intro_to_r_lecture2_excercise.R

kristineccles

2020-01-12

```
#############################################################
# Introduction to R
# Lecture 2- Statistics
# By: Kristin Eccles
# Written in R 3.6.2
############################################################

# Install Libraries
# only need to run this once
#install.packages(c("psych", "car", "stats", "corrplot", "factoextra","lmtest", "devtools"))

# Load Libraries
library(ggplot2)
library(psych) # describe and mutli.hist
```

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(car) #stats
```

```
## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:psych':
##
##     logit
```

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(stats)# cor, princomp
library(lmtest)
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
```

```
##     as.Date, as.Date.numeric
```

```
library(factoextra) #pca plots
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(devtools) #pca plots
```

```
## Loading required package: usethis
```

```
install_github("vqv/ggbiplot")
```

```
## Skipping install of 'ggbiplot' from a github remote, the SHA1 (7325e880) has not changed since last
##   Use `force = TRUE` to force installation
```

```
library(ggbiplot)
```

```
## Loading required package: plyr

## Loading required package: scales

##
## Attaching package: 'scales'

## The following objects are masked from 'package:psych':
##
##     alpha, rescale

## Loading required package: grid
```

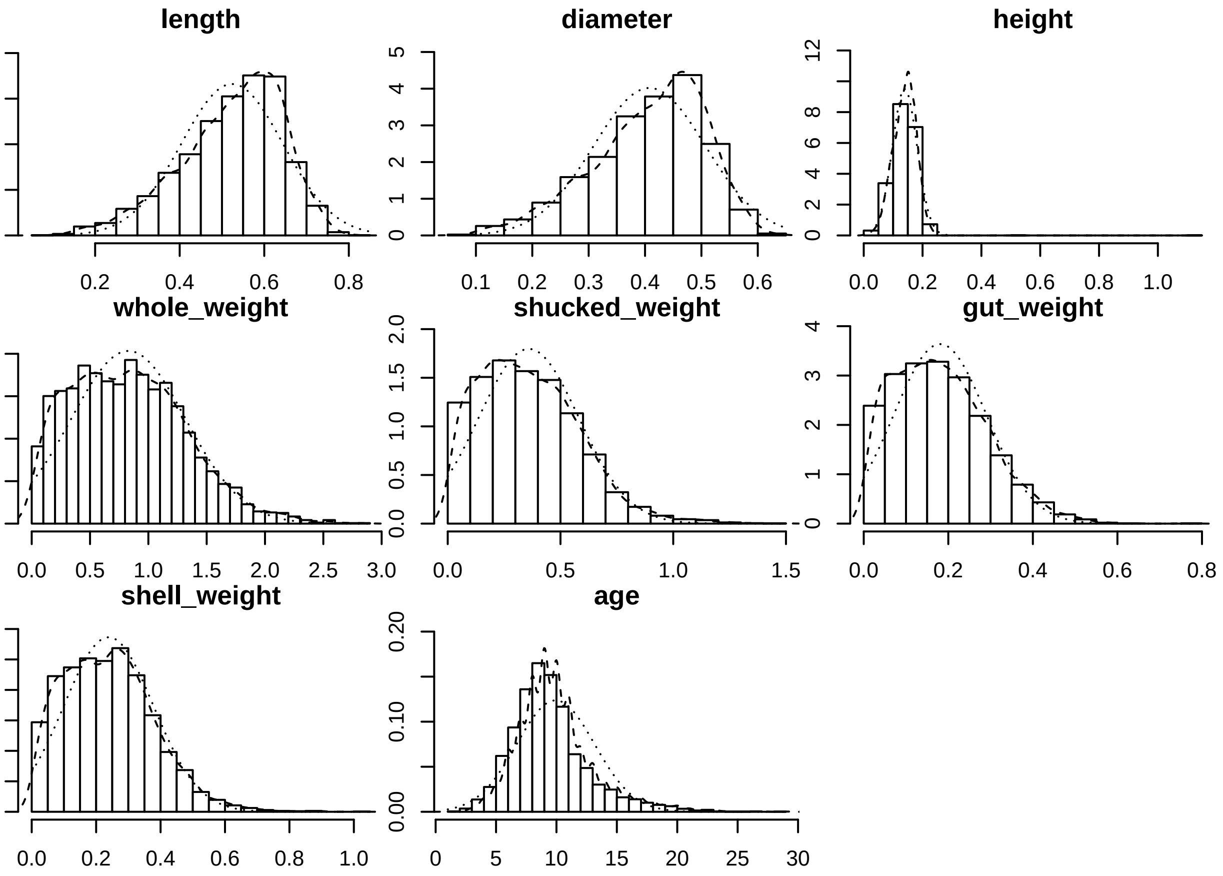
```
# Load data
# Dataset and metadata can be found at: https://archive.ics.uci.edu/ml/datasets/Abalone
# Abalone is a common name for any of a group of small to
# very large sea snails, marine gastropod molluscs in the family Haliotidae

# Objective: Predicting the age of abalone from physical measurements

abalone=read.csv("abalone.csv")

#Modify the data to create a subset of just mature abalones (Male and Female)
abalone_mature=subset(abalone, sex=="M" | sex=="F")

#Modify the data to create a subset of male abalones
abalone_male=subset(abalone, sex=="M")

#Modify the data to create a subset of female abalones
abalone_female=subset(abalone, sex=="F")
################################################################
# Exploratory data analysis
#### Descriptive Statistcs #####
summary(abalone)
```

```
##  sex           length          diameter          height        whole_weight
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415
##  M:1528   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395   Mean   :0.8287
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300   Max.   :2.8255
##  shucked_weight     gut_weight      shell_weight         age
##  Min.   :0.0010   Min.   :0.0005   Min.   :0.0015   Min.   : 1.000  
##  1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300   1st Qu.: 8.000  
##  Median :0.3360   Median :0.1710   Median :0.2340   Median : 9.000  
##  Mean   :0.3594   Mean   :0.1806   Mean   :0.2388   Mean   : 9.934  
##  3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290   3rd Qu.:11.000  
##  Max.   :1.4880   Max.   :0.7600   Max.   :1.0050   Max.   :29.000
```

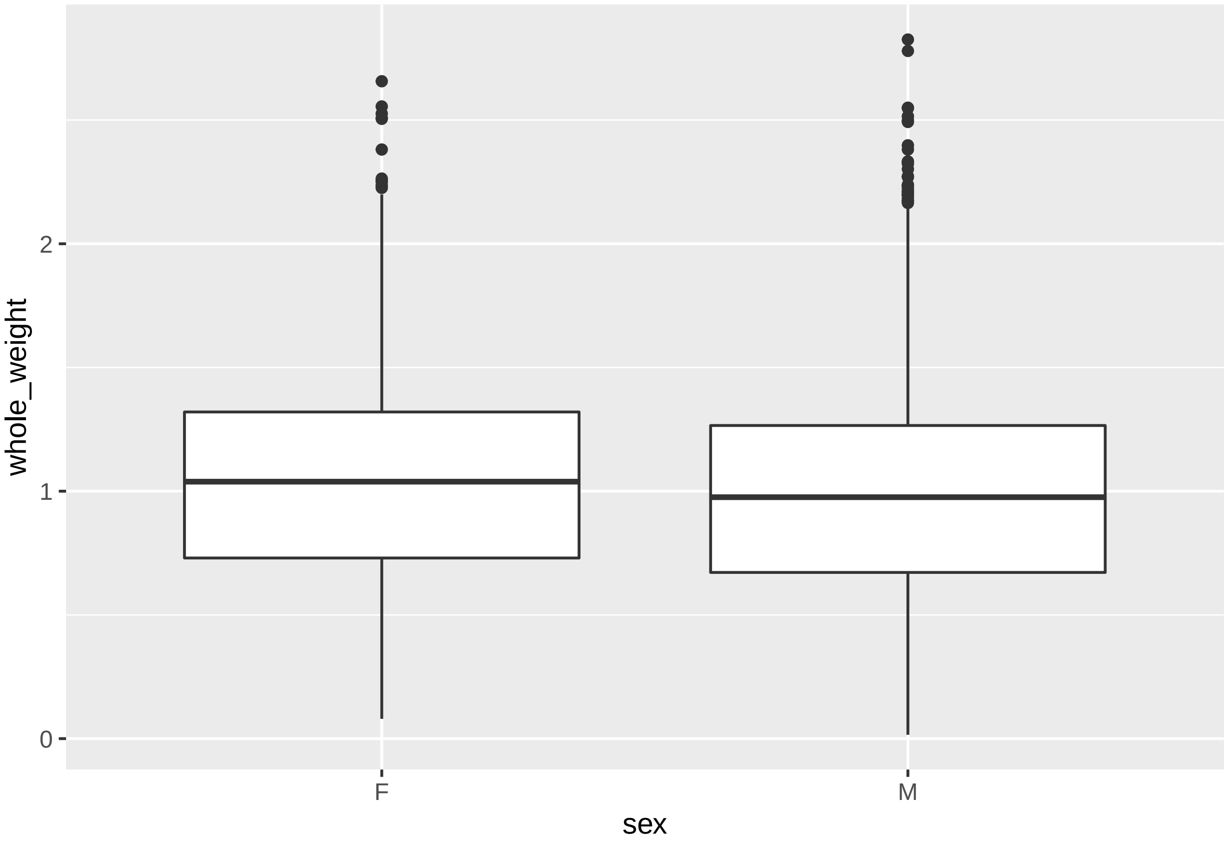
```
# no missing data
describe(abalone)
```

```
##                vars    n mean   sd median trimmed  mad  min   max range  skew
## sex*              1 4177 2.05 0.82   2.00    2.07 1.48 1.00  3.00  2.00 -0.10
## length            2 4177 0.52 0.12   0.55    0.53 0.12 0.08  0.82  0.74 -0.64
## diameter          3 4177 0.41 0.10   0.42    0.41 0.10 0.06  0.65  0.60 -0.61
## height            4 4177 0.14 0.04   0.14    0.14 0.04 0.00  1.13  1.13  3.13
## whole_weight      5 4177 0.83 0.49   0.80    0.80 0.53 0.00  2.83  2.82  0.53
## shucked_weight    6 4177 0.36 0.22   0.34    0.34 0.23 0.00  1.49  1.49  0.72
## gut_weight        7 4177 0.18 0.11   0.17    0.17 0.12 0.00  0.76  0.76  0.59
## shell_weight      8 4177 0.24 0.14   0.23    0.23 0.15 0.00  1.00  1.00  0.62
## age               9 4177 9.93 3.22   9.00    9.64 2.97 1.00 29.00 28.00  1.11
##                kurtosis   se
## sex*              -1.51 0.01
## length             0.06 0.00
## diameter          -0.05 0.00
## height            75.90 0.00
## whole_weight      -0.03 0.01
## shucked_weight     0.59 0.00
## gut_weight         0.08 0.00
## shell_weight       0.53 0.00
## age                2.32 0.05
```

```
# Make a histogram for all continuous variables
multi.hist(abalone[,2:9])
```



