**Dienst Curriculum & vorming**
Team secundair onderwijs
Guimardstraat 1 - 1040 BRUSSEL
www.katholiekonderwijs.vlaanderen

# Statistiek met het programma R

## INHOUDSOPGAVE

# 1 Inleiding

Via dit document leggen we uit hoe het softwareprogramma R kan worden gebruikt bij de leerplanrealisatie van de doelen statistiek. De tekst is niet bedoeld als leermateriaal voor de leerlingen, maar als hulpdocument voor de leraar.

In de verschillende leerplannen Wiskunde 2de en 3de graad in de D/A-finaliteit en D-finaliteit zijn leerplandoelen opgenomen over statistiek. De concrete nummering van de doelen hangt af van het leerplan.

Leerplandoelen statistiek in leerplan Wiskunde van de 2de graad:

| | |
|---|---|
| **LPD 1** | **De leerlingen stellen statistische gegevens voor aan de hand van passende voorstellingswijzen: absolute en relatieve frequentietabel, staafdiagram, cirkeldiagram, lijndiagram, histogram en boxplot.** |
| **LPD 2** | **De leerlingen bepalen centrum- en spreidingsmaten: rekenkundig gemiddelde, mediaan, modus, variatiebreedte, interkwartielafstand en standaardafwijking.** |
| **LPD 3** | **De leerlingen analyseren statistische gegevens aan de hand van voorstellingswijzen, centrum- en spreidingsmaten.** |
| **LPD 4** | **De leerlingen analyseren het verband tussen twee numerieke grootheden in een dataset met behulp van een spreidingsdiagram.** |

Leerplandoelen statistiek in leerplan Wiskunde van de 3de graad:

| | |
|---|---|
| **LPD 5** | **De leerlingen verklaren het belang van randomisatie en representativiteit bij steekproeven voor het formuleren van statistische besluiten over een populatie.** |
| **LPD 6** | **De leerlingen leggen in concrete situaties het verschil uit tussen samenhang en causaliteit.** |
| **LPD 7** | **De leerlingen gebruiken de normale verdeling als continu model bij gegeven data.** |
| **LPD 8** | **De leerlingen berekenen kansen bij een normaal verdeelde kansvariabele.** |

Leerplandoelen in het leerplan Statistiek van de 3de graad D-finaliteit (III-Stat-d):

| | |
|---|---|
| **LPD 9** | **De leerlingen lossen telproblemen zonder herhaling op met combinaties.** |
| **LPD 10** | **De leerlingen berekenen en interpreteren kansen met behulp van de binomiale verdeling.** |
| **LPD 11** | **De leerlingen toetsen hypothesen aan de hand van de begrippen nulhypothese, alternatieve hypothese, significantieniveau en p-waarde.** |
| **LPD 12** | **De leerlingen leggen in betekenisvolle situaties de betekenis van betrouwbaarheidsniveau, betrouwbaarheidsinterval en foutenmarge uit.** |
| **LPD 13** | **De leerlingen analyseren grote datasets met behulp van statistische software in functie van een statistisch onderzoek.** |

# 2 Het statistisch programma R

R is een taal en omgeving die de gebruiker toelaat statistische berekeningen te maken en grafische voorstellingen te maken.

Het statistisch programma *R* wordt het best samen gebruikt met het softwarepakket *RStudio*.

Hoewel R voldoende is, wordt aangeraden om ook de RStudio interface te downloaden.

RStudio is een interface (toolbox) rond R om het gebruik van R sterk te vereenvoudigen en overzichtelijk te maken. Beide programma's kan je gratis downloaden.

Eerst installeer je R via:

- Windows: http://cran.freestatistics.org/bin/windows/base/
- Mac: http://cran.r-project.org/bin/macosx/
- Linux: http://cran.r-project.org/bin/linux/

RStudio kan je downloaden via http://www.rstudio.com/products/rstudio/download/

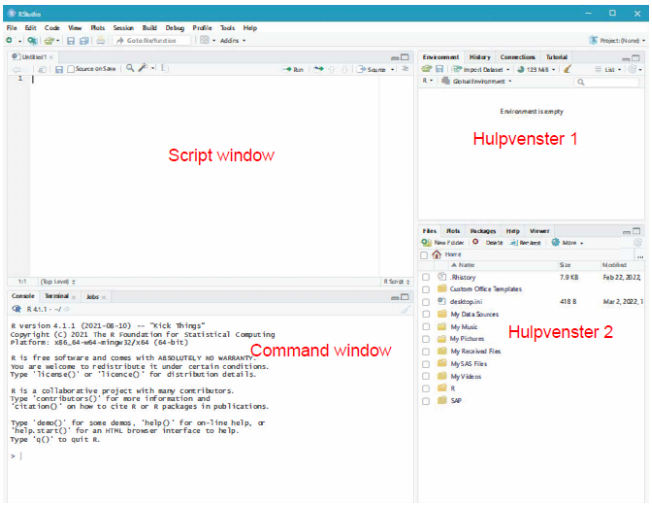
Bij "All Installers" vind je de versie voor jouw systeem.

## 2.1 Werking van RStudio.

Hoe werk je met R?

- Je schrijft een script met commando's en functies (een tekstbestand met extensie .R)
- Je uploadt de data in R in een toegankelijk formaat (bijv. CSV, Excel).
- Je voert de commando's van het script uit voor de specifieke data.
- Je bewaart de verkregen resultaten en figuren.

Om het overzichtelijk te houden werkt RStudio met vier deelvensters.




- De **command window of console** vind je linksonder. In dit venster worden alle commando's uitgevoerd. Het is echter inefficiënt om code rechtstreeks in dit venster in te geven. Bovendien kan je nadien je code dan niet opnieuw uitvoeren. Om dit te vermijden maken we aparte bestanden, scripts genaamd, waarin we de commando's schrijven en dan zullen uitvoeren in de console. Dit doen we in het scriptvenster. Dit venster kan je leeg maken door rechtsboven op het borsteltje te klikken of Ctrl+L.

- De **script window**. Als dit venster niet geopend werd bij het starten van RStudio, ga je linksboven naar File>New File>R script. Je krijgt nu een leeg bestand. In dit bestand kan je code invoeren en vervolgens opslaan in een script via File>Save zodat je ze later eenvoudig opnieuw kan hergebruiken. Het voordeel van een script is dat je het kan opslaan, zodat je later niet al je code opnieuw hoeft te typen.

- **Hulpvenster 1**. Dit venster heeft vier tabs:
  - ⇨ De tab "Environment" geeft alle variabelen, functies... weer die momenteel in het geheugen aanwezig zijn. Via deze tab zullen we later ook datasets inlezen. Het bezemsteelicoontje maakt alle bekende variabelen, datasets, functies... leeg.
  - ⇨ Onder de tab "History" vind je de vorige commando's terug die in de console werden ingegeven. Je kan er een commando aanklikken en het naar de console sturen via "To Console" of naar een script via "To Source".
  - ⇨ De tabs "Connections" en "Tutorial" zullen we niet gebruiken in deze introductie.

- **Hulpvenster 2.** Dit venster heeft zes tabs:
  - ⇨ Via "Files" kan je naar bestanden browsen en deze openen.
  - ⇨ In het tabblad "Plots" worden figuren getoond. Je kan hier een figuur groter maken via "Zoom". Via "Export" kan je de figuur opslaan (als PDF, JPG, PNG, . . .) of kopiëren om te gebruiken in o.a. Word. Als je meerdere figuren gemaakt hebt, kan je door deze figuren bladeren met de pijl-icoontjes. Ook hier kan je al je plots wegdoen door het borsteltje.
  - ⇨ Het tabblad "Help" kan nuttig zijn om dingen op te zoeken. Je vindt er ook de "An Introduction to R" handleiding waar je veel informatie over R kan vinden. Klik eerst op "home 🏠" en kijk dan bij Manuals. Ook met het commando `help(topic)` of `?topic` kan je steeds alle informatie van een topic opvragen. Onderaan het venster worden voorbeelden van gebruik gegeven die je ook zelf kan uitvoeren.

    ```
    # helpfunctie
    > help(hist) of ?hist
    ```
  - ⇨ In RStudio zijn packages (pakketten) sets van tools, functies en datasets die specifieke functionaliteiten toevoegen aan de R-programmeertaal. Ze zijn essentieel voor het uitbreiden van de mogelijkheden van R, omdat de basistaal zelf beperkte functionaliteit heeft. Er zijn talrijke pakketten ontwikkeld door andere gebruikers om analysetechnieken makkelijker te maken. Voor deze workshop moeten we verschillende pakketten installeren naast enkele functies die we maakten om alle LPD met R te realiseren.
  - ⇨ De tabs "Viewer" en "Presentation" vallen buiten het bereik van deze workshop.

## 2.2 Packages in R

Packages breiden de functionaliteit van R uit door **extra functies, datasets en tools** aan te bieden. De meeste R-packages worden gehost op CRAN, waar je ze kunt vinden, downloaden en installeren. CRAN is de officiële repository voor R-packages. RStudio biedt een handige interface voor het beheren van packages. Je kunt packages installeren, bijwerken en verwijderen met behulp van de knoppen en menu-opties in de interface. R-packages hebben vaak versieafhankelijkheden, wat betekent dat bepaalde versies van een package compatibel zijn met specifieke versies van andere packages. RStudio en R zorgen voor het beheer van deze afhankelijkheden. Gebruikers kunnen ook hun eigen packages maken en delen met anderen.

- Installatie van Packages:

Om een package te gebruiken, moet je het eerst installeren. Dit kan worden gedaan met behulp van het commando `install.packages("package_naam")`.

- Laden van Packages:

Nadat een package is geïnstalleerd, moet je het laden in je R-script of -console. Dit wordt gedaan met behulp van het commando `library(package_naam)`.

Packages voor deze workshop:

```
# pakketten installeren
> install.packages("readxl")
> install.packages("DescTools")
> install.packages("aplpack")
> install.packages("BSDA")
> install.packages("datasets")
> install.packages("MASS")
```

We maakten ook een eigen R-package dat een aantal extra functies bevat, m.n. ‘cirkeldiagram’, ‘grafbinom’, ‘grafnorm’, ‘kansbinom’, ‘kansnorm’ en ‘modus’. Dit pakket wordt niet gehost op CRAN, maar is beschikbaar via Github. Via het R-package ‘devtools’ kunnen ook pakketten vanop Github gebruikt worden.

```
> install.packages("devtools")
> library(devtools)
> install_github("WisKOV/StatSO")
```

# 3 Data

## 3.1 Soorten data

In RStudio kun je verschillende soorten gegevensstructuren en datatypes gebruiken. Hier zijn enkele veelgebruikte datatypes en structuren in R:

- Vector:

Een eendimensionale verzameling van gelijksoortige gegevenselementen.
Voorbeelden: c(1, 2, 3), c("a", "b", "c").

- Matrix:

Een tweedimensionale gegevensstructuur waarin gegevens worden georganiseerd in rijen en kolommen. Ze worden gemaakt met de functie `matrix()`.

- Dataframe:

Een tweedimensionale gegevensstructuur die verschillende datatypes kan bevatten in verschillende kolommen. Ze worden gemaakt met de functie `data.frame()`.

- Lijst:

Een geordende verzameling gegevens van verschillende datatypes. Ze worden gemaakt met de functie `list()`.

- Factor:

Een datatype dat wordt gebruikt om nominale of ordinale categorieën te vertegenwoordigen.
Ze worden gemaakt met de functie `factor()`.

## 3.2 Soorten bestanden

Doorgaans zullen gegevens niet binnen R worden ingetypt maar vanuit een of ander bestandsformaat worden geüpload.

