

Handleiding voor Therapy-monitoring

Deze handleiding maakt deel uit van het volgende artikel:

Van Lankveld, J., Leusink, P., & Peters, G.-J.Y. (2016) Therapie-monitoring in een blended online en face-to-face behandeling van een jonge man met situatieve erectieproblemen. Tijdschrift voor Seksuologie, 39(?).

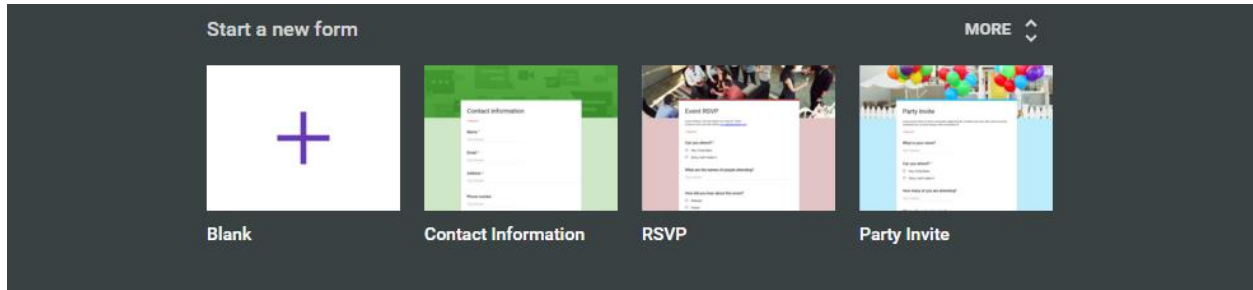
Naast de inhoudelijke informatie in dit artikel is een tweede doel om het therapeuten makkelijk te maken om zelf hun therapie te monitoren. Daarom wordt deze handleiding meegepubliceerd. De handleiding bestaat uit twee delen. In het eerste deel wordt uitgelegd hoe het gratis systeem Google Forms ingezet kan worden om data te verzamelen. In het tweede deel wordt uitgelegd hoe deze data vervolgens ingelezen en geanalyseerd kan worden met het gratis (open source) statistisch pakket R en het package Userfriendlyscience.

Deel 1: data verzamelen

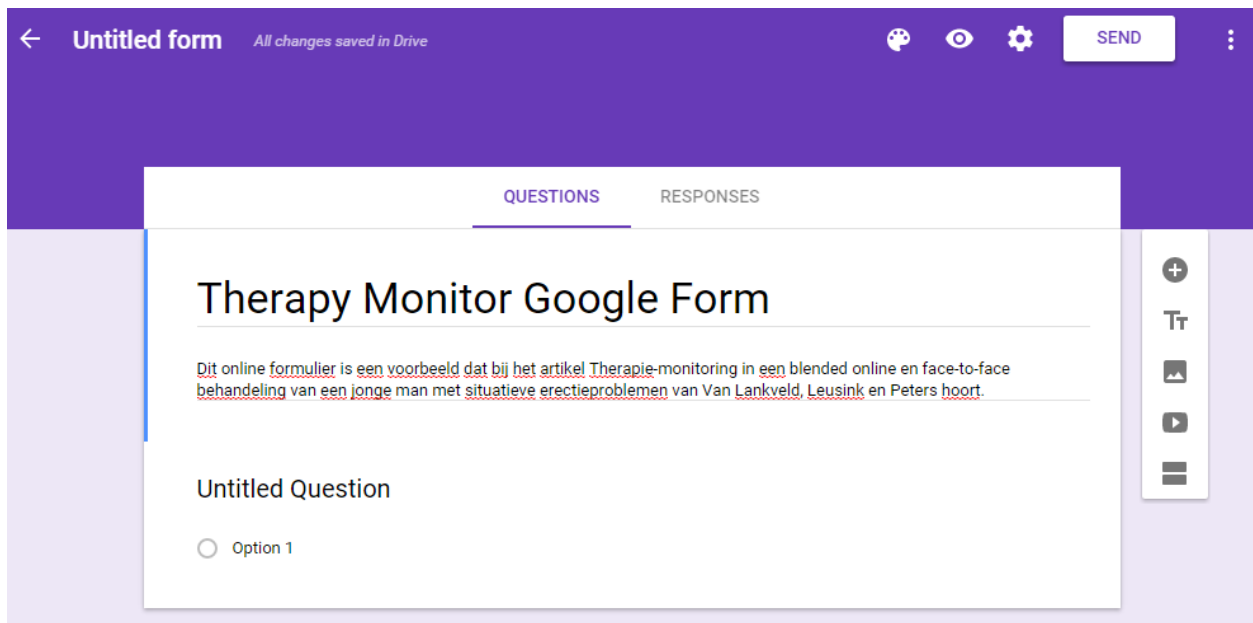
Er zijn veel softwarepakketten om online data te verzamelen. Een uiterst krachtige open source oplossing is LimeSurvey, en als de dataverzameling wat complexer is, is het aan te raden dat te gebruiken. In dit geval volstaat echter het gratis beschikbare Google Forms. Dit systeem is niet krachtig en weinig veelzijdig, maar volstaat voor eenvoudige dataverzameling, en het gebrek aan kracht en veelzijdigheid vertaalt zich naar een erg overzichtelijke en uitermate gebruikersvriendelijke interface, waardoor het zelfde voor mensen die zelden met computers werken goed te gebruiken is. Hieronder volgen stap-voor-stap instructies voor het opzetten van een 'Google Form'.

1. Ga naar <http://forms.google.com>
2. Gebruikers die niet al zijn ingelogt zullen nu eerst moeten inloggen. Gebruikers die nog geen Google Account hebben, moeten er eerst een aanmaken; dit kan door op 'Create account' of 'Account maken' te klikken. Deze tekst is zichtbaar onder het kader waar het email-adres ingegeven kan worden om in te loggen. Overigens is het niet nodig een Gmail account te hebben om een Google Account aan te maken; dit kan met elk email-adres.

