R Notebook

This is an [R Markdown](http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

plot(cars)



Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.

#I am writing my code from here using the Greg Martin Video  
?mean  
#Using a ? sign infront of any object in R shows its usages  
#show. line code shows us the usage of Arithmetic Mean  
?ChickWeight  
#It is data frame in R shows us the diet on early growth of chciks

2+3

## [1] 5

#we can directly use R as a calculator  
  
a<-3  
b<-4  
#i am assigning values in a and b variables  
a+b

## [1] 7

sum(a,b)

## [1] 7

#it is inbuilt function in R that is sum add values with in the parameters

ages <- c(5,6)  
ages

## [1] 5 6

#We can use concatenation in R means joining values  
#We can use the c() function to concatenate elements into a vector.  
sum(ages)

## [1] 11

names <- c("ajay","jay")  
#now we are using characters to concatenate  
friends <- data.frame(names,ages)  
#it is a data frame of names and ages  
str(friends)

## 'data.frame': 2 obs. of 2 variables:  
## $ names: chr "ajay" "jay"  
## $ ages : num 5 6

#This is shows the structure of data frame   
friends$ages

## [1] 5 6

# $ stands for datasign to look particular varaible  
friends$names

## [1] "ajay" "jay"

friends[,1]

## [1] "ajay" "jay"

friends[1,]

## names ages  
## 1 ajay 5

friends[2,]

## names ages  
## 2 jay 6

data()  
#this shows the data sets in R  
#view(starwars) to view starwars data set  
  
data(starwars)

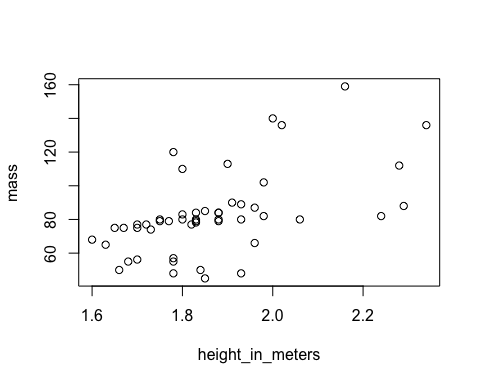
## Warning in data(starwars): data set 'starwars' not found

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.4 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

#install.packages("tidyverse") to install packages of tidyverse

#It is pipe operator it just pipe it in  
# rows are filtered where mass is less than 200 and height is greater than 150 and with a new column height\_in\_meters is generated by converting height from cm to meters  
#mutate means change the code changes the height   
#select shows the columns that which we want to work with   
#arrange will sort it  
starwars %>%   
 filter(height > 150 & mass <200) %>%  
 mutate(height\_in\_meters = height/100) %>%   
 select(height\_in\_meters,mass) %>%   
 arrange(mass) %>%   
 plot()



#View(msleep)  
glimpse(msleep)

## Rows: 83  
## Columns: 11  
## $ name <chr> "Cheetah", "Owl monkey", "Mountain beaver", "Greater shor…  
## $ genus <chr> "Acinonyx", "Aotus", "Aplodontia", "Blarina", "Bos", "Bra…  
## $ vore <chr> "carni", "omni", "herbi", "omni", "herbi", "herbi", "carn…  
## $ order <chr> "Carnivora", "Primates", "Rodentia", "Soricomorpha", "Art…  
## $ conservation <chr> "lc", NA, "nt", "lc", "domesticated", NA, "vu", NA, "dome…  
## $ sleep\_total <dbl> 12.1, 17.0, 14.4, 14.9, 4.0, 14.4, 8.7, 7.0, 10.1, 3.0, 5…  
## $ sleep\_rem <dbl> NA, 1.8, 2.4, 2.3, 0.7, 2.2, 1.4, NA, 2.9, NA, 0.6, 0.8, …  
## $ sleep\_cycle <dbl> NA, NA, NA, 0.1333333, 0.6666667, 0.7666667, 0.3833333, N…  
## $ awake <dbl> 11.9, 7.0, 9.6, 9.1, 20.0, 9.6, 15.3, 17.0, 13.9, 21.0, 1…  
## $ brainwt <dbl> NA, 0.01550, NA, 0.00029, 0.42300, NA, NA, NA, 0.07000, 0…  
## $ bodywt <dbl> 50.000, 0.480, 1.350, 0.019, 600.000, 3.850, 20.490, 0.04…

