Spreadsheets analyseren met Open Source Software:

R en OpenOffice.org Calc

Door: Arie Twigt

Inleiding

In deze handleiding wordt u geleerd hoe u spreadsheets en andere data kunt analyseren met open source software. Open source software is gratis, kunt u aanpassen naar uw eigen wensen en is constant in ontwikkeling. U wordt geleerd data op een eenvoudige en snelle manier te kunnen analyseren. Met de methode die in deze handleiding geleerd, kunt u uitgebreide handelingen bij uw spreadsheets sneller en eenvoudiger uitvoeren. In deze handleiding wordt gewerkt met de volgende twee open source softwarepakketten:

- R
- OpenOffice.org Calc

Achtergrondinformatie R.

R is een open source software pakket en programmeertaal die ontwikkeld is door Ross Ihaka en Robert Gentleman aan de universiteit van Auckland in Nieuw-Zeeland. Ontwikkelen. R heeft een eigen programmeertaal en heeft de nadruk op het toepassen van statistische functies. Met R kunt u met data eenvoudige handelingen doen zoals gemiddelden berekenen, maar ook uitgebreide voorspellingsmethoden. Omdat R open source is, kan iedereen R op zijn eigen manier bewerken. R is constant aan het groeien omdat er steeds meer ontwikkelaars aansluiten om nieuwe toepassingen voor R, meestal gratis, aan te bieden.

Achtergrondinformatie OpenOffice.org Calc

OpenOffice.org Calc is een van de programma's die in het softwarepakket van OpenOffice.org zit. Het programma OpenOffice.org Calc is vergelijkbaar met Microsoft Excel. Ook OpenOffice.org Calc is een open source programma en daarom gratis te downloaden en aan te passen naar uw eigen wensen. In deze handleiding wordt gekozen voor OpenOffice.org Calc omdat het veel mogelijkheden bied om data te bewerken. Daarbij is het een goedkoop alternatief tegenover andere softwarepakketten.

Deze handleiding is bestemd voor personen die Microsoft Windows en/of Mac OS X gebruiken.

Inhoudsopgave

De software downloaden

Zoals bij de inleiding al is aangegeven, wordt er in deze handleiding gebruik gemaakt van de softwarepakketten *R* en *OpenOffice.org Calc*. In dit hoofdstuk wordt uitgelegd hoe u deze softwarepakketten kunt downloaden.

R downloaden.

1. Tik de URL *www.r-project.org* in.
2. Aan de linkerkant van de website (Onder het logo van R) vindt u onder de dikgedrukte tekst **Download, Packages** de link CRAN, klik hier op.
3. De pagina **CRAN Mirrors** verschijnt. Zoek het land op waarin u zich bevindt, in dit geval *Netherlands*. U kunt nu kiezen tussen de mirror http://cran.xl-mirror.nl/ van XL-Data Amsterdam en http://cran-mirror.cs.uu.nl/ van Utrecht University. Klik op een van deze mirrors, het maakt niet uit welke mirror u kiest. In deze handleiding is voor de Utrecht University mirror gekozen.
4. Het venster The **Comprehensive R Archive Network** verschijnt. In het venster **Download and install R** kiest u voor het besturingssysteem dat u gebruikt.
5. **Bij Windows:**

   Klik in het venster **Download and install R** op Download R for Windows. Kies op de pagina **R for Windows** voor base en **install R for the first time**.
   De pagina **R-2.152** for Windows (32/64bit) verschijnt. Klik op Download R 2.15.2 for Windows. Klik op het venster R.2.15.2.exe openen OK.
   Als u deze stappen heeft uitgevoerd verschijnt het bestand **R-2.15.2-win.exe** in uw map met downloads.

   Bij Mac OS X:
   Klik in het venster **Download and install R** op Download R for MacOS X. Kies op de pagina **R for Mac OS X** onder het gedeelte **Files:** voor R-2.15.2.pkg (latest version). Klik op het venster R.2.15.2.pkg openen OK.
   Als u deze stappen heeft uitgevoerd, verschijnt het bestand **R-2.15.2.pkg** in uw map met downloads. Open dit bestand door de software op uw systeem te installeren. De eenvoudige stappen die u daarvoor moet uitvoeren kunt u van uw scherm af lezen.

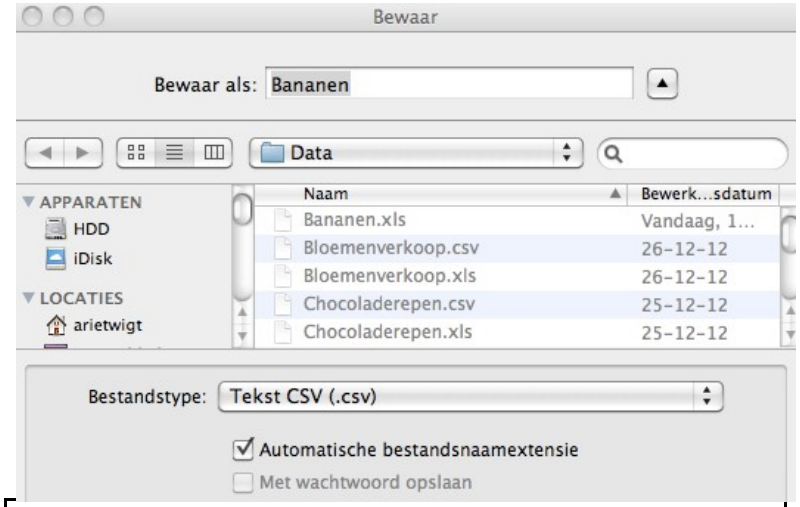
OpenOffice.org Calc downloaden

1. Tik de URL www.openoffice.org/download/other.html in . Er verschijnt een grote tabel met waarin u kunt kiezen voor verschillende talen en platforms om OpenOffice voor te downloaden.
2. In de derde tabel staat de taal *Dutch / Nederlands*, kies daarbij voor uw gewenste besturingssysteem (Windows of Mac OS X) door op Download te klikken.
3. Een nieuwe pagina wordt geopend. Boven aan de pagina staat dat u een paar seconden moet wachten voordat uw download tevoorschijn komt.
4. Het venster om OpenOffice te openen/downloaden verschijnt. Kies voor uw gewenste instellingen en klik op OK.
5. Als u deze stappen heeft uitgevoerd verschijnt het bestand Apache_OpenOffice_incubating_*versienaam*_*platformnaam*_install_nl in de map waar de downloads van uw browser terecht komen. Klik op dit bestand. Aan de hand van de vensters die verschijnen, kunt u uw gewenste instellingen kiezen en OpenOffice installeren op uw systeem.

Deel 1: Eenvoudige handelingen

In het eerste deel van deze handleiding worden de eenvoudige handelingen van R besproken. Dit zijn handelingen zoals het importeren van bestanden, eenvoudige rekensommen met de data en het eenvoudig visualiseren van data.

Voor het eerste deel van deze manual wordt het bestand 'Bloemenverkoop.xls' gebruikt. R kan, zonder speciaal geïnstalleerde packages, geen xls-bestanden (Excel-Bestanden) lezen. Hiervoor moet een xls-bestand opgeslagen worden als.csv bestand, het standaard bestandtype dat R kan lezen. Met Open Office gaat dit vrij eenvoudig:



Kies bij bestandtype voor *Tekst CSV (.csv)*

Veldscheidingsteken voor *(,)* en als *Tekstscheidingsteken* voor (")

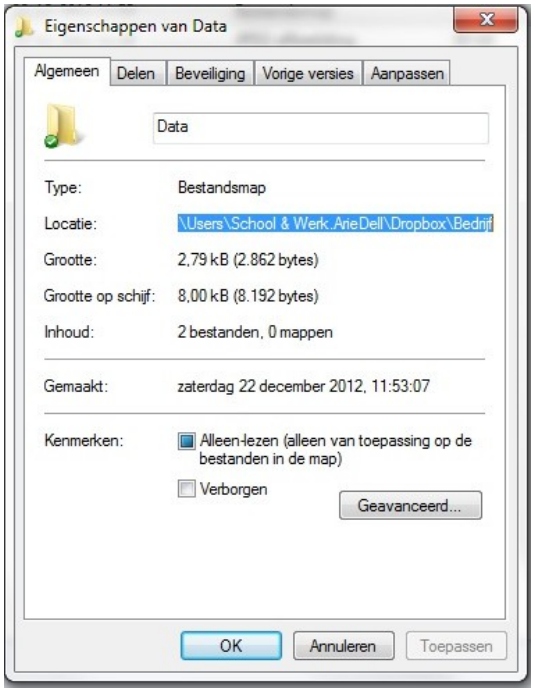
Bloemenverkoop.xls is nu geconverteerd naar een CSV-bestand en heet nu *Bloemenverkoop.csv*[1].

[1] Het xls-bestand blijft behouden.

1.1 De Working Directory instellen.

Als u R opent, komt de zogenaamde R console tevoorschijn. Dit is het venster waarin u werkt met R.

Voor het werken met R moet er een zogenaamde *Working Directory* aangemaakt worden. Dit is een gewone map die simpelweg aangemaakt kan worden door een rechtermuisklik en voor *Nieuwe map* te kiezen in Windows Verkenner of Finder in OS. In deze map, de Working Directory[2], moeten alle (data)bestanden worden geplaatst waar u met R mee wilt werken. Als deze map eenmaal is aangemaakt, geeft u in R aan dat deze aangemaakte map als Working Directory gebruikt wordt. Dit doet u als volgt:



Afbeelding 1 De locatie van de map in het venster *Eigenschappen* vinden.

1. **Huidige Working Directory weergeven:**
   Voer de code **getwd()** bij de R Console in,na het drukken van Enter laat R de locatie van de *Working Directory* die op dit moment gebruikt wordt.

2. **Locatie van de nieuwe Working Directory opzoeken:**
   Als dit de Working Directory is die u wilt gebruiken hoeft u niets meer te wijzigen. Als u een andere Working Directory wilt gebruiken dan R weergeeft na het invoeren van **getwd()** , voert u **setwd(***LOCATIE VAN UW WORKING DIRECTORY***)**[3] in. Deze locatie is eenvoudig te vinden als u in Verkenner (of Finder bij OS) op de rechtermuisknop klikt en daar kiest voor eigenschappen (of *Info* bij OS). De locatie van de map wordt in het venster weergegeven, selecteer en kopieer deze locatie.

[2] Deze map kunt u ook een andere naam geven dan *Working Directory*.

[3] Bij Windows kan het zijn dat u de richting van de slashes \ moet veranderen (naar /).

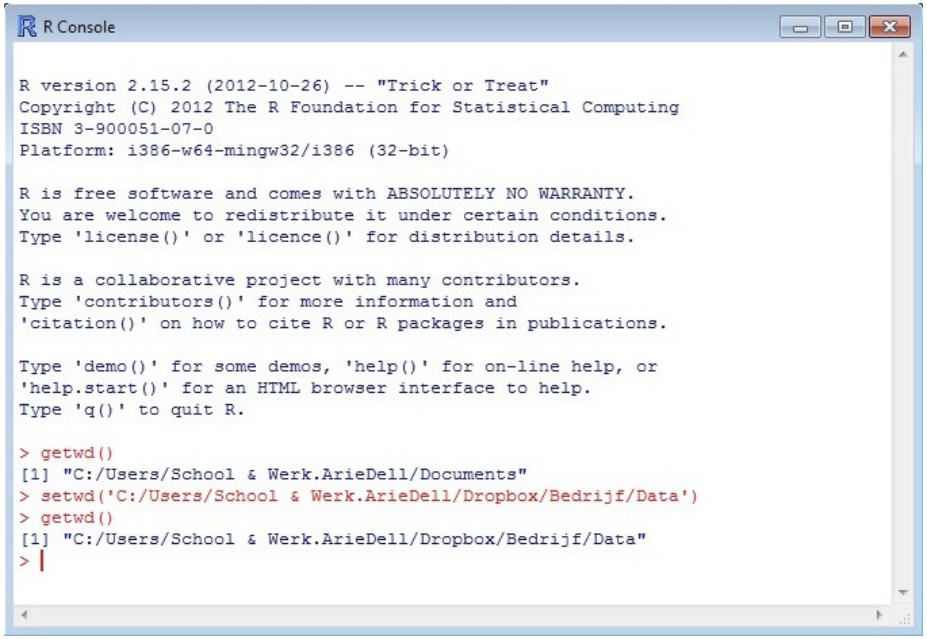
3. **Locatie van de nieuwe Working Directory in de R Console invullen:**

   Vul de locatie tussen de haakjes en aanhalingstekens in bij de code **setwd()** in R door het pijltje van de muis tussen de haakjes en aanhalingstekens te plaatsen->Rechtermuisklik->*Plakken* (of CTRL+V). In **afbeelding 2** is de locatie van de Working Directory als volgt met **setwd()** ingevuld. **Setwd('C:/Users/School & Werk.ArieDell/Dropbox/Bedrijf/Data')**.

   Let op! Het is belangrijk dat de locatie tussen aanhalingstekens geplaatst wordt!

4. **Controleren van de huidige Working Directory:**

   Als de bovenstaande stappen zijn uitgevoerd, moet de nieuwe Working Directory succesvol zijn ingesteld. Dit kunt u controleren door met de code **getwd()** te controleren van welke Working Directory er momenteel gebruik wordt gemaakt. Als na het indrukken van Enter de locatie van u zojuist ingestelde Working Directory wordt weergeven, is het instellen van de Working Directory succesvol verlopen en bent u klaar om te werken met R.



Afbeelding 2 De locatie van de Working Directory in de R console instellen.

1.2 Data klaarmaken voor gebruik.

Databestanden zijn er in verschillende vormen en maten. Bij de inleiding van deze manual is aangegeven hoe u een xls-bestand om kan zetten naar een csv-bestand. R kan in zijn standaardvorm namelijk geen xls-bestanden lezen, wel csv-bestanden.[4]

Controle.

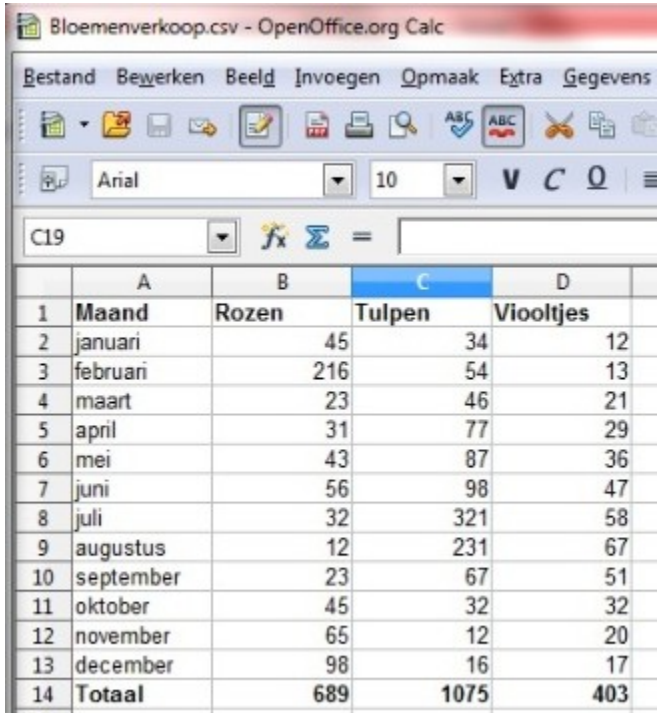
Het is echter ook belangrijk om in het xls-bestand of csv-bestand zelf te controleren of de data goed geschikt is om meet te analyseren. Om te kunnen bepalen of de data van een goede kwaliteit is, kunnen de volgende vier criteria gebruikt worden:

Nauwkeurigheid

Controle over de juistheid en betrouwbaarheid van de data.

