GLM Explanation with R

Mpelias Michael

23 February 2016

Contents

, , ,

1 Introduction

This lesson is written for explaining the GLM analysis in R - Code. The main goals are:

- Understand through plots when to use other linear models
- Understand the basic methodology in Generalized Linear Models
- Prediction using GLM analyses
- Learn the use of R for:
  - Linear Models (Generalized or not)
  - Plots using ggplot2 package
  -

For this lesson we will use several libraries, such as:

- ggplot2 for plotting (with GGally , gridExtra extensions)
- MASS with numerous Datasets and functions
-
-
-

The libraries needed should be installed. Please Run the Code below.

```
###Function for checking if pkg is installed
is_installed <- function(mypkg) is.element(mypkg, installed.packages()[,1])

####Packages needed ,"alr3","relimp"
package_names= c("MASS","corrplot","fortunes","mfp","ggplot2","GGally",
                 "gridExtra","leaps","elasticnet","caret","knitr","gridExtra","RVAideMemoire","car","vc
###Package installer and installation
for(package_name in package_names)
{
  if(!is_installed(package_name))
  {
    install.packages(package_name)
  }
  library(package_name,character.only=TRUE,quietly=TRUE,verbose=FALSE)
  cat("Package :",package_name," Acquired","\n")

}

rm(package_name,package_names,is_installed)
search()
```

2 Statistical Analysis

What is statistical analysis? It's the science of collecting, exploring and presenting large amounts of data to discover underlying patterns and trends. Statistics are applied every day - in research, industry and government - to become more scientific about decisions that need to be made[@greycite38599]. A conclusion is easily driven, we need to know the data and what is that question we want to make. In the table below is the very first basic Statistical [@Delorme A.] .

Goal	Dataset		
	Binomial or Discrete	**Continuous measurement (from a normal distribution)**	**Continuous measurement, Rank, or Score (from non-normal distribution)**
Example of data sample	List of patients recovering or not after a treatment	Readings of heart pressure from several patients	Ranking of several treatment efficiency by one expert
Describe one data sample	Proportions	Mean, SD	Median
Compare one data sample to a hypothetical distribution	χ^2 or binomial test	One-sample t test	Sign test or Wilcoxon test
Compare two paired samples	Sign test	Paired t test	Sign test or Wilcoxon test
Compare two unpaired samples	χ^2 square Fisher's exact test	Unpaired t test	Mann-Whitney test
Compare three or more unmatched samples	χ^2 test	One-way ANOVA	Kruskal-Wallis test
Compare three or more matched samples	Cochrane Q test	Repeated-measures ANOVA	Friedman test
Quantify association between two paired samples	Contingency coefficients	Pearson correlation	Spearman correlation

3 Descriptive Statistic

The very first thing we can do is check our Data. The variables can be either Categorical (Ordinal or Nominal) or Continuous.

And the first question to be made is to describe the Data : * Are they continuous or Categorical? * Are they Normally Distributed or not? * What Proportion exist for the Categorical?

Our Data set is `diamonds` and contains the prices and other attributes of almost 54,000 diamonds. Diamonds data frame has 53940 rows and 10 variables:

- price: price in US dollars
- carat: weight of the diamond
- cut: quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- colour: diamond colour, from J (worst) to D (best)
- clarity: a measurement of how clear the diamond is (I1 (worst), SI1, SI2, VS1, VS2, VVS1, VVS2, IF (best))
- x: length in mm (0-10.74)
- y: width in mm (0-58.9)

- z: depth in mm (0-31.8)
- depth: total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43-79)
- table: width of top of diamond relative to widest point (43-79)

For more information type `??diamonds`

The first 10 observations are:

carat	cut	color	clarity	depth	table	price	x	y	z
0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
0.29	Premium	I	VS2	62.4	58	334	4.20	4.23	2.63
0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48
0.24	Very Good	I	VVS1	62.3	57	336	3.95	3.98	2.47
0.26	Very Good	H	SI1	61.9	55	337	4.07	4.11	2.53
0.22	Fair	E	VS2	65.1	61	337	3.87	3.78	2.49
0.23	Very Good	H	VS1	59.4	61	338	4.00	4.05	2.39

The Continuous Variables seem to be:

- carat
- depth
- table
- price
- x
- y
- z

The statistics that describes for the Normally Distributed Variables are for :

- location measures (mean)
- measures of variability (SD)

while for Non -Normally Distributed Variables:

- location measures (median)
- measures of variability (quartiles, Min, Max)

The normality test we use is the Shapiro- Wilk for this session. The test has a limitation of 5.000 length Vector to check if is Normally Distributed so we pick a sample sized 5.000.

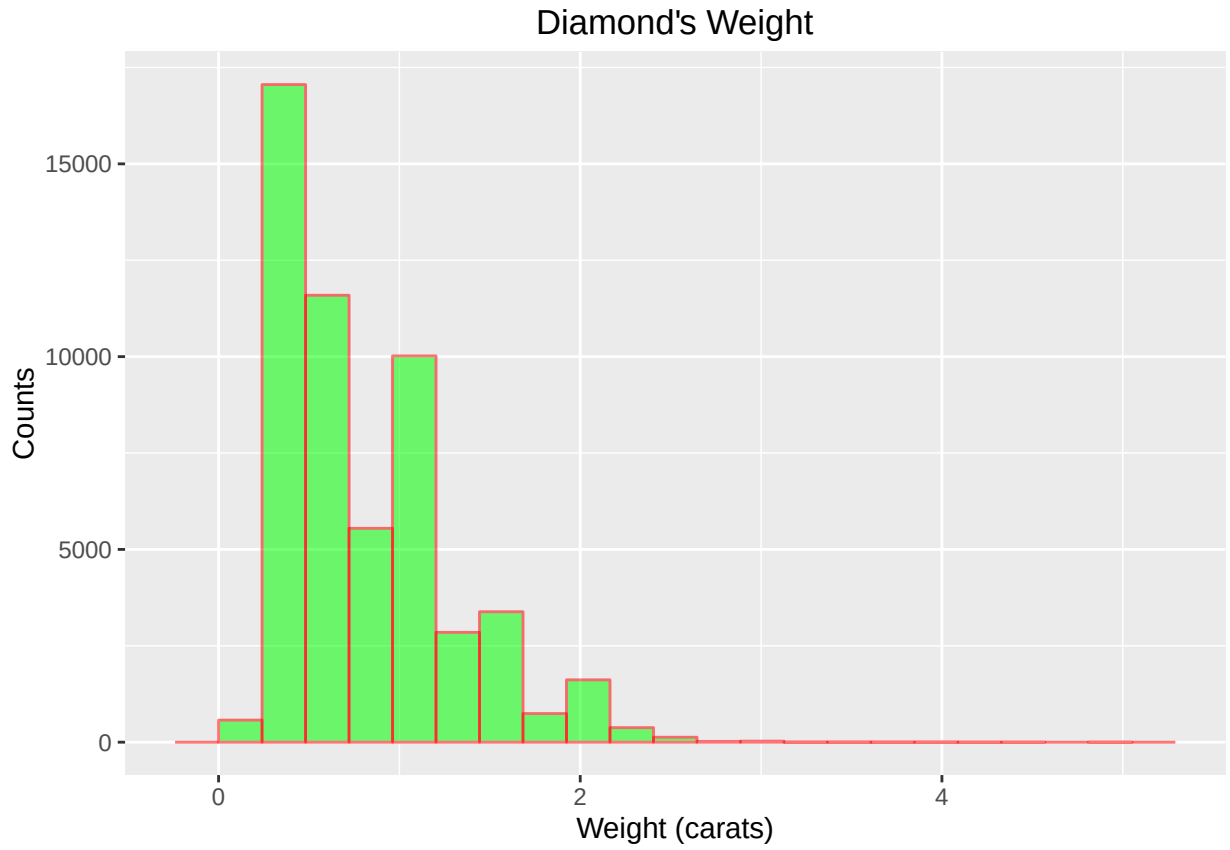
The first variable is carat, which counts the weight of the diamond.

```
shapiro.test(sample(diamonds[,1],size = 5000))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample(diamonds[, 1], size = 5000)
## W = 0.89669, p-value < 2.2e-16
```

As expected it is not normally distributed which can be shown in the plot below:

```
ggplot(data = diamonds,aes(x = carat)) + geom_histogram(bins = 20,colour= "red",
                                                        fill="green",alpha=0.55) +
  labs(title="Diamond's Weight",   x = "Weight (carats)", y= "Counts")
```



So the descriptive statistics are :

```
summary(diamonds[,1])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.2000  0.4000  0.7000  0.7979  1.0400  5.0100
```

Next we shall check the `depth` Variable.

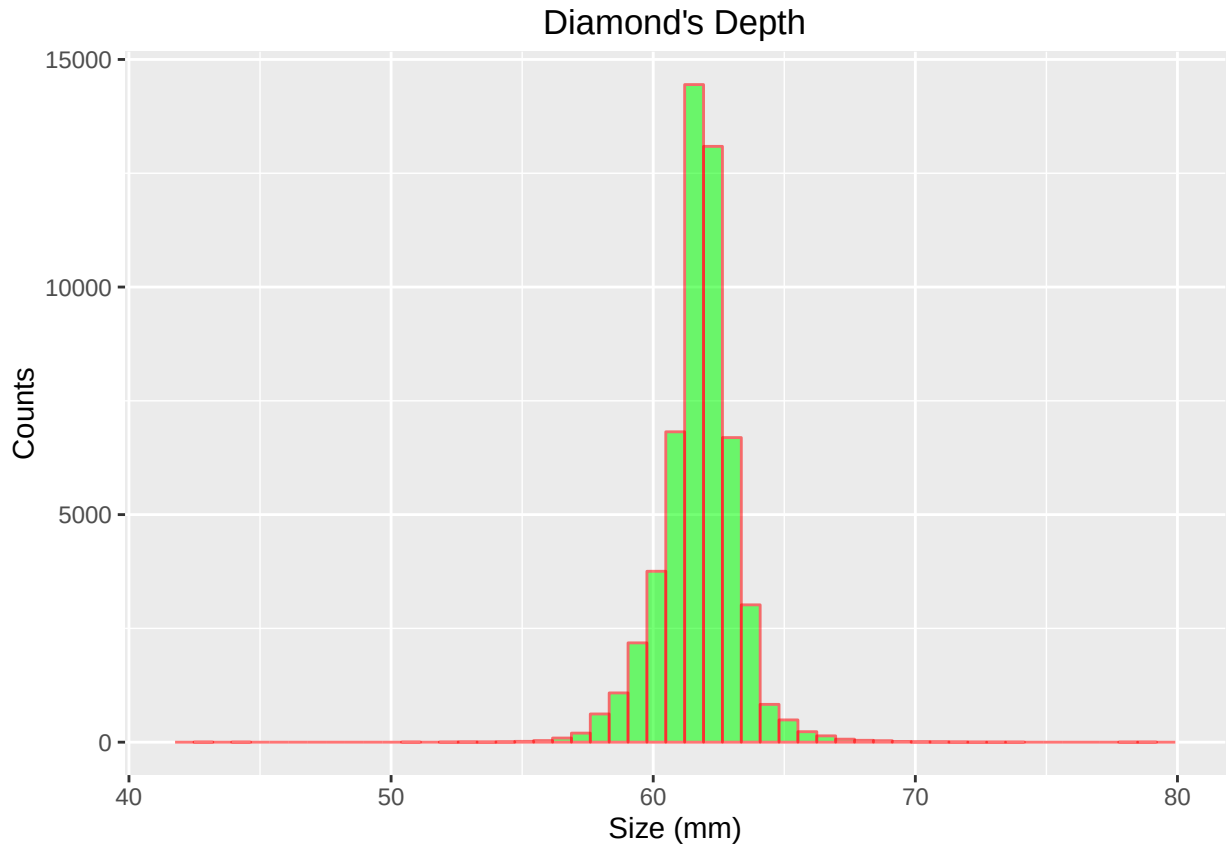
The Shapiro Wilk seems to reject the H_0 : The depth variable is normally distributed

```
shapiro.test(sample(diamonds$depth,size = 5000))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  sample(diamonds$depth, size = 5000)
## W = 0.96364, p-value < 2.2e-16
```

But Graphically that is not true. The plot seems to be very Normal.

```
ggplot(data = diamonds,aes(x = diamonds$depth)) + geom_histogram(bins = 50,colour= "red",
                                                                 fill="green",alpha=0.55) +
  labs(title="Diamond's Depth",    x = "Size (mm)", y= "Counts")
```



So the descriptive statistics are:

```
mean(diamonds$depth)
```

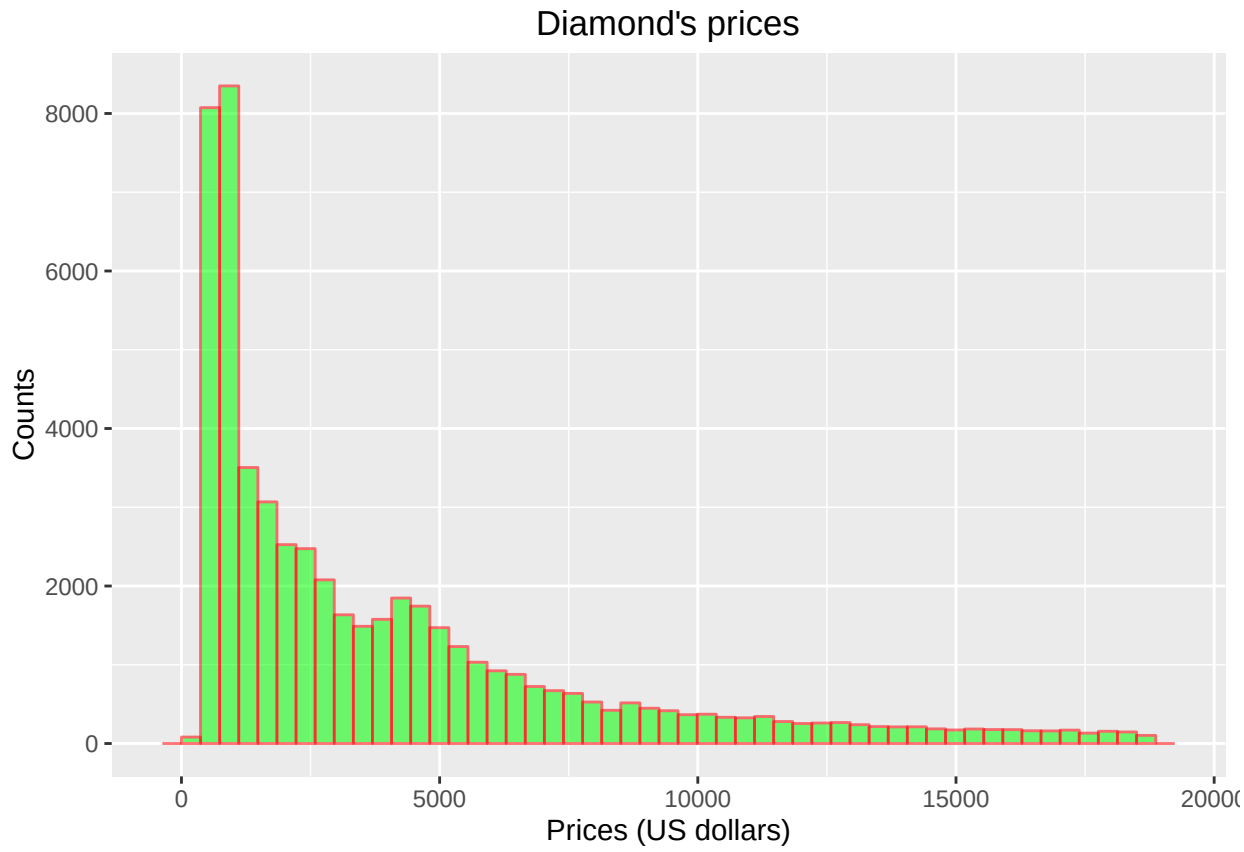
```
## [1] 61.7494
```

```
sd(diamonds$depth)
```

```
## [1] 1.432621
```

The price's plot does not resemble the Normal Distribution too.

```
ggplot(data = diamonds,aes(x = diamonds$price)) + geom_histogram(bins = 50,colour= "red",
                                                                 fill="green",alpha=0.55) +
  labs(title="Diamond's prices",    x = "Prices (US dollars)", y= "Counts")
```



