

VAN VRAGENLIJST TOT TABEL

VAN VRAGENLIJST TOT TABEL

In de sociale wetenschappen worden in de opleiding de grondbeginselen van de statistiek behandeld, maar er wordt geen aandacht besteed aan zaken als datastructuren en dataopbouw. Dit is echter voorwaardelijke kennis die noodzakelijk is voor de onderzoeksuitvoering. Dit rapport is dan ook bedoeld voor personen die in hun opleiding (of daarbuiten) met SPSS hebben leren werken, in aanraking zijn gekomen met beschrijvende en toetsende statistiek, maar die nu willen starten met het daadwerkelijk doen van kwantitatief onderzoek.

Het doel van het doen van onderzoek is het beantwoorden van een onderzoeksvraag. Het beantwoorden van een vraag vergt veelal het uitvoeren van berekeningen. In sociaal wetenschappelijk onderzoek is het meest gebruikte pakket SPSS. Onafhankelijk van welk pakket gekozen wordt, geldt dat er eerst moeten worden gezorgd voor een goed opgebouwd databestand. Voordat er met het analyseren van de data begonnen kan worden, dient er een aantal stappen worden gezet. Deze stappen worden beschreven in deel 1 van de rapportage. Het gaat dan om de datastructuur, hoe het databestand dient te worden opgebouwd, verschillende wijzen waarop data binnen een onderzoek kunnen worden verzameld en hoe deze data in SPSS kunnen worden gezet met behulp van stukken syntax. In deel 2 wordt ingegaan op hoe de gegevens in SPSS kunnen worden geanalyseerd. Eveneens wordt besproken hoe de output van de analyses eenvoudig in een tabel voor een onderzoeksrapport of artikel kan worden gezet.

Bestandsopbouw, datastructuur en
het automatisch construeren van tabellen

ISBN 978-90-71909-17-7



9 789071 909177 >

H.A.M. Everaert | A.A.M. Houtveen | S.K. Brokamp | M. Willemsen

H.A.M. Everaert
A.A.M. Houtveen
S.K. Brokamp
M. Willemsen

VAN VRAGENLIJST TOT TABEL

**Bestandsopbouw, datastructuur en het automatisch
construeren van tabellen**

**H.A.M. Everaert
A.A.M. Houtveen
S.K. Brokamp
M. Willemsen**

Hogeschool Utrecht
Faculteit Educatie
Kenniscentrum Educatie

Telefoon: 088-481 7747
E-mail: lectoraatlezen@hu.nl

Padualaan 97
Postbus 14007
3508 SB Utrecht

Kenniscentrum Educatie rapportnummer 15-01

Titel: Van vragenlijst tot tabel
Subtitel: Bestandsopbouw, datastructuur en het automatisch construeren
van tabellen

ISBN: 978-90-71909-17-7

NUR-code: 100
NUR-omschrijving: Educatieve uitgaven algemeen

Uitgever: Hogeschool Utrecht, Kenniscentrum Educatie van de
Faculteit Educatie
Drukkerij: Tuijtel

© 2015 Hogeschool Utrecht
Faculteit Educatie
Kenniscentrum Educatie
University of Applied Sciences Utrecht
Faculty of Education
Research Centre for Education

Auteur: Everaert, H.A.M.
Co-auteurs: Houtveen, A.A.M., Brokamp, S.K., Willemsen, M.

“Niets uit deze uitgave mag worden veeleelvoudigd en/of openbaar gemaakt worden door middel van druk, fotokopie, microfilm of op welke andere wijze dan ook zonder voorafgaande schriftelijke toestemming van het Kenniscentrum Educatie van de Faculteit Educatie.”

“No part of this book may be reproduced in any form, by print, photoprint, microfilm of any other means without written permission of the Department.”

INHOUDSOPGAVE

Inleiding	1
<u>Deel 1</u>	
Hoofdstuk 1	
DATASTRUCTUUR	3
1.1 Inleiding	3
1.2 Basisbestanden en hoofdbestanden	3
1.2.1 Basisbestanden	4
1.2.2 Hoofdbestanden	6
1.3 Opbouw datastructuur	8
1.4 Opslaan van data in de hoofdbestanden: lange en brede bestanden	10
1.5 Datum, meetmoment en koppelmeetmoment	11
1.5.1 Meetmoment	12
1.5.2 Koppelmeetmoment	13
1.6 Samenvatting	15
Hoofdstuk 2	
BESTANDSOPBOUW	17
2.1 Inleiding	17
2.2 SPSS VARIABLE VIEW	17
2.3 Variabelen basisbestanden	19
2.4 Variabelen hoofdbestanden	22
2.5 Samenvatting	24
Hoofdstuk 3	
DATA VERZAMELEN EN INVOEREN IN SPSS	25
3.1 Inleiding	25
3.2 Data verzamelen	25
3.2.1 Data op papier	25
3.2.2 Data in Excel	26
3.2.3 Data in leerling- of studentvolgsysteem	26
3.2.4 Data in digitaal survey systeem	27

3.3	Visuele inspectie van binnengekomen bestanden	28
3.4	Bestanden in SPSS invoeren	28
3.4.1	Data op papier	28
3.4.2	Data in Excel	29
3.4.3	Data in leerlingvolgsysteem – Parnassys	31
3.4.4	Data in leerlingvolgsysteem – ESIS	35
3.4.5	Data in leerlingvolgsysteem – Dotcom en CITO LOVS	35
3.4.6	Data in digitaal surveysysteem	36
3.4.7	Data in ORISIS	36
3.5	Databestanden samenvoegen	37
3.6	Samenvatting	38

Deel 2

Hoofdstuk 4

DATA TOEVOEGEN AAN HOOFD- EN BASISBESTANDEN 39

4.1	Inleiding	39
4.2	Datacleaning	39
4.2.1	Datacleaning – Data op papier, in Excel en in Leerlingvolgsystemen	39
4.2.2	Datacleaning – Digitaal survey systeem	40
4.3	Databestanden toevoegen aan hoofdbestand	42
4.4	Opslaan van databestanden	43
4.5	Samenvatting	44

Hoofdstuk 5

BEREKENINGEN MET BEHULP VAN SPSS 45

5.1	Inleiding	45
5.2	Typen data-analyses	45
5.3	Samenvoegen van benodigde gegevens voor analyses	46
5.4	Hercoderen van variabelen	50
5.5	Data-analyse	51
5.6	Samenvatting	55

Hoofdstuk 6	
RESULTATEN RAPPORTEREN IN TABELLEN	57
6.1 Inleiding	57
6.2 Resultaten van analyses wegschrijven in een nieuw databestand	57
6.3 Opschonen van gegevens voor rapportage	59
6.4 Vullen van de tabellen voor rapportage	65
6.5 Samenvatting	67
BIJLAGEN	68
Bijlage 1 Boeken over statistiek en SPSS	68

Inleiding

Het idee voor de voorliggende rapportage ontstond vanuit de constatering dat in de sociale wetenschappen op de opleiding geleerd wordt met SPSS te werken. Ook worden de grondbeginselen van de statistiek behandeld, maar er wordt geen aandacht besteed aan zaken als datastructuren en dataopbouw. Dit is echter voorwaardelijke kennis die noodzakelijk is voor de onderzoeksuitvoering, zeker wanneer er sprake is van grootschalig onderzoek. De voorliggende rapportage is dan ook bedoeld voor personen die in hun opleiding (of daarbuiten) met SPSS hebben leren werken, in aanraking zijn gekomen met beschrijvende en toetsende statistiek, maar die nu willen starten met het daadwerkelijk doen van kwantitatief onderzoek.

Deze rapportage beoogt uitdrukkelijk niet een SPSS handleiding te zijn. In bijlage 1 is daarom een lijst opgenomen met een aantal SPSS handboeken. Eveneens is deze rapportage niet bedoeld als statistiekboek. Voor een aantal voorbeelden van statistiekboeken verwijzen we eveneens naar bijlage 1.

De rapportage bestaat uit twee delen. Deel 1 richt zich op datastructuren en dataopbouw. Het doel van het doen van onderzoek is het beantwoorden van een onderzoeksvraag. Het beantwoorden van een vraag vergt veelal het uitvoeren van berekeningen. In sociaal wetenschappelijk onderzoek is het meest gebruikte pakket SPSS. Dit is een zeer omvangrijk pakket waarmee zowel beschrijvende als toetsende analyses uitgevoerd kunnen worden. Wanneer het beantwoorden van de vraag alleen eenvoudige beschrijvende analyses vraagt, wordt soms ook Excel benut. Onafhankelijk van welk pakket gekozen wordt, geldt dat er eerst moet worden gezorgd voor een goed opgebouwd databestand. Voordat met het analyseren van de data begonnen kan worden, dient een aantal stappen worden gezet. Deze stappen worden beschreven in deel 1 van deze rapportage. In hoofdstuk 1 wordt ingegaan op de datastructuur. In hoofdstuk 2 wordt vervolgens beschreven hoe verschillende databestanden worden opgebouwd. Hoofdstuk 3 geeft een overzicht van de diverse wijzen waarop data kunnen worden verzameld, kunnen worden opgevraagd en hoe deze data vervolgens in SPSS kunnen worden gezet met behulp van stukken syntax. Wij noemen dit het inlezen van de data.

Zodra de data zijn ingelezen, zijn de voorbereidingen afgerond en kan gestart worden met de data-analyse. Wij merken hierbij op dat de werkwijzen zoals wij deze voorstellen, mogelijk anders vormgegeven kunnen worden, dan wel nog verder kunnen worden geautomatiseerd. Met elke nieuwe versie van SPSS kunnen bepaalde commando's korter en efficiënter worden opgeschreven. Wij kiezen er voor dit alleen te doen wanneer bepaalde werkwijzen of syntaxen niet meer werken.

In deel 2 van deze rapportage komt aan de orde hoe de ingelezen gegevens in SPSS kunnen worden geanalyseerd. De eerste stap die uitgevoerd wordt binnen SPSS is het controleren van de data op fouten. Dit wordt datacleaning genoemd. In hoofdstuk 4 worden enkele basale syntaxen die gebruikt kunnen worden om de data te cleanen in SPSS, beschreven en toegelicht. In hoofdstuk 5 wordt beschreven hoe basale beschrijvende analyses, met behulp van syntaxen, kunnen worden uitgevoerd in SPSS. Hierbij worden steeds de bijbehorende syntaxen beschreven en van een toelichting voorzien. In hoofdstuk 6 wordt ingegaan op hoe de output van de analyses eenvoudig in een tabel voor een onderzoeksrapport of artikel kan worden gezet. Dit maakt het zelf overtypen van gegevens overbodig. Dat scheelt niet alleen tijd, maar voorkomt ook fouten.

DEEL 1

Hoofdstuk 1

DATASTRUCTUUR

1.1 Inleiding

In de inleiding is aangegeven dat er eerst een aantal stappen moet worden gezet alvorens data in SPSS kunnen worden ingelezen en de gewenste data-analyses daadwerkelijk kunnen worden uitgevoerd. In dit hoofdstuk wordt uitgelegd waarom dit nodig is en wat er mis kan gaan wanneer dit niet gebeurt. De eerste stap bestaat uit het vastleggen van de datastructuur en wordt beschreven in paragraaf 1.1. In paragraaf 1.2 wordt ingegaan op het verschil tussen basisbestanden en hoofdbestanden, welke gegevens deze bestanden moeten bevatten en hoe deze bestanden op een later tijdstip aan elkaar te koppelen zijn door het opnemen van een uniek nummer per respondent (*identifier*). In paragraaf 1.3 wordt het opbouwen van de datastructuur toegelicht. Het verschil tussen brede en lange bestanden wordt uitgelegd in paragraaf 1.4. Tot slot zal in paragraaf 1.5 worden uitgelegd hoe meetmomenten van elkaar kunnen worden onderscheiden in de bestanden.

1.2 Basisbestanden en hoofdbestanden

Wanneer het uit te voeren onderzoek slechts een eenmalige dataverzameling met één dataverzamelingsinstrument betreft (denk bijvoorbeeld aan het eenmalig afnemen van een vragenlijst bij een groep leerkrachten) kan zowel de informatie over de respondenten (naam, geslacht, leeftijd en/of ervaringsjaren) als de respons op het dataverzamelingsinstrument in eenzelfde databestand worden opgeslagen.

Wanneer het onderzoek omvangrijker is en vraagt om dataverzameling op meerdere meetmomenten en misschien zelfs met meerdere dataverzamelingsinstrumenten, is het sterk aan te raden te werken met twee typen databestanden die op latere tijdstippen aan elkaar kunnen worden gekoppeld: basisbestanden en hoofdbestanden.

Het belang van het scheiden en in aparte typen bestanden opslaan van statische en dynamische gegevens is het minimaliseren van de kans op fouten doordat in principe elk gegeven maar één keer in een databestand hoeft te worden opgenomen.

Basisbestanden zijn bestanden waarin gegevens worden vermeld die niet veranderen, zoals de eerder genoemde respondentkenmerken, oftewel de *statische* variabelen. Deze gegevens worden slechts éénmalig verzameld en opgeslagen.

In de hoofdbestanden worden de zogeheten *dynamische* variabelen opgenomen; dit zijn variabelen die gegevens kunnen bevatten welke op verschillende meetmomenten, bij dezelfde respondenten, zijn verzameld. Denk hierbij bijvoorbeeld aan vragenlijstgegevens die op meerdere meetmomenten zijn verzameld en waarvan de uitkomsten dus steeds weer anders kunnen zijn. Per dataverzamelingsinstrument (bijvoorbeeld een vragenlijst voor de leerkracht) wordt slechts één hoofdbestand aangemaakt.

Door in beide typen bestanden dezelfde koppelvariabelen op te nemen kunnen basisbestanden en hoofdbestanden (of delen ervan) worden gekoppeld op het moment dat dit nodig is (zoals wanneer analyses moeten worden uitgevoerd). Als koppelvariabelen fungeren unieke *identifiers*, ID's, van de respondenten, al dan niet in combinatie met de datum waarop toetsen of vragenlijsten zijn afgenomen. In onderstaande paragrafen worden de kenmerken van beide typen bestanden besproken.

1.2.1 Basisbestanden

Basisbestanden zijn 'losse' SPSS-bestanden waarin gegevens worden vermeld die niet veranderen (statische variabelen): de kenmerken van de deelnemers (respondenten) aan het onderzoek. In het onderzoek kunnen er respondenten zijn op meerdere niveaus. Elk respondent-niveau krijgt een eigen basisbestand.

Zo kan een basisbestand voor leerlingen (of studenten) worden aangemaakt, maar ook een basisbestand voor leraren, een basisbestand voor scholen, een basisbestand voor schoolbesturen en in een internationaal vergelijkende studie zelfs een basisbestand voor landen. Het aantal basisbestanden dat in het onderzoek nodig is, hangt af van de respondent-niveaus die worden bevraagd.

Om later bij de berekeningen de basisbestanden aan de hoofdbestanden te kunnen koppelen (hier wordt in hoofdstuk 5 op teruggekomen), wordt aan alle respondenten in de basisbestanden een uniek nummer, een ID (*identifier*), toegekend. Het ID is in principe een volgnummer waarbij de eerste respondent het ID 1 krijgt, de tweede respondent het ID 2, enz. Met uniek bedoelen we dat er geen respondenten binnen eenzelfde niveau zijn die hetzelfde nummer krijgen. Ook komt elke respondent in het basisbestand slechts één keer voor. Om de basisbestanden aan de hoofdbestanden te kunnen koppelen is het overigens ook noodzakelijk dat het ID niet wordt gewijzigd in de loop van het onderzoek. De basisbestanden worden alleen gewijzigd als er nieuwe respondenten of kenmerken van respondenten moeten worden toegevoegd.

Het opbouwen van de bestanden en ook het toekennen van de ID's dient met de grootste zorg te gebeuren. Een fout in de databestanden of de ID's heeft verstrekende gevolgen en kan van invloed zijn op het gehele onderzoek en op de analyses. Als gevolg van verkeerde of foutieve ID's worden, bijvoorbeeld, gegevens van leerlingen aan verkeerde leerkrachten gekoppeld of komen toetsresultaten bij 'andere' leerlingen terecht. Het gevolg is dat analyses ten behoeve van het beantwoorden van de onderzoeksvraag verkeerde uitkomsten opleveren.

Om een en ander te verduidelijken worden in schema 1 drie voorbeelden gegeven van basisbestanden op het respondentniveau van de leerlingen (met ID 'LRLGID'), leerkrachten (met ID 'LKRID') en scholen (met ID 'SCHOOLID'). In het basisbestand leerlingen worden alle leerlingen opgenomen met daarbij behorende kenmerken. Alle leerlingen komen slechts één keer voor in het 'basisbestand leerlingen'. Hetzelfde geldt voor de leerkrachten in het 'basisbestand leerkrachten' en de aan het onderzoek deelnemende scholen (in het 'basisbestand scholen').