#give first few rows of data head  
head(msleep)

## # A tibble: 6 × 11  
## name genus vore order conservation sleep\_total sleep\_rem sleep\_cycle awake  
## <chr> <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 Cheetah Acin… carni Carn… lc 12.1 NA NA 11.9  
## 2 Owl mo… Aotus omni Prim… <NA> 17 1.8 NA 7   
## 3 Mounta… Aplo… herbi Rode… nt 14.4 2.4 NA 9.6  
## 4 Greate… Blar… omni Sori… lc 14.9 2.3 0.133 9.1  
## 5 Cow Bos herbi Arti… domesticated 4 0.7 0.667 20   
## 6 Three-… Brad… herbi Pilo… <NA> 14.4 2.2 0.767 9.6  
## # ℹ 2 more variables: brainwt <dbl>, bodywt <dbl>

class(msleep$name)

## [1] "character"

#give the type of variable in the table   
length(msleep)

## [1] 11

length(msleep$name)

## [1] 83

#give the total values in the data set  
names(msleep)

## [1] "name" "genus" "vore" "order" "conservation"  
## [6] "sleep\_total" "sleep\_rem" "sleep\_cycle" "awake" "brainwt"   
## [11] "bodywt"

#give the all names of the in that dataset  
unique(msleep$vore)

## [1] "carni" "omni" "herbi" NA "insecti"

#gives only unique names in particular row  
missing <- !complete.cases(msleep)  
#complete cases shows the all data that isnot missing  
msleep[missing,]

## # A tibble: 63 × 11  
## name genus vore order conservation sleep\_total sleep\_rem sleep\_cycle awake  
## <chr> <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl>  
## 1 Cheet… Acin… carni Carn… lc 12.1 NA NA 11.9  
## 2 Owl m… Aotus omni Prim… <NA> 17 1.8 NA 7   
## 3 Mount… Aplo… herbi Rode… nt 14.4 2.4 NA 9.6  
## 4 Three… Brad… herbi Pilo… <NA> 14.4 2.2 0.767 9.6  
## 5 North… Call… carni Carn… vu 8.7 1.4 0.383 15.3  
## 6 Vespe… Calo… <NA> Rode… <NA> 7 NA NA 17   
## 7 Roe d… Capr… herbi Arti… lc 3 NA NA 21   
## 8 Goat Capri herbi Arti… lc 5.3 0.6 NA 18.7  
## 9 Grivet Cerc… omni Prim… lc 10 0.7 NA 14   
## 10 Star-… Cond… omni Sori… lc 10.3 2.2 NA 13.7  
## # ℹ 53 more rows  
## # ℹ 2 more variables: brainwt <dbl>, bodywt <dbl>

starwars %>%  
 select(name,height,mass)

## # A tibble: 87 × 3  
## name height mass  
## <chr> <int> <dbl>  
## 1 Luke Skywalker 172 77  
## 2 C-3PO 167 75  
## 3 R2-D2 96 32  
## 4 Darth Vader 202 136  
## 5 Leia Organa 150 49  
## 6 Owen Lars 178 120  
## 7 Beru Whitesun Lars 165 75  
## 8 R5-D4 97 32  
## 9 Biggs Darklighter 183 84  
## 10 Obi-Wan Kenobi 182 77  
## # ℹ 77 more rows

starwars %>%   
 select(ends\_with("color"))

## # A tibble: 87 × 3  
## hair\_color skin\_color eye\_color  
## <chr> <chr> <chr>   
## 1 blond fair blue   
## 2 <NA> gold yellow   
## 3 <NA> white, blue red   
## 4 none white yellow   
## 5 brown light brown   
## 6 brown, grey light blue   
## 7 brown light blue   
## 8 <NA> white, red red   
## 9 black light brown   
## 10 auburn, white fair blue-gray  
## # ℹ 77 more rows

starwars %>%   
 select(name, mass,everything())