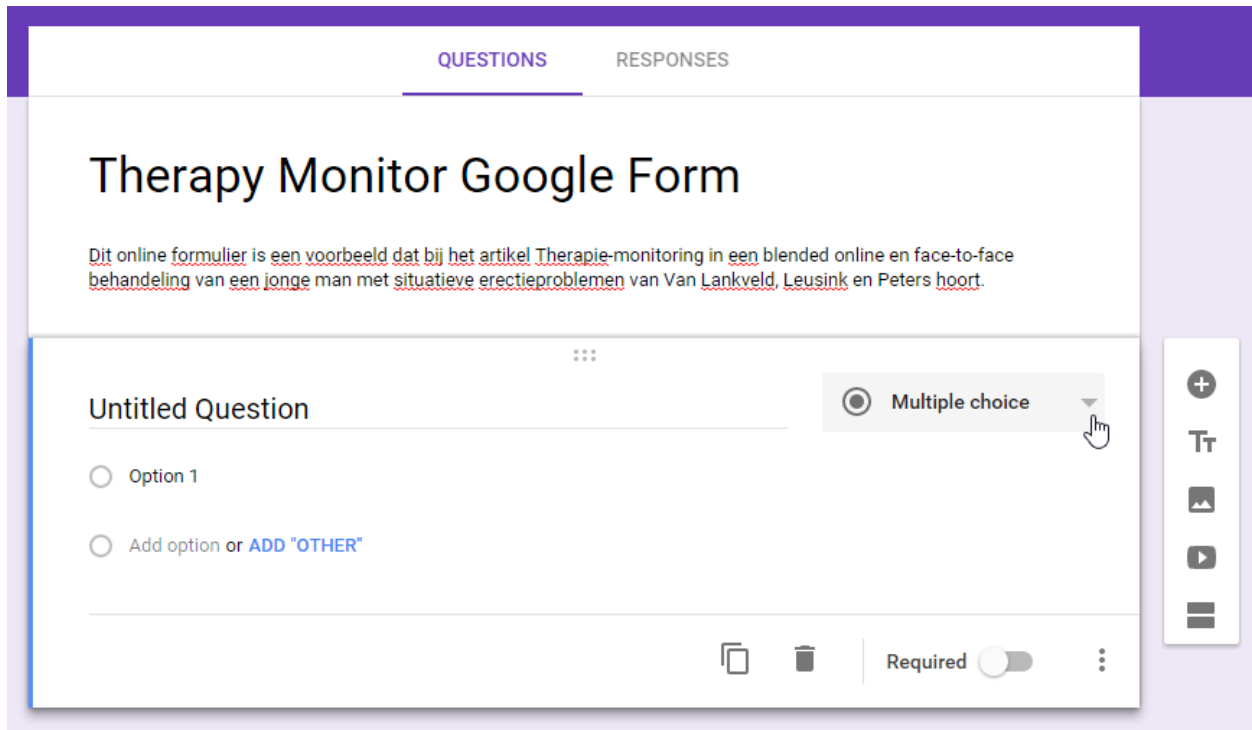
3. Eenmaal ingelogt kan een nieuw Google Form worden gemaakt door bij “Start a new form” of “Start een nieuw formulier” op de plus boven “Blank” te klikken in de balk die bovenaan in beeld staat:



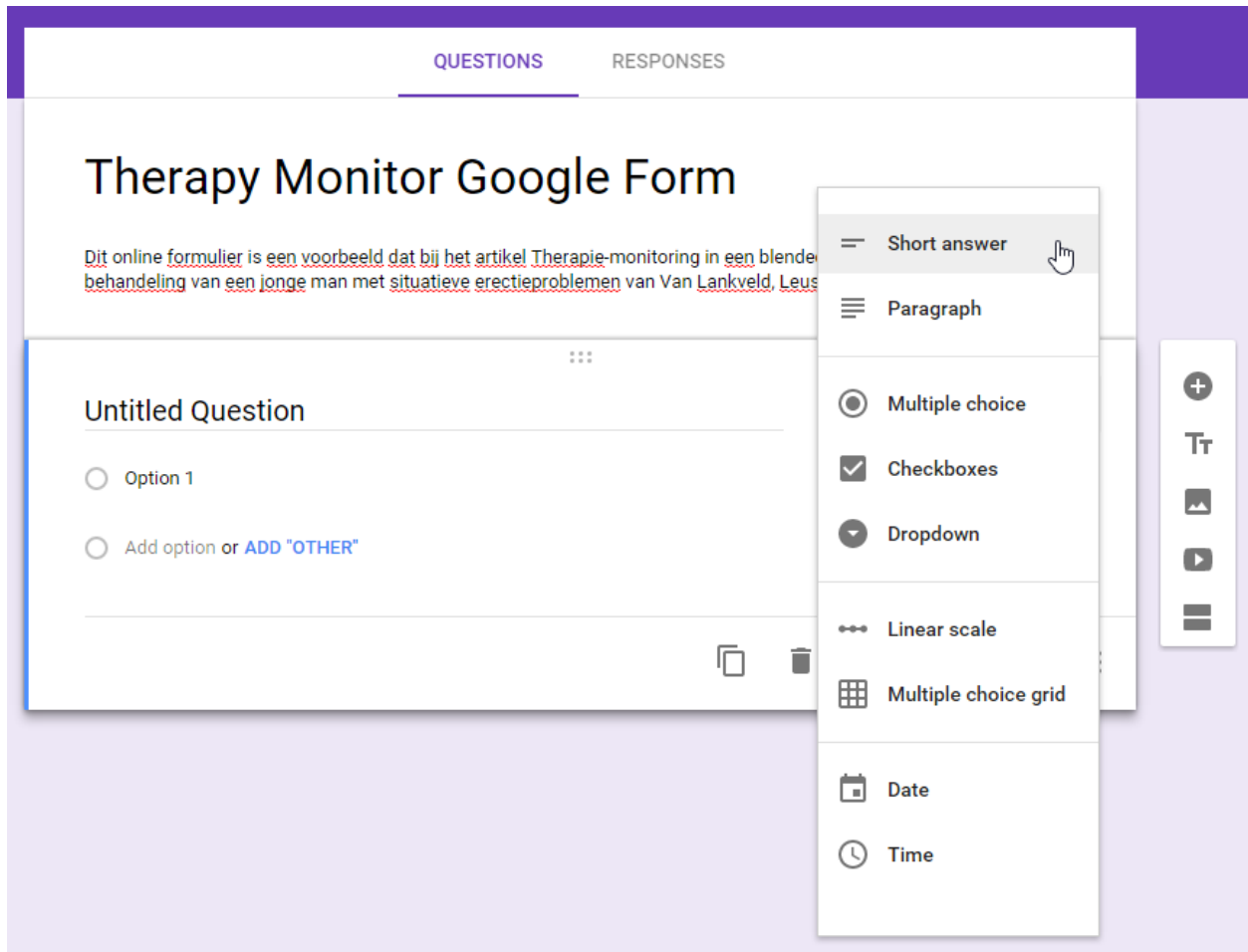
4. Google maakt nu een leeg formulier aan. Geef het formulier een titel die herkenbaar is voor uw cliënten. In dit voorbeeld gebruiken we “Therapy Monitor Google Form”.
5. Vlak onder de titel kan een inleidende tekst worden geplaatst. Geef deelnemers hier instructies voor de vragenlijst. In dit voorbeeld plaatsen we hier “Dit online formulier is een voorbeeld dat bij het artikel [Therapie-monitoring in een blended online en face-to-face behandeling van een jonge man met situatieve erectieproblemen van Van Lankveld, Leusink en Peters hoort.](#)”:



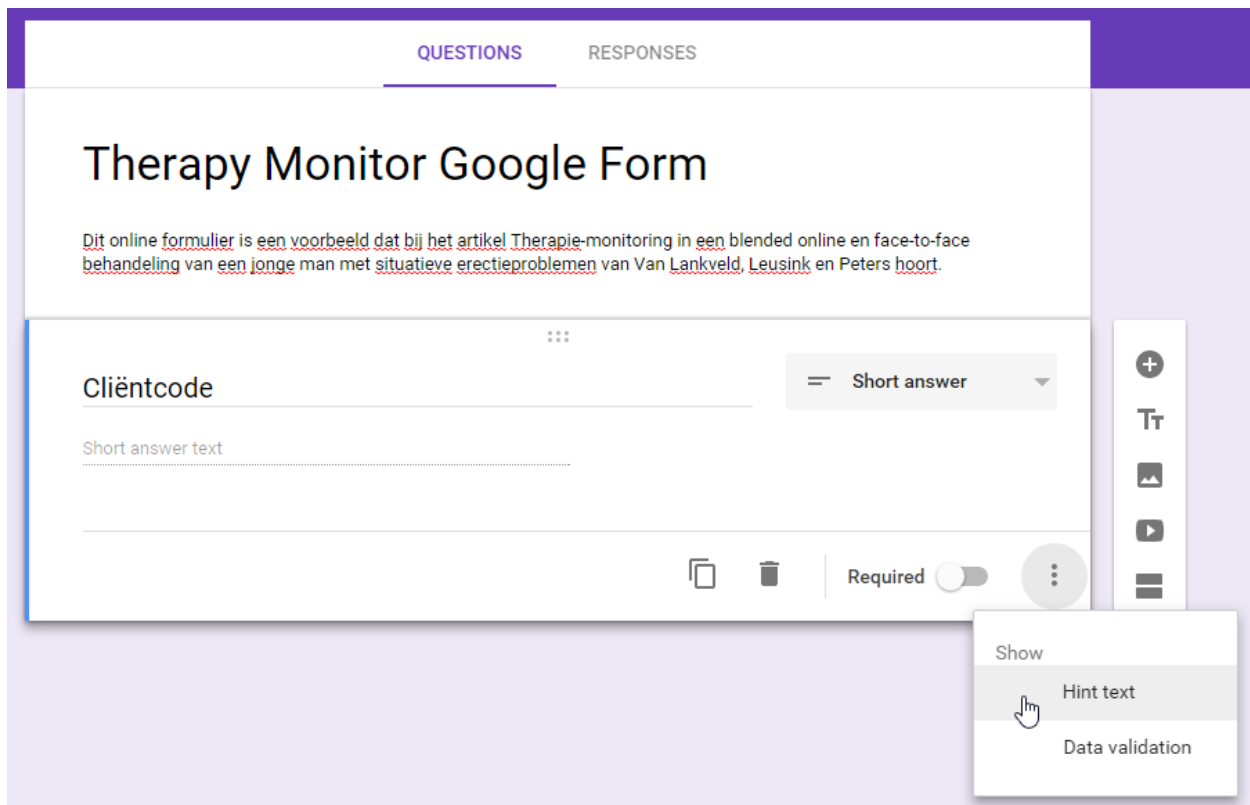
6. Nu kunnen vragen worden toegevoegd. Er zijn verschillende typen vragen die gebruikt kunnen worden. Start met een vraag van het 'tekst' type, zodat cliënten hun cliëntcode kunnen invoeren. Klik om dat te doen op de tekst 'Untitled Question' (waarschijnlijk 'Vraag zonder Titel' in het Nederlands):



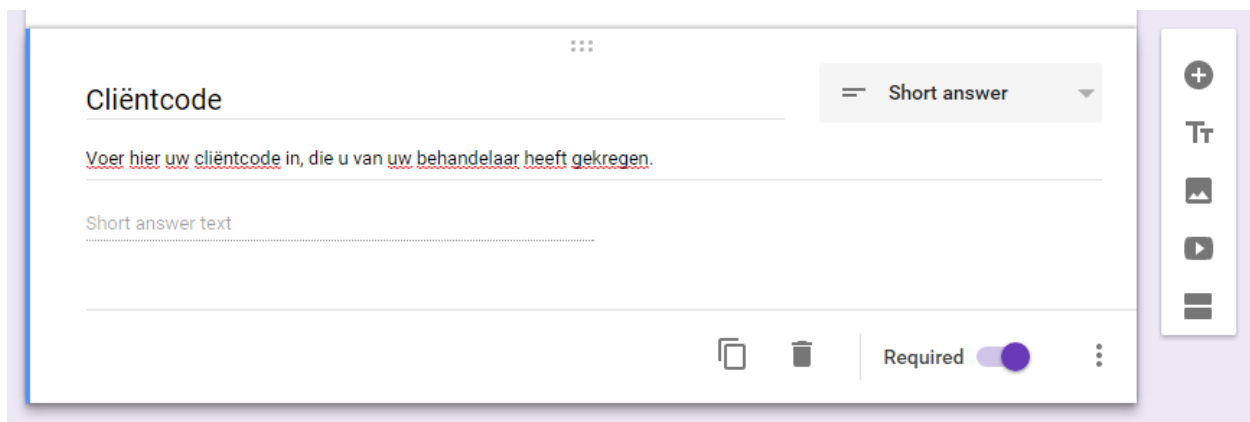
7. Klik dan op 'Multiple choice' (of 'Meerkeuze vraag') om het vraagtype te veranderen. Selecteer de bovenste optie, 'Short answer', of 'Kort antwoord':



8. Vul dan de vraag in, door weer op de standaardtekst te klikken en iets anders in te typen, bijvoorbeeld 'Cliëntcode'. Het is handig om extra uitleg toe te voegen. Klik daarom op het drie verticale puntjes die rechtsonder in het vraagkader staan, en selecteer 'Hint text':

