Cursus

Academiejaar 2018-2019

Dr. Jens Buysse, Bert Van Vreckem

HoGent
BEDRIJF
EN
ORGANISATIE

WWW.HOGENT.BE

Gegenereerd op 20 augustus 2018

Inhoudsopgave

Voorwoord

Deze cursus werd geschreven in het kader van de lessenreeks Onderzoekstechnieken aan de Hogeschool Gent. Ik wil hierbij gebruik maken om volgende mensen te bedanken bij het nakijken en verbeteren van de cursus.

- Cédric Berlez
- Jürgen Van Meerhaeghe
- Gianni Stubbe
- Jelle Elaut
- Thijs Van Der Burgt
- Lotte Potthé
- Özgür Akin
- Cedric De Vylder

Jens Buysse
08 februari 2016

1. Aan de slag

1.1 Studiewijzer

De studiewijzer geeft een overzicht van de belangrijkste informatie over deze cursus, o.a. leerdoelen, lesmateriaal, weekplanning en leeraanwijzingen. Lees alles aandachtig door!

1.1.1 Doel en plaats van de cursus in het curriculum

Deze cursus is een inleiding op wat tegenwoordig vaak *data science* genoemd wordt. Het doel is om je wegwijs te maken in het correct verzamelen, verwerken en analyseren van numerieke data en daar een onderzoeksverslag over te schrijven.

In de eerste plaats is dit een voorbereiding op de bachelorproef, waar je deze technieken in de praktijk zal moeten omzetten. Maar ook na je afstuderen blijft de kennis die je in deze cursus opdoet waardevol. Succesvolle bedrijven nemen beslissingen, niet op basis van buikgevoel of intuïtie, maar door het verzamelen van data. Aan de hand van de technieken die hier toegelicht worden, heb je voldoende achtergrond om vragen te beantwoorden als:

- Is een (web)applicatie snel genoeg voor de gebruikers? Is de gebruikerservaring consistent, of zit er grote variatie op responstijden?
- Welk van twee systemen (software of hardware) is het meest performant? Is het verschil tussen beide significant, of kunnen verschillen in de metingen te wijten zijn aan het toeval?
- Wanneer moeten aankopen van nieuwe apparatuur (bv. harde schijven, servers, geheugen, enz.) ingepland worden, op basis van historische gebruiksgegevens?

1.1.2 Leerdoelen en competenties

- Kan begrippen, formules, stellingen en de uitwerking ervan uit de beschrijvende en inductieve statistiek benoemen en verklaren
- Kan formules, stellingen uit de beschrijvende en inductieve statistiek in onderzoeksvraagstukken correct toepassen
- Kan data analyseren met statistische software
- Kan een gestructureerd wetenschappelijk document schrijven en voorzien van referenties in LaTeX
- Kan de wetenschappelijke methode vergelijken met niet-wetenschappelijke onderzoeksmethodes en daarbij voor- en nadelen opsommen

Deze vind je ook terug in de studiefiche.

1.1.3 Leerinhoud

Verder in dit hoofdstuk vind je instructies voor het installeren van de nodige software, en een korte inleiding op het werken met R, een programmeertaal voor data-analyse.

Hoofdstuk **??** geeft een inleiding op het verloop van een typisch onderzoeksproces en introduceert enkele basisconcepten van data-analyse.

Hoofdstuk **??** behandelt de analyse van een enkele variabele, meer bepaald centrum- en spreidingsmaten, en ook geschikte grafiektypes voor elk soort variabelen.

Hoofdstuk **??** introduceert het concept van het nemen van steekproeven uit een populatie, en de randvoorwaarden waaronder resultaten binnen een steekproef kunnen veralgemeend worden tot de gehele populatie.

Hoofdstuk **??** gaat hierop verder met de algemene werkwijze voor het voeren van statistische toetsen, en specifiek met toetsen voor uitspraken over het gemiddelde van een populatie: de z-toets en de t-toets.

Waar de vorige hoofdstukken telkens één variabele apart beschouwden, bekijkt Hoofdstuk **??** verschillende technieken om verbanden tussen twee variabelen te leggen, afhankelijk van het variabeletype.

Hoofdstuk **??** introduceert de χ^2-toets, waarmee je kan nagaan of de verdeling van een steekproef relevant is voor een populatie, of in hoeverre twee steekproeven een gelijkaardige verdeling hebben.

Hoofdstuk **??** geeft een inleiding op het analyseren van hoe de waarde van een variabele evolueert in de tijd aan de hand van wiskundige modellen die onder bepaalde voorwaarden ook toelaten om voorspellingen te doen.

1.1.4 Leermateriaal

Het belangrijkste leermateriaal voor dit opleidingsonderdeel is deze cursus, die ook de oefeningenopgaven bevat. Die wordt ter beschikking gesteld via Chamilo als PDF. Op Chamilo vind je ook de PDF's met de slides gebruikt tijdens de lessen.