## # A tibble: 87 × 14  
## name mass height hair\_color skin\_color eye\_color birth\_year sex gender  
## <chr> <dbl> <int> <chr> <chr> <chr> <dbl> <chr> <chr>   
## 1 Luke Sk… 77 172 blond fair blue 19 male mascu…  
## 2 C-3PO 75 167 <NA> gold yellow 112 none mascu…  
## 3 R2-D2 32 96 <NA> white, bl… red 33 none mascu…  
## 4 Darth V… 136 202 none white yellow 41.9 male mascu…  
## 5 Leia Or… 49 150 brown light brown 19 fema… femin…  
## 6 Owen La… 120 178 brown, gr… light blue 52 male mascu…  
## 7 Beru Wh… 75 165 brown light blue 47 fema… femin…  
## 8 R5-D4 32 97 <NA> white, red red NA none mascu…  
## 9 Biggs D… 84 183 black light brown 24 male mascu…  
## 10 Obi-Wan… 77 182 auburn, w… fair blue-gray 57 male mascu…  
## # ℹ 77 more rows  
## # ℹ 5 more variables: homeworld <chr>, species <chr>, films <list>,  
## # vehicles <list>, starships <list>

#shows the varaibles who names ends with color  
starwars %>%  
 rename("chaarcter" = "name") %>%   
 head()

## # A tibble: 6 × 14  
## chaarcter height mass hair\_color skin\_color eye\_color birth\_year sex gender  
## <chr> <int> <dbl> <chr> <chr> <chr> <dbl> <chr> <chr>   
## 1 Luke Sky… 172 77 blond fair blue 19 male mascu…  
## 2 C-3PO 167 75 <NA> gold yellow 112 none mascu…  
## 3 R2-D2 96 32 <NA> white, bl… red 33 none mascu…  
## 4 Darth Va… 202 136 none white yellow 41.9 male mascu…  
## 5 Leia Org… 150 49 brown light brown 19 fema… femin…  
## 6 Owen Lars 178 120 brown, gr… light blue 52 male mascu…  
## # ℹ 5 more variables: homeworld <chr>, species <chr>, films <list>,  
## # vehicles <list>, starships <list>

# to rename the names of varaible  
class(starwars$mass)

## [1] "numeric"

starwars$mass <- as.character(starwars$mass)  
class(starwars$mass)

## [1] "character"

#to change the type of variable we use this as  
starwars %>%   
 mutate(mass = as.numeric(mass))

## # A tibble: 87 × 14  
## name height mass hair\_color skin\_color eye\_color birth\_year sex gender  
## <chr> <int> <dbl> <chr> <chr> <chr> <dbl> <chr> <chr>   
## 1 Luke Sk… 172 77 blond fair blue 19 male mascu…  
## 2 C-3PO 167 75 <NA> gold yellow 112 none mascu…  
## 3 R2-D2 96 32 <NA> white, bl… red 33 none mascu…  
## 4 Darth V… 202 136 none white yellow 41.9 male mascu…  
## 5 Leia Or… 150 49 brown light brown 19 fema… femin…  
## 6 Owen La… 178 120 brown, gr… light blue 52 male mascu…  
## 7 Beru Wh… 165 75 brown light blue 47 fema… femin…  
## 8 R5-D4 97 32 <NA> white, red red NA none mascu…  
## 9 Biggs D… 183 84 black light brown 24 male mascu…  
## 10 Obi-Wan… 182 77 auburn, w… fair blue-gray 57 male mascu…  
## # ℹ 77 more rows  
## # ℹ 5 more variables: homeworld <chr>, species <chr>, films <list>,  
## # vehicles <list>, starships <list>

starwars$mass <- as.numeric(starwars$mass)  
class(starwars$mass)

## [1] "numeric"

df <- starwars  
#df stands for sta frame we are creating a new frame  
#View(df)  
df$sex <- as.factor(df$sex)  
class(df$sex)

## [1] "factor"

levels(df$sex)

## [1] "female" "hermaphroditic" "male" "none"

#gives the types of data presesnt  
  
df <-df %>%  
 mutate(sex = factor(sex,levels = c("male","female", "none","hermaphroditic")))  
levels(df$sex)

## [1] "male" "female" "none" "hermaphroditic"

#to change the order of data  
starwars %>%   
 select(mass,sex) %>%   
 filter(mass <55 & sex == "male")

## # A tibble: 6 × 2  
## mass sex   
## <dbl> <chr>  
## 1 17 male   
## 2 20 male   
## 3 40 male   
## 4 15 male   
## 5 45 male   
## 6 48 male

# to filter certain data as required  
starwars %>%   
 select(sex) %>%  
 mutate(sex = recode(sex, "male" = "man" , "female" = "woman"))

