --- |

## 

## 

| *# c* *# using sapply to calculate the percentage for nulls in each column* percentages=sapply(nullrows,function(x)x/553\*100) percentages[percentages>10] percentages[percentages>20] percentages[percentages>50] |
| --- |

| *# d* *# using sapply; for each column check if value is na and replace with mean using if-else* df2 = sapply(df, function(x) ifelse(is.na(x), mean(x, na.rm = TRUE), x)) View(df2) |
| --- |

### Q3.

I plot the boxplots for uptake for each type and treatment category. I could not group the columns to find deeper plots.

| dfco=CO2 View(dfco) *#* library(ggplot2) library(gridExtra) *# boxplots using ggplot* p1=ggplot(dfco, aes(uptake,Type),)+geom\_boxplot(col='magenta')+  theme\_minimal()+  scale\_fill\_distiller() p2=ggplot(dfco, aes(uptake,Treatment))+geom\_boxplot(col='green')+  theme\_minimal()+  scale\_fill\_distiller() grid.arrange(p1,p2) |
| --- |

## 

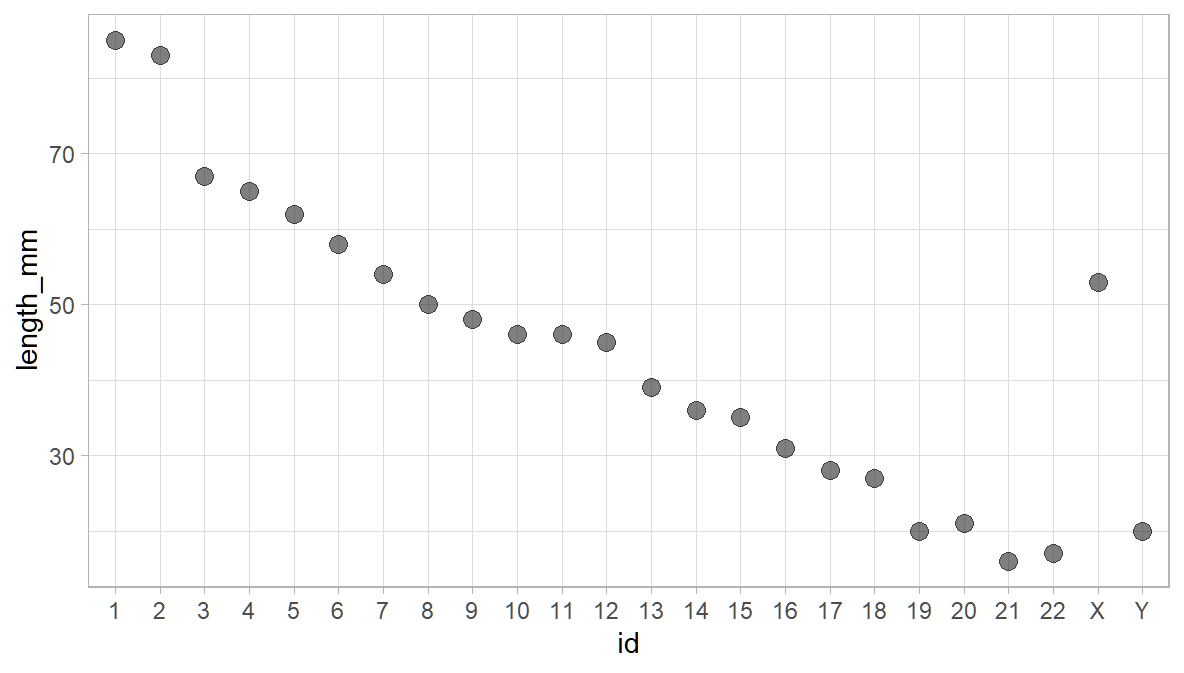
## TAsk 8

* A

| df=chromosome colnames(df) *# using dplr summarise function to calculate the stats. We select the columns by name.* *# combine mean, median and max as a list* summary\_stats2 <- df %>%  summarise(across(c(variations, protein\_codinggenes, mi\_rna), list(  mean = ~ mean(., na.rm = TRUE),  median = ~ median(., na.rm = TRUE),  max = ~ max(., na.rm = TRUE)  )))  print(summary\_stats2)  *# A tibble: 1 × 9*  variations\_mean variations\_median variations\_max protein\_codinggenes\_mean protein\_codinggenes\_me...¹ protein\_codinggenes\_...²  <dbl> <dbl> <dbl> <dbl> <dbl> <int> 1 6484572. 6172346 12945965 850. 836 2058 *# ℹ abbreviated names: ¹​protein\_codinggenes\_median, ²​protein\_codinggenes\_max* *# ℹ 3 more variables: mi\_rna\_mean <dbl>, mi\_rna\_median <dbl>, mi\_rna\_max <int>* |
| --- |

* B

| *# (not sure should be the relevant columns, plotting with id and length\_mm columns)* ggplot(df, aes(id,length\_mm)) +  geom\_point(alpha=0.50, size=3) +  theme\_light() |
| --- |



* C

| p1=ggplot(df, aes(length\_mm, protein\_codinggenes),color=protein\_codinggenes) +  geom\_point(alpha=0.50, size=3) +  geom\_smooth(method=lm, color='red') +  theme\_minimal() +  labs(title="Correlations",   x="Lenght mm", y = "protein\_codinggenes") p2=ggplot(df, aes(length\_mm, mi\_rna),color=mi\_rna) +  geom\_point(alpha=0.50, size=3) +  geom\_smooth(method=lm, color='blue') +  theme\_minimal() +  labs(title="Correlations",   x="Lenght mm", y = "mi rna") grid.arrange(p1,p2) |
| --- |

* D

| df=proteins View(df) names(df) summary\_stats3 <- df %>%  summarise(across(c(length, mass), list(  mean = ~ mean(., na.rm = TRUE),  median = ~ median(., na.rm = TRUE),  max = ~ max(., na.rm = TRUE)  ))) *#*  print(summary\_stats3) *# A tibble: 1 × 6*  length\_mean length\_median length\_max mass\_mean mass\_median mass\_max  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> 1 557. 414 34350 62061. 46140. 3816030   *# plotting - linear regression and histogram* p3=ggplot(df, aes(length, mass)) +  geom\_point(alpha=0.30, size=2, position = 'identity') +  geom\_smooth(method=lm,col='orange') +  theme\_minimal() +  scale\_fill\_brewer(palette="Dark2")+  labs(title="Regression",   x="Length", y = "Mass")  p4=ggplot(df, aes(length), fill(mass)) +  geom\_histogram(alpha=0.70, fill='orange') +  theme\_minimal() +  scale\_fill\_brewer(palette="Accent")+  labs(title="Histogram",   x="Length", y = "Mass") grid.arrange(p3,p4) |
| --- |