Schema 1 Voorbeeld van drie basisbestanden		
<i>Basisbestand leerlingen</i>	<i>Basisbestand leerkrachten</i>	<i>Basisbestand scholen</i>
LRLGID	LKRID	SCHOOLID
SCHOOLID	SCHOOLID	BRINN
Naamleerling	LKR_Naam	Schoolnaam
Geboortedatum	LKR_sekse	Adres
LRLG_sekse	LKR_Geboortedatum	Postcode
Leerlinggewicht	Woonplaats
....		Telefoonnr
		Emailadres1
		Emailadres2
	

1.2.2 Hoofdbestanden

Het tweede type bestanden zijn de hoofdbestanden. In deze bestanden zijn de variabelen opgenomen waarvan de gegevens per meetmoment steeds weer anders zijn; de dynamische variabelen. Een hoofdbestand moet aangemaakt worden voor ieder dataverzamelingsinstrument (toets/vragenlijst/observatielijst) waarmee in het onderzoek gegevens worden verzameld. **Alle** data die met het betreffende dataverzamelingsinstrument worden verzameld, worden in ditzelfde hoofdbestand opgenomen. Als er dus op meerdere meetmomenten met hetzelfde dataverzamelingsinstrument gegevens worden verzameld, worden deze gegevens allemaal in hetzelfde hoofdbestand opgenomen, en wel onder elkaar. In dat geval wordt een variabele opgenomen om de verschillende meetmomenten van elkaar te kunnen onderscheiden. Hier komen we in paragraaf 1.5 op terug. In langdurige onderzoeksprojecten met verschillende meetmomenten bevatten de hoofdbestanden dan ook duizenden en duizenden records.

Ter verduidelijking worden in schema 2 voorbeelden weergegeven van een aantal hoofdbestanden; van een toets (de hardopleestoets AVI) die bij leerlingen in een bestaand onderzoek wordt afgenomen, een vragenlijst die wordt afgenomen bij leerkrachten, en van een vragenlijst die bij de schoolleider wordt afgenomen.

Van alle in schema 2 opgenomen hoofdbestanden zijn een aantal variabelen (schuingedrukt) opgenomen die specifiek zijn voor het dataverzamelings-instrument. Voor het hoofdbestand van de AVI-toets betreft het toetsgegevens. Voor de hoofdbestanden van zowel de vragenlijst voor leerkracht als de vragenlijst schoolleider, gaat het om items uit deze vragenlijsten.

Schema 2 Voorbeeld van drie hoofdbestanden		
<i>Hoofdbestand AVI toets</i>	<i>Hoofdbestand Vragenlijst Leerkracht</i>	<i>Hoofdbestand Vragenlijst schoolleider</i>
LRLGID	LKRID	SCHOOLID
LKRID	SCHOOLID	NAAMBESTAND
SCHOOLID	NAAMBESTAND	Afnamedatum
NAAMBESTAND	LKR_Ervaringsjaren	Meetmoment
Afnamedatum	Afnamedatum	Koppelmeetmoment
Meetmoment	Meetmoment	<i>Aansturing1</i>
Koppelmeetmoment	Koppelmeetmoment	<i>Aansturing2</i>
Groepsnaam	<i>Zelf1</i>	<i>Aansturing3</i>
Groep	<i>Zelf2</i>	<i>Aansturing4</i>
<i>Avibeheersing</i>	<i>Zelf3</i>
<i>Aviïnstructie</i>	<i>Zelf4</i>	
<i>Avifrustratie</i>	<i>Zelf5</i>	
....	<i>Zelf6</i>	
	<i>Zelf7</i>	
	<i>Zelf8</i>	
	

In de eerste kolom van schema 2 kan worden afgelezen dat de resultaten op de AVI-toets van leerlingen van verschillende groepen (door toevoegen van variabele ‘Groep’) in het hoofdbestand AVI-toets worden opgenomen. Wanneer de toets op een ander tijdstip opnieuw wordt afgenomen komen de resultaten in ditzelfde bestand terecht en wel onder elkaar. Eenzelfde leerling komt dan op meerdere regels (rijen) in het bestand voor. Om de meetmomenten te kunnen onderscheiden moeten de variabelen ‘Meetmoment’ en ‘Koppelmeetmoment’ aan worden gemaakt in de hoofdbestanden. Deze variabelen worden nader toegelicht in paragraaf 1.5. Verder dienen de ID’s die aangemaakt zijn in de basisbestanden, ook te worden opgenomen in de hoofdbestanden zodat de data op verschillende meetmomenten en van verschillende respondentniveaus te koppelen zijn.

Voor de AVI-toets geldt dan uiteraard dat de *identifier* 'LRLGID' moet worden opgenomen, om de leerling te kunnen identificeren. Verder willen we ook kunnen achterhalen welke leerkracht op het moment van toetsafname les gaf aan de leerling. Door middel van het opnemen van de *identifier* voor de leerkracht, 'LKRID', wordt dit duidelijk. Door het toevoegen van de *identifier* 'SCHOOLID', kunnen we identificeren op welke school de leerling zat op het moment van toetsafname. Het opnemen van al deze ID's zorgt ervoor dat we ten alle tijde de juiste leerling kenmerken van een leerling uit het basisbestand kunnen koppelen aan zijn/haar toetsgegevens, maar ook dat we gegevens die op leerkracht- en/of schoolniveau zijn verzameld in verband kunnen brengen met de gegevens en resultaten op leerling niveau. Dit betekent dus dat de statische variabelen (respondent kenmerken) uit het basisbestand niet in de hoofdbestanden hoeven te worden opgenomen.

De tweede kolom in schema 2 geeft een overzicht van een aantal variabelen in het hoofdbestand van een vragenlijst voor de leerkracht. Voor de leerkrachtvragenlijst geldt uiteraard dat de *identifier* 'LKRID' moet worden opgenomen om de leerkracht te kunnen identificeren. Verder willen we ook kunnen achterhalen op welke school de leerkracht op een gegeven meetmoment werkzaam is, vandaar dat we ook de *identifier* 'SCHOOLID' opnemen.

In de derde en laatste kolom van schema 2 is het hoofdbestand voor een vragenlijst voor de schoolleider weergegeven. In dit bestand wordt de *identifier* 'SCHOOLID' opgenomen.

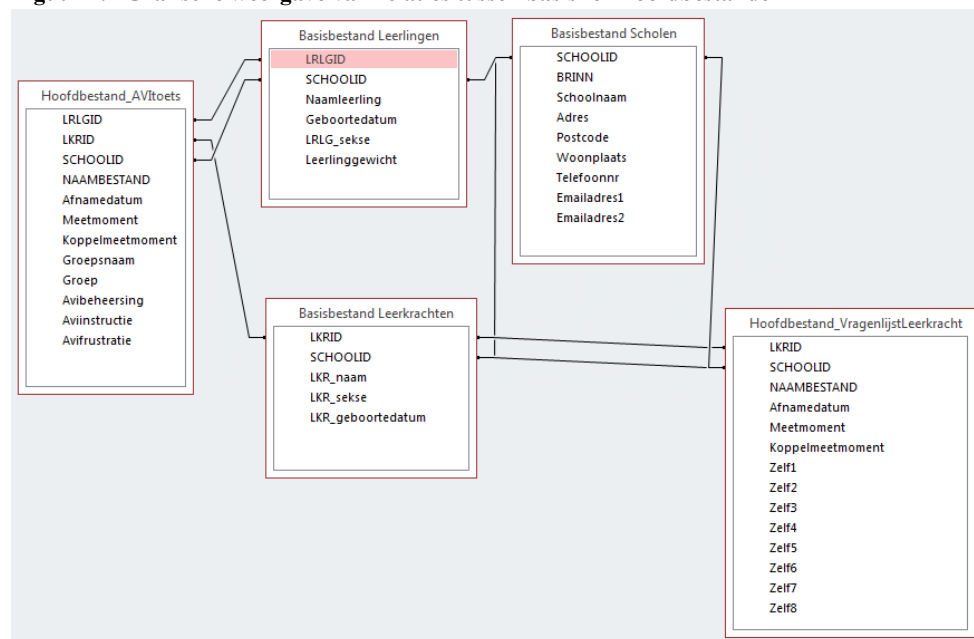
1.3 Opbouw datastructuur

Voordat de onderzoeker begint met het maken van de bestanden is het van belang allereerst de datastructuur schematisch te visualiseren. Het maken van een dergelijk schema geeft inzicht in de bestandsstructuur en -opbouw. In een schematische weergave legt de onderzoeker vast welke basisbestanden nodig zijn voor het vastleggen van de respondentkenmerken (de statische variabelen) en welke hoofdbestanden (voor de dynamische variabelen) aangemaakt moeten worden.

Doorgaans bestaat een hoofdbestand uit gegevens die met een dataverzamelings-instrument meerdere keren in de tijd, soms jaren na elkaar, worden verzameld. Daarnaast wordt per basis- en hoofdbestand weergegeven welke gegevens c.q. variabelen in deze bestanden moeten komen en welke respondent *identifiers* (ID's) in de bestanden moeten worden opgenomen, zodat bestanden later ten behoeve van de analyses aan elkaar kunnen worden gekoppeld door middel van deze ID's. In figuur 1.1 is een voorbeeld gegeven van een dergelijke schematische weergave, op basis van de basis- en hoofdbestanden die in paragraaf 1.2 als voorbeeld zijn genomen. De koppeling tussen de verschillende basis- en hoofdbestanden (op basis van ID's) wordt hierin met lijntjes aangegeven.

Wat en op welke manier er gekoppeld gaat worden hangt af van de data-analyse die ten behoeve van het beantwoorden van een onderzoeksvraag uitgevoerd moet worden.

Figuur 1.1 Grafische weergave van relaties tussen basis- en hoofdbestanden



1.4 Opslaan van data in de hoofdbestanden: lange en brede bestanden

De opbouw – bestandsstructuur – van de hoofdbestanden, hangt af van de analysetechniek die nodig is om de onderzoeksvraag te beantwoorden. Zo wordt in multilevel analyses vaak gebruikt gemaakt van lange databestanden, maar worden repeated measures ANOVA's juist op brede bestanden uitgevoerd. Bestanden kunnen dan ook op twee manieren worden opgebouwd: in de breedte (breed bestand) of in de lengte (lang bestand).

We spreken van een lang bestand als gegevens van alle respondenten (leerlingen, studenten, leerkrachten en of scholen) onder elkaar staan. Een respondent met zijn unieke *identifier* (ID, lees 'LRLGID', 'LKRID' of 'SCHOOLID') kan in zo'n bestand dus meerdere keren voorkomen. De scores van Pietje Puk op één dataverzamelingsinstrument (bijvoorbeeld een toets), op meerdere meetmomenten komen hier dus onder elkaar te staan. Dit betekent ook dat er per dataverzamelingsinstrument (toets/observatie/vragenlijst) een apart databestand moet worden aangelegd. In figuur 1.2 wordt een voorbeeld gegeven van (een deel van) een lang bestand. Lang opslaan van bestanden reduceert de kans op fouten in de bestanden. Daarom adviseren wij, wanneer er sprake is van meerdere meetmomenten, bestanden altijd lang op te bouwen.

Figuur 1.2 Meerdere gegevens van twee leerlingen in een lang bestand

LRLGID	Naamleerling	Afnamedatum	Meetmoment	Avibeheersing
2345	Pietje Puk	13.01.2015	1	6
2345	Pietje Puk	20.05.2015	2	8
2345	Pietje Puk	16.01.2016	3	9
2345	Pietje Puk	10.05.2016	4	11
43578	Jan Klaasen	21.05.2015	2	8
43578	Jan Klaasen	14.01.2016	3	7
43578	Jan Klaasen	19.05.2016	4	10

We spreken van een breed bestand als alle variabelen van een respondent op dezelfde regel (rij) staan. Elke respondent (met zijn unieke ID) komt dus ook maar één keer voor in een breed bestand. Zo worden de gegevens op een bepaalde toets (AVI) van Pietje Puk op de verschillende meetmomenten achter/naast elkaar gezet. In figuur 1.3 wordt een voorbeeld gegeven van (een deel van) een breed bestand.

Figuur 1.3 Meerdere gegevens van dezelfde twee leerlingen in een breed bestand

LRLGID	Naamleerling	Afnamedatum .1	Afnamedatum .2	Afnamedatum .3	Afnamedatum .4	Avibeheer sing.1	Avibeheer sing.2	Avibeheer sing.3	Avibeheer sing.4
2345	Pietje Puk	13.01.2015	20.05.2015	16.01.2016	10.05.2016	6	8	9	11
43578	Jan Klaasen	.	21.05.2015	14.01.2016	19.05.2016	.	8	7	10

In een breed bestand zouden zelfs gegevens van een andere toets (bijvoorbeeld de Drie-Minuten-Toets; DMT) naast deze AVI-toetsgegevens kunnen worden gezet. Eveneens kunnen gegevens worden toegevoegd die op een ander respondentniveau zijn verzameld (bijvoorbeeld vragenlijstgegevens van de leerkracht).

Wanneer voor bepaalde analyses een breed databestand nodig is, dan kan dit altijd gegenereerd worden op basis van de lange bestanden; omgekeerd is (iets) lastiger. Of het nodig is een breed bestand te genereren hangt af van de data-analyse die voor het beantwoorden van een onderzoeksvraag nodig is. Ook als van het begin af aan duidelijk is, dat de geplande data-analyse ter beantwoording van de onderzoeksvraag vraagt om een breed databestand, is het raadzaam de bestanden eerst lang op te bouwen. Dit in verband met de eerdergenoemde reductie van de kans op fouten.

1.5 Datum, meetmoment en koppelmeetmoment

Wanneer bij dezelfde respondenten op verschillende tijdstippen met hetzelfde dataverzamelingsinstrument gegevens worden verzameld, is het noodzakelijk een variabele toe te voegen die ervoor zorgt dat men kan identificeren om welk meetmoment het gaat. Gegevens van alle meetmomenten komen immers onder elkaar te staan in lange databestanden. In principe is dat de datum waarop de meting is verricht: de 'Afnamedatum'.

Wanneer bestanden die meerdere meetmomenten bevatten aan elkaar moeten worden gekoppeld voor een data-analyse die vraagt om een breed bestand (alle gegevens *naast* elkaar), gebeurt dit op basis van de reeds beschreven *identifiers* én afnamedatum. Dat laatste is vaak een probleem: een leerling maakt de AVI-toets bijvoorbeeld 16 mei 2016 en een stilleestoets op de 20^e van diezelfde maand, terwijl de leerkracht is geobserveerd op 10 januari 2016. Koppeling op basis van ‘Afnamedatum’ is hier onmogelijk. Ten einde deze gegevens te kunnen koppelen wordt de ‘Afnamedatum’ omgezet in twee andere variabelen in de hoofdbestanden: ‘Meetmoment’ en ‘Koppelmeetmoment’. In het voorbeeld van een lang bestand (figuur 1.2) waren de beide variabelen ‘Afnamedatum’ en ‘Meetmoment’ al te zien. In het onderstaande wordt hier verder op ingegaan.

1.5.1 Meetmoment

De variabele meetmoment wordt aangemaakt om te kunnen zien wanneer een dataverzamelingsinstrument is afgenomen. Wanneer een dataverzamelingsinstrument voor het eerst wordt afgenomen, is dit ‘Meetmoment’ 1. Het eerstvolgende ‘Meetmoment’ is 2, en zo verder. Het verdient de voorkeur dat er wordt doorgeteld in een volgend schooljaar, wanneer er sprake is van een onderzoek met dataverzameling dat meerdere jaren bestrijkt. Dit reduceert de kans op fouten die gemaakt kunnen worden. Bij de start van een nieuw schooljaar wordt dus niet opnieuw begonnen bij ‘1’. In tabel 1.1 geven we een voorbeeld van het gebruik van de variabele ‘Meetmoment’ binnen een tweejarig onderzoek met twee toetsen op leerlingniveau en een vragenlijst op leerkrachtniveau.

Tabel 1.1 Voorbeeld van het gebruik van de variabele ‘Meetmoment’ in een tweejarig onderzoek

Afnamedatum	AVI-toets leerlingen	Stilleestoets leerlingen	Vragenlijst leerkrachten
Oktober 2014	N.v.t.	1	N.v.t.
Januari 2015	1	2	1
Mei 2015	2	3	N.v.t.
Oktober 2015	N.v.t.	4	N.v.t.
Januari 2016	3	5	2
Mei 2016	4	6	N.v.t.