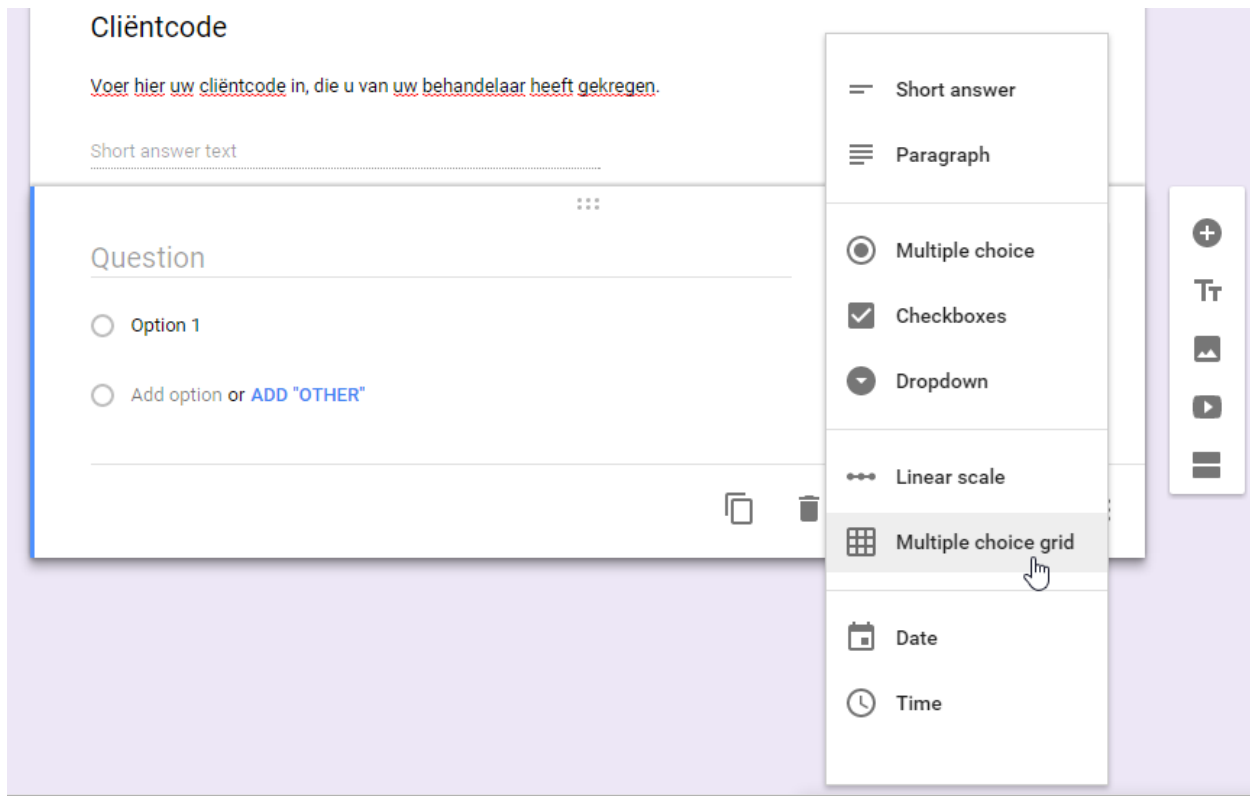


- Er verschijnt ruimte voor een 'Hint'. Voer hier een instructie in, bijvoorbeeld "Voer hier uw cliëntcode in, die u van uw behandelaar heeft gekregen."
- Verder is het goed om aan te geven dat deze vraag verplicht is. Gebruik daarvoor het schuifje bij 'Required', ook rechts onderin.
- De eerste vraag staat er nu al in. Als het goed is ziet dit er zo uit:



- Nu kunnen we starten met de vragen waarmee we echt data gaan verzamelen. Hiervoor gebruiken we vooral het Raster vraagtype. Maak nog een vraag aan

door op het plusje te klikken, dat in het kleine verticale kader aan de rechtskant staat. Kies voor die tweede vraag het 'Multiple Choice Grid' of 'Raster' type:



13. Als eerste vraag starten we met de PANAS, dus als vraagtekst voeren we “De afgelopen dagen (inclusief vandaag) voelde ik me ...” in. Als Hint voeren we informatie in over de antwoordopties: “1 = niet, 4 = matig, 7 = zeer; kies het toepasselijke antwoord voor elk onderdeel.”. Het resultaat zou er zo uit moeten zien:



⋮

De afgelopen dagen (inclusief vandaag) voelde ik me
 ...

Multiple choice grid ▾

1 = niet, 4 = matig, 7 = zeer; kies het toepasselijke antwoord voor elk onderdeel.

Row 1. Row 1	Column 1. Column 1
Row 2. Add option	Column 2. Add option



 Require one response per row
⋮

14. Nu kunnen we aangeven wat we in de rijen en kolommen willen hebben. De kolommen moeten uitsluitend nummers bevatten, in dit geval dus 1, 2, 3, 4, 5, 6, en 7. Dit is belangrijk, omdat we deze nummers straks in Deel 2 gaan analyseren. De rijen zijn de sub-vragen, in dit geval 'Opgewekt', 'Eenzaam', 'Angstig', 'Tevreden', 'Somber', 'Schuldig', 'Gelukkig', 'Ontspannen', 'Onzeker', 'Geïrriteerd', 'Enthousiast', en 'Boos'. Het resultaat ziet er zo uit:

⋮

De afgelopen dagen (inclusief vandaag) voelde ik me

...

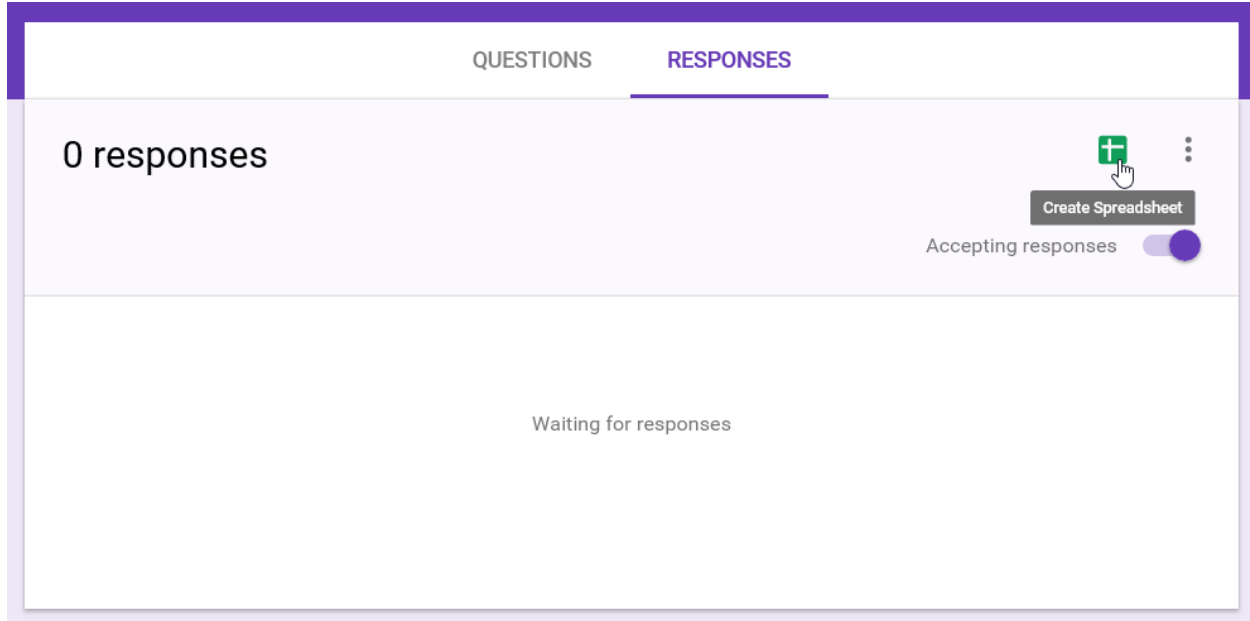
1 = niet, 4 = matig, 7 = zeer; kies het toepasselijke antwoord voor elk onderdeel.