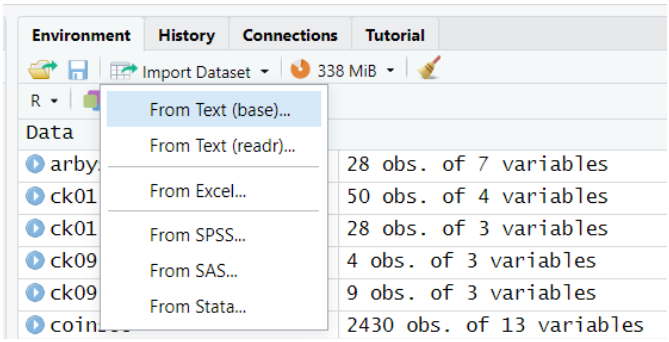
**CSV-BESTANDEN**

Men kan rechtstreeks csv-bestanden inlezen met behulp van de functie:

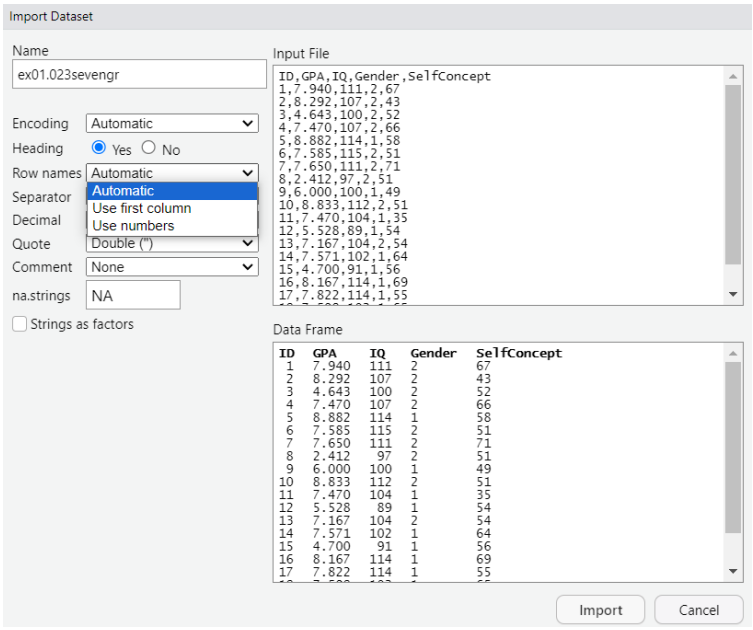
```
read.csv(file, header = TRUE, sep = ",", quote = "\"", dec = ".",
fill = TRUE, comment.char = "", ...)
```
of
```
read.csv2(file, header = TRUE, sep = ";", quote = "\"",dec = ",",
fill = TRUE, comment.char = "", ...)
```

Men kan een csv-bestand ook inlezen op de volgende manier:

- Duid in hulpvenster 1 “Import Dataset” aan en ga naar “From Tekst(base)”.




- Zoek het juiste bestand op je computer.
- Je kan links bovenaan de naam van de data veranderen naar “zevendejaar”.
- Je kan nog meegeven dat de rijnamen in de eerste kolom staan.




- Druk rechtsonder op “Import”.

**EXCEL-BESTANDEN**

Ook Excel-bestanden kunnen ingelezen worden. Hiervoor zal er eerst een extra pakket ingelezen moeten worden. Voer de volgende commando’s uit om gebruik te kunnen maken van deze extra functies:

```
> install.packages("readxl")
> library(readxl)
```

> help(package="readxl") → alle extra functies verschijnen nu in hulpvenster 2.

Het opzoeken van het Excel-formaat van het gegeven bestand doe je met:

```
> excel_format(path)
```

Gebruik een van de volgende functies naargelang het juiste Excel-formaat:

> read_excel(path)

> read_xls(path)

> read_xlsx(path)

Tip: zorg ervoor dat het bestand dat je wil inlezen niet openstaat op je computer.

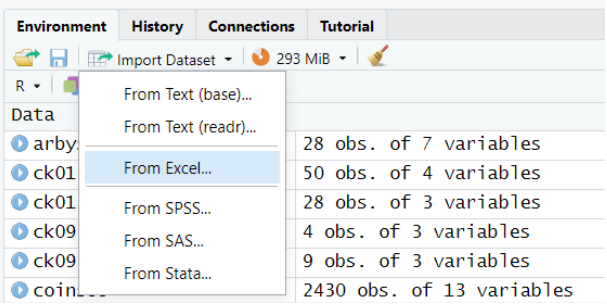
**Voorbeeld:**

```
> moviedata<-read_excel("ExcelMovieData.xlsx")
> View(moviedata)
> names(moviedata)
> head(moviedata)
```

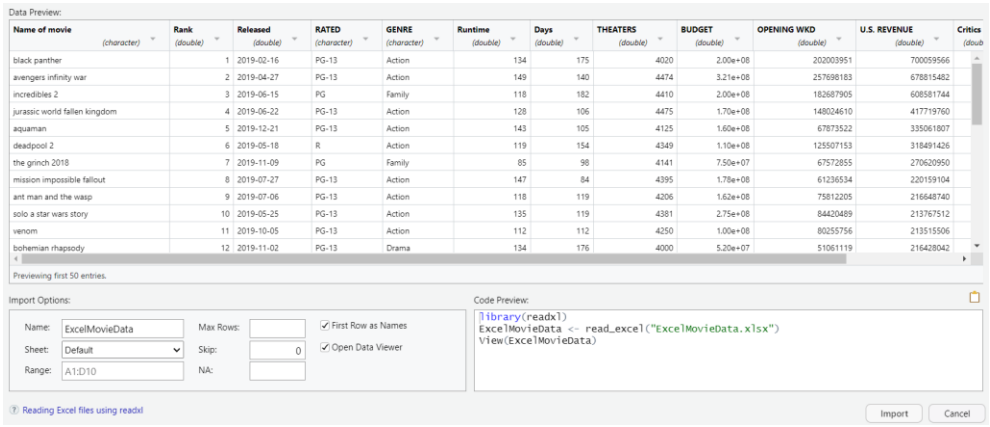
```
  `Name of movie`        Rank Released            RATED GENRE Runtime  Days THEATERS BUDGET `OPENING WKD` `U.S. REVENUE`
  <chr>                 <dbl> <dttm>              <chr> <chr>   <dbl> <dbl>    <dbl>  <dbl>         <dbl>          <dbl>
1 black panther             1 2019-02-16 00:00:00 PG-13 Acti…     134   175     4020 2   e8     202003951      700059566
2 avengers infinity war     2 2019-04-27 00:00:00 PG-13 Acti…     149   140     4474 3.21e8     257698183      678815482
3 incredibles 2             3 2019-06-15 00:00:00 PG    Fami…     118   182     4410 2   e8     182687905      608581744
4 jurassic world fallen…    4 2019-06-22 00:00:00 PG-13 Acti…     128   106     4475 1.7 e8     148024610      417719760
5 aquaman                   5 2019-12-21 00:00:00 PG-13 Acti…     143   105     4125 1.6 e8      67873522      335061807
6 deadpool 2                6 2019-05-18 00:00:00 R     Acti…     119   154     4349 1.10e8     125507153      318491426
# i 2 more variables: Critics <dbl>, Audience <dbl>
```

Men kan een Excel-bestand ook inlezen op de volgende manier:

- Duid in hulpvenster 1 “Import Dataset” aan en ga naar “From Excel”.




- Zoek het juiste bestand op je computer.
- Je kan de naam van de data veranderen links onderaan naar “moviedata”.
- Je hebt hier onderaan ook nog meerdere opties naargelang de structuur van je dataset.



#### R-BESTANDEN

Er bestaan ook veel datasets die al in een R-bestand staan. Deze kan je zeer gemakkelijk inlezen door dubbel te klikken op het bestand. RStudio opent dit bestand dan meteen.

#### DATABESTANDEN IN R

Ook in RStudio zelf vind je meerdere pakketten die extra datasets bevatten zoals de packages "datasets" en "MASS".

```
> library(datasets)
> library(MASS)
> help(package="MASS")
> help(package="datasets")
```

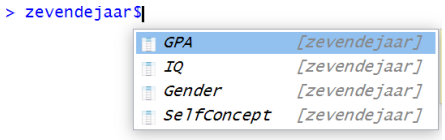
## 3.3 Selectie binnen een dataset

#### SELECTIE VAN EEN KOLOM

Voorbeeld GPA

Na het $ teken kan je een keuze maken tussen de verschillende variabelen.

```
> zevendejaar$
```




zevendejaar[,1] of zevendejaar$GPA geeft:

```
[1]  7.940  8.292  4.643  7.470  8.882  7.585  7.650  2.412  6.000  8.833  7.470  5.528
[13]  7.167  7.571  4.700  8.167  7.822  7.598  4.000  6.231  7.643  1.760  6.419  9.648
[25] 10.700 10.580  9.429  8.000  9.585  9.571  8.998  8.333  8.175  8.000  9.333  9.500
[37]  9.167 10.140  9.999 10.760  9.763  9.410  9.167  9.348  8.167  3.647  3.408  3.936
[49]  7.167  7.647  0.530  6.173  7.295  7.295  8.938  7.882  8.353  5.062  8.175  8.235
[61]  7.588  7.647  5.237  7.825  7.333  9.167  7.996  8.714  7.833  4.885  7.998  3.820
[73]  5.936  9.000  9.500  6.057  6.057  6.938
```

**SELECTIE VAN EEN RIJ**

```
> zevendejaar[1,]
```

```
   GPA  IQ Gender SelfConcept
1 7.94 111      2          67
```

**SELECTIE VOLGENS EEN CRITERIUM VAN EEN VARIABELE**

```
> zevendejaarman=subset(zevendejaar,Gender==2)
> zevendejaarvrouw=subset(zevendejaar,Gender==1)
```

```
head(zevendejaarman)
```

```
    GPA  IQ Gender SelfConcept
1 7.940 111      2          67
2 8.292 107      2          43
3 4.643 100      2          52
4 7.470 107      2          66
6 7.585 115      2          51
7 7.650 111      2          71
```

```
head(zevendejaarvrouw)
```

```
     GPA  IQ Gender SelfConcept
5  8.882 114      1          58
9  6.000 100      1          49
11 7.470 104      1          35
12 5.528  89      1          54
14 7.571 102      1          64
15 4.700  91      1          56
```

# 4 Voorstellingswijzen statistische gegevens

## 4.1 Histogram

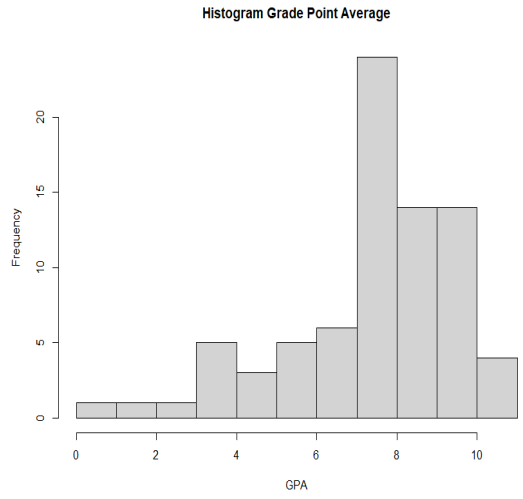
```
hist(Dataset$Variabele)
```

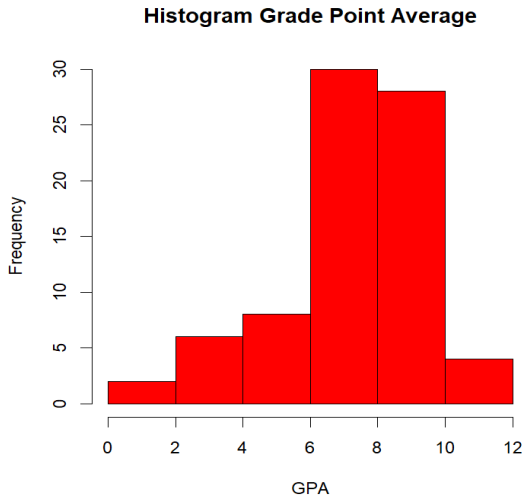
Mogelijke argumenten bij de functie 'hist':

- Hoofdtitel: `main = "…"`
- x-as benoemen: `xlab = "…"`
- y-as benoemen: `ylab = "…"`
- Kleur toevoegen: `col = "…"`
- Grenzen x-as: `xlim = c(a,b)`
- Grenzen y-as: `ylim = c(a,b)`
- Proporties y-as: `freq = FALSE`
- Aantal klassen: `nclass = …`