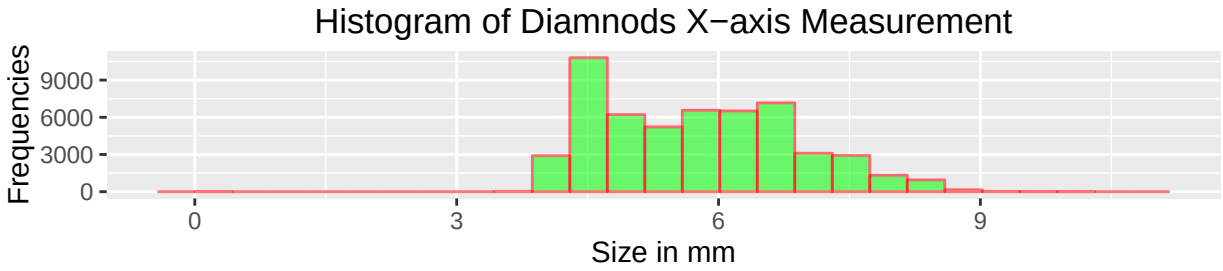
So the descriptive statistics are :

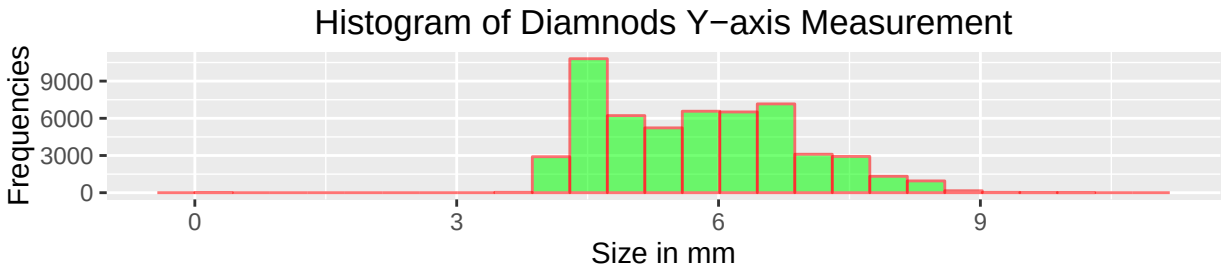
```
summary(diamonds$price)
```

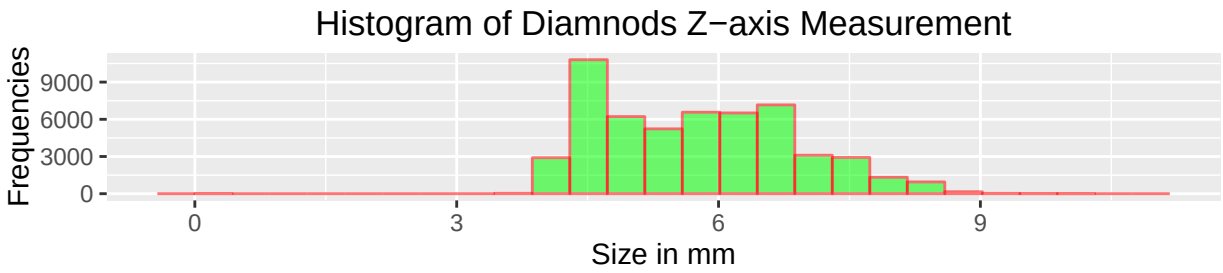
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     326     950    2401    3933    5324   18820
```

The x, Y, Z plots does not resemble the Normal Distribution too.

```
plot_x=ggplot(data = diamonds,aes(x = diamonds$x)) +
  geom_histogram(bins = 25,colour= "red",fill="green",alpha=0.55) +labs(title="Histogram of Diamnods X-a
plot_y=ggplot(data = diamonds,aes(x = diamonds$x)) +
  geom_histogram(bins = 25,colour= "red",fill="green",alpha=0.55)  +labs(title="Histogram of Diamnods Y-
plot_z=ggplot(data = diamonds,aes(x = diamonds$x)) +
  geom_histogram(bins = 25,colour= "red",fill="green",alpha=0.55)  +labs(title="Histogram of Diamnods Z-
grid.arrange(plot_x,plot_y,plot_z)
```







So the descriptive statistics will be :

```
apply(diamonds[,8:10],2 , summary)
```

```
##                x      y      z
## Min.       0.000  0.000  0.000
## 1st Qu.    4.710  4.720  2.910
## Median     5.700  5.710  3.530
## Mean       5.731  5.735  3.539
## 3rd Qu.    6.540  6.540  4.040
## Max.      10.740 58.900 31.800
```

The Categorical Variables are:

- cut
- colour
- clarity

Second is the variable `cut` that shows the quality of the cut. It's a Categorical with Fair, Good, Very Good, Premium, Ideal

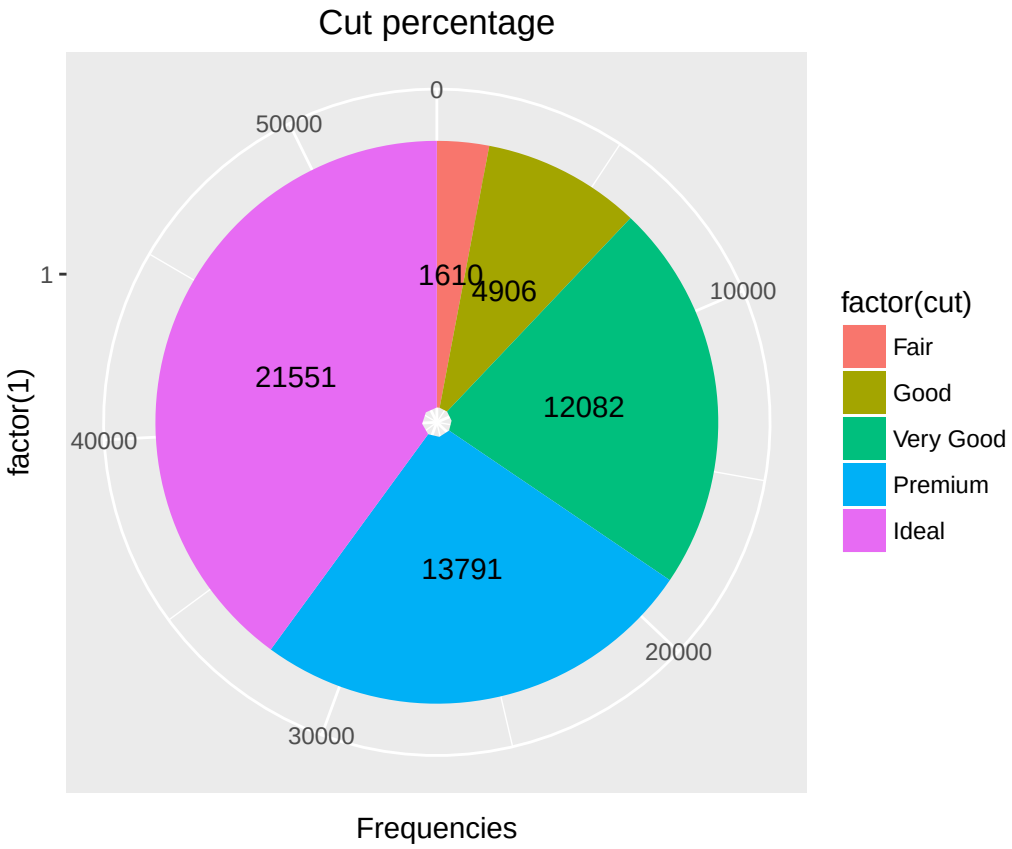
```
table(diamonds[,2])
```

```
##
##      Fair      Good Very Good   Premium     Ideal
##      1610      4906     12082     13791     21551
```

```
pie <- ggplot(data = diamonds,aes(x = factor(1), fill = factor(cut))) +
  geom_bar(position = "stack")

at <- as.numeric(cumsum(table(diamonds$cut))-0.5*table(diamonds$cut))


pie + coord_polar(theta = "y") + ylab("Frequencies") + labs(title = "Cut percentage")  +
  annotate(geom = "text", y = at , x = 1, label = summary(diamonds[,2]) )
```



Next is the colour of the diamond with 7 categories from D (the best) to J (the worst)

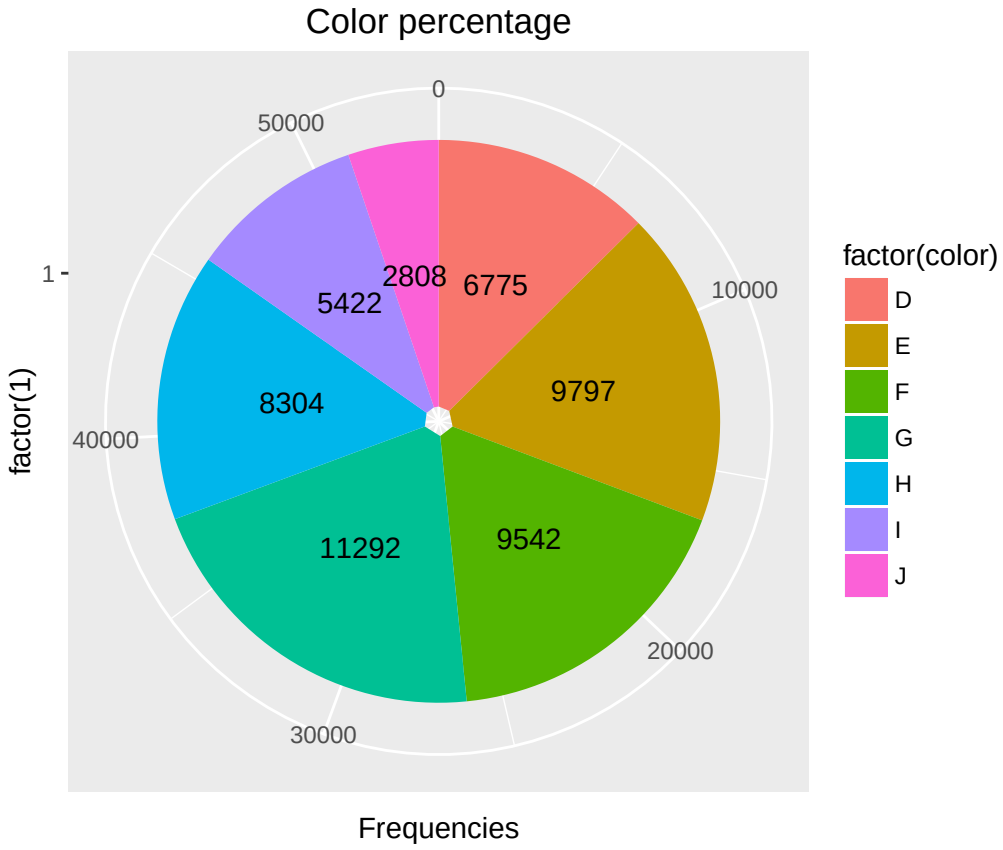
```
table(diamonds[,3])
```

```
##
##     D     E     F     G     H     I     J
##  6775  9797  9542 11292  8304  5422  2808
```

```
pie <- ggplot(data = diamonds,aes(x = factor(1), fill = factor(color))) +
  geom_bar(position = "stack")

at <- as.numeric(cumsum(table(diamonds$color))-0.5*table(diamonds$color))

pie + coord_polar(theta = "y") + ylab("Frequencies") + labs(title = "Color percentage")  +
  annotate(geom = "text", y = at , x = 1, label = summary(diamonds[,3]) )
```



Next is the clarity of the diamond with 8 categories.

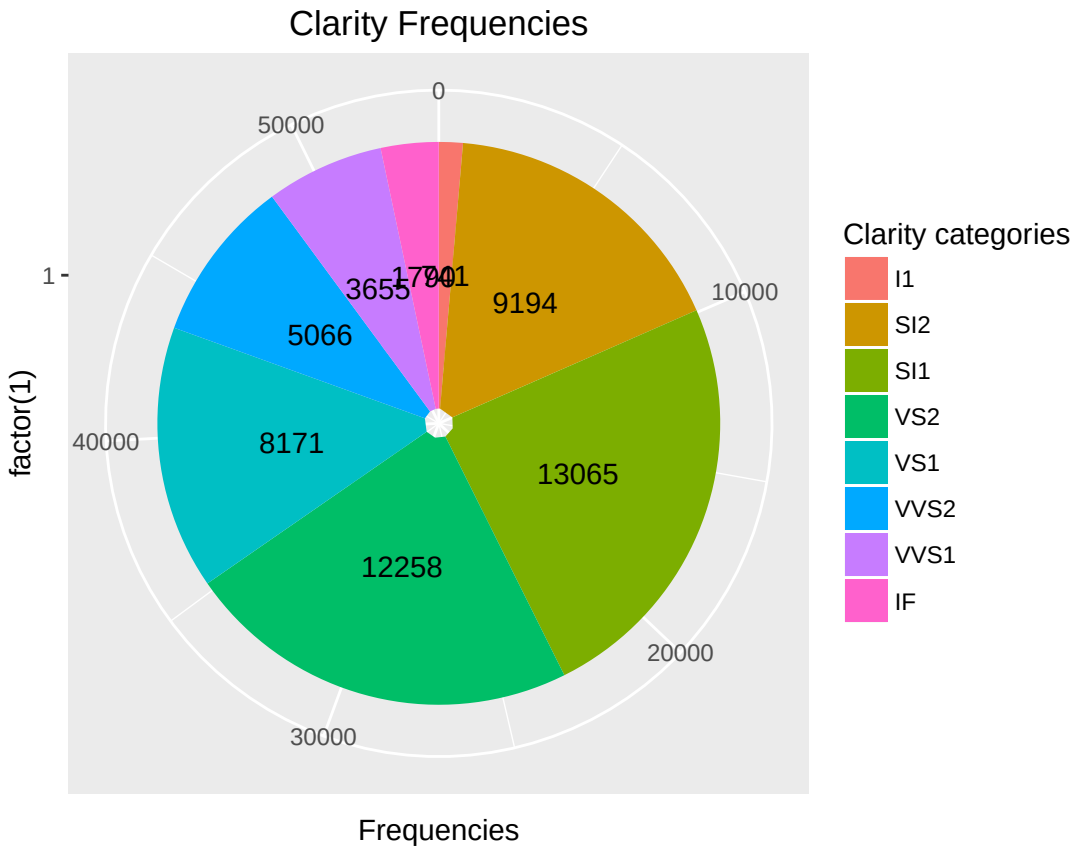
```
table(diamonds[,4])
```

```
##
##    I1   SI2   SI1   VS2   VS1  VVS2  VVS1    IF
##   741  9194 13065 12258  8171  5066  3655  1790
```

```
pie <- ggplot(data = diamonds,aes(x = factor(1), fill = factor(clarity))) +
  geom_bar(position = "stack")

at <- as.numeric(cumsum(table(diamonds$clarity))-0.5*table(diamonds$clarity))

pie + coord_polar(theta = "y") + ylab("Frequencies") + labs(title = "Clarity Frequencies")  +
  annotate(geom = "text", y = at , x = 1, label = summary(diamonds[,4]) ) + guides(fill=guide_legend("C
```



3.1 One Sample comparison to a Hypothetical Distribution.

3.1.1 One-Sample t-test , Wilcoxon , Binomial and X^2 test

The next question we need to make is whether our Variables fit a Hypothetical Distribution.

For the Categorical there are two tests the Binomial (and the X^2 test).

Suppose that we know that the percentage of the ideal cuts are 40% of all cuts and we want to check if that proportion appears in our Data-Set. The Binomial test Hypothesis testing is :

- H_0 The Proportion of the Ideal Cut is 40%
- H_1 Otherwise

```
prop.test(sum(diamonds$cut == "Ideal"),dim(diamonds)[1],p = 0.40)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  sum(diamonds$cut == "Ideal") out of dim(diamonds)[1], null probability 0.4
## X-squared = 0.046367, df = 1, p-value = 0.8295
## alternative hypothesis: true p is not equal to 0.4
## 95 percent confidence interval:
##  0.3954011 0.4036863
## sample estimates:
```

```
##         p
## 0.3995365
```

In this example the p-value = 0.4 so we can't reject the null Hypothesis.

If the variable is Continuous and normally Distributed then we make a one sample t-test.

- H_0 The mean doesn't differ from 55
- H_1 The mean differs from 55

Suppose we want to check whether the averaging depth differs or not from 55.

```
t.test(x = diamonds$depth,mu=61)
```

```
##
##  One Sample t-test
##
## data:  diamonds$depth
## t = 121.49, df = 53939, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 61
## 95 percent confidence interval:
##  61.73731 61.76150
## sample estimates:
## mean of x
##   61.7494
```

The p-value <0.001 so we reject the null hypothesis and accept with great certainty that the mean is not equal to 55.

For the non-normal Variable the appropriate test is the Wilcoxon test.