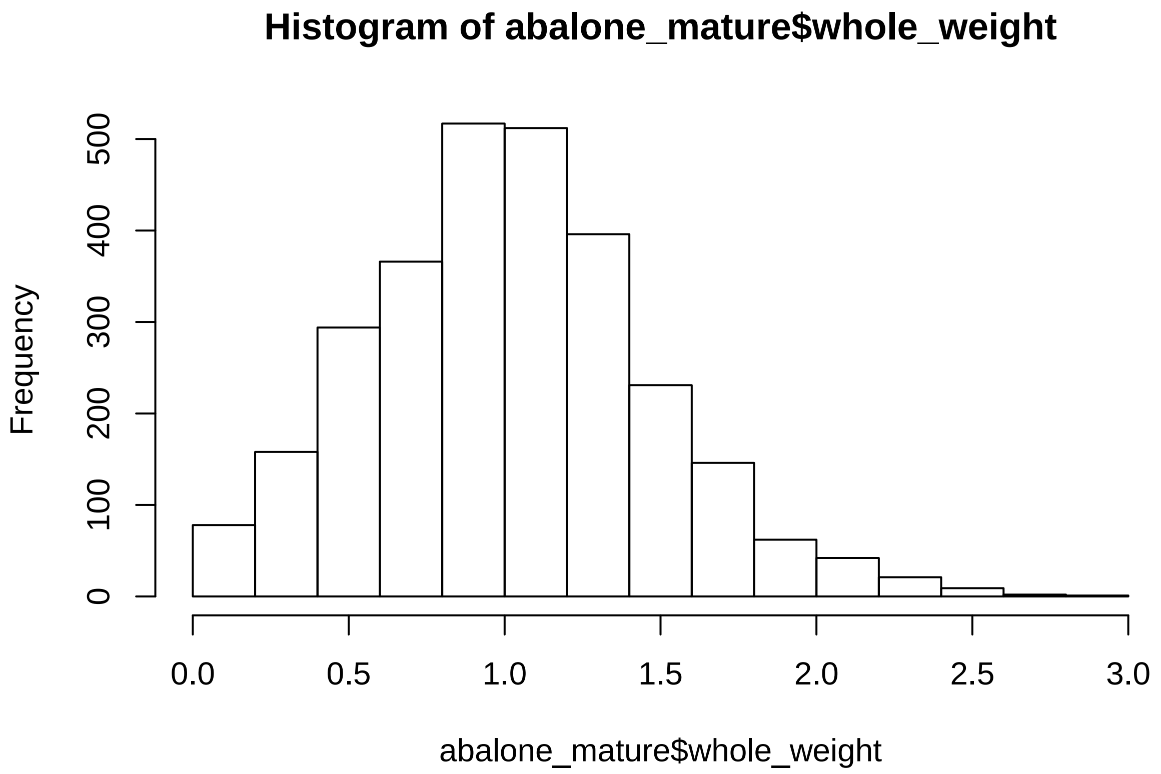
```
#### T-Test #####
# plot the data
plot1 = ggplot(abalone_mature, aes(x=sex, y=whole_weight))+
  geom_boxplot()
plot1
```



```
# Test assumptions
# test for normality of raw data
shapiro.test(abalone_mature$whole_weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  abalone_mature$whole_weight
## W = 0.98987, p-value = 2.637e-13
```

```
# fail- these test are highly influenced by n
hist(abalone_mature$whole_weight)
```



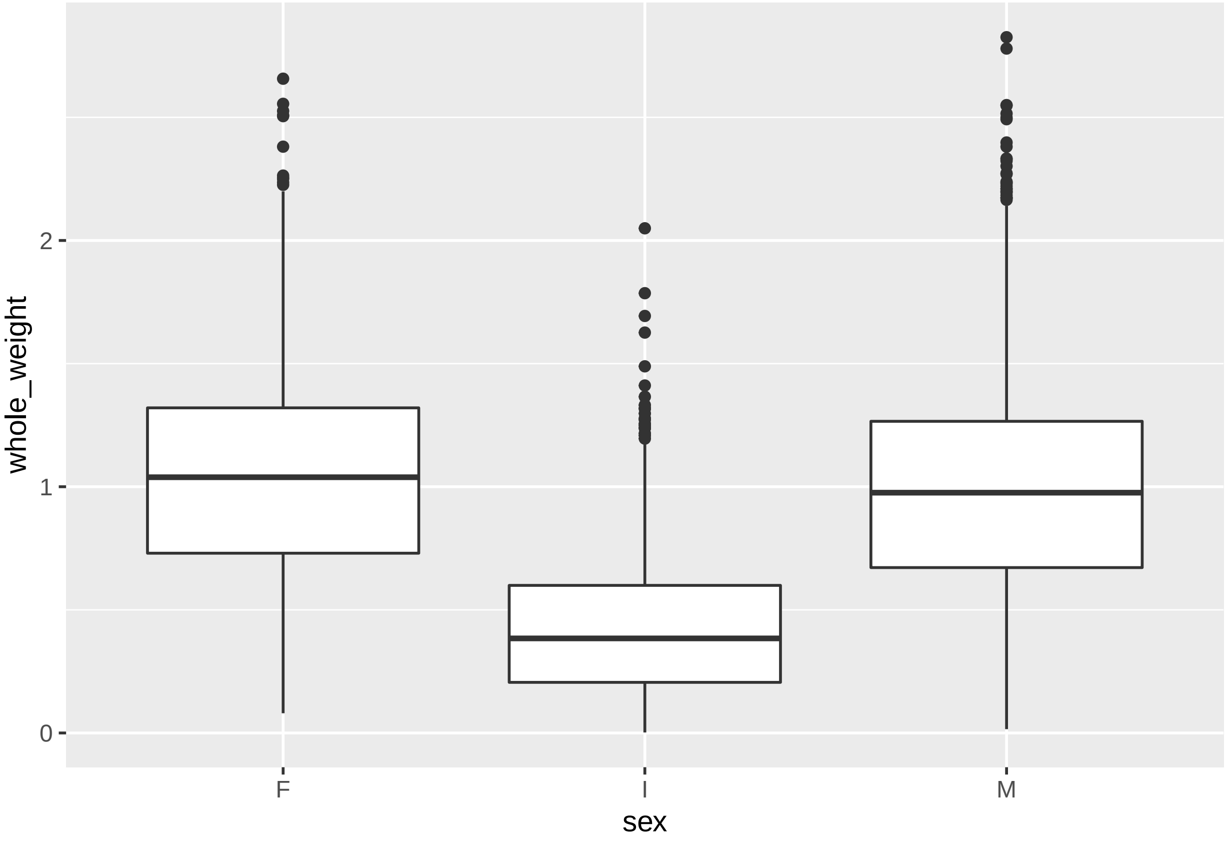
```
# not normal but ok

# test homogenity of variance
leveneTest(whole_weight ~ sex, data=abalone_mature)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value  Pr(>F)
## group    1    5.12 0.02373 *
##       2833
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

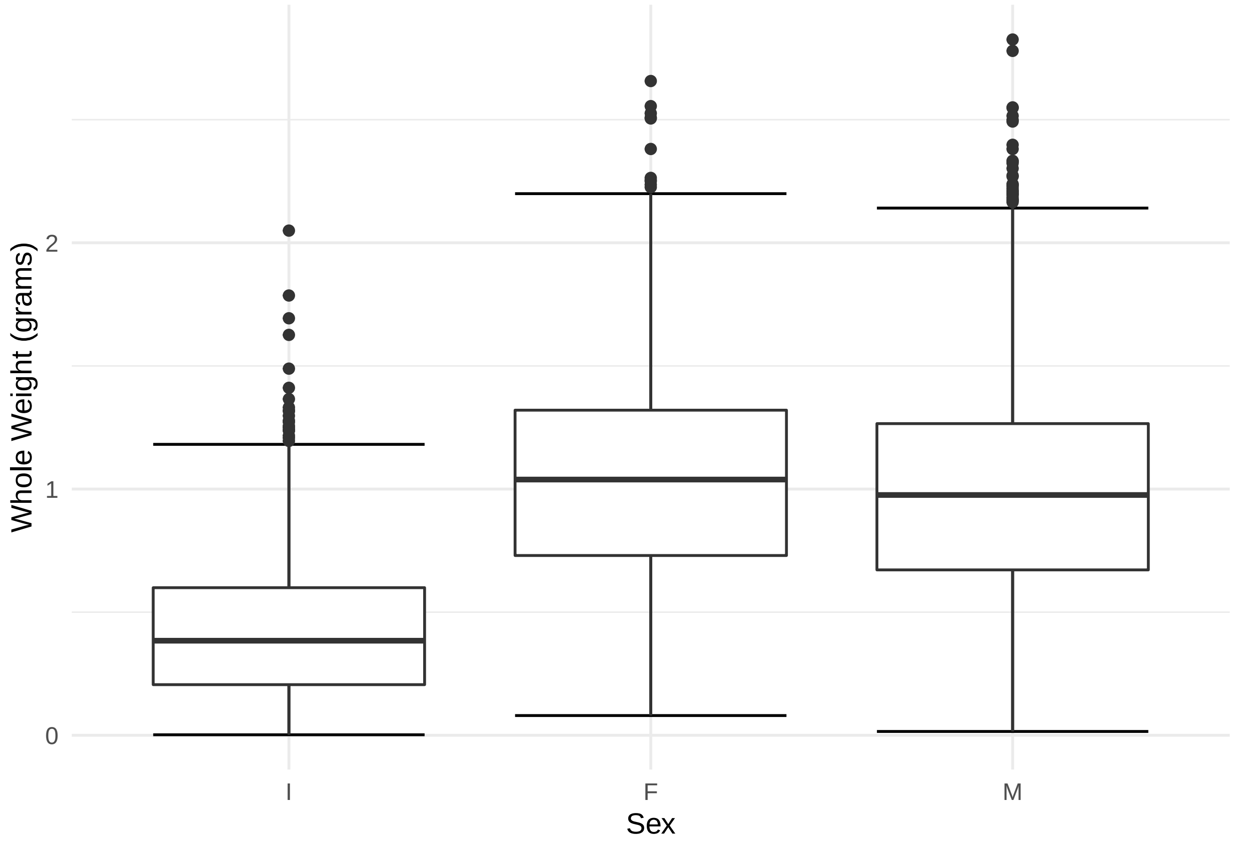
```
# the variance is not homogenous between the two groups- we must we the Welch's two sample t-test
# This is the default

# Run the T-test
t.test(data=abalone_mature,whole_weight~sex)
```

```
##
##  Welch Two Sample t-test
##
## data:  whole_weight by sex
## t = 3.2531, df = 2820.4, p-value = 0.001155
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.02187753 0.08826789
## sample estimates:
## mean in group F mean in group M
##       1.0465321       0.9914594
```

```
# There is a difference of 0.05g between male and females. On average this is 5.4% higher (difference/m
# While this is statistically significant it may not be biologically significant.