Row 1. Opgewekt	×	Column 1. 1	×
Row 2. Eenzaam	×	Column 2. 2	×
Row 3. Angstig	×	Column 3. 3	×
Row 4. Tevreden	×	Column 4. 4	×
Row 5. Sombor	×	Column 5. 5	×
Row 6. Schuldig	×	Column 6. 6	×
Row 7. Gelukkig	×	Column 7. 7	×
Row 8. Ontspannen	×	Column 8. Add option	
Row 9. Onzeker	×		
Row 10. Geïrriteerd	×		
Row 11. Enthousiast	×		
Row 12. Boos	×		
Row 13. Add option			

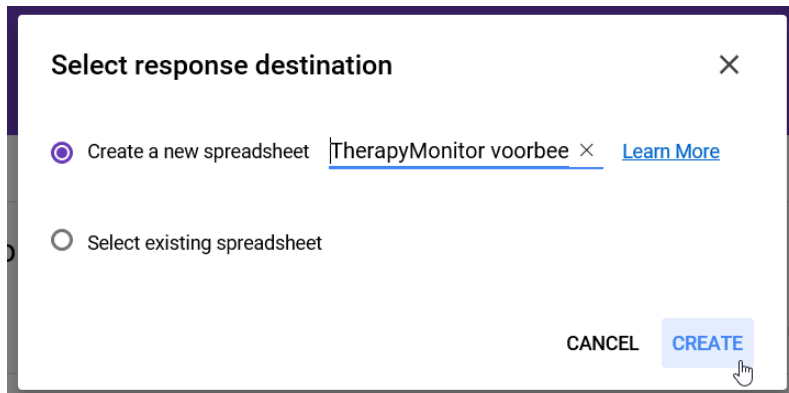
+
Tt
▲
▶
▬

15. Onderin kan de vraag weer verplicht worden gemaakt om te voorkomen dat rijen per ongeluk worden vergeten.
16. Nu kunnen meer vragen worden toegevoegd, totdat alle gewenste data is verzameld. Bovendien zijn andere vraagtypes ook mogelijk. Hiermee kan natuurlijk vrij geëxperimenteerd worden.
17. Om een vliegende start te realiseren hebben de auteurs het voorbeeld, dat is gebruikt voor deze studie, openbaar gemaakt. Dit kan worden geopend en gekopieerd naar de eigen Google account via https://docs.google.com/forms/d/1NB3R_yvGonoKO35QkvgeyeQQbFVOTQJ8MafjtDob2E0/copy.

18. Om de data toegankelijk te maken, is de link nodig naar de spreadsheet waar de data in komt te staan. Klik hiervoor op de tab 'Responses', en klik dan op het spreadsheet icoontje om een spreadsheet aan te maken:



19. Kies een naam en bevestig:



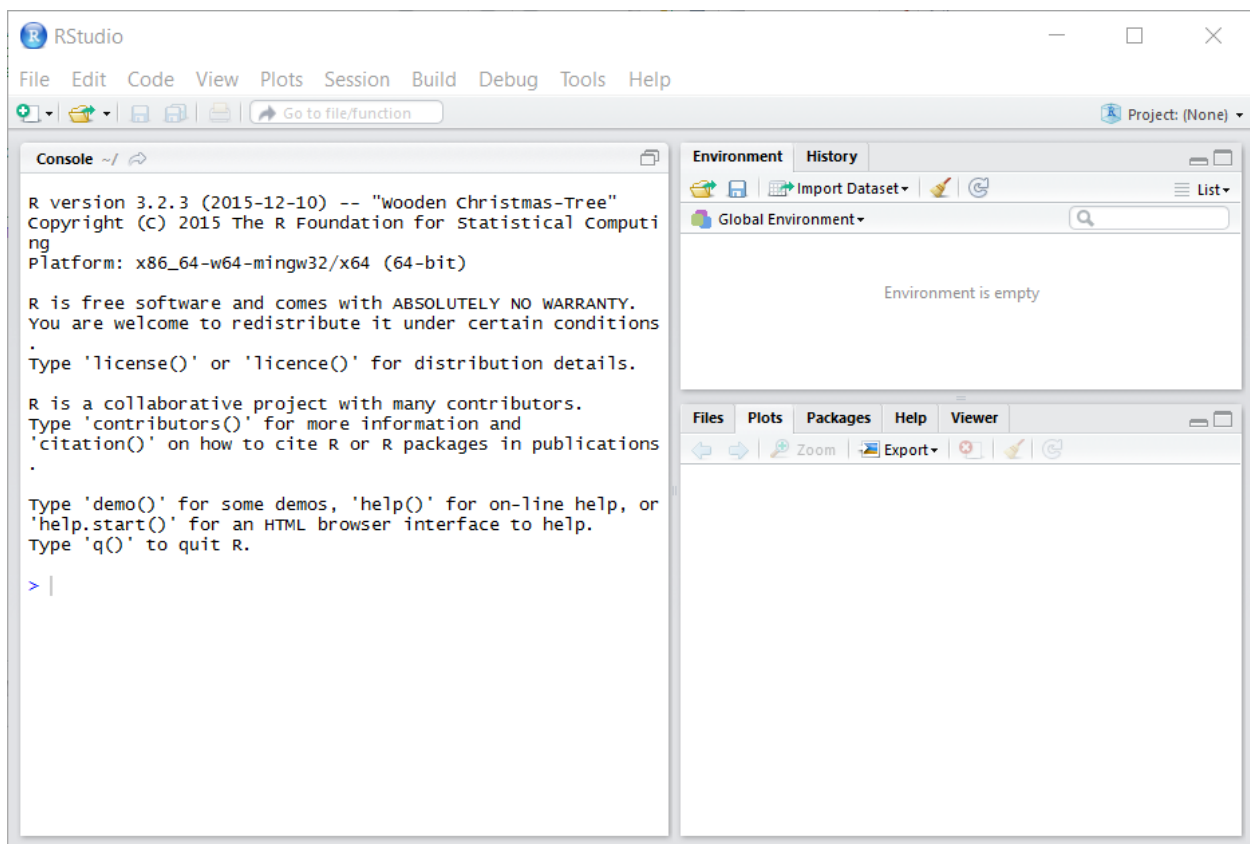
20. Kopieer vervolgens de URL van deze spreadsheet uit de adresbalk van de browser. Deze is in de volgende stap nodig om de data in te lezen voor analyse.

Deel 2: data analyseren

Om de vergaarde data te analyseren gebruiken we het open source statistische pakket R. We gebruiken package Googlesheets om de data te importeren, package httpuv om

dat iets gebruikersvriendelijker te maken, en package Userfriendlyscience voor de analyse. Bovendien gebruiken we 'R Studio om een aantal handige functies aan R toe te voegen.