```
wilcox.test(x = diamonds$price,mu=2401 , paired = FALSE)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  diamonds$price
## V = 923010000, p-value < 2.2e-16
## alternative hypothesis: true location is not equal to 2401
```

3.2 Two Sample comparison

3.2.1 Paired Samples

Next Statistical analysis that we may have is to compare samples. If these samples are paired meaning that we have two related observations (i.e. two observations per subject) then we use Wilcoxon Signed-Rank Test for the Categorical data and Paired t-test for the Normally distributed continuous data and Wilcoxon Signed-Rank Test for the other continuous.

For the following examples I will use the hsb2 dataset of UCLA found in `http://www.ats.ucla.edu/stat/data/hsb2.csv`

3.2.1.1 Sign Test mcnemar.test(X)

3.2.1.2 Paired t-test

If the variables are Continuous then we can make a Paired t-test to check if the means of the tow samples differ. The Hypothesis testing is:

- H_0: The means are equal
- H_1: The means are not equal

```
t.test(hsb2$write, hsb2$read, paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  hsb2$write and hsb2$read
## t = 0.86731, df = 199, p-value = 0.3868
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.6941424  1.7841424
## sample estimates:
## mean of the differences
##                   0.545
```

3.2.1.3 Wilcoxon Signed-Rank Test

The Wilcoxon Signed-Rank Test is for continuous Data that are not Normally distributed and the Hypothesis Test is :

- H_0: The samples are identical
- H_1: Otherwise

```
wilcox.test(hsb2$write, hsb2$read, paired = TRUE)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  hsb2$write and hsb2$read
## V = 9261, p-value = 0.3666
## alternative hypothesis: true location shift is not equal to 0
```

Here we check whether the Write Vs Read exams of the same student are identically distributed. The p-value is greater than 0.05 so we do not reject the null Hypothesis.

3.2.2 Independent Samples

3.2.2.1 X^2 test and Fisher's exact test

If the samples are independent and Categorical then the appropriate statistic is X^2 test. X^2 test creates a kxn contingency table (in this case a 2x2) and check X^2 test has the following assumption that all contingency cells having expected values < 5, if that is not true then we make a Fisher's exact test.

3.2.2.2 Independent (unpaired) samples t-test

We are assuming Normality in both samples. For example if we are interested to check if the `ideal` cut have on average higher depth than the `Fair` cut we make a independent t-test.

```r
with(diamonds, t.test(depth[cut == "Fair"], depth[cut == "Ideal"]))
```

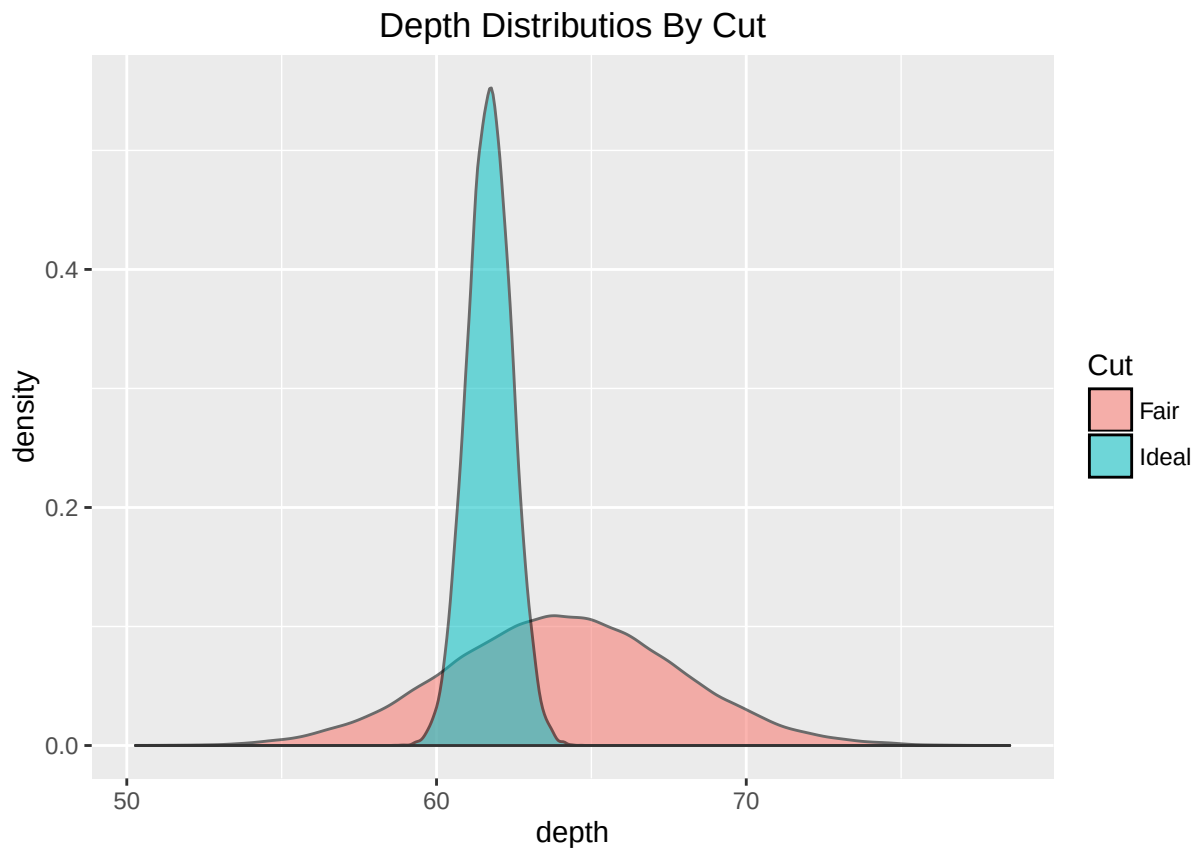
```
##
##  Welch Two Sample t-test
##
## data:  depth[cut == "Fair"] and depth[cut == "Ideal"]
## t = 25.648, df = 1618.4, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  2.153915 2.510637
## sample estimates:
## mean of x mean of y
##  64.04168  61.70940
```

The average mean of depth is 64.04 and 61.70 for the `Fair` and the `Ideal` cuts. The p-value < 0.001 so with great confidence we can say that the average depth in the `Fair` cutted diamonds is higher than the depth in the `Ideal` cutted diamonds. The comparison that the t-test does can be drawn

```r
mean1= mean(diamonds[diamonds$cut=="Ideal",]$depth)
mean2= mean(diamonds[diamonds$cut=="Fair",]$depth)
SD1= sd(diamonds[diamonds$cut=="Ideal",]$depth)
SD2= sd(diamonds[diamonds$cut=="Fair",]$depth)

X= data.frame(cbind(depth=c(Ideal = rnorm(50000,mean1,SD1), Fair = rnorm(50000,mean2,SD2))),Cut=c(rep("


ggplot(X,aes(x=depth, fill= Cut)) +
  geom_density(alpha=0.55) +
  labs(title="Depth Distributios By Cut")
```



3.2.2.3 Mann-Whitney-Wilcoxon Test (non parametric test) Two data samples are independent if they come from distinct populations and the samples do not affect each other. Using the Mann-Whitney-Wilcoxon Test, we can decide whether the population distributions are identical without assuming them to follow the normal distribution.

```
wilcox.test(carat ~ cut, data=diamonds[diamonds$cut == "Ideal" | diamonds$cut == "Fair",])
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  carat by cut
## W = 24671000, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

3.3 Three or more samples comparison

When we have 3 or more samples first and the samples are "Paired" (matched), then the appropriate statistical analyses are Cochrane Q test for categorical, Repeated measures ANOVA and Friedman test for Continuous depending on the Normality.

3.3.1 Paired

3.3.1.1 Cochran's Q test If the samples are matched (repeated measures of the same person) and Categorical we can use the Cochrane Q test. The Hypothesis Test is : * H_0 : The difference in probabilities is equal to 0 * H_1 : The difference in probabilities is not equal to 0

The Statistical test is X^2 distributed and it is calculated by the following:

$$Q = \frac{(k-1)(k\sum_{i=1}^{N} G_j^2 - (\sum_{i=1}^{N} G_j)^2)}{k\sum_{i=1}^{N} L_i - \sum_{i=1}^{N} L_i^2}$$

For Example let's suppose that we have a series of drugs to cure 4 diseases (Disease A, Disease B, Disease C ,Disease D) the results are binomial 0 for no cure, 1 for Cure and there are 4 drugs (Drug A ,Drug B ,Drug C ,Drug D) to be checked if they have the same results or not.

	Disease A	Disease B	Disease C	Disease D
Drug A	1	1	1	1
Drug B	1	1	0	1
Drug C	0	0	0	0
Drug D	0	1	0	0

```
##
##  Cochran's Q test
##
## data:  results by drugs, block = diseases
## Q = 8.5714, df = 3, p-value = 0.03557
## alternative hypothesis: true difference in probabilities is not equal to 0
## sample estimates:
## proba in group Drug A proba in group Drug B proba in group Drug C
##                  1.00                  0.75                  0.00
## proba in group Drug D
##                  0.25
##
##         Pairwise comparisons using Wilcoxon sign test
##
##        Drug A Drug B Drug C
## Drug B    1.0      -      -
## Drug C    0.5   0.50      -
## Drug D    0.5   0.75      1
##
## P value adjustment method: fdr
```

The p-value < 0.05 shows that we reject the null Hypothesis meaning that the effectiveness of at least two treatments differ. The cochran.qtest() command for less than 0.05 p-value makes a pairwise Wilcoxon sign test.

3.3.1.2 Repeated Measures ANOVA. If the samples are Continuous and satisfy the following assumptions :

- Sphericity (Homogeneity over the measures ~ equal Variances of the residuals)
- Normality

The Hypothesis is :

- H_0 : The mean is the same during the measurements

- H_1 : Otherwise

The data below are randomly generated.

ID	month	Blood_Pressure
1	1	18.87905
2	1	19.53965
3	1	23.11742
4	1	20.14102
5	1	20.25858
6	1	23.43013

```
##
## Error: ID
##           Df Sum Sq Mean Sq F value Pr(>F)
## Residuals  9  27.03   3.003
##
## Error: ID:month
##           Df Sum Sq Mean Sq F value   Pr(>F)
## month      3 254.92   84.97   26.77 3.03e-08 ***
## Residuals 27  85.72    3.17
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

ID	Blood_Pressure.1	Blood_Pressure.2	Blood_Pressure.3	Blood_Pressure.4
1	18.87905	20.44816	12.86435	14.85293
2	19.53965	18.71963	14.56405	13.40986
3	23.11742	18.80154	12.94799	15.79025
4	20.14102	18.22137	13.54222	15.75627
5	20.25858	16.88832	13.74992	15.64316
6	23.43013	21.57383	11.62661	15.37728

```
## Note: model has only an intercept; equivalent type-III tests substituted.

##
## Univariate Type III Repeated-Measures ANOVA Assuming Sphericity
##
##                  SS num Df Error SS den Df        F    Pr(>F)
## (Intercept) 11343.9      1   27.028      9 3777.413 4.030e-13 ***
## month         254.9      3   85.715     27   26.767 3.032e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Mauchly Tests for Sphericity
##
##       Test statistic p-value
## month        0.30399 0.10403
##
##
```

```
## Greenhouse-Geisser and Huynh-Feldt Corrections
##  for Departure from Sphericity
##
##        GG eps Pr(>F[GG])
## month 0.58505  1.353e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##          HF eps  Pr(>F[HF])
## month 0.7155256 1.96712e-06
```

The result is significant indicating that the mean of blood pressure is not the same by month.

3.3.2 Friedman Test

If we don't have Normality we can use the Friedman Test which is the Repeated Measures ANOVA non parametric equivalent. The Hypothesis test is :

- H_0: The distributions (whatever they are) are the same across repeated measures
- H_1: The distributions across repeated measures are different

```
friedman.test(y = as.matrix(Blood.Pressure.wide[,2:5]))
```

```
##
##  Friedman rank sum test
##
## data:  as.matrix(Blood.Pressure.wide[, 2:5])
## Friedman chi-squared = 21.36, df = 3, p-value = 8.862e-05
```

```
friedman.test(Blood.Pressure$Blood_Pressure,Blood.Pressure$month,Blood.Pressure$ID)
```

```
##
##  Friedman rank sum test
##
## data:  Blood.Pressure$Blood_Pressure, Blood.Pressure$month and Blood.Pressure$ID
## Friedman chi-squared = 21.36, df = 3, p-value = 8.862e-05
```

Both ways show that the blood pressure is changing by the month.

3.3.3 Independent

When the samples are independent we use X^2 for Categorical , One- Way ANOVA and Kruskal Wallis.

3.3.3.1 One-Way ANOVA

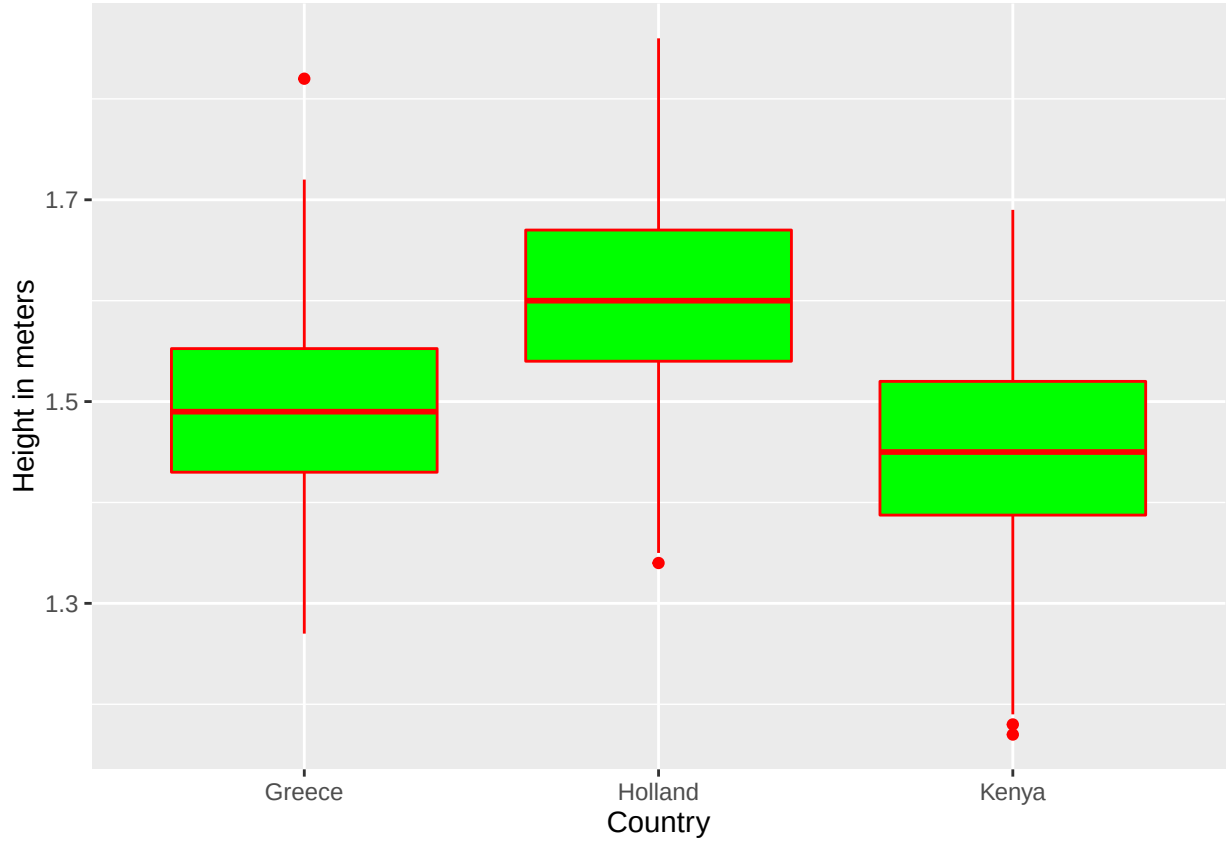
If the samples are Continuous and satisfy the following assumptions :

- Sphericity (Homogeneity over the measures ~ equal Variances of the residuals)
- Normality

Then One-Way ANOVA is the appropriate statistic. For example we have the height of 200 children from 3 different Countries.