Tijdigheid

Controle of de data up-to-date is of in ieder geval over de juiste tijd gaat.



	A	B	C	D
1	**Maand**	**Rozen**	**Tulpen**	**Viooltjes**
2	januari	45	34	12
3	februari	216	54	13
4	maart	23	46	21
5	april	31	77	29
6	mei	43	87	36
7	juni	56	98	47
8	juli	32	321	58
9	augustus	12	231	67
10	september	23	67	51
11	oktober	45	32	32
12	november	65	12	20
13	december	98	16	17
14	**Totaal**	**689**	**1075**	**403**

Afbeelding 3 Een eenvoudig csv-bestand in OpenOffice.org Calc.

Compleetheid

Controle of er geen missende data is en de controle of het databestand breed en diep genoeg is om analyses op uit te voeren.

Consistentie

Controle of bij de data dezelfde waardes en termen gebruikt worden over de verschillende databestanden en bronnen.

[4] Om R in staat te maken voor het lezen van meer typen bestanden, kunnen er verschillende *Packages* worden geïnstalleerd. In de Appendix van deze manual wordt kunt u zien hoe *Packages* kunnen worden geïnstalleerd. Voor de vaardigheden met R die in deze cursus worden geleerd hoeft u echter geen gebruik te maken van *Packages.*

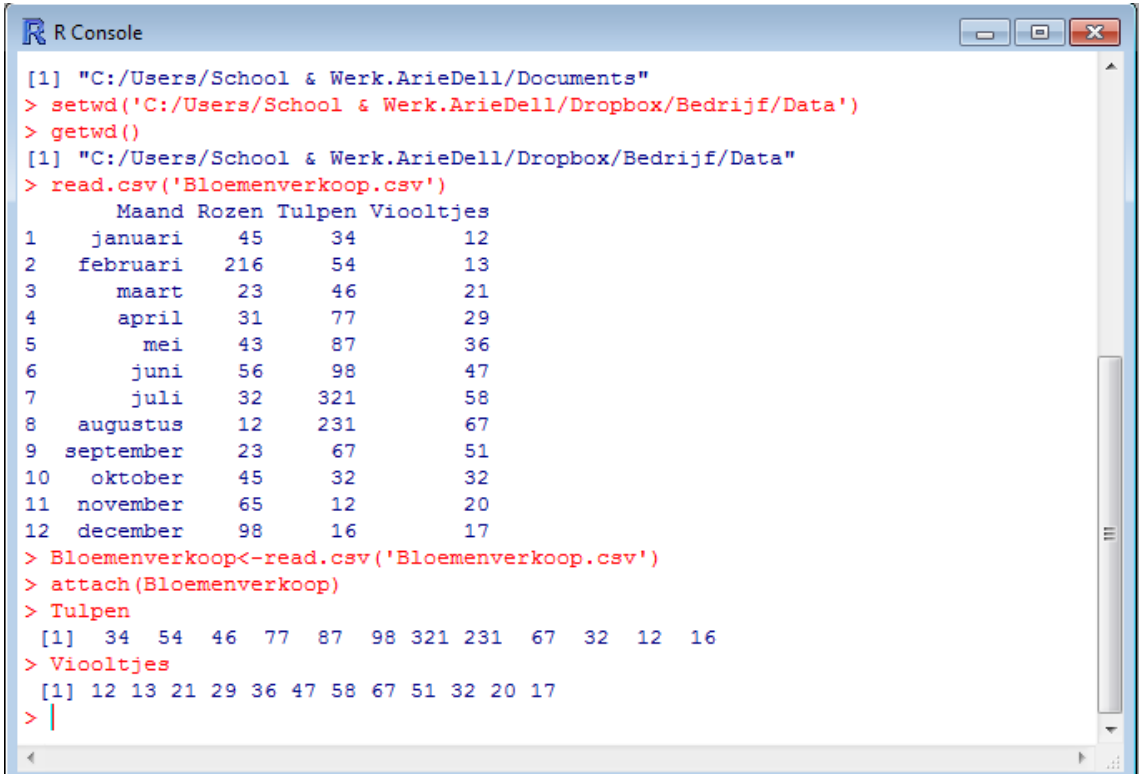
Transformatie.

Om een databestand zo goed mogelijk te kunnen analyseren met R, is het verstandig om het bestand zo simpel en eenvoudig mogelijk in te delen. Allerlei soorten tekst, kleurgebruik of afbeeldingen kunnen het best uit het bestand verwijderd worden als u het met R wilt analyseren. Zo voorkomt u mogelijke foutmeldingen of andere vervelende complicaties bij R. In **afbeelding 3** is een voorbeeld weergegeven van het eenvoudige bestand *Bloemenverkoop.csv.* Het bestand is al geconverteerd van xls-bestand naar csv-bestand.

Let op! In de afbeelding is ook te zien dat er in het bestand de totalen van de verkoop van de verschillende bloemen worden weergegeven. R leest bij een csv-bestand de bovenste regel als categorieën (hier dus Maand, Rozen, Tulpen en Viooltjes) en de overige regels als data over die categorieën. R herkent hier de regel Totaal niet. Bij wijze van spreken denkt R dat Totaal een dertiende maand is. Het is dus verstandig om de regel Totaal te verwijderen. Dit levert in het verdere verloop bij het analyseren geen problemen op, R kan naderhand namelijk alle totalen zelf weer berekenen als daar om gevraagd wordt.

1.3 Data importeren in de R Console

Controleer of de juiste *Working Directory* ingesteld is. Controleer ook of de databestanden in deze map zijn geplaatst[5]. Als deze twee handelingen in orde zijn bent u klaar om databestanden te importeren in de R console.



Afbeelding 4 Het importeren van een databestand, het aanmaken van een variabele van het databestand en een matrix maken van het databestand.

Csv-bestanden importeren

U kunt zelf selecteren welk bestand u wilt importeren in de R console. Dit doet u als volgt:

1. **Het databestand importeren:**

   Voer de code **read.csv('*NAAM VAN UW BESTAND*')** in. Bij het voorbeeld in **afbeelding 4** wordt het bestand *Bloemenverkoop.csv* geïmporteerd. Hiervoor wordt de code **read.csv('Bloemenverkoop.csv')** ingevoerd. Na het indrukken van Enter verschijnt de informatie uit het databestand in de R console.

   Let op! De code die in R wordt ingevoerd is hoofdlettergevoelig, 'bloemenverkoop.csv' wordt door R niet gevonden. Let hierbij ook op dat de naam van het bestand tussen *'aanhalingstekens'* is geplaatst.

[5] Zoals in hoofdstuk 1.1 over de *Working Directory* wordt uitgelegd: De *Working Directory* is de map waar R zoekt naar de databestanden om te lezen. De databestanden moeten dus in deze map worden geplaatst om door R gevonden te worden.

2. **Een variabele aanmaken voor het bestand:**

   De volgende stap is variabele aanmaken voor het databestand. In de Inleiding wordt uitgelegd dat met R variabelen aangemaakt kunnen worden met de **<-** code. Voor het databestand *Bloemenverkoop.csv* wordt nu een variabele gemaakt. Dit gaat aan de hand van de volgende code:

   Bloemenverkoop<-read.csv('Bloemenverkoop.csv')[6] . Het bestand *Bloemenverkoop.csv* wordt door R nu niet langer meer gezien als een extern bestand, maar als een variabele in deze sessie.

   Het nut van een variabele aanmaken voor het bestand is dat nu niet meer de steeds de code **read.csv('Bloemenverkoop.csv')** ingevoerd moet worden als u dit bestand wilt gebruiken. Nu er een variabele is aangemaakt voor het bestand, hoeft u alleen maar **Bloemenverkoop** in te voeren om informatie te weergeven over het bestand.

3. **Een matrix maken van het databestand:**

   Door een matrix te maken van de zojuist aangemaakte variabele, kan de R Console op een betere manier de data van de variabele lezen. Dit gaat eenvoudig met de volgende code: **attach(***Naam van de variabele van het databestand****)**. In **afbeelding 4** wordt de code **attach(Bloemenverkoop)** gebruikt.

   U kunt nu de categorieën van de variabelen op een eenvoudige manier weergeven met R. Als u bijvoorbeeld informatie wilt over de categorie *Tulpen* wilt weergeven, voert u simpelweg de code **Tulpen** in. U zult in het volgende hoofdstuk zien dat er vóór de code **Tulpen** verschillende functies geplaatst kunnen worden om de categorie tulpen te analyseren.

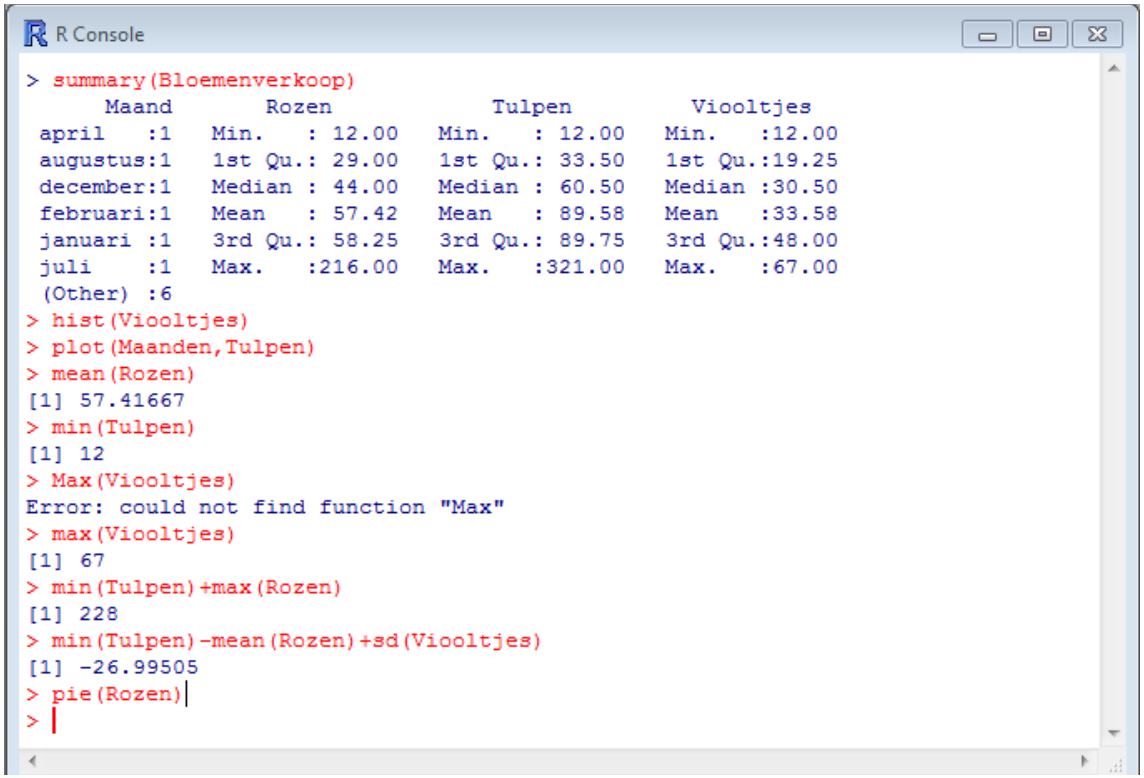
[6] Het is niet noodzakelijk om de variabele *Bloemenverkoop* te noemen. Met de <- code herkent R het databestand voor welke naam u het ook geeft. Bloemetjes <- read.csv('Bloemenverkoop.csv') kan bijvoorbeeld ook. De variabele heet dan in het vervolg *Bloemetjes.*

1.4 Basisfuncties voor analyseren met R.

In het vorige hoofdstuk wordt uitgelegd hoe er een variabele aangemaakt kan worden voor een databestand. Daarbij is ook uitgelegd hoe er van de aangemaakte variabele een matrix gemaakt kan worden met de code **attach()**.

Samenvatting van het databestand:

Met de code **summary(***NAAM VAN DE VARIABELE***)** wordt er een samenvatting gegeven van het databestand die aan de variabele gekoppeld is. In **afbeelding 5** wordt een voorbeeld gegeven van een samenvatting over het bestand *Bloemenverkoop.csv.*



Afbeelding 5 Samenvatting van het databestand en diverse statistische functies.

Omdat het bestand is omgezet in een variabele, kan een samenvatting eenvoudig weergegeven worden met de code: **summary(Bloemenverkoop)**. Na het drukken op Enter verschijnt de samenvatting van het bestand.
Onder de categorie *Maand* wordt simpelweg geteld hoe veel keer een betreffende maand in het bestand voor komt. Dit komt omdat de categorie *Maand* geen cijfers bevat, maar alleen namen van maanden. De categorie maand is hierom geen *numerieke categorie* maar een categorische categorie.

Merk op dat de variabele Bloemenverkoop ook weer bestaat uit een aantal variabelen, namelijk: Maand, Rozen, Tulpen en Viooltjes.

Over de categorieën *Rozen*, *Tulpen* en *Viooltjes* wordt de volgende informatie weergegeven:

Min. (Minimum) en Max. (Maximum):

Min. geeft aan wat het minimum is van een variabele. In het voorbeeld bij **afbeelding 5** wordt aangegeven dat er in de minste maand 12 rozen zijn verkocht. Minder dan 12 rozen in een maand zijn er in het hele jaar niet verkocht omdat 12 het minimum is.
Hetzelfde geldt voor *Max.*, dat aangeeft wat de verkoopcijfers zijn in de maand waar in het meeste aantal rozen zijn verkocht.

1st Qu. En 3rd Qu. en Median:

Hier worden het eerste en het derde kwartiel weergegeven van de categorie. Het eerste kwartiel wordt aangegeven met *1st Qu..* Bij het eerste kwartiel worden de laagste 25% van de getalswaarden bij elkaar opgeteld. Bij het voorbeeld in **afbeelding 5** geeft *1st Qu.* bij Rozen aan dat de 25% laagste verkoopaantallen bij elkaar 29 Rozen zijn. Bij het derde kwartiel *3rd Qu.* worden de 25% hoogste verkoopaantallen opgeteld met 58,25 als uitkomst.
Median weergeeft de mediaan van de variabele, dit is in theorie het tweede kwartiel. *Median* geeft aan dat als de verkoopcijfers op volgorde van laag naar hoog worden gezet, wat het middelste getal zal zijn[7].

Mean:

Mean. geeft het gemiddelde aan van de categorie. In het voorbeeld van **afbeelding 5** is het gemiddelde aantal verkochte rozen per maand 57,42.

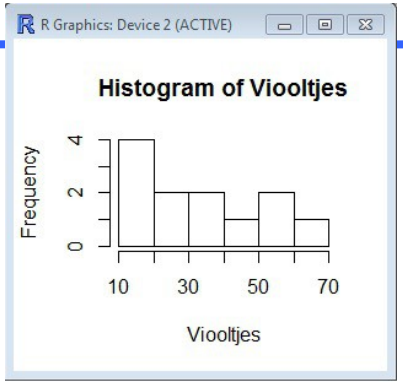
Analyseren van data met grafieken en diagrammen:

Met R kan een databestand eenvoudig met verschillende statistische functies geanalyseerd worden. Door al het werk dat hiervoor is gedaan, zoals het aanmaken van variabelen en het maken van een matrix, kunnen de functies worden toegepast met eenvoudige codes. Hier worden een paar voorbeelden gegeven van de functies die gebruikt kunnen worden om het bestand *Bloemenverkoop.csv* te analyseren.

[7] Als er geen middelste getal is, weergeeft de mediaan het gemiddelde van de middelste twee getallen.

