Introductie Bayesiaanse Analyses & R

Meten en Evalueren

Sven De Maeyer

Department of Training and Education Sciences

Inhoud

- NHST
- Bayesiaans intro
- Bayesiaans Stan
- Voorbeelden

De "holy grail": $p < 0.05$?

The greatest obstacle that I encounter among students and colleagues is the tacit belief that the proper objective of statistical inference is to test null hypotheses. This is the proper objective, the thinking goes, because Karl Popper argued that science advances by falsifying hypotheses. Karl Popper (1902–1994) is possibly the most influential philosopher of science, at least among scientists. He did persuasively argue that science works better by developing hypotheses that are, in principle, falsifiable. Seeking out evidence that might embarrass our ideas is a normative standard, and one that most scholars—whether they describe themselves as scientists or not—subscribe to. So maybe statistical procedures should falsify hypotheses, if we wish to be good statistical scientists. – *McElreath*

NHST

Stel:

- vaststelling: Meisjes score sign. hoger dan Jongens op Begrijpend Lezen in PIRLS study ($t = 2.56$; $df = 2366$; $p = 0.0105$)
- Wat betekent die p-waarde nu echt?

Is NHST falsificationist?

Is NHST falsificationist? Null hypothesis significance testing, NHST, is often identified with the falsificationist, or Popperian, philosophy of science. However, usually NHST is used to falsify a null hypothesis, not the actual research hypothesis. So the falsification is being done to something other than the explanatory model. This seems the reverse from Karl Popper's philosophy. – McElreath

Wanneer spreken we van NHST?

Alle methodes waarbij we gebruik maken van *steekproevenverdelingen van imaginaire data...*

Een vb.:

p-waarde bij de t-test is de kans dat we een t-waarde uitkomen die minstens zo groot is als die in onze data

INDIEN - en hier komt de imaginaire data -

we volledig dezelfde steekproefmethode zouden toepassen; en de nulhypothese opgaat.

Waarom 'weggaan' van NHST?

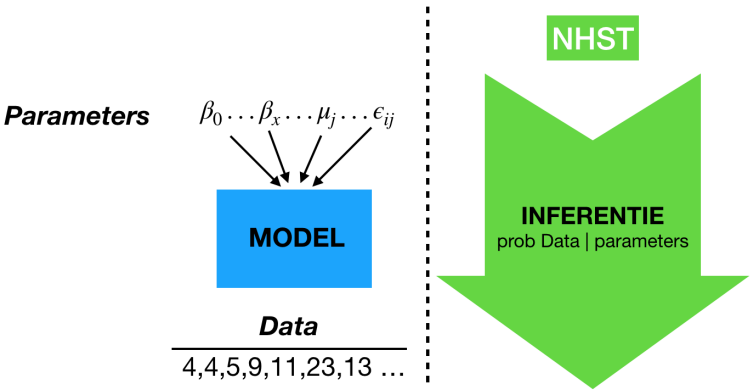
De roep om NHST te verlaten is groot. Bv.:

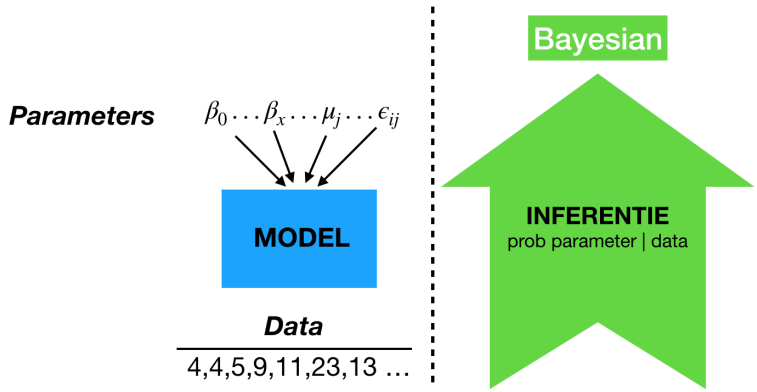
> *the arbitrary classification of results into 'significant' and 'non-significant' is unnecessary for and often damaging to valid interpretation of data; and that estimation of the size of effects and the uncertainty surrounding our estimates will be far more important for scientific inference and sound judgment than any such classification. – (Greenland et al., 2016)*

Belangrijkste redenen:

- Null hypotheses zijn vaak a priori fout;
- NHST ontkent effectgrootte en onzekerheid.