```
height.DF= data.frame(  cbind(  Height= round(c(rnorm(200,1.5,0.1),rnorm(200,1.6,0.1),rnorm(200,1.45,0.1
                                Country=c(rep(0,200),rep(1,200) ,rep(2,200) )))
height.DF$Country = factor(height.DF$Country, labels = c("Greece", "Holland", "Kenya"))

ggplot(height.DF, aes(x = as.factor(Country), y = Height)) +
  geom_boxplot(fill = "green", colour = "red") + xlab("Country") +
  ylab("Height in meters")
```



Tests for Sphericity (Homogeneity of Variance) :

- Bartlett test Hypothesis:
- H_0 : The Variances across groups are Homogenous
- H_1 : The Variances across groups are not Homogenous

```
bartlett.test(Height ~ Country, data = height.DF)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  Height by Country
## Bartlett's K-squared = 1.0269, df = 2, p-value = 0.5984
```

We do not reject the H_0.

- Fligner-Killeen (median) test Hypothesis:

- H_0 : The Variances in each of the groups are the same
- H_1 : The Variances in each of the groups are not the same

```
fligner.test(Height ~ Country, data = height.DF)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Height by Country
## Fligner-Killeen:med chi-squared = 0.73324, df = 2, p-value =
## 0.6931
```

We do not reject the H_0.

- and Levene (median) test Hypothesis:
- H_0 : The Variances across groups are Homogenous
- H_1 : The Variances across groups are not Homogenous

```
leveneTest(Height ~ Country, data = height.DF) ## Levene (median) test
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##        Df F value Pr(>F)
## group   2  0.4591  0.632
##       597
```

```
leveneTest(Height ~ Country, data = height.DF,center=mean) ## Levene (mean) test
```

```
## Levene's Test for Homogeneity of Variance (center = mean)
##        Df F value Pr(>F)
## group   2  0.3989 0.6713
##       597
```

```
oneway.test(Height ~ Country,data = height.DF)
```

```
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  Height and Country
## F = 124.22, num df = 2.00, denom df = 397.66, p-value < 2.2e-16
```

```
summary(aov(Height ~ Country,data = height.DF))
```

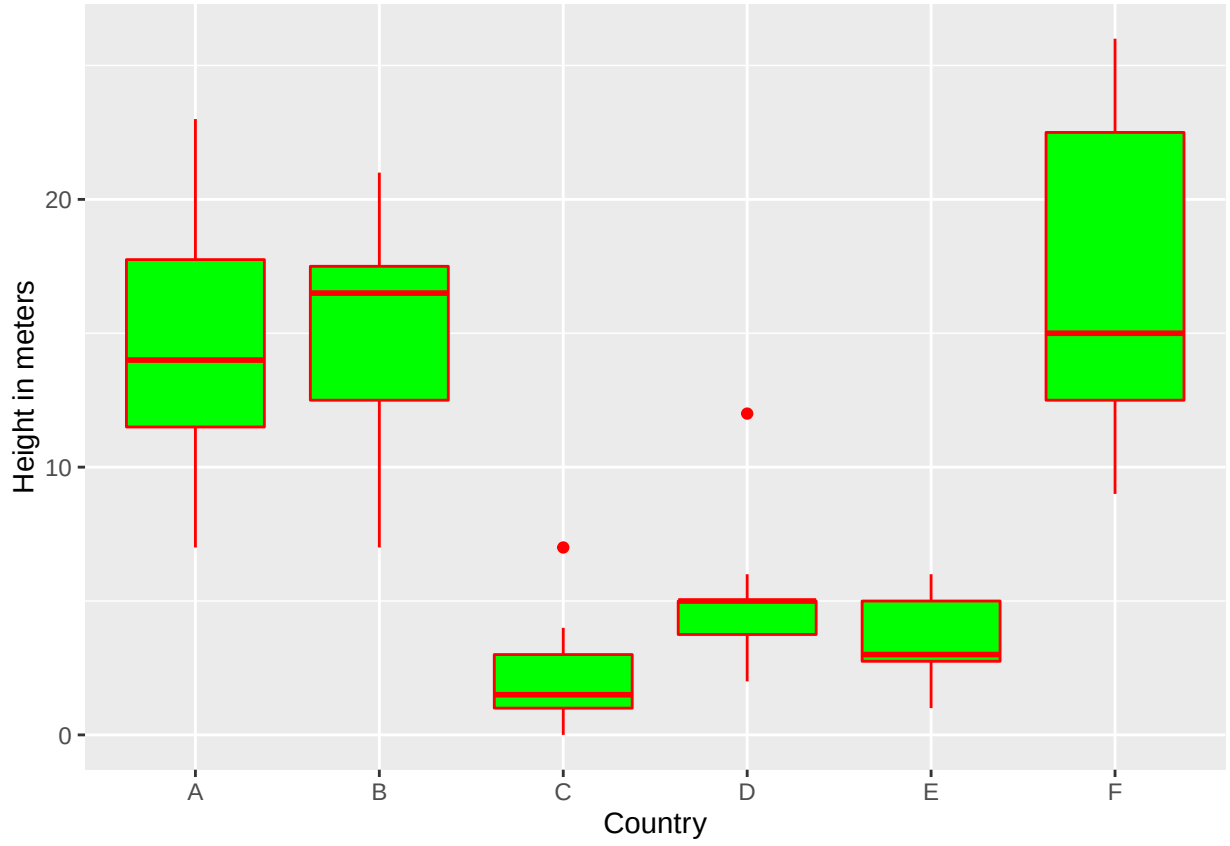
```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Country       2  2.508  1.2540   129.5 <2e-16 ***
## Residuals   597  5.779  0.0097
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It is obvious by the plot that there is a difference in the mean Hight of these 3 countries. The p-value<0.001 shows that there is a statistically significant difference.

3.3.3.2 Kruskal-Wallis test A non parametric equivalent to the One-Way ANOVA is Kruskal-Wallis test.

```
data("InsectSprays")

ggplot(InsectSprays, aes(x = as.factor(spray), y = count)) +
  geom_boxplot(fill = "green", colour = "red") + xlab("Country") +
  ylab("Height in meters")
```




```
kruskal.test(count ~ spray,data = InsectSprays)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  count by spray
## Kruskal-Wallis chi-squared = 54.691, df = 5, p-value = 1.511e-10
```

3.4 Quantify the association between two paired samples

The Correlation Analysis for Categorical variables we use Contingency Coefficients, for Normally distributed samples Pearson and Spearman and Kendal correlation for non parametric.

3.4.1 Contigency Coefficients

To quantify the association two paired Categorical samples we use the Contingency coefficients. In the `vcd` library the `assocstats()` function computes the Pearson chi-Squared test, the Likelihood Ratio chi-Squared test, the phi coefficient (if table 2x2), the contingency coefficient and Cramer's V for possibly stratified contingency tables.

Example:

ID	Treatment	Sex	Age	Improved
57	Treated	Male	27	Some
46	Treated	Male	29	None
77	Treated	Male	30	None
17	Treated	Male	32	Marked
36	Treated	Male	46	Marked
23	Treated	Male	58	Marked

	Placebo	Treated
None	29	13
Some	7	7
Marked	7	21

```
##
## Call: xtabs(formula = ~Improved + Treatment, data = Arthritis)
## Number of cases in table: 84
## Number of factors: 2
## Test for independence of all factors:
##  Chisq = 13.055, df = 2, p-value = 0.001463
##                      X^2 df  P(> X^2)
## Likelihood Ratio 13.530  2 0.0011536
## Pearson          13.055  2 0.0014626
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.367
## Cramer's V        : 0.394
```

3.4.2 Pearson Correlation

For two Normally distributed paired samples if we want to check the association and quantify it then the Pearson Correlation. The correlation coefficient of two variables in a data sample is their covariance divided by the product of their individual standard deviations. It is a normalized measurement of how the two are linearly related.

Formally, the sample correlation coefficient is defined by the following formula, where $S_X = \sum_{i=1}^{n}(x_i - \bar{x})^2$ and $S_Y = \sum_{i=1}^{n}(y_i - \bar{y})^2$ are the sample standard deviations, and $S_{XY} = \sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{y})$ is the sample covariance.

$$r_{XY} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(x_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}}$$

Example.

```r
BMI= matrix(c(1,58,115,2,59,117,3,60,120,4,61,123,5,62,126,6,63,129,7,64,132,8,65,135,9,
              66,139,10, 67,142,11, 68,146,12, 69,150,13, 70,154,14, 71,159,15, 72,164 ) ,
            ncol = 3 ,byrow = T, dimnames = list(c(),c("ID","Height","Weight")) )


cor.test(BMI[,2],BMI[,3],method = "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  BMI[, 2] and BMI[, 3]
## t = 37.855, df = 13, p-value = 1.088e-14
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9860970 0.9985447
## sample estimates:
##       cor
## 0.9954948
```

As we can see we get a value close to 1 meaning a high positive association. Generally from the Pearson's rho we get two kinds of information :

- the sign of the association (positive or negative)
- the magnitude of the association (the closer to zero the weaker it is)

3.4.3 Spearman and Kendal correlation

A Spearman correlation is used when one or both of the variables are not assumed to be normally distributed and interval (but are assumed to be ordinal). The values of the variables are converted in ranks and then correlated.

```r
cor.test(USJudgeRatings$CONT ,USJudgeRatings$INTG,method = "spearman")
```

```
## Warning in cor.test.default(USJudgeRatings$CONT, USJudgeRatings$INTG,
## method = "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  USJudgeRatings$CONT and USJudgeRatings$INTG
## S = 15581, p-value = 0.2576
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## -0.1764773
```

```r
cor.test(USJudgeRatings$CONT ,USJudgeRatings$INTG,method = "kendal")
```

```
## Warning in cor.test.default(USJudgeRatings$CONT, USJudgeRatings$INTG,
## method = "kendal"): Cannot compute exact p-value with ties
```

```
##
##  Kendall's rank correlation tau
##
## data:  USJudgeRatings$CONT and USJudgeRatings$INTG
## z = -1.1036, p-value = 0.2698
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##       tau
## -0.120344
```

3.5 Simple Linear Regression Analysis.

We can simulate a Data Frame to work on.

```
set.seed(123)
Males<-rnorm(50, 1.8, 0.10)
Females<-rnorm(50, 1.65, 0.1)
Height<-round(c(Males,Females),2)
Gender<-factor(c(rep(0,50),rep(1,50)) )
levels(Gender)<-c("Male","Female")
e<- rnorm(100,0,5)
Weight<-round( 25*Height^2 -5*as.integer(Gender) + e)

BMI.DF= data.frame(Weight,Height,Gender)

BMI.DF = BMI.DF[sample(nrow(BMI.DF)),]
rm(e,Females,Males,Gender,Height,Weight)
```

We have a Data Frame of 50 women and 50 men and their Height-Weight measurements, let's take a look into.

```
kable(head(BMI.DF,10))
```

	Weight	Height	Gender
99	53	1.63	Female
14	77	1.81	Male
89	59	1.62	Female
56	70	1.80	Female
38	79	1.79	Male
43	57	1.67	Male
67	65	1.69	Female
8	56	1.67	Male
32	76	1.77	Male
62	49	1.60	Female

```
str(BMI.DF)
```

```
## 'data.frame':    100 obs. of  3 variables:
##  $ Weight: num  53 77 59 70 79 57 65 56 76 49 ...
##  $ Height: num  1.63 1.81 1.62 1.8 1.79 1.67 1.69 1.67 1.77 1.6 ...
```

```
##  $ Gender: Factor w/ 2 levels "Male","Female": 2 1 2 2 1 1 2 1 1 2 ...
```

```
summary(BMI.DF)
```

```
##      Weight          Height         Gender
##  Min.   :41.00   Min.   :1.420   Male  :50
##  1st Qu.:59.75   1st Qu.:1.657   Female:50
##  Median :66.00   Median :1.740
##  Mean   :67.51   Mean   :1.734
##  3rd Qu.:76.25   3rd Qu.:1.810
##  Max.   :95.00   Max.   :2.020
```

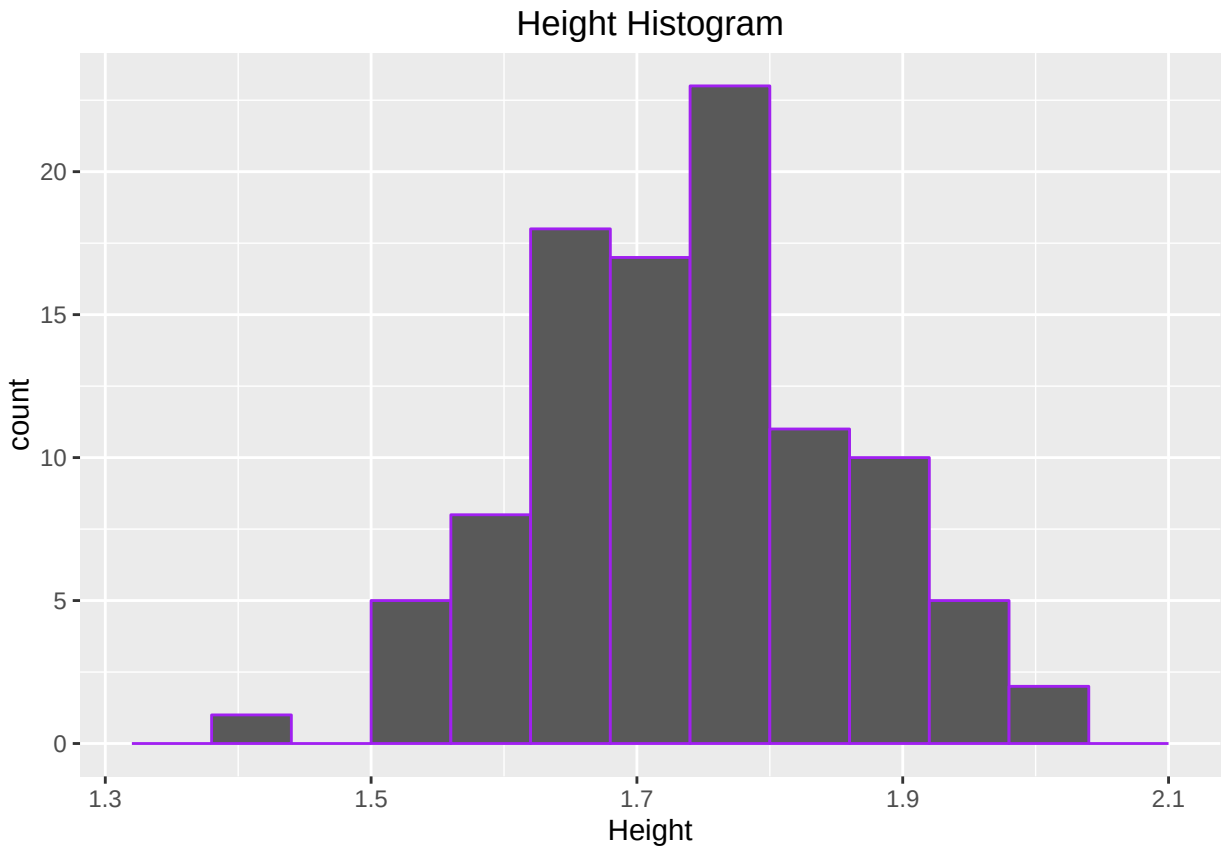
With what we know till we can make several Statistical Analyses and answer many questions. Such as :

- Is Height Normally Distributed?
  - Shapiro Wilk
  - Density (or Histogram) Plot

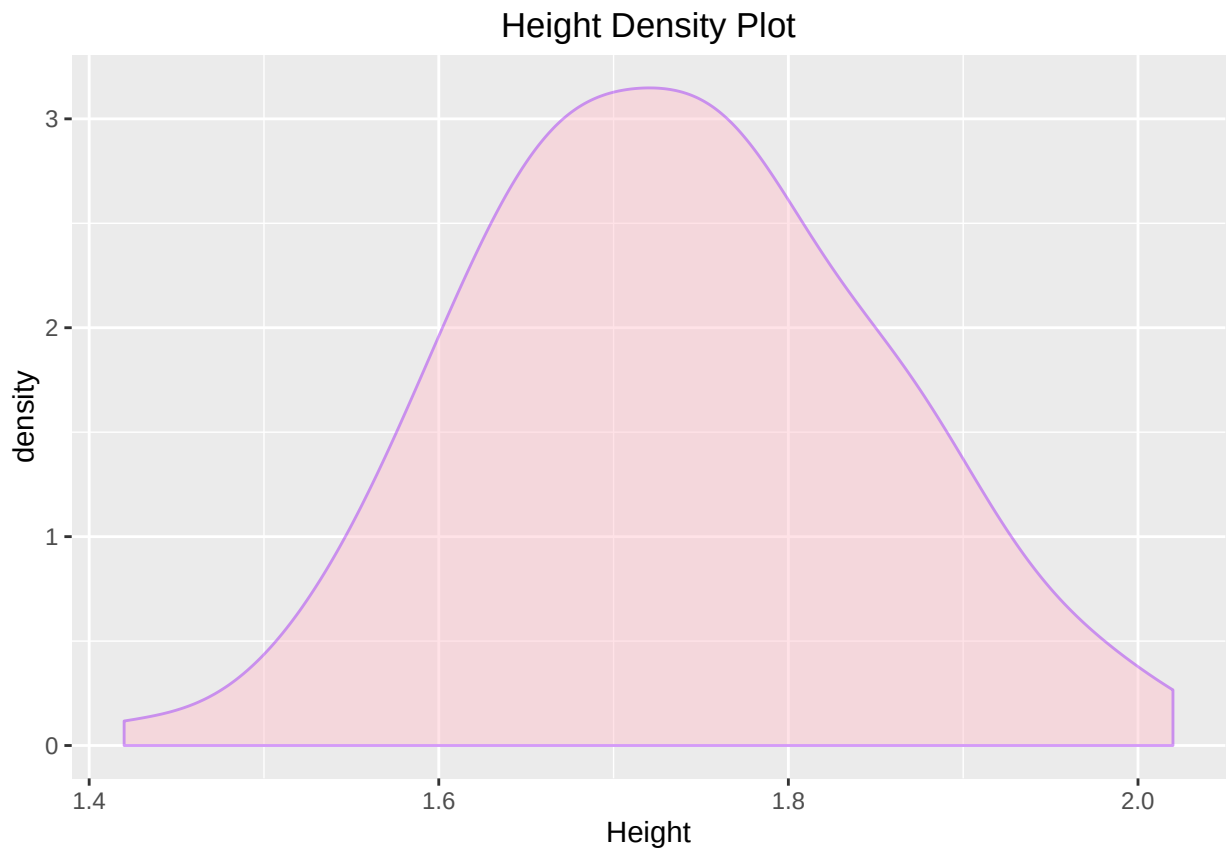
```
shapiro.test(BMI.DF$Height)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  BMI.DF$Height
## W = 0.99439, p-value = 0.956
```

```
ggplot(BMI.DF,aes(Height)) +
  geom_histogram(color= "Purple",bins=10)   +
  labs(title= "Height Histogram")
```




```
ggplot(BMI.DF,aes(Height)) +
  geom_density(color= "Purple",adjust= 1.1,fill="Pink",alpha = 0.45 ) +
  labs(title= "Height Density Plot")
```