**Voorbeeld vanuit dataset:**

```
> hist(zevendejaar$GPA, main= "Histogram Grade Point Average", xlab="GPA")
> hist(zevendejaar$GPA, main= "Histogram Grade Point Average", xlab="GPA", nclass = 6, col="red")
```




## 4.2 Staafdiagram

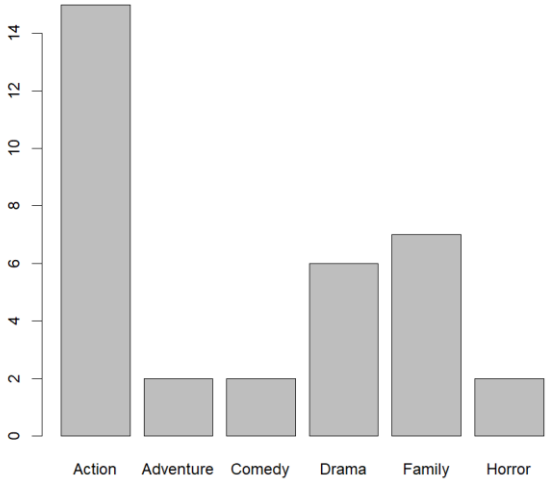
```
plot(Dataset$Variabele)
```

of

```
Naam <- table(Dataset$Variabele)
barplot(Naam)
```

**Voorbeeld vanuit dataset:**

```
> genremoviedata=table(moviedata$GENRE)
> genremoviedata

Action Adventure     Comedy      Drama     Family     Horror
    15         2          2          6          7          2

# De invoer van barplot moet een tabel zijn.
> barplot(genremoviedata)
```




```
# Of je maakt van de variabele GENRE in de oorspronkelijke dataset een factor.
> plot(factor(moviedata$GENRE))
```

## 4.3 Taartpunt- of cirkeldiagram

Eerst een tabel maken van de variabele:

```
Naam <- table(Dataset$Variabele)
pie(Naam)
```

Kleuren definiëren:

```
Kleurtjes <- c("red","blue","yellow")
pie(Naam, col=Kleurtjes)
```

of kleuren van de regenboog:

```
pie(Naam, col=rainbow(length(Naam)))
```

Labels aanpassen naar relatieve frequentie:

```
Eigenlabels <- prop.table(Naam)*100
```

Afronden op x cijfers na de komma:

```
Eigenlabels <- round(Eigenlabels,x)
```

%-teken invoegen na een spatie:

```
Eigenlabels <- paste(Eigenlabels,"%", sep=" ")
```

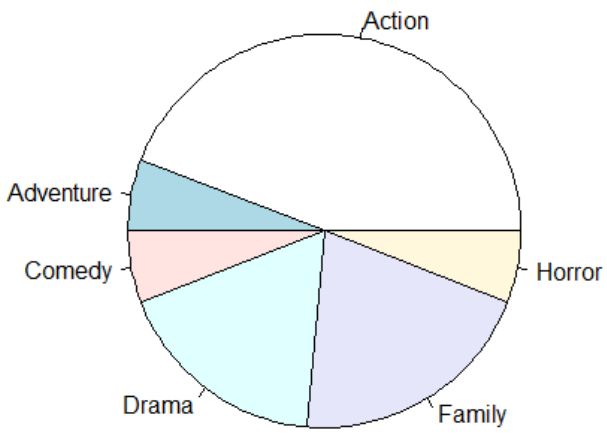
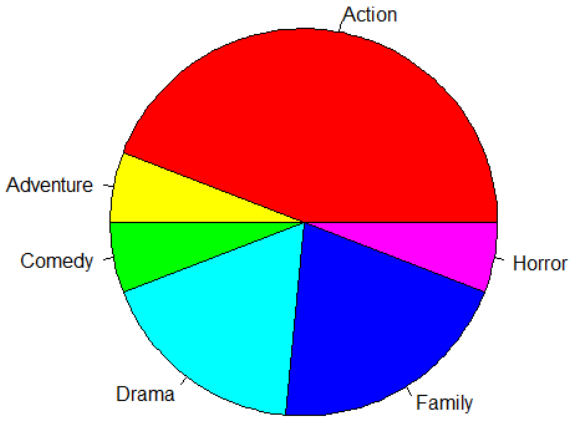
Legende maken:

```
pie(Naam, col=Kleurtjes, labels=Eigenlabels)
legend("topleft",c("x","y",…,"z"), fill=kleurenobject)
```

bv. "kleurtjes" zoals hierboven

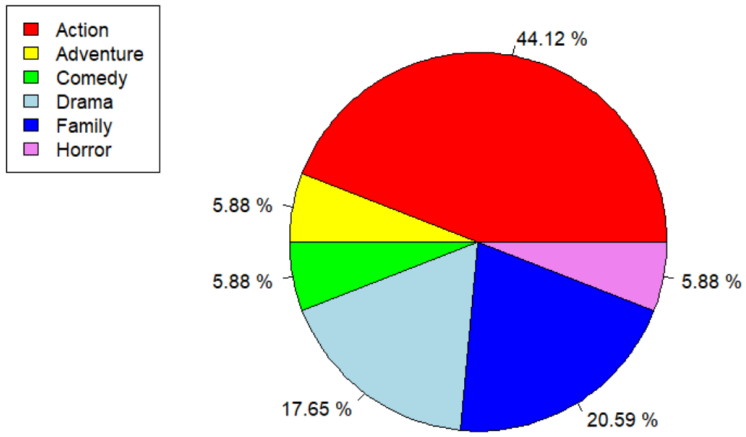
**Voorbeeld vanuit dataset:**

```
# Eerst een tabel maken van de variabele.
> genremoviedata=table(moviedata$GENRE)
> genremoviedata

Action Adventure     Comedy      Drama     Family     Horror
    15          2          2          6          7          2
```

| > pie(genremoviedata) #of dus kort:<br>> pie(table(moviedata$GENRE)) | > Pie(genremoviedata,<br>col=rainbow(length(genremoviedata))) |
|---|---|
|  |  |

**Extra functie van pakket StatSO** (zie deel 2.2 om het pakket te installeren):

```
> library(StatSO)
# De functie 'cirkeldiagram' maakt een cirkeldiagram waarbij de categorieën in een legende staan en de relatieve frequenties bij het cirkeldiagram verschijnen.
> cirkeldiagram(moviedata$GENRE)
```




## 4.4 Stengelbladdiagram

Een enkel stengelbladdiagram:

```
stem(Dataset$Variabele, scale=1)
```

Voor een dubbel stengelbladdiagram hebben we een extra pakket nodig:

```
> install.packages("aplpack")
> library(aplpack)
```

Commando:

```
stem.leaf.backback(Dataset1$Variabele, Dataset2$Variabele, trim.outliers=F)
```

**Voorbeeld vanuit dataset:**

```
> stem(zevendejaar$GPA)
The decimal point is at the |

  0 | 58
  2 | 44689
  4 | 06791259
  6 | 011224922333556666666788899
```

```
  8 | 000022222334789900222334455666 8
 10 | 01678
```

# Door ‘scale = 2’ te nemen als argument zal het diagram ongeveer twee keer zo lang worden als bij de standaard ‘scale=1’. In dit voorbeeld heeft ‘scale = 2’ als extra voordeel dat de numerieke waarden vanuit de dataset eenduidig en correct kunnen worden afgelezen.

```
> stem(zevendejaar$GPA, scale=2)
  The decimal point is at the |

  0 | 5
  1 | 8
  2 | 4
  3 | 4689
  4 | 0679
  5 | 1259
  6 | 0112249
  7 | 22333556666666788899
  8 | 0000222223347899
  9 | 002223344556668
 10 | 01678
```

# dubbel stengelbladdiagram

```
> install.packages("aplpack")
> library(aplpack)
> zevendejaarman=subset(zevendejaar,Gender==2)
> head(zevendejaarman)
    GPA  IQ Gender SelfConcept
1 7.940 111      2          67
2 8.292 107      2          43
3 4.643 100      2          52
4 7.470 107      2          66
6 7.585 115      2          51
7 7.650 111      2          71

> zevendejaarvrouw=subset(zevendejaar,Gender==1)
```

```
> stem.leaf.backback(zevendejaarman$GPA,zevendejaarvrouw$GPA,trim.outliers=F)
_____________________________________________________
  1 | 2: represents 1.2, leaf unit: 0.1
    zevendejaarman$GPA      zevendejaarvrouw$GPA
_____________________________________________________
    1                5|  0 |
    2                7|  1 |
    3                4|  2 |
    6              986|  3 |4                1
    9              860|  4 |7                2
   10                0|  5 |259              5
   13              910|  6 |0024             9
  (13)  9986666554211|  7 |234558889      (9)
   21       832211100|  8 |137899          13
   12      7655443111|  9 |03559            7
    2               75| 10 |17               2
                      | 11 |
_____________________________________________________
n:                  47      31
```

# 5 Centrum- en spreidingsmaten

## 5.1 Centrummaten

### MEDIAAN

```
median(Dataset$Variabele)
```

Indien de variabele nog niet numeriek is: `median(as.numeric(Dataset$Variabele))`

Wanneer een dataset lege velden bevat (NA's), moet R hier rekening mee houden: `na.rm=TRUE`

### GEMIDDELDE

- Som van alle waarnemingen: `sum(Dataset$Variabele)`
- Missende waarden weglaten: `sum(Dataset$Variabele, na.rm=TRUE)`
- Aantal waarnemingen per variabele: `length(Dataset$Variabele)`
- Missende waarden weglaten: `length(na.omit(Dataset$Variabele)`
- Gemiddelde berekenen: `sum / length`
- Rekenkundig gemiddelde: `mean(Dataset$Variabele, na.rm=TRUE)`

### KWARTIELEN, DECIELEN EN PERCENTIELEN

- Kwartielen: `quantile(Dataset$Variabele, na.rm=TRUE)`
- Decielen (bv. 10%): `quantile(Dataset$Variabele, c(.10), na.rm=TRUE)`
- Percentielen (bv. 1%): `quantile(Dataset$Variabele, c(.01), na.rm=TRUE)`

### MODUS

**Extra functie van pakket StatSO** (zie deel 2.2 om het pakket te installeren):

```
> library(StatSO)
# De functie 'modus' bepaalt de modus (of modi) van numerieke of categorische gegevens. Indien er verschillende waarden zijn die het meest voorkomen, dan worden deze allemaal gegeven.

modus(Dataset$Variabele)
```

### SAMENVATTING MET VERSCHILLENDE CENTRUMMATEN

Toont mediaan, gemiddelde, NA's, kwartielen, laagste en hoogste waarde:

```
summary(Dataset$Variabele)
```

**Voorbeeld vanuit dataset:**

```
# mediaan
> median(moviedata$Runtime)
[1] 118

# gemiddelde
> mean(moviedata$Runtime)
[1] 118.0588

# verschillende kwartielen
> quantile(moviedata$Runtime, na.rm=TRUE)
    0%    25%    50%    75%   100%
 84.00 107.75 118.00 133.00 149.00

# modus
> library(StatSO)
> modus(moviedata$Runtime)
[1] 134
> modus(moviedata$GENRE)
[1] "Action"

# Samenvatting toont mediaan, gemiddelde, NA's, kwartielen, laagste en hoogste waarde.
> summary(moviedata$Runtime)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   84.0   107.8   118.0   118.1   133.0   149.0
```

## 5.2 Spreidingsmaten

**VARIATIEBREEDTE**

Kleinste meetwaarde: `min(Dataset$Variabele, na.rm=TRUE)`

Grootste meetwaarde: `max(Dataset$Variabele, na.rm=TRUE)`

Kleinste en grootste meetwaarde: `range(Dataset$Variabele, na.rm=TRUE)`

De variatiebreedte:

```
max(Dataset$Variabele, na.rm=TRUE)- min(Dataset$Variabele, na.rm=TRUE)
```

**INTERKWARTIELAFSTAND**

Variatiebreedte bepalen tussen 1ste en 3de kwartiel:

```
quantile(Dataset$Variabele,c(.75),na.rm=T)-
quantile(Dataset$Variabele,c(.25),na.rm=T)
```