Histogram hist():

Met de code **hist(***VARIABELE***)**, kunt u R een histogram laten weergeven van de variabele die u kiest. In **afbeelding 6** wordt een histogram weergeven van de categorie Viooltjes. Hiervoor is de volgende code gebruikt: **hist(Viooltjes)**

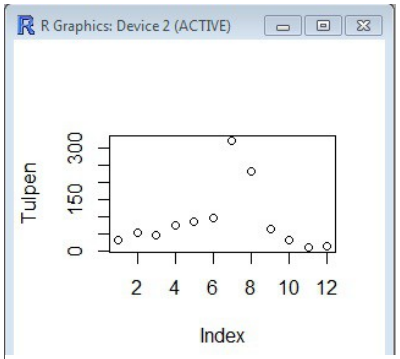


Afbeelding 6 Histogram van de variabele *Viooltjes*.

Grafiek plot():

Met de code **plot(***VARIABELE 1***,** *VARIABELE 2***)**, kunt u R een grafiek laten weergeven van de variabelen die u kiest. In **afbeelding 7** wordt een grafiek weergeven over de verkoop van het aantal tulpen per maand over het hele jaar. Hiervoor is de volgende code gebruikt: **plot(Maanden, Tulpen)**.

Let op! De tussen de haakjes gaat de variabele op de x-as op plek 1 en de categorie voor de y-as op plek 2. Let ook op de hoofdlettergevoeligheid van R.



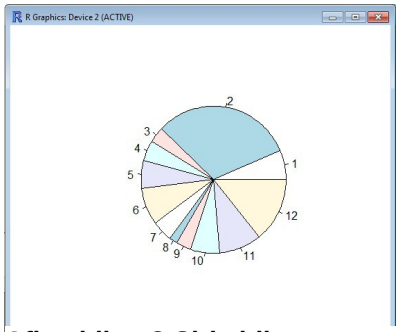
Afbeelding 7 Grafiek van de variabelen *Maand* en *Tulpen*.

Cirkeldiagram pie():

Met de code **pie(***VARIABELE***)**, kunt u R een cirkeldiagram laten weergeven van de categorie die u kiest. In **afbeelding 8** wordt een cirkeldiagram weergeven van de categorie Rozen. Het valt op dat in maand 2, februari, veruit de meeste rozen zijn verkocht. Waarschijnlijk vanwege Valentijnsdag.

Analyseren met overige statistische functies:

Omdat het databestand gekoppeld is aan een variabele en er van de variabele een matrix is gemaakt, kunnen er met R eenvoudig overige statistische functies gebruikt worden om de data te analyseren. Achter een functie hoeft u namelijk telkens alleen maar de variabele die u wilt analyseren tussen de haakjes achter de code te zetten waarbij u na het drukken van Enter het resultaat krijgt. In de Appendix vindt u de codes voor specifieke statistische functies. Hier worden de meest gebruikte statistische functies en codes besproken.



Afbeelding 8 Cirkeldiagram van de variabele *Rozen*.

Let op! Bij het invoeren van statistische functies is het belangrijk om te letten op de hoofdlettergevoeligheid, in afbeelding is te zien dat bij het invoeren van Max (met hoofdletter) een foutmelding geeft als resultaat.

Codes voor statistische functies worden altijd ingevoerd met een kleine letter.

Gemiddelde/ mean():
Het gemiddelde van een variabele kunt u berekenen met de code:

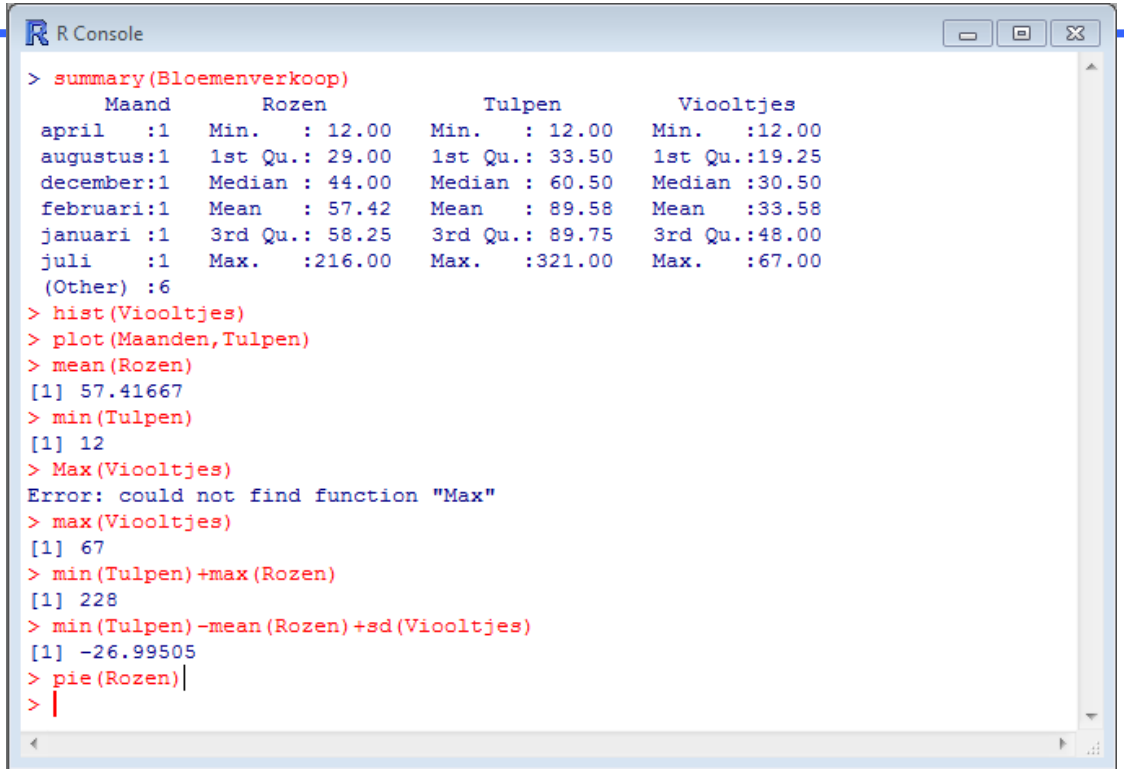
mean(*VARIABELE*)

In **afbeelding 9** geeft de code **mean(Rozen)** het resultaat 57.41.667. Het gemiddeld aantal rozen verkocht per maand is 57.

Minimum/ min():
Het minimum van een categorie kunt u berekenen met de code

min(*VARIABELE*)

In **afbeelding 9** geeft de code **min(Tulpen)** het resultaat 12. In de maand waarin de verkoop van de tulpen minimaal was, zijn er 12 tulpen verkocht.



Afbeelding 9 Statistische functies uitvoeren met R.

Maximum/ max():
Het maximum van een categorie kunt u berekenen met de code

max(*VARIABELE*)

In **afbeelding 9** geeft de code **max(Viooltjes)** het resultaat 67. In de maand waarin de verkoop van de viooltjes maximaal was, zijn er 67 viooltjes verkocht.

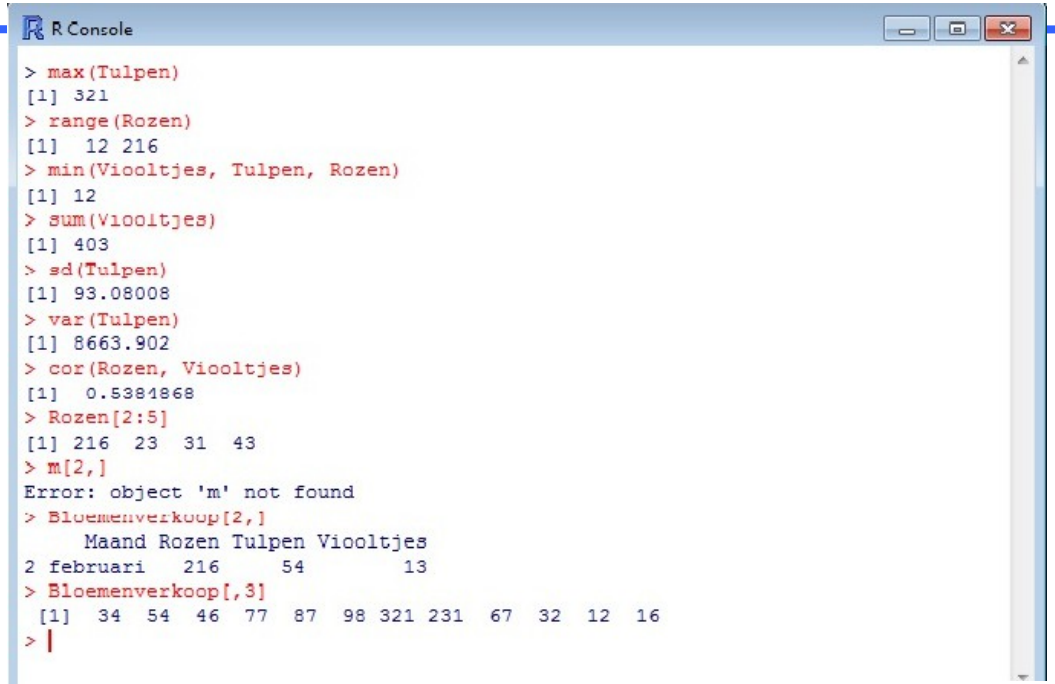
Merk op uit de afbeelding dat u ook berekeningen kunt doen met verschillende gegevens. In het voorbeeld dat wordt gegeven ziet u dat het minimum van de tulpen wordt verminderd met het gemiddeld aantal verkochte rozen waarbij de standaarddeviatie van het aantal verkochte viooltjes wordt opgeteld. Om het maken van berekeningen in de vingers krijgen kunt u experimenteren met het maken van berekeningen.

Totaal / sum():

Het totaal van een categorie kunt u berekenen met de code **sum(*VARIABELE*)**[8]. In **afbeelding 10** wordt het totaal aantal verkochte viooltjes weergegeven door middel van de code **sum(Viooltjes).** Het resultaat is 403, dat concludeert dat er in het jaar totaal (alle maanden in het databestand opgeteld) 403 Viooltjes zijn verkocht.

Bereik / range() :

De code **range(*VARIABELE*)** weergeeft het bereik, het minimum en maximum, van de betreffende categorie. In **afbeelding 10** wordt naar het bereik van de categorie Rozen gezocht met de code **range(Rozen)**. Dit geeft het resultaat 12 216. Het minimum en maximum aantal rozen dat verkocht is in het betreffende jaar zijn respectievelijk 12 en 216.



Afbeelding 10 Overige statistische functies met R.

Standaardafwijking(standaarddeviatie) / sd() en variantie / var() :

De standaardafwijking en variantie worden gebruikt om de spreiding en mate waarin verschillende waarden van een categorie verschillen aan te geven. De standaardafwijking van een categorie kunt u vinden met de code **sd(*VARIABELE*)** en de variantie met **var(*VARIABELE*)** . In **afbeelding 10** wordt de standaardafwijking van de categorie Tulpen gevonden met de code **sd(Tulpen)**. Dit geeft het resultaat 93,08008[9]. De variantie van de categorie Tulpen wordt gevonden met de code **var(Tulpen)** en geeft 8663,902 als resultaat. (Valt het u op dat de standaarddeviatie de wortel is van de variantie?)

[8] Hier ziet u uiteindelijk waarom het verstandig is om in het excel- of csv-bestand de totalen van de categorieën weg te halen. Als u de totalen heeft laten staan telt de sum-functie ook het totaal bij de maanden op. Hierdoor wordt met de sum-functie het totaal twee keer zo veel weergegeven.

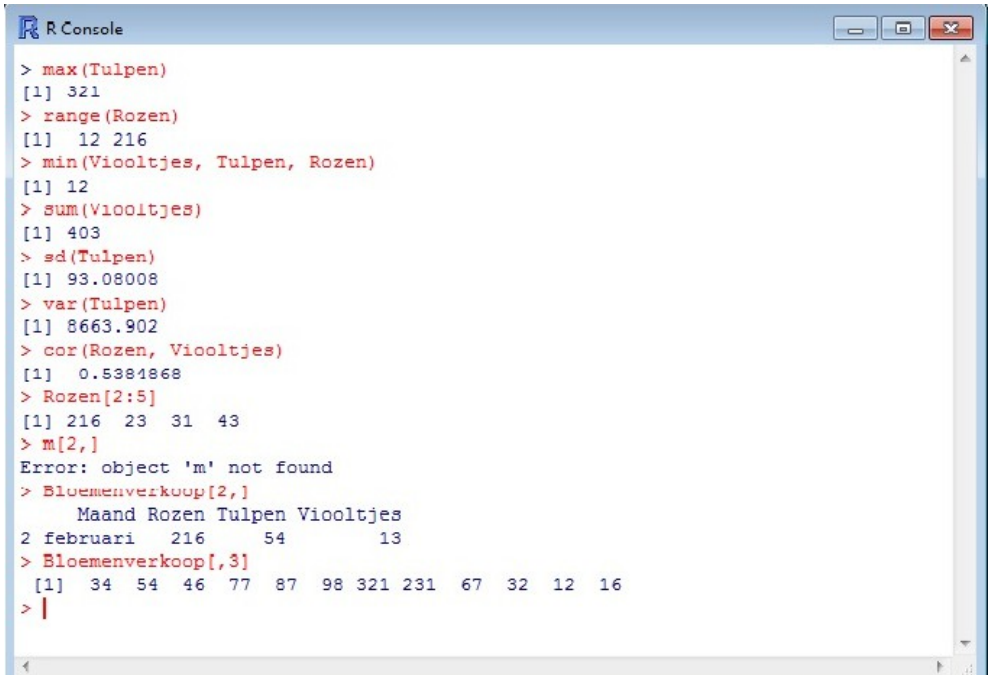
[9] Bij geavanceerde en handmatige statistische berekeningen zijn de standaardafwijking en variantie redelijk belangrijke waarden, bijvoorbeeld voor het berekenen van een betrouwbaarheidsinterval. Voor deze cursus zijn de standaardafwijking en variantie echter niet belangrijk, maar is het wel handig om te weten dat u deze eenvoudig kunt vinden met de daarbij horende codes. Handmatig berekenen van de variantie en standaardafwijking neemt namelijk veel tijd in beslag.

http://www.arietwigt.wordpress.com/

Correlatie / cor() :

Met de correlatie-functie kunt u de samenhang vinden tussen twee verschillende variabelen of in dit geval categorieën. Een correlatie van 1 staat hierbij voor een perfecte positieve samenhang, een correlatie van -1 staat voor een perfecte negatieve samenhang. Een correlatie van 0 betekent geen samenhang.

Om de correlatie tussen twee verschillende variabelen of categorieën te vinden gebruikt u de volgende code: **cor(***NAAM1*,*NAAM2*). In **afbeelding 10** wordt de correlatie tussen de verkopen van rozen en viooltjes opgezocht met de code **cor(Rozen,Viooltjes),** dit heeft 0.5384868 als uitkomst. Er is in dit databestand dus een enige vorm van samenhang tussen verkoop van rozen en viooltjes.



Afbeelding 11 Overige statistische functies met R.

Getallen of gegevens weergeven/opzoeken uit de matrix

Als u informatie over gegevens wilt weergeven over een bepaalde fractie van een categorie, heeft R hier ook codes voor. Hiervoor typt u als eerst de betreffende variabele in. Daarachter typt u tussen de vierhoekige haakjes de rij, de kolom of het interval dat u wilt weergeven. Aan de hand van een paar voorbeelden wordt laten zien hoe u dit kunt doen.

