SAMPLE STATISTICAL METHODS IN R

Eric Odongo

Dec 11, 2020

STATISTICAL METHODS COMMONLY USED IN DATA ANALYSIS

To determine the appropriate method to use we first look at the nature of the variables in question and their distribution.

Parametric methods are used when we have data that follows normal distribution while non-parametric methods are used when our data violates the normality assumption.

Checking for normality

We use the following methods to check for normality.

You can download sample data from here:

https://www.kaggle.com/spscientist/students-performance-in-exams

a.Histograms - used to investigate the distribution of a single variable.

```r
#load required packages
#install.packages("PerformanceAnalytics")
library(knitr)
library(tidyverse)
library(janitor)
library(vcd)
library(PerformanceAnalytics)

#data source:
#https://www.kaggle.com/spscientist/students-performance-in-exams

#load data into R
data = read.csv("StudentsPerformance.csv")
dim(data)
```

```
## [1] 1000    8
```

```r
names(data)
```

```
## [1] "gender"                      "race.ethnicity"
## [3] "parental.level.of.education" "lunch"
## [5] "test.preparation.course"     "math.score"
## [7] "reading.score"               "writing.score"
```

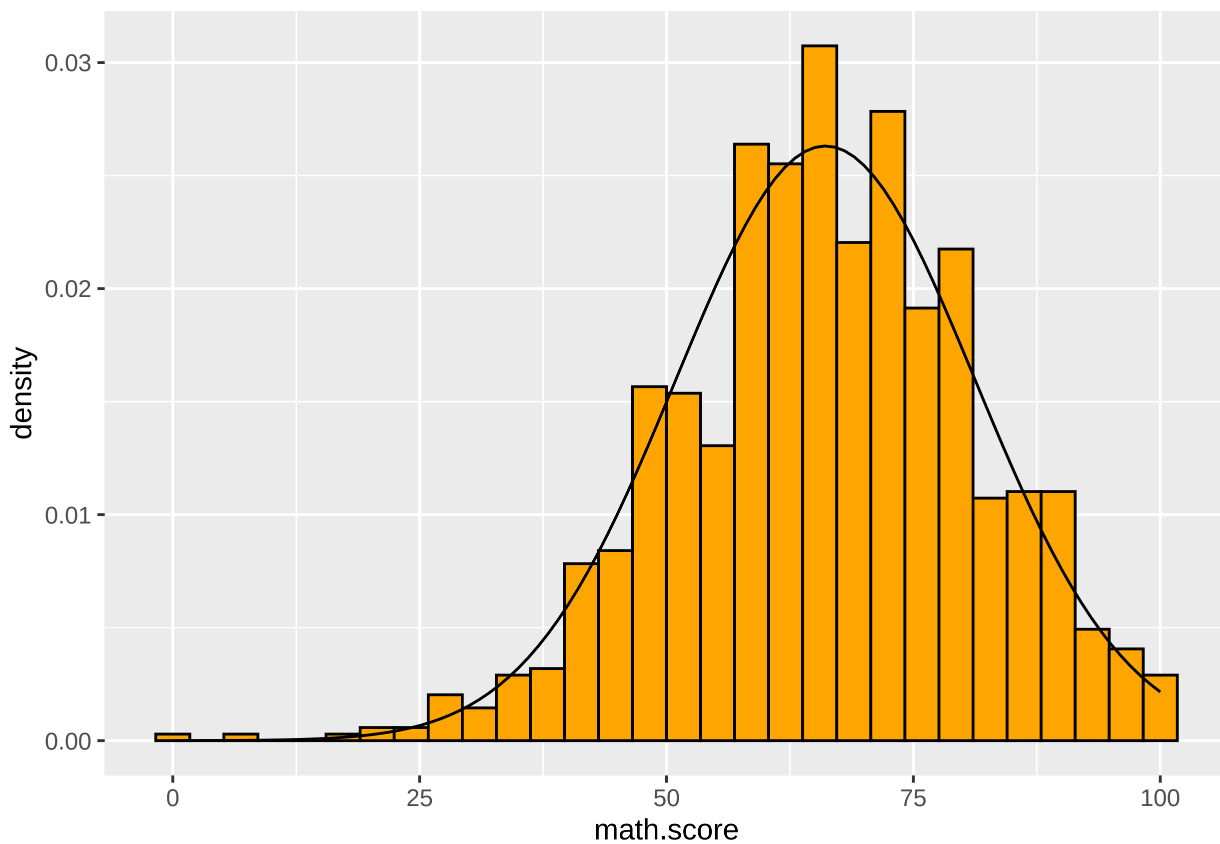
```r
str(data, vec.len = 1)
```

```
## 'data.frame':    1000 obs. of  8 variables:
##  $ gender                     : chr  "female" ...
##  $ race.ethnicity             : chr  "group B" ...
##  $ parental.level.of.education: chr  "bachelor's degree" ...
##  $ lunch                      : chr  "standard" ...
##  $ test.preparation.course    : chr  "none" ...
##  $ math.score                 : int  72 69 ...
##  $ reading.score              : int  72 90 ...
##  $ writing.score              : int  74 88 ...
```

```r
sum(!complete.cases(data))
```