1. Download R van <https://cran.r-project.org/> en installeer R.
2. Download R Studio van <https://www.rstudio.com/products/rstudio/download/> en installeer R Studio.
3. Start R Studio. Het ziet er misschien wat intimiderend uit, maar schijnt bedriegt:



4. Er kunnen commando's worden getype in het linker paneel (de 'Console'), en dit gaan we nu gebruiken om de benodigde 'packages' te installeren. Packages zijn stukjes extra functionaliteit. In dit geval zijn er twee extra packages nodig. Deze kunnen worden geïnstalleerd met de volgende twee commando's:

```
install.packages("googlesheets");  
install.packages("httpuv");  
install.packages("userfriendlyscience");
```

R gaat dan deze twee packages installeren, en bovendien alle packages die nodig zijn om deze twee packages te gebruiken (“also installing the dependencies ‘jsonlite’, ‘openssl’, ‘cellranger’, ‘httr’, ‘tidyr’, ‘XML’, ‘xml2’”).

5. Laad de packages met de volgende drie commando’s:

```
require("googlesheets");  
require("httpuv");  
require ("userfriendlyscience");
```

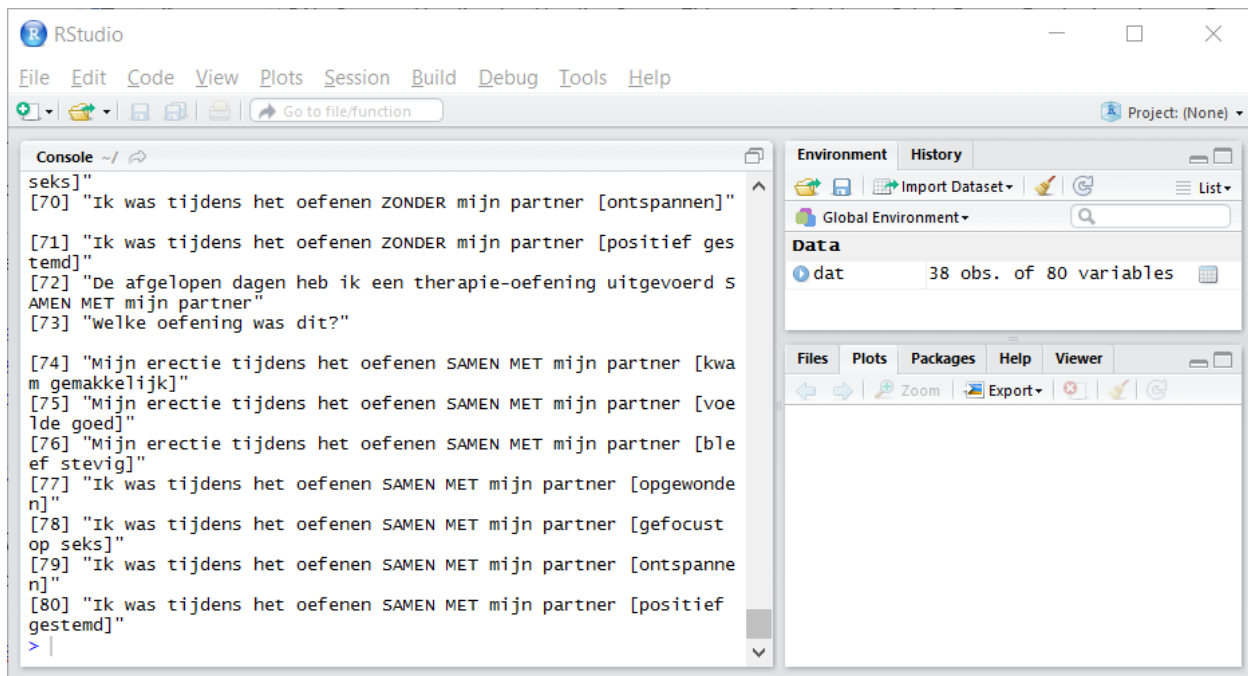
6. Nu kunnen we de data inladen. Voor dit voorbeeld gebruiken we de data die ook zijn beschreven in dit artikel. De URL om dan te gebruiken is . Gebruik bij eigen data de URL van de bewuste spreadsheet die in stap 20 van Deel 1 is gekopieerd uit de adresbalk. De laden de data in dit voorbeeld dus nog in met:

```
dat <-  
data.frame(gs_read(gs_url('https://docs.google.com/spreadsheets/d/1w1Kk  
BJF5kHbITacpCrFMtKrFm4pHBSpnG0c6qV3TsiA/edit?usp=sharing')));
```

Deze data is nu opgeslagen in een zogenaamd ‘dataframe’ dat ‘dat’ heet.

7. We kunnen nu de vragen in het Google Form, oftewel de kolommen in de spreadsheet, bekijken met het volgende commando:

```
names(dat);
```



8. Dit lijkt goed te zijn gegaan. We kunnen daarom verder met de analyse. Eerst moeten we de antwoorden op een aantal vragen omdraaien. Dit betreft specifiek de 22^{ste}, 25^{ste}, 28^{ste}, en 31^{ste} vraag:

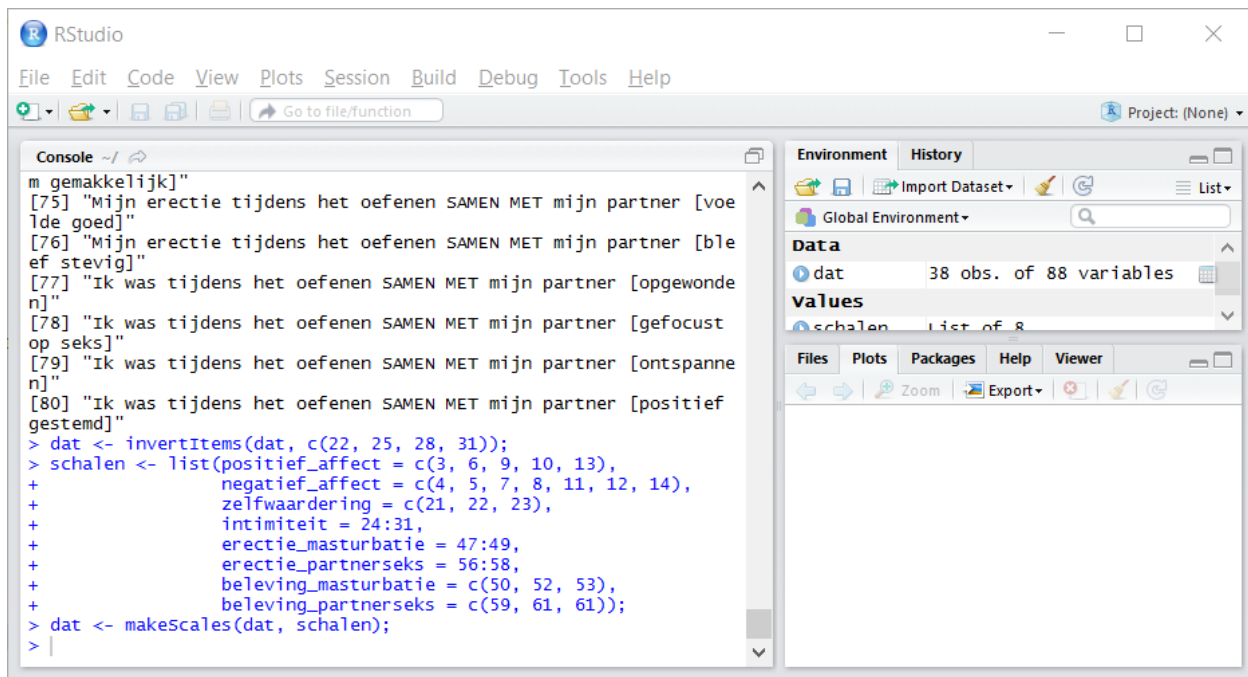
```
dat <- invertItems(dat, c(22, 25, 28, 31));
```