Interval / *NAAM VARIABELE VAN HET DATABESTAND***[interval] :**

In **afbeelding 10** wordt er met de code **Rozen[2:5]** verkoopcijfers van de rozen weergeven voor de maanden februari tot en met mei. De verkopen voor deze maanden zijn respectievelijk 216, 23, 31 en 43 geweest.

Rijen / * NAAM VARIABELE VAN HET DATABESTAND ***[***NAAM VAN DE RIJ***,] en kolommen /** *NAAM VARIABELE VAN HET DATABESTAND***[,***NAAM VAN DE KOLOM***]:**

Als u informatie uit een bepaalde rij wilt weergeven gebruikt u de code

* NAAM VARIABELE VAN HET DATABESTAND ***[***RIJNUMMER***,].** In **afbeelding 10** worden de gegevens over de maand februari (rij 2) weergegeven met de code **Bloemenverkoop[2,]**. Er wordt weergegeven dat er in de maand februari 216 rozen, 54 tulpen en 13 viooltjes zijn verkocht. Dit kunt u ook doen met de code **Bloemenverkoop[februari,]**

U kunt ook informatie vinden over kolommen, in dit geval de categorieën van de bloemen. In **afbeelding 10** wordt met de code **Bloemenverkoop[,3]** informatie gegeven over de derde kolom, in dit geval Tulpen. Dit kunt u ook doen met de code

Afbeelding 12 Specifiek informatie zoeken uit het databestand.

Bloemenverkoop[,"Tulpen"].

Op deze manier kunt u eenvoudig naar specifieke informatie zoeken uit het databestand. Als u bijvoorbeeld wilt weten hoeveel rozen er in november zijn verkocht gebruikt u de code **Bloemenverkoop[november,'Rozen'[10]]**.

Let op! Gebruik altijd de naam van de variabele waaraan het databestand gekoppeld is in deze code. De namen Matrix of m werken niet en geven een foutmelding.

[10] De naam van de variabele die u wilt opzoeken moet u tussen aanhalingstekens plaatsen.

1.5 Sessie opslaan en laden.

De sessie opslaan.

Als u op een later tijdstip verder wilt gaan met de huidige sessie in de R console, kunt u de sessie opslaan.

1. Dit gaat met de code **save.image('***LOCATIE WAAR U DE SESSIE WIL OPSLAAN****/***NAAM VAN DE SESSIE****.Rdata')**.
2. Om de locatie te vinden waar u het bestand wilt te kunnen vinden, kunt u de code **getwd()** invoeren. Hiervan kunt u het pad kopiëren die naar de huidige Working Directory leidt.
   U kunt ook in de map waar u de opgeslagen sessie wilt plaatsen via de rechtermuisknop naar het venster Eigenschappen gaan. Daar kunt u de locatie kopiëren van de map om deze uiteindelijk tussen de haakjes en aanhalingstekens van de code **save.image()** te plaatsen.

Het is verstandig om de sessie op te slaan in de map die u als Working Directory gebruikt.

Afbeelding 13 De locatie en naam van de sessie bepalen om op te slaan.

3. Achter de laatste slash (/) van de locatie typt u de naam in die u de sessie wilt geven. Achter de naam plaatst u de toevoeging *.Rdata*.
   Het voorbeeld in **afbeelding 13** laat zien dat de sessie wordt opgeslagen onder de naam *Sessie1.Rdata*.
   In het voorbeeld is ook te zien dat het bestand is opgeslagen in de map Data, de map die voor deze sessie is gebruikt als Working Directory.
4. Als u de code en de locatie met bestandsnaam tussen de haakjes en aanhalingstekens hebt geplaats, zoals in afbeelding, kunt u op Enter drukken om de sessie op te slaan.

5. Als u hierna de R console afsluit door op kruisje te drukken (of via een andere manier afsluit), komt het dialoogvenster met de tekst *Save workspace image?* in beeld. Klik hiervoor altijd op *Ja* om er zeker van te zijn dat de sessie goed is opgeslagen.

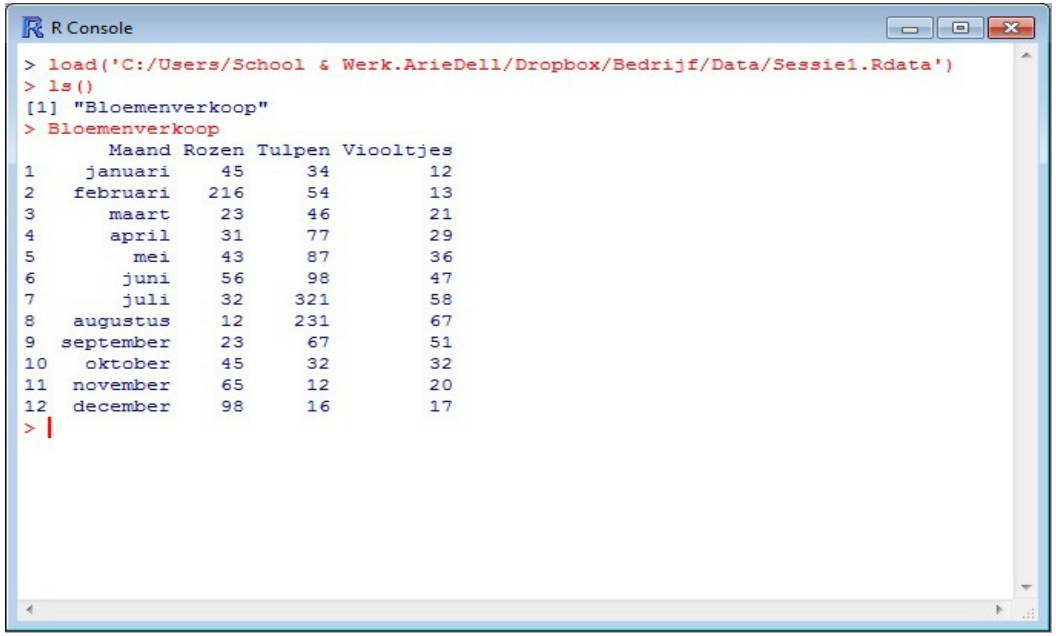
Een bestaande sessie laden

Er zijn twee manieren om een bestaande of door u eerder opgeslagen sessie te laden.

U zoekt in Windows Verkenner(of in Finder bij OS) naar het .Rdata bestand en opent het met een dubbelklik.

U opent de R Console en voert de code **load('***LOCATIE WAAR DE SESSIE IS OPGESLAGEN***/***NAAM VAN HET BESTAND***')** en drukt daarna op Enter.

Met de code **ls()** kunt u een overzicht laten verschijnen van de variabelen die zich in de sessie plaatsvinden.
In **afbeelding 14** is te zien dat *Bloemenverkoop* de enige variabele is die aangemaakt is voor een databestand.



Afbeelding 14 Een eerdere sessie laden en opzoeken aan welke variabelen een databestand is gekoppeld.

1.6 Overzicht Deel 1.

Dit is het einde van het eerste deel van deze workshop. Als u alle stappen heeft gevolgd, bent u in staat om:

Een xls-bestand om te zetten in een csv-bestand met scheidingstekens die door R gelezen kunnen worden.

Een Working Directory van een sessie met R instellen, opslaan en laden.

Controleren aan de hand van vier criteria of een data bestand geschikt is om te gebruiken.

Een csv-bestand importeren naar de R Console.

R een variabele laten definiëren.

Een matrix maken van een databestand of variabele.

R grafieken en diagrammen laten maken aan de hand van de geïmporteerde data.

Statistische functies/formules op een eenvoudige manier uitvoeren op het databestand met R.

Berekeningen maken met R.

Gegevens uit het databestand op een snelle manier terug kunnen zoeken met R.

Deel 2: Uitgebreid data analyseren en voorspellingsmethodes.

U heeft in het eerste deel van deze handleiding kunnen wennen aan R. De basishandelingen van R zijn in Deel 1 uitgelegd. Om de code van R een beetje in de vingers te krijgen, is het aangeraden om veel te oefenen op verschillende datasets.

Bij Deel 2 ligt de nadruk op verbanden tussen variabelen in de dataset en daarmee voorspellingsmodellen mee te creëren. Dit wordt gedaan aan de hand van een regressieanalyse. Het is bij deel twee vooral belangrijk om de resultaten die R geeft, te interpreteren.

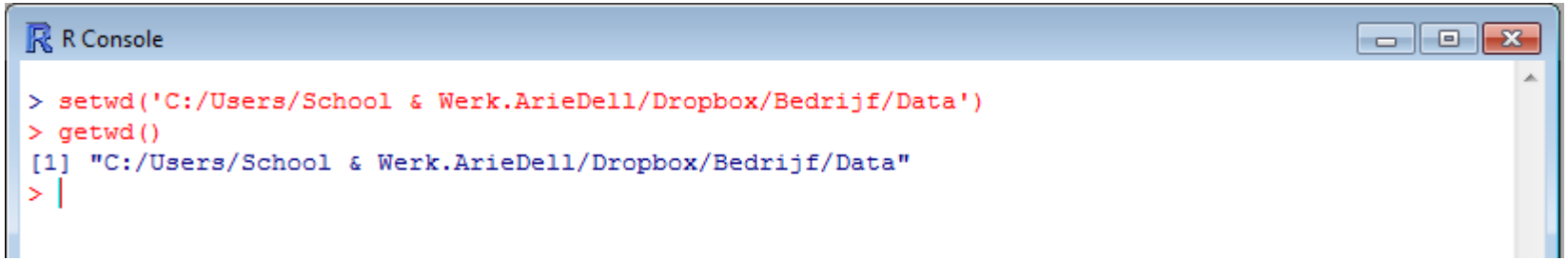
Bij Deel 2 worden de bestanden Projecten.csv en Projecten2.csv als voorbeelden gebruikt. Dit zijn grotere bestanden met meer en verschillende type variabelen. In de praktijk komt het natuurlijk voor dat er nog grote databestanden worden gebruikt dan Projecten.csv en Projecten2.csv. Echter is het principe qua data analyseren hetzelfde.

Veel succes met deel 2 van deze handleiding!

2.1 Working Directory instellen.

In Deel 1 heeft u kunnen zien hoe een Working Directory wordt ingesteld en gecontroleerd. Ook in dit deel controleert u of de gewenste map als Working Directory is ingesteld.

1. Controleer met de code **getwd()** wat de huidige Working Directory is.
2. Als niet de juiste Working Directory is ingesteld, voer dan de locatie van de map die u als Working Directory wilt instellen in. Op pagina 5 en 6 van deze manual kunt u zien hoe u eenvoudig de locatie van de map kunt vinden en kunt plakken in de R console. Gebruik de **code setwd(***LOCATIE VAN DE DOOR U GEWENSTE WORKING DIRECTORY***)**. Druk daarna Enter om de Working Directory in te stellen.
3. Controleer nogmaals met de code **getwd()** wat de huidige Working Directory is. Als dit klopt kunt u verder gaan.



Afbeelding 15 De Working Directory instellen en checken.

2.2 Bestand klaar maken voor gebruik.

In dit deel wordt een uitgebreider bestand gebruikt als voorbeeld, het bestand *Projecten.xls/Projecten.csv*[11]. Ook hier is het belangrijk om na te gaan of het bestand geschikt is om te importeren voor analyse met R. Deze controle kunt u doen aan de hand van de 4 criteria die in het eerste deel worden genoemd:

Nauwkeurigheid.
Compleetheid.
Tijdigheid.
Consistentie.

Eerst moet het databestand worden geopend.

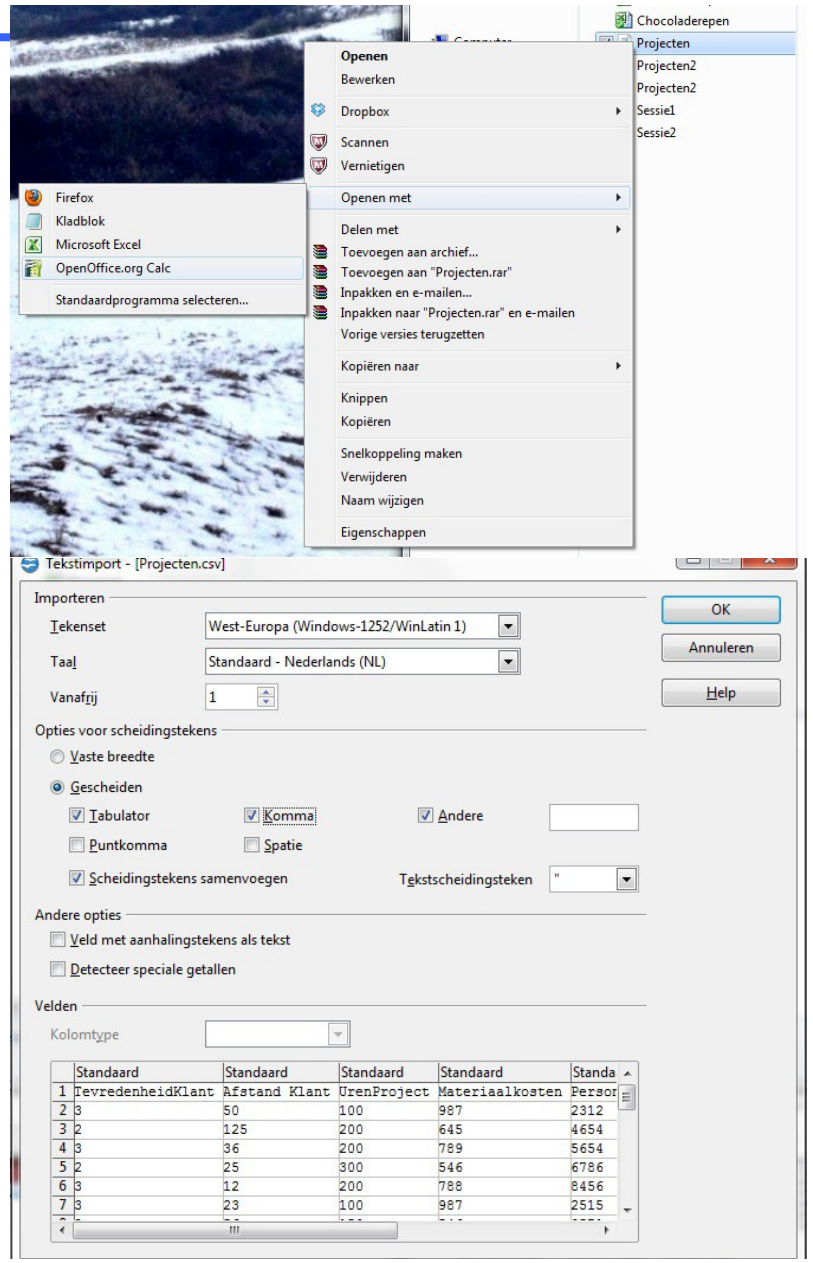
Een bestaand csv-bestand openen met OpenOffice.org Calc

Met OpenOffice.org Calc kunt u ook csv-bestanden openen. Omdat dit een bestand is dat alleen maar bestaat uit tekst, cijfers en tekens moet er van te voren ingesteld worden hoe het in de spreadsheet terecht komt.

U kunt een csv-bestand als volgt openen met OpenOffice.org Calc:

1. Selecteer het csv-bestand dat u wilt openen met de rechtermuisknop. Klik op *Openen met.* en selecteer daar OpenOffice.org Calc.

2. Het venster *Tekstimport* verschijnt, zie **afbeelding 16**.
   In dit venster is het gedeelte *Opties voor scheidingstekens* belangrijk. Hierbij kunt u namelijk aangeven wat er in dit csv-bestand als scheidingsteken moet worden gezien.
   Als het csv-bestand is opgeslagen met een komma als scheidingsteken, vinkt u



Afbeelding 16 Een csv-bestand met OpenOffice.org Calc openen en de Tekstimport instellen.