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

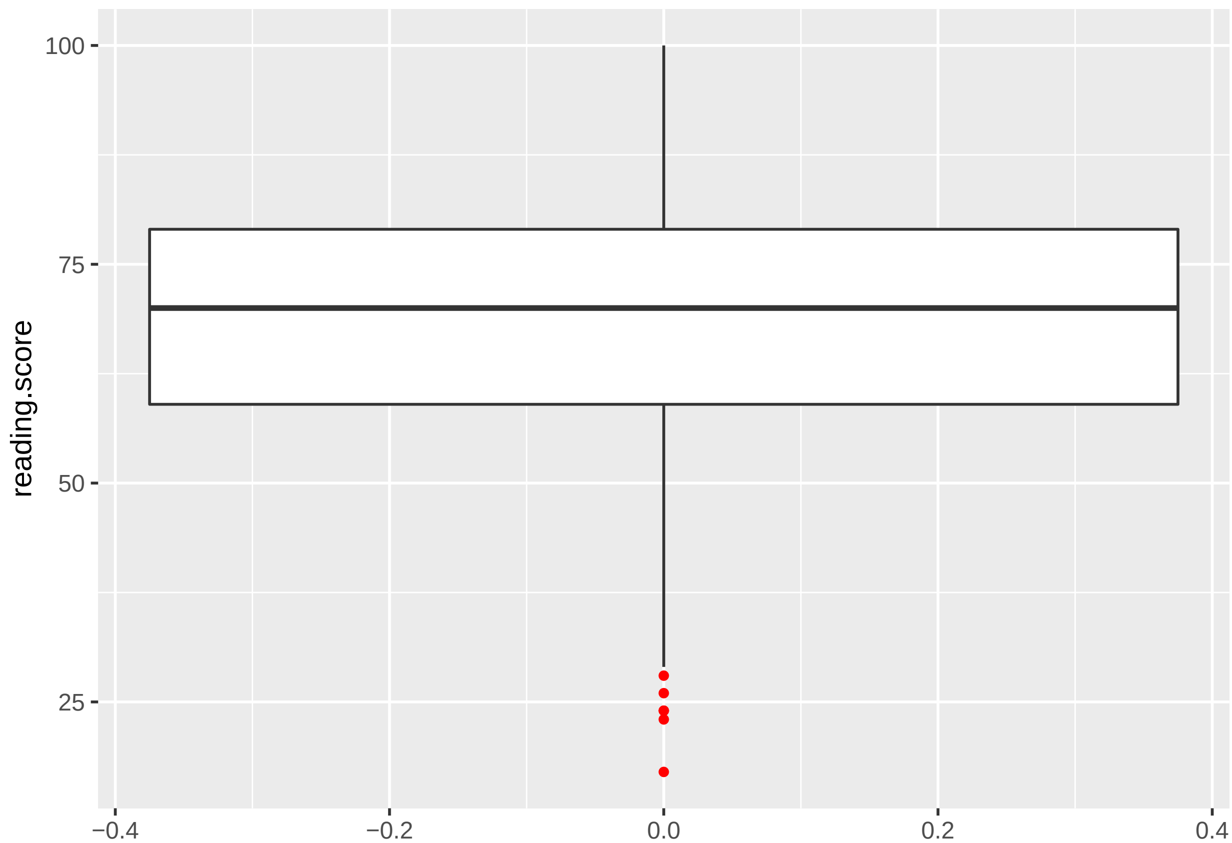
```r
#Lets see the distribution of maths scores
ggplot(data = data, aes(x = math.score)) +
  geom_histogram(aes(y = ..density..), color = "black",
                 fill = "orange") +
  stat_function(fun = dnorm, args = list(mean = mean(data$math.score),
                                         sd = sd(data$math.score)))
```



```
#if the data is normal the histogram will approximately fit into the
#normal curve
```