Daarnaast krijgen studenten toegang tot een GitHub-repository met de broncode voor:

- Deze cursus
- De slides van lessen
- Broncodevoorbeelden in R voor alle technieken die in de cursus aan bod komen.

Errata en wijzigingen aan de cursus worden in GitHub aangebracht. De PDF's op Chamilo zullen niet noodzakelijk bijgewerkt worden. Studenten kunnen zelf de laatste versies van alle documenten met LaTeX genereren.

De software die nodig is voor dit opleidingsonderdeel is gratis/open source. Instructies voor de installatie kan je vinden in Sectie 1.2.

1.1.5 Werkvormen

Studenten afstandsleren kunnen vragen stellen tijdens de contactmomenten. Dit zijn echter geen lesmomenten! Het rooster vind je in de Chamilo-cursus "Informatie voor studenten TILE."

Studenten dagonderwijs krijgen één uur per week hoorcollege en twee uur werkcollege.

1.1.6 Werk- en leeraanwijzingen

Het opleidingsonderdeel *Onderzoekstechnieken* wordt door veel studenten als moeilijk ervaren. Dat is begrijpelijk, want het onderwerp ligt dan ook buiten de comfortzone van de doorsnee informatica-student en we weten allemaal dat wiskundige vakken niet de populairste van onze opleiding zijn.

Er zijn twee manieren om hier mee om te gaan. Je kan de weg van de minste weerstand nemen: je concentreren op de vakken die je graag doet en een dag voor het examen de cursus doornemen in de hoop dat je voldoende punten bij elkaar sprokkelt om een tien te halen. De ervaring leert dat deze strategie niet succesvol is, wat blijkt uit het lage slagingspercentage in de eerste zittijd (in academiejaar 2016-2017 was dat ca. 35% voor het dagonderwijs en 10% voor afstandsleren).

Enkele tips om wél meteen te slagen voor dit vak:

- Kom naar de theorieles en *neem actief nota's*;
- Werk ook voor dit vak *buiten de contactmomenten*. Herhaal de geziene theorie en werk oefeningen af waarmee je nog niet klaar was. Noteer zaken die je niet snapt of waar je vast zit, en stel je vraag bij het eerstvolgende werkcollege.
- Gebruik goede *leertechnieken*. Je vindt een goed overzicht van leertechnieken waarvan het effect wetenschappelijk aangetoond is via de website van *The Learning Scientists*[1].
  - *Spaced practice:* Studeer in meerdere kleine sessies (minstens één keer per week) en niet in grote blokken. Blokkeer een vast moment in je weekagenda/lesplanning.
  - *Retrieval practice:* Neem een leeg blad papier en probeer zoveel mogelijk zaken over een bepaald onderwerp op te schrijven vanuit je herinnering (dus zonder in de cursus te kijken). Controleer dit daarna aan de hand van je lesnota's en in de cursus.
  - *Elaboration:* Stel jezelf vragen over hoe dingen (bv. formules, toetsingsprocedures, ...) in elkaar zitten en waarom dat zo is. Overleg met medestudenten. Vraag je lector

[1] http://www.learningscientists.org/

Week	Theorie	Oefeningen
1	Intro, Onderzoeksproces	Software installeren, LaTeX
2	Analyse van 1 variabele	Wetenschappelijk schrijven
3	Steekproefonderzoek	Analyse van 1 variabele
4	Steekproefonderzoek	Steekproefonderzoek
5	Toetsingsprocedures (z-toets)	Steekproefonderzoek
6	Toetsingsprocedures (t-toets)	Toetsingsprocedures
7	Analyse van 2 variabelen	Toetsingsprocedures
—	**Paasvakantie**	—
8	Analyse van 2 variabelen	Analyse van 2 variabelen
9	χ^2-toets	Analyse van 2 variabelen
10	Tijdreeksen	χ^2-toets
11	Toelichting bachelorproef	Tijdreeksen
12	Herhaling	Herhaling

Tabel 1.1: Weekplanning van de cursus.

om meer uitleg indien nodig. Leg verbanden tussen verschillende onderwerpen in de cursus (bv. vergelijk toetsingsprocedures).
- *Interleaving:* Wissel onderwerpen af tijdens het studeren.
- Gebruik *concrete voorbeelden* om abstracte ideeën te begrijpen. In de cursus worden voorbeelden gegeven, probeer er zelf te bedenken. Overleg met medestudenten en vraag eventueel feedback aan je lector.
- *Dual coding:* Combineer woord en beeld, probeer de leerstof die je instudeert visueel voor te stellen.

Uiteindelijk komt het er op neer dat je voldoende tijd en inspanning investeert om te studeren voor dit vak.

1.1.7 Studiebegeleiding en planning

Studenten **afstandsleren** die vragen hebben over de leerstof kunnen in de eerste plaats terecht op het forum in Chamilo. Wanneer je een oefening gemaakt hebt en twijfelt over de correcte oplossing, kan je de lector per mail contacteren. Zet dan in de onderwerpregel "[OZT][TILE]". Deze mails worden niet dagelijks beantwoord, dus het kan even duren voordat je reactie krijgt.