## # A tibble: 87 × 1  
## sex   
## <chr>  
## 1 man   
## 2 none   
## 3 none   
## 4 man   
## 5 woman  
## 6 man   
## 7 woman  
## 8 none   
## 9 man   
## 10 man   
## # ℹ 77 more rows

#recode function changes the data as we supposed

mean(df$height)

## [1] NA

#mean is function we find out the mean of the data   
#mean function does'nt know hoe to deal with missing data  
mean(starwars$height , na.rm= TRUE)

## [1] 174.6049

#na.rm means it removes the na values from the data frame

names <- c("ajay", "harika" , "jay" , "hari", "ajay")  
age <- c(20,22,21,23,20)  
friends <- data.frame(names,age)  
friends %>%   
 distinct()

## names age  
## 1 ajay 20  
## 2 harika 22  
## 3 jay 21  
## 4 hari 23

#distinct function used to remove the duplicates in the data frame

starwars %>%   
 mutate(height\_m = height/100) %>%  
#above code will create a new varaible iwth height >100  
 select(name, height, height\_m)

## # A tibble: 87 × 3  
## name height height\_m  
## <chr> <int> <dbl>  
## 1 Luke Skywalker 172 1.72  
## 2 C-3PO 167 1.67  
## 3 R2-D2 96 0.96  
## 4 Darth Vader 202 2.02  
## 5 Leia Organa 150 1.5   
## 6 Owen Lars 178 1.78  
## 7 Beru Whitesun Lars 165 1.65  
## 8 R5-D4 97 0.97  
## 9 Biggs Darklighter 183 1.83  
## 10 Obi-Wan Kenobi 182 1.82  
## # ℹ 77 more rows

#shows the tables in the slect feild  
starwars %>%   
 mutate(height\_m = height/100) %>%  
 select(name,height, height\_m) %>%  
 mutate(tallness =  
 if\_else(height\_m <1,  
 "short",  
 "tall"))

## # A tibble: 87 × 4  
## name height height\_m tallness  
## <chr> <int> <dbl> <chr>   
## 1 Luke Skywalker 172 1.72 tall   
## 2 C-3PO 167 1.67 tall   
## 3 R2-D2 96 0.96 short   
## 4 Darth Vader 202 2.02 tall   
## 5 Leia Organa 150 1.5 tall   
## 6 Owen Lars 178 1.78 tall   
## 7 Beru Whitesun Lars 165 1.65 tall   
## 8 R5-D4 97 0.97 short   
## 9 Biggs Darklighter 183 1.83 tall   
## 10 Obi-Wan Kenobi 182 1.82 tall   
## # ℹ 77 more rows

#tallness is new vaaraible where the result of if else stored  
#if condition is true the short is if ti false the tall is assigned

library(gapminder)  
#View(gapminder)  
#it is also a inbulit data set   
data <- select(gapminder, country, year, lifeExp)  
#we just created a new data frame with only three rows of data in it  
wide\_data <- data %>%  
 pivot\_wider(names\_from = year , values\_from = lifeExp)  
  
#View(wide\_data)  
#pivot wider function allows us to view table in different fromat   
#above code used year and name and life Exp as values  
#table now shows countiers lifeExp in certain years   
#lets change the same data using pivot longer

longer\_data <- wide\_data %>%  
 pivot\_longer(2:13,  
 names\_to = "year",  
 values\_to = "lifeExp")  
#view(longer\_data)  
#all data in the names feild now moved to a new dat feild called as year   
# all the data in values now to moved lifeExp as earlier  
#

min(msleep$awake)

## [1] 4.1

#mininum of all numeric values presentin that awake data feild  
max(msleep$awake)

## [1] 22.1

#this shows the maximum  
IQR(msleep$awake)

## [1] 5.9

#this is interquartile range  
range(msleep$awake)

## [1] 4.1 22.1

#this is range of the data it between mi and max   
#centrality  
mean(msleep$awake)

## [1] 13.56747

median(msleep$awake)

## [1] 13.9

#varaince  
var(msleep$awake)

## [1] 19.82106

#this is the varaince of the data  
summary(msleep$awake)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 4.10 10.25 13.90 13.57 16.15 22.10

#this is gives everything about the data  
msleep %>%   
 select(awake,sleep\_total) %>%   
 summary()