NHST vs. Bayesian





Soorten statistische analyses

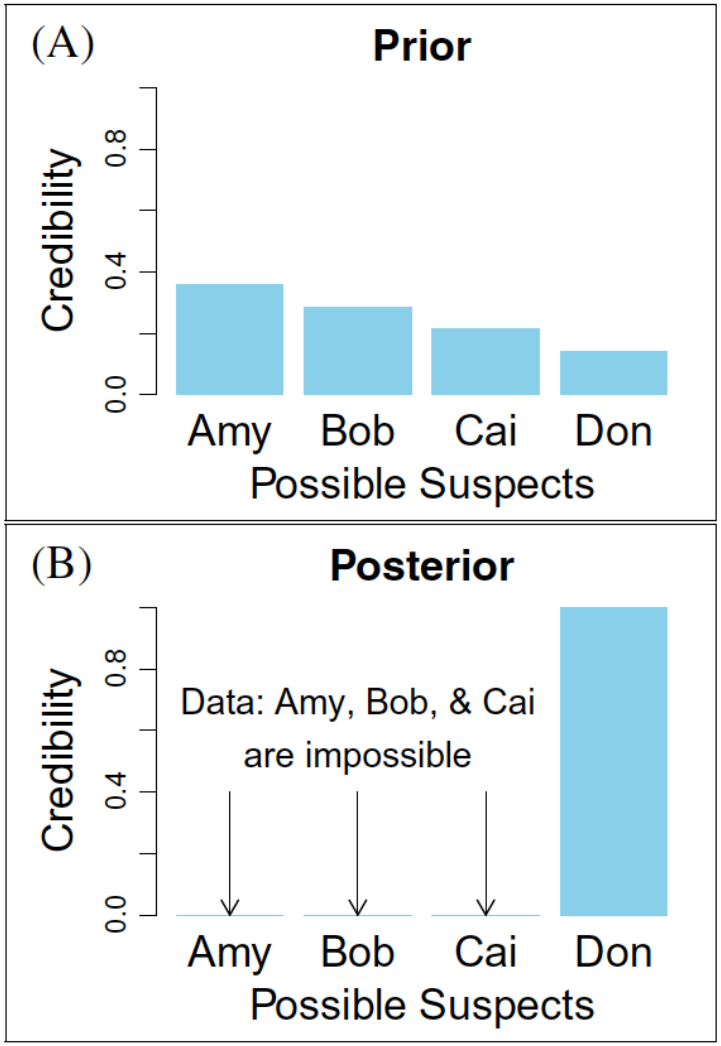
KRUSCHKE AND LIDDELL

	Frequentist	**Bayesian**
Hypothesis test	*p* value (null hypothesis significance test)	Bayes factor
Estimation with uncertainty	maximum likelihood estimate with confidence interval (The "New Statistics")	posterior distribution with highest density interval

Figure 1: Classificatie van methodes volgens Kruschke en Liddell (2016). . .

Bayesiaanse werkwijze // Sherlock Holmes' methode

"We need EVIDENCE Watson!"



Bayesiaanse werkwijze

Bayesiaanse werkwijze is vergelijkbaar met Holmes' methode

Aanvankelijke kans $\rightarrow$ Data/observatie $\rightarrow$ Uiteindelijke kans schuldig

Prior $\rightarrow$ Data/observatie $\rightarrow$ Posterior

Bayesiaanse Inferentie ... Opwarmertje!

prof. dr. Van Helsing (de beroemde Vampierenjager)

Wou geïnformeerd aan de slag gaan in z'n zoektocht naar Vampieren!

Z'n bezorgdheid: Hoeveel vampieren zijn er eigenlijk in de populatie?

Oplossing: z'n fameuze 'Vampire Test' loslaten op 200 random personen uit de populatie

Resultaat: 10.5% uit de sample is Vampier

Maar: hoe zit het in de gehele populatie?

Bayesiaanse Inferentie ... Opwarmertje!

Wat zegt de 'klassieke' benadering?

Betrouwbaarheidsinterval opstellen rond 10.5

$$Steekproeffout.Prop = \sqrt{\frac{p * (1 - p)}{n}}$$

Hier:

$$Steekproeffout.Prop = \sqrt{\frac{0.105 * (1 - 0.105)}{200}} = 0.0216766$$

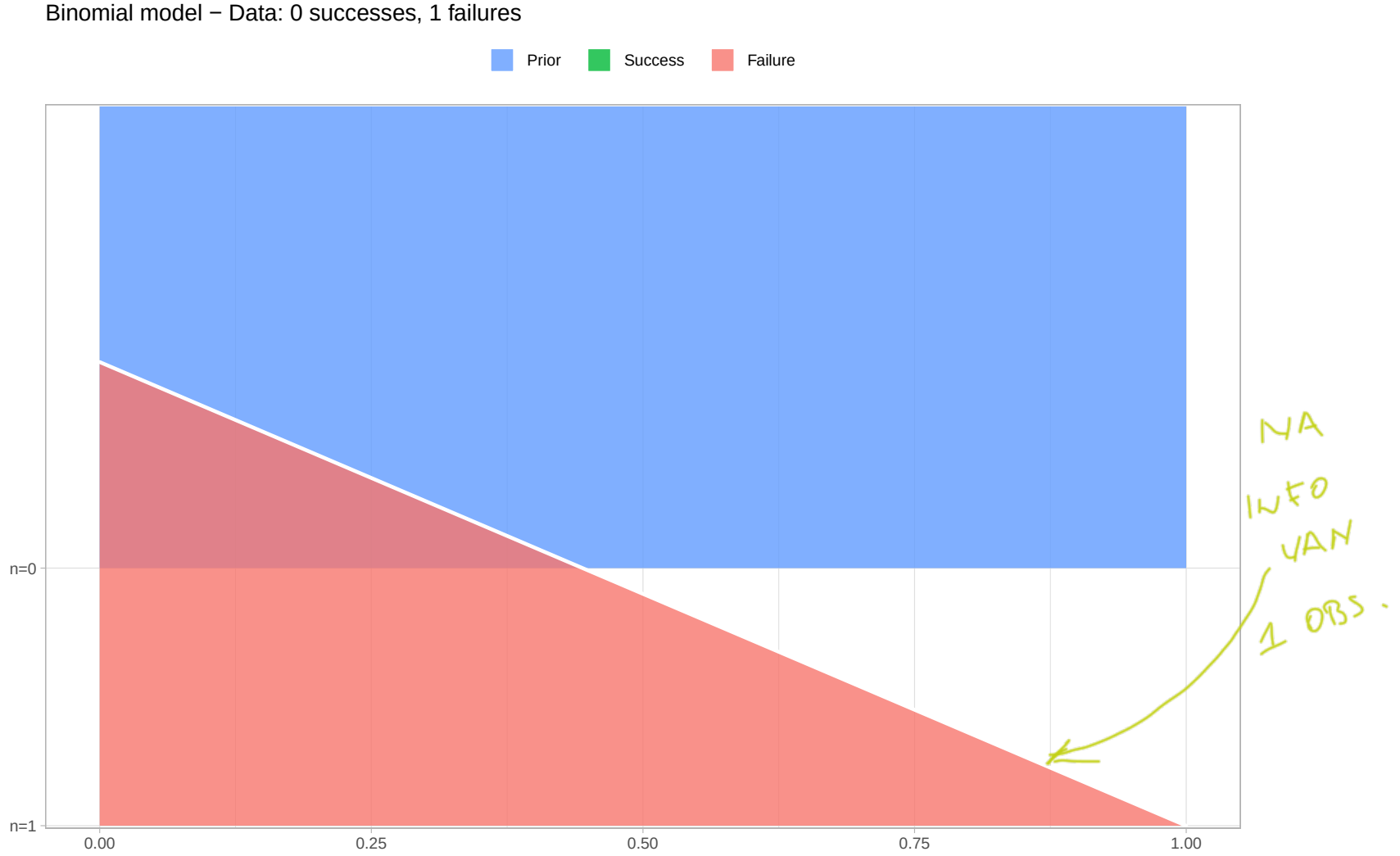
Betrouwbaarheidsinterval:

- Ondergrens $= 0.105 - (1.96 * 0.0216766) = 0.0625119$
- Bovengrens $= 0.105 + (1.96 * 0.0216766) = 0.1474881$