[11] Zie op pagina 4 hoe u een xls-bestand opslaat als csv-bestand met een komma als scheidingsteken.

hier **alleen** *Komma* **en** *Tabulator* (tabs) aan als scheidingstekens[12]. In het voorbeeld onderaan het venster ziet u een voorbeeld van hoe uw spreadsheet er uit gaat zien. In het voorbeeld in **afbeelding 16** zijn de instellingen juist geselecteerd. Als dit ook bij u het geval is, klikt u op OK in het venster.

3. Het csv-bestand wordt door OpenOffice.org Calc in de vorm van een spreadsheet weergegeven.

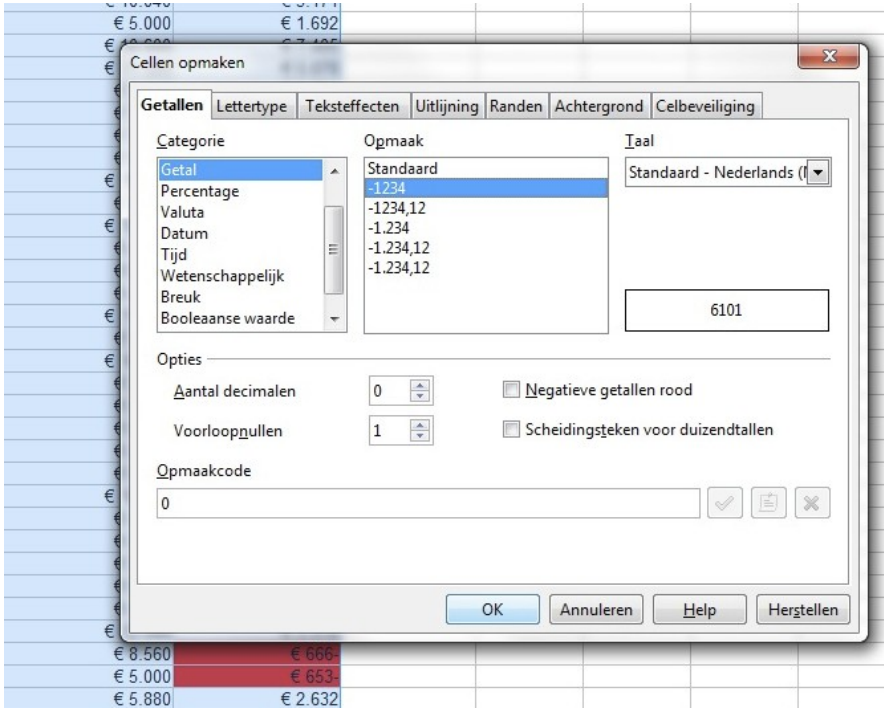
Uw databestand controleren in OpenOffice.org Calc

Als u met OpenOffice.org Calc een databestand of spreadsheet heeft geopend, kunt u cellen op maken door ze te selecteren en op de rechtermuisknop te klikken.

Hiermee opent u het venster *Cellen opmaken* (zie **afbeelding 17**). Hier kunt u de op verschillende manieren de opmaak van de cellen en vormen van de getallen wijzigen.

Het kan wel eens voorkomen dat u te maken heeft met decimalen. In het Nederlands worden decimale getallen achter een komma worden weergegeven. R gebruikt hiervoor echter een punt in plaats van een komma (Engels-VS instelling). Met het venster *Cellen opmaken* kunt u onder het keuzemenu *Taal* voor *Engels(VS)* kiezen. Dit is de manier waarop R getallen leest en dus beter beschikbaar gemaakt om te analyseren.

Verder kunt u in het venster *Cellen opmaken* de categorieën van de cellen , zoals percentages en valuta, wijzigen. Het is aan te raden om voor het analyseren met R altijd te kiezen voor de categorie *Getal*.



Afbeelding 17 Cellen opmaken in OpenOffice.org Calc

[12] Het kan voorkomen dat u ook *Spatie* moet instellen als scheidingsteken. Aan de hand van het voorbeeld dat in het venster *Tekstimport* wordt gegeven, kunt u zien wat in uw geval de juiste instelling is.

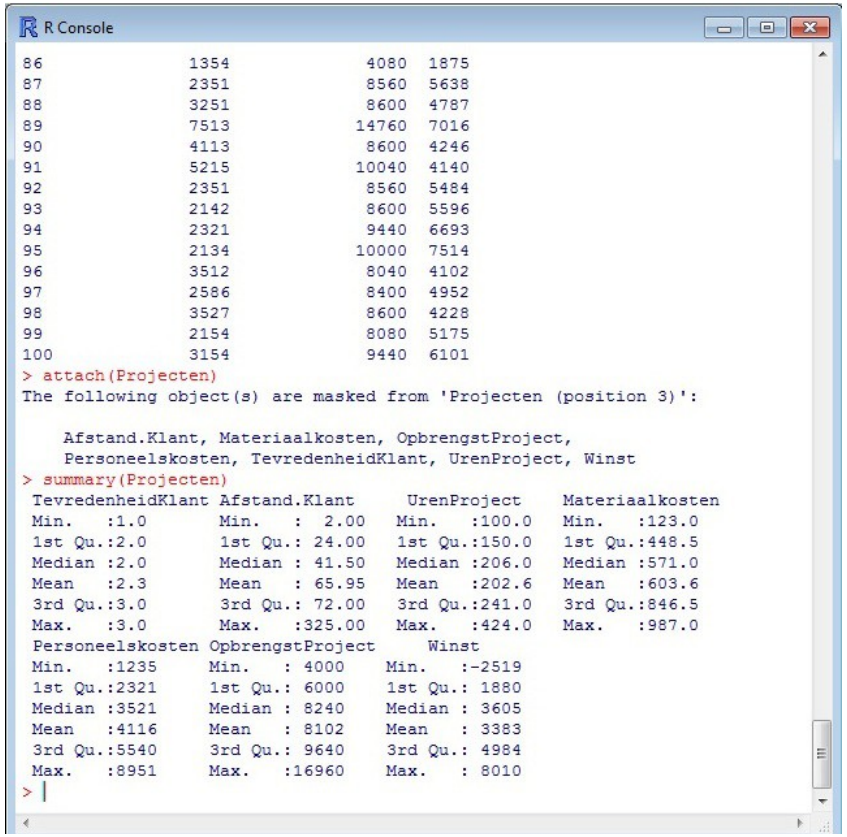
2.3 Importeren en een matrix maken van het databestand.

Vorige stappen nagaan.
In de vorige hoofdstukken heeft u kunnen zien hoe u de Working Directory kunt instellen, hoe u het databestand kunt converteren naar een csv-bestand en hoe u de data kunt controleren. Als u deze stappen heeft gedaan bent u klaar om het databestand te importeren naar de R Console.

Databestand importeren.
Bij het voorbeeld in **afbeelding 18** wordt het databestand *Projecten.csv* geïmporteerd. Dit is gedaan met de code **read.csv('Projecten.csv').**

Een variabele aanmaken.
Een snellere stap is bij het importeren het databestand Projecten.csv, er gelijk een variabele voor aanmaken. Dit kunt u doen aan de hand van de volgende code **Projecten<-read.csv('Projecten.csv')**. Het bestand heeft de variabele gekregen met de naam *Projecten*.

Een matrix maken.
Om een matrix van de code te maken gebruikt u de code **attach**(*NAAM VAN DE VARIABELE*).



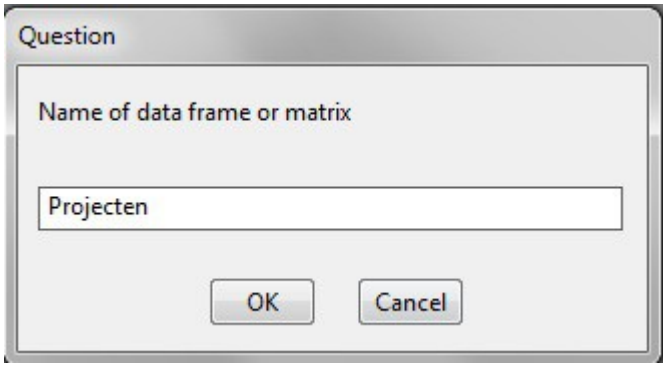
Afbeelding 18 Van het nieuw geïmporteerde databestand een matrix maken en een samenvatting geven.

Bij het voorbeeld in **afbeelding** wordt er een matrix gemaakt van de variabele Projecten met de code **attach(Projecten)**. Het is voor het overzicht misschien prettig om R een samenvatting te laten maken van uw databestand. Dit kunt u doen met de code **summary(***NAAM VAN DE VARIABELE***)**.
In het voorbeeld wordt er een samenvatting gemaakt van het databestand *Projecten.csv* met de variabele *Projecten* aan de hand van de code **summary(Projecten)**.

2.4 Data bewerken in R.

In de R console kunt u met een eenvoudige Data editor, de data uit een geïmporteerd databestand wijzigen.

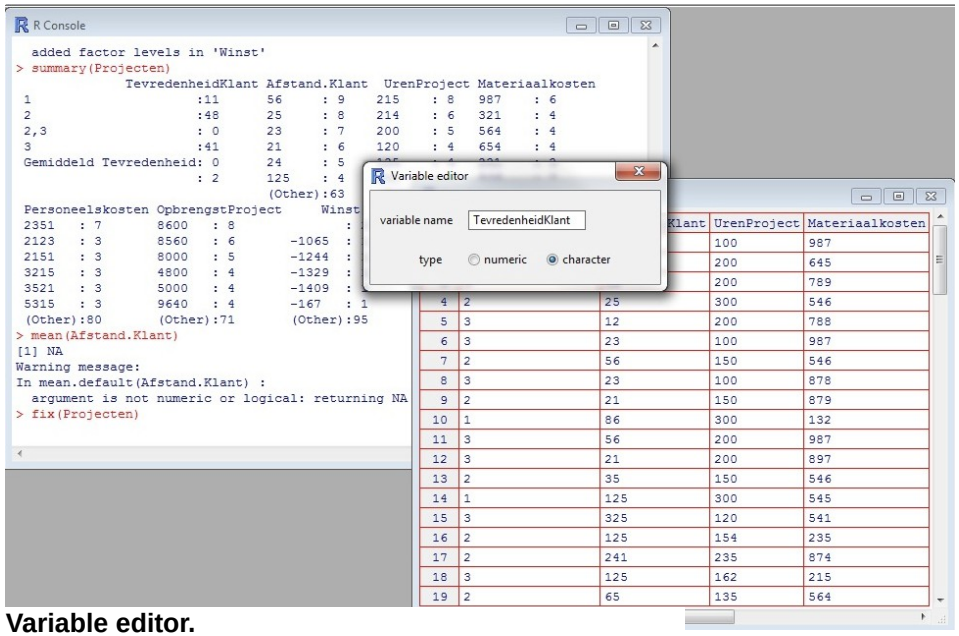
De data editor.
De Data editor opent u door in het menu venster te klikken op *Edit* -> klik op *Data editor.*
Het dialoogvenster *Question, name of data frame over matrix* verschijnt. Hier typt u de naam van de matrix[13] in, klik daarna op *OK*. Bij **afbeelding 19** wordt als voorbeeld gegeven dat de variabele *Projecten* wordt ingevoerd.



Afbeelding 19 Naam van de variabele waaraan het databestand is gekoppeld invoeren.

Nadat u voor een variabele heeft gekozen om te wijzigen, verschijnt de data in de vorm van een spreadsheet. Hier kunt u op verschillend cellen klikken en de data wijzigen als u dat wilt.
Als u in de spreadsheet boven aan op een van de categorieën klikt, verschijnt het dialoogvenster *Variable editor*. Hier kunt u kiezen of de type data van de betreffende variabele *numeric (*data in de vorm van cijfers) of *character*(data in de vorm van categorieën).

In **afbeelding 20** ziet u dat variabele *TevredenheidKlant* op het type *character* staat. Dit kunt u hier wijzigen in het type *numeric*.



Variable editor.

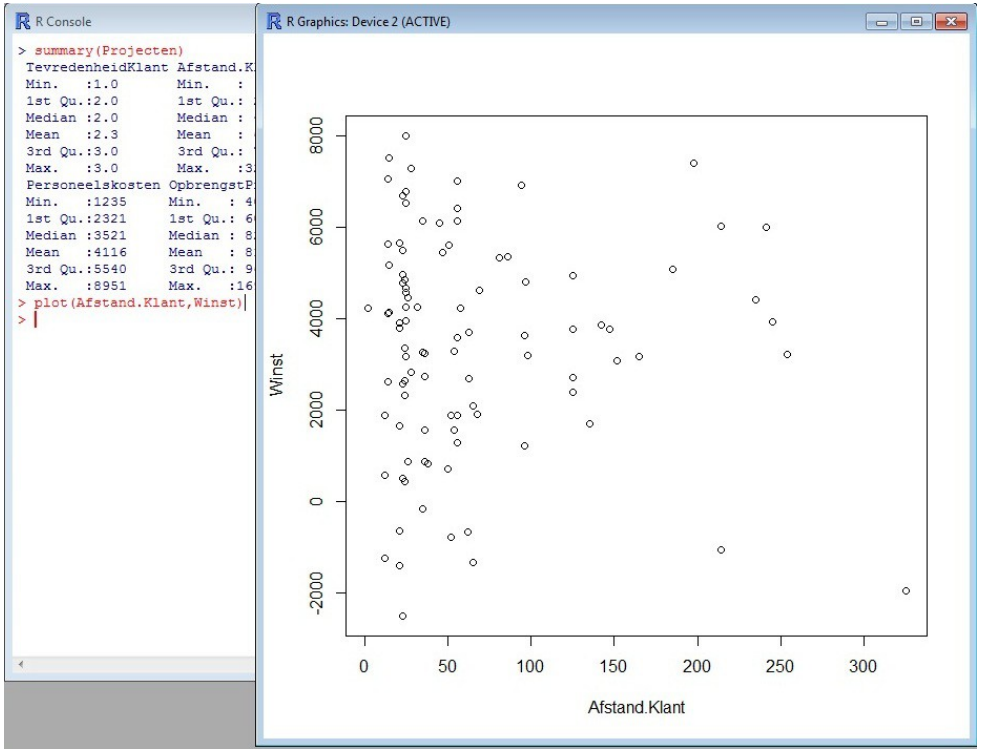
[13] Door de code **attach()** is er van de variabele *projecten,* een matrix gemaakt.

2.5 Grafieken en andere visuele weergaven van data.

In het eerste deel van deze manual hebt u kunnen zien dat R visuele weergaven kan toepassen op data. Bij uitgebreide databestanden als *Projecten.csv* kan dit een goed hulpmiddel zijn.

Variabelen uit het databestand gebruiken.
U hebt kunnen zien dat er voor het databestand *Projecten.csv* een variabele is aangemaakt. Ook hebt u kunnen zien dat er van deze variabele een matrix is gemaakt.
Omdat er een matrix is gemaakt, kunt u simpelweg de verschillende variabele in het databestand invoeren om er visuele effecten meet toe te passen.

Visuele effecten toepassen.
Bij het voorbeeld in **afbeelding 20** wordt er een grafiek gemaakt van de variabelen *Afstand Klant* en *Winst*. Dit wordt gedaan met de code:
plot(Afstand Klant, Winst).U kunt uit de grafiek aflezen dat de meeste klanten zich binnen een straal van 100 km bevinden en dat een winst tussen de 2000 en 6000 het meest voor komt[14]



Afbeelding 21 Data visualiseren.