Studenten **dagonderwijs** kunnen vragen stellen tijdens de werkcolleges, of ook op het forum.

In Tabel 1.1 vind je een overzicht van de lesplanning voor het dagonderwijs die ook als leidraad kan dienen voor de studieplanning van studenten afstandsleren.

1.1.8 Evaluatie

Dagonderwijs

- Eerste examenperiode:
  - 70% periodegebonden evaluatie: schriftelijk examen, bestaande uit een deel gesloten

boek (theorie) en een deel met voorbereiding op pc (oefeningen)
  - 30% niet-periodegebonden evaluatie: het voeren van een mini-onderzoek in groep, bestaande uit een literatuurstudie, opzetten van een reproduceerbaar experiment, verzamelen van meetgegevens en die statistisch analyseren, en er een verslag over schrijven
- Tweede examenperiode:
  - 70% periodegebonden evaluatie: schriftelijk examen, bestaande uit een deel gesloten boek (theorie) en een deel met voorbereiding op pc (oefeningen)
  - 30% niet-periodegebonden evaluatie: er wordt geen tweede examenkans georganiseerd voor dit onderdeel. Wanneer een student in de eerste examenkans niet geslaagd was voor de opdracht blijft de beoordeling voor deze evaluatievorm of de afwezigheid voor deze evaluatievorm geldig voor de tweede examenkans.

Afstandsleren

- Eerste examenperiode:
  - 70% periodegebonden evaluatie: schriftelijk examen, bestaande uit een deel gesloten boek (theorie) en een deel met voorbereiding op pc (oefeningen)
  - 30% niet-periodegebonden evaluatie: individuele opdracht, schrijven van een paper
- Tweede examenperiode:
  - 70% periodegebonden evaluatie: schriftelijk examen, bestaande uit een deel gesloten boek (theorie) en een deel met voorbereiding op pc (oefeningen)
  - 30% niet-periodegebonden evaluatie: er wordt geen tweede examenkans georganiseerd voor dit onderdeel. Wanneer een student in de eerste examenkans niet geslaagd was voor de opdracht blijft de beoordeling voor deze evaluatievorm of de afwezigheid voor deze evaluatievorm geldig voor de tweede examenkans.

1.2 Installatie software

Voor de cursus onderzoekstechnieken maak je gebruik van verschillende softwarepakketten. Hier vind je wat uitleg over de installatie en hoe je er mee aan de slag kan.

- Git client (versiebeheersysteem);
- LaTeX compiler;
- LaTeX editor;
- Jabref (bibliografische databank);
- R (statistische analysesoftware);
- Rstudio (IDE voor R).

Sommige van deze applicaties nemen veel schijfruimte in, dus zorg dat je voldoende ruimte vrij hebt.

In vele andere cursussen rond statistiek of onderzoekstechnieken wordt gebruik gemaakt van commerciële software: SPSS of SAS voor data-analyse, MS Office voor de opmaak van documenten. In deze cursus wordt er expliciet voor gekozen om open source of gratis software te gebruiken. Het grootste voordeel is dat je die ook na je afstuderen nog kan gebruiken zonder dat jij of je bedrijf/organisatie softwarelicenties moet aankopen.

Bovendien zijn de tools die we zullen gebruiken kwalitatief minstens even goed dan hun commerciële tegenhangers. R, een programmeertaal voor statistische analyse, wordt wereldwijd gebruikt

in academische én professionele context. Volgens de TIOBE-index[2] zit R intussen bijna in de top-10 van alle programmeertalen en de taal zit sinds een vijftal jaar in een vrij sterk stijgende trend. De kans is dus niet onbestaande dat je het in je professionele loopbaan nog zal tegenkomen, of het zal kunnen toepassen voor het oplossen van datagerelateerde problemen. Feedback die we kregen van oud-studenten bevestigt dit.

LaTeX is een markuptaal en tekstzetsysteem voor de professionele vormgeving van documenten. De bedoeling is dat de auteur zich vooral moet bezig houden met het logisch structureren van een tekst, en dat het vormgeven op papier wordt overgenomen door de software. Het aanleren van de markuptaal vraagt wat inspanning, maar het is een investering die rendeert wanneer je een lang document (zoals een scriptie) op een professionele, strakke manier wil opmaken. Er zijn in het verleden nog zelden of nooit bachelorproeven ingediend die in MS Word geschreven waren en die een voldoende goede opmaak hadden. Het lijkt veel eenvoudiger om een tekst op te stellen in Word, maar het is zo goed als onmogelijk om in een lang document een consistente en professioneel ogende opmaak te realiseren.

1.2.1 Windows

Omdat het hier toch gaat om een vrij groot aantal applicaties, kunnen Windows-gebruikers beter gebruik maken van de Chocolatey package manager[3] in plaats van alles manueel te downloaden en installeren.