**GEMIDDELDE ABSOLUTE AFWIJKING**

Afwijking t.o.v. het gemiddelde:

```
Dataset$Variabele_afw<-Dataset$Variabele - mean(Dataset$Variabele)
```

Absolute waarde nemen van de afwijkingen (`abs( )`):

```
Dataset$Abs_afw<-abs(Dataset$Variabele_afw)
```

Gemiddelde absolute afwijking:

```
mean(Dataset$Abs_Afw)
```

**(STEEKPROEF)VARIANTIE**

```
var(Dataset$Variabele, na.rm=TRUE)
```

of

Variabele (afwijkingen) kwadrateren en wegschrijven:

```
Dataset$Variabele_gekwadr<-Dataset$Variabele_afw^2
```

Som nemen van de gekwadrateerde variabelen:

```
Kwadratensom<-sum(Dataset$Variabele_gekwadr, na.rm=TRUE)
```

Quotiënt van de som van de kwadraten en het aantal variabelen - 1:

```
Kwadratensom / (length(na.omit(Dataset$Variabele_gekwadr)) - 1)
```

**(STEEKPROEF)STANDAARDAFWIJKING**

Wortel nemen uit een getal : `sqrt( )`

Standaardafwijking : `sd(Dataset$Variabele, na.rm=TRUE)`

**Voorbeeld vanuit dataset:**

```
# variatiebreedte
> max(zevendejaar$IQ)-min(zevendejaar$IQ)
[1] 64
```

```
# standaardafwijking
> sd(zevendejaar$IQ)
[1] 13.17097

# variantie
> var(zevendejaar$IQ)
[1] 173.4745
```

## 5.3 Boxplot

Boxplot van 1 variabele:

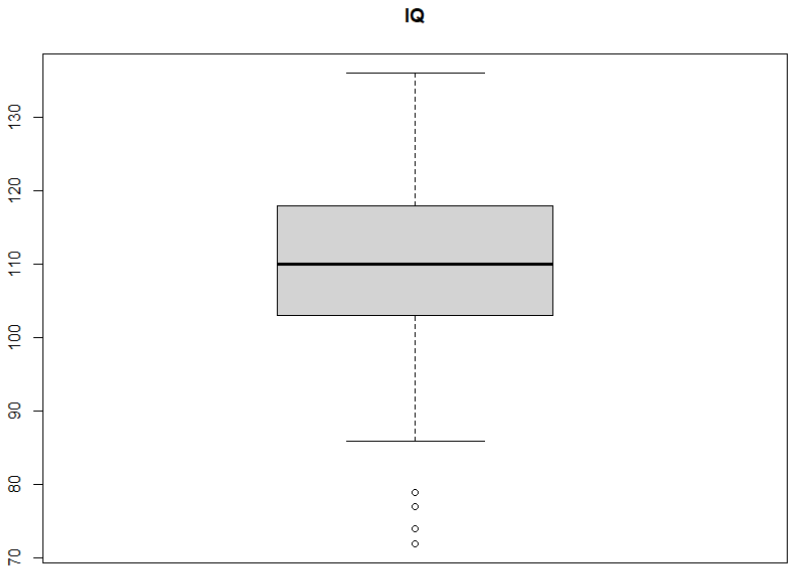
```
boxplot(Dataset$Variabele, horizontal=FALSE)
```

Je kan ook meerdere boxplotten naast mekaar plaatsen waarbij de gegevens van 1 variabele worden gegroepeerd naargelang de waarde van een categorische variabele en waarbij er dan een boxplot wordt gemaakt van de gegevens per categorie:

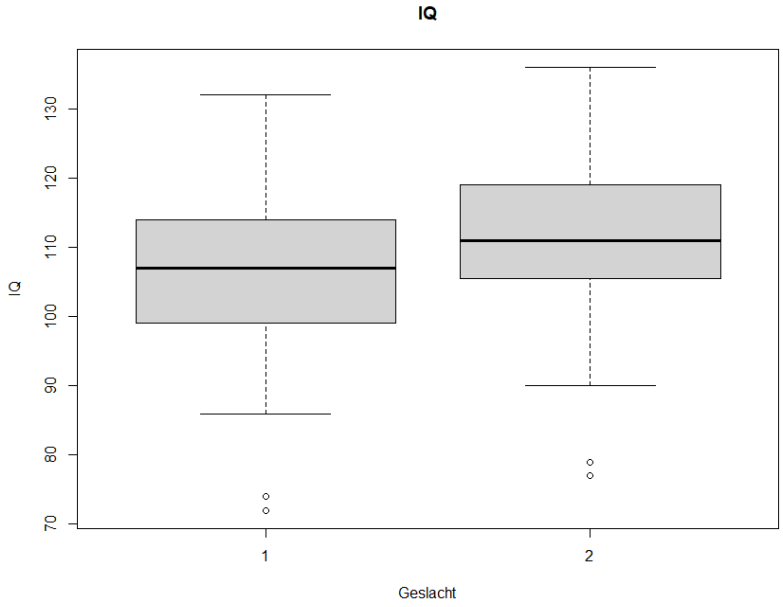
```
boxplot(Dataset$Variabele1~Dataset$Variabele2)
```

**Voorbeeld vanuit dataset:**

```
> boxplot(zevendejaar$IQ, main="IQ")
```



```
> boxplot(zevendejaar$IQ~zevendejaar$Gender, main="IQ", xlab="Geslacht", ylab="IQ")
```



# 6 Het spreidingsdiagram

**PUNTENWOLK**

```
plot(Dataset$Variabele1, Dataset$Variabele2)
```

Extra commando's voor visuele voorstelling van bv. titels:

```
plot(Dataset$Variabele1,Dataset$Variabele2, main="Naam van de dataset",
sub="subtitel",xlab="naam x-variabele", ylab="naam y-variabele")
```

Alle mogelijke spreidingsdiagrammen van 2 variabelen binnen 1 dataset: `pairs(Dataset)`

**TRENDLIJN**

Lineaire regressie: `lm(Dataset$Variabele2~Dataset$Variabele1)`

Lineaire regressie door de oorsprong:

`lm(Dataset$Variabele2~-1+Dataset$Variabele1)` of

`lm(Variabele2~-1+Variabele1, data=dataset)`

Algemeen kwadratisch model:

`lm(Dataset$Variabele2~Dataset$Variabele1+I(Dataset$Variabele1^2))`

Zuiver kwadratisch model: `lm(Dataset$Variabele2~-1+I(Dataset$Variabele1^2))`

Resultaat opslaan in nieuw object: `resultaat=lm(…)`

Trendlijn tekenen: `abline(resultaat)`

Info over de regressie: `summary(resultaat)`

R-kwadraat (de verklaarde variabiliteit/totale variabiliteit in respons, tussen 0 en 1):

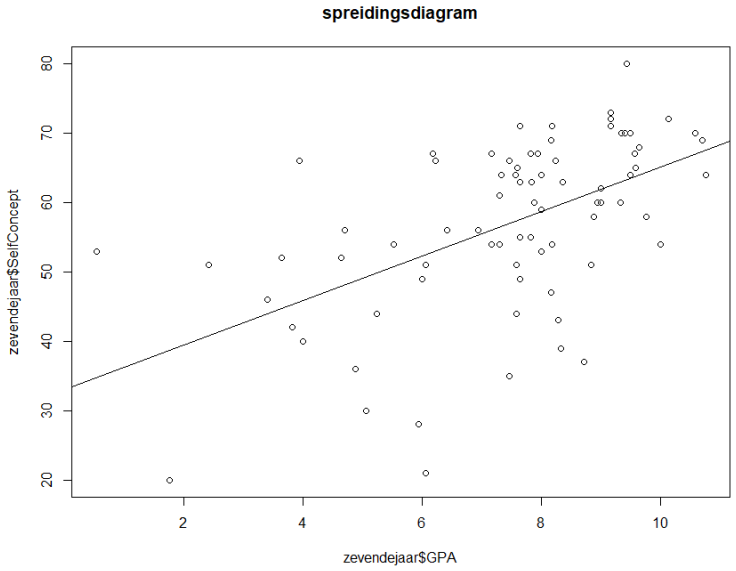
`summary(resultaat)$r.squared`

**CORRELATIE**

Correlatie tussen twee variabelen: `cor(Dataset$Variabele1,Dataset$Variabele2)`

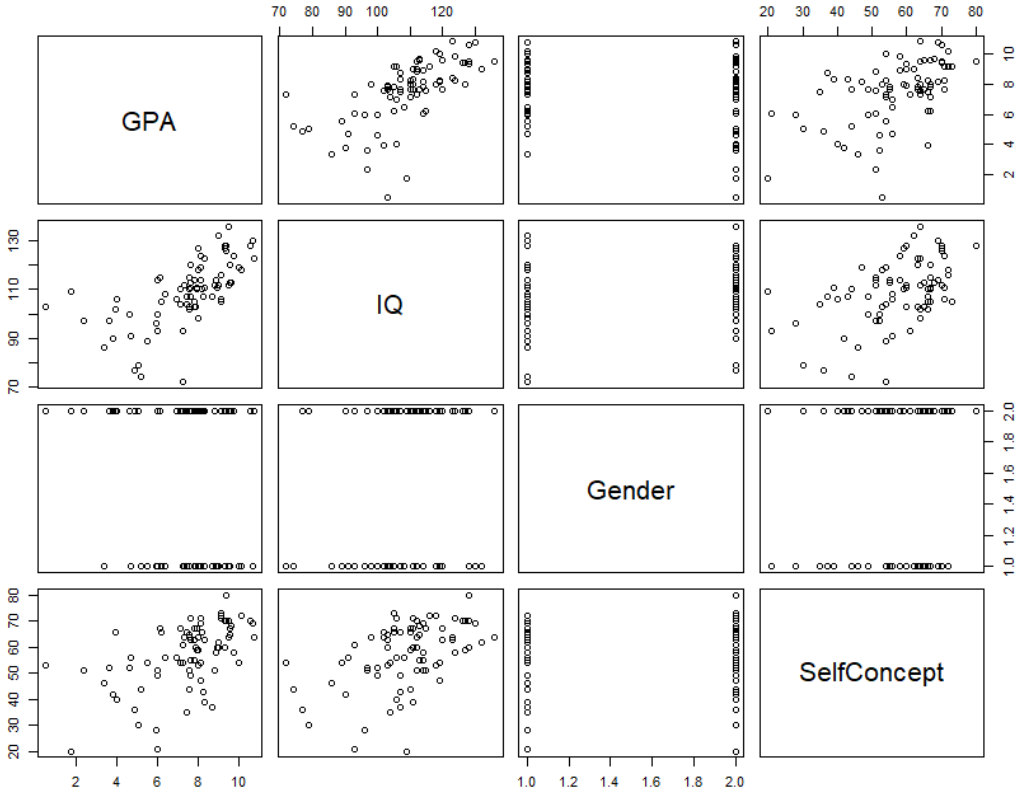
**Voorbeeld vanuit dataset:**

```
> plot(zevendejaar$GPA, zevendejaar$SelfConcept,main="spreidingsdiagram")
> regressielijn=lm(SelfConcept~GPA,data=zevendejaar)
> abline(regressielijn)
```




```
> summary(regressielijn)

Call:
lm(formula = SelfConcept ~ GPA, data = zevendejaar)

Residuals:
    Min      1Q  Median      3Q     Max
-31.511  -4.865   1.630   7.144  20.284

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   33.109      4.408   7.512 9.42e-11 ***
GPA            3.203      0.570   5.620 3.01e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.5 on 76 degrees of freedom
Multiple R-squared:  0.2936,   Adjusted R-squared:  0.2843
F-statistic: 31.59 on 1 and 76 DF,  p-value: 3.006e-07


> cor(zevendejaar$GPA,zevendejaar$SelfConcept)

[1] 0.5418329
```

```
> pairs(zevendejaar)
```



# 7 De normale verdeling

## 7.1 Kansverdeling en kansen

y-waarden van dichtheidsfunctie berekenen:

```
y=dnorm(x, mean = 0, sd = 1)
```

Cummulatieve kans P(X<x) uitrekenen:

```
pnorm(x, mean = 0, sd = 1, lower.tail=TRUE)
```