9. Daarna kunnen we specificeren welke vragen samen schalen vormen. Zo vormen de 3^{de}, 6^{de}, 9^{de}, 10^{de}, en 13^{de} vragen samen de schaal 'positief affect'. Deze schalen, en de vragen in elke schaal, slaan we op in een zogenaamd 'object' dat we 'schalen' noemen:

```
schalen <- list(positief_affect = c(3, 6, 9, 10, 13),
               negatief_affect = c(4, 5, 7, 8, 11, 12, 14),
               zelfwaardering = c(21, 22, 23),
               intimiteit = 24:31,
               erectie_masturbatie = 47:49,
               erectie_partnerseks = 56:58,
               beleving_masturbatie = c(50, 52, 53),
               beleving_partnerseks = c(59, 61, 61));
```

10. En dan geven we het commando om deze schalen ook echt aan te maken en toe te voegen aan de dataset:

```
dat <- makeScales(dat, schalen);
```



The screenshot shows the RStudio interface. The console on the left contains the following R code and its output:

```
m gemakkelijk]"
[75] "Mijn erectie tijdens het oefenen SAMEN MET mijn partner [voe
lde goed]"
[76] "Mijn erectie tijdens het oefenen SAMEN MET mijn partner [ble
ef stevig]"
[77] "Ik was tijdens het oefenen SAMEN MET mijn partner [opgewonde
n]"
[78] "Ik was tijdens het oefenen SAMEN MET mijn partner [gefocust
op seks]"
[79] "Ik was tijdens het oefenen SAMEN MET mijn partner [ontspanne
n]"
[80] "Ik was tijdens het oefenen SAMEN MET mijn partner [positief
gestemd]"
> dat <- invertItems(dat, c(22, 25, 28, 31));
> schalen <- list(positief_affect = c(3, 6, 9, 10, 13),
+               negatief_affect = c(4, 5, 7, 8, 11, 12, 14),
+               zelfwaardering = c(21, 22, 23),
+               intimiteit = 24:31,
+               erectie_masturbatie = 47:49,
+               erectie_partnerseks = 56:58,
+               beleving_masturbatie = c(50, 52, 53),
+               beleving_partnerseks = c(59, 61, 61));
> dat <- makeScales(dat, schalen);
>
```

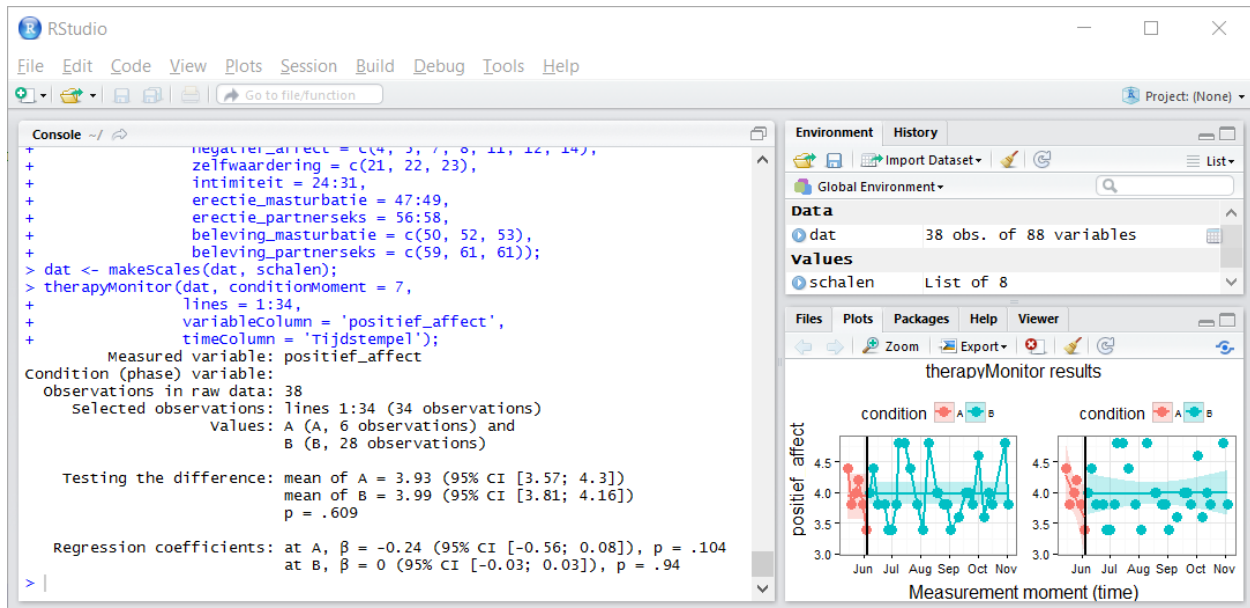
The Environment panel on the right shows the 'Global Environment' with a 'Data' section containing 'dat' (38 obs. of 88 variables) and 'schalen' (List of 8).

11. Zoals hierboven te zien is geeft R weinig feedback. Dat is geruststellend: in het geval van R geldt meestal dat geen nieuws goed nieuws is. We kunnen nu starten met de daadwerkelijke analyse. De therapie startte in dit voorbeeld na sessie 6, dus dat gaan we specificeren in de analyse:

```
therapyMonitor(dat, conditionMoment = 7,
               lines = 1:34,
               variableColumn = 'positief_affect',
               timeColumn = 'Timestamp');
```

In dit commando specificert 'dat' het dataframe waar de data in staat. 'conditionMoment' geeft aan op welke regel de therapie begon, in dit geval regel 7. Met 'lines' wordt aangegeven welke regels moeten worden geanalyseerd: in dit geval alleen regels 1 tot en met 34. Met 'variableColumn' wordt aangegeven welke variabele bekeken moet worden. In dit eerste voorbeeld kijken we alleen naar positief affect. Tot slot geeft timeColumn aan in welke kolom de tijd en datum staat opgeslagen. Die heet in dit voorbeeld 'Timestamp'.