Na installatie van Chocolatey[4], voer je volgende commando's uit als Administrator in een CMD of PowerShell terminal:

```
choco install -y git
choco install -y miktex
choco install -y texstudio
choco install -y JabRef
choco install -y r.project
choco install -y r.studio
```

Wie toch de "klassieke" werkwijze wil hanteren vindt hier de verschillende softwarepakketten:

- Git client: https://git-scm.com/download/win
- LaTeX compiler: https://miktex.org/download
- TeXStudio: http://www.texstudio.org/
- Jabref: https://www.fosshub.com/JabRef.html
- R: https://lib.ugent.be/CRAN/
- Rstudio: https://www.rstudio.com/products/rstudio/download/#download

[2] **\url {https://www.tiobe.com/tiobe-index/}**.
[3] https://chocolatey.org/
[4] https://chocolatey.org/install

1.2.2 macOS

macOS gebruikers installeren de nodige software best via de Homebrew[5] package manager[6]:

```
brew install git
brew cask install mactex
brew cask install texstudio
brew cask install jabref
brew install Caskroom/cask/xquartz
brew install --with-x11 r
brew cask install --appdir=/Applications rstudio
```

Wie toch alles manueel wil installeren kan de applicaties hier downloaden:

- Git client: https://git-scm.com/download/mac
- LaTeX compiler: https://www.tug.org/mactex/mactex-download.html
- TeXStudio: http://www.texstudio.org/
- Jabref: https://www.fosshub.com/JabRef.html
- R: https://lib.ugent.be/CRAN/
- Rstudio: https://www.rstudio.com/products/rstudio/download/#download

1.2.3 Linux

Op RStudio na zijn alle nodige softwarepakketten beschikbaar in de repositories van de meest gebruikte Linux-distributies. We geven hier command-line instructies voor enerzijds Ubuntu (Xenial/16.04) en Debian 9 en anderzijds Fedora.

Ubuntu/Debian

Controleer eerst de link naar de laatste versie van RStudio via de website.

```
sudo aptitude update
sudo aptitude install texlive-latex-base texlive-latex-extra \
  texlive-lang-european texlive-bibtex-extra texlive-extra-utils \
  biber git texstudio jabref r-base
wget https://download1.rstudio.org/rstudio-xenial-1.1.414-amd64.deb
sudo dpkg -i ./rstudio-xenial-1.1.414-amd64.deb
```

Fedora

Controleer eerst de link naar de laatste versie van RStudio via de website. Dit is één lang commando:

```
sudo dnf install git texstudio R \
```

[5] https://brew.sh/

[6] **Let op!** Deze werkwijze is nog niet getest. Feedback van Mac-gebruikers is welkom!

```
java-1.8.0-openjdk-openjfx texlive-collection-latex \
texlive-texliveonfly texlive-babel-dutch \
https://download1.rstudio.org/rstudio-1.1.414-x86_64.rpm
```

Je kan JabRef ook installeren vanuit de Fedora package repository, maar dan krijg je een verouderde versie. Je kan dan beter de "Platform Independent Runnable Jar" downloaden via de projectwebsite[7]. Die kan je dan opstarten vanuit de shell met het commando (hier voorbeeld voor versie 4.1):

```
java -jar JabRef-4.1.jar
```

1.3 Configuratie

1.3.1 Git, GitHub

Wellicht heb je Git al geconfigureerd voor enkele van je andere vakken. Kijk eventueel alles nog eens na! Als alles ok is, kan je deze sectie overslaan.

Wij raden aan om Git via de command line te gebruiken. Zo krijg je het beste inzicht in de werking. Het commando `git status` geeft op elk moment een goed overzicht van de toestand van je lokale repository en geeft aan met welk commando je een stap verder kan zetten of de laatste stap ongedaan kan maken.

Als je nog geen GitHub-account hebt, kies dan een gebruikersnaam die je na je afstuderen nog kan gebruiken (dus bv. niet je HoGent login). De kans is erg groot dat je tijdens je carrière nog van GitHub gebruik zult maken. Koppel ook je HoGent-emailadres aan je GitHub account (je kan meerdere adressen registreren). Op die manier kan je aanspraak maken op het GitHub Student Developer Pack[8], wat je gratis toegang geeft tot een aantal in principe betalende producten en diensten.

Windows-gebruikers voeren volgende instructies uit via Git Bash, macOS- en Linux-gebruikers via de standaard (Bash) terminal.

```
git config --global user.name 'Pieter Stevens'
git config --global user.email 'pieter.stevens.u12345@student.hogent.be'
git config --global push.default simple
```

Maak ook een SSH-sleutel aan om het synchroniseren met GitHub te vereenvoudigen (je moet dan geen wachtwoord meer opgeven bij push/pull van/naar een private repository).

```
ssh-keygen
```

Volg de instructies op de command-line, druk gewoon ENTER als je gevraagd wordt een wachtwoordzin (pass phrase) in te vullen. In de home-directory van je gebruiker (bv. `c:\Users\Pieter`

[7] https://jabref.org/
[8] https://education.github.com/pack

op Windows, `/Users/pieter` op Mac, `/home/pieter` op Linux) is nu een directory met de naam .ssh/ aangemaakt met twee bestanden: `id_rsa` (je private key) en `id_rsa.pub` (je public key). Open dit laatste bestand met een teksteditor en kopieer de volledige inhoud naar het klembord. Ga vervolgens naar je GitHub profiel en kies in het menu links voor SSH and GPG keys. Klik rechtsboven op de groene knop met "New SSH Key" en plak de inhoud van je publieke sleutel in het veld "Key". Bevestig je keuze.