## awake sleep\_total   
## Min. : 4.10 Min. : 1.90   
## 1st Qu.:10.25 1st Qu.: 7.85   
## Median :13.90 Median :10.10   
## Mean :13.57 Mean :10.43   
## 3rd Qu.:16.15 3rd Qu.:13.75   
## Max. :22.10 Max. :19.90

#we can two or three rows at a time

table(msleep$vore)

##   
## carni herbi insecti omni   
## 19 32 5 20

#create a table for certian field of data with count under it  
msleep %>%   
 select(vore,order) %>%   
 filter(order %in% c("Rodentia" , "Primates")) %>%   
 table()

## order  
## vore Primates Rodentia  
## carni 1 1  
## herbi 1 16  
## omni 10 2

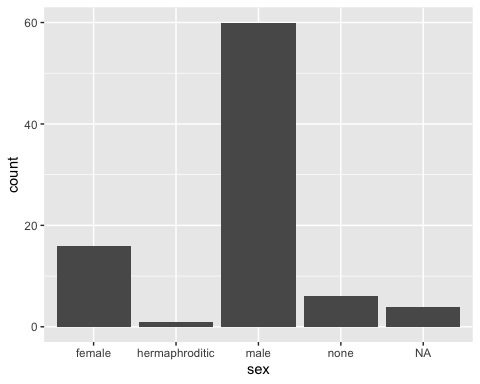
#this is filter basically checks whether anyone of these concatenation are in that order   
#table gives the output in fomrat of table

plot(pressure)



#this plot the data of mercury under pressure and temperature

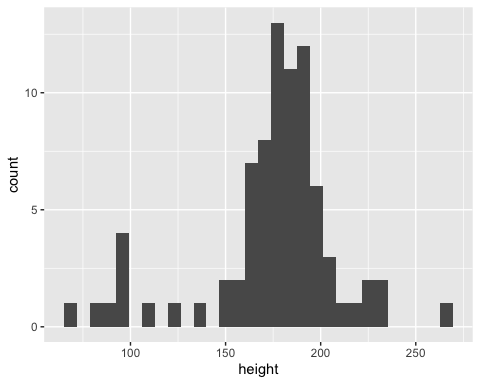
ggplot(data=starwars,  
 mapping= aes(x= sex))+  
 geom\_bar()



#This is grammar of graph plot   
#I kept x axis as gender   
#geom\_bar give bar graph

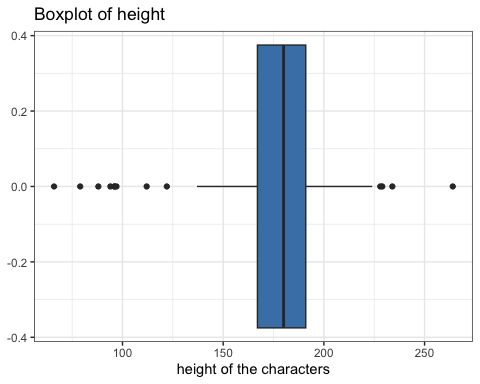
starwars %>%  
 drop\_na(height) %>%   
#drop\_na removes null values  
 ggplot(mapping=aes(x=height))+  
 geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#we are mappinga histogram geom\_histo  
#by this way we can manipluate dat before ploting into ggplot

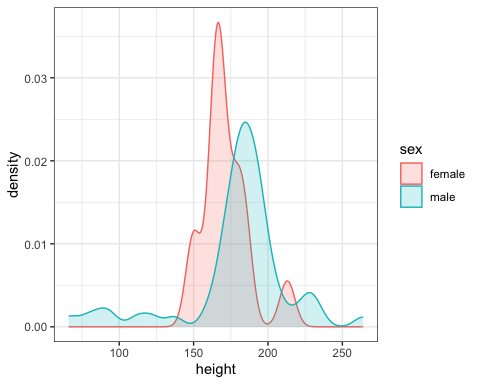
starwars %>%   
 drop\_na(height) %>%   
 ggplot(mapping=aes(x=height))+  
#height is numberic varaible is can represented with either with a histogram or a box plot  
#we ca use density plot even   
 geom\_boxplot(fill = "steelblue")+  
#above line of the code dill color to the plot  
 theme\_bw()+  
#we can add themes to R the basic one is bw(blakc and white)  
 labs(title= "Boxplot of height", x="height of the characters")