Uit deze schematische voorstelling wordt duidelijk dat er problemen optreden wanneer wij deze verschillende gegevens zouden willen koppelen ten behoeve van een data-analyse die vraagt om een bestand waarin variabelen uit verschillende basis- en hoofdbestanden met elkaar gecombineerd moeten worden. Wanneer we koppelen op de variabele ‘Meetmoment’ gaat het mis, omdat de meetmomenten van de gebruikte dataverzamelingsinstrumenten niet samenvallen. Om dit op te lossen wordt naast de variabele ‘Meetmoment’ ook de variabele ‘Koppelmeetmoment’ opgenomen in de hoofdbestanden (zie ook schema 2 en figuur 1.1).

1.5.2 Koppelmeetmoment

Om gegevens uit verschillende hoofdbestanden te kunnen koppelen om te komen tot een breed bestand, is het nodig dat er een variabele is waarmee de meetmomenten op een uniforme (lees: vergelijkbare) wijze worden aangegeven. We gaan hierbij uit van het allereerste tijdstip of meetmoment. In het voorbeeld in tabel 1.1 zien we dat er maximaal 3 meetmomenten per schooljaar zijn per dataverzamelingsinstrument. We hebben dan dus 3 koppelmeetmomenten per jaar. Ook bij de variabele ‘Koppelmeetmoment’ wordt er doorgeteld bij een volgend schooljaar. Voor de verzamelde gegevens uit het voorbeeld in tabel 1.1 ziet dit er als volgt uit (tabel 1.2):

Tabel 1.2 Voorbeeld van het gebruik van de variabele ‘Koppelmeetmoment’ in een tweejarig onderzoek

Afnamedatum	Koppelmeetmoment	AVI	Stillestoets	Vragenlijst
Oktober 2014	1	N.v.t.	1	N.v.t.
Januari 2015	2	2	2	2
Mei 2015	3	3	3	N.v.t.
Oktober 2015	4	N.v.t.	4	N.v.t.
Januari 2016	5	5	5	5
Mei 2016	6	6	6	N.v.t.

Nu zien we dus dat er voor de AVI-toets alleen gegevens zijn op ‘Koppelmeetmoment’ 2, 3, 5 en 6. Voor de vragenlijst voor de leerkracht zijn er alleen gegevens op ‘Koppelmeetmoment’ 2 en 5. Voor de stillestoets zijn er resultaten op alle Koppelmeetmomenten.

Wanneer de gegevens gekoppeld (of naast elkaar gezet) worden op basis van de variabele ‘Koppelmeetmoment’, weten we zeker dat de gegevens die op hetzelfde moment in het jaar verzameld zijn, aan elkaar gekoppeld worden en niet aan data die op een ander moment verzameld zijn; ‘Koppelmeetmoment’ 2 is nu immers voor beide toetsen en de afgenomen vragenlijst identiek.

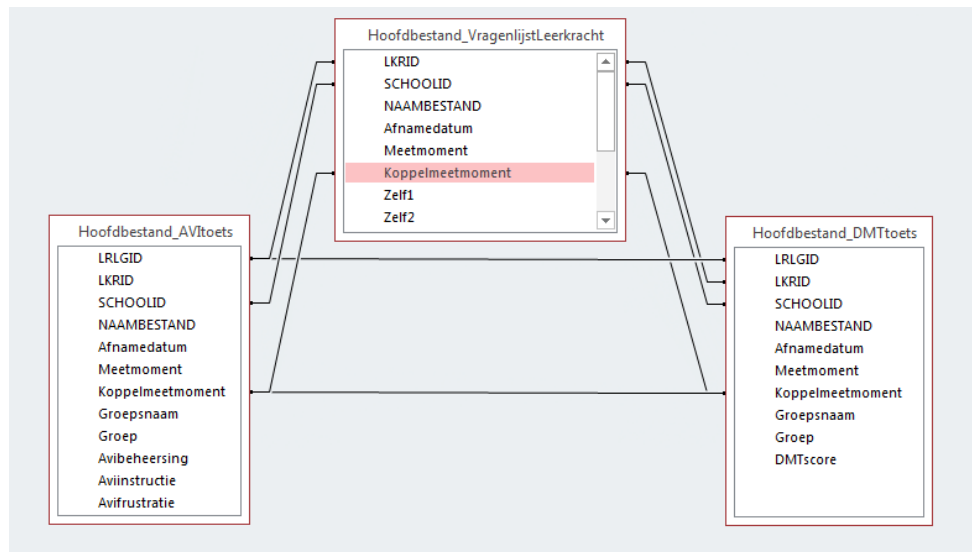
Ter verduidelijking worden in tabel 1.3 de variabelen ‘Afnamedatum’, ‘Meetmoment’ en ‘Koppelmeetmoment’ naast elkaar gezet op basis van het voorbeeldonderzoek uit de voorgaande tabellen (1.1 en 1.2).

Tabel 1.3 Voorbeeld van het gebruik van de variabelen ‘Afnamedatum’, ‘Meetmoment’ en ‘Koppelmeetmoment’ in een tweejarig onderzoek

	Meetmoment per dataverzamelingsinstrument			Koppelmeetmoment in alle hoofdbestanden
Afnamedatum	AVI-toets leerlingen	Stillestoets leerlingen	Vragenlijst leerkrachten	
Oktober 2014	N.v.t.	1	N.v.t.	1
Januari 2015	1	2	1	2
Mei 2015	2	3	N.v.t.	3
Oktober 2015	N.v.t.	4	N.v.t.	4
Januari 2016	3	5	2	5
Mei 2016	4	6	N.v.t.	6

In Figuur 1.4 is een voorbeeld gegeven van de koppeling tussen drie bestanden op hetzelfde ‘Koppelmeetmoment’. Het betreft in dit voorbeeld een vragenlijst voor de leraar (met *identifier* ‘LKRID’) en twee verschillende toetsen op leerlingniveau (met *identifier* ‘LRLGID’), AVI en DMT. In tegenstelling tot figuur 1.1, waar verschillende bestanden alleen zijn gekoppeld op basis van meerdere *identifiers* in verschillende databestanden, wordt in Figuur 1.4 gekoppeld op meerdere *identifiers* (‘LRGID’ en ‘LKRID’) in combinatie met de variabele ‘Koppelmeetmoment’. Een goede datastructuur garandeert dat men gegevens uit verschillende databestanden altijd weer bij elkaar kan brengen in één bestand ten behoeve van de gewenste analyses.

Figuur 1.4 Voorbeeld van koppeling door LKRID, LRLGID en Koppelmeetmomenten met elkaar te verbinden.



1.6 Samenvatting

In dit hoofdstuk is het verschil tussen basisbestanden en hoofdbestanden uitgelegd. Voorafgaand aan het uitzetten van de dataverzamelingsinstrumenten die in het onderzoek ingezet gaan worden, is het van belang na te denken over de opbouw van de databestanden en hoe de data dienen te worden opgeslagen. Door een schema te maken van de variabelen die nodig zijn om de onderzoeksvragen te beantwoorden, ontstaat inzicht in de basis- en hoofdbestanden die nodig zijn en welke respondent-*identifiers* (ID's) gebruikt moeten worden om de afzonderlijke bestanden aan elkaar te kunnen koppelen. Wanneer er sprake is van meerdere meetmomenten, dienen tevens de variabelen 'Meetmoment' en 'Koppelmeetmoment' aan te worden gemaakt. Hiermee kunnen de gegevens die bij de respondenten met eenzelfde dataverzamelingsinstrument op verschillende momenten worden verzameld, worden onderscheiden. Met de variabele 'Koppelmeetmoment' kunnen, in combinatie met de respondent-*identifiers*, gegevens die (eventueel op verschillende respondentniveaus) met verschillende dataverzamelingsinstrumenten zijn verzameld, worden gekoppeld.

Hoofdstuk 2

BESTANDSOPBOUW

2.1 Inleiding

In hoofdstuk 1 is uitgelegd hoe de datastructuur van een onderzoek moet worden vastgelegd. Wanneer duidelijk is welke basis- en hoofdbestanden in het onderzoek nodig zijn en welke variabelen in deze bestanden opgenomen dienen te worden, kan worden begonnen met de bestandsopbouw. Met bestandsopbouw bedoelen wij uitdrukkelijk niet het vullen van SPSS-bestanden met onderzoeksgegevens. Dit is een volgende stap (beschreven in hoofdstuk 3). Bestandsopbouw betreft het aanmaken van de losse basis- en hoofdbestanden in het programma dat gebruikt gaat worden voor het uitvoeren van de berekeningen (in ons geval SPSS) en het in elk van de bestanden aanmaken en definiëren (van een label voorzien) van de variabelen. Alle variabelen die zijn benoemd in de datastructuur worden nu overgenomen in SPSS. Ook wordt in SPSS een zogeheten codeboek aangelegd. De kenmerken van de variabelen worden in SPSS ingevoerd bij het tabblad VARIABLE VIEW. Dit wordt toegelicht in paragraaf 2.2. We introduceren dan ook een paar SPSS syntaxen. Daarna zullen in paragraaf 2.3 voorbeelden van variabelen worden gegeven die in de basisbestanden staan. In de laatste paragraaf (2.4) worden voorbeelden gegeven van veel voorkomende variabelen in de hoofdbestanden.

2.2 SPSS Variabele view

Bestandsopbouw betreft de opbouw van de bestanden (basis- en hoofdbestanden) binnen het programma SPSS.

Onder VARIABLE VIEW in SPSS worden de variabelen met hun kenmerken ingevoerd en weergegeven.

Het gaat per variabele om de volgende kenmerken:

- NAME; hier wordt de variabele van een naam voorzien. Het is van belang een betekenisvolle en herkenbare naam te kiezen. Items die samen een schaal vormen krijgen steeds dezelfde naam met een volgnummer erachter;
- TYPE; dit is het type van de variabele; STRING zijn letters, cijfers en tekens (het wordt gelezen als een tekst), NUMERIC zijn alleen cijfers, DATE kan worden gebruikt als een datum moet worden ingevoerd;
- WIDTH; staat voor de maximale lengte van de variabele;
- DECIMALS; het aantal decimalen dat zichtbaar mag zijn in DATA VIEW;
- LABEL; dit is de uitgebreide beschrijving van de variabele naam (hier kan bijvoorbeeld de vraag die gesteld is letterlijk ingevuld worden). In verband met de overdraagbaarheid en volledigheid wordt aangeraden dit altijd in te vullen;
- VALUES; hier moeten numerieke waarden van de variabele worden gespecificeerd. (bijvoorbeeld LKR_sekse; 1=vrouw 2=man);
- MISSING; dit is de waarde die getoond moet worden als er gegevens ontbreken; dit mag nooit een waarde zijn die in het onderzoek betekenis heeft (kies bijvoorbeeld een hoog getal, zoals 9999. Het is wel van belang de WIDTH (zie hierboven) hierop in te stellen);
- COLUMNS; hier kan de breedte van het vakje worden ingevuld zoals deze zichtbaar is in DATAVIEW (dit doet SPSS automatisch);
- ALIGN; geeft aan hoe de uitlijning is in DATA VIEW (dit doet SPSS automatisch);
- MEASURE; beschrijft het meetniveau:
 - o NOMINAL; De waarden zijn dan uitsluitend bedoeld om categorieën (jongen/meisje en man/vrouw) aan te duiden van een variabele ('LRLG_sekse' respectievelijk 'LKR_sekse');

- ORDINAL; Een variabele is ordinaal als de antwoordmogelijkheden op een logische wijze op volgorde kunnen worden gezet, zoals bij de items ('Zelf1' tot en met 'Zelf8, zie schema 2, hoofdstuk 1) die gecombineerd kunnen worden tot een Likertschaal. De afstand tussen de verschillende antwoordmogelijkheden is niet (altijd) even groot (bijvoorbeeld 'eens', 'neutraal', 'oneens');
- SCALE; Een variabele heeft een interval (scale) meetniveau als de waarden (antwoordmogelijkheden) van de variabele uitsluitend en uitputtend zijn, logisch gesorteerd kunnen worden en de afstanden tussen de verschillende waarden steeds even groot zijn. Als hieraan is voldaan mogen kengetallen als het gemiddelde en de standaarddeviatie berekend worden. Denk bijvoorbeeld aan de variabele 'Leeftijd', maar ook de zojuist genoemde Likertschaal.

2.3 Variabelen Basisbestanden

In deze paragraaf geven we een aantal voorbeelden van variabelen die in de basisbestanden kunnen worden opgenomen. Daarbij bouwen we consequent voort op de in hoofdstuk 1 ontwikkelde datastructuur en de daarbinnen opgenomen databestanden. Naast het aanmaken en definiëren van deze variabelen in SPSS, is het ook aan te raden de variabelen met hun kenmerken vast te leggen in een codeboek. Als alle variabelen met bijbehorende omschrijvingen zijn gedefinieerd in SPSS dan kunnen via het commando CODEBOOK alle eigenschappen van de variabelen worden opgevraagd. De output die met behulp van dit commando wordt verkregen kan dan als codeboek worden opgeslagen.

Hieronder vragen we de eigenschappen op van de variabelen 'LRLGID', 'Naamleerling' en 'Afnamedatum'.

```
CODEBOOK LRLGID Naamleerling Afnamedatum
/VARINFO POSITION LABEL TYPE FORMAT MEASURE
VALUELABELS MISSING /STATISTICS NONE.
```


In onderstaande tabellen (tabel 2.1, 2.2 en 2.3) geven we per basisbestand voorbeelden van variabelen met hun kenmerken.

Het aanmaken en definiëren van variabelen in een SPSS databestand kan op twee manieren; via ‘klikken’ in het menu en via een syntax. Het werken met een syntax heeft de voorkeur omdat hiermee fouten kunnen worden voorkomen. Met ‘klikken op OK’ in het menu kan achteraf niet meer worden achterhaald welke stappen er precies zijn gezet. Door de stappen vast te leggen in een syntax is dit wel voor iedereen na te gaan. Een syntax kan in principe eenvoudig worden aangemaakt door in een keuzemenu niet op OK, maar juist op PASTE te klikken. Alle tot dan toe gemaakte keuzes in dit menu worden als een tekst weggeschreven in een syntax-bestand. Vanuit de aangemaakte syntax kan het zojuist geplakte commando dan uitgevoerd worden. Uiteraard kan men de commando’s ook gewoon zelf typen.

Voor iedere variabele wordt in onderstaande tabellen het format weergegeven zoals het gebruikt dient te worden in de syntax. Een A betekent een STRING-variabele en een F is een NUMBER. Het getal achter de A of F geeft weer uit hoeveel tekens de variabele mag bestaan (WIDTH). Daarnaast wordt het label van de variabele weergegeven.

In tabel 2.1 geven we enkele voorbeelden van variabelen, met hun kenmerken, die in het basisbestand met gegevens van leerlingen worden aangemaakt.

Tabel 2.1 Voorbeelden van variabelen met hun kenmerken in het basisbestand ‘leerlingen’		
NAME	LABEL	FORMAT
LRLGID	Uniek nummer van de leerling	F6
SCHOOLID	Het unieke nummer van de school	F6
Naamleerling	Volledig naam van de leerling	A100
Geboortedatum	Geboortedatum leerling	EDATE11
LRLG_sekse	Geslacht (1; meisje, 2; jongen)	F1
Leerlinggewicht	Leerlinggewicht (opleidingsniveau ouders) (0=geen, 1=0.3, 2=1.2)	F1

De variabelen in tabel 2.1 kunnen worden aangemaakt in SPSS VARIABLE VIEW door de namen ervan in te typen in de kolom NAME. Standaard verschijnt dan in de kolom TYPE ‘NUMERIC’, in de kolom WIDTH een ‘8’ en in de kolom DECIMALS een ‘2’.

Dit kan handmatig worden aanpast zodat de variabelen de gewenste eigenschappen bevatten. De eigenschappen van de variabelen kunnen echter ook worden aangepast met behulp van SPSS commando's; dit verdient, zoals eerder aangegeven, de voorkeur. Hieronder laten we zien hoe het LABEL, FORMAT en LEVEL van de variabele 'LRLGID' bijgewerkt kunnen worden:

```
VARIABLE LABELS LRLGID "Uniek nummer van de leerling".
FORMATS LRLGID (F6.0).
VARIABLE LEVEL LRLGID (NOMINAL).
```

Uiteraard kunnen ook andere eigenschappen van deze variabele (en van de andere variabelen) worden bijgewerkt met behulp van stukken syntax.

In tabel 2.2 worden voorbeeldvariabelen met hun kenmerken uit het basisbestand met gegevens van leerkrachten weergegeven.