#### ANOVA ####
# plot the data
plot2 = ggplot(abalone, aes(x=sex, y=whole_weight))+
  geom_boxplot()
plot2
```



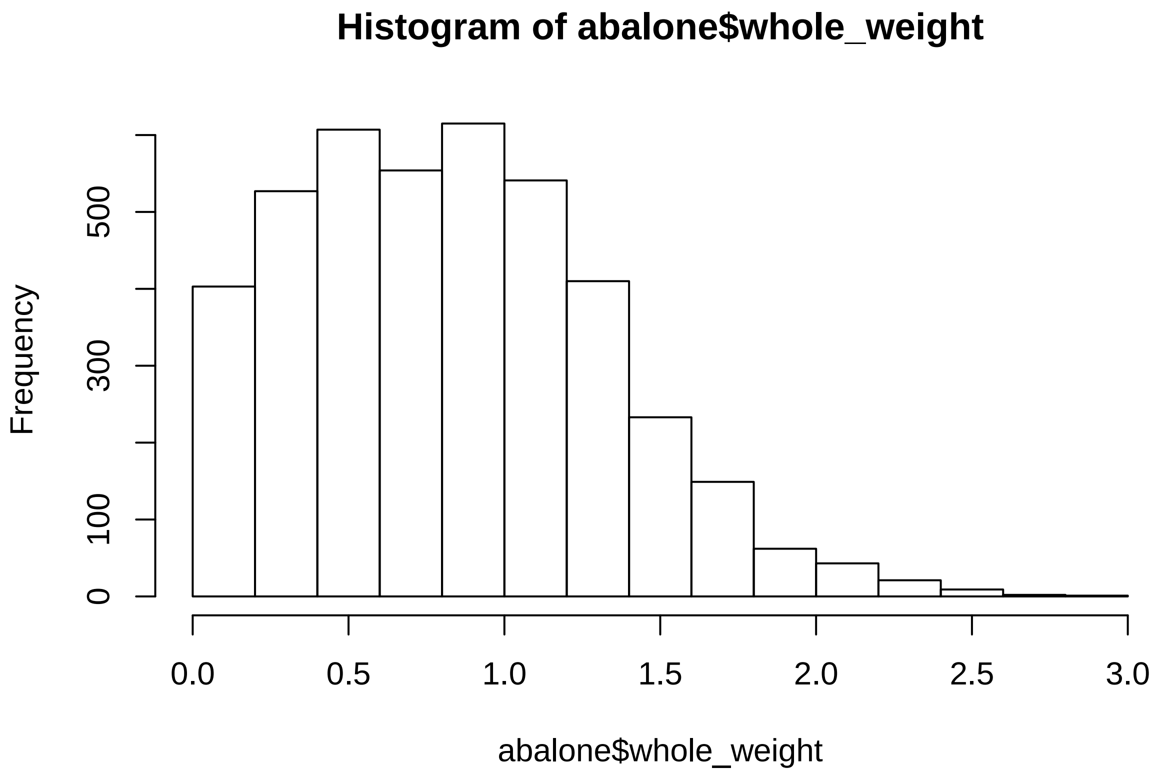
```
# Reorder factors
abalone$sex_order = factor(abalone$sex, levels = c("I", "F", "M"))

plot3 = ggplot(abalone, aes(x=sex_order, y=whole_weight))+
  stat_boxplot(geom ='errorbar') +
  geom_boxplot()+
  # add error bars to the plot
  xlab("Sex")+
  ylab("Whole Weight (grams)")+
  theme_minimal()
plot3
```



```
# Test assumptions
# test for normality of raw data
shapiro.test(abalone$whole_weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  abalone$whole_weight
## W = 0.97228, p-value < 2.2e-16
```

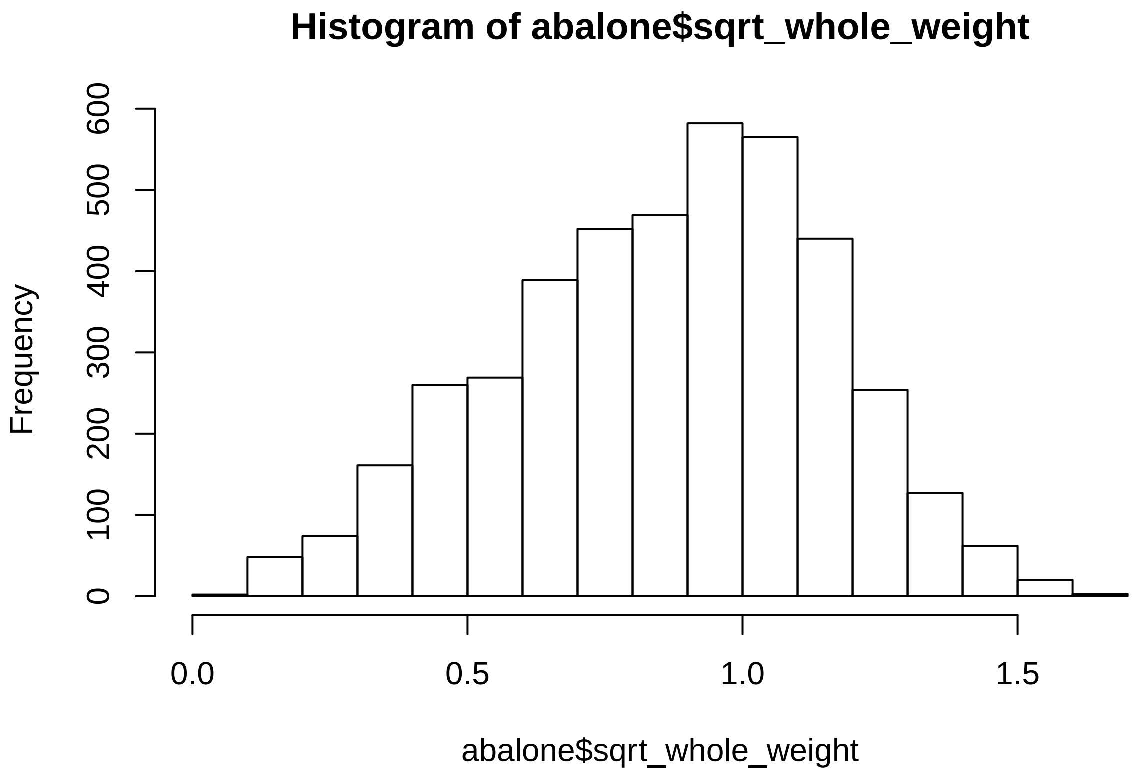
```
# fail- these test are highly influenced by n
hist(abalone$whole_weight)
```



```
# not normal but ok

# sqrt the variable
abalone$sqrt_whole_weight = sqrt(abalone$whole_weight)
# test for normality of raw data
shapiro.test(abalone$sqrt_whole_weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  abalone$sqrt_whole_weight
## W = 0.99041, p-value = 3.23e-16
```

```
# fail- these test are highly influenced by n
hist(abalone$sqrt_whole_weight)
```



```
# not normal but histogram looks better


# test homogenity of variance
leveneTest(sqrt_whole_weight ~ sex, data=abalone)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##         Df F value    Pr(>F)
## group    2  8.2212 0.0002733 ***
##       4174
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

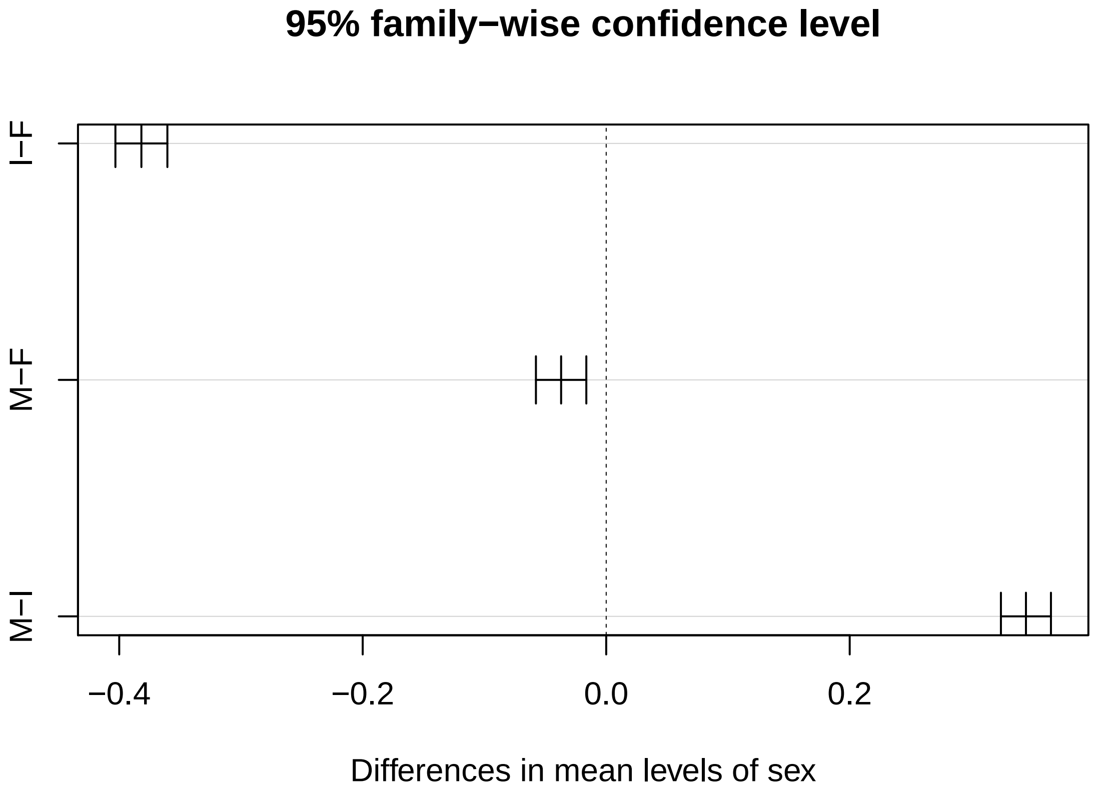
```
# the variance is not homogenous between the three groups
# This is the default


# This is a typeI anova- testing between groups
anova1=anova(lm(sqrt_whole_weight~sex, data=abalone))
anova1
```

```
## Analysis of Variance Table
##
## Response: sqrt_whole_weight
##             Df Sum Sq Mean Sq F value    Pr(>F)
## sex          2 120.21  60.107    1095 < 2.2e-16 ***
## Residuals 4174 229.12   0.055
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# p-value is low so we reject the H0, there is a difference between the groups
# need to follow this up with a Tukey's post-hoc test
Tukey1= TukeyHSD(aov(sqrt_whole_weight~sex, data=abalone))
Tukey1
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = sqrt_whole_weight ~ sex, data = abalone)
##
## $sex
##            diff        lwr         upr   p adj
## I-F -0.38178468 -0.4031319 -0.36043749 0.0e+00
## M-F -0.03702239 -0.0577186 -0.01632618 8.3e-05
## M-I  0.34476230  0.3242121  0.36531253 0.0e+00
```