95% van de steekproeven (n=200) resulteert in prop. tussen 6.25% en 14.75%

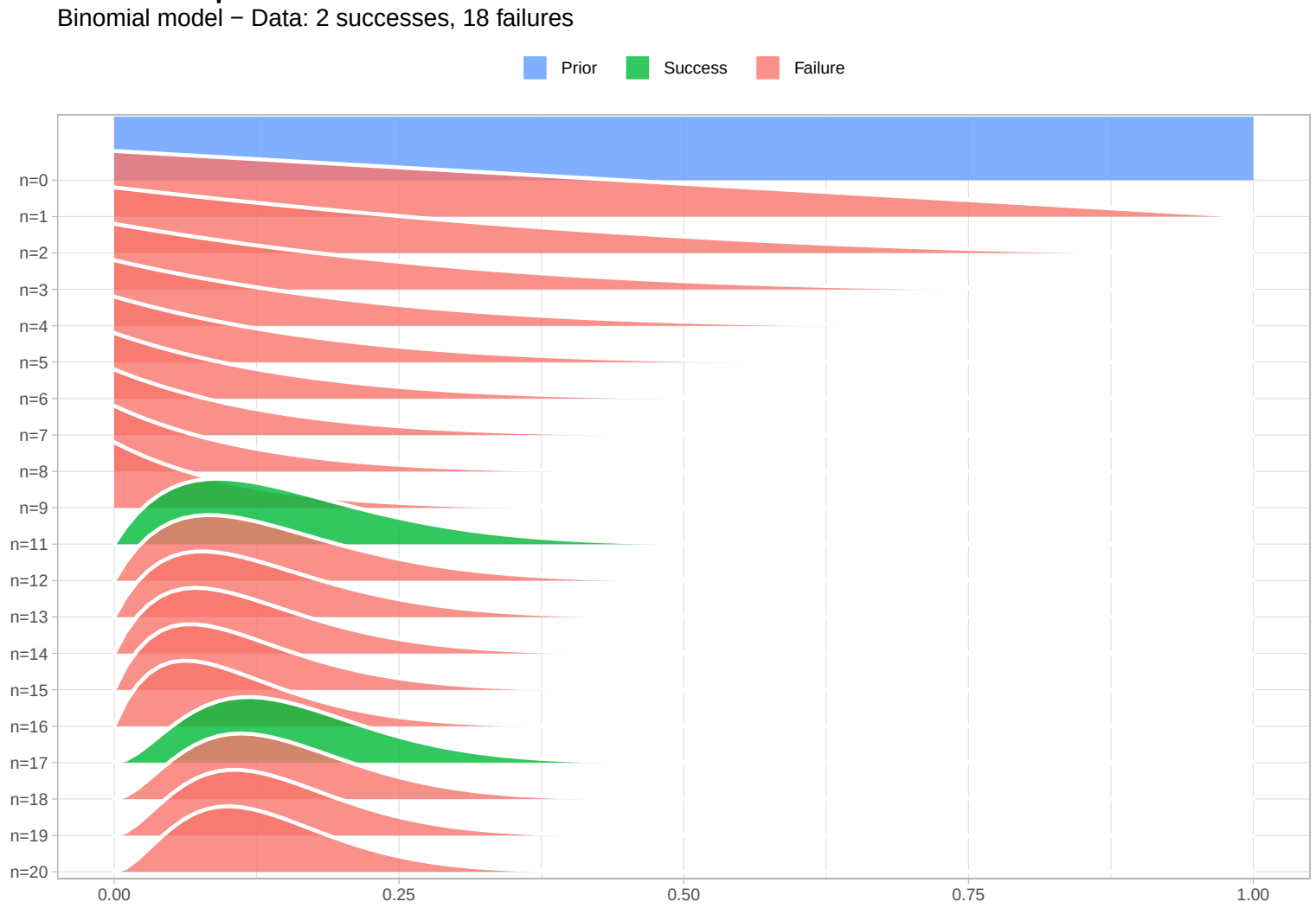
Bayesiaanse Inferentie . . . Opwarmertje!

Hoe werkt Bayesian Inference? Evolutie van schatting van de proportie per extra datapunt



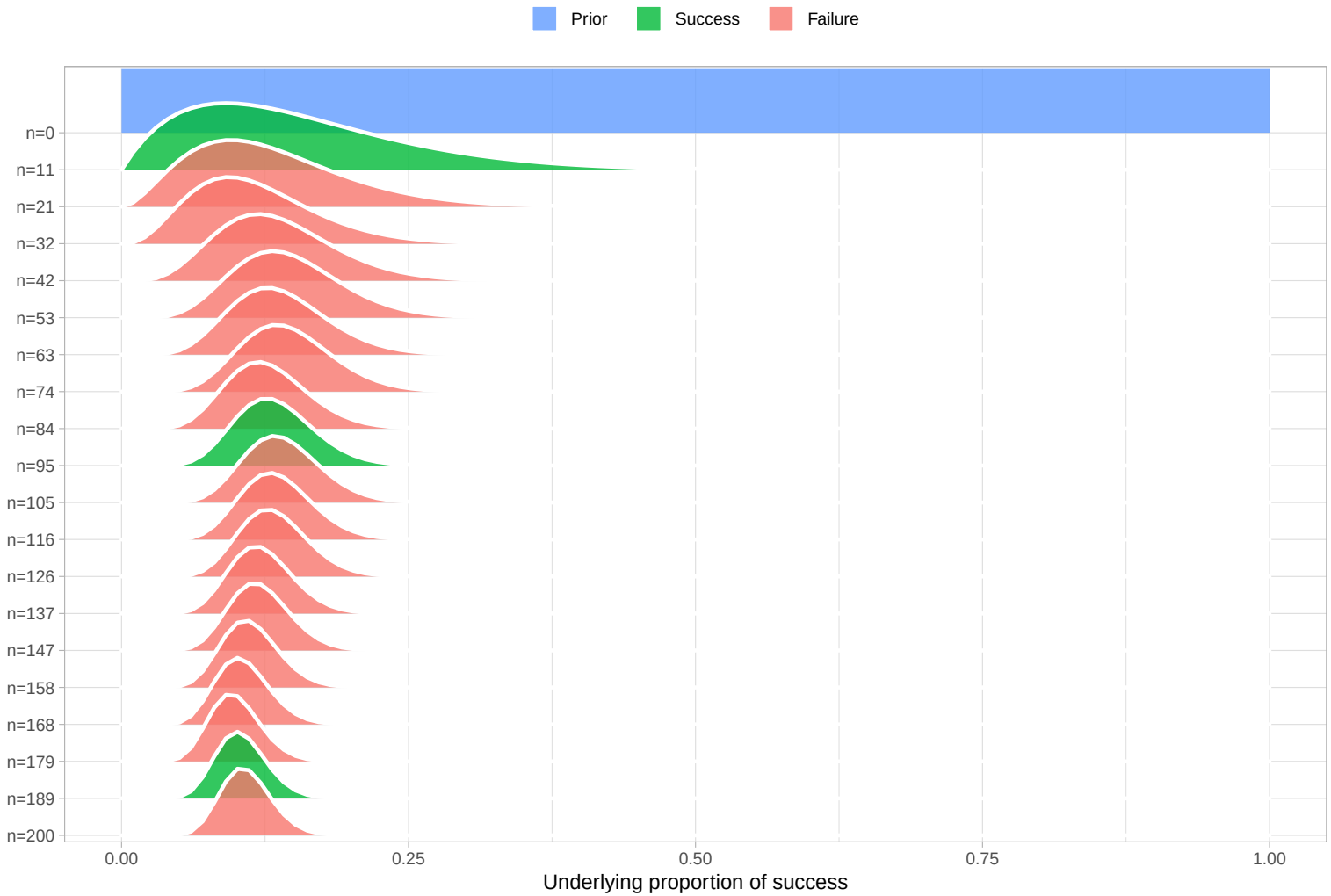
Bayesiaanse Inferentie . . . Opwarmertje!