Test nu of je de code van de cursus Onderzoekstechnieken kan downloaden. Ga in de Bash shell naar een directory waar je dit project lokaal wil bijhouden en voer uit:

```
git clone git@github.com:HoGentTIN/onderzoekstechnieken-cursus.git
```

Als dit lukt, is er nu een directory aangemaakt met dezelfde naam als de repository. Doe tijdens het semester regelmatig `git pull` om de laatste wijzigingen in het cursusmateriaal bij te werken. Pas zelf geen bestanden aan binnen deze repository, dit zal leiden tot conflicten.

1.3.2 TeXstudio

Controleer deze instellingen via menu-item *Options* > *Configure TeXstudio*:

- Build
  - Default Compiler: pdflatex
  - Default Bibliography tool: biber
- Editor:
  - Indentation mode: Indent and Unindent Automatically
  - Replace Indentation Tab by Spaces: Aanvinken
  - Replace Tab in Text by spaces: Aanvinken
  - Replace Double Quotes: English Quotes: ‘‘’’

Om te testen of TeXstudio goed werkt, kan je het bestand `cursus/cursus-onderzoekstechnieken.tex` openen. Kies *Tools* > *Build & View* (of druk F5) om de cursus te compileren in een PDF-bestand.

Veel functionaliteiten van LaTeX zitten in aparte packages die niet noodzakelijk standaard geïnstalleerd zijn. De eerste keer dat je een bestand compileert, is het dan ook mogelijk dat er extra packages moeten gedownload worden. MiKTeX zal een pop-up tonen om je toestemming te vragen, bevestig dit. Op Linux is het mogelijk dat je deze packages nog manueel moet installeren. De eerste keer compileren kan enkele minuten duren zonder dat je feedback krijgt over wat er gebeurt. Even geduld, dus!

Indien er zich fouten voordoen bij de compilatie, kan je onderaan in het tabblad Log een overzicht krijgen van de foutboodschappen.

1.3.3 JabRef

JabRef[9] is een GUI voor het bewerken van BibTeX-bestanden, een soort database van bronnen uit de wetenschappelijke- of vakliteratuur voor een LaTeX-document.

[9] http://www.jabref.org/

Kies in het menu voor *File* > *Switch to BibLaTeX* mode. Dit maakt de bestandsindeling van de bibliografische databank compatibel met dat van de cursus en het aangeboden LaTeX-sjabloon voor de bachelorproef.

Kies in het *Preferences*-venster voor de categorie *File* en geef een directory op voor het bijhouden van PDFs van de gevonden bronnen onder *Main file directory*. Het is heel interessant om alle gevonden artikels te downloaden en onder die directory bij te houden. Nog beter is om als naam van het bestand de BibTeX key te nemen (typisch naam van de eerste auteur + jaartal, bv. `Knuth1998.pdf`). Je kan het bestand dan makkelijk openen vanuit JabRef.

Voor meer gedetailleerde informatie over het bijhouden van bibliografische referenties, zie de bachelorproefgids (**VanVreckem2017**).

1.4 Gebruik van R

R is een softwareprogramma voor datamanipulatie, berekening en het grafisch voorstellen van data. Het heeft onder meer:

1. een effectieve gegevensbeheer- en opslagfaciliteit,
2. een reeks operatoren voor berekeningen op arrays, in het bijzonder matrices,
3. een grote verzameling van instrumenten voor data-analyse,
4. grafische faciliteiten voor data-analyse en weergave en
5. een goed ontwikkelde, eenvoudige en effectieve programmeertaal (genaamd 'S').

R heeft een ingebouwde hulpfaciliteit die vergelijkbaar is met die van UNIX man-pages. Voor meer informatie over elke specifieke functie, bijvoorbeeld `solve`, kan je volgende commando oproepen

```
> help (solve)
```

Een alternatief is

```
> ?solve
```

1.4.1 Commando's opslaan en output uitvoeren

Als de commando's in een extern bestand worden opgeslagen, bv. `commands.R` in de werkmap, dan kunnen deze op elk moment uitgevoerd worden in een R-sessie met de opdracht

```
> source ("commands.R")
```

De functie `sink`,

```
> sink ("record.lis")
```

Zal alle volgende uitvoer van de console naar een extern bestand, `record.lis`, wegschrijven. Het bevel

```
> sink()
```

Herstelt de output opnieuw naar de console.