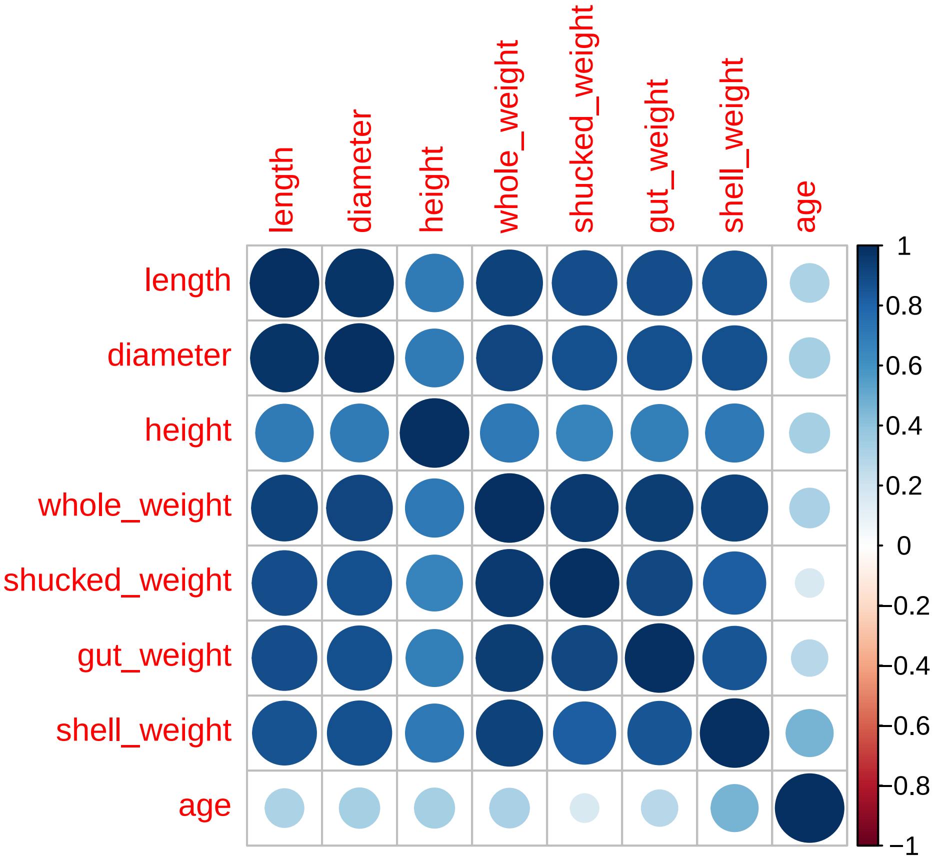
```
# I is lower than male and female- biologically this makes sense
# M is lower F - same results as above

# plot the differences
plot(Tukey1)
```



```
#### Correlation #####
cor=cor(abalone_mature[,2:9])
cor
```

```
##                   length  diameter    height whole_weight shucked_weight
## length         1.0000000 0.9780172 0.7003917    0.9217383      0.8866157
## diameter       0.9780172 1.0000000 0.7087055    0.9171707      0.8737180
## height         0.7003917 0.7087055 1.0000000    0.7167916      0.6606406
## whole_weight   0.9217383 0.9171707 0.7167916    1.0000000      0.9561181
## shucked_weight 0.8866157 0.8737180 0.6606406    0.9561181      1.0000000
## gut_weight     0.8885901 0.8771016 0.6867432    0.9473267      0.9007912
## shell_weight   0.8672809 0.8771149 0.7146359    0.9298901      0.8222420
## age            0.3117605 0.3393996 0.3349048    0.3275389      0.1677021
##                gut_weight shell_weight       age
## length          0.8885901    0.8672809 0.3117605
## diameter        0.8771016    0.8771149 0.3393996
## height          0.6867432    0.7146359 0.3349048
## whole_weight    0.9473267    0.9298901 0.3275389
## shucked_weight  0.9007912    0.8222420 0.1677021
## gut_weight      1.0000000    0.8543508 0.2752091
## shell_weight    0.8543508    1.0000000 0.4655449
## age             0.2752091    0.4655449 1.0000000
```

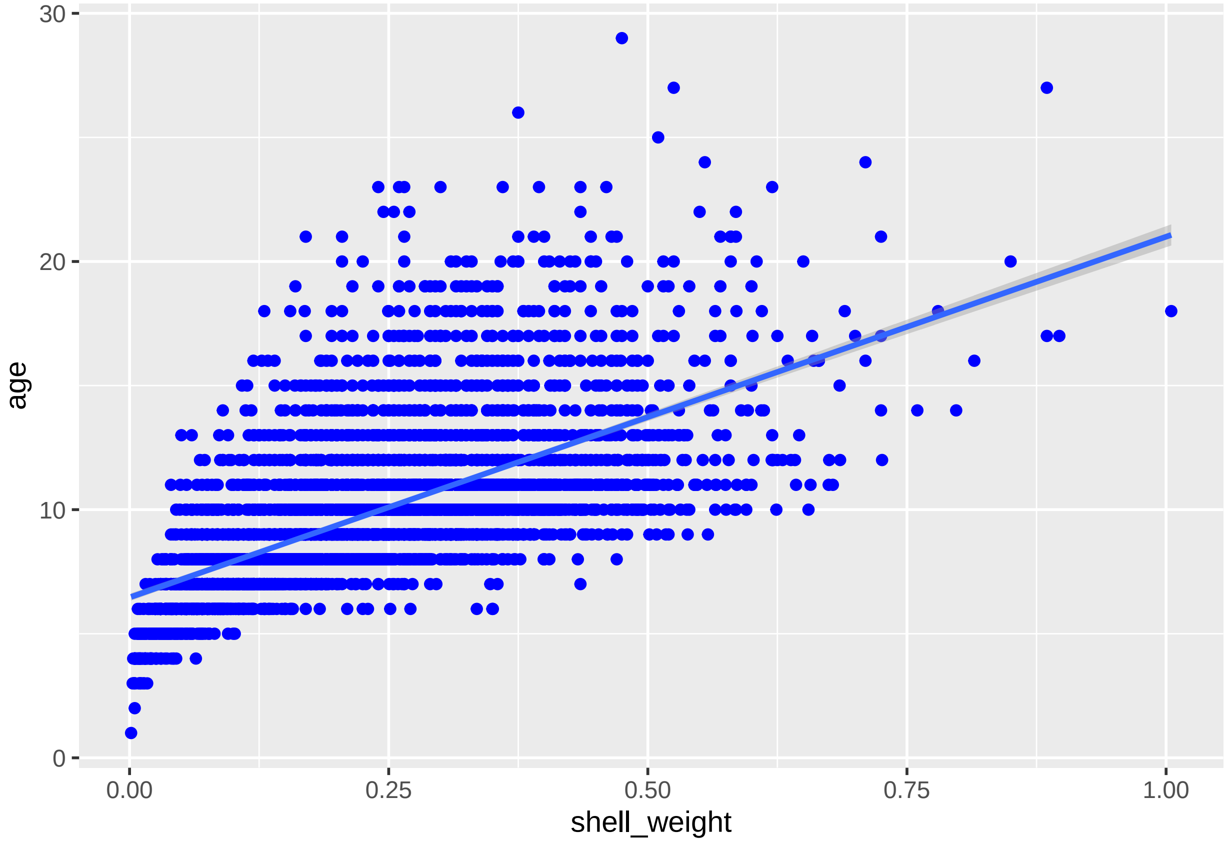
```
# visualize the correlations using corrplot
# https://cran.r-project.org/web/packages/corrplot/vignettes/corrplot-intro.html
corrplot(cor)
```



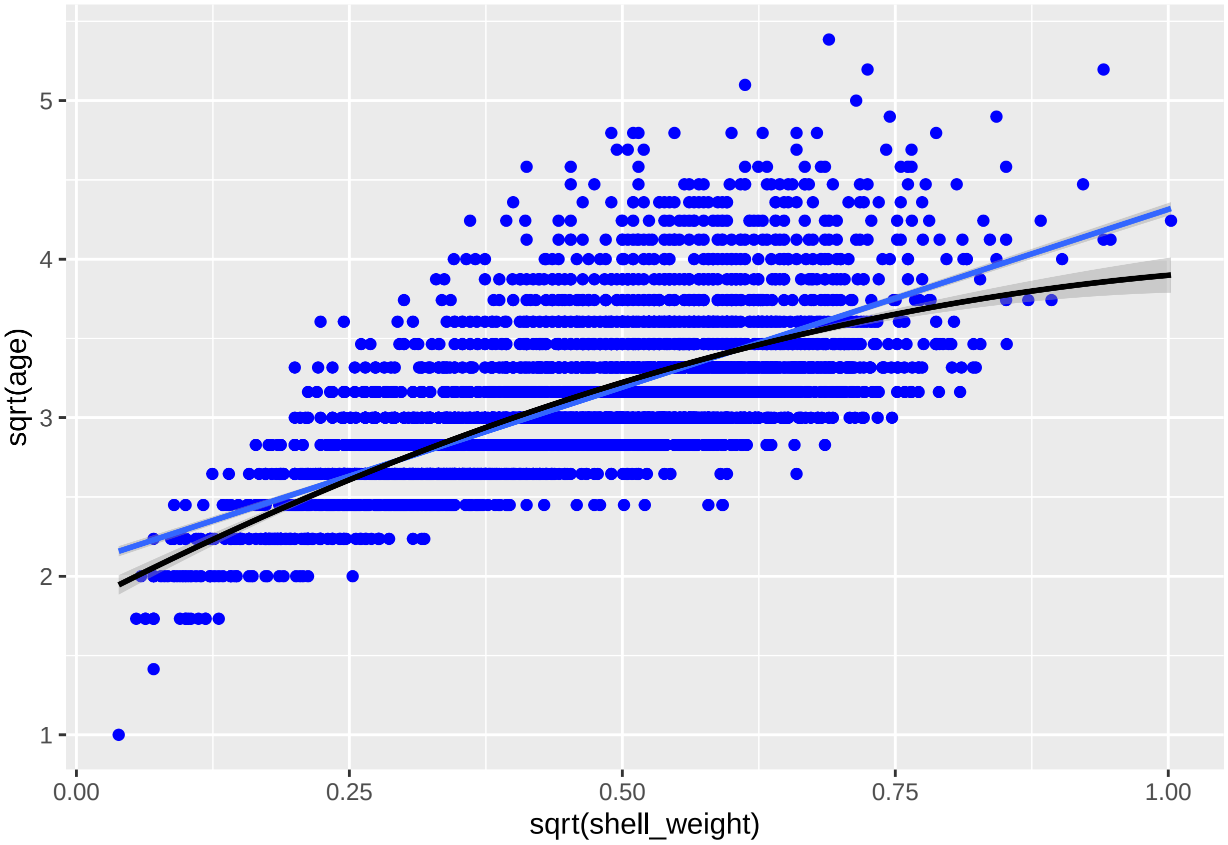
```
# testing single correlation
?cor.test
cor.test(abalone$age, abalone$shell_weight)
```

```
##
##  Pearson's product-moment correlation
##
## data:  abalone$age and abalone$shell_weight
## t = 52.084, df = 4175, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.6088342 0.6456138
## sample estimates:
##      cor
## 0.627574
```

```
#### Linear Regression ####
# Make a plot
ggplot(data = abalone, aes(x = shell_weight, y = age)) +
  geom_point(color='blue') +
  geom_smooth(method = "lm", se = TRUE)
```



```
# Make a plot
ggplot(data = abalone, aes(x = sqrt(shell_weight), y = sqrt(age))) +
  geom_point(color='blue') +
  geom_smooth(method = "lm", se = TRUE)+
  stat_smooth(method = "lm", formula = y ~ poly(x, 2), size = 1, color="black")
```



```
# Linear model
lm1=lm(age~shell_weight, data=abalone)
summary(lm1)
```

```
##
## Call:
## lm(formula = age ~ shell_weight, data = abalone)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9830 -1.6005 -0.5843  0.9390 15.6334
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.46212    0.07715   83.76   <2e-16 ***
## shell_weight 14.53568    0.27908   52.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.51 on 4175 degrees of freedom
## Multiple R-squared:  0.3938, Adjusted R-squared:  0.3937
## F-statistic:  2713 on 1 and 4175 DF,  p-value: < 2.2e-16
```

```
# test assumptions on RESIDUALS
resettest(lm1) # fail
```

```
##
##  RESET test
```

```
##
## data:  lm1
## RESET = 92.293, df1 = 2, df2 = 4173, p-value < 2.2e-16
```

```
dwtest(lm1) # fail
```

```
##
##  Durbin-Watson test
##
## data:  lm1
## DW = 1.0157, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

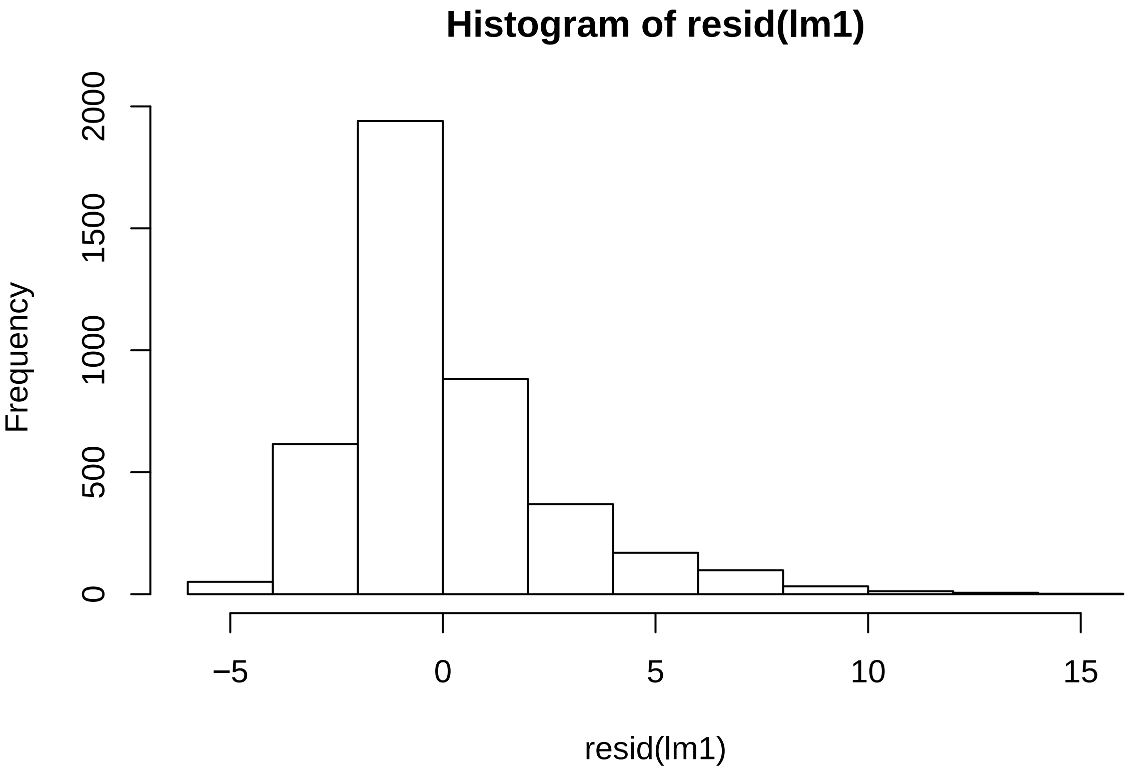
```
bptest(lm1) # fail
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm1
## BP = 125.07, df = 1, p-value < 2.2e-16
```