Hoe werkt Bayesian Inference? Evolutie van schatting van de proportie per extra datapunt



Bayesiaanse Inferentie . . . Opwarmertje!

Hoe werkt Bayesian Inference? De gehele dataset van Van Helsing



Bayesiaanse Inferentie ... Opwarmertje!

Hoe werkt Bayesian Inference?

Informatie halen uit de posterior:

```
quantile(Vamp_Posterior,c(0.025, 0.5, 0.975))
```

```
##       2.5%        50%      97.5%
## 0.06950138 0.10799312 0.15568667
```

=> 95% MEEST WAARSCHIJNLIJKE PAR. ERGENS TUSSEN 7% & 15,6% VAMPIEREN

Bayesian Theorem: een Vb.

Stel er is een bloedtest die 95% van de vampieren correct detecteert $Pr(positive|vampire) = 0.95$

In 1% van de testen wordt een gewone sterveling verkeerdelijk als vampier aangeduid
$Pr(positive|mortal) = 0.01$

Vampieren zijn bovendien vrij uitzonderlijk: slechts 0.1% van de populatie is vampier $Pr(vampire) = 0.001$

De bloedtest van onze kerel is positief! Wat is de kans dat

het werkelijk een vampier is? $Pr(vampire|positive) = ?$

Bayesian Theorem - met kansen...

$$Pr(B|A) = \frac{Pr(A|B) * Pr(A)}{Pr(B)}$$

$$Pr(vampire|positive) = \frac{Pr(positive|vampire) * Pr(vampire)}{Pr(positive)}$$

Daarbij is $Pr(positive)$ de gemiddelde kans op een positief testresultaat
$Pr(positive) =$
$Pr(positive|vampire)Pr(vampire) + Pr(positive|mortal)Pr(1 - Pr(vampire))$

```
PrPV <- 0.95
PrPM <- 0.01
PrV <- 0.001
PrP <- PrPV*PrV + PrPM*(1-PrV)
( PrVP <- PrPV*PrV / PrP )
```

```
## [1] 0.08683729
```

Probabiliteitstheorie is eigenlijk werken met aantallen

Makkelijker/intuïtiever op te lossen als je in werkelijke aantallen denkt!

- Populatie = 100 000
- N vampieren = 100
- N vampieren positief getest = 95
- N stervelingen = 99 900
- N sterveling positief getest = 999

Kans dat iemand die positief test een vampier is = ?

$$\frac{95}{95+999} = \frac{95}{1094} = 0.087$$

Bayesiaanse inferentie

Bayesiaanse inferentie is gebaseerd in probabiliteitstheorie en net als alles in probabiliteitstheorie niet meer of minder dan

MOGELIJKHEDEN TELLEN

en de

MOGELIJKHEDEN VERGELIJKEN

Bayesiaanse inferentie - kernbegrippen

3 kernbegrippen:

- de PARAMETER, de grootheid/het getal waarin we interesse hebben;
- de LIKELIHOOD, de waarschijnlijkheid van wat we zien (= de data) gegeven een zekere parameterwaarde;
- de PRIOR PROBABILITY, de probabiliteit van een parameterwaarde zonder rekening te houden met informatie (= voor de data);
- de POSTERIOR PROBABILITY, de probabiliteit van een parameterwaarde gegeven de data.

$$Pr(parameter|Data) = \frac{Pr(Data|parameter) * Pr(parameter)}{Pr(Data)}$$

$$Posterior = \frac{Likelihood * Prior}{AverageLikelihood}$$

Bayesiaanse inferentie - een voorbeeld

Hoeveel centiliter bloed drinkt een gemiddeld vampier per bijtbeurt?
= tikje complexer: nl 2 parameters in het model: gemiddelde en standaardafwijking

$$Bloed_i \sim Normal(\mu, \sigma)$$

- PARAMETERS = gemiddelde en sd hoeveelheid bloed in cl;
- LIKELIHOOD = verdeling van hoeveelheid bloed in de populatie;
- PRIOR PROBABILITY = kans per waarde hoeveelheid bloed als potentieel werkelijk gemiddelde en sd;
- POSTERIOR PROBABILITY = kans per waarde hoeveelheid bloed als potentieel werkelijk gemiddelde en sd, gegeven data die we verzamelden.