R geeft dan de uitkomsten, inclusief een grafische weergave van deze resultaten:

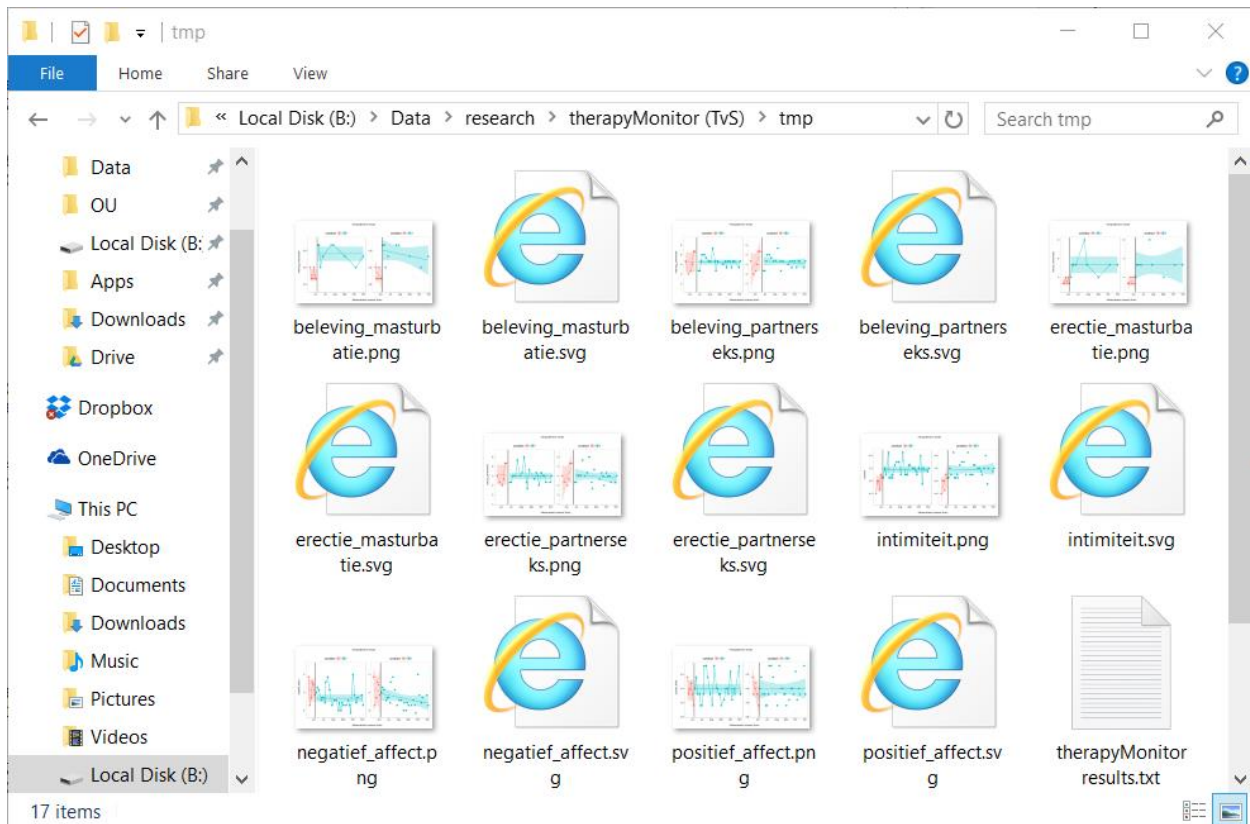


Het plotje dat is gegenereerd kan worden opgeslagen door op de 'Export' knop te klikken (deze is net boven het plotje in het paneel rechtsonder zichtbaar). Het plotje kan dan als PDF of als afbeelding worden opgeslagen. Door op 'Zoom' te klikken, kan het plotje groter op het scherm worden getoond.

12. Tot slot is het ook mogelijk om gelijk meerdere analyses uit te voeren. Hiervoor wordt het gerelateerde commando 'therapyMonitor.multi' gebruikt. Er kan dan bovendien worden aangegeven waar de resultaten moeten worden opgeslagen, zodat de plotjes en uitkomsten later geraadpleegd kunnen worden:

```
therapyMonitor.multi(dat, lines=1:34, conditionMoment=7,
                    variableColumn=names(schalen),
                    timeColumn = 'Tijdstempel', outputFiles=TRUE,
                    outputFilePath=
                        'B:/Data/research/therapyMonitor (TvS)/tmp');
```

Met 'outputFiles=TRUE' geven we aan dat we de uitkomsten willen opslaan in bestanden, en vervolgens geven we met 'outputFilePath' aan waar die bestanden moeten worden opgeslagen. R bewaart de uitkomsten vervolgens netjes:



Deze resultaten kunnen dan vervolgens bijvoorbeeld worden besproken met de client.

Lukt de installatie van een package niet?

Soms lukt het R niet om een package goed te downloaden en installeren. In dat geval meldt R dat met de melding `“there is no package called ‘...’”`. Door terug te lezen wordt dan duidelijk dat R inderdaad moeite had met de installatie (`“unable to move temporary installation ...”`):

```
> install.packages("httpuv");
Installing package into 'B:/R-library/library-64'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/
3.2/httpuv_1.3.3.zip'
Content type 'application/zip' length 871806 bytes (851 K
B)
downloaded 851 KB

package 'httpuv' successfully unpacked and MD5 sums check
ed
warning in install.packages :
  unable to move temporary installation 'B:\R-library\lib
rary-64\file395411d2475e\httpuv' to 'B:\R-library\library
-64\httpuv'

The downloaded binary packages are in
  C:\Users\gjal't-Jorn Peters\AppData\Local\Temp\Rtm
pw311rf\downloaded_packages
> require("httpuv");
Loading required package: httpuv
warning message:
In library(package, lib.loc = lib.loc, character.only = T
RUE, logical.return = TRUE, :
  there is no package called 'httpuv'
> |
```

In dat geval moet het commando om het package te installeren simpelweg worden herhaald. Vaak gaat het dan wel goed:

```
> require("httpuv");
Loading required package: httpuv
warning message:
In library(package, lib.loc = lib.loc, character.only = T
RUE, logical.return = TRUE, :
  there is no package called 'httpuv'
> install.packages("httpuv");
Installing package into 'B:/R-library/library-64'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/
3.2/httpuv_1.3.3.zip'
Content type 'application/zip' length 871806 bytes (851 K
B)
downloaded 851 KB

package 'httpuv' successfully unpacked and MD5 sums check
ed

The downloaded binary packages are in
  C:\Users\gjal't-Jorn Peters\AppData\Local\Temp\Rtm
pw311rf\downloaded_packages
> require("httpuv");
Loading required package: httpuv
> |
```