Kwantielen van normale verdeling uitrekenen:

```
qnorm(p, mean = 0, sd = 1)
```

Willekeurige waarden genereren uit een normale verdeling:

```
rnorm(n, mean = 0, sd = 1)
```

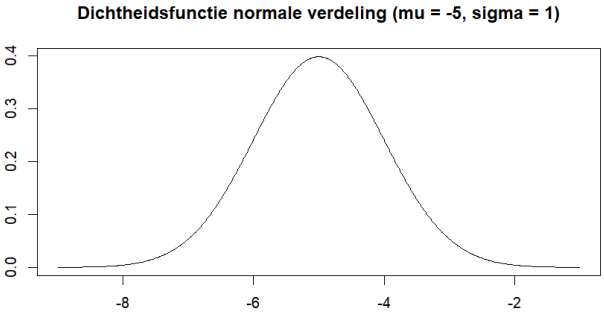
**Extra functies van pakket StatSO** (zie deel 2.2 om het pakket te installeren):

```
> library(StatSO)
# De functie 'grafnorm' tekent de grafiek van de dichtheidsfunctie bij een normale verdeling.
grafnorm(mean = 0, sd = 1)
# De functie 'kansnorm' berekent de kans en geeft de bijhorende oppervlakte bij een normale verdeling.
kansnorm(beginwaarde = -Inf, eindwaarde = +Inf, mean = 0, sd = 1, ...)
```

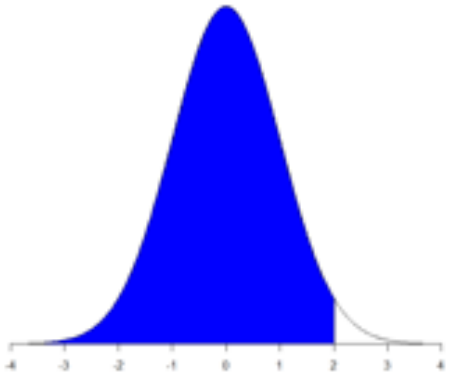
**Voorbeelden:**

```
> pnorm(2, mean=0, sd=1)
[1] 0.9772499
> pnorm(4, mean=5, sd=1, lower.tail=FALSE)
[1] 0.8413447
> qnorm(0.975, mean=2, sd=0.5, lower.tail=TRUE)
[1] 2.979982
> qnorm(0.025, mean=2, sd=0.5, lower.tail=FALSE)
[1] 2.979982
```

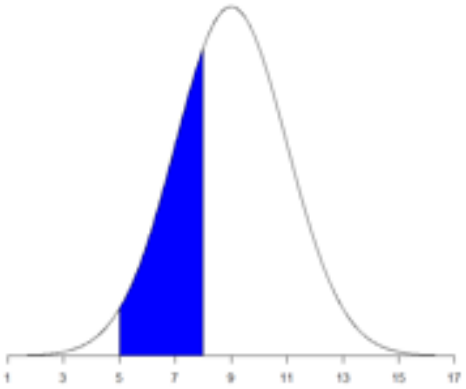
```
> library(StatSO)
> grafnorm(-5,1)
```




```
> kansnorm(eindwaarde=2)
[1] 0.9772499
```



```
> kansnorm(beginwaarde=5, eindwaarde=8, gemiddelde=9, standaardafwijking=2)
[1] 0.2857874
```



## 7.2 Z-score

```
(Dataset$Variabele-mean(Dataset$Variabele,na.rm=T)) /
sd(Dataset$Variabele,na.rm=T)
```

of

```
scale(Dataset$Variabele)
```

**Voorbeeld vanuit dataset:**

```
> (zevendejaar$IQ-mean(zevendejaar$IQ,na.rm=T))/sd(zevendejaar$IQ,na.rm=T)
 [1]  0.157689421 -0.146008723 -0.677480474 -0.146008723  0.385463028  0.461387564
 [7]  0.157689421 -0.905254082 -0.677480474  0.233613957 -0.373782331 -1.512650369
[13] -0.373782331 -0.525631402 -1.360801297  0.385463028  0.385463028 -0.449706866
[19] -0.221933259 -0.297857795  0.309538493  0.005840349 -0.070084187  0.309538493
[25]  1.600255603  1.448406531  1.448406531  0.689161172  0.309538493  0.841010244
…
> scale(zevendejaar$IQ)
               [,1]
 [1,]  0.157689421
 [2,] -0.146008723
 [3,] -0.677480474
 [4,] -0.146008723
 [5,]  0.385463028
 [6,]  0.461387564

[75,]  0.233613957
[76,]  0.385463028
[77,] -1.208952225
[78,] -0.221933259
attr(,"scaled:center")
[1] 108.9231
attr(,"scaled:scale")
[1] 13.17097
```

## 7.3 Q-Q plot

Twee plots op een scherm krijgen: `par(mfrow=c(1,2))`

Resetten (1 plot op scherm): `par(mfrow=c(1,1))`

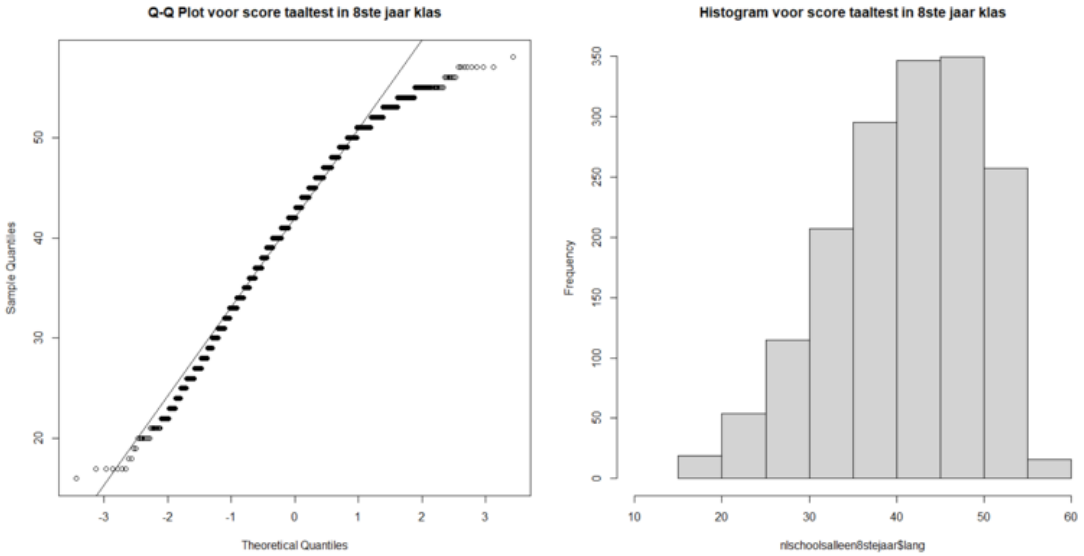
De steekproefgegevens uitzetten t.o.v. de theoretische kwantielen van de standaardnormale verdeling: `qqnorm(Dataset$Variabele, main="titel")`

Theoretische lijn toevoegen: `qqline(Dataset$Variabelen)`

**Voorbeeld vanuit dataset:**

```
# We gebruiken de dataset nlschools die in het pakket "MASS" zit.
> library(MASS)
# We gaan een Q-Q plot opstellen voor de gemiddelde taalscore voor leerlingen van het 8ste jaar die in een aparte klas zaten. Hiervoor gaan we de data dus eerst opsplitsen in twee subdatasets.
```

```
> nlschoolsalleen8stejaar=subset(nlschools,COMB==0)
> nlschoolsgecombineerdeklas=subset(nlschools, COMB==1)

# We maken een Q-Q plot en een histogram.
> par(mfrow=c(1,2))
> qqnorm(nlschoolsalleen8stejaar$lang, main="Q-Q Plot voor score taaltest in 8ste jaar klas")
> qqline(nlschoolsalleen8stejaar$lang)
> hist(nlschoolsalleen8stejaar$lang, main="Histogram voor score taaltest in 8ste jaar klas",
xlim=c(10,60))
> par(mfrow=c(1,1))
```



# 8 De Binomiale verdeling

Kans op 1 enkele waarde:

```
dbinom(x, size, prob)
```

Cummulatieve kans P(X<=x) uitrekenen:

```
pbinom(q, size, prob, lower.tail=TRUE)
```

Kwantielen van binomiale verdeling uitrekenen:

```
qbinom(p, size, prob, lower.tail=TRUE)
```

Willekeurige waarden genereren uit een binomiale verdeling:

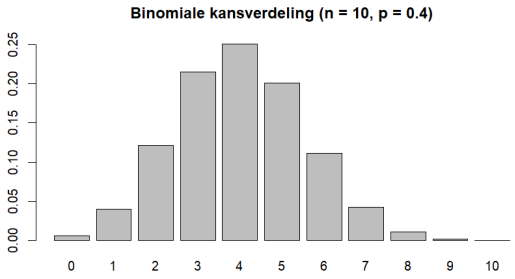
```
rbinom(n, size, prob)
```

**Extra functies van pakket StatSO** (zie deel 2.2 om het pakket te installeren):

```
> library(StatSO)
# Deze functie berekent de kans en geeft de bijhorende oppervlakte bij een binomiale verdeling.
grafbinom(size, prob)
# Deze functie berekent de kans en geeft de bijhorende oppervlakte bij een binomiale verdeling.
kansbinom(beginwaarde, eindwaarde, size, prob)
```

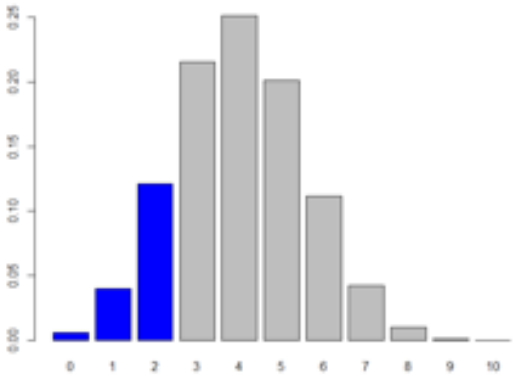
**Voorbeelden:**

```
# Staafdiagram bij B(n=10,p=0.4)
> library(StatSO)
> grafbinom(10,0.4)
```

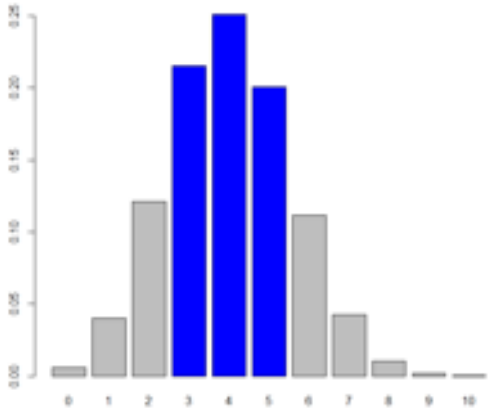



```
# P(X=2) bij B(n=10,p=0.4)
> dbinom(2,10,0.4)
[1] 0.1209324

# P(X<=2) bij B(n=10,p=0.4)
> pbinom(2,10,0.4)
[1] 0.1672898

# kleinste waarde a zodat P(X<=a)>=0.5 bij B(n=10,p=0.4)
> qbinom(0.5,10,0.4)
[1] 4
```

```
> kansbinom(0,2,10,0.4)
[1] 0.1672898
```




```
> kansbinom(3,5,10,0.4)
[1] 0.6664716
```



# 9 Betrouwbaarheidsintervallen en hypothesetoetsen

## 9.1 Voor gemiddeldes met gekende populatiestandaardafwijking

Verdeling (populatiestandaardafwijking $\sigma$ gekend): $\bar{X} \sim N\left(\mu, \frac{\sigma}{\sqrt{n}}\right)$

**BETROUWBAARHEIDSINTERVAL**

Hiervoor hebben we een extra pakket nodig. We installeren dit 1 keer.

```
> install.packages("DescTools")
```

Daarna hoeven we dit pakket enkel bij de start nog in te lezen.

```
> library(DescTools)
```