b.Boxplot

```
ggplot(data = data) +
  geom_boxplot(aes(y = reading.score),
               outlier.colour = "red", outlier.shape = 16)
```

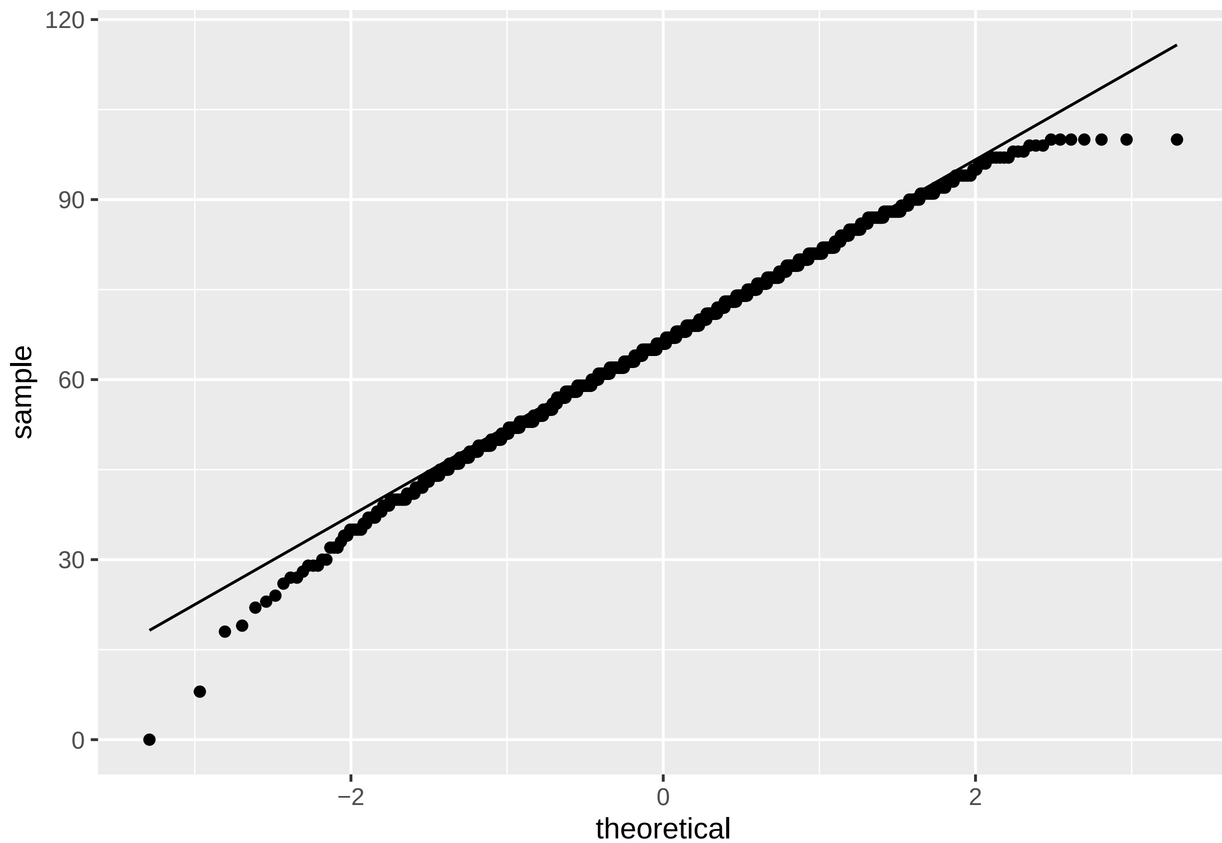


c.QQ Plots

If the variable follows a normal distribution, the quantiles must be perfectly in line with the "theoretical" normal quantiles.

A straight line on the QQ Plot tells us we have a normal distribution.

```
ggplot(data = data) +
  stat_qq(aes(sample = math.score)) +
  stat_qq_line(aes(sample = math.score))
```



d.Kolmogorov Smirnov test

We use this to formally test the hypothesis that our data is normal.

Nully hypothesis: Our data is normal.

Alternative hypothesis: Our data is not normal

We reject the null hypothesis if our p-value is less than the specified significance level.

```
ks.test(data$math.score, "pnorm", mean = mean(data$math.score),
        sd = sd(data$math.score))
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  data$math.score
## D = 0.030855, p-value = 0.297
## alternative hypothesis: two-sided
```

```
#at 5% significance level we fail to reject the null hypothesis
```

e.Shapiro Method

```
shapiro.test(data$math.score)
```

```
##
```

```
##  Shapiro-Wilk normality test
##
## data:  data$math.score
## W = 0.99315, p-value = 0.0001455
```

```r
#at 5% significance level we reject the null hypothesis
```

PARAMETRIC METHODS

These methods are only employed once we have ascertained that our data follows the normal distribution, otherwise non-parametric methods should be exploited.

Univariate analysis

For continuous variables we get summary statistics and for categorical data we get counts per level.

```r
#get the mode/class of each variable/column name
sapply(data, class)
```

```