Tabel 2.2 Voorbeelden van variabelen met hun kenmerken in het basisbestand 'leerkrachten'		
NAME	LABEL	FORMAT
LKRID	Unieke nummer van de leerkracht	F6
SCHOOLID	Het unieke nummer van de school	F6
LKR_Naam	Volledige naam van de leerkracht	A100
LKR_sekse	Geslacht van de leerkracht (1=vrouw; 2=man)	F1
LKR_Geboortedatum	Geboortedatum leerkracht	A100

Ten slotte zijn voorbeelden van variabelen met hun kenmerken die in het basisbestand met gegevens van scholen voorkomen weergegeven in tabel 2.3.

Tabel 2.3 Voorbeelden van variabelen met hun kenmerken in het basisbestand 'scholen'		
NAME	LABEL	FORMAT
SCHOOLID	Het unieke nummer van de school	F6
BRINN	BRINN code/instellingscode van de school	A100
Schoolnaam	De naam van de school	A100
Adres	Adres van de school (straatnaam plus nummer)	A100
Postcode	Postcode van de school	A100
Woonplaats	De plaats waar de school zich bevindt	A100
Telefoonnr	Telefoonnummer van de school	A100
Emailadres1	Emailadres van eerste contactpersoon van de school	A100
Emailadres2	Emailadres van tweede contactpersoon van de school	A100

2.4 Variabelen Hoofdbestanden

In deze paragraaf worden enkele voorbeelden gegeven van hoofdbestanden en de variabelen die daarin voor kunnen komen. Het gaat hierbij dus om het aanmaken van variabelen in de hoofdbestanden. Evenals met de basisbestanden is het ook raadzaam van deze bestanden codeboeken aan te leggen.

In tabel 2.4 wordt weergegeven welke variabelen met bijhorende kenmerken, thuishoren in het hoofdbestand van een toets, in dit geval de AVI-toets.

Tabel 2.4 Voorbeelden van variabelen met hun kenmerken in het hoofdbestand 'AVI'		
NAME	LABEL	FORMAT
LRLGID	Uniek nummer per leerling	F6
LKRID	Unieke nummer van de leerkracht	F6
SCHOOLID	Het nummer dat de school heeft in het onderzoek	F6
NAAMBESTAND	Het bestand waar de gegevens uit komen	A100
Afnamedatum	Afnamedatum AVI toets	EDATE11
Meetmoment	Het betreffende meetmoment waar de toetsgegevens bij horen	F2
Koppelmeetmoment	Het betreffende koppelmeetmoment waar de toetsgegevens bij horen	F2
Groepsnaam	Is de naam voor de groep die de school gebruikt	A100
Groep	Is het numerieke getal van de klas waarin de leerling zit	F1
Avibeheersing	Hoogst behaalde AVI beheersingsniveau door de leerling	F2
Aviinstructie	Hoogst behaalde AVI instructieniveau door de leerling	F2
Avifrustratie	AVI frustratieniveau behaald door de leerling	F2

Zoals in paragraaf 2.3 is aangegeven, kunnen de kenmerken van de variabelen met behulp van een aantal SPSS commando's worden bijgewerkt zodat de variabelen de gewenste eigenschappen bevatten.

In tabel 2.5 wordt een voorbeeld gegeven voor een vragenlijst die wordt afgenomen bij een leerkracht.

Tabel 2.5 Voorbeelden van variabelen met hun kenmerken in het hoofdbestand 'Vragenlijst leerkracht'		
NAME	LABEL	FORMAT
LKRID	Unieke nummer van de leerkracht	F6
SCHOOLID	Het nummer dat de school heeft in het onderzoek	F4
NAAMBESTAND	Het bestand waar de gegevens uit komen	A100
LKR_Ervaringsjaren	Aantal jaren ervaring dat de leerkracht heeft op betreffend meetmoment	F2
Afnamedatum	Datum waarop de leerkracht de vragenlijst heeft ingevuld	EDATE11
Meetmoment	Het betreffende meetmoment waar de toetsgegevens bij horen	F2
Koppelmeetmoment	Het betreffende koppelmeetmoment waar de toetsgegevens bij horen	F2
Zelf1	Item 1 schaal 'vertrouwen in eigen kunnen': Ik kan zelfs de moeilijkste kinderen onderwijzen.	F2
Zelf2	Item 2 schaal 'vertrouwen in eigen kunnen': Ik heb in de klassensituatie vaak niet de mogelijkheid om opdrachten aan te passen aan het niveau van individuele leerlingen.	F2
Zelf3	Item 3 schaal 'vertrouwen in eigen kunnen': Ik kan vrijwel elk probleem aan.	F2
Zelf4	Item 4 schaal 'vertrouwen in eigen kunnen': Wanneer een leerling een hoger cijfer/beschrijvende beoordeling krijgt dan gewoonlijk, dan komt dit vrijwel altijd doordat ik die leerling beter les heb gegeven.	F2
Zelf5	Item 5 schaal 'vertrouwen in eigen kunnen': Als ik echt mijn best doe, kan ik zelfs met de meest moeilijke of ongemotiveerde leerling voldoende bereiken.	F2
Zelf6	Item 6 schaal 'vertrouwen in eigen kunnen': Wanneer een aantal leerlingen sneller vorderingen maakt dan gebruikelijk, komt dit doordat ik een meer effectieve benadering heb gebruikt.	F2
Zelf7	Item 7 schaal 'vertrouwen in eigen kunnen': Als één van mijn leerlingen een opdracht niet kan maken, kan ik precies aangeven wat daaraan te moeilijk was voor hem/haar.	F2
Zelf8	Item 8 schaal 'vertrouwen in eigen kunnen': Ik weet niet wat ik moet doen als kinderen niet willen leren.	F2

Ten slotte zijn voorbeelden van variabelen, met hun kenmerken, uit het hoofdbestand voor een vragenlijst voor schoolleiders opgenomen in tabel 2.6.

Tabel 2.6 Voorbeelden van variabelen met hun kenmerken in het hoofdbestand 'Vragenlijst schoolleider'

NAME	LABEL	FORMAT
SCHOOLID	Het nummer dat de school heeft in het onderzoek	F4
NAAMBESTAND	Het bestand waar de gegevens uit komen	A100
Afnamedatum	Datum waarop de leerkracht de vragenlijst heeft ingevuld	EDATE11
Meetmoment	Het betreffende meetmoment waar de toetsgegevens bij horen	F2
Koppelmeetmoment	Het betreffende koppelmeetmoment waar de toetsgegevens bij horen	F2
Aansturing1	Item 1 schaal 'aansturing door de schoolleider': Is er een kartrekker voor de implementatie van het project benoemd?	F2
Aansturing2	Item 2 schaal 'aansturing door de schoolleider': Is de taakstelling van de kartrekker vastgelegd?	F2
Aansturing3	Item 3 schaal 'aansturing door de schoolleider': Heeft de kartrekker uren voor de uitvoering van zijn/haar taken?	F2
Aansturing4	Item 4 schaal 'aansturing door de schoolleider': Is vastgelegd wanneer de kartrekker feedback geeft aan de directie?	F2

2.5 Samenvatting

In dit hoofdstuk is besproken wat onder bestandsopbouw wordt verstaan; namelijk het creëren van lege basis- en hoofdbestanden met de bijbehorende variabelen en hun kenmerken. In SPSS zijn bij VARIABLE VIEW de variabelen met de betreffende kenmerken te zien. Deze variabelen kunnen via SPSS VARIABLE VIEW worden aangemaakt en gedefinieerd. Het verdient de voorkeur om de kenmerken van de variabelen aan te maken met behulp van een syntax. Die syntax kan men namelijk bewaren en eventueel ook voor andere bestanden met dezelfde variabelen gebruiken. Voor de variabelen, zowel in de basisbestanden als in de hoofdbestanden, worden de kenmerken vooraf vastgelegd.

Daarnaast is het raadzaam een codeboek aan te maken met daarin de variabelen met bijbehorende kenmerken uit de verschillende basisbestanden en hoofdbestanden.

Hoofdstuk 3

DATA VERZAMELEN EN INVOEREN IN SPSS

3.1 Inleiding

Naast het vaststellen van de datastructuur voor het onderzoek (hoofdstuk 1) en het werken aan de bestandsopbouw (hoofdstuk 2) moet er een besluit worden genomen over de manier waarop onderzoeksgegevens worden verzameld.

Er zijn verschillende manieren om data te verzamelen; op papier (paragraaf 3.2.1), via Excelbestanden (paragraaf 3.2.2), via leerling- of studentvolgsystemen (paragraaf 3.2.3), en via digitale survey systemen (zoals Formdesk en Survey Monkey; paragraaf 3.2.4). Wanneer deze data binnenkomen is het verstandig deze eerst aan een korte visuele inspectie te onderwerpen (paragraaf 3.3) alvorens deze in SPSS in te voeren. Fouten en omissies kunnen zo al vroegtijdig worden opgespoord en er kan op dat moment dan direct om opheldering bij de respondent worden gevraagd. Ongeacht de manier van dataverzameling zullen de data uiteindelijk moeten worden ingevoerd in het rekenpakket SPSS. Afhankelijk van hoe de data binnenkomen, vraagt dit om bepaalde handelingen. In paragraaf 3.4 wordt besproken welke handelingen er nodig zijn om de data binnen het rekenpakket SPSS te krijgen. In paragraaf 3.5 wordt vervolgens besproken hoe gegevens uit meerdere systemen bij elkaar kunnen worden gevoegd in SPSS. Deze stap is van toepassing wanneer de respondenten in het onderzoek eenzelfde soort gegevens op verschillende manieren aanleveren.

3.2 Data verzamelen

3.2.1 Data op papier

Allereerst is het goed op te merken dat deze variant in afnemende mate voorkomt. Maar in sommige gevallen vraagt het type data dat wordt verzameld of de groep respondenten waarbij de data worden verzameld, om verzameling op papier.

Denk hierbij bijvoorbeeld aan grote groepen leerlingen waarbij een toets klassikaal moet worden afgenomen. Dataverzameling op papier is redelijk foutgevoelig, omdat gegevens later overgetypt moeten worden om ze in SPSS te kunnen verwerken.

3.2.2 Data in Excel

Data kunnen ook worden aangeleverd in Excel. Vaak worden dan door de onderzoeker voorgedefinieerde Excelbestanden naar de respondenten gestuurd om ervoor te zorgen dat alle data op de door de onderzoeker gewenste manier worden aangeleverd. De respondenten sturen de door hen ingevulde Excelbestanden veelal per email terug aan de onderzoeker.

3.2.3 Data in leerling- of studentvolgsysteem

Data die door scholen via een leerlingvolgsysteem worden verzameld, kunnen door de onderzoeker opgevraagd worden via een export vanuit het betreffende systeem. In deze systemen worden alle schoolkenmerken, leerlingkenmerken en toetsresultaten verzameld en opgeslagen. Door de gegevens uit de leerlingvolgsystemen te halen hoeven de scholen de gegevens niet nog een keer in een apart bestand over te typen. Dit vraagt echter om voorzieningen aan de kant van de scholen en bij sommige systemen ook aan de kant van de onderzoekers.

In alle systemen zitten al (enkele) exportmogelijkheden. Het hangt echter van de onderzoeksvra(a)g(en) en de daarbij benodigde data af, of het leerlingvolgsysteem en de bijbehorende exportmogelijkheden alle benodigde gegevens bevat. Dit zal eerst door de onderzoeker moeten worden uitgezocht. Indien de gegevens uit het systeem ontoereikend zijn voor het beantwoorden van de onderzoeksvra(a)g(en), is het nodig aanvullende gegevens via een ander systeem te verzamelen.

Er zijn vier leerlingvolgsystemen in het basisonderwijs te onderscheiden die veel door scholen worden gebruikt:

- Parnassys;
- ESIS;
- Dotcom;
- CITO LOVS.

Bij Parnassys en ESIS is het zo dat wanneer de school in het systeem een export maakt, er een Excelbestand met de gegevens wordt gegenereerd dat door de school aan de onderzoeker kan worden gemaild. Bij CITO LOVS en Dotcom wordt er gewerkt met een export die door het systeem in een zip-bestand wordt gezet. Dit zip-bestand kan dan door de school aan de onderzoeker worden gemaild.

Binnen de Hogeschool Utrecht wordt ook een systeem gebruikt om gegevens van studenten op te slaan: OSIRIS. In dit systeem worden alle studentkenmerken en hun resultaten op toetsen verzameld en opgeslagen. De gegevens uit dit systeem kunnen via een export in Excel worden gezet en aan de onderzoeker worden aangeleverd.

3.2.4 Data in digitaal survey systeem

Wanneer het onderzoek vraagt om het verzamelen van gegevens door middel van vragenlijsten of observatielijsten, dan kan dit door middel van een online survey systeem. Formdesk en Survey Monkey zijn veelgebruikte, gebruiksvriendelijke systemen. De onderzoeker creëert in een van deze systemen de gewenste vragenlijst. Het systeem genereert dan een link naar deze lijst. Deze link kan door de onderzoeker per email aan de respondenten worden gestuurd. Of er kan gekozen worden om een (persoonlijke) uitnodiging tot invullen van de lijst te laten versturen door het systeem. Respondenten kunnen deze vragenlijst nu op ieder gewenst moment en gewenste plaats invullen. In het vervolg van deze rapportage wordt met digitale survey systemen Formdesk bedoeld. Andere digitale survey systemen worden niet verder uitgewerkt.

3.3 Visuele inspectie van binnengekomen data

Voordat de aangeleverde bestanden in SPSS kunnen worden ingevoerd, moeten deze gecontroleerd worden. Dit duiden we aan met de term ‘visuele inspectie’. Visuele inspectie voorkomt dat eventuele fouten in het aangeleverde bestand worden ingevoerd in het betreffende basis-, dan wel hoofdbestand. Daarnaast zorgt dit er ook voor dat er minder datacleaning nodig is alvorens de data-analyses worden uitgevoerd. Wanneer de data binnenkomen moet allereerst worden bekeken of het inderdaad de gegevens betreft die zijn opgevraagd; gaat het bijvoorbeeld echt om de toetsgegevens van het gewenste meetmoment. Ook moet worden nagegaan of de gegevens op de juiste plek staan en of de gegevens volledig zijn aangeleverd. Bij de leerlingvolgsystemen (Parnassys, ESIS, Dotcom en CITO LOVS) moet er vooral op worden gelet of de juiste gegevens in de export staan en of er geen incorrecte gegevens bij staan. De afnamedatum (van de toets) is hiervoor zeer belangrijk. Soms wordt de afnamedatum niet genoemd, maar een schooljaar met een nummer. Hier moet goed op gelet worden, want soms zijn de gegevens niet up-to-date. Dit kan er mogelijk voor zorgen dat gegevens van een eerder meetmoment in de bestanden terechtkomen. Wanneer uit de visuele inspectie blijkt dat er gegevens ontbreken, onvolledig dan wel onjuist zijn, is het raadzaam eerst hierover contact op te nemen met de respondent alvorens de gegevens in SPSS in te voeren.

3.4 Bestanden in SPSS invoeren

Na de visuele inspectie kan het daadwerkelijke invoeren van de data in SPSS beginnen. In deze paragraaf zullen we per systematiek beschrijven hoe dit in zijn werk gaat.

3.4.1 Data op papier

Data die op papier zijn aangeleverd moeten handmatig in SPSS worden ingevoerd. Dit betekent dat de aangeleverde gegevens in SPSS moeten worden overgetypt in vooraf klaargezette databestanden.

Deze manier van dataverwerking is gevoelig voor fouten. Een andere mogelijkheid is het inscannen van de op papier binnengekomen gegevens via een speciale scanner of tekstverwerkingsprogrammatuur (bijvoorbeeld Kurzweil).

3.4.2 Data in Excel

Het verwerken van via een Excelbestand aangeleverde data in SPSS, gaat met behulp van een syntax. Hieronder zullen we basale stukken syntax bespreken die gebruikt kunnen worden om data vanuit een Excelbestand in SPSS te zetten.

Allereerst gaat het om het syntax commando GET DATA om het Excelbestand op te halen en in te lezen in SPSS:

```
GET DATA
/TYPE=XLSX
/FILE='F:\pad\naambestand.xlsx'
/SHEET=name 'Sheet1'
/CELLRANGE=full
/READNAMES=on
/ASSUMEDSTRWIDTH=32767.
EXECUTE.
DATASET NAME DataToets WINDOW=FRONT.
```

Dit stuk syntax zorgt ervoor dat de data uit Excel in SPSS worden gezet. Dit commando wordt uitgevoerd voor ieder in te lezen Excelbestand. Een aantal opmerkingen bij de syntax: door middel van de laatste regel, DATASET NAME wordt er tijdens het werken in SPSS een naam aan het databestand gegeven. Bij FILE moet het juiste pad naar het Excelbestand staan, anders wordt dit commando niet uitgevoerd en wordt er een foutmelding gegenereerd. Dit geldt ook voor de naam die bij SHEET staat; daar moet de naam staan van het tabblad uit het Excelbestand dat ingelezen dient te worden. Bij TYPE moet goed gelet worden op de versie van Excel waarmee gewerkt wordt; XLSX geeft aan dat het om Excel 2007 of een latere versie gaat.