#labs for labels the x axis and the title of the plot

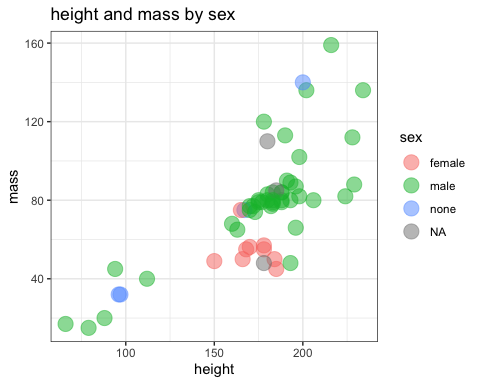
starwars %>%   
 filter(sex %in% c("male", "female")) %>%  
#filter the only male amd females   
 ggplot(mapping=aes(x=height,  
 color=sex,  
 fill=sex))+  
#above plot add color to different sex  
 geom\_density(alpha= 0.2)+  
#alpha value depends how dark we need the coloring  
#density plot same as histogram  
 theme\_bw()

## Warning: Removed 5 rows containing non-finite outside the scale range  
## (`stat\_density()`).



#this is theme as above

starwars %>%   
 filter(mass<200) %>%   
#filter the mass by. 200   
 ggplot(aes(x=height,y=mass,color=sex))+  
#above code we are using both x and y axis  
#color is again assigned to sex   
 geom\_point(size=5, alpha=0.5)+  
#geometry id scatter plot that point and is size 5  
 theme\_bw()+  
 labs(title= "height and mass by sex")



starwars %>%   
 filter (mass<200) %>%   
 ggplot(aes(height,mass, color=sex))+  
 geom\_point(size=3 , alpha=0.8)+  
 geom\_smooth()+  
#geom-smooth add a linear layer model above the point  
 facet\_wrap(~sex)+  
#this wrap make sex feild into different feild easy to visulize a pertical type  
 theme\_bw()+  
 labs(title="height and mass by sex")

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : span too small. fewer data values than degrees of freedom.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : pseudoinverse used at 95.48

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : neighborhood radius 71.52

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : There are other near singularities as well. 10716

## Warning in sqrt(sum.squares/one.delta): NaNs produced

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : span too small. fewer  
## data values than degrees of freedom.

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : pseudoinverse used at  
## 95.48

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : neighborhood radius  
## 71.52

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : reciprocal condition  
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : There are other near  
## singularities as well. 10716

## Warning in stats::qt(level/2 + 0.5, pred$df): NaNs produced

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : span too small. fewer data values than degrees of freedom.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : pseudoinverse used at 177.97

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : neighborhood radius 2.035

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,  
## : There are other near singularities as well. 25.351

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : span too small. fewer  
## data values than degrees of freedom.

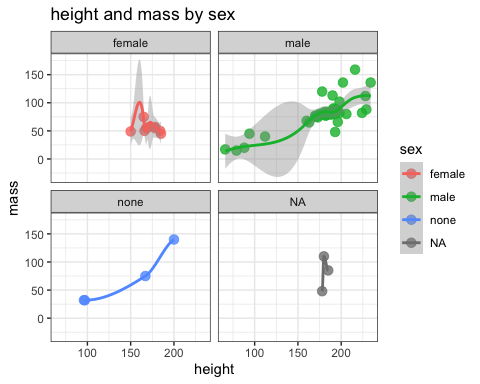
## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : pseudoinverse used at  
## 177.97

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : neighborhood radius  
## 2.035

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : reciprocal condition  
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x  
## else if (is.data.frame(newdata))  
## as.matrix(model.frame(delete.response(terms(object)), : There are other near  
## singularities as well. 25.351

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning  
## -Inf  
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning  
## -Inf



#theme and lable and everthing is same

gapminder %>%  
 filter(continent %in% c("Africa", "Europe")) %>%  
#filter the continent from the table  
 { t.test(lifeExp ~ continent, data = .,   
#independent t-test to compare life expectancy between the two continents  
 alternative = "two.sided") }

##   
## Welch Two Sample t-test  
##   
## data: lifeExp by continent  
## t = -49.551, df = 981.2, p-value < 2.2e-16  
## alternative hypothesis: true difference in means between group Africa and group Europe is not equal to 0  
## 95 percent confidence interval:  
## -23.95076 -22.12595  
## sample estimates:  
## mean in group Africa mean in group Europe   
## 48.86533 71.90369