##                      gender              race.ethnicity
##                 "character"                 "character"
## parental.level.of.education                       lunch
##                 "character"                 "character"
##     test.preparation.course                  math.score
##                 "character"                   "integer"
##               reading.score               writing.score
##                   "integer"                   "integer"
```

```r
head(data)
```

```
##   gender race.ethnicity parental.level.of.education        lunch
## 1 female        group B           bachelor's degree     standard
## 2 female        group C                some college     standard
## 3 female        group B             master's degree     standard
## 4   male        group A          associate's degree free/reduced
## 5   male        group C                some college     standard
## 6 female        group B          associate's degree     standard
##   test.preparation.course math.score reading.score writing.score
## 1                    none         72            72            74
## 2               completed         69            90            88
## 3                    none         90            95            93
## 4                    none         47            57            44
## 5                    none         76            78            75
## 6                    none         71            83            78
```

```r
#we perform column transformations to capture informtion on #categorical columns
data$gender = factor(data$gender)

data$race.ethnicity = factor(data$race.ethnicity)
```

```r
data$parental.level.of.education = factor(data$parental.level.of.education)

data$lunch = factor(data$lunch)

data$test.preparation.course = factor(data$test.preparation.course)

#summary stats for continuous variables
summary(data$math.score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   57.00   66.00   66.09   77.00  100.00
```

```r
summary(data$reading.score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   59.00   70.00   69.17   79.00  100.00
```