Bayesiaanse inferentie - een voorbeeld

LIKELIHOOD : $Bloed_i \sim Normal(\mu, \sigma :)$

PRIOR μ : $\mu \sim Normal(33, 15)$

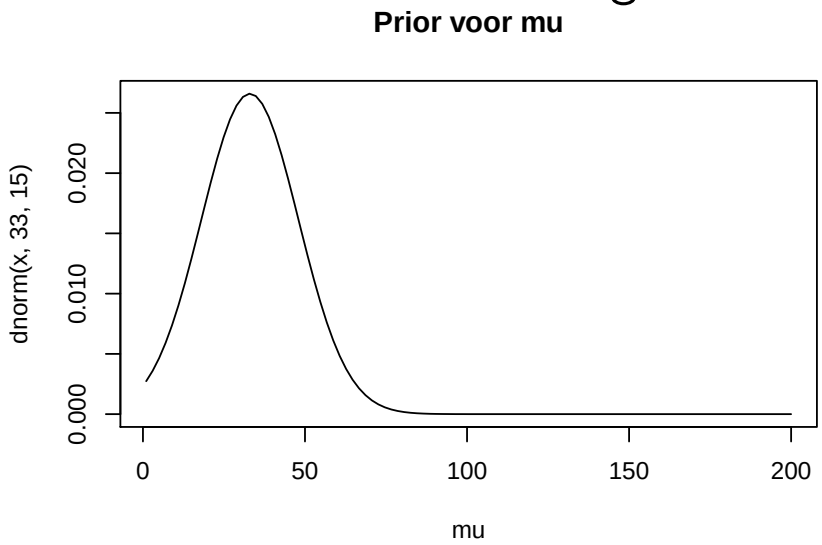
PRIOR σ : $\sigma \sim Uniform(0, 40)$

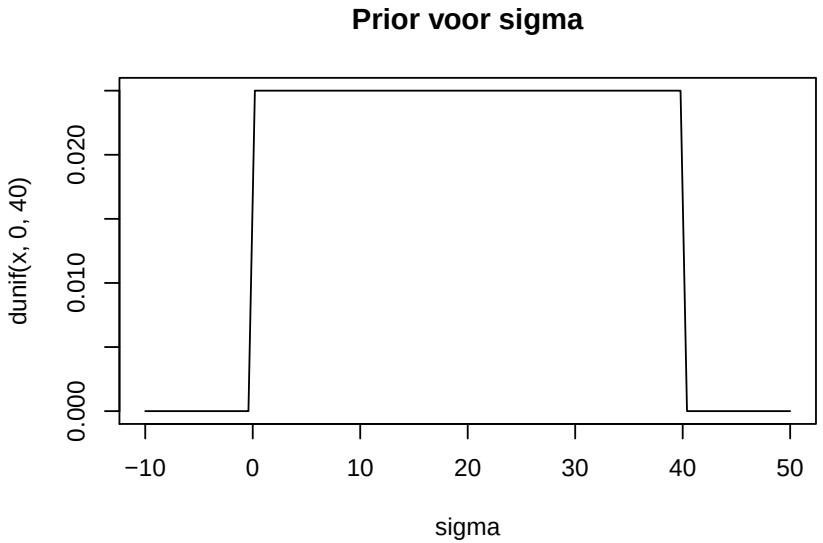
De data die we verzamelen:

```
N <- 10
Bloed <- c(34,44,22,14,68,26,54,12,23,26)
```

De priors even inspecteren . . .

Onze "voorkennis" is "vaag"