Nadat de data vanuit Excel in SPSS zijn gezet, is het van belang een aantal variabelen aan te maken. Hieronder worden deze besproken aan de hand van de syntax waarmee deze variabelen kunnen worden gecreëerd.

In hoofdstuk 1 is gesproken over het belang van het toevoegen van de variabelen 'Meetmoment' en 'Koppelmeetmoment', wanneer er sprake is van gegevens die meerdere keren in een onderzoek worden verzameld. De variabele 'Meetmoment' kan als volgt worden gecreëerd:

```
COMPUTE Meetmoment=3.  
FORMATS Meetmoment (F1.0).  
VARIABLE LABELS Meetmoment "Januari 2016".  
EXECUTE.
```

Verder is het aan te raden een variabele 'Schooljaar' aan te maken. In combinatie met de variabele 'Meetmoment' kan zo altijd worden nagegaan welke gegevens het precies betreft.

```
STRING SCHOOLJAAR(A9).  
COMPUTE SCHOOLJAAR="2015-2016".
```

Om naderhand te achterhalen wanneer de betreffende gegevens in SPSS zijn ingevoerd, kan er een datum van invoeren worden aangemaakt. Dit kan natuurlijk handmatig, maar het is handiger om dit via een syntax te doen. De datum kan namelijk uit "het geheugen" van SPSS worden gehaald. De syntax ziet er als volgt uit:

```
STRING Importdatum(A10).  
COMPUTE Importdatum=$DATE.  
COMPUTE Importdatum1=NUMBER(Importdatum,EDATE11).  
ALTER TYPE Importdatum1(EDATE11).  
DELETE VARIABLES Importdatum.  
RENAME VARIABLES Importdatum1=Importdatum.  
EXECUTE.
```

Om de datum van invoeren van de gegevens uit het geheugen van SPSS te halen wordt voorgaande syntax gebruikt. Hier wordt eerst een variabele aangemaakt ('Importdatum') en dan de datum "uit het geheugen van SPSS" gehaald. Deze datum heeft een Amerikaanse notatie, omdat SPSS in het Amerikaans "denkt". We laten in voorgaande syntax daarom een nieuwe variabele aanmaken ('Importdatum1'), waarin deze Amerikaanse datum wordt omgezet naar EDATE11 (Europese notatie). Uiteindelijk vervangen we dan de variabele met de Amerikaanse notatie door de variabele met de Europese notatie.

Wanneer data vanuit Excel worden ingelezen in SPSS, neemt SPSS automatisch de bovenste regel van het Excelbestand (de 'kopjes') als variabelennamen over. Het kan zo zijn dat deze namen niet overeenkomen met de variabelennamen die vooraf voor de basis- en hoofdbestanden zijn gekozen. Wanneer dit het geval is, is het nodig ervoor te zorgen dat de gegevens bij de juiste, vooraf gedefinieerde, variabelen(namen) terechtkomen. Dit gebeurt door eerst een variabele aan te maken met de gewenste naam (bijvoorbeeld 'Naamschool'), daarna de gegevens van de oude variabele (uit Excel) (bijvoorbeeld 'School') hierin op te nemen en dan de oude variabele te verwijderen:

```
STRING Naamschool(A100).  
EXECUTE.  
COMPUTE Naamschool=School.  
EXECUTE.  
DELETE VARIABLES School.  
EXECUTE.
```

3.4.3 Data in leerlingvolgsysteem - Parnassys

Wanneer scholen hun gegevens aanleveren via een export uit het leerlingvolgsysteem Parnassys, dan gebeurt dit in de vorm van een Excelbestand. Dit bestand bestaat uit twee tabbladen; één tabblad met de leerlinggegevens en één tabblad met toetsresultaten.

De gegevens van deze beide tabbladen moeten eerst samengevoegd worden om ze te kunnen opnemen in het juiste hoofdbestand. Dit samenvoegen gebeurt aan de hand van een koppelvariabele die in beide tabbladen voorkomt. Parnassys geeft in het Excelbestand een eigen nummer voor iedere leerling (LRLGID) en dit is dan ook de koppelvariabele tussen de twee tabbladen.

Om de twee tabbladen samen te kunnen voegen worden eerst de leerlinggegevens opgehaald, daarna de toetsresultaten. De toetsgegevens worden vervolgens toegevoegd aan de leerlinggegevens. Hieronder wordt met een aantal algemene stukken syntax gedemonstreerd hoe dat kan worden gedaan. Om de leerlinggegevens in te lezen wordt bij SHEET de naam van het tabblad ingevuld met deze gegevens. Wanneer het gaat om het inlezen van de toetsgegevens, moet deze naam worden aangepast. Tevens moet dan de naam van de dataset (bij DATASET NAME) worden aangepast, anders wordt het daarvoor aangemaakte bestand met leerlinggegevens overschreven. In SPSS kunnen namelijk twee geopende bestanden niet dezelfde naam hebben.

```
GET DATA
/TYPE=XLS
/FILE='F:\map\naambestand.xls'
/SHEET=name 'Leerlingen'
/CELLRANGE=full
/READNAMES=on
/ASSUMEDSTRWIDTH=32767.
EXECUTE.
DATASET NAME Leerling WINDOW=FRONT.
```

Onderstaande syntax wordt gebruikt om het format van bepaalde variabelen te wijzigen, zodat deze de kenmerken hebben die vooraf zijn gedefinieerd in de basis- dan wel hoofdbestanden. Met het commando ALTER TYPE kunnen variabelen heel eenvoudig in String (A) dan wel Numeriek (F) format worden gewijzigd.

```
ALTER TYPE Achternaam Roepnaam Tussenvoegsel (A100).
```

Alvorens de leerlinggegevens kunnen worden samengevoegd (dit heet ‘mergen’) met de toetsgegevens, moeten deze gesorteerd worden. Dit moet in beide bestanden op dezelfde manier gebeuren, anders kan er niet gemerged worden. Zoals eerder gezegd, worden de gegevens samengevoegd op de koppelvariabele ‘LRLGID’. Deze variabele sorteren we oplopend (Ascending; A), zowel in het bestand met leerlinggegevens als in het toetsbestand.

```
SORT CASES LRLGID (A).
```

Daarna wordt het bestand met toetsresultaten geopend en moet ervoor gezorgd worden dat het format van de variabele ‘LRLGID’ precies hetzelfde is als die in het bestand van de leerlinggegevens en moet deze koppelvariabele op dezelfde manier gesorteerd zijn. Deze stappen worden uitgevoerd met een syntax naar analogie van bovenstaande stukken syntax.

Zoals eerder aangegeven worden de bestanden gekoppeld op de variabele ‘LRLGID’. Deze variabele wordt in onderstaande syntax aangegeven achter BY. SPSS weet hierdoor op welke variabele de bestanden moeten worden gekoppeld. Het SPSS-bestand met toetsgegevens (Toets) wordt toegevoegd aan het SPSS-bestand met leerlinggegevens (Leerling). Door het uitvoeren van deze syntax worden de toetsgegevens achter de juiste leerling gezet.

```
DATASET ACTIVATE Leerling.  
MATCH FILES /TABLE=*  
/FILE='DataToets'  
/BY LRLGID.  
EXECUTE.
```

Let op: SPSS zal de bestanden alleen samenvoegen wanneer de koppelvariabele unieke waarden bevat; een LRLGID (de koppelvariabele) mag dus slechts één keer voorkomen in een bestand. SPSS zal een foutmelding geven bij uitvoeren van bovenstaande syntax wanneer dit niet het geval is (zie ook paragraaf 3.5).

Na uitvoeren van bovenstaande syntax is er een nieuw (samengevoegd) bestand ontstaan. Dit nieuwe bestand moet worden toegevoegd aan het gewenste hoofdbestand. Dit betekent dat de variabelen in dit nieuwe bestand hetzelfde moet zijn als het betreffende hoofdbestand, anders zal het toevoegen niet lukken. Hieronder zullen de meest voorkomende commando's worden besproken, die nodig zijn om dat bestand zo klaar te zetten dat het toegevoegd kan worden aan het betreffende hoofdbestand.

RENAME, voor het wijzigen/vervangen van de variabele naam.

```
RENAME VARIABLES (Datum=Afnamedatum).  
EXECUTE.
```

De door de scholen toegestuurde Parnassys export bevat alle toetsen van alle jaren van alle leerlingen. Eerst zullen dan ook de juiste gegevens moeten worden geselecteerd. Dat gebeurt met het commando SELECT IF. In onderstaande syntax wordt eerst de gewenste toets geselecteerd, daarna het toetsonderdeel (dan wel toetsonderdelen) en dan de 'Afnamedatum' (die binnen een bepaalde, door ons aangegeven periode, moet liggen).

```
SELECT IF (Naambestand= 'AVI 2009').  
SELECT IF (ANY(Toetsonderdeel, 'AVI-Beheersing')).  
SELECT IF (Afnamedatum >= DATE.DMY(1,10,2015) & Afnamedatum  
<= DATE.DMY(31,10,2015)).  
EXECUTE.
```

In de aangeleverde export staan vaak meer variabelen dan voor het onderzoek nodig zijn. Met onderstaand commando, MATCH FILES in combinatie met KEEP, kunnen bepaalde variabelen uit een bestand behouden worden. Het alternatief is DELETE VARIABLES, hierbij worden ongewenste variabelen verwijderd. Het voordeel van het werken met het commando MATCH FILES / KEEP (in tegenstelling tot DELETE VARIABLES) is dat meteen zichtbaar is welke variabelen behouden blijven.

```
MATCH FILES FILE = *  
/KEEP Meetmoment Naambestand Schooljaar Naamschool SCHOOLID.  
EXECUTE.
```

Naast bovenstaande commando's kan het natuurlijk nodig zijn om, afhankelijk van de keuzes die in het onderzoek zijn gemaakt, enkele variabelen toe te voegen aan dit bestand. We verwijzen voor enkele voorbeelden hiervan naar paragraaf 3.4.2.

3.4.4 Data in leerlingvolgsysteem - ESIS

Scholen die met het leerlingvolgsysteem ESIS werken kunnen hun gegevens aanleveren via een export. Evenals bij Parnassys is de uitkomst van deze export een Excelbestand. Dit Excelbestand kan verwerkt worden op de wijze die reeds in paragraaf 3.4.2 is beschreven.

3.4.5 Data in leerlingvolgsysteem – Dotcom en CITO LOVS

Scholen die met het leerlingvolgsysteem CITO LOVS of Dotcom werken kunnen hun gegevens eveneens aanleveren via een export. Om de door scholen aangeleverde CITO-exports in te kunnen lezen in SPSS moeten ze door de onderzoekers eerst uitgelezen worden in ACCESS. Vervolgens kunnen de gegevens dan in Excel worden gezet en op de wijze die reeds in paragraaf 3.4.2 is beschreven, in SPSS worden gezet.

Hiervoor heeft de onderzoeker wel toegang nodig tot de ‘monitorverzamel database’ van CITO, welke ervoor zorgt dat de gegevens van de school in ACCESS kunnen worden uitgelezen. Deze toegang is niet vanzelfsprekend en moet met CITO worden geregeld.

Er bestaat een alternatief om de gegevens van de school toch in SPSS te kunnen krijgen zonder dat daarvoor de monitor van CITO nodig is. De door de school aangeleverde export bestaat uit een zipbestand dat kan worden uitgepakt. Dit zipbestand bevat verschillende XML-bestanden en deze kunnen in Excel worden geopend. Door de onderzoeker moet worden bepaald welke XML-bestanden de voor het onderzoek relevante gegevens bevatten. Wanneer deze Excelbestanden worden opgeslagen kunnen ze, zoals reeds in paragraaf 3.4.2 is beschreven, in SPSS worden gezet.

3.4.6 Data in digitaal survey systeem

De door de respondenten ingevulde gegevens worden in een online database (in Formdesk) opgeslagen en moeten door de onderzoeker uit deze database worden gedownload en opgeslagen. In Formdesk worden diverse opties gegeven voor wat betreft het format en type bestand waarin de gegevens kunnen worden gedownload. Het eenvoudigst is te kiezen voor een Excelbestand, omdat dan de werkwijze van paragraaf 3.4.2 kan worden gevolgd om het bestand in SPSS in te lezen.

3.4.7 Data in ORISIS

Gegevens vanuit het studentvolgsysteem OSIRIS kunnen worden opgevraagd en aangeleverd via een export in Excel. Dit Excelbestand kan verwerkt worden op de wijze die reeds in paragraaf 3.4.2 is beschreven.

3.5 Databestanden samenvoegen

In eenzelfde onderzoek kan het voorkomen dat respondenten bepaalde gegevens via verschillende systemen aanleveren. Zo kunnen er scholen zijn die hun toetsgegevens aanleveren via een vooraf door de onderzoeker gedefinieerd Excelformulier en weer andere scholen leveren diezelfde toetsgegevens aan via een export uit hun leerlingvolgsysteem.

In dit geval zullen de gegevens uit deze systemen eerst apart in SPSS moeten worden ingelezen alvorens zij kunnen worden samengevoegd en in het betreffende hoofdbestand worden weggeschreven. Hoe het apart inlezen van gegevens uit verschillende systemen in SPSS moet gebeuren is reeds beschreven in paragraaf 3.4.

Om de aparte SPSS-bestanden samen te voegen, kan er gewerkt worden met het commando ADD FILES. Door dit commando kunnen SPSS-bestanden onder elkaar worden geplakt, mits ze exact dezelfde variabelen (met dezelfde variabelenkenmerken) bevatten. Wanneer de variabelen in de samen te voegen bestanden niet identiek aan elkaar zijn, zal er een foutmelding optreden en worden de bestanden niet samengevoegd. Hieronder is een voorbeeld te zien van het ADD FILES commando. Er wordt allereerst een dataset geactiveerd (DATASET ACTIVATE); in dit voorbeeld de gegevens die we uit Parnassys hebben gehaald. Aan dit bestand wordt vervolgens de gewenste dataset toegevoegd; in het voorbeeld hieronder gegevens die uit Excel zijn gehaald. De losse dataset die nu is toegevoegd (met de gegevens uit Excel), wordt gesloten.

```
DATASET ACTIVATE GegevensParnasSys.  
ADD FILES /FILE=*  
/FILE='GegevensExcel'.  
EXECUTE.  
DATASET CLOSE GegevensExcel.
```

De dataset die nu de gegevens van de twee oorspronkelijk databestanden bevat, krijgt een nieuwe naam (DATASET NAME), wordt geactiveerd (DATASET ACTIVATE) en vervolgens opgeslagen (SAVE OUTFILE).

```
DATASET NAME DataSetToets.  
DATASET ACTIVATE DataSetToets.  
SAVE OUTFILE='F:\ GegevensUit2Systemen_TA.sav'  
/COMPRESSED.
```

3.6 Samenvatting

In dit hoofdstuk is aan de orde gesteld op welke verschillende wijzen de onderzoeksdata kunnen worden verzameld. De eventuele voor- en nadelen van elke systematiek, voor zowel respondent als onderzoeker, moeten in overweging worden genomen bij het maken van een keuze hierin. Wanneer deze keuze is gemaakt en de data daadwerkelijk worden verzameld, is het belangrijk de data bij binnenkomst aan een korte visuele inspectie te onderwerpen; eventuele fouten en omissies in de aangeleverde gegevens kunnen zo direct worden opgespoord en, waar nodig, kan bij de respondent om opheldering worden gevraagd. Ongeacht de manier van dataverzameling worden de gegevens in het rekenpakket SPSS ingelezen. Afhankelijk van de manier waarop de data binnenkomen variëren de handelingen die de onderzoeker moet verrichten in SPSS. In de meeste gevallen gaat het uiteindelijk om een Excelbestand dat in SPSS moet worden ingevoerd. Wanneer respondenten in een onderzoek eenzelfde soort gegevens via verschillende systemen aanleveren, moeten deze, wanneer ze in SPSS zijn ingevoerd, worden samengevoegd met een stuk syntax.