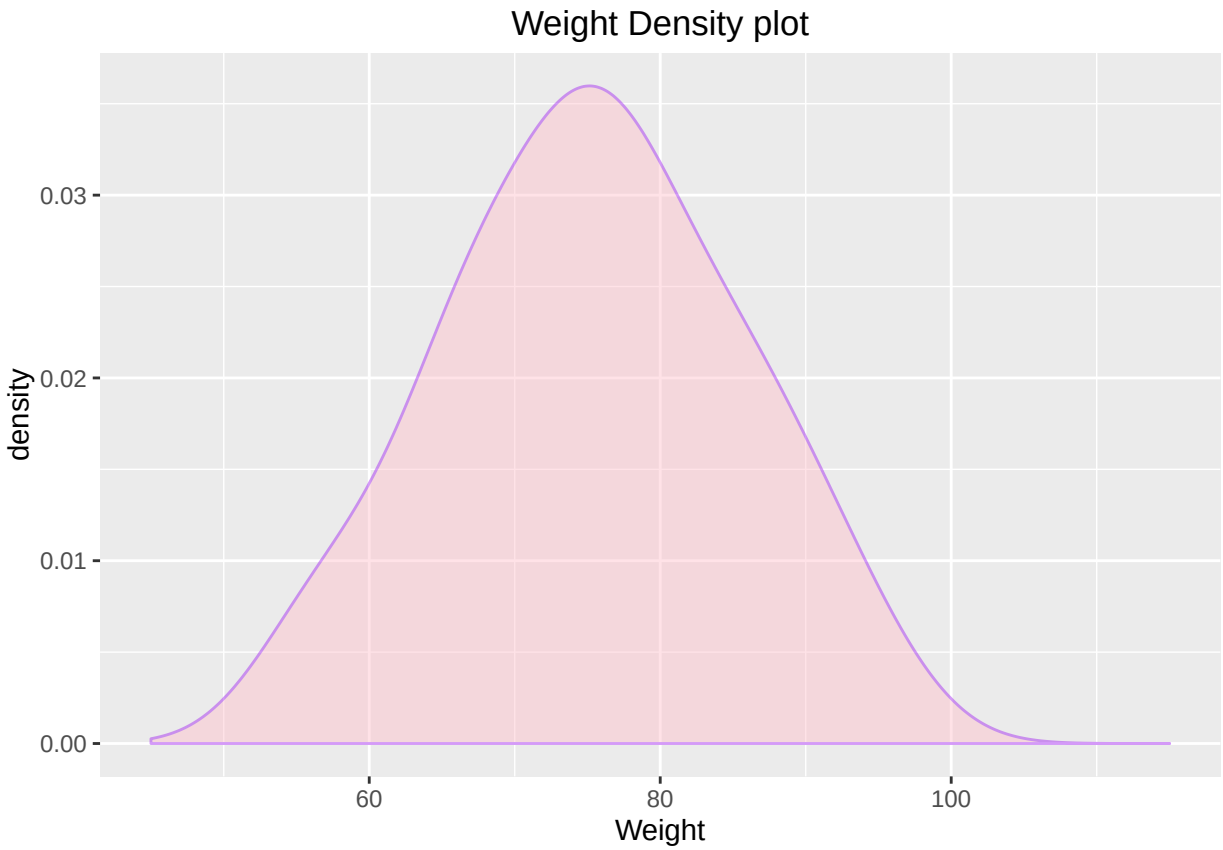


- Is the Average weight of Men good (suppose a Healthy average is 77 Kilograms)
  - One Sample t-test
  - Density Plot

```
t.test(BMI.DF[BMI.DF$Gender == "Male",]$Weight, mu=75)
```

```
##
##  One Sample t-test
##
## data:  BMI.DF[BMI.DF$Gender == "Male", ]$Weight
## t = 0.21308, df = 49, p-value = 0.8321
## alternative hypothesis: true mean is not equal to 75
## 95 percent confidence interval:
##  72.47067 78.12933
## sample estimates:
## mean of x
##      75.3
```

```
ggplot(BMI.DF[BMI.DF$Gender == "Male",],aes(Weight)) +
  geom_density(color= "Purple",adjust= 1.1,fill="Pink",alpha = 0.45)   +  xlim(45,115)  +
labs(title= "Weight Density plot")
```

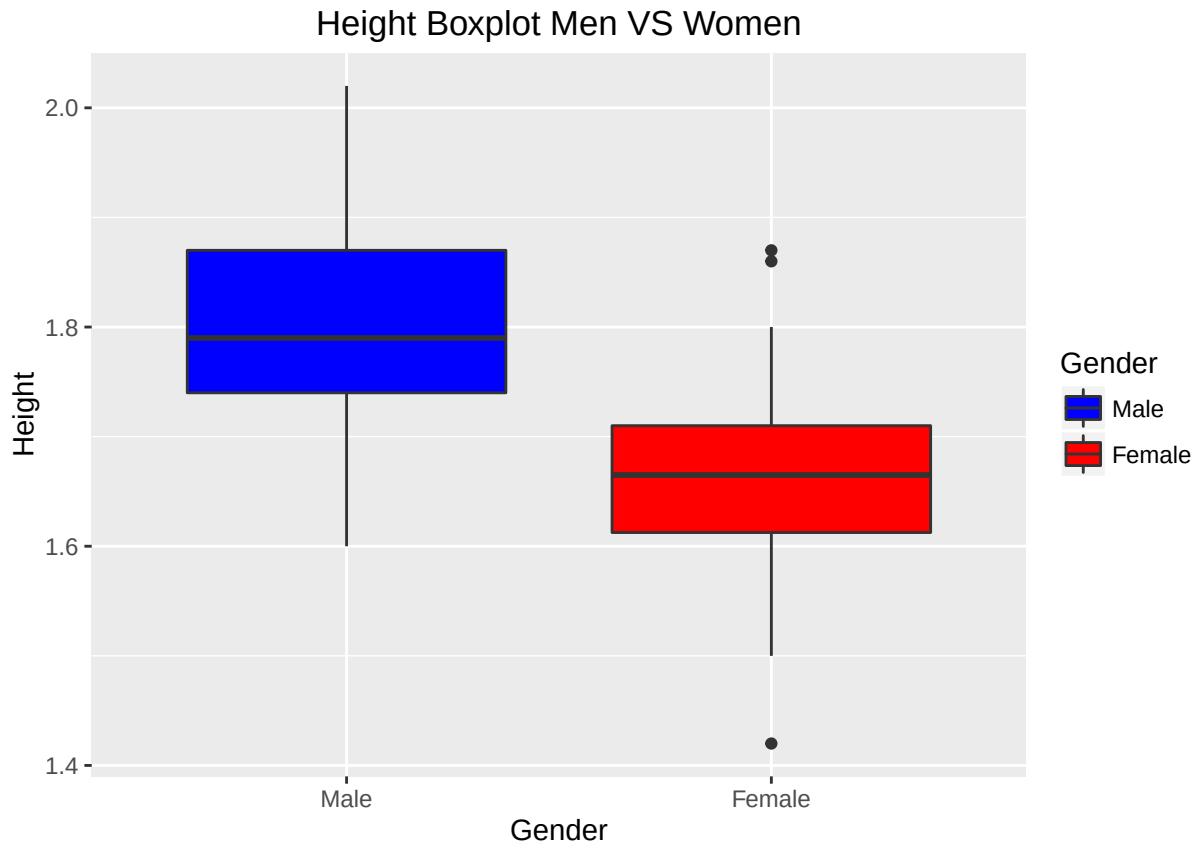


- Is the Average Height of Men equal to women?
  - Two Sample t-test
  - Boxplot

```
t.test(BMI.DF[BMI.DF$Gender == "Male",]$Height, BMI.DF[BMI.DF$Gender == "Female",]$Height )
```

```
##
##  Welch Two Sample t-test
##
## data:  BMI.DF[BMI.DF$Gender == "Male", ]$Height and BMI.DF[BMI.DF$Gender == "Female", ]$Height
## t = 7.5548, df = 97.873, p-value = 2.245e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.1021926 0.1750074
## sample estimates:
## mean of x mean of y
##    1.8036    1.6650
```

```
ggplot(BMI.DF,aes(x = factor(Gender),Height)) +
    geom_boxplot(aes(fill=factor(BMI.DF$Gender))) +
    xlab("Gender") + ggtitle("Height Boxplot Men VS Women") +
    scale_fill_manual(name = "Gender", values = c("Blue","Red")
                      , labels = c("0" = "Men", "1" = "Women"))
```

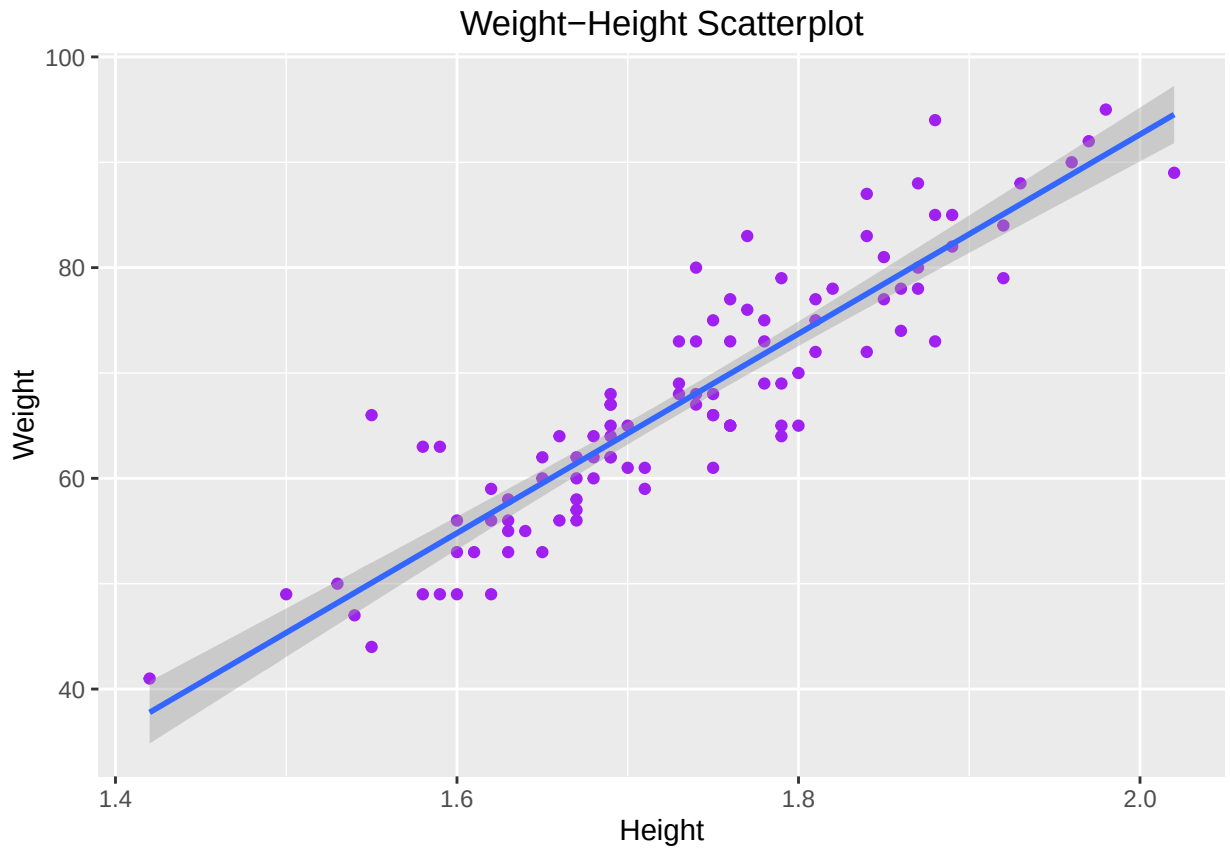


- How much are Height and Weight Correlated?
  - Pearson Correlation
  - Scatterplot

```
cor(BMI.DF$Weight,BMI.DF$Height,method = "pearson")
```

```
## [1] 0.9065306
```

```
ggplot(BMI.DF,aes(y = Weight,x = Height)) +
  geom_point(color= "Purple") +  ## Control the points
  labs(title= "Weight-Height Scatterplot") +
  geom_smooth(method = "lm")
```



How much information can we gather using correlation? Not much... We want to know more than if there is a Correlation between 2 variables and how strong it is.

The setting of a simple linear regression model is:

- The Data
- Depended Variable: Y $\sim N(\mu, \sigma^2)$
- Independent variable X_i
- The Model: $Y = b_0 + b_1 * X + e$
  - where e $N(0, \sigma^2)$
  - and b_1 is the coefficient to be estimated

An upgrade is Simple Linear Regression and the assumptions implied are :

1. linearity and additivity of the relationship between dependent and independent variables.
2. The mean of residuals is zero
3. statistical independence of the errors
4. homoscedasticity and normality of the errors
5. The X variable and residuals are uncorrelated

If we want suppose that `Weight` is the Dependent Variable and `Height` the Independent then we can fit a linear model of $Y = b_0 + b_i * x_i + e_i$:

- Y : Weight

- X : Height

```
fit<-lm(Weight~Height,data= reg)
summary(fit)
```

```
##
## Call:
## lm(formula = Weight ~ Height, data = reg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7780 -3.7374 -0.3593  2.8379 15.9208
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -96.517      7.732  -12.48   <2e-16 ***
## Height        94.578      4.449   21.26   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.082 on 98 degrees of freedom
## Multiple R-squared:  0.8218, Adjusted R-squared:   0.82
## F-statistic: 451.9 on 1 and 98 DF,  p-value: < 2.2e-16
```

- Transforming Variables

b_0 is the intercept of the model. It is the averaged `Weight` for Height $= 0$. But `Height` can't be equal to zero. A good way to deal with that is to Centralize the data (if it is necessary to interpret the intercept) meaning to add or subtract the mean (or minimum, maximum).

b_1 is the slope of the model. It is the averaged increase in `Weight` for an increase in `Height` by 1. In our example the `Height` is measured in meters. So the interpretation of the slope would be : For the increase of 1 `METER` in height the average increase in `Weight` is 81.59 Kilograms .