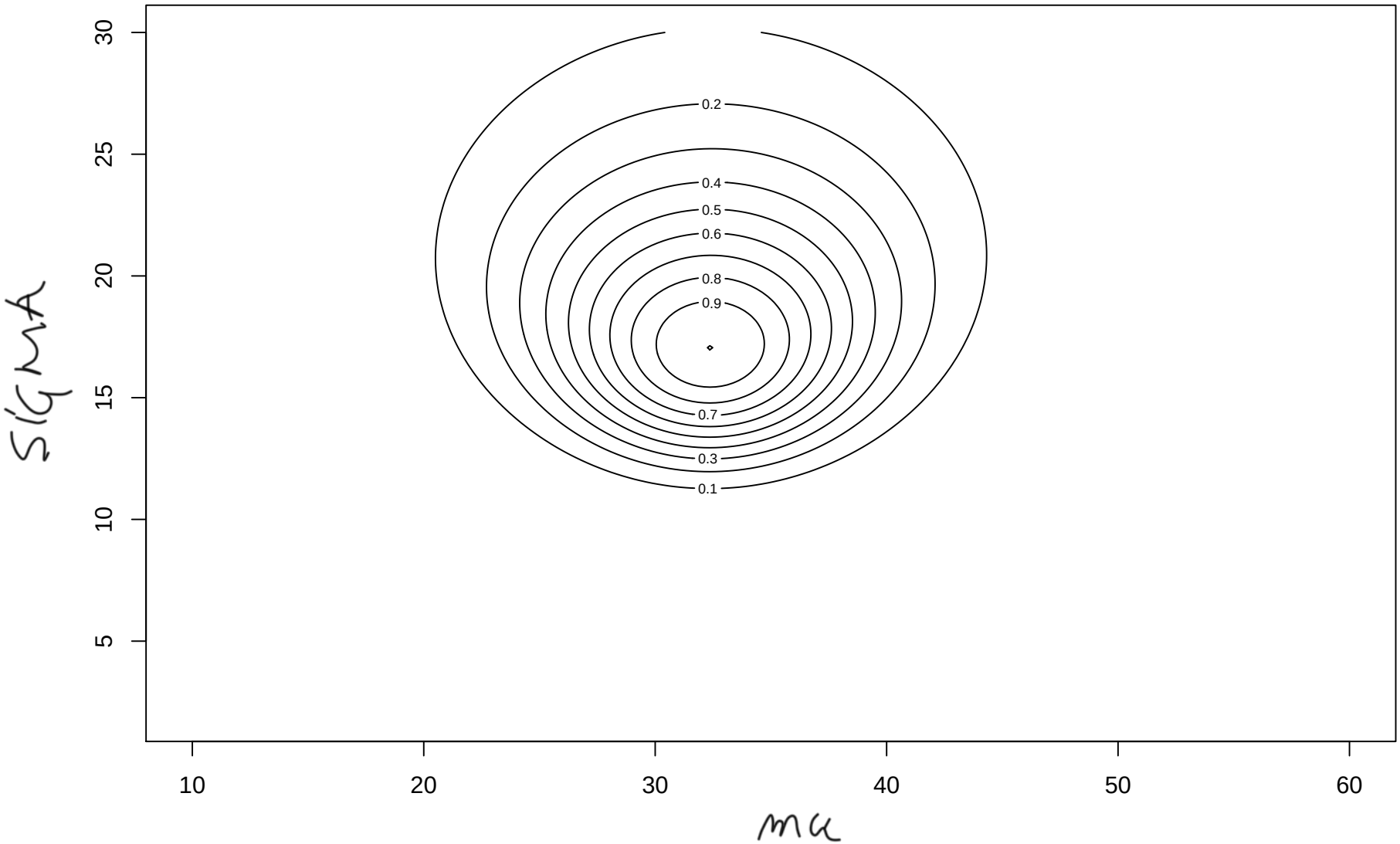


Bayesiaanse analyse = updaten van de prior dmv data

Complexe manuele berekening van de posterior:

```
mu.list <- seq( from=10, to=60 , length.out=200 )
sigma.list <- seq( from=2 , to=30 , length.out=200 )
post <- expand.grid( mu=mu.list , sigma=sigma.list )
post$LL <- sapply( 1:nrow(post) , function(i) sum( dnorm(
  Bloed ,
  mean=post$mu[i] ,
  sd=post$sigma[i] ,
  log=TRUE ) ) )
post$prod <- post$LL + dnorm( post$mu , 33 , 15 , TRUE ) +
dunif( post$sigma , 0 , 40 , TRUE )
post$prob <- exp( post$prod - max(post$prod) )
```