1.4.2 R omgeving en workspace

De entiteiten die R creëert en manipuleert staan bekend als objecten. Deze kunnen variabelen zijn, arrays van cijfers, reeksen, functies of meer algemene structuren die uit dergelijke componenten zijn gebouwd. Tijdens een R-sessie worden objecten gemaakt en opgeslagen op naam. Het R commando

```
> objects()
```

geeft een overzicht van alle objecten die gemaakt zijn tot op dat moment. De verzameling van objecten die momenteel zijn opgeslagen, heet de werkruimte. Om objecten te verwijderen is de functie `rm` beschikbaar:

```
> rm (x, y, z, inkt, junk, temp, foo, bar)
```

Alle objecten die tijdens een R-sessie zijn aangemaakt, kunnen permanent in een bestand worden opgeslagen voor gebruik in de toekomstige R sessies. Als u aangeeft dat u dit wilt doen, worden de objecten geschreven naar een bestand met extensie `.RData`

In dit hoofdstuk onderzoeken we hoe je een dataset definieert in R. Er worden slechts twee commando's onderzocht. De eerste is voor het eenvoudig toewijzen van gegevens, en de tweede is voor het lezen in een databestand. Er zijn verschillende manieren om gegevens in een R-sessie te lezen, maar we richten ons op slechts twee om het eenvoudig te houden.

1.4.3 Toewijzing

De meest directe manier om een lijst met nummers op te slaan is via een opdracht met behulp van het `c`-commando. (C staat voor "combineren.") Het idee is dat een lijst met nummers onder een bepaalde naam wordt opgeslagen, en de naam wordt gebruikt om te verwijzen naar de gegevens. Een lijst wordt gespecificeerd met de opdracht `c`, en de toewijzing wordt geduid met de symbolen "<-". Een andere term die gebruikt wordt om de lijst met nummers te omschrijven is `vector`.

De cijfers binnen de `c`-opdracht worden gescheiden door komma's. Als voorbeeld kunnen we een nieuwe variabele maken, genaamd "`x`".

```
> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
```

Wanneer je dit commando invoert, mag je geen uitvoer zien behalve een nieuwe opdrachtregel. Het commando maakt een lijst met nummers genaamd "x". Om te zien welke elementen zijn opgenomen in x, typ zijn naam en druk op de enter-toets.

Als u met één van de nummers wilt werken, kunt u hier toegang krijgen tot de variabele en vervolgens vierkante haakjes noteren die aangeven welk nummer u wilt beschouwen:

```
> x[2]
[1] 5.6
```

1.4.4 Een csv file lezen

We gaan ervan uit dat het gegevensbestand een csv bestand is: "komma-gescheiden waarden"(csv). Dat wil zeggen, elke regel bevat een rij met waarden die getallen of letters kunnen zijn, en elke waarde wordt gescheiden door een komma. We gaan ervan uit dat de eerste rij een lijst met labels bevat. Het idee is dat de labels in de bovenste rij gebruikt worden om te verwijzen naar de verschillende variabelen per rij.

Het commando om het gegevensbestand te lezen is `read.csv`. We moeten tenminste één argument geven aan de opdracht.

> **Oefening 1.1.** *Ga met het help commando na wat de parameters zijn van het commando. Probeer daarna het bestand* `computers.csv` *in te lezen.* ■

Als u niet zeker bent welke bestanden in de huidige werkmap zitten, kunt u het commando `dir` gebruiken om een lijst van bestanden weer te geven. Het `getwd` commando gebruikt u om de huidige werkmap te bepalen.

Het databestand komt uit de publicatie van (**Stengos2005**). Deze dataset bevat data van 1993 tot 1995 over de prijzen van computers. Je kan nagaan wat het effect van de toevoeging van cd-rom-station is op de prijs van de computer of het effect van de kloksnelheid op de prijs.

```
> dir()
[1] "breakingbad.csv"  "Desktop"            "Documents"          "
    Downloads"         "dumps"              "earch php-"         "
    examples.desktop"
[8] "f.r"              "kids.csv"           "kmissles.csv"       "
    kmissles.ods"      "Music"              "out.pdf"            "
    Pictures"
[15] "public"           "Public"             "R"                  "
    Templates"         "test"               "test.php"           "
    Videos"
> getwd()
[1] "/home/eothein"
```

Als u niet zeker weet welke kolommen gedefinieerd zijn, kunt u `names()` gebruiken:

```
> names(computers)
[1] "price"   "speed"   "hd"      "ram"     "screen"  "cd"
    "multi"   "premium" "ads"     "trend"
```