Commando: `MeanCI(Variabele, sd=… ,conf.level=0.95)`

**Voorbeeld vanuit dataset:**

```
# We willen een 95% betrouwbaarheidsinterval opstellen voor het populatiegemiddelde van een taalscore bij meisjes in het zevende jaar.

> MeanCI(zevendejaarvrouw$GPA,sd=sd(zevendejaarvrouw$GPA),conf.level=0.95)
    mean   lwr.ci   upr.ci
7.696548 7.090788 8.302309
> length(zevendejaarvrouw$GPA)
[1] 31

# Dit geeft hetzelfde resultaat als de behandelde formule vanuit de theorie.
> mean(zevendejaarvrouw$GPA) + c(qnorm(0.025),qnorm(0.975)) *
sd(zevendejaarvrouw$GPA)/sqrt(length(zevendejaarvrouw$GPA))
[1] 7.090788 8.302309
```

**HYPOTHESETOETS**

Hiervoor hebben we een extra pakket nodig. We installeren dit 1 keer.

```
> install.packages("BSDA")
```

Daarna hoeven we dit pakket enkel bij de start nog in te lezen.

```
> library(BSDA)
```

Commando:

```
z.test(Variabele, alternative="two.sided", mu=0, sigma.x=NULL,
conf.level=0.95)
```

**Voorbeeld vanuit dataset:**

```
# We willen nagaan op het 5% significantieniveau of we kunnen besluiten dat het populatiegemiddelde van een taalscore bij meisjes uit het zevende jaar respectievelijk verschillend, groter en kleiner is dan 7. We veronderstellen dat de populatiestandaardafwijking gekend is en gelijk is aan 1,72.

> sd(zevendejaarvrouw$GPA)
[1] 1.720813
> z.test(zevendejaarvrouw$GPA,alternative="two.sided", mu=7, sigma.x=1.72, conf.level=0.95)
      One-sample z-Test
data:  zevendejaarvrouw$GPA
z = 2.2548, p-value = 0.02415
alternative hypothesis: true mean is not equal to 7
95 percent confidence interval:
 7.091074 8.302023
sample estimates:
mean of x
 7.696548
# Merk op dat we als output hetzelfde 95% betrouwbaarheidsinterval krijgen als met het commando MeanCI.
> MeanCI(zevendejaarvrouw$GPA, sd=1.72)
    mean   lwr.ci   upr.ci
7.696548 7.091074 8.302023
> z.test(zevendejaarvrouw$GPA,alternative="greater", mu=7, sigma.x=1.72)
        One-sample z-Test
data:  zevendejaarvrouw$GPA
z = 2.2548, p-value = 0.01207
alternative hypothesis: true mean is greater than 7
95 percent confidence interval:
 7.188418       NA
sample estimates:
mean of x
 7.696548
> z.test(zevendejaarvrouw$GPA, alternative="less", mu=7, sigma.x=1.72)
      One-sample z-Test
data:  zevendejaarvrouw$GPA
z = 2.2548, p-value = 0.9879
alternative hypothesis: true mean is less than 7
95 percent confidence interval:
       NA 8.204678
sample estimates:
mean of x
 7.696548
```

## 9.2 Voor gemiddeldes met ongekende populatiestandaardafwijking

In de praktijk beschikken we meestal niet over de populatiestandaardafwijking. Dus moeten we gebruik maken van de schatting op basis van de steekproef. Dit betekent dat we in de noemer van de z-waarde de populatiestandaardafwijking moeten vervangen door de steekproef-standaardafwijking. Door deze extra bron van variabiliteit moet de standaard normale verdeling vervangen worden door een t-verdeling met n-1 vrijheidsgraden (een verdeling met langere staarten).

Verdeling: $\frac{\bar{X}-\mu}{\frac{s}{\sqrt{n}}} \sim t_{n-1}$

**BETROUWBAARHEIDSINTERVAL EN HYPOTHESETOETS**

We werken hier met één functie voor beiden.

```
t.test(Variabele, alternative="two.sided", mu=0, sigma.x=NULL,
conf.level=0.95)
```

**Voorbeeld vanuit dataset:**

# We willen nagaan op het 5% significantieniveau of we kunnen besluiten dat het populatiegemiddelde van een taalscore bij meisjes uit het zevende jaar respectievelijk verschillend, groter en kleiner is dan 7. We kennen de populatie standaardafwijking nu niet.

```
> t.test(zevendejaarvrouw$GPA,alternative="two.side",mu=7)
      One Sample t-test
data:  zevendejaarvrouw$GPA
t = 2.2537, df = 30, p-value = 0.03168
alternative hypothesis: true mean is not equal to 7
95 percent confidence interval:
 7.065349 8.327748
sample estimates:
mean of x
 7.696548
```

```
> t.test(zevendejaarvrouw$GPA,alternative="greater", mu=7)
> t.test(zevendejaarvrouw$GPA,alternative="less", mu=7)
```

## 9.3 Voor proporties

Verdeling: $\hat{p} \sim N\left(p, \sqrt{\frac{p \cdot (1-p)}{n}}\right)$

**BETROUWBAARHEIDSINTERVAL**

Hiervoor hebben we een extra pakket nodig. We installeren dit 1 keer.

```
> install.packages("DescTools")
```

Daarna hoeven we dit pakket bij de start enkel nog in te lezen.

```
> library(DescTools)
```

Commando:

```
BinomCI(x, n, conf.level=0.95, sides=c("two.sided", "left", "right"),
method=c("wilson", "wald", "waldcc",…))
```

**Voorbeeld vanuit dataset:**

```
# We willen een BTI opstellen voor het populatiepercentage meisjes dat een IQ hoger dan 115 heeft in de dataset zevendejaar. Hiervoor tellen we eerst het aantal meisjes dat een IQ hoger dan 115 heeft in de steekproef.
> aantalvrouwIQ=sum(zevendejaarvrouw$IQ>=115)
> aantalvrouwIQ
[1] 6

# We tellen ook het aantal meisjes in de steekproef.
> totaalaantalvrouw=length(zevendejaarvrouw$IQ)
> totaalaantalvrouw
[1] 31
> BinomCI(aantalvrouw,totaalaantalvrouw, method="wald")
           est     lwr.ci    upr.ci
[1,] 0.1935484 0.05447271 0.3326241

# Door de methode 'Wald' te gebruiken verkrijgen we het betrouwbaarheidsinterval dat overeenkomt met de theorie vanuit het secundair onderwijs.
> steekproefpercentagevrouw = aantalvrouwIQ/totaalaantalvrouw
> steekproefpercentagevrouw + c(qnorm(0.025),qnorm(0.975)) * sqrt(steekproefpercentagevrouw * (1-steekproefpercentagevrouw) /totaalaantalvrouw)
[1] 0.05447271 0.33262406
```

**HYPOTHESETOETS**

```
prop.test(x,n,p=NULL,alternative=c("two.sided", "less",
"greater"),conf.level=0.95, correct=TRUE/FALSE)
```

Men geeft best als argument 'correct=FALSE' mee. Zo komen leerlingen dezelfde p-waarde uit als deze die men bekomt vanuit de onderliggende theorie. Indien je 'correct=TRUE' meegeeft, dan gaat men nog een bepaalde continuïteitscorrectie toepassen.

**Voorbeeld vanuit dataset:**

```
# We willen testen of 25% van de jongens in alle zevendejaars een IQ heeft van minstens 115
(tweezijdige test). Hiervoor tellen we eerst het aantal jongens dat een IQ hoger dan 115 heeft in de
steekproef.
> aantalmanIQ=sum(zevendejaarman$IQ>=115)
> aantalmanIQ
[1] 17
We tellen ook het totaal aantal jongens in de steekproef.
> totaalaantalman=length(zevendejaarman$IQ)
> totaalaantalman
[1] 47
# hypothesetoets voor proporties
> prop.test(aantalmanIQ,n=totaalaantalman, p=0.25, alternative="two.sided", correct=FALSE)

        1-sample proportions test without continuity correction
data:  aantalmanIQ out of totaalaantalman, null probability 0.25
X-squared = 3.1277, df = 1, p-value = 0.07697
alternative hypothesis: true p is not equal to 0.25
95 percent confidence interval:
 0.2396621 0.5046410
sample estimates:
        p
0.3617021

# Merk op dat enkel de p-waarde overeenkomt met wat de leerlingen in de theorie hebben geleerd.
Het betrouwbaarheidsinterval wijkt hier licht af van de theorie en van wat de functie "BinomCI"
geeft.
> BinomCI(aantalmanIQ,totaalaantalman, method="wald")
           est    lwr.ci    upr.ci
[1,] 0.3617021 0.2243339 0.4990704
```

```
# We willen nu testen of meer dan 25% van de jongens in alle zevendejaars een IQ heeft van minstens 115 (eenzijdige test).
> prop.test(aantalmanIQ,n=totaalaantalman, p=0.25, alternative="greater", correct=FALSE)

        1-sample proportions test without continuity correction

data:  aantalman out of totaalaantalman, null probability 0.25
X-squared = 3.1277, df = 1, p-value = 0.03849
alternative hypothesis: true p is greater than 0.25
95 percent confidence interval:
 0.2568757 1.0000000
sample estimates:
        p
0.3617021
```

# 10 Oefeningen

## 10.1 Oefening 1

Gebruik de dataset fancost99

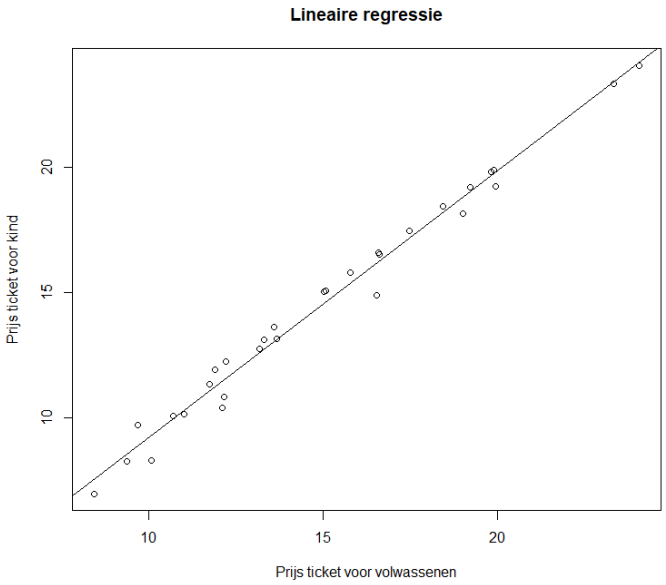
a) Maak spreidingsdiagrammen van alle variabelen met het commando pairs.

b) Teken de lineaire regressielijn die de prijs voor de tickets voor kinderen voorspelt a.d.h.v. de prijs voor de volwassenen. Geef ook het voorschrift van deze regressielijn.

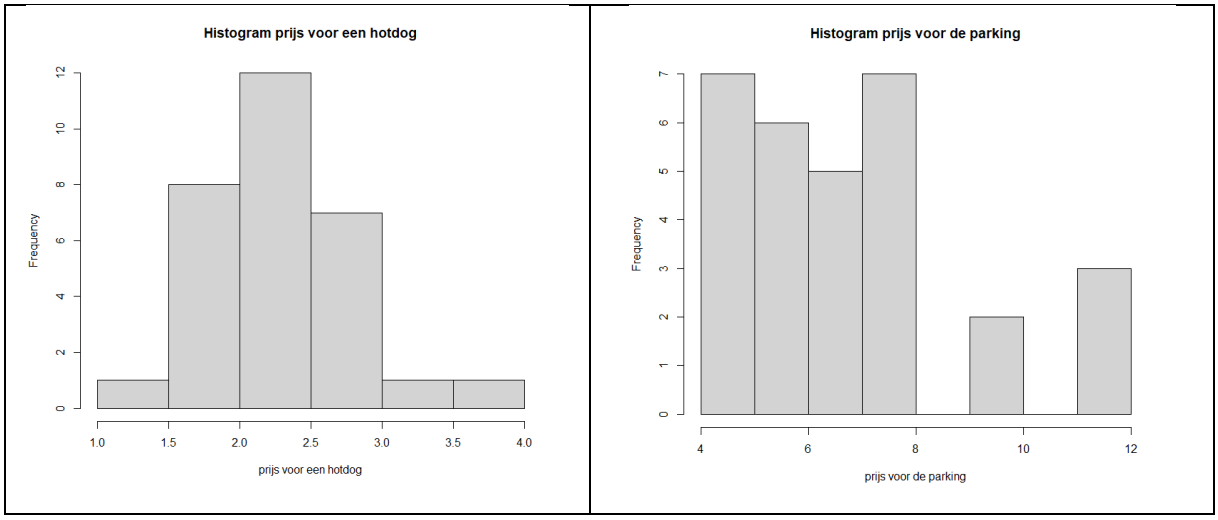
c) Wat is de correlatiecoëfficiënt hier?