# Test if means are different (not one-sided)

#ANOVA  
gapminder %>%  
 filter(year == 2007) %>%  
# only rows where the year is 2007  
 filter(continent %in% c("Africa", "Europe")) %>%  
#only Africa and Europe  
 aov(lifeExp~ continent , data = .) %>%   
#Perform an ANOVA test to compare life expectancy between the two continents  
 summary()

## Df Sum Sq Mean Sq F value Pr(>F)   
## continent 1 9927 9927 159.2 <2e-16 \*\*\*  
## Residuals 80 4988 62   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

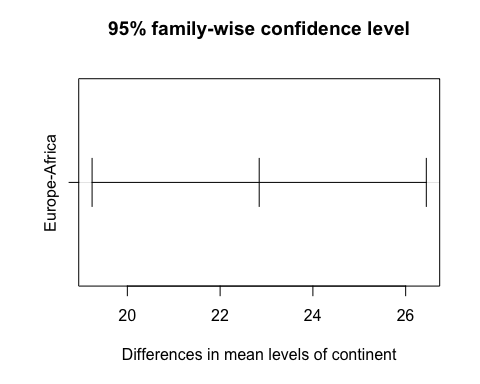
#in the form summary

gapminder %>%  
 filter(year == 2007) %>%   
 filter(continent %in% c("Africa", "Europe")) %>%  
 aov(lifeExp~ continent , data = .) %>%   
 TukeyHSD()

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = lifeExp ~ continent, data = .)  
##   
## $continent  
## diff lwr upr p adj  
## Europe-Africa 22.84256 19.2399 26.44522 0

#it the same the above code but its piped into TukeyHSD  
#a

gapminder %>%  
 filter(year == 2007) %>%   
 filter(continent %in% c("Africa", "Europe")) %>%  
 aov(lifeExp~ continent , data = .) %>%   
 TukeyHSD() %>%   
 plot()



#it the same the above code but its ploted

head(iris)

## Sepal.Length Sepal.Width Petal.Length Petal.Width Species  
## 1 5.1 3.5 1.4 0.2 setosa  
## 2 4.9 3.0 1.4 0.2 setosa  
## 3 4.7 3.2 1.3 0.2 setosa  
## 4 4.6 3.1 1.5 0.2 setosa  
## 5 5.0 3.6 1.4 0.2 setosa  
## 6 5.4 3.9 1.7 0.4 setosa

#Chi-Squared  
flowers <- iris %>%  
#Create a new dataset from the built-in 'iris'   
 mutate(size = cut(Sepal.Length,  
#variable size based on Sepal.Length  
 breaks = 3,  
 labels = c("small", "Medium", "Large"))) %>%  
 select(Species, size)  
 #Select only the Species and newly created size columns  
  
# chi squared goodness of fit test  
flowers %>%  
 select(size) %>%  
 table() %>%  
#frequency table for size  
 chisq.test()

##   
## Chi-squared test for given probabilities  
##   
## data: .  
## X-squared = 28.44, df = 2, p-value = 6.673e-07

#above line perform a Chi-Square test

# chi square test of independence  
flowers %>%  
 table() %>%  
 chisq.test()

##   
## Pearson's Chi-squared test  
##   
## data: .  
## X-squared = 111.63, df = 4, p-value < 2.2e-16

#linear model  
head(cars,10)

## speed dist  
## 1 4 2  
## 2 4 10  
## 3 7 4  
## 4 7 22  
## 5 8 16  
## 6 9 10  
## 7 10 18  
## 8 10 26  
## 9 10 34  
## 10 11 17

cars %>%  
 lm(dist ~ speed, data = .) %>%  
#linear regression model using dist based on speed  
 summary()

##   
## Call:  
## lm(formula = dist ~ speed, data = .)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -29.069 -9.525 -2.272 9.215 43.201   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -17.5791 6.7584 -2.601 0.0123 \*   
## speed 3.9324 0.4155 9.464 1.49e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 15.38 on 48 degrees of freedom  
## Multiple R-squared: 0.6511, Adjusted R-squared: 0.6438   
## F-statistic: 89.57 on 1 and 48 DF, p-value: 1.49e-12

#displays summary