```
shapiro.test(resid(lm1)) # fail
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(lm1)
## W = 0.89287, p-value < 2.2e-16
```

```
hist(resid(lm1))
```



```
lm2=lm(sqrt(age)~sqrt(shell_weight), data=abalone)
summary(lm1)
```

```
##
## Call:
## lm(formula = age ~ shell_weight, data = abalone)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9830 -1.6005 -0.5843  0.9390 15.6334
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.46212    0.07715   83.76   <2e-16 ***
## shell_weight 14.53568    0.27908   52.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.51 on 4175 degrees of freedom
## Multiple R-squared:  0.3938, Adjusted R-squared:  0.3937
## F-statistic:  2713 on 1 and 4175 DF,  p-value: < 2.2e-16
```

```r
# test assumptions on RESIDUALS
resettest(lm2) # fail
```

```
##
##  RESET test
##
## data:  lm2
## RESET = 49.913, df1 = 2, df2 = 4173, p-value < 2.2e-16
```

```r
dwtest(lm2) # fail
```

```
##
##  Durbin-Watson test
##
## data:  lm2
## DW = 0.97512, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

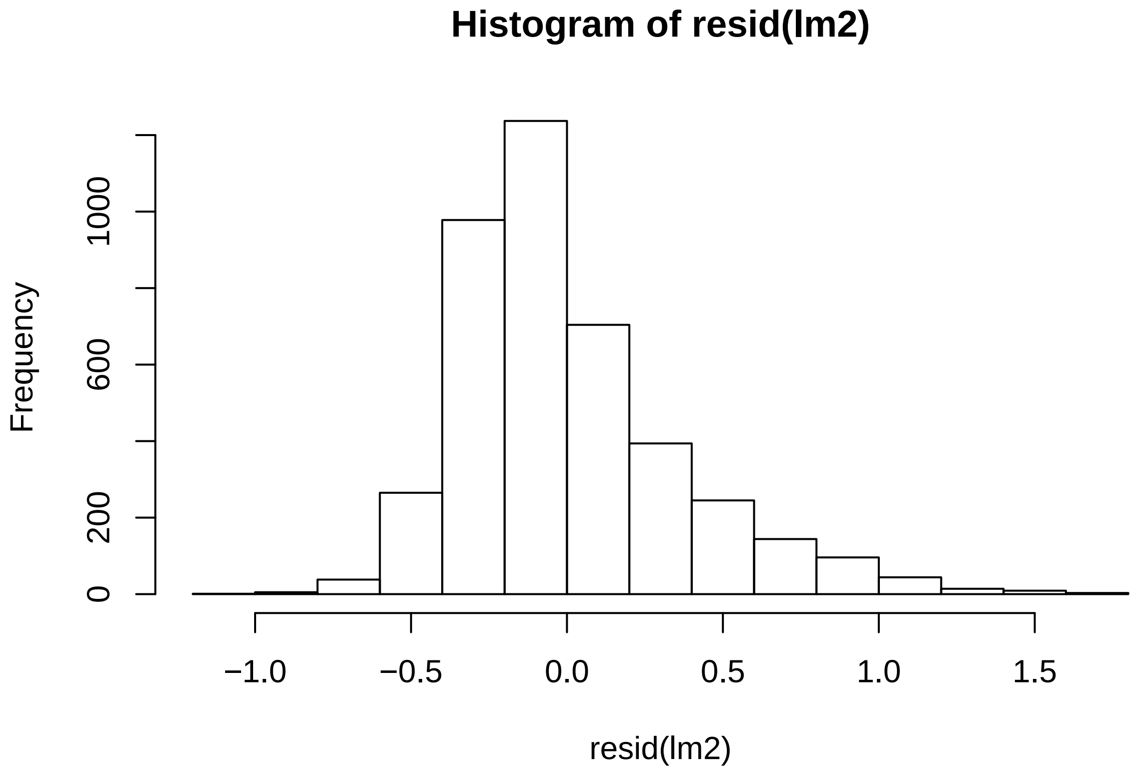
```r
bptest(lm2) # fail
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm2
## BP = 71.872, df = 1, p-value < 2.2e-16
```

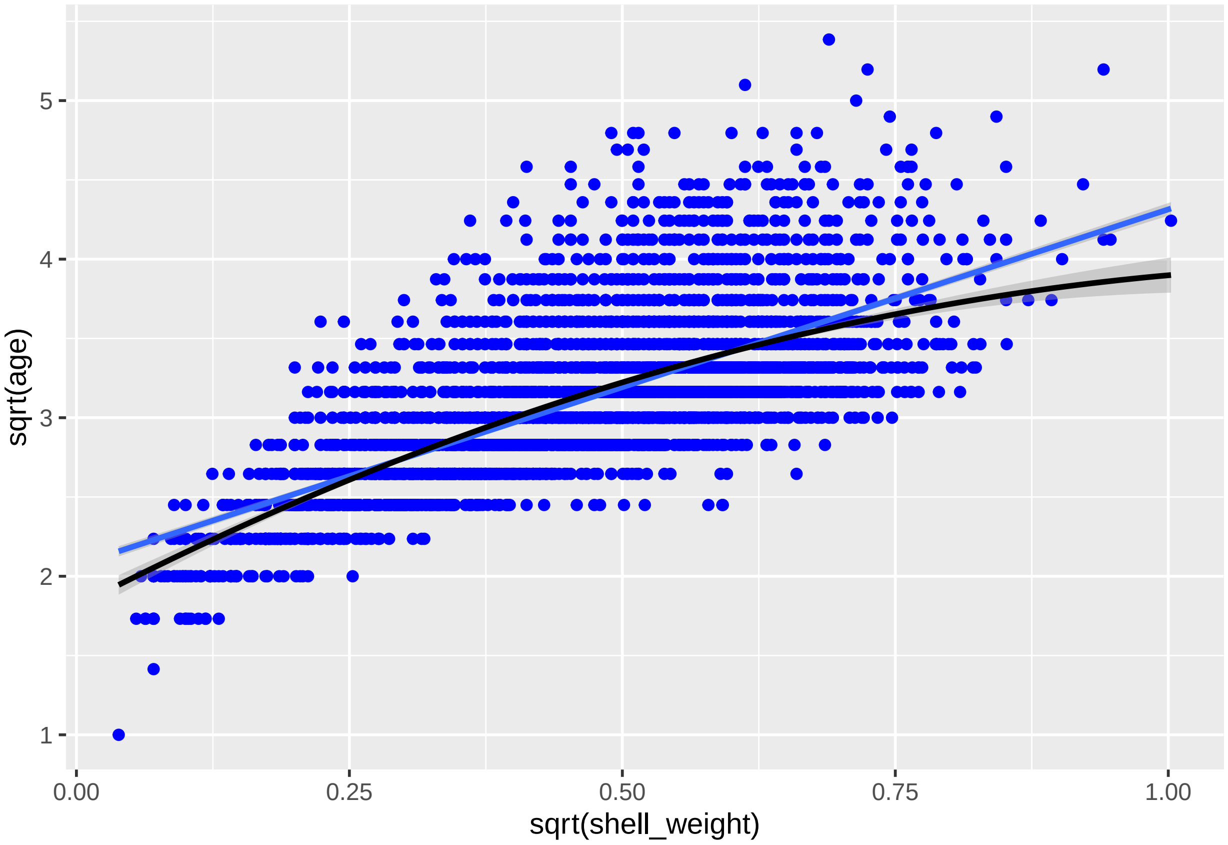
```r
shapiro.test(resid(lm2)) # fail
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(lm2)
## W = 0.92985, p-value < 2.2e-16
```

```r
hist(resid(lm2))
```



```r
# still all fail but are better

# Maybe a non-linear curve would be a better fit?
# Make a plot
ggplot(data = abalone, aes(x = sqrt(shell_weight), y = sqrt(age))) +
  geom_point(color='blue') +
  geom_smooth(method = "lm", se = TRUE)+
  stat_smooth(method = "lm", formula = y ~ poly(x, 2), size = 1, color="black")
```



```
lm2.1=lm(sqrt(age)~poly(sqrt(shell_weight),2), data=abalone)
summary(lm2.1)
```

```
##
## Call:
## lm(formula = sqrt(age) ~ poly(sqrt(shell_weight), 2), data = abalone)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.95295 -0.23976 -0.08011  0.16321  1.82007
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    3.112643   0.005518 564.112  < 2e-16 ***
## poly(sqrt(shell_weight), 2)1 22.024802   0.356612  61.761  < 2e-16 ***
## poly(sqrt(shell_weight), 2)2 -2.832547   0.356612  -7.943 2.52e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3566 on 4174 degrees of freedom
## Multiple R-squared:  0.4816, Adjusted R-squared:  0.4813
## F-statistic:  1939 on 2 and 4174 DF,  p-value: < 2.2e-16
```

```
# test assumptions on RESIDUALS
resettest(lm2.1) # fail
```

```
##
##  RESET test
```

```
##
## data:  lm2.1
## RESET = 18.015, df1 = 2, df2 = 4172, p-value = 1.622e-08
```

```r
dwtest(lm2.1) # fail
```

```
##
##  Durbin-Watson test
##
## data:  lm2.1
## DW = 0.96582, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

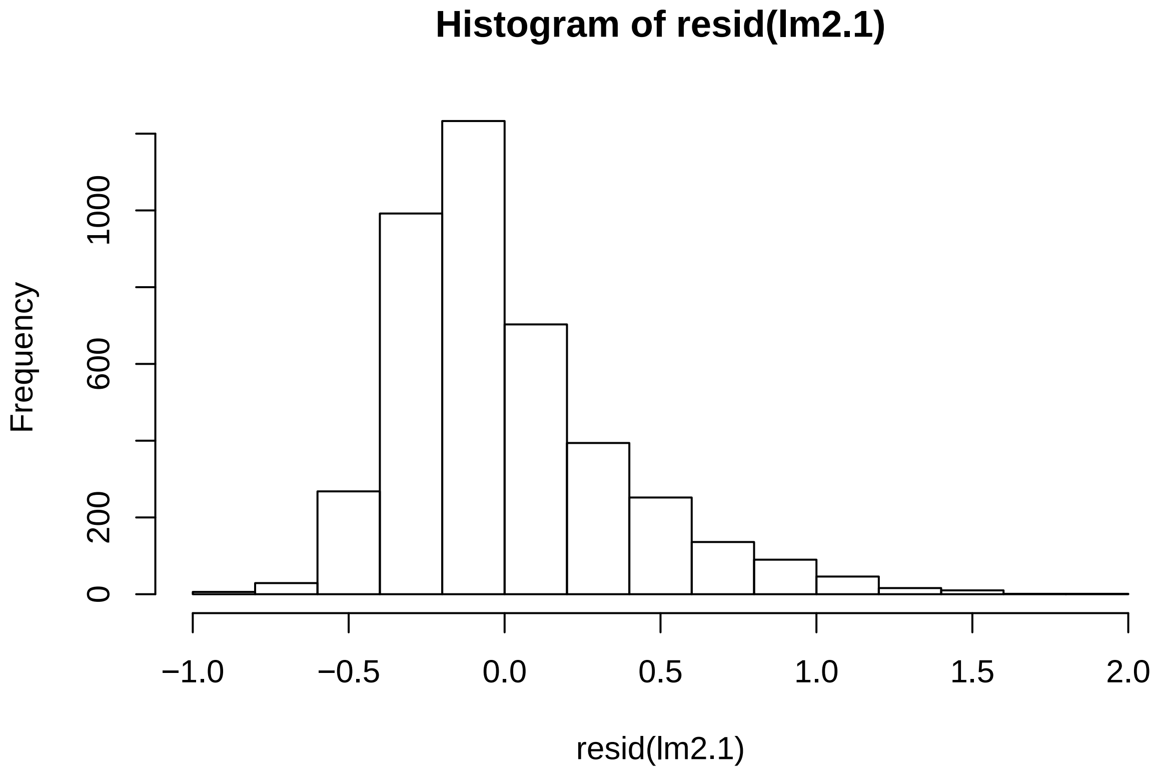
```r
bptest(lm2.1) # fail
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm2.1
## BP = 94.574, df = 2, p-value < 2.2e-16
```