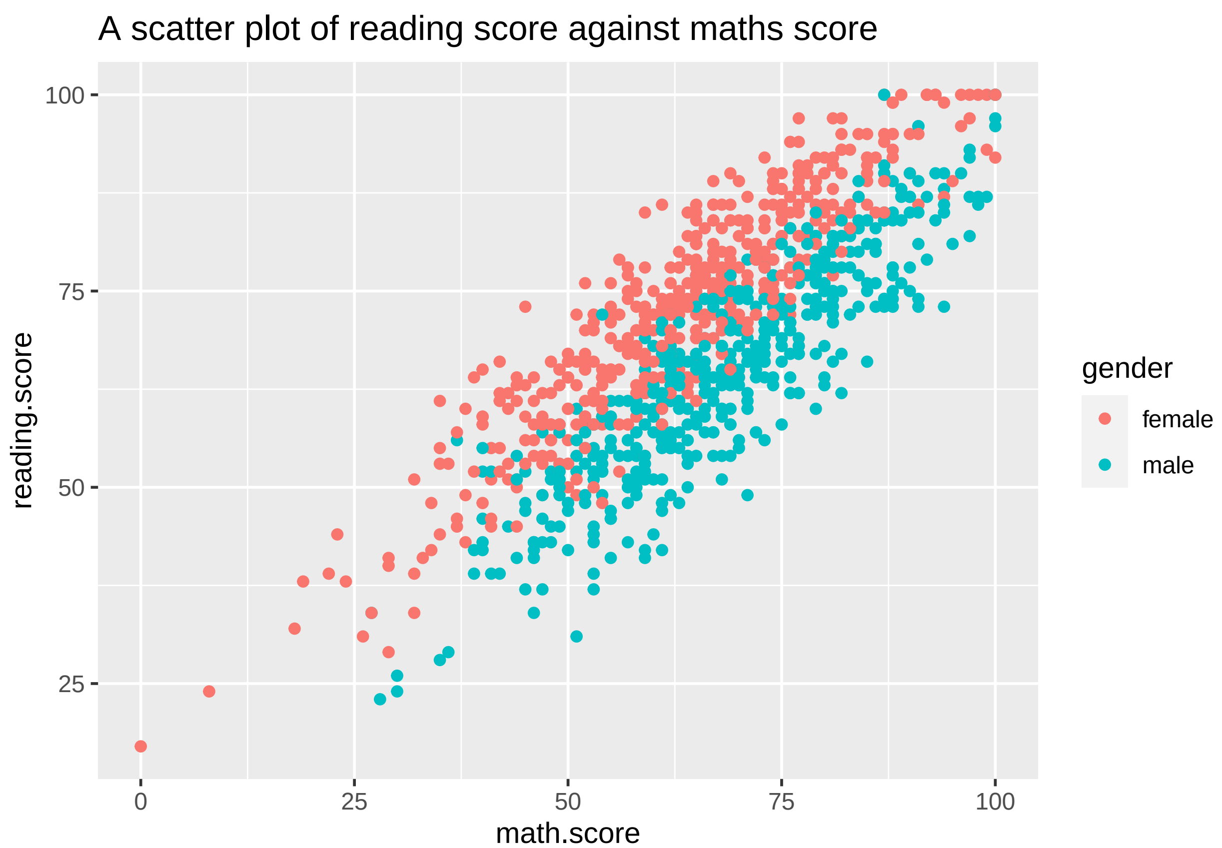
```r
summary(data$writing.score)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.00   57.75   69.00   68.05   79.00  100.00
```