DEEL 2

Hoofdstuk 4

DATA TOEVOEGEN AAN HOOFD- EN BASISBESTAND

4.1 Inleiding

In het voorgaande hoofdstuk is besproken op welke manieren data kunnen worden verzameld en hoe deze gegevens in SPSS kunnen worden gezet met behulp van een syntax. Nadat de gegevens zijn ingelezen en eventueel samengevoegd met andere bestanden worden deze bestanden eerst gecleaned. De datacleaning wordt beschreven in paragraaf 4.2. Hierna kunnen de gegevens worden toegevoegd aan het juiste hoofd- dan wel basisbestand (paragraaf 4.3). De handelingen voor het toevoegen van data aan het basisbestand wijken niet af van de handelingen voor het toevoegen van data aan het hoofdbestand. Daarom wordt in dit hoofdstuk alleen het toevoegen van data aan het hoofdbestand besproken. Tot slot wordt kort ingegaan op het opslaan van de bestanden (paragraaf 4.4).

4.2 Datacleaning

4.2.1 Datacleaning – Data op papier, in Excel en in Leerlingvolgsystemen

In SPSS ingelezen gegevens afkomstig uit Excel, Parnassys, ESIS, DOTCOM, CITO-LOVS en Osiris worden gecleaned op het moment dat de gegevens SPSS worden ingelezen (zie hoofdstuk 3). Extra datacleaning is meestal niet nodig. Er kan voor wat betreft deze gegevens meteen worden overgegaan tot het toevoegen van de SPSS bestanden aan het gewenste hoofd- dan wel basisbestand (paragraaf 4.3).

4.2.2 Datacleaning –Digitaal survey systeem

Gegevens die via Formdesk (of en ander digitaal survey systeem) zijn verzameld, moeten nog wel gecleaned worden. Nadat de gegevens uit het Excelbestand in SPSS zijn ingelezen (zie paragraaf 3.4.6) gaat het voor wat betreft de datacleaning om een aantal stappen die hieronder, met behulp van syntaxen, worden beschreven. Als voorbeeld worden vragenlijstgegevens van een leerkracht gebruikt.

1. Hernoemen van variabelen. Deze stap zal in veel gevallen niet nodig zijn omdat in Formdesk bij het aanmaken van de survey de variabelennamen door de onderzoeker zelf kunnen worden gekozen. Wanneer dit om enige reden niet is gebeurd, kan dit achteraf nog worden aangepast. Dit gebeurt met het commando RENAME VARIABLES:

```
RENAME VARIABLES (naam=LKR_Naam).  
EXECUTE.
```

2. Vervolgens kunnen VARIABLE LABELS en/of VALUE LABELS aan de variabelen worden toegekend. In principe zal dit niet nodig zijn, omdat dit databestand later wordt toegevoegd aan het juiste hoofdbestand waarin de variabelen al met hun (variable en/of value) labels zijn gedefinieerd.

```
VARIABLE LABELS LKR_Naam "Volledige naam van de  
leerkracht".  
EXECUTE.
```

```
VALUE LABELS  
Zelf1 Zelf2 Zelf3 Zelf4 Zelf5 Zelf6 Zelf7 Zelf8  
0 'Nee'  
1 'Ja'.
```

3. Met behulp van het commando ALTER TYPE kan het format van de variabelen worden aangepast. In principe kan in Formdesk worden aangegeven of een variabele tekst of alleen numerieke gegevens bevat. Wanneer meerdere variabelen die in het databestand achter elkaar staan in het hetzelfde format moeten worden veranderd, kan dit worden aangeven met TO. Alleen de eerste en laatste variabele worden dan benoemd.

```
ALTER TYPE Zelf1 TO Zelf8 (F2.0).
```

4. Vervolgens worden de variabelen 'Meetmoment' en 'Koppelmeetmoment' aangemaakt om de gegevens in het hoofdbestand van elkaar te kunnen scheiden.

```
COMPUTE Meetmoment = 2.  
VALUE LABELS Meetmoment 2 'januari 2016'.  
EXECUTE.
```

```
COMPUTE Koppelmeetmoment = 5.  
VALUE LABELS Koppelmeetmoment 5 'januari 2016'.  
EXECUTE.
```

5. Soms gebeurt het dat respondenten beginnen met het invullen van een survey en hier dan na het beantwoorden van enkele vragen weer mee stoppen. Vaak zal er dan voor worden gekozen om deze lijsten uit het bestand te verwijderen. In andere gevallen vult een respondent eenzelfde lijst meerdere keren in, terwijl dit niet de bedoeling was. Er zal dan een beslissing moeten worden genomen over welke data behouden worden. Het strekt tot aanbeveling in de syntax zelf (door gebruik van *) vast te leggen welke beslissingen worden genomen, zodat dit te allen tijde kan worden teruggezocht. Alle tekst die in een syntax achter een '*' wordt geschreven, wordt door SPSS niet als commando gezien en dus overgeslagen. Hiermee kunnen verhelderende opmerkingen in de syntax worden geplaatst die later kunnen worden teruggelezen.

4.3 Databestanden toevoegen aan hoofdbestand

Na de datacleaning is de volgende stap het toevoegen van deze data aan het gewenste hoofdbestand. Elk dataverzamelingsinstrument heeft, zoals eerder vermeld, een eigen hoofdbestand waarin al eerdere meetmomenten zijn opslagen. De nieuwe data moeten hieraan worden toegevoegd. Hieronder worden de te volgen stappen aangegeven voor het toevoegen van data aan een bepaald hoofdbestand. Het is van belang op te merken dat we hierbij ervan uitgaan dat het bestand dat we wensen toe te voegen aan het desbetreffende hoofdbestand, al in SPSS is geopend.

1. Ophalen van het betreffende hoofdbestand en het geven van een naam aan dit bestand in het werkgeheugen van SPSS.

```
GET FILE='F:\Leerkrachten  
\Hoofdbestand_VragenlijstLeerkracht.sav'.  
DATASET NAME HOOFDBESTAND_VRAGENLIJSTLKR.  
WINDOW=FRONT.
```

2. Om gegevens toe te kunnen voegen aan dit hoofdbestand, moeten we er eerst voor zorgen dat het hoofdbestand “actief” is (DATASET ACTIVATE). Wanneer we dit niet doen, lopen we het risico dat SPSS de gegevens aan het verkeerde bestand toevoegt. Vervolgens voegen we met ADD FILES het gewenste bestand toe aan het hoofdbestand.

```
DATASET ACTIVATE  
HOOFDBESTAND_VRAGENLIJSTLKR .  
ADD FILES /FILE=*  
/FILE= 'DataSetToeTeVoegen'.  
EXECUTE.
```

3. Nadat de syntax in stap 2 is uitgevoerd, is het belangrijk te controleren of het toe te voegen bestand daadwerkelijk is toegevoegd aan het hoofdbestand. Vaak zal SPSS een waarschuwing of foutmelding geven wanneer het gewenste bestand niet kan worden toegevoegd.

Een veel voorkomende oorzaak is dat de (formats van de) variabelen in het hoofdbestand en in het toe te voegen bestand niet overeenkomen. Als geen foutmelding wordt gegeven, en het toevoegen ogenschijnlijk lijkt te zijn gelukt, is het toch belangrijk te bekijken of de gewenste gegevens echt zijn toegevoegd. De eenvoudigste manier is visuele inspectie van het hoofdbestand; zijn de toe te voegen gegevens inderdaad te zien in het hoofdbestand? Is het hoofdbestand inderdaad vermeerderd met het aantal cases uit het toe te voegen bestand? Nog beter is het om de frequentieverdeling van de toe te voegen dataset op te vragen en te controleren of dit aantal ook in het hoofdbestand is opgenomen door de frequentieverdeling van het hoofdbestand voor én na het commando ADD FILES op te vragen. Deze controlestep is zeer belangrijk.

Op het moment dat we gegevens opnemen in het hoofdbestand moet altijd worden geverifieerd of de respondenten ('LRLGID', 'LKRID' en 'SCHOOLID') al zijn opgenomen in het basisbestand met de relevante variabelen. Als dat niet het geval is, dan moeten deze gegevens worden toegevoegd in het betreffende basisbestand. Nieuwe respondenten krijgen een nieuw en uniek ID, dat zowel in het basisbestand als in de relevante hoofdbestanden moet worden opgenomen.

4.4 Opslaan van databestanden

Nadat gegevens zijn toegevoegd aan het hoofdbestand is het uiteraard van belang dit hoofdbestand opnieuw op te slaan. Hierbij is het belangrijk het gewijzigde hoofdbestand onder de bestaande naam, op de daarvoor aangewezen plaats, op te slaan. Het is niet aan te raden wanneer er gegevens zijn toegevoegd aan het hoofdbestand, dit onder een andere naam op te slaan. Op deze manier kan worden voorkomen dat er meerdere versies van één hoofdbestand bestaan en er verwarring ontstaat over de volledigheid van gegevens in het hoofdbestand en welke versie de meest recente is. Dit geldt eveneens voor de basisbestanden.

Wanneer er op een later tijdstip analyses worden uitgevoerd die tot wijzigingen in het hoofdbestand leiden (zoals bij het hercoderen van variabelen of bij het doen van berekeningen die in het bestand worden toegevoegd), worden deze wijzigingen overigens niet opgeslagen. Wanneer een onderzoeker analyses wil uitvoeren op (delen van) het hoofdbestand, gebeurt dit via commando's in een syntax. Door alle commando's in een syntax te zetten en deze syntax op te slaan, kunnen de berekeningen altijd gereproduceerd worden. De resultaten van de berekeningen worden door SPSS in een output-bestand gezet; uiteraard kan dit output-bestand wel worden opgeslagen.

4.5 Samenvatting

In dit hoofdstuk is het toevoegen van data aan het hoofdbestand aan bod gekomen. In het voorgaande hoofdstuk waren de door de respondent aangeleverde data al in SPSS gezet. De volgende stap is de (eventuele) datacleaning. Voor gegevens uit een digitaal survey systeem zijn hier een aantal stappen in aan te geven. Na de datacleaning volgt het toevoegen van deze data aan het hoofdbestand. Bij het opslaan van bestanden moet gelet worden op het feit dat een bestand niet onder meerdere namen wordt opgeslagen om er zo voor te zorgen dat er slechts één versie van dat bestand blijft bestaan. Ook worden niet alle wijzigingen opgeslagen in de hoofdbestanden.

HOOFDSTUK 5

BEREKENINGEN MET BEHULP VAN SPSS

5.1 Inleiding

Tot dusver hebben we besproken dat voordat de onderzoeker *überhaupt* gegevens gaat verzamelen, hij of zij eerst moet nadenken en visualiseren welke variabelen met welke kenmerken, in welke databestanden moeten worden opgenomen. We hebben daarbij onderscheid gemaakt tussen hoofd- en basisbestanden. Wanneer de data zijn verzameld en deze zijn toegevoegd aan de betreffende basis- en hoofdbestanden (zoals besproken in hoofdstuk 3 en 4), kunnen de gewenste analyses worden gedaan om de onderzoeksvraag te beantwoorden. In paragraaf 5.2 worden eerst een aantal mogelijke data-analyses aangestipt. Vervolgens worden in paragraaf 5.3 aan de hand van een concreet voorbeeld, de statische gegevens uit de basisbestanden gecombineerd (samengevoegd) met de dynamische gegevens uit het hoofdbestand. Voor de analyse wordt doorgaans een nieuw bestand aangemaakt. Het hercoderen van variabelen staat centraal in paragraaf 5.4 en in paragraaf 5.5 wordt een voorbeeld van een data-analyse besproken, aan de hand van een concrete onderzoeksvraag.

5.2 Typen data-analyses

Afhankelijk van de onderzoeksvraag bepaalt de onderzoeker welke analyses moeten worden uitgevoerd en welke gegevens daarvoor nodig zijn. De onderzoeker kan zich beperken tot beschrijvende statistiek. Hierbij gaat het bijvoorbeeld om kengetallen als gemiddelden, standaarddeviaties, minimale en maximale scores en/of percentages. Wanneer het bij een onderzoeksvraag echter gaat om een verschil of samenhang tussen variabelen of groepen respondenten, dan wordt meestal een beroep gedaan op toetsende statistiek. Voorbeelden van dergelijke analyses zijn t-toetsen, (M)ANOVA, GLM, factoranalyse en (multilevel) regressie-analyse.

Al deze analyses - en nog heel veel meer - kunnen met SPSS worden uitgevoerd. Uiteraard is deze lijst van analyses niet uitputtend en het gaat ook te ver om al deze analyses in dit rapport de revue te laten passeren. Wij verwijzen naar bijlage 1 voor een lijst met statistiek- en SPSS-boeken. In de volgende paragrafen zullen we ons beperken tot paar voorbeelden van eenvoudige toetsende statistiek.

5.3 Samenvoegen van benodigde gegevens voor analyses

Soms is het voor het beantwoorden van de onderzoeksvraag nodig dat gegevens uit meerdere databestanden worden gehaald. Als we bijvoorbeeld willen weten of vrouwen en mannen op meetmoment 1 verschillend scoren op een bepaald deel van een leerkrachtvragenlijst (bijvoorbeeld 'vertrouwen in eigen kunnen'), dan hebben we niet genoeg aan de gegevens in het hoofdbestand met vragenlijstgegevens. Het geslacht van de respondenten is immers een statische variabele en opgenomen in het basisbestand (zie hoofdstuk 1).

In beide bestanden bevinden zich dus gegevens van leerkrachten en die gegevens moeten we combineren (samenvoegen) in één databestand zodat we de onderzoeksvraag kunnen beantwoorden. In paragraaf 3.5 hebben we het commando ADD FILES besproken; hiermee kunnen gegevens uit twee bestanden die dezelfde variabelen bevatten onder elkaar worden gezet. Daar gaat het in dit geval niet om. Wij willen gegevens uit twee bestanden *naast* in plaats van onder elkaar zetten; dit doen we met het commando MATCH FILES.

Dit kan op drie verschillende manieren:

- FILE – FILE (dit is de standaardinstelling van SPSS): de cases van beide databestanden (files) worden samengevoegd. Dit kan alleen wanneer de koppelvariabele (het ID) in beide bestanden slechts één keer voorkomt.
- FILE – TABLE: de koppelvariabele (ID) komt meerdere keren voor in het ene bestand (file), en slechts één keer in het andere bestand. Aan de actieve dataset, worden nu de variabelen uit het andere bestand toegevoegd, ook al komt de ID meerdere keren voor. Ook worden uit het toe te voegen bestand alleen gegevens toegevoegd van ID's die in het actieve bestand zitten.

Als het toe te voegen bestand nog andere ID's bevat die niet in het actieve bestand voorkomen, dan worden deze gegevens niet toegevoegd. Deze manier van koppelen wordt vooral gebruikt als men gegevens uit een basisbestand (bijvoorbeeld de geslacht van de leerling) wil toevoegen aan gegevens uit een hoofdbestand (bijvoorbeeld aan toetsgegevens van leerlingen op verschillende meetmomenten). Gegevens van leerkrachten uit het basisbestand waarvan geen gegevens in het hoofdbestand staan, zijn immers overbodig.

- TABLE – FILE: slechts één case per ID in het ene bestand, meerdere cases per ID in het andere bestand.

Het verdient de voorkeur om, alvorens de bestanden samengevoegd worden, eerst een selectie te maken van de benodigde gegevens in beide bestanden. Wanneer het voor de onderzoeksvraag enkel nodig is naar één bepaald meetmoment te kijken, selecteren we dit meetmoment alvorens we gaan samenvoegen. Dit doen we, omdat de kans op fouten bij het samenvoegen van (heel) grote bestanden vele malen groter is. Ook is het dan heel moeilijk om te controleren of het samenvoegen foutloos is gegaan.

Wij richten ons vanaf nu op de voorbeeld onderzoeksvraag die wij aan het begin van deze paragraaf hebben genoemd: 'Scoren vrouwen en mannen op meetmoment 1 verschillend op een bepaald deel (vertrouwen in eigen kunnen) van een leerkrachtvragenlijst?'.