Wanneer u het commando `read.csv` gebruikt, gebruikt R een specifiek soort variabele, dat een dataframe heet. Alle gegevens worden opgeslagen in het dataframe als afzonderlijke kolommen. Als u niet zeker weet wat voor variabele u hebt, dan kunt u de opdracht `attributes` gebruiken. Hiermee worden alle dingen vermeld die R gebruikt om de variabele te beschrijven:

```
attributes(computers)
$names
[1] "price"   "speed"   "hd"      "ram"     "screen"  "cd"
    "multi"   "premium" "ads"     "trend"
```

```
4
5  $class
6  [1] "tbl_df"     "tbl"        "data.frame"
7
8  $row.names
9  [1]   1   2   3   4   5   6   7   8   9  10  11  12
         13  14  15  16  17  18  19  20  21  22  23  24
         25  26  27
10 [28]  28  29  30  31  32  33  34  35  36  37  38  39
         40  41  42  43  44  45  46  47  48  49  50  51
         52  53  54
11 ...
12 [ reached getOption("max.print") -- omitted 5259 entries ]
13
14 $spec
15 cols(
16 price = col_integer(),
17 speed = col_integer(),
18 hd = col_integer(),
19 ram = col_integer(),
20 screen = col_integer(),
21 cd = col_character(),
22 multi = col_character(),
23 premium = col_character(),
24 ads = col_integer(),
25 trend = col_integer()
26 )
```

1.4.5 Data types

We kijken naar enkele manieren waarop R gegevens kan opslaan en organiseren. Dit is echter een inleiding dus beschouwen we maar een kleine subset van de verschillende datatypes die door R worden herkend.

Numbers

De meest eenvoudige manier om een nummer op te slaan is om een variabele van een enkel getal te nemen:

```
1 > a <- 3
2 >
```

Hiermee kunt u allerlei basisoperaties doen en opslaan:

```
1 > b <- sqrt(a*a+3)
2 > b
3 [1] 3.464102
```

Als u een lijst met nummers wilt initialiseren, kan het `numeric` commando worden gebruikt. Om bijvoorbeeld een lijst van 10 nummers te maken, gebruikt u de volgende opdracht. Je kan ook kijken naar het type van de variabele.

```
> a <- numeric(10)
> a
[1] 0 0 0 0 0 0 0 0 0 0
> typeof(a)
[1] "double"
```

Strings

Een tekenreeks wordt gespecificeerd door gebruik te maken van aanhalingstekens. Zowel enkelvoudige als dubbele aanhalingstekens zullen werken:

```
> a <- "hello"
> a
[1] "hello"
> b <- c("hello","there")
> b
[1] "hello" "there"
> b[1]
[1] "hello"
```

Factors

Vaak bevat een experiment proeven voor verschillende niveaus van een verklarende variabele. Bijvoorbeeld een nominale variabele die gecodeerd wordt met een integer. De verschillende niveaus worden ook factoren genoemd.

Je geeft aan dat een variabele een factor is met behulp van het `factor` commando.

Data frames

Data kan worden opgeslagen aan de hand van dataframes. Dit is een manier om verschillende vectoren van verschillende types te nemen en ze op te slaan in dezelfde variabele. De vectoren kunnen van alle soorten zijn. Een dataframe kan bijvoorbeeld verschillende vectoren bevatten en elke lijst kan een vector zijn van factoren, strings of nummers.

Er zijn verschillende manieren om gegevensframes te maken en te manipuleren. De meeste zijn buiten het bereik van deze introductie. Ze worden hier alleen genoemd om een meer volledige beschrijving te geven.

```
> a <- c(1,2,3,4)
> b <- c(2,4,6,8)
> levels <- factor(c("A","B","A","B"))
> bubba <- data.frame(first=a,
                      second=b,
                      f=levels)
```

```
> bubba
first second f
1     1      2 A
2     2      4 B
3     3      6 A
4     4      8 B
> summary(bubba)
     first           second     f
Min.   :1.00   Min.   :2.0   A:2
1st Qu.:1.75   1st Qu.:3.5   B:2
Median :2.50   Median :5.0
Mean   :2.50   Mean   :5.0
3rd Qu.:3.25   3rd Qu.:6.5
Max.   :4.00   Max.   :8.0
> bubba$first
[1] 1 2 3 4
> bubba$second
[1] 2 4 6 8
> bubba$f
[1] A B A B
Levels: A B
```

Logische variabelen

Een ander belangrijk gegevenstype is het logische type. Er zijn twee vooraf gedefinieerde variabelen, `TRUE` en `FALSE`.