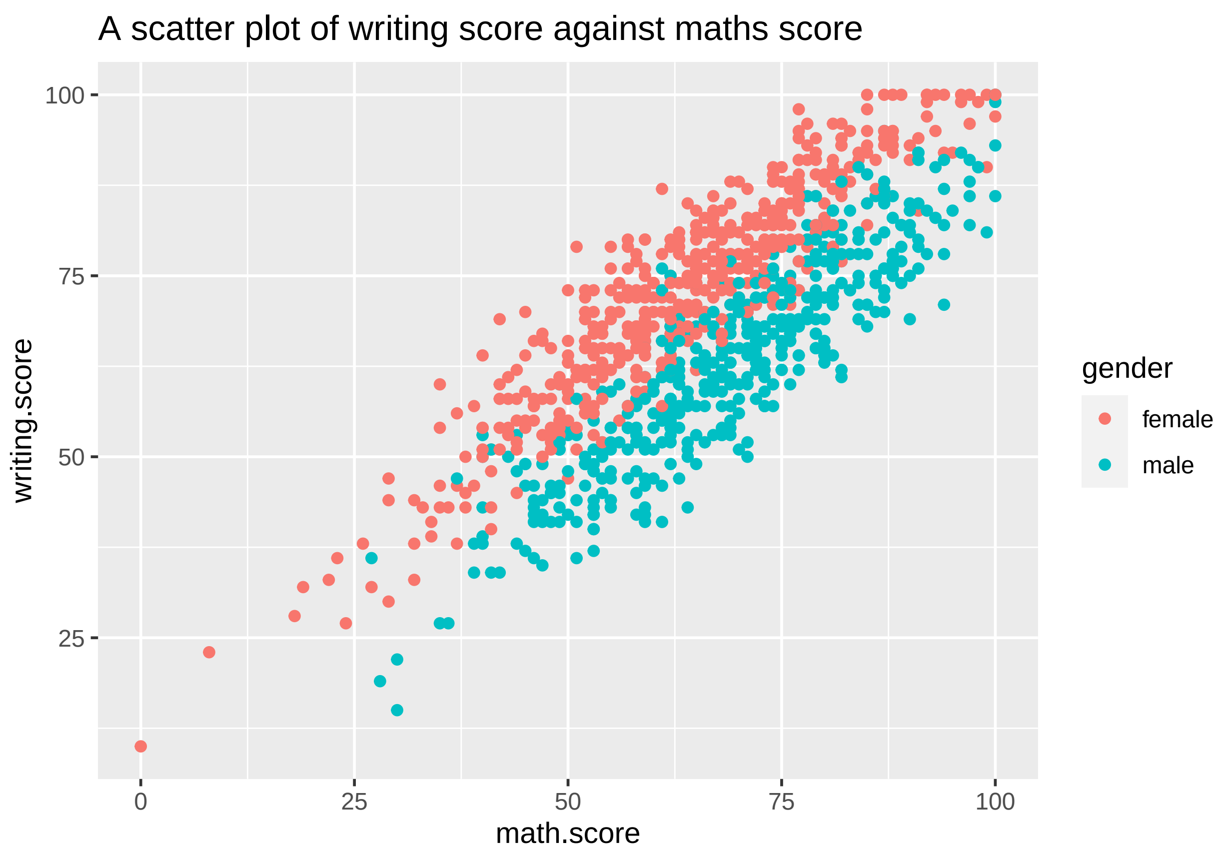
Bivariate analysis

We seek relations among two or more variables.

```r
#lets start with relations among our continuous variables
ggplot(data = data) +
  geom_point(aes(x = math.score, y = reading.score, color = gender)) +
  ggtitle("A scatter plot of reading score against maths score")
```




```r
ggplot(data = data) +
  geom_point(aes(x = math.score, y = writing.score, color = gender)) +
  ggtitle("A scatter plot of writing score against maths score")
```