That interperetation was really awful, in both Statistical and Medical way. We can transform the `Height` variable into cm so that it makes sense. What do we expect to happen after these trasformations? The Intercept will go to the average Weight for average Height, and the slope will be divided by 100.

```
fit<- lm(Weight~I(100*(Height-mean(Height))),data= reg)
summary(fit)
```

```
##
## Call:
## lm(formula = Weight ~ I(100 * (Height - mean(Height))), data = reg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7780 -3.7374 -0.3593  2.8379 15.9208
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     67.51000    0.50820  132.84   <2e-16 ***
## I(100 * (Height - mean(Height)))  0.94578    0.04449   21.26   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.082 on 98 degrees of freedom
## Multiple R-squared:  0.8218, Adjusted R-squared:   0.82
## F-statistic: 451.9 on 1 and 98 DF,  p-value: < 2.2e-16
```

Now let's check the assumptions:

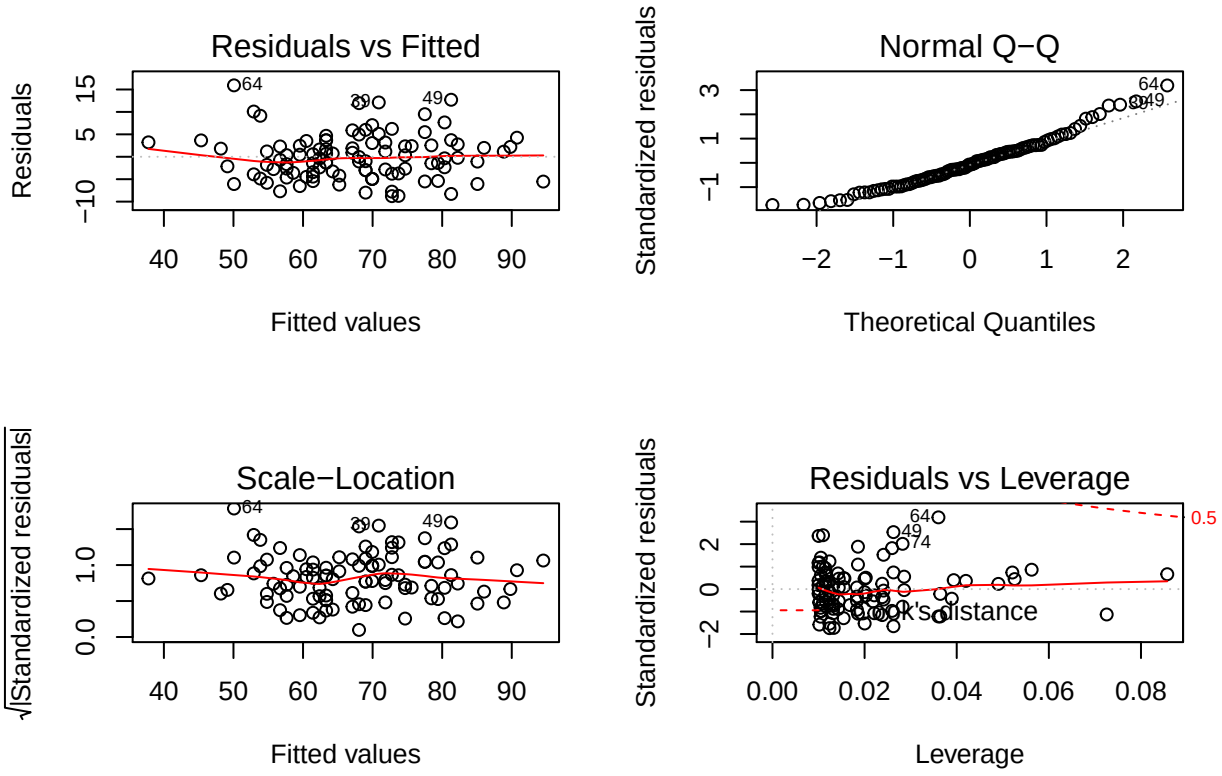
- The first is obvious that we assume linearity.
- The second can be found by simply checking the mean of the residuals.

```
round(mean(fit$residuals),5) ## Rounded up to the fifth decimal
```

```
## [1] 0
```

The Homoscedacity and the Normality of Residuals are shown in the plots below (plus the influence points plot).

```
par(mfrow = c(2, 2))
plot(fit)
```



In the first and third plot we check the Homoscedacity and the red lines should be straight.In our case there assumption is satisfied and there is no pattern. In the second (top right) we check the Normality the closer

the point are in the line the better. In the fourth plot we observe that there are no outliers aka no influence points.

There is an automated way to check the most important assumptions of the linear model.

```
fit<- lm(Weight~I(100*(Height-mean(Height))),data= reg)

library(gvlma)
gvlma(fit)
```

```
##
## Call:
## lm(formula = Weight ~ I(100 * (Height - mean(Height))), data = reg)
##
## Coefficients:
##                       (Intercept)  I(100 * (Height - mean(Height)))
##                           67.5100                            0.9458
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = fit)
##
##                     Value  p-value                   Decision
## Global Stat        9.2070 0.056128    Assumptions acceptable.
## Skewness           7.0386 0.007977 Assumptions NOT satisfied!
## Kurtosis           0.5792 0.446638    Assumptions acceptable.
## Link Function      0.6974 0.403675    Assumptions acceptable.
## Heteroscedasticity 0.8919 0.344962    Assumptions acceptable.
```

But generally the plots are better.

- The `Height` variable is uncorrelated to the residuals.

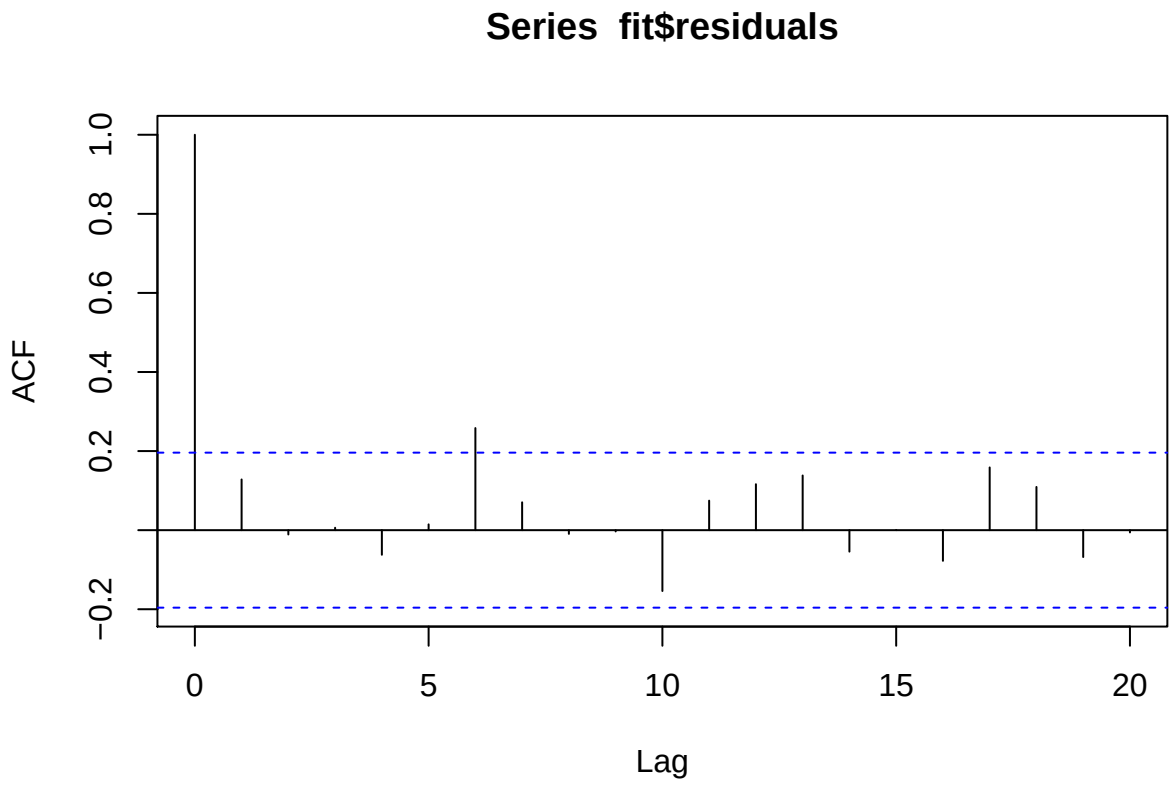
```
cor.test(reg$Height, fit$residuals)
```

```
##
##  Pearson's product-moment correlation
##
## data:  reg$Height and fit$residuals
## t = 1.6085e-16, df = 98, p-value = 1
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1964181  0.1964181
## sample estimates:
##        cor
## 1.6248e-17
```

- The independence of residuals can be checked with the `acf()` function that checks Autocorrelation.

```r
library(lmtest)
library(lawstat)

acf(fit$residuals) # plot of Autocorrelation
```




```r
runs.test(fit$residuals) # H0: Is autocorrelation Random? Yes
```

```
##
##  Runs Test - Two sided
##
## data:  fit$residuals
## Standardized Runs Statistic = 0.60305, p-value = 0.5465
```

```r
dwtest(fit) # H0: Is autocorrelation zero? No
```

```
##
##  Durbin-Watson test
##
## data:  fit
## DW = 1.7335, p-value = 0.08931
## alternative hypothesis: true autocorrelation is greater than 0
```

Two tests have failed.

3.6 Multiple Linear Regression

Now we can go further by inserting more variables in a model.

The setting of a Multiple linear regression model is:

- Depended Variable: Y $\sim N(\mu, \sigma^2)$
- Independent variables X_i
- The Model: $Y = b_0 + b_i * X_i + e$
  - where e $N(0, \sigma^2)$
  - and b_1 is the coefficient to be estimated

We have the same assumptions to check plus Multicollinearity

1. linearity and additivity of the relationship between dependent and independent variables.
2. The mean of residuals is zero
3. statistical independence of the errors
4. homoscedasticity and normality of the errors
5. The X variable and residuals are uncorrelated
6. check for multicollinearity.

LungCap	Age	Height	Smoke	Gender	Caesarean
6.475	6	62.1	no	male	no
10.125	18	74.7	yes	female	no
9.550	16	69.7	no	female	yes
11.125	14	71.0	no	male	no
4.800	5	56.9	no	male	no
6.225	11	58.7	no	female	no

```
## 'data.frame':    725 obs. of  6 variables:
##  $ LungCap  : num  6.47 10.12 9.55 11.12 4.8 ...
##  $ Age      : int  6 18 16 14 5 11 8 11 15 11 ...
##  $ Height   : num  62.1 74.7 69.7 71 56.9 58.7 63.3 70.4 70.5 59.2 ...
##  $ Smoke    : Factor w/ 2 levels "no","yes": 1 2 1 1 1 1 1 1 1 1 ...
##  $ Gender   : Factor w/ 2 levels "female","male": 2 1 1 2 2 1 2 2 2 2 ...
##  $ Caesarean: Factor w/ 2 levels "no","yes": 1 1 2 1 1 1 2 1 1 1 ...
```

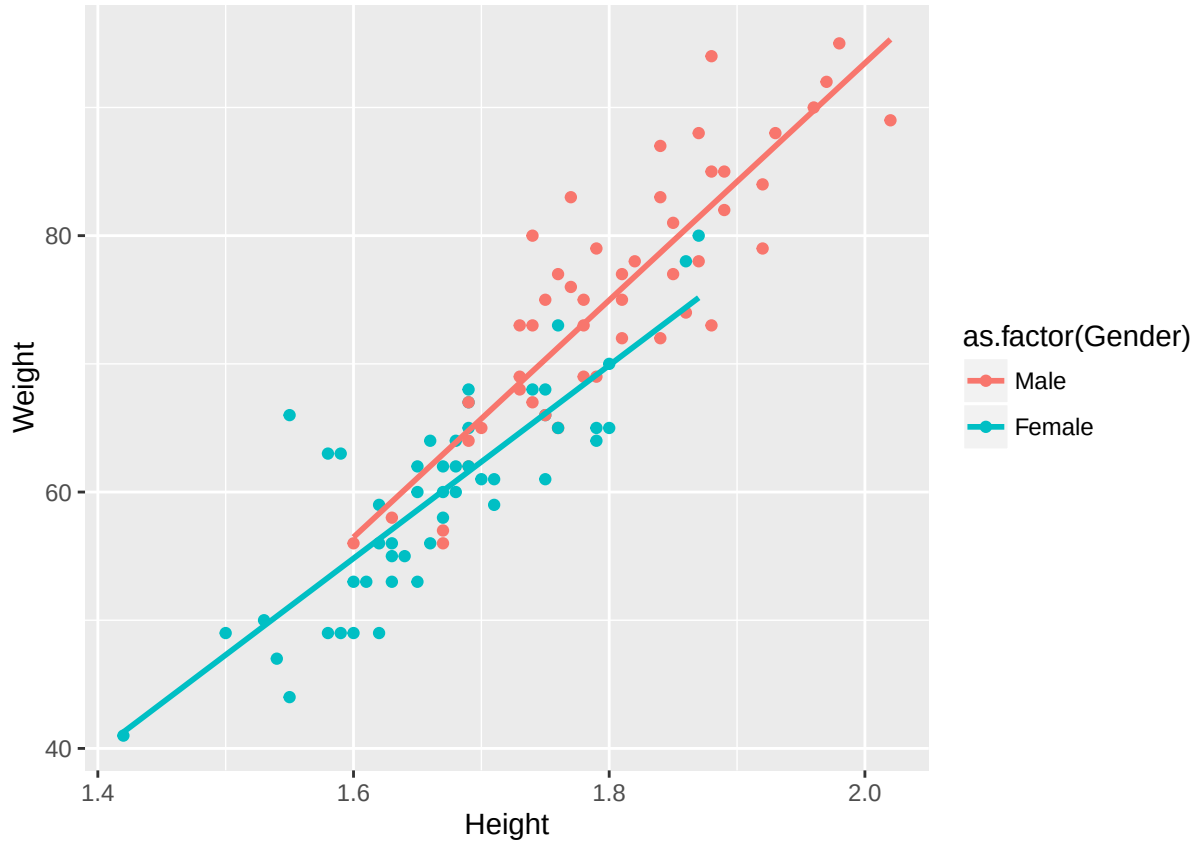
We will make a brief check on the Variables.

3.6.1 Multicollinearity

We have to check for perfect multicollinearity. Let's use all the variables in the `BMI` dataset. No variable should have a score over 4.

```
fit<- lm(Weight~I(100*(Height-mean(Height))) + as.factor(Gender),data= reg)

ggplot(data = reg , aes(y=Weight,x=Height,colour=as.factor(Gender))) + geom_point() +
  geom_smooth(method = "lm",se=F)
```




```
summary(fit)
```

```
##
## Call:
## lm(formula = Weight ~ I(100 * (Height - mean(Height))) + as.factor(Gender),
##     data = reg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7324 -3.6066 -0.6738  3.0289 15.9622
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      69.4654     0.7807   88.98  < 2e-16 ***
## I(100 * (Height - mean(Height)))  0.8419     0.0535   15.74  < 2e-16 ***
## as.factor(Gender)Female          -3.9108     1.2223   -3.20  0.00186 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.858 on 97 degrees of freedom
## Multiple R-squared:  0.8388, Adjusted R-squared:  0.8355
## F-statistic: 252.4 on 2 and 97 DF,  p-value: < 2.2e-16
```

```
vif(fit)
```

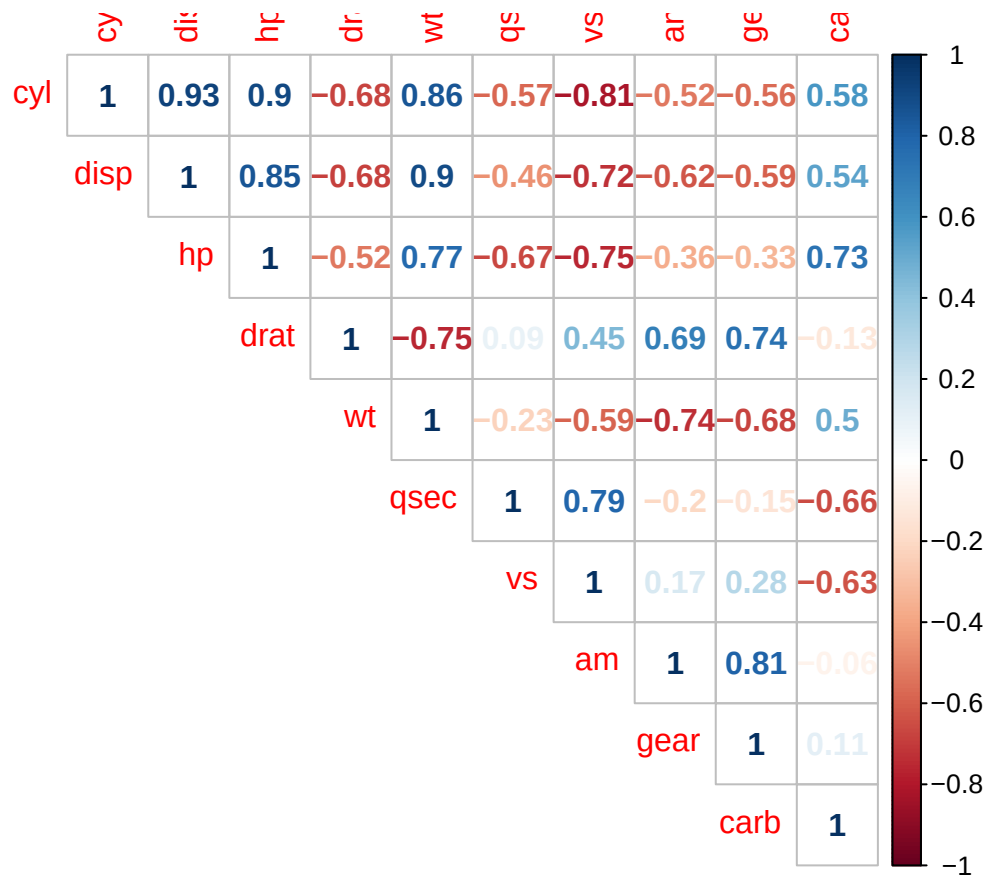
```
## I(100 * (Height - mean(Height)))                as.factor(Gender)
##                          1.582401                         1.582401
```