In de appendix van deze manual vindt u de codes voor de visuele effecten die u nog meer met R kunt toepassen.

[14] Dit is nog een vrij zwakke beschrijving van de data, echter kunt u door de grafiek wel in een oog opslag simpele informatie over de data vinden. Geavanceerde beschrijving van de data kunt u doen aan de hand van correlaties en een Regressie. Deze onderwerpen worden vanaf hoofdstuk 2.6 in deze manual besproken.

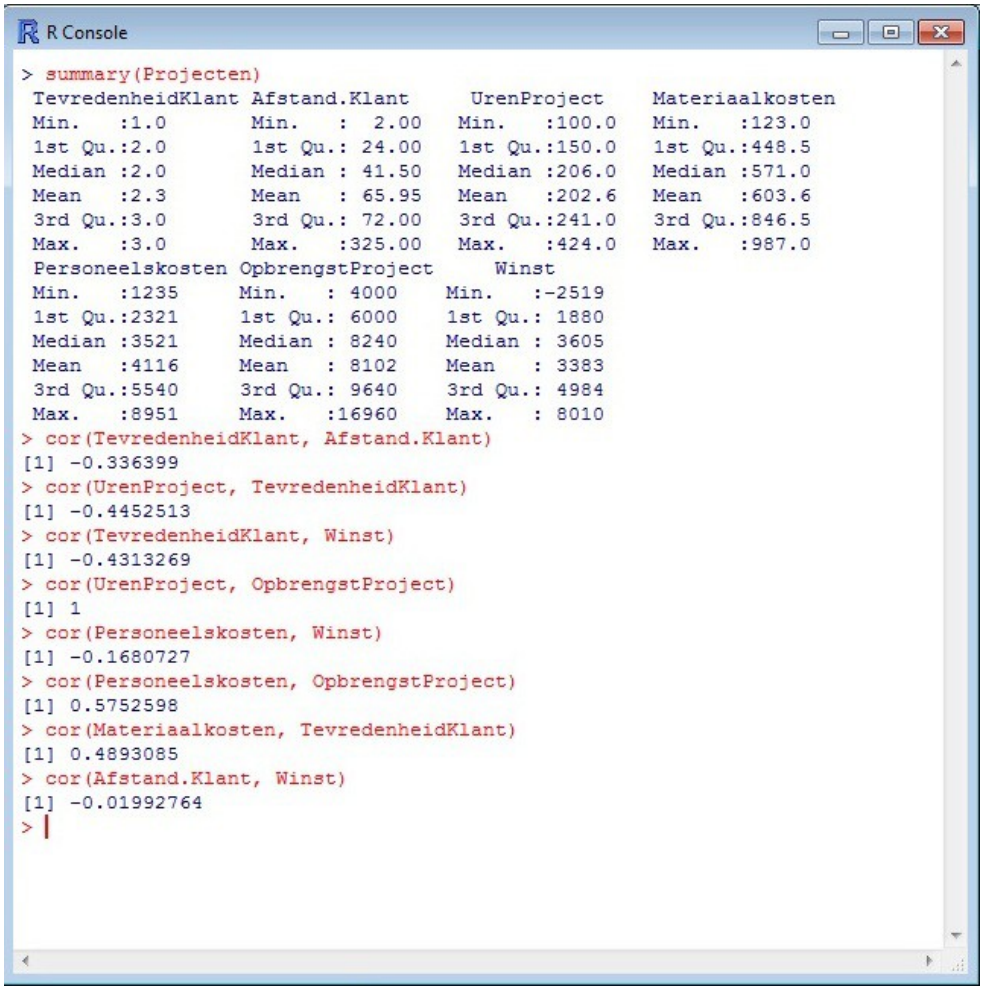
2.6 Correlaties.

Met correlaties kunt u de samenhang tussen twee variabelen vinden. Een correlatie van 1 staat hierbij voor een perfecte positieve samenhang, een correlatie van -1 staat voor een perfecte negatieve samenhang. Een correlatie van 0 betekent geen samenhang.
De waarde van de correlatie zegt echter niets, alleen de mate van correlatie.

Een correlatie berekenen.
U kunt R een correlatie tussen twee variabelen laten berekenen met de code **cor(***VARIABELE 1***,***VARIABELE 2***)** .

Bij de voorbeelden in **afbeelding 22** ziet u dat er correlaties worden berekend van verschillende variabelen uit het bestand *Projecten.csv*.

Interpretatie van een correlatie.
In **afbeelding 22** ziet u dat de variabelen *UrenProject* en *TevredenheidKlant* een vrij negatieve correlatie hebben. Met de gegeven waarde *-0.4452513* is op te merken dat het aantal uren dat het project in beslag neemt een vrij aanzienlijk negatief effect heeft op de tevredenheid van de klant. De tevredenheid van de klant daalt als het aantal uren van het project toeneemt.



Afbeelding 22 Correlaties tussen verschillende variabelen opzoeken met R.

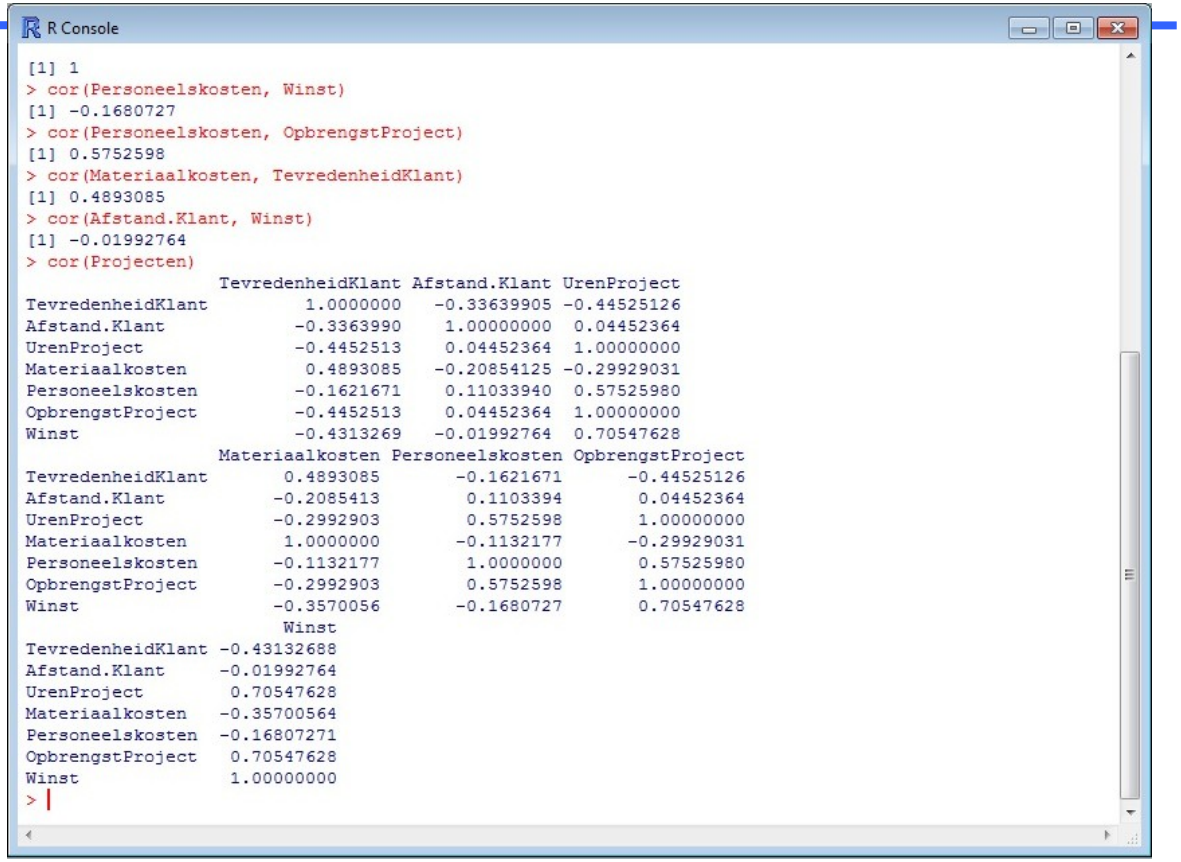
Op deze manier kunt u door correlaties op verschillende variabelen toe te passen, zien welke verbanden tussen variabelen sterk en zwak zijn.

Opsomming van alle correlaties.

U kunt met R een overzicht maken van de correlaties die alle variabelen onderling hebben. Dit doet u met de code **cor(***NAAM VAN DE VARIABELE VAN HET DATABESTAND***)** . Door de variabele die aan het gehele databestand gekoppeld is in de code toe te passen, worden alle variabelen in het databestand meegenomen.

In het voorbeeld bij **afbeelding 23** wordt met de code **cor(Projecten)** een overzicht gegeven van alle correlaties tussen de variabelen onderling. Dit bespaart u veel tijd als u graag alle correlaties wilt weten.

In **afbeelding 23** kunt u zien dat de variabelen *UrenProject* en *Winst* een sterke correlatie hebben.



Afbeelding 23 R met één code alle mogelijke correlaties op laten sommen.

2.7 Enkelvoudige lineaire regressieanalyse.

Met een regressieanalyse kunt u net als met de correlatie de samenhang tussen variabelen vinden. Bij een regressieanalyse krijgt u echter wel een formule waarbij u toekomstige uitkomsten over een variabele kunt voorspellen. Bij een regressieanalyse zit het hem vooral in de interpretatie van de uitkomst.

Variabelen in een regressie
In dit hoofdstuk beperken we ons tot de eenvoudige lineaire regressieanalyse. Dit betekend dat er een variabele wordt voorspeld aan de hand van een andere variabele. De voorspelde variabele wordt de afhankelijke variabele genoemd. De variabele waarmee de afhankelijke variabele voorspeld wordt, heet de verklarende variabele.

Code
De code die R gebruikt om een eenvoudige regressieanalyse uit te voeren is **lm(***AFHANKELIJKE VARIABELE* **~***VERKLARENDE VARIABELE***,** *VARIABELE VAN DE DATASET***)**. U kunt elke numerieke variabele gebruiken voor de regressie. Echter kunt u met de ene combinatie beter voorspellingen doen dan met de ander.
Om de regressie te kunnen interpreteren, moet u de regressie in naam geven. Dit doet u met de code **<-** . Als u een regressieanalyse wilt uitvoeren en deze later wilt interpreteren gebruikt u dus de code *NAAM DIE U DE REGRESSIE WILT GEVEN* **<- lm(***AFHANKELIJKE VARIABELE* **~***VERKLARENDE VARIABELE***,** *VARIABELE VAN DE DATASET***)**.[15]

Bij het voorbeeld in **afbeelding 24** wordt de volgende code gebruikt **Regressie1<-lm(TevredenheidKlant~Afstand.Klant, Projecten)**.

Het geschatte model
Bij het voorbeeld in **afbeelding 24** wordt de tevredenheid van de klant verklaard met de afstand die tussen het bedrijf en de klant is. R geeft aan de hand van de regressie de volgende formule: *TevredenheidKlant = 2.524811 + - 0,003409X*, waarbij x staat voor het aantal kilometers. Deze formule heet het geschatte model. Als u alleen de naam van de regressie invoert, verschijnt het geschatte model.

[15] U hoeft niet constant de regressie een andere naam te geven. Als u dezelfde naam gebruikt voor nieuwe regressie, wordt deze over de andere regressie opgeslagen. Als u uw oude regressie dus wilt bewaren moet u voor de volgende regressies wel andere namen kiezen.

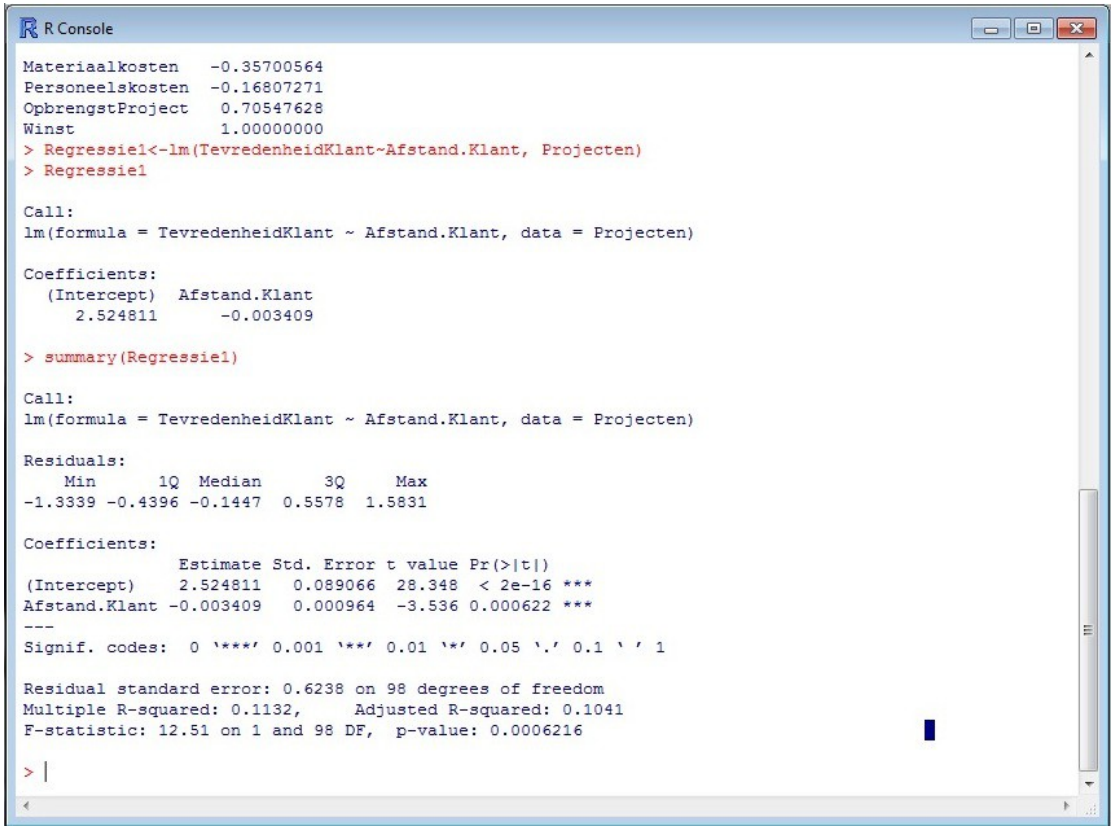
De regressieanalyse uitvoeren

Met de code **summary(***NAAM VAN DE REGRESSIE***)** krijgt u het overzicht van de regressie. Hierbij moet u kijken naar het gedeelte dat onder de tekst *Coefficients* staat. U ziet in achter de rijen *(Intercept)* en *(Afstand.Klant)* dezelfde waarden als in het geschatte model staan.[16]

In het voorbeeld van **afbeelding 24** zijn dezelfde waarden te zien als bij de uitkomst van de regressieanalyse.

Significantie

Achter de rijen staan 3 sterren weergegeven. De sterren geven aan dat de betreffende rij significant is, dit wil zeggen dat het voor deze regressie van belang is dat deze variabele wordt gebruikt. Een aanduiding met één ster is al genoeg om aan te tonen dat de variabele significant is. Des te meer sterren achter de rij staan, des te meer significantie de variabele heeft.
Als er achter de betreffende rij geen sterren staan, is de variabele niet significant en kan het dus weg gelaten worden in de regressie. De variabele is dan niet verklarend genoeg in de regressie.



Afbeelding 24 Een regressieanalyse uitvoeren met R.