Oplossing:

```
> pairs(fancost)
> plot(fancost$Tickets,fancost$kids, main="Lineaire regressie", xlab= "Prijs ticket voor volwassenen", ylab="Prijs ticket voor kind")
> regressielijn=lm(kids~Tickets,data=fancost)
> summary(regressielijn)

      Call:
      lm(formula = kids ~ Tickets, data = fancost)

      Residuals:
          Min       1Q   Median       3Q      Max
      -1.2867 -0.2342  0.1450  0.3760  0.8237

      Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
      (Intercept) -1.47917    0.38733  -3.819 0.000681 ***
      Tickets      1.06751    0.02492  42.846  < 2e-16 ***
   ---
   Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

   Residual standard error: 0.5554 on 28 degrees of freedom
   Multiple R-squared:  0.985, Adjusted R-squared:  0.9844
   F-statistic:  1836 on 1 and 28 DF,  p-value: < 2.2e-16

> abline(regressielijn)
> cor(fancost$kids,fancost$Tickets)
      [1] 0.9924599
```




```
> hist(fancost$hot_dog, main="Histogram prijs voor een hotdog",xlab="prijs voor een hotdog")
> hist(fancost$tix_parking, main="Histogram prijs voor de parking", xlab="prijs voor de parking")
```




## 10.2 Oefening 2

Gebruik de dataset place92.

a) Bereken het gemiddelde, de mediaan, de standaardafwijking, het eerste en derde kwartiel.

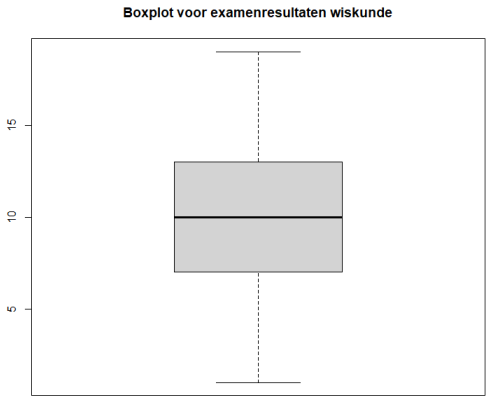
b) Maak een boxplot.

c) Maak een qqplot.

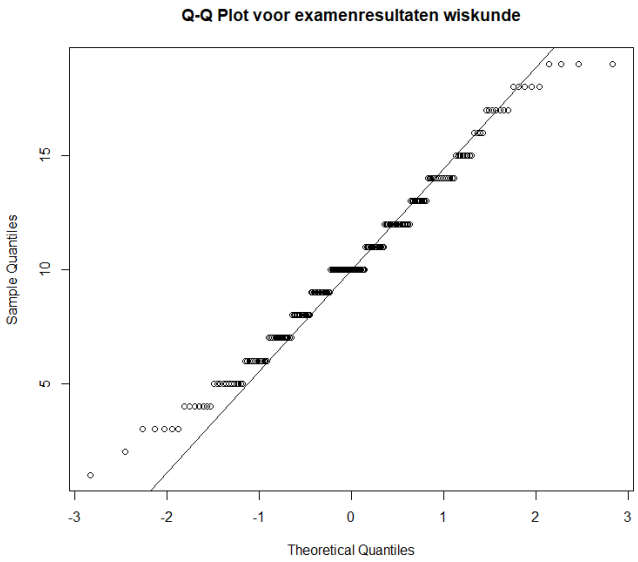
Oplossing:

```
> summary(place92)
     score
 Min.   : 1.00
 1st Qu.: 7.00
 Median :10.00
 Mean   :10.22
 3rd Qu.:13.00
 Max.   :19.00
> sd(place92$score)
[1] 3.858725
```

```
> boxplot(place92$score, main="Boxplot voor examenresultaten wiskunde")
```




```
> qqnorm(place92$score, main="Q-Q Plot voor examenresultaten wiskunde")
> qqline(place92$score)
```



## 10.3 Oefening 3

Gebruik de datasets westernstate en easternstate.

a) Maak een tweezijdige stengelbladdiagram diagram voor het percentage vrouwen. Is er een uitschieter?

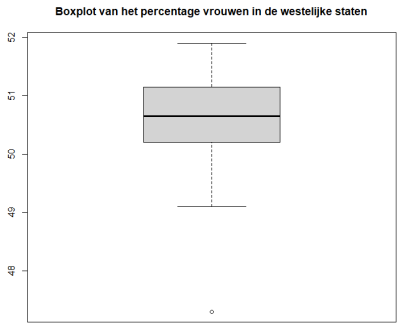
b) Vind je deze uitschieter ook terug als je een boxplot maakt?

Oplossing:

```
> library(aplpack)
> stem.leaf.backback(westernstate$`percentage female`, easternstate$`percentage female`,trim.outliers=F)
```

```
_____________________________________________________
  1 | 2: represents 1.2, leaf unit: 0.1
westernstate$`percentage female`
                          easternstate$`percentage female`
_____________________________________________________
   1               3| 47* |
                    | 47. |
                    | 48* |
                    | 48. |
   3               11| 49* |
   4                9| 49. |
   9            43220| 50* |
  (7)         8887655| 50. |
   8              3300| 51* |000134            6
   4              9886| 51. |5555555666788   (13)
                      | 52* |0001112           7
                      | 52. |
                      | 53* |
_____________________________________________________
n:                  24       26
_____________________________________________________
```

```
> boxplot(westernstate$`percentage female`, main="Boxplot van het percentage vrouwen in de westelijke staten")
```



## 10.4 Oefening 4

Gebruik de dataset marriagesages.

a) Bij hoeveel van deze 100 huwelijken is de bruid jonger dan de bruidegom?

b) Geef een 90% betrouwbaarheidsinterval voor de proportie van alle huwelijken waarbij de bruid jonger is dan de bruidegom.

c) Ga na of deze gegevens de stelling ondersteunen dat de bruid jonger is dan de bruidegom in meer dan de helft van alle huwelijken.

d) Maak een nieuwe variabele die het verschil in leeftijd tussen de bruidegom en de bruid bevat. De nulhypothese veronderstelt dat dit verschil gelijk is aan 0, en de alternatieve hypothese veronderstelt dat het verschil groter is dan 0. Test deze hypothese.

Oplossing:

```
> proportiejonger=mean(marriagesages$vrouw<marriagesages$man)
> somjonger=sum(marriagesages$vrouw<marriagesages$man)
> aantalhuwelijken=length(marriagesages$vrouw)
> library(DescTools)
> BinomCI(somjonger,aantalhuwelijken, conf.level=0.9,method="wald")
      est    lwr.ci    upr.ci
[1,] 0.67 0.5926569 0.7473431
> prop.test(somjonger,n=aantalhuwelijken, p=0.5, alternative="greater",correct=FALSE)
        1-sample proportions test without continuity correction

data:  somjonger out of aantalhuwelijken, null probability 0.5
X-squared = 11.56, df = 1, p-value = 0.0003369
alternative hypothesis: true p is greater than 0.5
95 percent confidence interval:
 0.5890729 1.0000000
sample estimates:
   p
0.67
> t.test(verschilleeftijd,alternative="greater", mu=0)
        One Sample t-test

data:  verschilleeftijd
t = 3.8045, df = 99, p-value = 0.000123
alternative hypothesis: true mean is greater than 0
95 percent confidence interval:
 1.082065      Inf
sample estimates:
mean of x
     1.92
```

## 10.5 Oefening 5

Gebruik de dataset genphys en genphysanders.

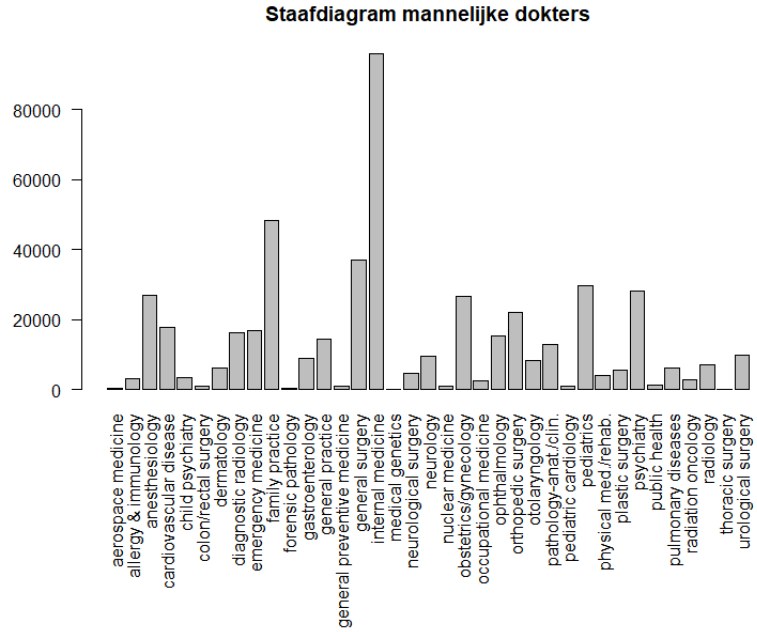
a) Maak een barplot per geslacht.

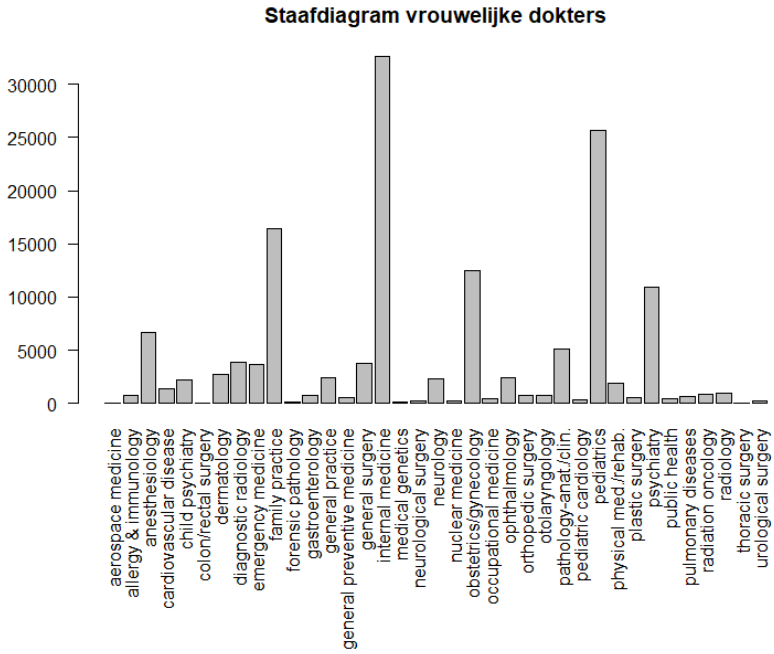
b) Maak een cirkeldiagram per geslacht.

c) Je kan ook een barplot maken waar Man en Vrouw direct naast elkaar staan. Hiervoor moet de data echter anders gerangschikt staan. Gebruik hiervoor de dataset genphysanders. We lezen hiervoor een extra pakket in, namelijk ggplot2.