We can solve that by omitting the highly correlated variables We check the correlation plot (same as in Correlation Analysis) and choose one out of the groups of the highly correlated variables.

```
rm(mtcars)
```

```
## Warning in rm(mtcars): object 'mtcars' not found
```

```
corrplot(cor(mtcars[,-1],method = "spearman"),method = "number",type = "upper")
```



Interpreted from below plot. Correlated pairs:

- disp, cyl, hp, wt
- gear, am
- hp, carb

```
mod <- lm(mpg ~ disp + am + carb + qsec + drat, data=mtcars)
vif(mod)
```

```
##     disp       am     carb     qsec     drat
## 3.860740 3.359487 1.836703 2.885109 2.798873
```

Now that we know our assumptions let start building a model.

```
library(AER)
```

```
## Loading required package: sandwich
```

```
##
## Attaching package: 'AER'
```

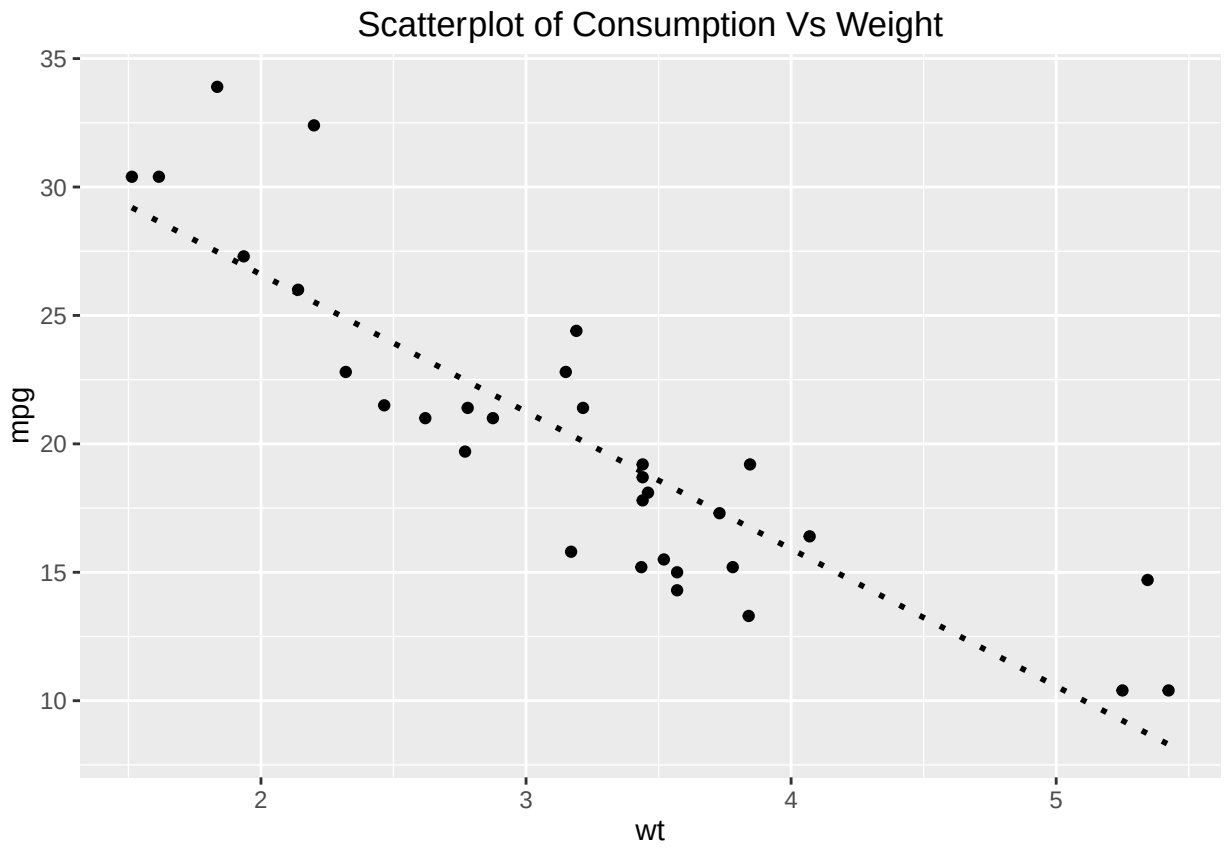
```
## The following object is masked from 'package:VGAM':
##
##     tobit
```

```
data("HousePrices")
```

If we suppose that the mpg is our dependent Variable we observe that there is a high correlation with the cylinder number, the disposition, the weight and the Horse Power.

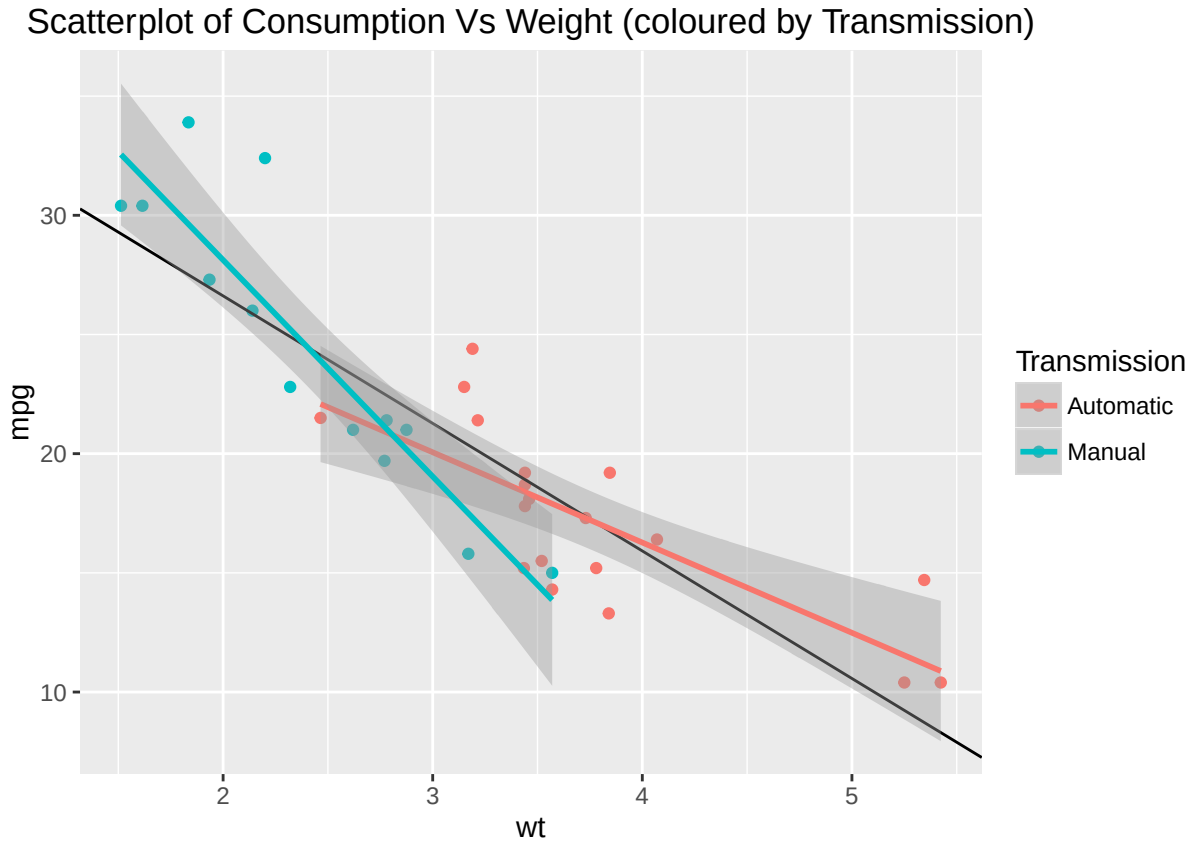
Graphically we can see that the points are following a line.

```
ggplot(mtcars, aes(wt, mpg)) + geom_point() + geom_smooth(method="lm",color="black",se = F,linetype = "
```



But is there a change in the consumption between automatic and manual transmission?

```
ggplot(mtcars, aes(wt, mpg, colour=factor(am))) + geom_point() + geom_abline(intercept = 37.32 , slope =
```



As we can see there is a great difference in the slopes between the Manual and Automatic transmission

```
library(MASS)
fit <- lm(mpg~.,data=mtcars)
step <- stepAIC(fit, direction="both")
```

```
## Start:  AIC=70.9
## mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am + gear + carb
##
##        Df Sum of Sq    RSS    AIC
## - cyl   1    0.0799 147.57 68.915
## - vs    1    0.1601 147.66 68.932
## - carb  1    0.4067 147.90 68.986
## - gear  1    1.3531 148.85 69.190
## - drat  1    1.6270 149.12 69.249
## - disp  1    3.9167 151.41 69.736
## - hp    1    6.8399 154.33 70.348
## - qsec  1    8.8641 156.36 70.765
## <none>              147.49 70.898
## - am    1   10.5467 158.04 71.108
## - wt    1   27.0144 174.51 74.280
##
## Step:  AIC=68.92
```

```
## mpg ~ disp + hp + drat + wt + qsec + vs + am + gear + carb
##
##        Df Sum of Sq    RSS    AIC
## - vs    1    0.2685 147.84 66.973
## - carb  1    0.5201 148.09 67.028
## - gear  1    1.8211 149.40 67.308
## - drat  1    1.9826 149.56 67.342
## - disp  1    3.9009 151.47 67.750
## - hp    1    7.3632 154.94 68.473
## <none>              147.57 68.915
## - qsec  1   10.0933 157.67 69.032
## - am    1   11.8359 159.41 69.384
## + cyl   1    0.0799 147.49 70.898
## - wt    1   27.0280 174.60 72.297
##
## Step:  AIC=66.97
## mpg ~ disp + hp + drat + wt + qsec + am + gear + carb
##
##        Df Sum of Sq    RSS    AIC
## - carb  1    0.6855 148.53 65.121
## - gear  1    2.1437 149.99 65.434
## - drat  1    2.2139 150.06 65.449
## - disp  1    3.6467 151.49 65.753
## - hp    1    7.1060 154.95 66.475
## <none>              147.84 66.973
## - am    1   11.5694 159.41 67.384
## - qsec  1   15.6830 163.53 68.200
## + vs    1    0.2685 147.57 68.915
## + cyl   1    0.1883 147.66 68.932
## - wt    1   27.3799 175.22 70.410
##
## Step:  AIC=65.12
## mpg ~ disp + hp + drat + wt + qsec + am + gear
##
##        Df Sum of Sq    RSS    AIC
## - gear  1     1.565 150.09 63.457
## - drat  1     1.932 150.46 63.535
## <none>              148.53 65.121
## - disp  1    10.110 158.64 65.229
## - am    1    12.323 160.85 65.672
## - hp    1    14.826 163.35 66.166
## + carb  1     0.685 147.84 66.973
## + vs    1     0.434 148.09 67.028
## + cyl   1     0.414 148.11 67.032
## - qsec  1    26.408 174.94 68.358
## - wt    1    69.127 217.66 75.350
##
## Step:  AIC=63.46
## mpg ~ disp + hp + drat + wt + qsec + am
##
##        Df Sum of Sq    RSS    AIC
## - drat  1     3.345 153.44 62.162
## - disp  1     8.545 158.64 63.229
## <none>              150.09 63.457
```

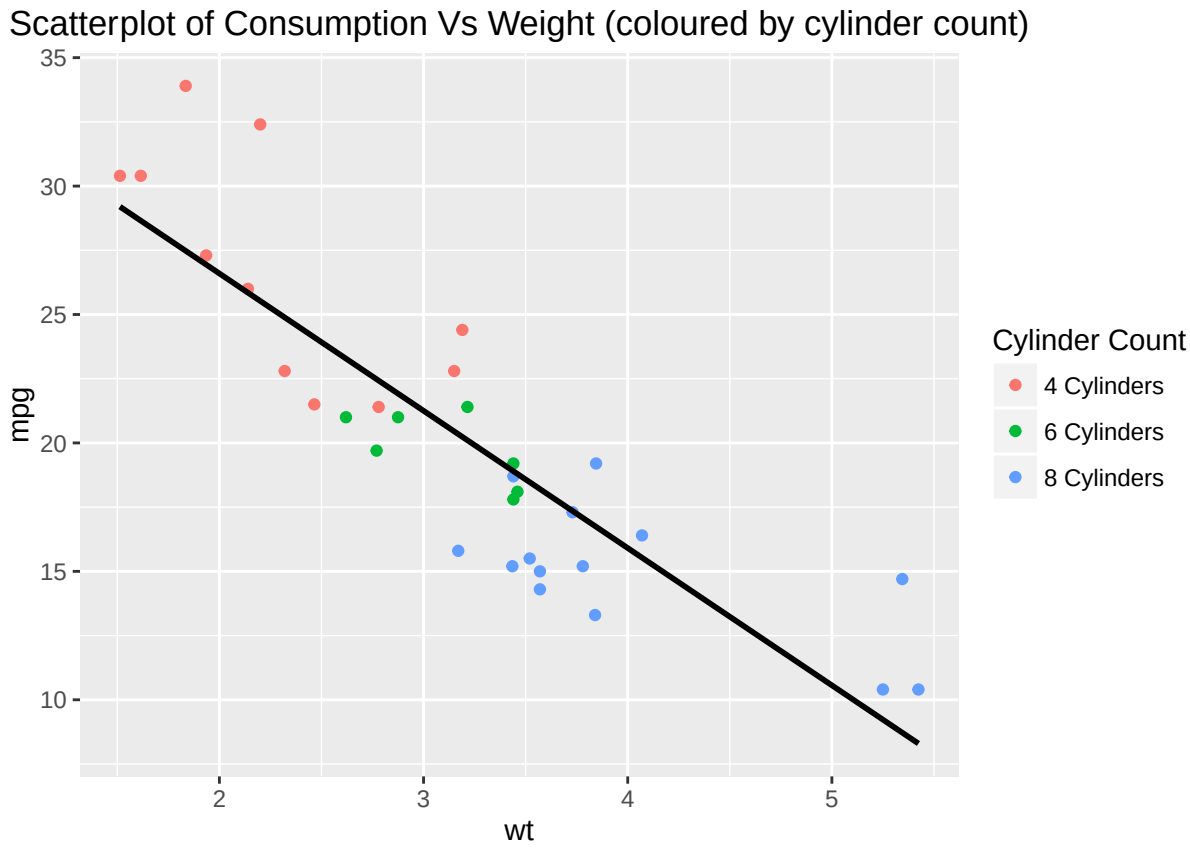
```
## - hp    1    13.285 163.38 64.171
## + gear  1     1.565 148.53 65.121
## + cyl   1     1.003 149.09 65.242
## + vs    1     0.645 149.45 65.319
## + carb  1     0.107 149.99 65.434
## - am    1    20.036 170.13 65.466
## - qsec  1    25.574 175.67 66.491
## - wt    1    67.572 217.66 73.351
##
## Step:  AIC=62.16
## mpg ~ disp + hp + wt + qsec + am
##
##        Df Sum of Sq    RSS    AIC
## - disp  1     6.629 160.07 61.515
## <none>              153.44 62.162
## - hp    1    12.572 166.01 62.682
## + drat  1     3.345 150.09 63.457
## + gear  1     2.977 150.46 63.535
## + cyl   1     2.447 150.99 63.648
## + vs    1     1.121 152.32 63.927
## + carb  1     0.011 153.43 64.160
## - qsec  1    26.470 179.91 65.255
## - am    1    32.198 185.63 66.258
## - wt    1    69.043 222.48 72.051
##
## Step:  AIC=61.52
## mpg ~ hp + wt + qsec + am
##
##        Df Sum of Sq    RSS    AIC
## - hp    1     9.219 169.29 61.307
## <none>              160.07 61.515
## + disp  1     6.629 153.44 62.162
## + carb  1     3.227 156.84 62.864
## + drat  1     1.428 158.64 63.229
## - qsec  1    20.225 180.29 63.323
## + cyl   1     0.249 159.82 63.465
## + vs    1     0.249 159.82 63.466
## + gear  1     0.171 159.90 63.481
## - am    1    25.993 186.06 64.331
## - wt    1    78.494 238.56 72.284
##
## Step:  AIC=61.31
## mpg ~ wt + qsec + am
##
##        Df Sum of Sq    RSS    AIC
## <none>              169.29 61.307
## + hp    1     9.219 160.07 61.515
## + carb  1     8.036 161.25 61.751
## + disp  1     3.276 166.01 62.682
## + cyl   1     1.501 167.78 63.022
## + drat  1     1.400 167.89 63.042
## + gear  1     0.123 169.16 63.284
## + vs    1     0.000 169.29 63.307
## - am    1    26.178 195.46 63.908
```

```
## - qsec  1   109.034 278.32 75.217
## - wt    1   183.347 352.63 82.790
```