Voorspellingskracht van de regressie / Adjusted R-squared[17]

In het resultaat vindt ook de *Adjusted R-squared*. Deze geeft de voorspellingskracht van het geschatte model aan. Het percentage dat het geschatte model juist voorspelt wordt achter de *Adjusted R-squared* weergegeven in decimalen. Een *Adjusted R-squared* van 1 betekent dus dat 100% van de voorspellingen die met het geschatte model gedaan worden juist zijn. Een *Adjusted R-squared* van 0,5 betekend dat 50% van de voorspellingen die met het geschatte model gedaan worden juist zijn.

---

[16] Op de kolommen *Std.Error*, *t value en Pr(>ItI)* hoeft u niet te letten. De waarden onder deze kolommen zijn voor sommige statistische berekeningen van waarde, echter niet in deze cursus.

[17] Word ook wel de determinatie coëfficiënt genoemd.

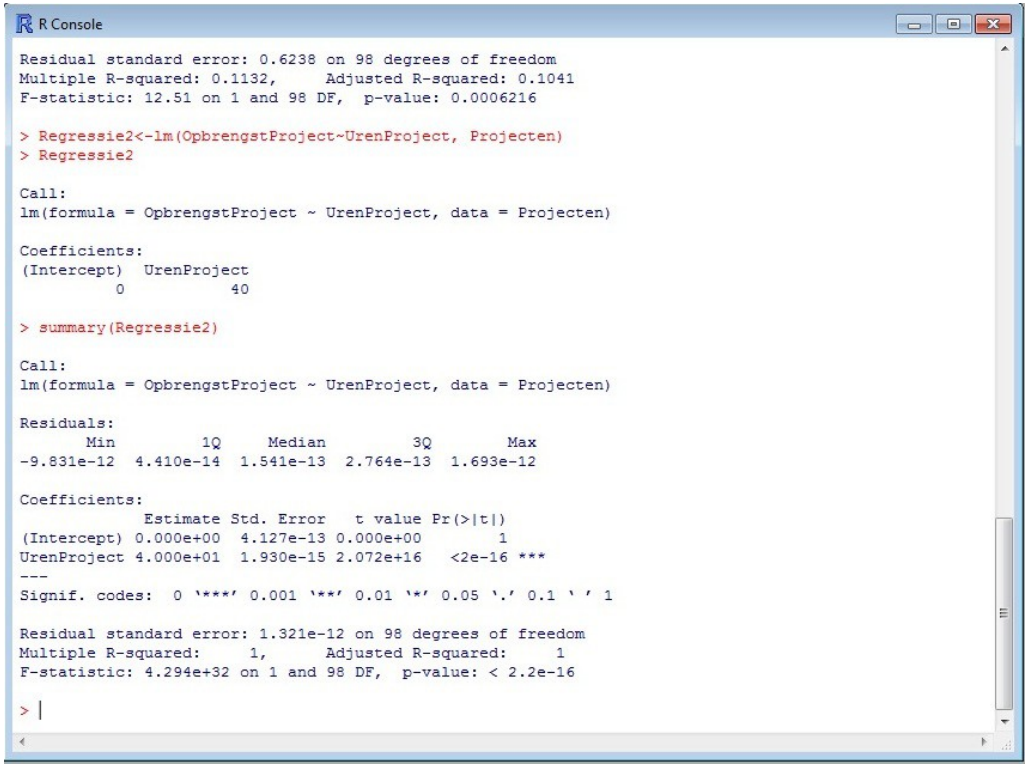
Het is aan de gebruiker zelf om te beoordelen, aan de hand van *Adjusted R-squared,* of het geschatte model goed genoeg is. Over het algemeen is een model met een *Adjusted R-squared* boven de 0,5 een redelijk model.

Bij het voorbeeld dat wordt gegeven in **afbeelding 24** is de *Adjusted R-squared* 0.1041. Dit betekent dat in 10,41% van de gevallen het geschatte model de tevredenheid van de klant juist voorspeld. Omdat dit vrij laag is kan het geschatte model *TevredenheidKlant = 2.524811 + - 0,003409x* als slecht beoordeeld worden.

In **afbeeldin25** wordt er een andere regressie uitgevoerd. Hier wordt er naar een voorspellingsmodel gezocht om de opbrengst van het project te voorspellen met het aantal uren dat er in is gestoken.
Hiervoor wordt de volgende code gebruikt **Regressie2<-lm(OpbrengstProject~UrenProject, Projecten)**.

Door de code **Regressie2** in te voeren, komt het volgende geschatte model tevoorschijn: *OpbrengstProject = 0 + 40X.* Hierin is *x* het aantal uren dat er in het project wordt gestoken.

Met de code **Summary(Regressie2)** komen de gegevens van de regressie tevoorschijn. Daarin is te zien dat alleen de variabele *Uren Project* significant is (Intercept kan net zo goed weggehaald worden omdat deze 0 is).
Verder is te zien dat de *Adjusted R-squared* gelijk aan 1 is. Dat wil zeggen dat met het geschatte model 100% betrouwbaar is. Dit is logisch want blijkbaar rekent het bedrijf 40 per uur en stijgt de opbrengst met 40 voor elk uur dat er in het project gestoken wordt.



Afbeelding 25 Een nieuwe regressieanalyse uitvoeren met R.

2.8 Meervoudige Regressieanalyse.

In het vorige hoofdstuk is besproken hoe u een eenvoudige lineaire regressie met R kunt maken. Hierbij wordt er een afhankelijke variabele voorspeld met één verklarende variabele. Echter kunt u ook een afhankelijke variabele voorpellen met meerdere verklarende variabelen. Dit wordt een meervoudige regressie genoemd.

Code

Bij R is er weinig verschil in code tussen een enkelvoudige regressie en een meervoudige regressie.
Een meervoudige regressie kunt u uitvoeren met de volgende code: **(***NAAM VAN DE REGRESSIE***)<-lm(***AFHANKELIJKE VARIABELE***~***VERKLARENDE VARIABELE 1***+***VERKLARENDE VARIABELE 2***+***VERKLARENDE VARIABELE X***, ***VARIABELE VAN DE DATASET***)**. U kunt zo veel verklarende variabelen gebruiken als u wilt, zolang u ze maar met een + in de code blijft toevoegen.

Op dezelfde manier als bij een enkelvoudige regressie kunt u het geschatte model en de gegevens van de regressie tevoorschijn halen. Respectievelijk met de codes: *NAAM VAN DE REGRESSIE*, en **summary(***NAAM VAN DE REGRESSIE***)**.

Interpretatie van de regressie

In tegenstelling tot de enkelvoudige regressie, staan er in het resultaat van de meervoudige regressie meerdere rijen van variabelen. Het principe is hetzelfde, aan de hand van de sterren achter de rij worden de significanties van de variabelen laten zien.

```
> Regressie<-lm(Winst~Personeelskosten+Materiaalkosten+TevredenheidKlant, Projecten)
> Regressie

Call:
lm(formula = Winst ~ Personeelskosten + Materiaalkosten + TevredenheidKlant,
    data = Projecten)

Coefficients:
      (Intercept)    Personeelskosten    Materiaalkosten  TevredenheidKlant
        8845.8805           -0.2964           -1.9292         -1338.5787

> summary(Regressie)

Call:
lm(formula = Winst ~ Personeelskosten + Materiaalkosten + TevredenheidKlant,
    data = Projecten)

Residuals:
    Min      1Q  Median      3Q     Max
-4869.0 -1123.5    15.4  1281.8  6511.4

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       8845.8805   943.3177   9.377 3.24e-15 ***
Personeelskosten    -0.2964     0.1038  -2.855 0.005278 **
Materiaalkosten     -1.9292     0.9469  -2.037 0.044364 *
TevredenheidKlant -1338.5787   360.4634  -3.713 0.000343 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2046 on 96 degrees of freedom
Multiple R-squared: 0.2756,     Adjusted R-squared: 0.2529
F-statistic: 12.17 on 3 and 96 DF,  p-value: 8.094e-07
```

Afbeelding 26 Een meervoudige regressieanalyse uitvoeren met R.

In **afbeelding 26** wordt er de volgende regressieanalyse gedaan:
De winst voorspellen aan de hand van personeelskosten, materiaalkosten en de tevredenheid van de klant. Hiervoor wordt de volgende code gebruikt.
Regressie<-lm(Winst~Personeelskosten+Materiaalkosten+TevredenheidKlant, Projecten). Door de naam van de

regressie **Regressie** in te toetsen, verschijnt het geschatte model: *Winst = 8845.8805 + (-0.2964)X1 + (-1.9292)X2 + (-1338.5787)X3*.
Waarbij X1 staat voor de personeelskosten, X2 voor de materiaalkosten en X3 voor de tevredenheid van de klant.

Met de code **summary(Regressie)** komt in het voorbeeld de regressieanalyse tevoorschijn. Valt het u op dat alle rijen van variabelen significant zijn? De een is echter minder significant dan de ander, maar toch zijn alle variabelen significant genoeg om in het model toe te passen.
In het voorbeeld is de *Adjusted R-squared* slechts 0.2529. De drie verklarende variabelen zijn dus slechte variabelen om de winst mee te voorspellen.

U hebt kunnen zien dat een meervoudige regressie op dezelfde manier werkt als een enkelvoudige regressie.

Let op! Het begrip voorspellen moet u interpreteren als voorspellen in uw eigen dataset. Uw dataset is een afspiegeling van uw eigen situatie, niet hoe het er in alle situaties op elk moment in de wereld aan toe gaat. Het blijven namelijk maar statistische functies die verbanden in uw data analyseren. U bent met de regressieanalyse dus beperkt uitspraken te doen over voorspellingen in uw eigen situatie.

2.9 Meervoudige of enkelvoudige regressieanalyse met categorische variabelen.

In de vorige twee hoofdstukken is er een regressieanalyse gedaan met numerieke variabelen, variabelen met een numerieke waarde. Het kan wel eens voorkomen dat u in een databestand niet-numerieke variabelen of waarden tegenkomt. Deze worden categorische variabelen genoemd. Dit zijn variabelen die namen of categorieën bevatten.

Het voorbeeldbestand

Bij de laatste twee hoofdstukken wordt er in de voorbeelden gebruik gemaakt van het databestand *Projecten2.csv*.
U kunt zien dat het dezelfde data bevat als het bestand *Projecten.csv.*Echter zijn er aan dit bestand 3 nieuwe, categorische, variabelen toegevoegd: *Werkgroep(A,B of C)*, *Maand (januari t/m december)* en *TypeProject(Alfa, Beta, Gamma, Delta).*

De code

De code die gebruikt wordt voor een regressieanalyse met categorische variabelen verschild niet veel met de regressie met alleen numerieke variabelen.
Het verschil is dat de categorische variabele in de code omgezet moet worden in een numerieke variabele. Dit gaat met de code **factor(***NAAM VAN DE CATEGORISCHE VARIABELE**)**. Dit stukje code moet u in de regressie toepassen. Om dit duidelijk aan u te kunnen laten zien, wordt hier een voorbeeld voor gebruikt.

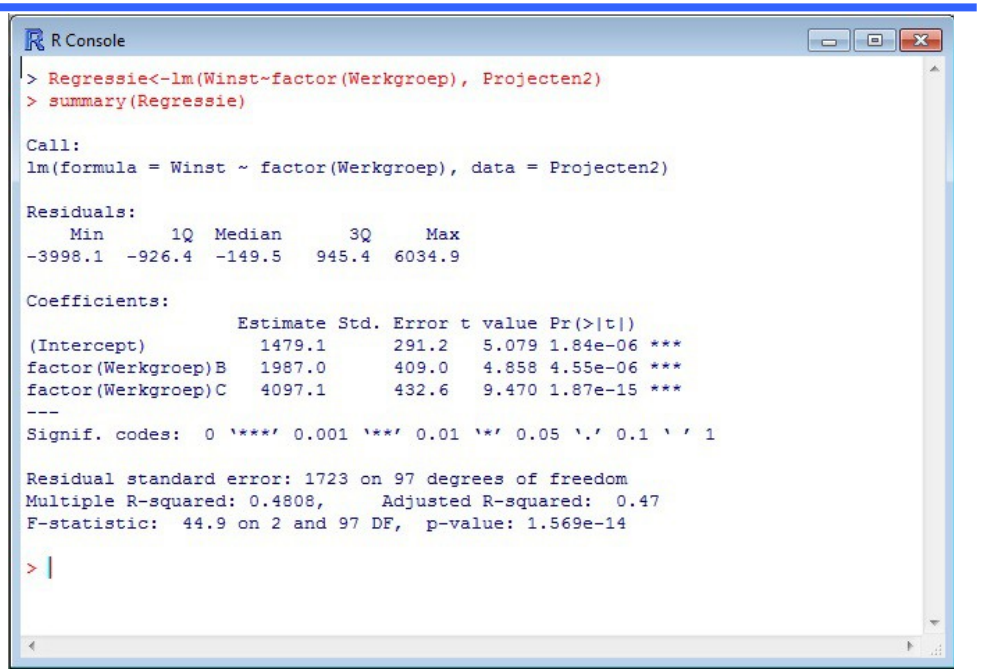
```
R Console

> Projecten2<-read.csv('Projecten2.csv')
> attach(Projecten2)
The following object(s) are masked from 'Projecten':

    Afstand.Klant, Materiaalkosten, Personeelskosten, Winst
The following object(s) are masked from 'Projecten2 (position 4)':

    Afstand.Klant, Maand, Materiaalkosten, Opbrengst.Project,
    Personeelskosten, Tevredenheid.Klant, Type.Project, Uren.Project,
    Werkgroep, Winst
> summary(Projecten2)
 Tevredenheid.Klant Afstand.Klant     Uren.Project   Materiaalkosten
 Min.   :1.0        Min.   :  2.00   Min.   :100.0   Min.   :123.0
 1st Qu.:2.0        1st Qu.: 24.00   1st Qu.:150.0   1st Qu.:448.5
 Median :2.0        Median : 41.50   Median :206.0   Median :571.0
 Mean   :2.3        Mean   : 65.95   Mean   :202.6   Mean   :603.6
 3rd Qu.:3.0        3rd Qu.: 72.00   3rd Qu.:241.0   3rd Qu.:846.5
 Max.   :3.0        Max.   :325.00   Max.   :424.0   Max.   :987.0

 Personeelskosten Opbrengst.Project     Winst        Werkgroep      Maand
 Min.   :1235     Min.   : 4000     Min.   :-2519   A:35      april   : 9
 1st Qu.:2321     1st Qu.: 6000     1st Qu.: 1880   B:36      februari: 9
 Median :3521     Median : 8240     Median : 3605   C:29      januari : 9
 Mean   :4116     Mean   : 8102     Mean   : 3383             maart   : 9
 3rd Qu.:5540     3rd Qu.: 9640     3rd Qu.: 4984             augustus: 8
 Max.   :8951     Max.   :16960     Max.   : 8010             december: 8
                                                              (Other) :48

 Type.Project
 Alfa :30
 Beta :22
 Delta:15
 Gamma:33


>
```

Afbeelding 27 Het nieuwe databestand *Projecten 2* importeren.

Enkelvoudige regressieanalyse met categorische variabelen

Bij het voorbeeld in **afbeelding 28** wordt de afhankelijke variabele *Winst* voorspeld met de verklarende variabele *Werkgroep*. Omdat werkgroep een categorische variabele is en er voor een regressieanalyse met numerieke waarden gerekend moet worden, moet de categorische variabele *Werkgroep* omgezet worden in een numerieke variabele. In het

http://www.arietwigt.wordpress.com/
voorbeeld wordt dit gedaan met de volgende code: **Regressie<-lm(Winst~factor(Werkgroep), Projecten2)**. Door **Regressie** in te toetsen wordt het geschatte model gepresenteerd. Het geschatte model bij deze regressieanalyse is *Winst = 1479.1 +(0 A) + (1987 B) + (4097.1 C)*[18]. Hierbij wordt alleen de waarde opgeteld voor de betreffende werkgroep. Als de winst dus geschat moet worden als Werkgroep C het project doet, is de berekening: 1479.1 + 1987 = 3.466.1. Door de code **summary(Regressie)** wordt de regressieanalyse gepresenteerd.