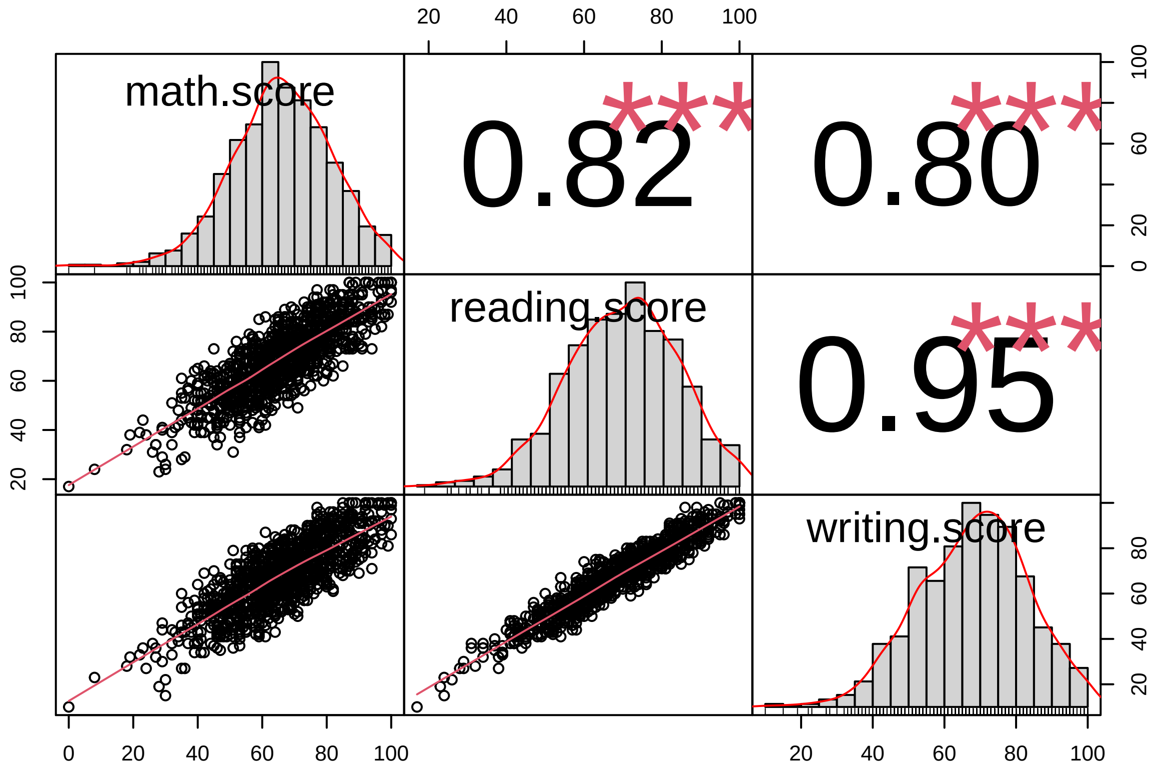

```r
ggplot(data = data) +
  geom_point(aes(x = writing.score, y = reading.score,
                 color = gender)) +
  ggtitle("A scatter plot of reading score against writing score")
```

For numeric variables we can also do correlation analysis and visualize these correlations.

```
#correlation matrix for our numeric columns/variables
cor_mat = cor(cbind(data$math.score, data$reading.score,
                    data$writing.score))
cor_mat
```

```
##           [,1]      [,2]      [,3]
## [1,] 1.0000000 0.8175797 0.8026420
## [2,] 0.8175797 1.0000000 0.9545981
## [3,] 0.8026420 0.9545981 1.0000000
```

```
chart.Correlation(data[6:8], histogram = T, method = "pearson")
```




```
#To carry out tests on these correlations and get the confidence
#interval use cor.test(x, y)
```

Contingency Tables

These are useful for displaying counts across various levels of a *categorical* variable.

```
kable(tabyl(data, gender))
```

gender	n	percent
female	518	0.518
male	482	0.482

```
kable(tabyl(data, lunch))
```

lunch	n	percent
free/reduced	355	0.355
standard	645	0.645

```r
kable(tabyl(data, parental.level.of.education))
```

parental.level.of.education	n	percent
associate's degree	222	0.222
bachelor's degree	118	0.118
high school	196	0.196
master's degree	59	0.059
some college	226	0.226
some high school	179	0.179

```r
kable(tabyl(data, test.preparation.course))
```

test.preparation.course	n	percent
completed	358	0.358
none	642	0.642

```r
kable(tabyl(data, race.ethnicity))
```

race.ethnicity	n	percent
group A	89	0.089
group B	190	0.190
group C	319	0.319
group D	262	0.262
group E	140	0.140

```r
#you can do cross tabulations across two variables at once
kable(tabyl(data, gender, lunch))
```