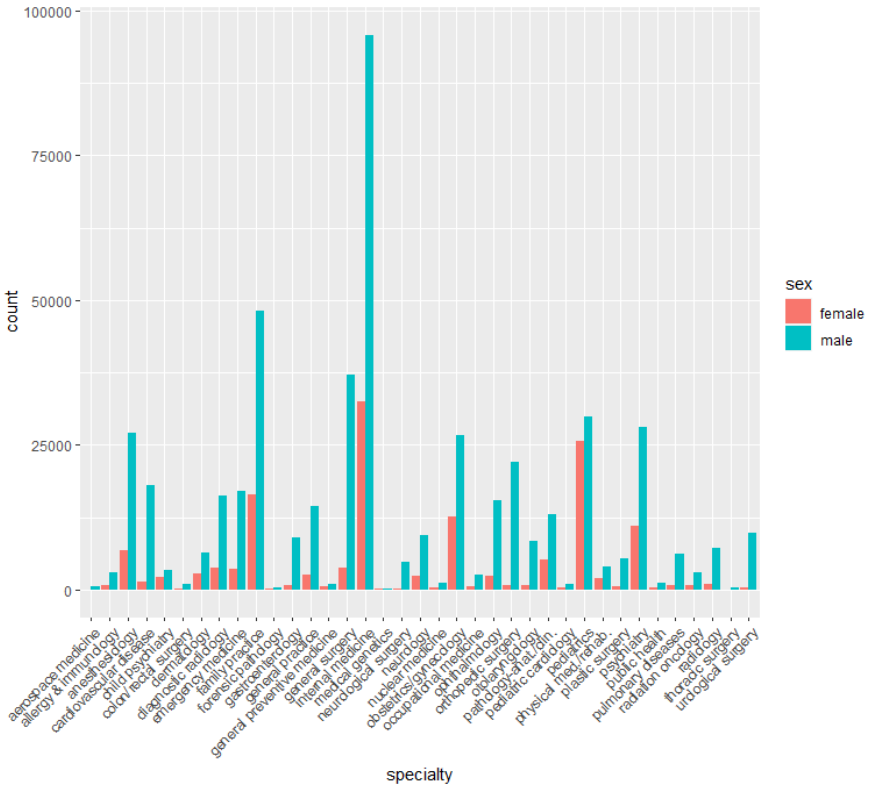
Oplossing:

```
> par(mar=c(15,4,4,2))
> barplot(genphys$men, main="Staafdiagram mannelijke dokters",
names.arg=rownames(genphys),las=2)
> barplot(genphys$women, main="Staafdiagram vrouwelijke dokters",
names.arg=rownames(genphys),las=2)
```

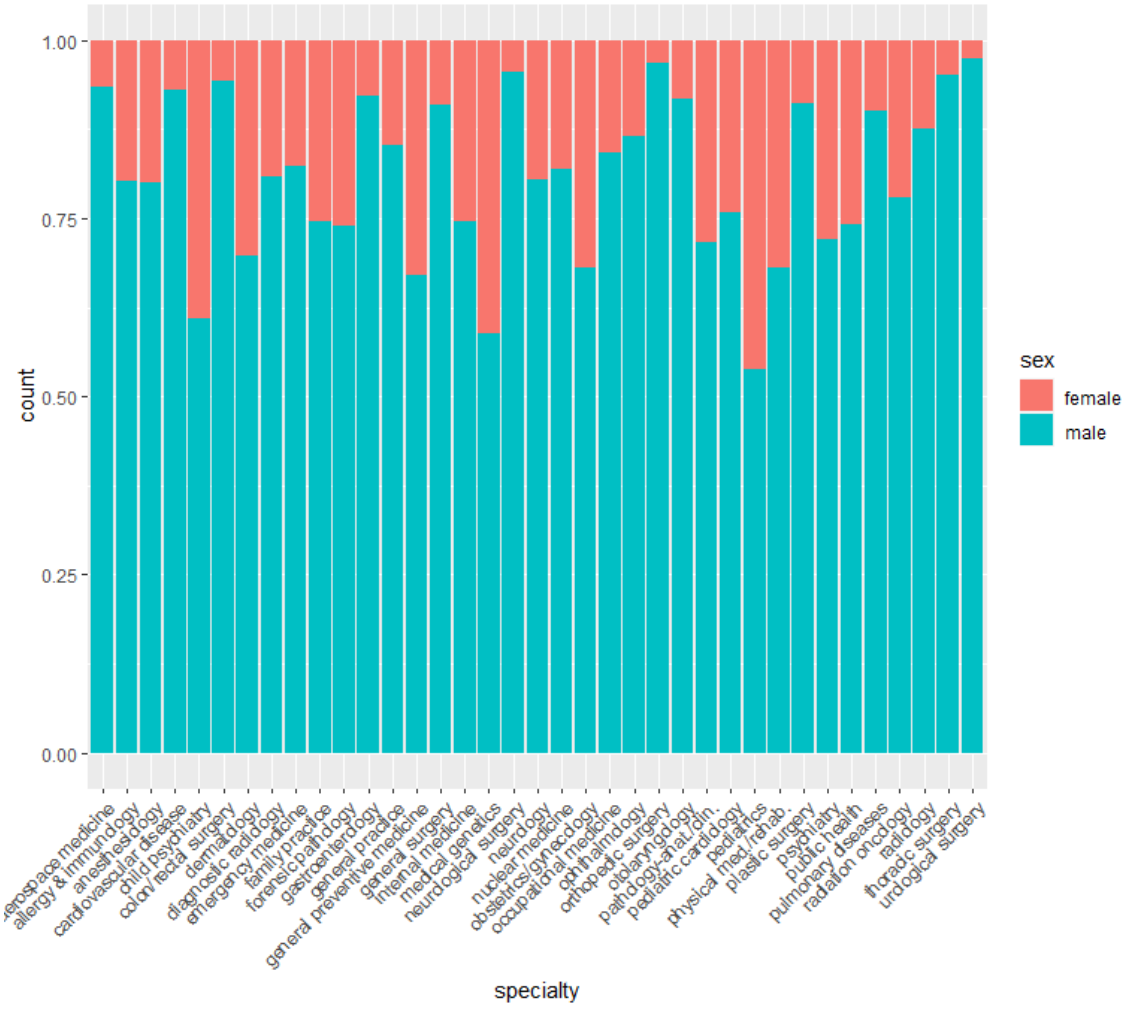





```
> install.packages("ggplot2")
> library(ggplot2)
> #dodge wil zeggen naast elkaar
> ggplot(genphysanders,aes(x=specialty,y=count,fill=sex))+geom_col(position="dodge")+
+   theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```



```
> #fill is om met relatieve frequenties te werken
> ggplot(genphysanders,aes(x=specialty,y=count,fill=sex))+geom_col(position="fill")+
+   theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```



# 11 Datasets ter inspiratie

Datasets uit Moore, McCabe en Craig (2021):

https://www.macmillanlearning.com/studentresources/college/collegebridgepage/ips10e.html

Datasets uit Rumsey (2021):

https://www.wiley.com/en-us/Statistics+II+For+Dummies%2C+2nd+Edition-p-9781119827399

Datasets uit Rossman en Chance (2011):

https://bcs.wiley.com/he-bcs/Books?action=resource&bcsId=6830&itemId=047054208X&resourceId=26588

# 12 Bronnen

De Maeyer, S., Ardies, J., Coertjens, L., & Kavadias, D. (2011). *Univariate statistiek voor de menswetenschappen. Een openleerpakket in R*. Academia Press.

De Maeyer, S., Coertjens, L., & Ardies, J. (2012). *Bivariate en multivariate statistiek met R: Een open leerpakket in R*.

Moore , D.S., McCabe, G.P., and Craig B.A. (2021). *Introduction to the Practice of Statistics* (10th Edition), W.H. Freeman & Company.

Rossman, A.J. and Chance, B.L. (2001). *Workshop Statistics: Discovery with Data*. (2th edition), John Wiley & Sons.

Rumsey, D.J. (2021). *Statistics II For Dummies*, John Wiley & Sons.

# 13 Gebruikte datasets: informatie

**Movie dataset** (Rumsey, 2021). (Excel-bestand)

In deze dataset werden films opgenomen die in 2018 meer dan $100,000,000 opbrachten aan de totale Amerikaanse box office-inkomsten. 34 films voldeden hier aan. De volgende variabelen beschrijven de data:

Naam: Naam van de film

Rang (inkomsten): waarbij 1= de hoogste Amerikaanse box office-inkomsten voor 2018.

Releasedatum: dag en maand waarop de film werd uitgebracht

Beoordeling: de beoordeling door de Motion Picture Association van de inhoud van de film. (PG betekent bijvoorbeeld aanbevolen ouderlijk toezicht; zie motionpictures.org voor meer details).

Genre: overheersend type filmcategorie (bv. Actie, drama, horror, komedie).

Speelduur: Aantal minute dat de film duurt.

Dagen: het aantal dagen dat de film in de Amerikaanse bioscopen was

Theaters: het aantal bioscopen waarin de film in de VS is vertoond (een paar films beschikten niet over deze informatie, daarom is dit leeg)

Budget: het geldbedrag dat de film heft gekost om te maken.

Openingsweekend: de hoeveelheid geld die is verdiend in het eerste weekend dat de film in de Amerikaanse bioscopen verscheen.

Amerikaanse inkomsten: de hoeveelheid geld die de film heeft verdiend tijdens de periode dat deze in Amerikaanse theaters te zien was (hou er rekening mee dat sommige films van 2018 tot 2019 in de bioscoop te zien waren).

**Zevende leerjaar (**Moore, McCabe and Craig, 2021). (R-bestand)

In deze dataset worden gegevens weergegeven van 78 studenten in het zevende leerjaar van een plattelandsschool in het Midwesten. De onderzoeker was geïnteresseerd in de relatie tussen het “zelfbeeld” van de studenten en hun intellectuele prestaties.

De data bestaat uit:

GPA: cijfergemiddelde

IQ: de score voor een standard IQ-test

Gender: gecodeerd met een 1 voor vrouwelijk en een 2 voor mannelijk.

SelfConcept: De score van elke student op de Piers-Harris Children's Self-Concept Scale (een psychologische test die door de onderzoeker is uitgevoerd)

**In RStudio**

Ook in RStudio zelf vind je meerdere pakketten die extra datasets bevatten zoals package "datasets" en "MASS".

**Genphys99** (Rossman and Chance, 2001)

In deze dataset wordt er van 37 specialiteiten in de geneeskunde een opsplitsing gemaakt naar hoeveel mannelijke en vrouwelijke artsen er in deze takken van de geneeskunde actief zijn in 1997. Deze data komt uit "1999 World Almanac and Book of Facts".

**Fancost** (Rossman and Chance, 2001)

De gegevens werden verzameld door een marketing onderzoeksteam om een zicht te krijgen op de kosten voor het bijwonen van Major League Baseball-wedstrijden in 1999. De onderzochte variabelen zijn: de prijs van kaartjes voor volwassenen, de prijs van kinderkaartjes, de parkeerkost, de prijs van het programma, de prijs voor een medium dop, prijs voor klein bier, hoeveel ounces er in een klein biertje zitten, de prijs voor een kleine frisdrank, hoeveel ounces er in een kleine frisdrank zitten en de prijs voor een middelgrote hotdog.

**Place92** (Rossman and Chance, 2001)

De afdeling Wiskunde en Informatica van Dickinson College geeft elk najaar een examen aan eerstejaarsstudenten die van plan zijn calculus te gaan volgen. De scores op het examen worden gebruikt om te bepalen in welk niveau van calculus een student moet worden geplaatst. Het examen bestaat uit 20 meerkeuzevragen. In deze dataset vindt men de scores terug van de 213 studenten die in 1992 het examen hebben afgelegd.

**States** (Rossman and Chance, 2001)

Welke staten hebben een hoger percentage van vrouwelijke inwoners en welke staten hebben doorgaans een hoger aantal autodiefstallen. Dit werd in 1990 voor de 50 staten van America verzameld. De data worden opgesplitst in 2 datasets nl. **Westernstate** en **Easternstate** naargelang de ligging t.o.v. de Mississippi rivier.

**Marriage_Ages** (Rossman and Chance, 2001) (zelf ingeven)

In deze tabel vinden we de leeftijd terug van 100 koppels die in 1993 een huwelijksvergunning hadden aangevraagd in Pennysylvania.