Afbeelding 28 Categorische variablen bij de regressieanalyse gebruiken.

U kunt zien dat het de code **factor()** in de code van de regressie wordt geplaats. Het kan gezien worden als een kleine toevoeging aan de categorische variabele.

Meervoudige regressieanalyse met categorische variabelen

Een regressieanalyse uitvoeren met meerdere categorische variabelen gaat volgens hetzelfde principe als bij een enkelvoudige regressieanalyse met categorische variabelen. Hiervoor gebruikt u de volgende code *NAAM VAN DE REGRESSIE***<-lm(***AFHANKELIJKE VARIABELE***~ factor(***VERKLARENDE VARIABELE 1***)+ as.numeric(***VERKLARENDE VARIABELE 2***)+ factor(***VERKLARENDE VARIABELE X…***), VARIABELE VAN HET BESTAND)**.
Bij het voorbeeld in **afbeelding** ziet u dat de afhankelijke variabele *Winst* voorspeld wordt met de verklarende variabelen *Werkgroep* en *Type.Project*. Hiervoor wordt de volgende code gebruikt:
Regressie<-lm(Winst~factor(Werkgroep)+as.numeric(Type.Project), Projecten2).

Door **Regressie** in te voeren verschijnt het geschatte model: *Winst = 2706 + (0 A) + (1123 B) +(3274 C) + (-1121 Beta)+ (-2562 Delta) + (-140 Gamma)*. Ook in deze regressie met categorische variabelen wordt alleen de waarde opgeteld als de categorie van toepassing is. Als bijvoorbeeld de winst voorspeld moet worden als werkgroep C type project Gamma gaat doen, is de voorspelde winst 2706 + 3274 -140 = 5840.

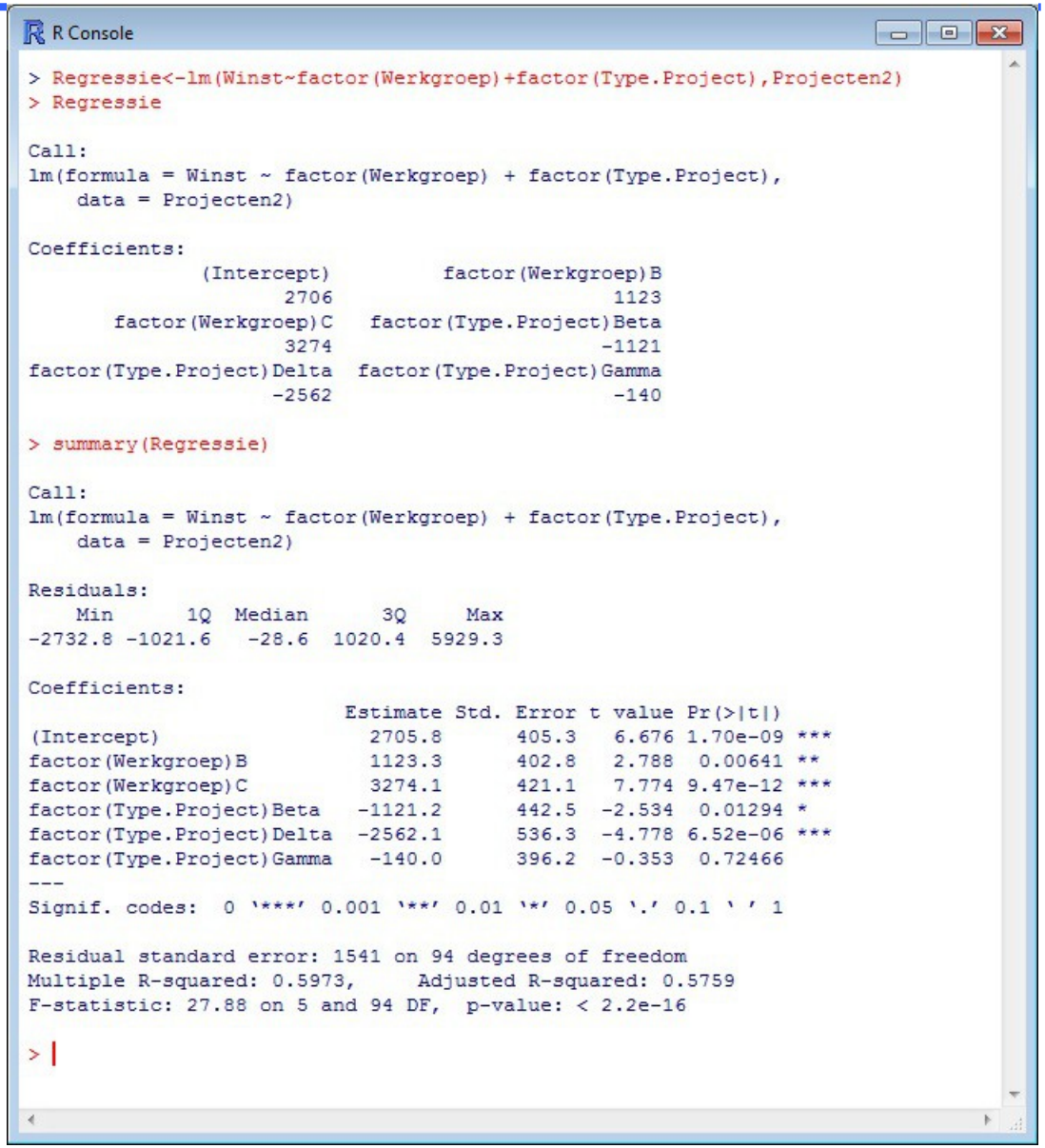
[18] U vraagt zich misschien af waarom de waarde bij A nul is. Deze zit al in de *Intercept* (1479) verwerkt.

http://www.arietwigt.wordpress.com/

Door de code **summary(Regressie)** in te voeren, wordt de regressieanalyse gepresenteerd.

Aan de rijen van variabelen kunt u zien dat niet alle variabelen even significant zijn. Bij het voorbeeld in **afbeelding 29** is de rij met de factor *Gamma* bijvoorbeeld niet significant. Dit betekend dat de voorspellingen die worden gedaan als gamma er bij wordt betrokken, niet erg betrouwbaar zijn. Het is niet erg om de factor Gamma in de formule te laten staan, maar de voorspellingen die worden gedaan met betrekking tot Gamma zijn niet betrouwbaar.

De *Adjusted R-square* van 0.5759 laat zien dat de kwaliteit van het geschatte model redelijk hoog is.



Afbeelding 29 De meervoudige regressieanalyses met categorische variabelen interpreteren.

Appendix

Codes in R die relevant voor u kunnen zijn.

NAAM NIEUWE VARIABELE<-C(WAARDE,WAARDE,WAARDE,MEERDERE WAARDEN)	EEN VARIABELE TOEWIJZEN AAN DE HAND VAN EEN WAARDE OF EEN REEKS WAARDEN.
NAAM NIEUWE VARIABELE<-*DATA OF ANDERE WAARDEN*	EEN VARIABELE TOEWIJZEN AAN EEN DATASET OF ANDERE WAARDE.
ATTACH(*VARIABELE*)	EEN MATRIX MAKEN VAN EEN VARIABELE DIE EEN DATASET BEVAT.
MIN(*VARIABELE*) OF MIN(*VARIABELE*, *VARIABELE* ….MEERDERE VARIABELEN)	DE MINIMUM WAARDE UIT EEN VARIABELE OF UIT EEN REEKS VARIABELEN OPZOEKEN.
MAX(*VARIABELE*)OF MAX(*VARIABELE*, *VARIABELE* ….MEERDERE VARIABELEN)	DE MAXIMUM WAARDE UIT EEN VARIABELE OF UIT EEN REEKS VARIABELEN UITZOEKEN.
LENGHT(*VARIABELE*)	HET AANTAL WAARDEN IN EEN VARIABELE WEERGEVEN.
SUM(*VARIABELE*)	DE WAARDEN VAN ALLE VARIABELEN BIJ ELKAAR OPTELLEN.
MEAN(*VARIABELE*)	HET GEMIDDELDE VAN DE WAARDEN DIE EEN VARIABELE BEVAT BEREKENEN.
SD(*VARIABELE*)	DE STANDAARDDEVIATIE VAN DE WAARDEN DIE EEN VARIABELE BEVAT BEREKENEN.
detach(*VARIABELE*)	DE MATRIX VAN DE VARIABELE LOSKOPPELEN.

MEDIAN(*VARIABELE*)	De mediaan uit de waarden die de variabele bevat, berekenen.
COR(*VARIABELE1*, *VARIABELE2*)	De correlatie tussen twee variabelen berekenen.
PLOT(*VARIABELE1*, *VARIABELE2*)	Een grafiek maken van twee variabelen.
HIST(*VARIABELE*)	Een histogram maken van een variabele.
BOXPLOT(*VARIABELE*)	Een box plot maken van een variabele.
BARPLOT(*VARIABELE1*, *VARIABELE2*, MEERDERE VARIABELEN)	Een staafdiagram maken van meerdere variabelen.
DOTPLOT(*VARIABELE1*, *VARIABELE2*)	Een puntdiagram maken van twee variabelen.
PIE(*VARIABELE*)	Een cirkeldiagram maken van een variabele.
VARIABELE[*POSITIE OF NAAM VARIABELE 1*, *POSITIE OF NAAM VARIABELE 2*]	De waarde opzoeken in een matrix van een variabele.
READ.CSV('*NAAM VAN HET CSV-BESTAND*')	Een csv-bestand importeren in R.
NAAM VAN DE REGRESSIE<-LM(*AFHANKELIJKE VARIABELE*~*VERKLARENDE VARIABELE1*+*VERKLARENDE VARIABELE2*+*MEERDERE VARIABELEN*)	Een regressie analyse maken met een afhankelijke variabele of en één of meerdere verklarende variabelen.
SUMMARY(*NAAM VAN DE REGRESSIE*)	De gegevens van de regressieanalyse tevoorschijn halen.

http://www.arietwigt.wordpress.com/

Packages installeren

Zoals bij de inleiding al is aangegeven, is R een open source softwarepakket. Het is voor iedereen dus vrij om verschillende toepassingen te ontwikkelen die toegepast kunnen worden met R. Zo'n toepassing wordt bij R een package genoemd. Voorbeelden van packages zijn:

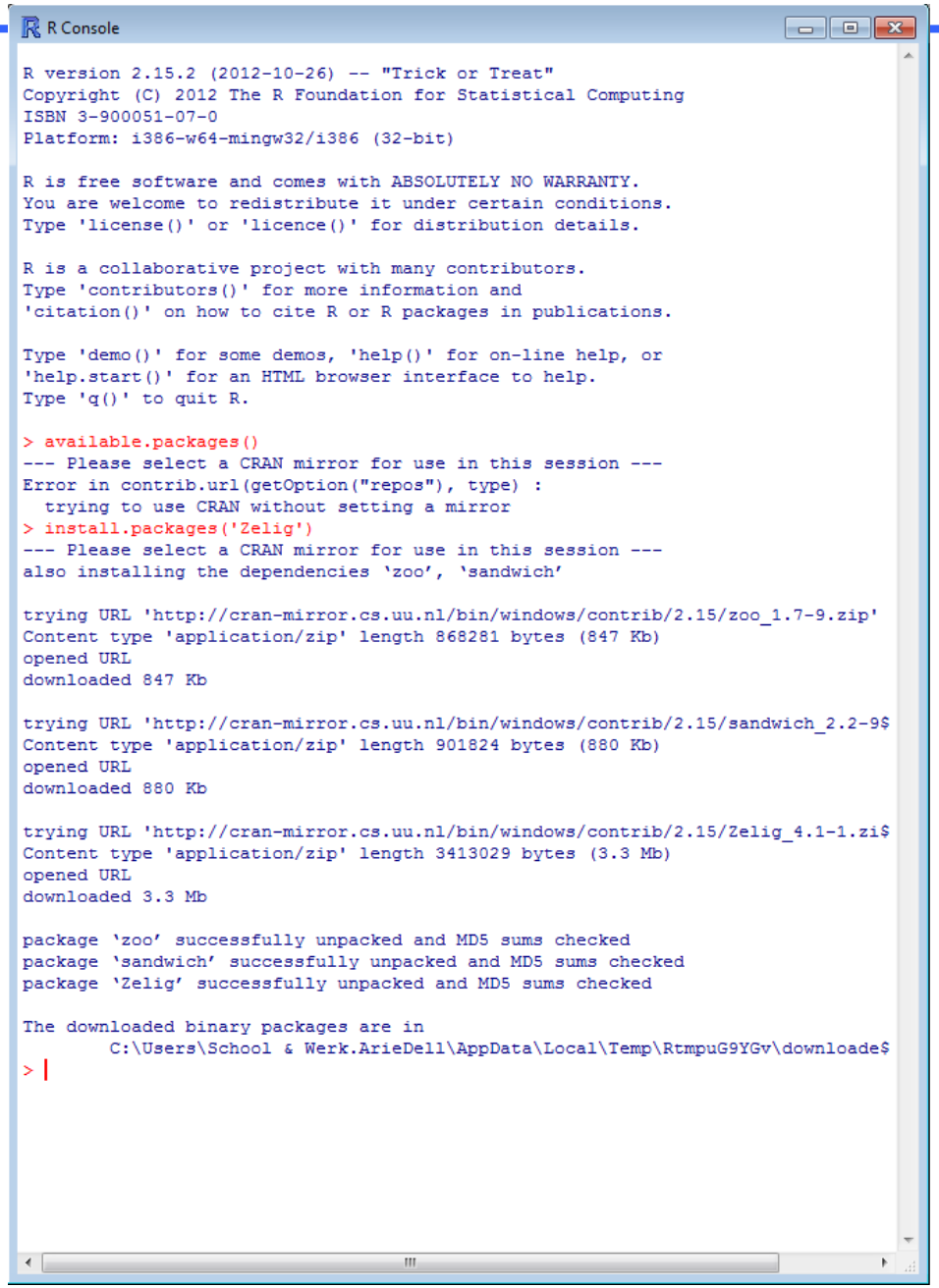
- *Zelig*: Om op een nog meer uitgebreide manier regressieanalyses uit te voeren.
- *Foreign*: Om meer bestandtypen met R te kunnen lezen.

Als u R heeft gedownload heeft R nog geen packages geïnstalleerd. Dit is in eerste instantie geen probleem omdat u met R zonder packages prima data kunt analyseren, in ieder geval op de manier die in deze handleiding wordt uitgelegd.

Mocht u toch packages willen installeren in uw R console, kunt u dat op de volgende manieren doen:

Voer de code **available.packages()** in. Als u op Enter drukt verschijnt er een venster waarin u een mirror moet selecteren. Selecteer de mirror voor het land waarin u zich bevindt.
Als u dit gedaan heeft verschijnt er een lijst met packages die u kunt installeren. Als u een package heeft gevonden, kopieert u de naam van de package. En plakt het tussen haakjes en aanhalingstekens in de code **install.packages('***NAAM VAN DE PACKAGE DIE U WILT INSTALLEREN****')**. Door op Enter te drukken wordt de package geïnstalleerd en geeft R aan in welke map het bestand van deze package zich bevindt.

-EINDE VAN DE MANUAL-



Afbeelding 30 Een package installeren in R.