```r
shapiro.test(resid(lm2.1)) # fail
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(lm2.1)
## W = 0.92516, p-value < 2.2e-16
```

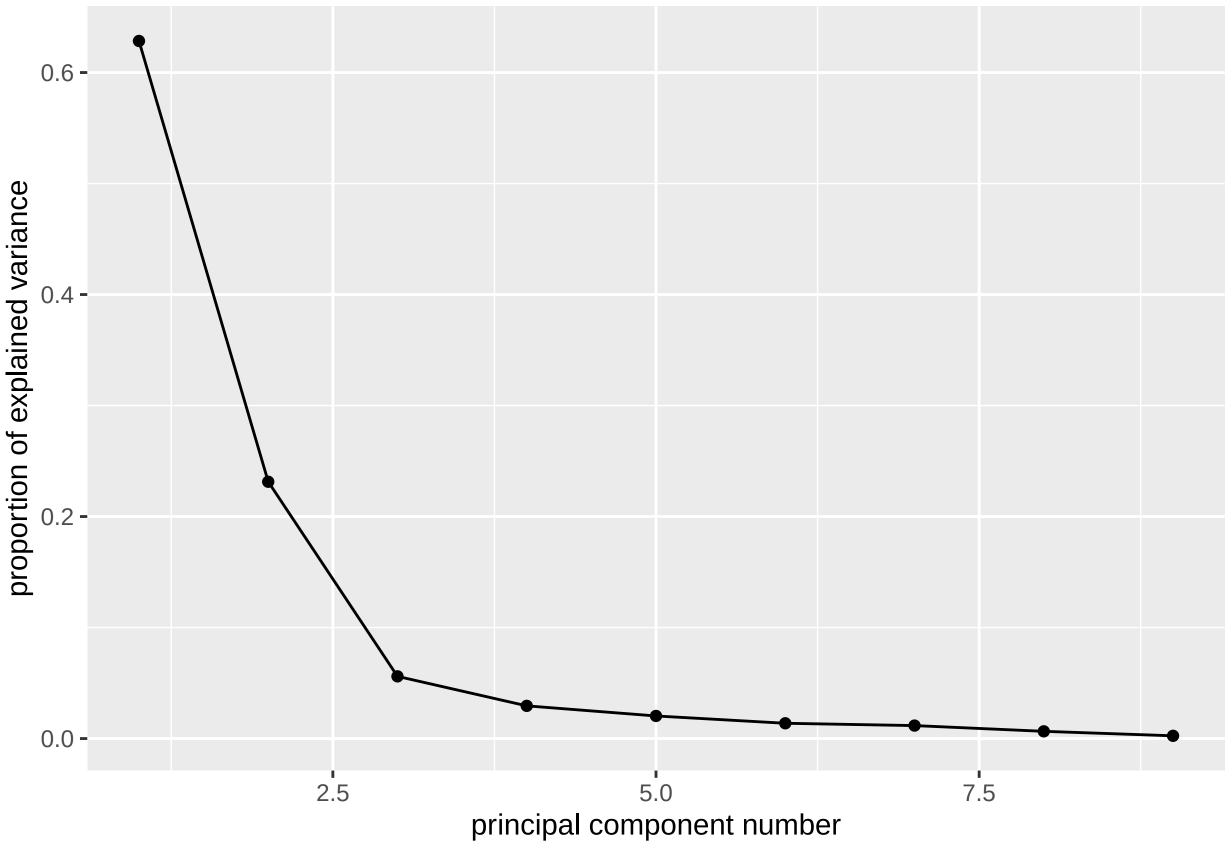
```r
hist(resid(lm2.1))
```



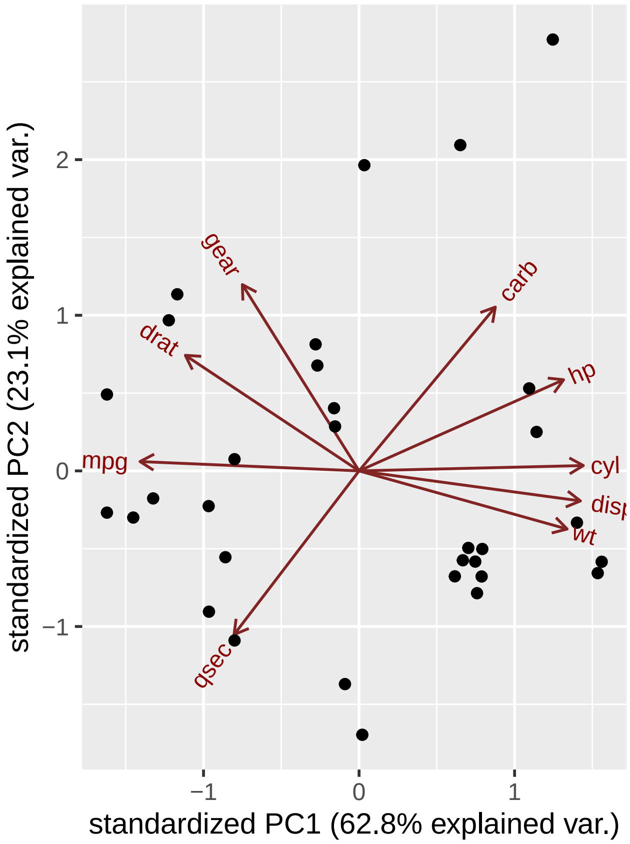
```r
# still all fail
# could we make this a multivariate regression?