Om het hoofdbestand met vragenlijstgegevens samen te voegen met de gegevens uit het basisbestand leerkrachten moeten de volgende stappen doorlopen worden:

1. Openen van het juiste hoofdbestand in SPSS (GET FILE) en toewijzen van een bestandsnaam (DATASET NAME). De syntax hiervoor ziet er als volgt uit:

```
GET FILE='F:\Leerkrachten\  
Hoofdbestand_VragenlijstLeerkracht.sav.  
DATASET NAME HOOFDBESTAND_VRAGENLIJSTLKR.
```

2. Als de vragenlijstgegevens op meerdere momenten zijn verzameld, moeten we er eerst nog voor zorgen dat we het juiste meetmoment ('Meetmoment' 1) hebben geselecteerd. Dit kan met het commando SELECT IF.

```
DATASET ACTIVATE  
HOOFDBESTAND_VRAGENLIJSTLKR.  
SELECT IF Meetmoment = 1.  
EXECUTE.
```

3. We moeten dit hoofdbestand, dat nu alleen nog gegevens van 'Meetmoment' 1 bevat, samenvoegen (mergen) met het basisbestand leerkrachten, omdat zich daarin de variabele 'LKR_sekse' (geslacht) bevindt. We openen daarom ook het basisbestand leerkrachten en geven dat bestand ook een naam om mee te werken.

```
GET FILE='F:\Basisbestanden\LEERKRACHTBESTAND.sav'.  
DATASET NAME LEERKRACHT.
```

4. Voordat we de bestanden kunnen samenvoegen, moeten we een variabele aanwijzen waarmee we de koppeling gaan maken, een koppelvariabele. Dit is in dit geval het 'LKRID'. Door hierop te koppelen worden straks de gegevens van een leerkracht uit het basisbestand (inclusief 'LKR_sekse') achter zijn of haar vragenlijstgegevens geplaatst. Het samenvoegen zal echter alleen lukken als de gegevens in beide bestanden gesorteerd zijn (SORT CASES) op de koppelvariabele, en wel oplopend (ascending (A)).

```
DATASET ACTIVATE  
HOOFDBESTAND_VRAGENLIJSTLKR.  
SORT CASES LKRID (A).  
EXECUTE.
```

```
DATASET ACTIVATE LEERKRACHT.  
SORT CASES LKRID (A).  
EXECUTE.
```

5. Aan het hoofdbestand met vragenlijstgegevens willen we de gegevens uit het basisbestand leerkrachten toevoegen. We activeren het bestand waaraan we de gegevens willen toevoegen; het hoofdbestand met daarin de vragenlijstgegevens. Vervolgens willen we daar het basisbestand met leerkrachtgegevens aan toevoegen. Achter BY geven we aan op welke variabele SPSS de gegevens uit beide bestanden moet koppelen ('LKRID').

```
DATASET ACTIVATE HOOFDBESTAND_  
VRAGENLIJSTLKR.  
MATCH FILES /FILE=*  
/TABLE='LEERKRACHT'  
/BY LKRID  
EXECUTE.
```

6. Het is aan te bevelen na het uitvoeren van dit commando te controleren of het samenvoegen daadwerkelijk goed is gegaan. Dit kan door een visuele inspectie van het bestand. Nog beter is het om de frequentieverdeling van de variabelen van de toe te voegen dataset op te vragen, voor én na het commando MATCH FILES, en te controleren of deze ook op dezelfde manier (bijvoorbeeld zonder vreemde vermenigvuldigingen, staan de variabelen op de juiste plek) in het hoofdbestand zijn opgenomen. Deze controlestep is zeer belangrijk.
7. Het basisbestand leerkrachten sluiten we, dit hebben we niet meer nodig, omdat de gegevens hieruit zijn toegevoegd aan het hoofdbestand.

```
DATASET CLOSE LEERKRACHT.
```

8. Het nieuwe, gewijzigde hoofdbestand wordt niet opgeslagen (zie ook paragraaf 4.4). Er kan voor gekozen worden het bestand onder een andere naam op te slaan, maar in principe hoeft dit niet. Doordat alle stappen in een syntax worden gezet en deze syntax wordt opgeslagen, kunnen de analyses en daarbij behorende stappen te allen tijde herhaald worden.

Wel geven we het actieve bestand een andere naam in het werkgeheugen van SPSS.

`DATASET NAME ANALYSES_ LEERKRACHT.`

Met bovenstaande stappen hebben we nu een databestand klaarstaan waarmee we de berekeningen kunnen uitvoeren die nodig zijn om de onderzoeksvraag te beantwoorden.

5.4 Hercoderen van variabelen

Bij het afnemen van surveys komt het vaak voor dat zowel positief als negatief geformuleerde items worden gebruikt. Dit betekent dan ook dat de scoring bij deze items niet gelijk is (bijvoorbeeld bij positief geformuleerde items: 0=nee, 1=ja; bij negatief geformuleerde items: 1=nee; 0=ja). De scores op deze items kunnen niet zomaar bij elkaar worden opgeteld. Wanneer we een schaalscore willen berekenen over een set items waarvan enkele items negatief geformuleerd zijn, moeten we ervoor zorgen dat alle items op dezelfde manier gecodeerd zijn. Dit gaat in SPSS via het commando `RECODE`. Hierbij kan worden gekozen voor twee opties; `RECODE INTO DIFFERENT VARIABLES` (hierbij worden voor de te hercoderen variabelen nieuwe variabelen aangemaakt waarin de hergecodeerde score wordt weggeschreven; de originele score blijft dus behouden in de originele variabele) en `RECODE INTO SAME VARIABLES` (hierbij worden de originele scores overschreven door de hergecodeerde scores). In de meeste gevallen kiezen we `RECODE INTO DIFFERENT VARIABLES`, zelfs al slaan we de wijzigingen niet op in het originele hoofdbestand. Door met nieuwe variabelen te werken kunnen we gemakkelijk nagaan of het hercoderen goed is verlopen, een fout hierbij is namelijk zo gemaakt. Hieronder geven we een voorbeeld van het hercoderen op beide manieren. Het betreft items die samen het 'vertrouwen in eigen kunnen' van de leerkracht meten. Het gaat om 8 items, waarvan item 'Zelf3' en item 'Zelf8' negatief geformuleerd zijn.

RECODE INTO DIFFERENT VARIABLES:

```
RECODE Zelf3 Zelf8 (0=3) (1=2) (2=1) (3=0) INTO Zelf3rec Zelf8rec.  
EXECUTE.
```

RECODE INTO SAME VARIABLES:

```
RECODE Zelf3 Zelf8 (0=3) (1=2) (2=1) (3=0).  
EXECUTE.
```

Na het (eventuele) hercoderen, is het bestand klaar om te gebruiken voor het beantwoorden van de onderzoeksvraag.

5.5 Data-analyse

In deze paragraaf laten we allereerst zien welke berekeningen moeten worden uitgevoerd om onze onderzoeksvraag ('Scoren mannen en vrouwen op meetmoment 1 verschillend op het onderdeel 'vertrouwen in eigen kunnen' van een leerkrachtvragenlijst?') te beantwoorden.

Om de onderzoeksvraag te beantwoorden moeten we de volgende stappen ondernemen:

1. Aanmaken van de schaalscore 'vertrouwen in eigen kunnen'. Het gaat hier om een gemiddelde score per leraar. Aanmaken van een nieuwe variabele gaat in SPSS in dit geval met het commando COMPUTE. Er kan op verschillende manieren worden omgegaan met het berekenen van een gemiddelde score. Als men eerder heeft gekozen voor RECODE INTO DIFFERENT VARIABLES:

```
COMPUTE Zelf_schaal=MEAN(Zelf1, Zelf2, Zelf3rec, Zelf4,  
Zelf5, Zelf6, Zelf7, Zelf8rec).  
EXECUTE.
```


Of als men heeft gekozen voor RECODE INTO SAME VARIABLES:

```
COMPUTE Zelf_schaal=MEAN(Zelf1 TO Zelf8).  
EXECUTE.
```

In bovenstaand voorbeeld wordt er gewerkt met het subcommando MEAN. Dit betekent dat er een gemiddelde wordt uitgerekend over de beschikbare scores binnen de items 'Zelf1' tot en met 'Zelf8'. Wanneer er echter sprake is van ontbrekende scores, doordat een leerkracht slechts een deel van de vragen heeft ingevuld, zal door SPSS een gemiddelde worden berekend over de beschikbare scores. Dit zorgt ervoor dat schaalscores tussen leerkrachten onvergelykbaar worden (bijvoorbeeld: bij de ene leerkracht gaat het om een schaalscore over 6 items en bij de andere leerkracht gaat het om een schaalscore over 8 items). Daarom geniet het de voorkeur met de volgende optie te werken:

```
COMPUTE Zelf_schaal= (Zelf1 + Zelf2 + Zelf3rec + Zelf4 +  
Zelf5 + Zelf6 + Zelf 7 + Zelf8rec) / 8.  
EXECUTE.
```

In bovenstaand alternatief worden de scores bij elkaar opgeteld en dan gedeeld door het aantal items dat de schaal bevat. Wanneer er nu sprake is van ontbrekende scores zal er geen schaalscore worden berekend. Bij leerkrachten die slechts een deel van de vragen hebben ingevuld, zal de te berekenen variabele 'Zelf_schaal' dan ook leeg (SYSTEM MISSING) blijven.

2. Nadat we voor alle leerkrachten de score op 'vertrouwen in eigen kunnen' hebben berekend, kunnen we gaan bekijken of vrouwen en mannen inderdaad verschillende scores. Dit doen we door eerst de vrouwen te selecteren en dan hun gemiddelde score te berekenen op de schaal 'vertrouwen in eigen kunnen'.

Het selecteren van de vrouwen in het bestand gaat als volgt. We maken een filter aan waardoor alleen respondenten die voldoen aan 'LKR_sekse=1' (vrouw) worden geselecteerd. In alle berekeningen die we hierop volgend doen, zullen *alleen* de geselecteerde respondenten worden meegenomen.

```
DATASET ACTIVATE ANALYSES_LEERKRACHT.  
USE ALL.  
COMPUTE filter_$=(LKR_sekse = 1).  
VARIABLE LABELS filter_$ LKR_sekse = 1 (FILTER).  
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.  
FORMATS filter_$ (f1.0).  
FILTER BY filter_$.  
EXECUTE.
```

3. Nu we de juiste respondenten (vrouwen) hebben geselecteerd, kunnen we bekijken wat hun gemiddelde score is. Dit doen we middels het commando FREQUENCIES. Hieronder wordt dit commando weergegeven. Achter de regel STATISTICS kan worden aangegeven in welke maten we geïnteresseerd zijn. Wij zijn nu alleen geïnteresseerd in het gemiddelde (MEAN) en ook de spreiding (STDDEV, standaarddeviatie), maar ook de minimale score, maximale score, range, variantie en enkele andere maten zijn op te vragen.

```
FREQUENCIES VARIABLES=Zelf_schaal  
/STATISTICS=MEAN STDDEV  
/ORDER=ANALYSIS.
```

SPSS zal de door ons opgevraagde resultaten nu in een output-bestand zetten.

4. We moeten de selectie die we in stap 2 hebben gemaakt uitzetten, zodat we daarna een nieuwe selectie van respondenten (mannen) kunnen maken.

```
FILTER OFF.  
USE ALL.  
EXECUTE.
```

5. Vervolgens kunnen stap 2, 3 en 4 worden herhaald om de gemiddelde schaalscore voor de mannen te berekenen. In stap 2 zal dan het selectie criterium wijzigen, namelijk 'LKR_sekse = 2'.

Na het doorlopen van deze stappen kunnen de gemiddelde scores van vrouwen en mannen met elkaar worden vergeleken.

Het berekenen van de gemiddelde score voor zowel de vrouwen als de mannen kan ook in een keer. We werken dan niet met een filter, maar met het commando SPLIT FILE. Vervolgens vragen we met FREQUENCIES het gemiddelde en de standaarddeviatie van de variabele 'Zelf_schaal' op:

```
SORT CASES BY LKR_sekse.  
SPLIT FILE LAYERED BY LKR_sekse.  
  
FREQUENCIES VARIABLES=Zelf_schaal  
/STATISTICS=MEAN STDDEV  
/ORDER=ANALYSIS.
```

Met SPLIT FILE OFF schakelen we de zojuist gemaakte indeling van de output naar sekse weer uit.

In het volgende hoofdstuk wordt beschreven hoe deze gegevens op eenvoudige wijze in een tabel kunnen worden gezet.

5.6 Samenvatting

In hoofdstuk 5 is besproken hoe met behulp van data-analyse een onderzoeksvraag kan worden beantwoord. Het is hierbij van belang na te gaan welke gegevens nodig zijn om de onderzoeksvraag te kunnen beantwoorden. Aan de hand van een concrete onderzoeksvraag zijn de stappen aangegeven die gezet moeten worden om de vraag met behulp van data-analyse te beantwoorden. Zo is besproken hoe (delen van) bestanden samen moeten worden gevoegd, hoe variabelen kunnen worden gehercodeerd, en hoe uiteindelijk enkele basale analyses kunnen worden gedaan.

Hoofdstuk 6

RESULTATEN RAPPORTEREN IN TABELLEN

6.1 Inleiding

In dit hoofdstuk gaan we bekijken hoe de resultaten van analyses geautomatiseerd in tabellen kunnen worden gezet voor een rapportage. In paragraaf 6.2 bespreken we hoe de resultaten van analyses direct in een nieuw databestand kunnen worden weggeschreven. Via de commando's OMS en CTABLES wordt de output van de berekeningen in een tabel gezet en deze tabel wordt als databestand opgeslagen. Om op een later moment geen handmatige wijzigingen meer te hoeven doen aan de gegevens, schonen we deze eerst nog in SPSS op en schrijven deze gegevens pas daarna weg in Excel (paragraaf 6.3). In paragraaf 6.4 wordt ten slotte stap voor stap uitgelegd hoe een tabel voor rapportage automatisch kan worden gevuld met de gegevens die eerder vanuit SPSS in Excel zijn opgeslagen.

6.2 Resultaten van analyses wegschrijven in een nieuw databestand

We gaan ook nu weer uit van de onderzoeksvraag 'Scoren vrouwen en mannen op meetmoment 1 verschillend op het onderdeel 'vertrouwen in eigen kunnen' van een leerkrachtvragenlijst?'. We rapporteren daartoe het gemiddelde en standaarddeviatie van zowel mannen als vrouwen op deze schaal. In SPSS kunnen we deze gegevens berekenen en meteen wegschrijven naar een nieuw databestand waarin de door ons gewenste kengetallen zijn opgenomen. De manier om dit te doen is via het commando OMS, dit staat voor OUTPUT MANAGEMENT SYSTEM; hiermee kan output van de resultaten in een databestand worden gezet. Binnen dit OMS commando maken we gebruik van een aanvullend commando, namelijk CTABLES (Custom Tables, te vinden in het SPSS menu onder ANALYZE – CUSTOM TABLES).

Hiermee kunnen we een tabel maken met daarin de gegevens waarover we willen rapporteren. Deze tabel schrijven we dan met behulp van het OMS commando weg naar een nieuw SPSS-bestand. De combinatie van deze twee commando's ziet er, voor onze onderzoeksvraag, als volgt uit:

```
DATASET ACTIVATE ANALYSES_LEERKRACHT.
```

```
OMS
```

```
/SELECT TABLES
```

```
/IF COMMANDS=['CTables'] SUBTYPES=['Custom Table']
```

```
/DESTINATION FORMAT=SAV
```

```
OUTFILE='F:\Tabel_onderzoeksvraag.sav'.
```

```
CTABLES
```

```
/VLABELS VARIABLES= LKR_sekse Zelf_schaal DISPLAY=LABEL
```

```
/TABLE Zelf_schaal [S]/[MEAN 'zelfschaal_mean' F5.2, STDDEV
```

```
'zelfschaal_sd' F5.2, VALIDN 'zelfschaal_N' F40.0] BY LKR_sekse
```

```
/CATEGORIES VARIABLES=LKR_sekse ORDER=A KEY=VALUE
```

```
EMPTY=EXCLUDE.
```

```
OMSEND.
```

In bovenstaande syntax zijn enkele woorden schuingedrukt. Dit zijn de zaken die we zelf moeten invullen en dus per onderzoeksvraag veranderen. Achter VLABELS VARIABLES plaatsen we alle variabelen waarover we willen rapporteren, in dit geval het geslacht van de leerkracht ('LKR_sekse') en de schaalscore op 'vertrouwen in eigen kunnen' ('Zelf_schaal'). Op de volgende regel zien we TABLE ... BY ...; hier geven we aan welke resultaten we precies willen rapporteren, in dit geval gemiddelde, standaarddeviatie en grootte van de respondentengroep voor de variabele 'Zelf_schaal', met bijbehorende naam en kenmerken. En vervolgens geven we achter BY aan op welke variabele we de resultaten willen uitsplitsen, namelijk het geslacht van de leerkracht ('LKR_sekse').

Achter CATEGORIES VARIABLES geven we aan welke de variabele is die de verschillende categorieën aangeeft waarop we de resultaten uitsplitsen, wederom het geslacht van de leerkracht ('LKR_sekse').