Tables

Een andere manier om informatie op te slaan is in een tabel. We kijken alleen maar naar het maken en definiëren van tabellen.

```
> a <- factor(c("A","A","B","A","B","B","C","A","C"))
> results <- table(a)
> results
a
A B C
4 3 2
> attributes(results)
$dim
[1] 3

$dimnames
$dimnames$a
[1] "A" "B" "C"


$class
[1] "table"
```

```
> summary(results)
Number of cases in table: 9
Number of factors: 1
```

Als je rijen wilt toevoegen aan uw tabel, voeg dan nog een vector toe als argument van de tabelopdracht. In het onderstaande voorbeeld hebben wij twee vragen. In de eerste vraag staan de reacties 'Nooit', 'Soms' of 'Altijd'. In de tweede vraag staan de reacties 'Ja', 'No' of 'Maybe'. De set van vectoren 'a', en 'b' bevatten het antwoord voor elke meting. Het derde punt in 'a' is hoe de derde persoon op de eerste vraag reageerde en het derde punt in 'b' is hoe de derde persoon op de tweede vraag reageerde.

```
> a <- c("Sometimes","Sometimes","Never","Always","Always","
    Sometimes","Sometimes","Never")
> b <- c("Maybe","Maybe","Yes","Maybe","Maybe","No","Yes","No")
> results <- table(a,b)
> results
           b
a           Maybe No Yes
  Always        2  0   0
  Never         0  1   1
  Sometimes     2  1   1
```

Matrix

Een matrix is een verzameling van gegevens die zijn aangebracht in een tweedimensionale rechthoekige indeling. Een voorbeeld van een matrix is bijvoorbeeld als volgt:

$$\begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix}$$

```
> A = matrix(
+   c(2, 4, 3, 1, 5, 7), # the data elementen
+   nrow=2,              # aantal rijen
+   ncol=3,              # aantal kolommen
+   byrow = TRUE)        # vul de matrix aan per rij

> A                      # print de matrix
     [,1] [,2] [,3]
[1,]    2    4    3
[2,]    1    5    7

> A[2, 3]       # element op 2de rij , 3de kolom
[1] 7

> A[2, ]        # de 2de rij
[1] 1 5 7
```

```
> A[ ,c(1,3)]  # de eerste en de derde kolom
     [,1] [,2]
[1,]    2    3
[2,]    1    7
```

1.5 Oefeningen

Oefening 1.2. *Bekijk de dataset mtcars. Geef de waarde terug voor de eerste rij, tweede kolom. Geef ook het aantal rijen, het aantal kolommen. Geef ook een preview van het volledige data frame. Geef enkel de kolom terug met de definities van de cylinders. Om een data frame te bekomen met de twee kolommen mpg en hp, pakken we de kolomnamen in een indexvector in met single square bracket operator. Probeer ook eens op te zoeken hoe je een rijrecord van de ingebouwde data set mtcars bepaalt.* ■

Oefening 1.3. *Maak zelf een willekeurige datafile aan in Excel en probeer deze in te lezen in R. Zijn er nog dataformaten die ondersteund worden door R?* ■

Oefening 1.4. *Genereer een 4x5 array en noem die x. Geneer daarna een 3x2 array waar de eerste kolom de rij-index kan zijn van x en de tweede kolom een kolomindex voor x. Vervang de elementen gedefinieerd door de index in i in x door 0.* ■

Oefening 1.5. *Genereer een vector waar een voornaam en een achternaam in komen. Benoem ook de naam van de kolommen. Geef daarna ook voornaam terug van het eerste element van de array.* ■

Oefening 1.6. *Probeer voor de datafile `rainforest` in de library `DAAG` te tellen hoeveel rijen er zijn per species die volledig en compleet zijn (dus geen n.a. bevatten). Je kan hiervoor `with, table, complete.cases` voor gebruiken.* ■

Oefening 1.7. *Genereer een vector met de waarden $e^x cos(x)$ voor $x = 3, 3.1, 3.2, \ldots, 6$* ■

Oefening 1.8. *Bereken: $\sum_i^{100}(i^3 + 4i^2)$* ■

Appendices

A. Notatie

Notatie	**Betekenis**
$X = \{x_1, x_2, \ldots, x_n\}$	Een stochastische variabele X met n waarnemingen x_i (voor $i : 1 \ldots n$)
N	De populatieomvang
n	De steekproefgrootte
μ (mu)	Het gemiddelde (ook: verwachtingswaarde) over heel de *populatie*.
$\overline{x}$	Het gemiddelde over de *steekproef*
σ (sigma)	De standaardafwijking over heel de populatie
σ^2 (sigma)	De variantie over heel de populatie
s	De standaardafwijking van de steekproef
s^2	De variantie van de steekproef
$X \sim Nor(\mu, \sigma)$	De variabele X is *normaal verdeeld* met gemiddelde μ en standaardafwijking σ
$Z \sim Nor(0,1)$	Z is een variabele met een kansverdeling die de *standaardnormaalverdeling* volgt, dus met gemiddelde 0 en standaardafwijking 1
$M \sim Nor(\mu_{\overline{x}}, \sigma_{\overline{x}})$	De kansverdeling van het steekproefgemiddelde (cfr. de centrale limietstelling, Sectie **??**)
$\mu_{\overline{x}}$	De verwachtingswaarde bij de kansverdeling van het steekproefgemiddelde
$\sigma_{\overline{x}}$	De standaardafwijking bij de kansverdeling van het steekproefgemiddelde
α	Een significantieniveau (voor een statistische toets)
$1 - \alpha$	Een betrouwbaarheidsniveau (voor een betrouwbaarheidsinterval)