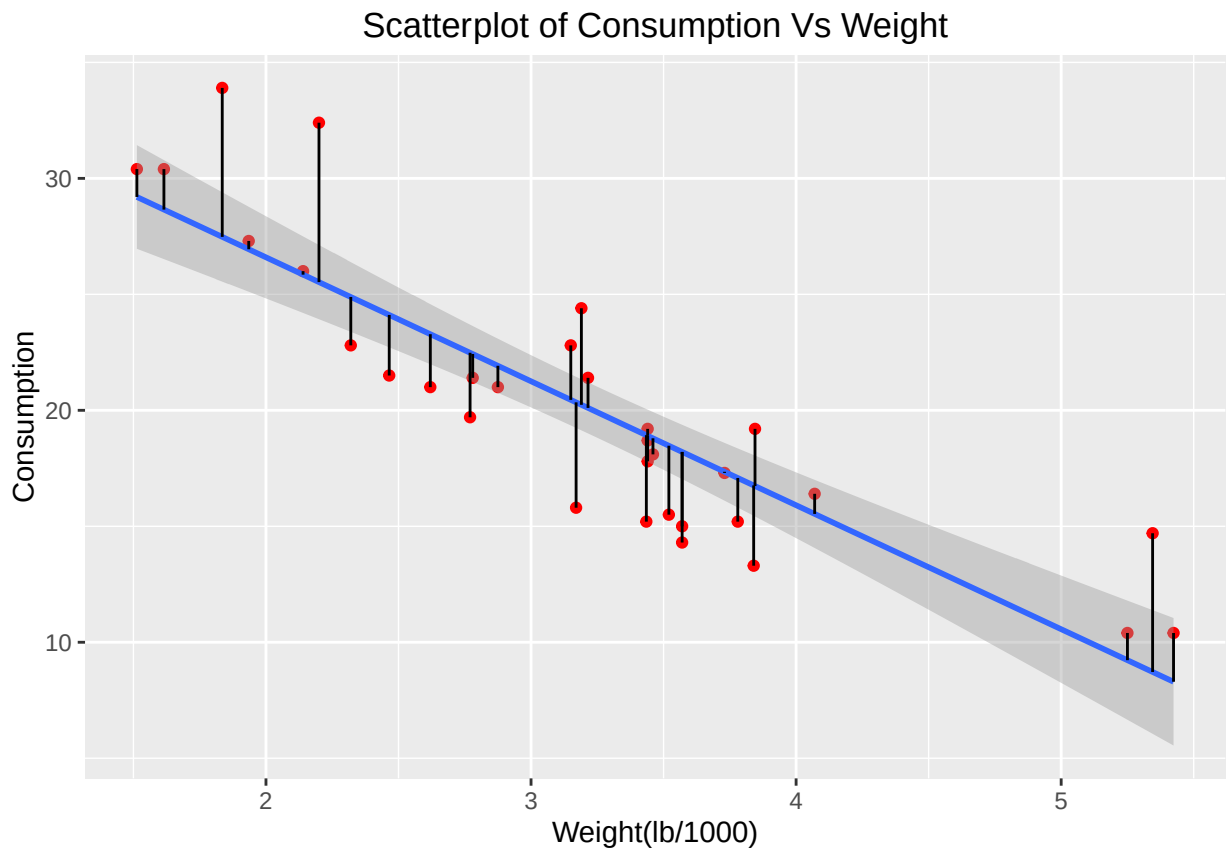
```
step$anova # display results
```

```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## mpg ~ cyl + disp + hp + drat + wt + qsec + vs + am + gear + carb
##
## Final Model:
## mpg ~ wt + qsec + am
##
##
##     Step Df   Deviance Resid. Df Resid. Dev      AIC
## 1                              21   147.4944 70.89774
## 2  - cyl  1 0.07987121        22   147.5743 68.91507
## 3   - vs  1 0.26852280        23   147.8428 66.97324
## 4 - carb  1 0.68546077        24   148.5283 65.12126
## 5 - gear  1 1.56497053        25   150.0933 63.45667
## 6 - drat  1 3.34455117        26   153.4378 62.16190
## 7 - disp  1 6.62865369        27   160.0665 61.51530
## 8   - hp  1 9.21946935        28   169.2859 61.30730
```

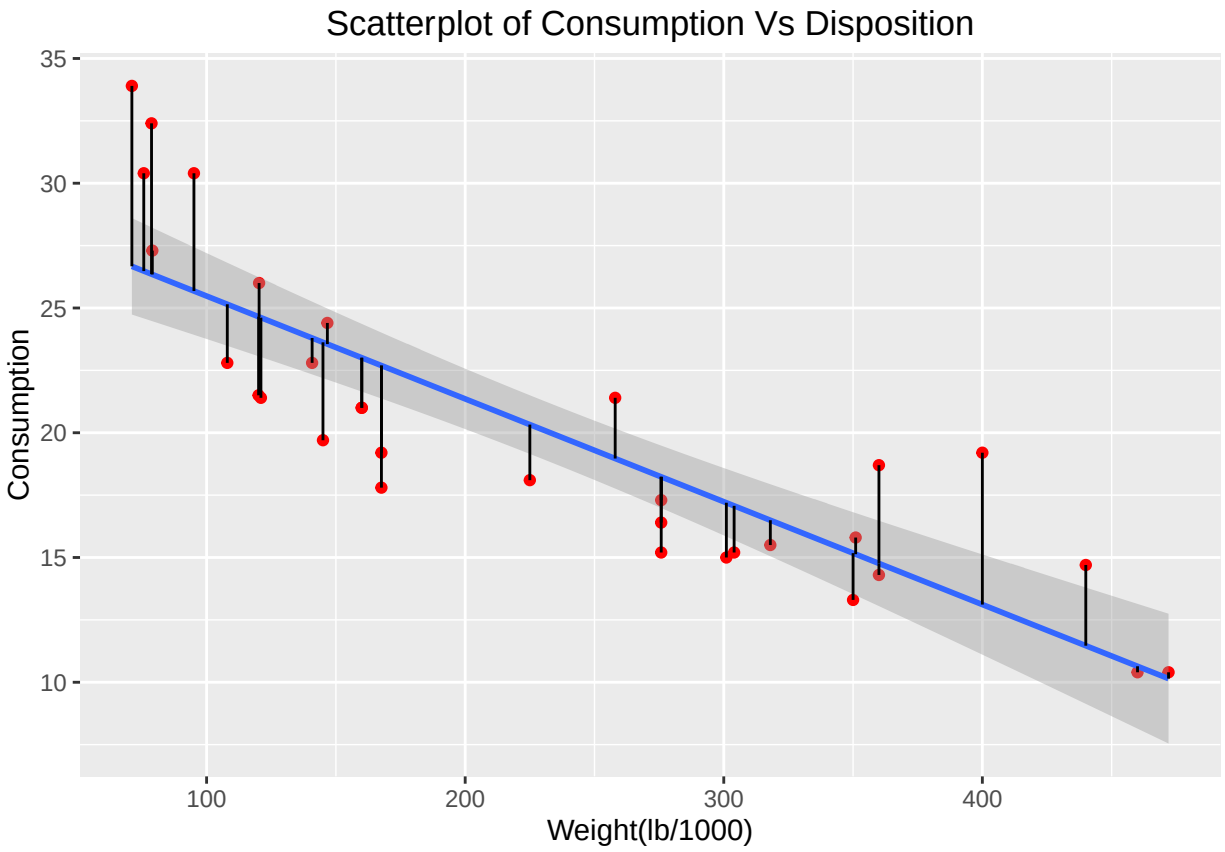
```
ggplot(mtcars, aes(wt, mpg, colour=factor(mtcars$cyl))) + geom_point() + geom_smooth(method="lm",color=
                      labels=c("4 Cylinders", "6 Cylinders", "8 Cylinders"))
```



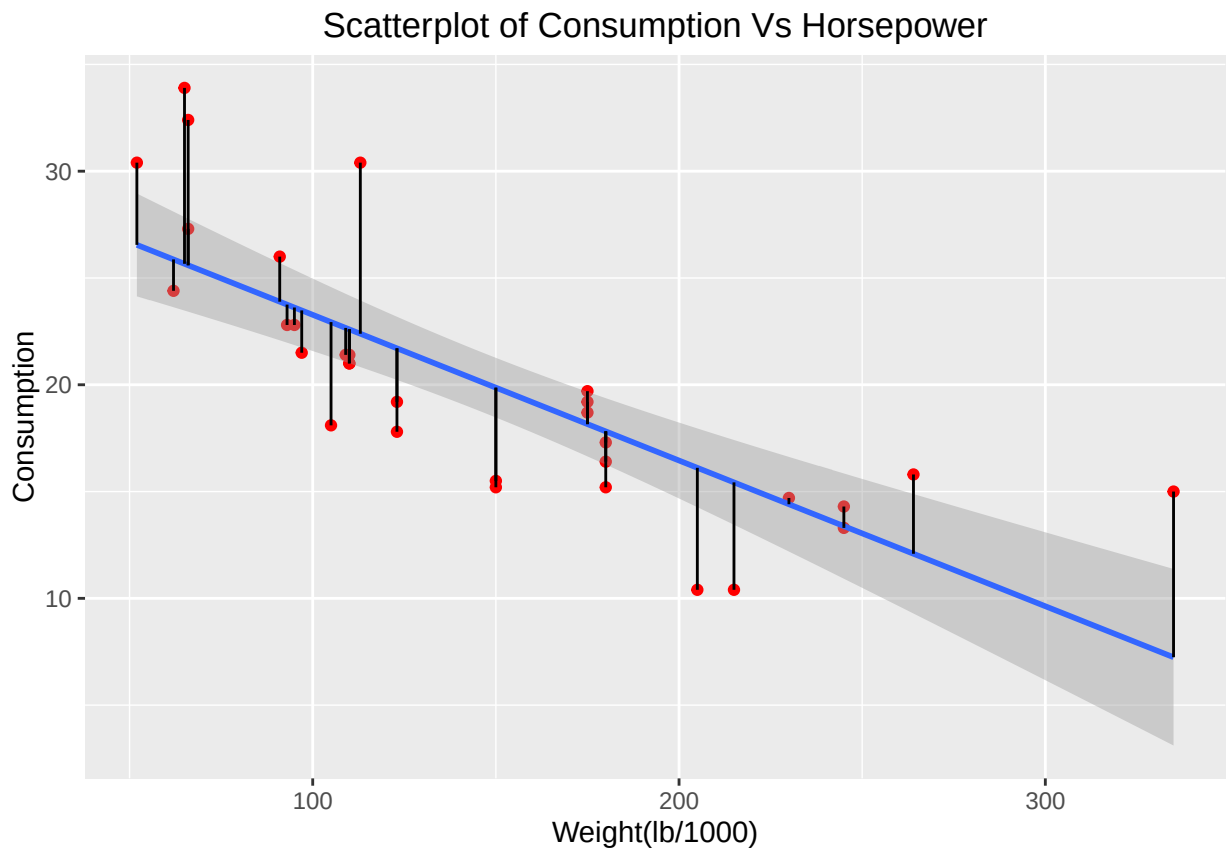

```
fit=lm(mpg~wt, data=mtcars)
pre <- predict(fit)
ggplot(mtcars, aes(wt, mpg)) + geom_point(color="red") + geom_smooth(method="lm") +
  geom_segment(aes(x =wt , y =mpg , xend = wt, yend = pre)) +
  ylab("Consumption") + xlab("Weight(lb/1000)")+ labs(title = "Scatterplot of Consumption Vs Weight")
```



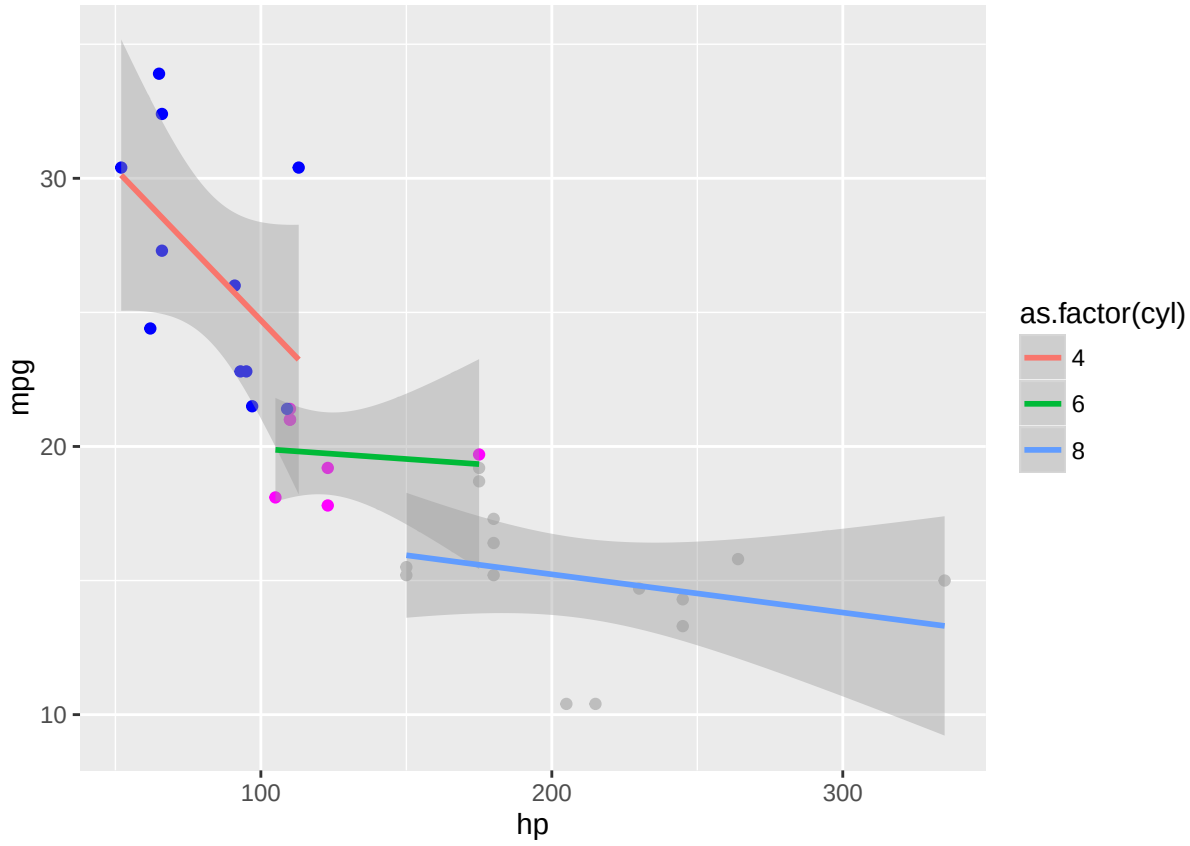

```
fit=lm(mpg~disp, data=mtcars)
pre <- predict(fit)
ggplot(mtcars, aes(disp, mpg)) + geom_point(color="red") + geom_smooth(method="lm") +
  geom_segment(aes(x =disp , y =mpg , xend = disp, yend = pre)) +
  ylab("Consumption") + xlab("Weight(lb/1000)")+ labs(title = "Scatterplot of Consumption Vs Dispositio
```




```
fit=lm(mpg~hp, data=mtcars)
pre <- predict(fit)
ggplot(mtcars, aes(hp, mpg)) + geom_point(color="red") + geom_smooth(method="lm") +
  geom_segment(aes(x =hp , y =mpg , xend = hp, yend = pre)) +
  ylab("Consumption") + xlab("Weight(lb/1000)")+ labs(title = "Scatterplot of Consumption Vs Horsepower
```




```
ggplot(mtcars, aes(hp, mpg,group=as.factor(cyl), colour=as.factor(cyl) )) + geom_point(color=as.factor(
```



4 Tranforming Linear Regression

4.1 Checking The Data

5 References