gender	free/reduced	standard
female	189	329
male	166	316

```r
#Possible tests with count data
#Chi-square test for independence
chisq.test(data$gender, data$race.ethnicity)
```

```
##
##  Pearson's Chi-squared test
##
## data:  data$gender and data$race.ethnicity
## X-squared = 9.0274, df = 4, p-value = 0.06042
```

```r
#Cochran-Mantel-Haenszel test
#this test gives us an assessment of the relationship between x1
#and x2 stratified by(or controlling for) x3
mantelhaen.test(data$test.preparation.course, data$race.ethnicity,
                data$gender)
```

```
##
##  Cochran-Mantel-Haenszel test
##
## data:  data$test.preparation.course and data$race.ethnicity and data$gender
## Cochran-Mantel-Haenszel M^2 = 5.4917, df = 4, p-value = 0.2405
```

```r
#Cramer's V
#This measures association between nominal variables
assocstats(table(data$gender, data$race.ethnicity))
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 9.0526  4 0.059798
## Pearson          9.0274  4 0.060419
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.095
## Cramer's V        : 0.095
```

```r
#Fisher's exact test
fisher.test(data$lunch, data$gender)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  data$lunch and data$gender
## p-value = 0.509
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.8366388 1.4297884
## sample estimates:
## odds ratio
##   1.093467
```

```r
#McNemar's test
#This is used to test the null hypothesis that the proportions are
#equal across matched pairs
mcnemar.test(data$gender, data$test.preparation.course)
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  data$gender and data$test.preparation.course
## McNemar's chi-squared = 49.766, df = 1, p-value = 1.732e-12
```

Tests for continuous variables

The t-test This tests for equality of means across two groups. If you have more than two groups to compare then **anova** should be used.

```
#we want to test if the mean performance in maths is significantly
#different for the two genders.
#we reject the null hypothesis if our p-value is less than the #specified alpha(significance level)
t.test(math.score ~ gender, data = data)
```

```
##
##  Welch Two Sample t-test
##
## data:  math.score by gender
## t = -5.398, df = 997.98, p-value = 8.421e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -6.947209 -3.242813
## sample estimates:
## mean in group female   mean in group male
##             63.63320             68.72822
```

```
var.test(math.score ~ gender, data = data)
```

Test for equal variance

```
##
##  F test to compare two variances
##
## data:  math.score by gender
## F = 1.1644, num df = 517, denom df = 481, p-value = 0.09016
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.9764071 1.3877941
## sample estimates:
## ratio of variances
##           1.164396
```

NON PARAMETRIC METHODS

When the normality assumption cannot be ascertained we drift to non-parametric statistics. These are also called distribution-free methods.

i) Wilcoxon Rank sum Test

One sample static for the hypothesis that the distribution of the given vector is symmetric about the mean *mu*.

When given two vectors or two samples, Mann-Whitney test is performed.

```r
x = c(1,3,4,5,6,8,9)
y = c(2,5,7,11,23,7)

wilcox.test(x, mu = 5, exact = F)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  x
## V = 11, p-value = 1
## alternative hypothesis: true location is not equal to 5
```

```r
wilcox.test(x, y, mu = 5, exact = F) #Mann-Whitney test
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x and y
## W = 2, p-value = 0.008133
## alternative hypothesis: true location shift is not equal to 5
```

ii) Kruskal Wallis test

This tests the null hypothesis that the location parameters of the distribution of x are the same in each group(sample).

```r
kruskal.test(math.score ~ gender, data = data)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  math.score by gender
## Kruskal-Wallis chi-squared = 25.565, df = 1, p-value = 4.277e-07
```

Analytic Power and Sample Size Calculations

It's common in survey statistics to work with sample data. The main question is how do you determine the size of your sample?

```r
#find sample size for two-sample t-test
power.t.test(delta=0.5, power=0.8)
```

```
##
##      Two-sample t test power calculation
##
##               n = 63.76576
##           delta = 0.5
##              sd = 1
##       sig.level = 0.05
##           power = 0.8
```

```
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
#find power for two-sample t-test
power.t.test(delta=0.5, n=200)
```

```
##
##      Two-sample t test power calculation
##
##               n = 200
##           delta = 0.5
##              sd = 1
##       sig.level = 0.05
##           power = 0.9987689
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
#for more please consult the pwr package
```

References

R Markdown: The Definitive Guide

using-r-and-rstudio-for-data-management-statistical-analysis-and-graphics-2nd-edit