Nu kunnen we de originele dataset sluiten (DATASET CLOSE) en het bestand openen (GET FILE) waarin we zojuist met behulp van het OMS commando onze tabel met resultaten hebben weggeschreven.

```
DATASET CLOSE ANALYSES_LEERKRACHT.
```

```
GET FILE='F:\ Tabel_onderzoeksvraag.sav.
```

```
DATASET NAME TABEL_ONDERZOEKSVRAAG.
```

In de volgende paragraaf laten we zien hoe de gegevens die in dit databestand staan, kunnen worden opgeschoond zodat ze zonder verdere handmatige aanpassingen in de gewenste tabel in Word kunnen worden geplaatst.

6.3 Opschonen van gegevens voor rapportage

De variabelen in het zojuist gecreëerde databestand zijn automatisch aangemaakt in SPSS. En hoewel we de variabelen zelf namen en een format hebben meegegeven (in het commando CTABLES), is de wijze waarop de gegevens nu in het bestand staan, nog niet helemaal geschikt voor rapportage. Door de gegevens in SPSS op te schonen, hoeft dit later niet handmatig in de tabellen in Word nog te gebeuren.

Hiervoor worden de volgende stappen gevolgd:

1. Aanpassen van de variabelennamen. Omdat we met meerdere categorieën werken (vrouwen en mannen) worden ook de door ons gewenste resultaten voor beide groepen weergegeven. Om hierin te differentiëren, geeft SPSS de variabelennamen automatisch extensies. Deze extensies maken niet meteen duidelijk met welke gegevens we te maken hebben, daarom passen we deze aan naar eenduidigere namen.

```
RENAME VARIABLES (zelfschaal_mean=
zelfschaal_mean_vrouw).
RENAME VARIABLES (zelfschaal_sd=zelfschaal_sd_vrouw).
RENAME VARIABLES (zelfschaal_N=zelfschaal_N_vrouw).
RENAME VARIABLES (zelfschaal_mean_A=
zelfschaal_mean_man).
RENAME VARIABLES (zelfschaal_sd_A=zelfschaal_sd_man).
RENAME VARIABLES (zelfschaal_N_A=zelfschaal_N_man).
```

2. Door het commando OMS worden automatisch een aantal labels van de tabel als variabelen weggeschreven. Deze variabelen bevatten geen onderzoeksgegevens en hebben we niet nodig. We kunnen deze variabelen dan ook verwijderen met DELETE VARIABLES.

```
DELETE VARIABLES Command_Subtype_Label_Var1.
```

Als alternatief kunnen we met het commando MATCH FILES ook de variabelen noemen die we willen behouden. Achter KEEP zetten we dan de variabelen die we willen behouden.

```
MATCH FILES FILE = *
/KEEP zelfschaal_mean_vrouw zelfschaal_sd_vrouw
zelfschaal_N_vrouw zelfschaal_mean_man zelfschaal_sd_man
zelfschaal_N_man.
EXECUTE.
```


3. Wellicht een van de belangrijkste zaken bij het opschonen van de gegevens is, dat we ervoor moeten zorgen dat alle te rapporteren gegevens als tekst variabelen (STRING) in SPSS staan. Er ontstaan namelijk problemen bij het wegschrijven van numerieke waarden naar Excel. Het probleem is dat een numerieke variabele hierbij altijd al zijn decimalen behoudt, ook al zijn er slechts twee decimalen te zien. SPSS onthoudt altijd het volledige getal. Wanneer deze gegevens in Excel worden gezet, wordt dit originele getal (met zestien decimalen) meegenomen.
4. Het kost dan heel veel moeite om in Excel van al deze decimalen af te komen. Om dit te voorkomen zetten we de numerieke waarde met slechts twee decimalen 'vast' als een tekst variabele in SPSS. Dit doen we met het commando ALTER TYPE.

```
ALTER TYPE zelfschaal_mean_vrouw zelfschaal_sd_vrouw  
zelfschaal_N_vrouw zelfschaal_mean_man zelfschaal_sd_man  
zelfschaal_N_man (A8).
```

In bovenstaande syntax geeft (A8) weer dat het om een stringvariabele (tekst) gaat met een maximale lengte van 8 karakters.

5. Ontbrekende scores geven we standaard weer met een '-' in de rapportage. Hiervoor maken we voor elke variabele waarbij we, in het geval van ontbrekende scores een '-' willen weergeven, een nieuwe variabele aan (commando COMPUTE) die weergeeft hoeveel karakters een bepaalde variabele bevat. Als het aantal karakters 0 is (het veld is dus leeg), moet er op die plek een '-' worden gezet. Dit gebeurt met het commando IF.

De syntax hiervoor ziet er als volgt uit:

```
COMPUTE lenght_zelfschaal_mean_vrouw  
=CHAR.LENGTH(zelfschaal_mean_vrouw).  
IF (lenght_zelfschaal_mean_vrouw =0) zelfschaal_mean_vrouw  
='.'  
EXECUTE.
```

```
COMPUTE lenght_zelfschaal_sd_vrouw  
=CHAR.LENGTH(zelfschaal_sd_vrouw).  
IF (lenght_zelfschaal_sd_vrouw =0) zelfschaal_sd_vrouw  
='.'  
EXECUTE.
```

```
COMPUTE lenght_zelfschaal_mean_man  
=CHAR.LENGTH(zelfschaal_mean_man).  
IF (lenght_zelfschaal_mean_man =0) zelfschaal_mean_man  
='.'  
EXECUTE.
```

```
COMPUTE lenght_zelfschaal_sd_man  
=CHAR.LENGTH(zelfschaal_sd_man).  
IF (lenght_zelfschaal_sd_man =0) zelfschaal_sd_man  
='.'  
EXECUTE.
```

6. Wanneer het gaat om rapporteren van gemiddelde (M) en standaarddeviatie (SD), gaat dit altijd volgens het format 'M (SD)'. In SPSS kunnen we ervoor zorgen dat het gemiddelde en standaarddeviatie samen worden gevoegd tot één variabele in het door ons gewenste format. Dit gaat via het commando CONCAT. In onderstaande syntax wordt het door ons gewenste gemiddelde samengevoegd met de standaarddeviatie tussen ronde haken. Er wordt eerst een nieuwe tekst variabele met STRING aangemaakt.

Binnen deze syntax zorgt het commando TRIM ervoor dat alle spaties – LTRIM voor links en RTRIM voor rechts van de variabele – rondom het getal worden verwijderd. Als we dit niet zouden toevoegen, bestaat de kans dat we te maken krijgen met veel extra spaties. Verder maken we in deze syntax ook de ronde haken aan die rondom de standaarddeviatie worden gezet.

```
STRING zelfschaal_vrouw (A20).  
COMPUTE zelfschaal_vrouw=CONCAT  
(RTRIM(LTRIM(zelfschaal_mean_vrouw)), " ("  
RTRIM(LTRIM(zelfschaal_sd_vrouw)), ")).  
EXECUTE.
```

```
STRING zelfschaal_man (A20).  
COMPUTE zelfschaal_man=CONCAT  
(RTRIM(LTRIM(zelfschaal_mean_man)), " ("  
RTRIM(LTRIM(zelfschaal_sd_man)), ")).  
EXECUTE.
```

7. Nu hebben we alle variabelen gereedstaan om in de rapportage op te nemen. Als tussenstap willen we alleen de variabelen behouden in het bestand die we nog nodig hebben (hiermee worden de overbodige variabelen verwijderd). Dit gebeurt met het commando MATCH FILES. De variabelen die we niet willen behouden zijn de variabelen die we in stap 5 hebben aangemaakt (die de lengte van variabelen aangeven) en de ‘losse’ gemiddelden en standaarddeviaties (we hebben immers in stap 6 nieuwe variabelen aangemaakt waarin de gemiddelden en standaarddeviaties zijn opgenomen).

```
MATCH FILES FILE = *  
/KEEP zelfschaal_N_vrouw zelfschaal_N_man zelfschaal_vrouw  
zelfschaal_man.  
EXECUTE.
```

8. Met afronden van bovenstaande stap hebben we nu een SPSS-bestand met daarin precies de variabelen die we nodig hebben, namelijk het aantal vrouwelijke respondenten (zelfschaal_N_vrouw), het aantal mannelijke respondenten (zelfschaal_N_man), gemiddelde en standaarddeviatie van de vrouwelijke respondenten (zelfschaal_vrouw), en gemiddelde en standaarddeviatie van de mannelijke respondenten (zelfschaal_man). Nu willen we deze gegevens in een tabel voor een rapportage in Word krijgen. Hiervoor is het nodig het SPSS-bestand met de juiste gegevens eerst als Excelbestand op de slaan. Dat gaat met de volgende syntax:

```
SAVE TRANSLATE OUTFILE='F:\Tabel_onderzoeksvraag.xlsx'  
/TYPE=XLS  
/VERSION=12  
/MAP  
/REPLACE  
/FIELDNAMES  
/CELLS=VALUES  
/KEEP= zelfschaal_N_vrouw zelfschaal_N_man  
zelfschaal_vrouw zelfschaal_man.
```

Hiermee is het proces in SPSS afgerond. In de volgende paragraaf laten we zien hoe de gegevens vanuit Excel in Word kunnen worden gezet.

6.4 Vullen van de tabellen voor rapportage

Aan het eind van de voorgaande paragraaf hebben we een Excelbestand gecreëerd met daarin de scores voor zowel mannen als vrouwen op de schaal ‘vertrouwen in eigen kunnen’. Dit Excelbestand ziet er als volgt uit (tabel 6.1):

Tabel 6.1 Weergave van Excelbestand met gegevens voor rapportage			
zelfschaal_N_vrouw	zelfschaal_N_man	zelfschaal_vrouw	zelfschaal_man
5	5	2.20 (.84)	1.80 (.84)

In tabel 6.1 is te zien dat op de eerste regel de variabelennamen worden weergegeven. Op de regel daaronder worden dan de bijbehorende waarden vermeld.

Voordat nu verder kan worden gegaan, moet er allereerst voor worden gezorgd dat in Word de tabel, die moet worden gevuld met gegevens, klaarstaat.

Voor de onderzoeksgegevens uit ons voorbeeld kan deze tabel er als volgt uitzien (tabel 6.2):

Tabel 6.2 Voorbeeld rapportagetabel voor weergeven resultaten voor ‘vertrouwen in eigen kunnen’ voor mannen en vrouwen (lege tabel)		
	Vrouwen	Mannen
N		
Gem. (sd)		

Nu kunnen de volgende stappen worden gevolgd om ervoor te zorgen dat bovenstaande tabel automatisch wordt gevuld:

1. Zorg ervoor dat het Word-document met daarin de te vullen tabel openstaat. Het Excelbestand met benodigde gegevens moet zijn gesloten, anders treedt er een foutmelding op.
2. In het Word-document moet via het menu ‘Verzendlijsten’ – ‘Adressen selecteren’ – ‘Bestaande lijst gebruiken’ het benodigde Excelbestand worden opgezocht en geselecteerd.

3. Wanneer het juiste Excelbestand is geselecteerd, kunnen nu in de lege tabel de gewenste variabelen op de juiste plek in het document worden gezet. Ga hiervoor in de tabel met de cursor op de plek staan waar een variabele moet worden ingevoegd. Vervolgens kan in het menu via ‘Verzendlijsten’ – ‘Samenvoegvelden invoegen’ de juiste variabele worden aangeklikt. Deze verschijnt dan op de gekozen plek in de tabel (zie tabel 6.3). Op deze manier kunnen alle variabelennamen vanuit Excel op de juiste plek in de tabel in Word worden gezet.

Tabel 6.3 Voorbeeld rapportagetabel voor weergeven resultaten voor ‘vertrouwen in eigen kunnen’ voor mannen en vrouwen (met ingevoegde variabelennamen)

	Vrouwen	Mannen
N	<<zelfschaal_N_vrouw>>	<<zelfschaal_N_man>>
Gem. (sd)	<<zelfschaal_vrouw>>	<<zelfschaal_man>>

4. Nadat alle variabelen op de juiste plek in de tabel zijn gezet, is het van belang goed te controleren of hier geen fouten bij zijn gemaakt. Staan alle variabelennamen inderdaad op de juiste plek?
5. Nu kan ervoor worden gezorgd dat de tabel daadwerkelijk gevuld gaat worden met waarden. Hiervoor moet in het menu in het Word-document gekozen worden voor ‘Verzendlijsten’ – ‘Voltooien en samenvoegen’ – ‘Afzonderlijke documenten bewerken’. Er komt dan een dialoogvenster in beeld waarin gevraagd wordt welke gegevens moeten worden samengevoegd. Hier kiezen we voor de optie ‘Alles’, die standaard is aangevinkt. Door op ‘OK’ te drukken kan verder worden gegaan.

6. In Word wordt nu automatisch een nieuw document aangemaakt met daarin onze tabel, (uiteraard) gevuld met de juiste (fictieve) gegevens (tabel 6.4).

Tabel 6.4 Voorbeeldtabel voor weergeven resultaten voor ‘vertrouwen in eigen kunnen’ voor mannen en vrouwen (met automatisch ingevoegde gegevens)		
	Vrouwen	Mannen
N	5	5
Gem. (sd)	2.20 (.84)	1.80 (.84)

7. De tabel voor rapportage is nu gereed en kan worden opgeslagen.

6.5 Samenvatting

In dit hoofdstuk hebben we laten zien hoe resultaten van analyses grotendeels geautomatiseerd in een tabel voor een rapportage kunnen worden gezet. Via de commando's OMS (en CTABLES) kan een tabel met resultaten als databestand in SPSS worden weggeschreven. Wanneer SPSS dit bestand voor ons genereert, zijn de gegevens nog niet direct klaar om in een rapportagetabel te zetten. Dit komt deels door enkele restricties in SPSS en deels door de manier waarop wij de gegevens willen rapporteren. Via enkele commando's en stukken syntax kunnen de gegevens zo worden opgeschoond dat ze gereed zijn voor rapportage en er geen handmatige wijzigingen meer hoeven te worden aangebracht. Het opgeschoonde SPSS-bestand wordt vervolgens als Excelbestand opgeslagen en dit Excelbestand kan daarna in Word, in de door de onderzoeker gewenste tabel, worden ingevoegd.

Bijlage 1 Boeken over statistiek en SPSS

- Baarda, B., Dijkum, C. van, & Goede, M. de (2014). *Basisboek statistiek met SPSS*. Groningen: Noordhoff Uitgevers B.V.
- Baguley, T. (2012). *Serious stats. A guide to advanced statistics for the behavioral sciences*. Palgrave Macmillan.
- Bercken, J.H.L. van den, & Voeten, M.J.M. (2002). *Variantieanalyse: de GLM-benadering*. Groningen: Stenfert Kroese.
- Field, A. (2015). *Discovering statistics using SPSS*. Los Angeles: Sage.
- Field, A. & Hole, G. (2003). *How to design and report experiments*. Los Angeles/London/New Delhi/Singapore/Washington DC: Sage.
- Grotenhuis, M. te & Visscher, C. (2007). *SPSS met syntax*. Assen: Van Gorcum.
- Heck, R.H., Thomas, S.L., & Tabata, L.N. (2010). *Multilevel and longitudinal modeling with IBM SPSS*. New York/London: Routledge.
- Heck, R.H., Thomas, S.L., & Tabata, L.N. (2012). *Multilevel modeling of categorical outcomes using IBM SPSS*. New York/London: Routledge.
- Howitt, D. & Cramer, D. (2011). *Introduction to SPSS statistics in psychology. For version 19 and earlier*. Harlow (England): Pearson.
- IBM. *Programming and Data Management for IBM SPSS Statistics 23: A Guide for IBM SPSS Statistics and SAS Users*. IBM
- Lammers, J., Pelzer, B., Hendrickx, J. & Eisinga, R. (2007). *Categorische data analyse met SPSS. Inleiding in loglineaire analysetechnieken*. Assen: Van Gorcum.
- Levesque, R. & SPSS Inc. (2007). *SPSS Programming and Data Management. A Guide for SPSS and SAS Users*. SPSS Inc.

- Peet, A. van, Namesnik, K., & Hox, J. (2011). *Toegepaste Statistiek. Beschrijvende technieken*. Groningen: Noordhoff Uitgevers B.
- Peet, A. van, Namesnik, K., & Hox, J. (2012). *Toegepaste Statistiek. Inductieve technieken*. Groningen: Noordhoff Uitgevers B.V.
- Stevens, J (2009). *Applied Multivariate Statistics for the Social Sciences*. New York/Londen: Routledge.
- Tabachnick, B.G. & Fidell, L.S. (2013/2007). *Using multivariate statistics*. Boston: Pearson Education, Inc/Allyn and Bacon.
- Vocht, A. de (2015). *Basishandboek SPSS 23*. Utrecht: Bijleveld press.
- Voeten, M.J.M., & Bercken, J.H.L. van den (2003). *Lineaire regressieanalyse*. Groningen: Stenfert Kroese.