#### PCA ####
# Example 1
mtcars.pca <- prcomp(mtcars[,c(1:7,10,11)], center = TRUE,scale. = TRUE)

#Variance explained
summary(mtcars.pca)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     2.3782 1.4429 0.71008 0.51481 0.42797 0.35184 0.32413
## Proportion of Variance 0.6284 0.2313 0.05602 0.02945 0.02035 0.01375 0.01167
## Cumulative Proportion  0.6284 0.8598 0.91581 0.94525 0.96560 0.97936 0.99103
##                           PC8     PC9
## Standard deviation     0.2419 0.14896
## Proportion of Variance 0.0065 0.00247
## Cumulative Proportion  0.9975 1.00000
```

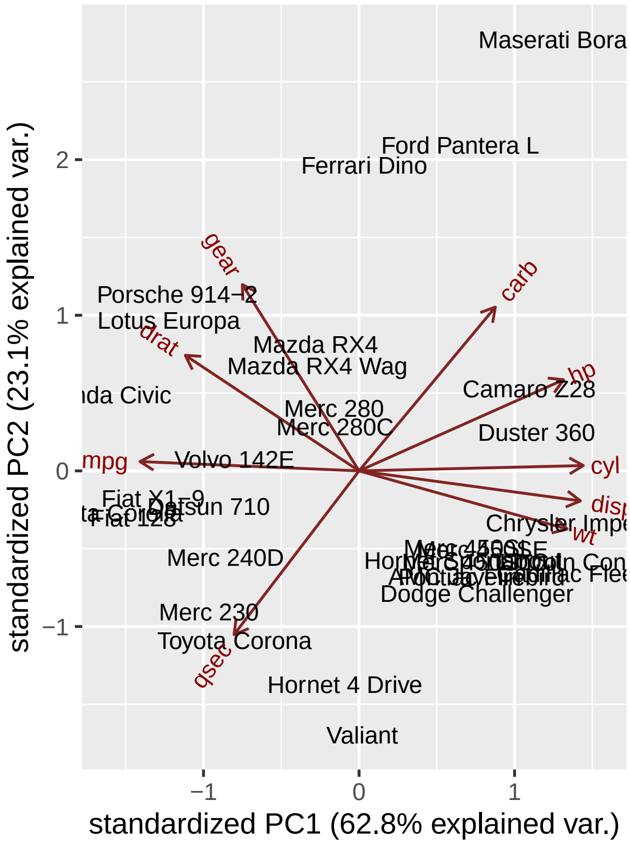
```
# scree plot
ggscreeplot(mtcars.pca)
```



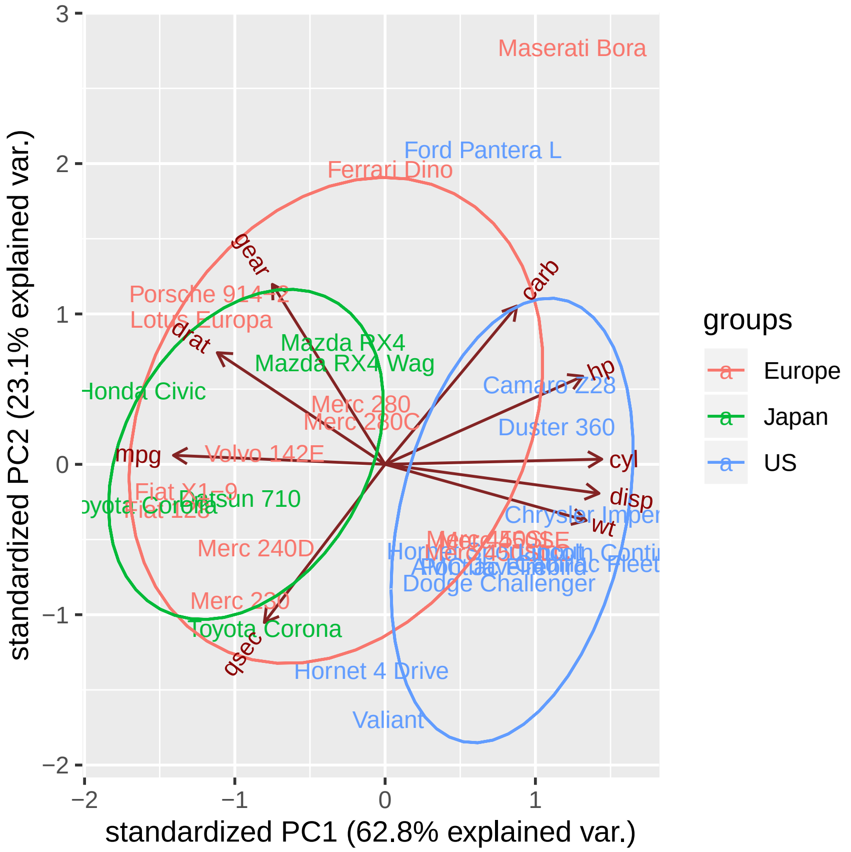
```
# Biplots
ggbiplot(mtcars.pca)
```



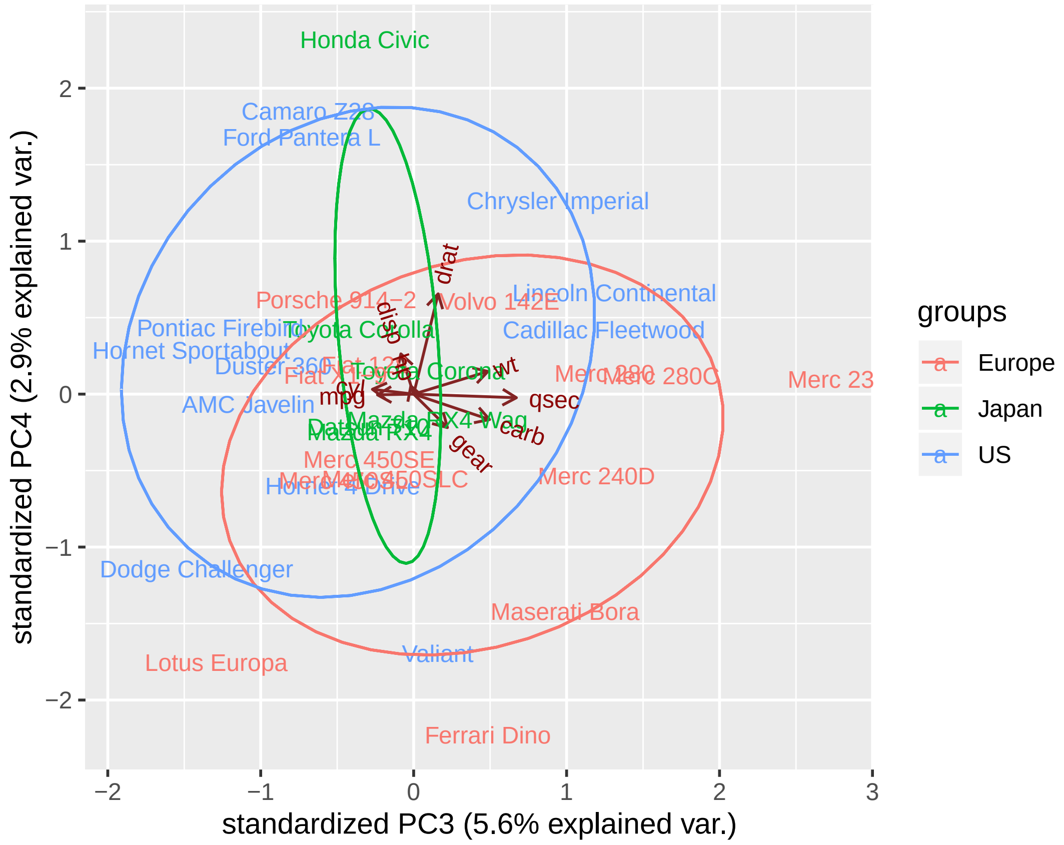
```r
ggbiplot(mtcars.pca, labels=rownames(mtcars))
```



```
mtcars.country <- c(rep("Japan", 3), rep("US",4),
                    rep("Europe", 7),rep("US",3), "Europe", rep("Japan", 3),
                    rep("US",4), rep("Europe", 3), "US", rep("Europe", 3))

# Biplot
ggbiplot(mtcars.pca,ellipse=TRUE,  labels=rownames(mtcars), groups=mtcars.country)
```



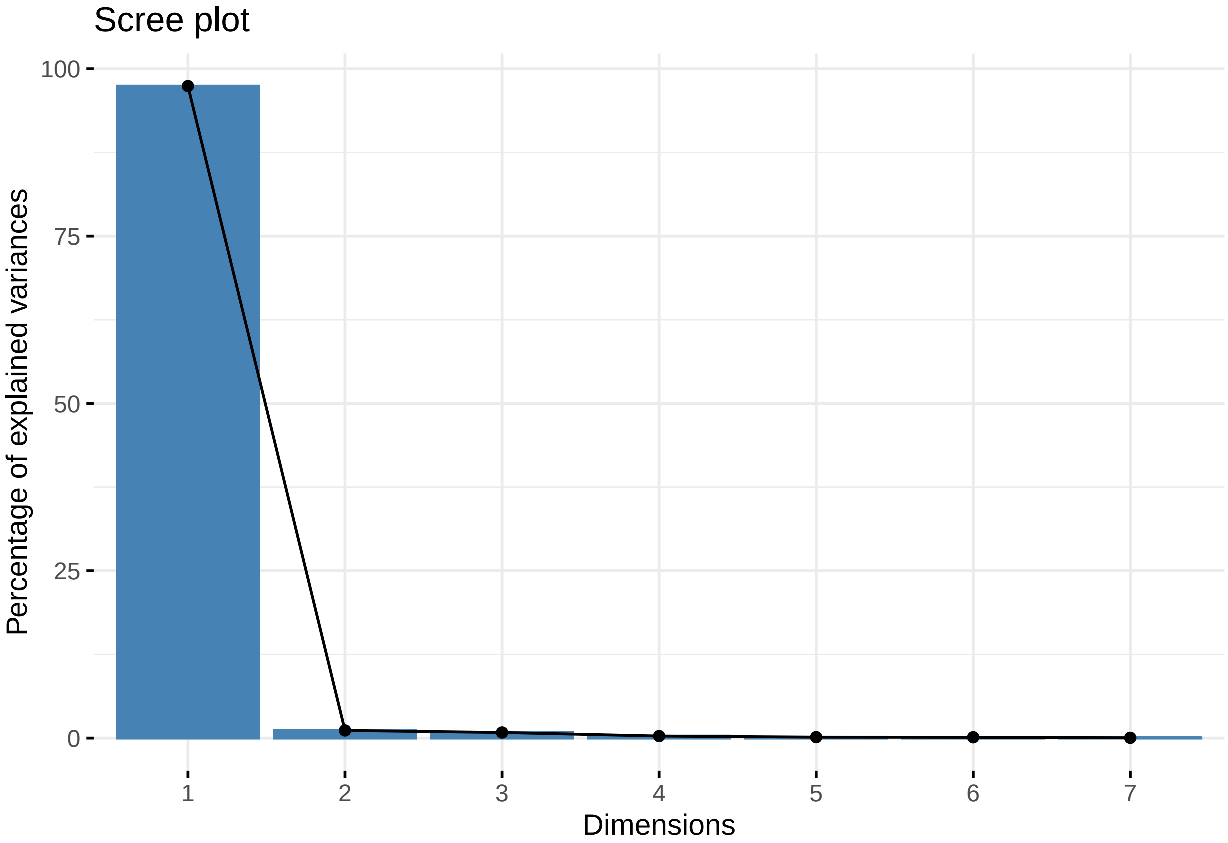
```
ggbiplot(mtcars.pca,ellipse=TRUE,choices=c(3,4),
         labels=rownames(mtcars), groups=mtcars.country)
```



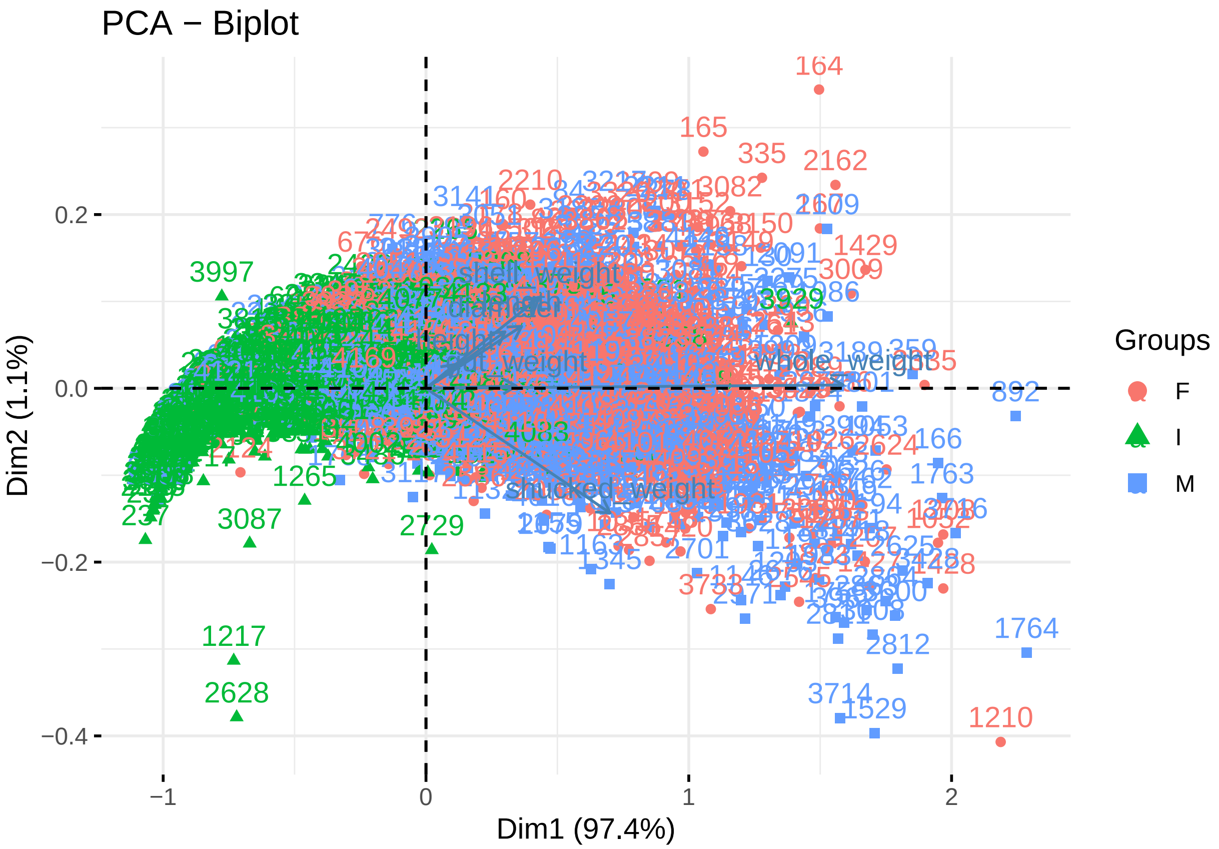
```
###
# Example 2 with abalone
pca1 = prcomp(abalone[,2:8], center = TRUE, scale. = FALSE)
pca1
```

```
## Standard deviations (1, .., p=7):
## [1] 0.58152448 0.06296055 0.05392322 0.03247929 0.02212835 0.02065884 0.01217135
##
## Rotation (n x k) = (7 x 7):
##                      PC1         PC2         PC3          PC4         PC5
## length         0.19315606  0.35006929 -0.65543596 -0.038784599  0.15584501
## diameter       0.15955208  0.31882074 -0.50547308  0.018060452  0.07483574
## height         0.05928271  0.13475175 -0.08607958  0.004683252 -0.92444847
## whole_weight   0.84261922  0.01882402  0.31147028 -0.127977156  0.16797945
## shucked_weight 0.37195895 -0.70343169 -0.33727250  0.353767145 -0.16244383
## gut_weight     0.18225102  0.01294771  0.02506135 -0.762977566 -0.20728245
## shell_weight   0.22834926  0.51216078  0.30999426  0.523911759 -0.13392483
##                         PC6          PC7
## length         -0.0005606153 -0.620285186
## diameter        0.0302034552  0.781379947
## height          0.3377048831 -0.047395498
## whole_weight    0.3846953125 -0.006247874
## shucked_weight -0.3184028855  0.012572505
## gut_weight     -0.5828809182  0.033732861
## shell_weight   -0.5439869513 -0.033321509
```

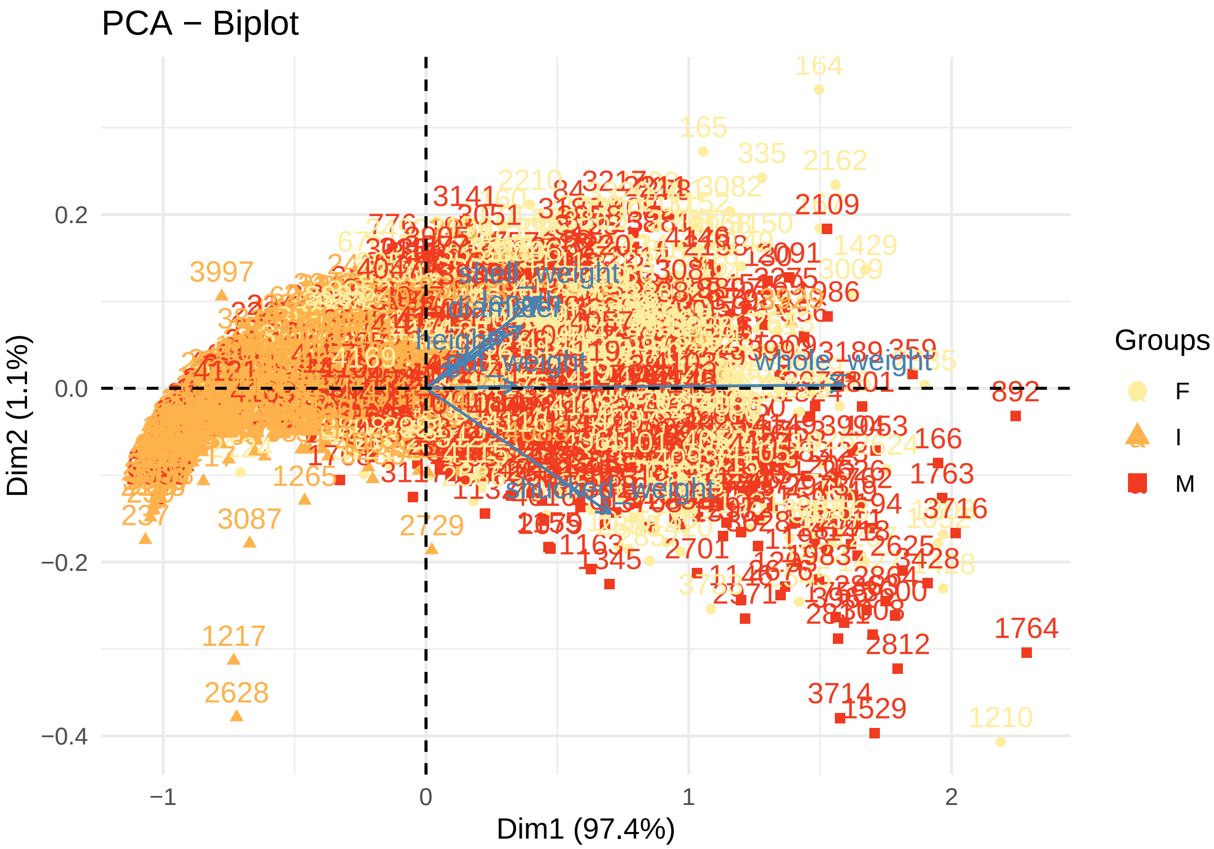
```r
# plots
fviz_eig(pca1)
```



```r
fviz_pca_biplot(pca1, habillage=abalone$sex)
```



```r
fviz_pca_biplot(pca1, habillage=abalone$sex)+
  scale_color_brewer(palette="YlOrRd") +
  theme_minimal()
```



```
# get the loadings (eigvenvectors)
loadings= pca1$rotation
loadings
```

```
##                      PC1         PC2         PC3          PC4         PC5
## length        0.19315606  0.35006929 -0.65543596 -0.038784599  0.15584501
## diameter      0.15955208  0.31882074 -0.50547308  0.018060452  0.07483574
## height        0.05928271  0.13475175 -0.08607958  0.004683252 -0.92444847
## whole_weight  0.84261922  0.01882402  0.31147028 -0.127977156  0.16797945
## shucked_weight 0.37195895 -0.70343169 -0.33727250  0.353767145 -0.16244383
## gut_weight    0.18225102  0.01294771  0.02506135 -0.762977566 -0.20728245
## shell_weight  0.22834926  0.51216078  0.30999426  0.523911759 -0.13392483
##                         PC6          PC7
## length        -0.0005606153 -0.620285186
## diameter       0.0302034552  0.781379947
## height         0.3377048831 -0.047395498
## whole_weight   0.3846953125 -0.006247874
## shucked_weight -0.3184028855  0.012572505
## gut_weight    -0.5828809182  0.033732861
## shell_weight  -0.5439869513 -0.033321509
```

```
# make a dataframe with age aand scores
# the scores The coordinates of the individuals (observations) on the principal components.
pca_scores= pca1$x
pca_lm=as.data.frame(cbind(abalone$age, pca_scores))

# linear regression
lm1=lm(V1~PC1+PC2, data=pca_lm)
```

```r
summary(lm1)
```

```
## 
## Call:
## lm(formula = V1 ~ PC1 + PC2, data = pca_lm)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -6.9538 -1.4075 -0.4151  0.8910 15.2801 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  9.93368    0.03508  283.18   <2e-16 ***
## PC1          2.96944    0.06033   49.22   <2e-16 ***
## PC2         23.96137    0.55723   43.00   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.267 on 4174 degrees of freedom
## Multiple R-squared:  0.5058, Adjusted R-squared:  0.5055 
## F-statistic:  2136 on 2 and 4174 DF,  p-value: < 2.2e-16
```

```r
lm1=lm(V1~PC1+PC2+PC3+PC4+PC5, data=pca_lm)
summary(lm1)
```

```
## 
## Call:
## lm(formula = V1 ~ PC1 + PC2 + PC3 + PC4 + PC5, data = pca_lm)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -7.7267 -1.3971 -0.4208  0.9277 15.1438 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  9.93368    0.03471 286.174  < 2e-16 ***
## PC1          2.96944    0.05970  49.740  < 2e-16 ***
## PC2         23.96137    0.55140  43.456  < 2e-16 ***
## PC3          5.36884    0.64381   8.339  < 2e-16 ***
## PC4          4.01619    1.06887   3.757 0.000174 ***
## PC5         -4.45174    1.56886  -2.838 0.004568 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 2.243 on 4171 degrees of freedom
## Multiple R-squared:  0.5164, Adjusted R-squared:  0.5158 
## F-statistic: 890.8 on 5 and 4171 DF,  p-value: < 2.2e-16
```

```r
resettest(lm1) # fail
```

```
## 
##  RESET test
## 
## data:  lm1
## RESET = 33.084, df1 = 2, df2 = 4169, p-value = 5.553e-15
```

```
dwtest(lm1) # fail
```

```
##
##  Durbin-Watson test
##
## data:  lm1
## DW = 1.3704, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

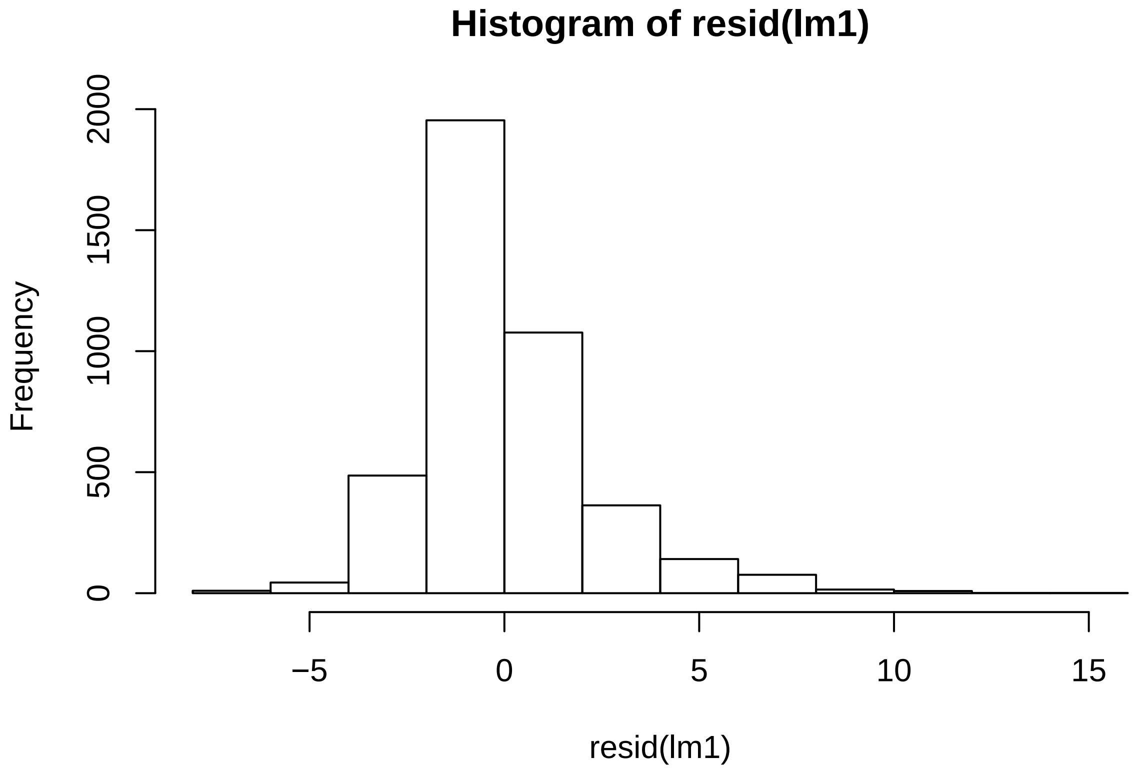
```
bptest(lm1) # fail
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm1
## BP = 376.3, df = 5, p-value < 2.2e-16
```

```
shapiro.test(resid(lm1)) # fail
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(lm1)
## W = 0.92019, p-value < 2.2e-16
```

```
hist(resid(lm1))
```



```
vif(lm1)
```

```
## PC1 PC2 PC3 PC4 PC5
##   1   1   1   1   1
```