De posterior interpreteren - contourplot



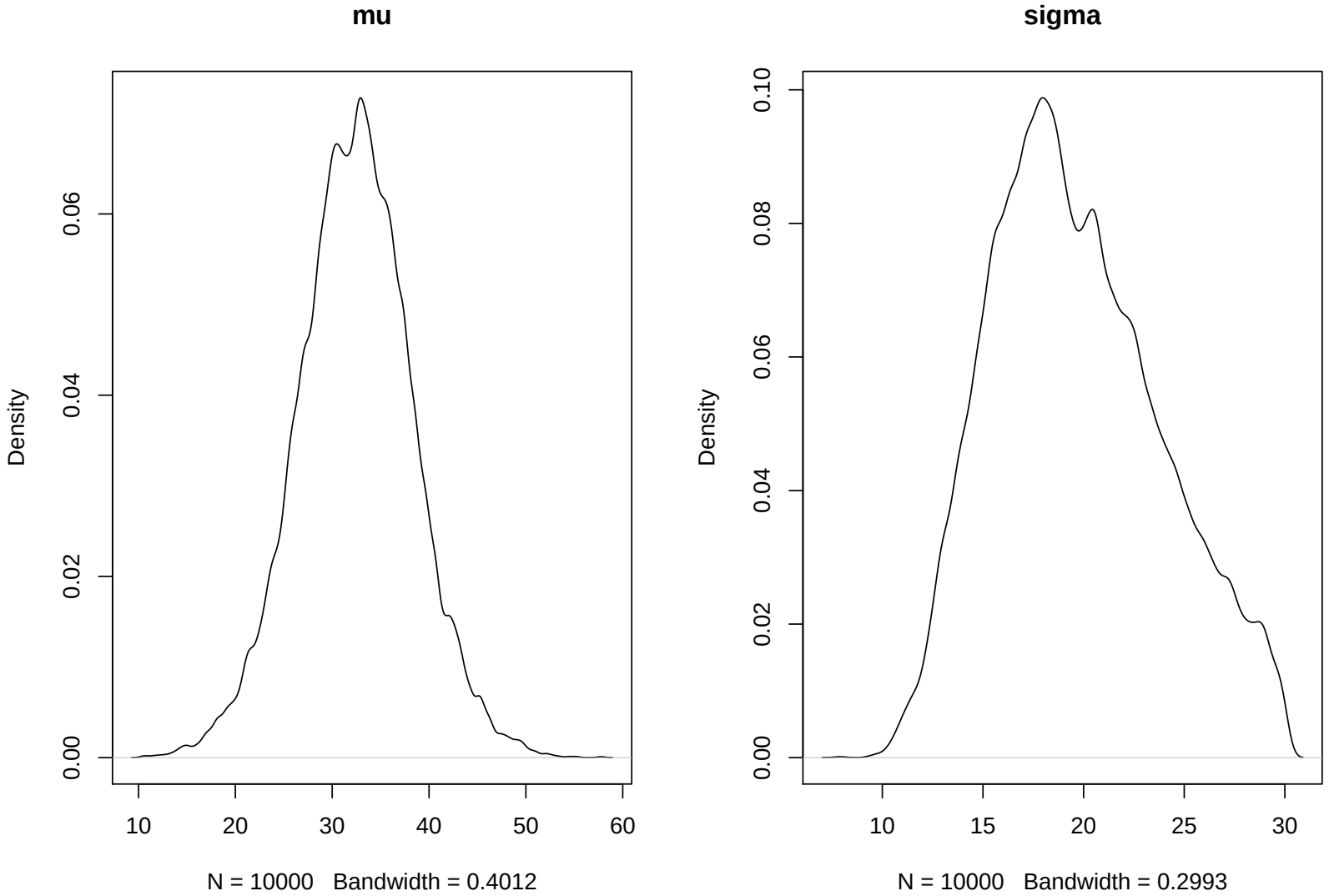
De posterior interpreteren

Eerst sample parameters (n=10 000) trekken, waarbij meest waarschijnlijke parameterwaarden grootste kans hebben:

```
sample.rows <- sample( 1:nrow(post) , size=1e4 , replace=TRUE ,
prob=post$prob )
sample.mu <- post$mu[ sample.rows ]
sample.sigma <- post$sigma[ sample.rows ]
```

De posterior interpreteren - visueel

De posterior visueel



De posterior interpreteren - HPDI

Highest Probability Density Intervals

```
HPDI(sample.mu)
```

```
##    |0.89    0.89|
## 22.81407 41.40704
```

```
HPDI(sample.sigma)
```

```
##    |0.89    0.89|
## 12.69347 26.06030
```

Wat met complexere modellen?

Stel je een regressiemodel voor:

$$Y_i = \beta_0 * Cons + \beta_1 * X_1 + \beta_2 * X_2 + \beta_3 * X_3 + \epsilon_i$$

Dit impliceert al meerdere parameterschattingen:

- een intercept;
- regressiecoëfficiënten;

Complexiteit kan snel oplopen $\rightarrow$ zeer veel mogelijke combinaties van mogelijke waarden waarvoor posterior probabiliteit berekend dient te worden

de oplossing:
Markov chain Monte Carlo (MCMC)

Markov chain Monte Carlo - Waarom?

Complexe modellen $\rightarrow$ Meerdere parameterschattingen

Bijgevolg is de posterior een 'joint probability' functie die niet meer voor alle mogelijke combinaties is te berekenen...

We willen de 'joint posterior' BENADEREN

De vraag is: HOE DOEN WE DIT HET EFFICIËNTSTE?

Oplossing: STEEKPROEVEN TREKKEN VAN COMBINATIES VAN PARAMETERWAARDEN

Maar: RANDOM STEEKPROEVEN IS NIET EFFICIËNT!

Markov chain Monte Carlo - The tale of King Markov

King Markov: koning van een eilandengroep

Elk eiland verschillend aantal bewoners prop. aan oppervlakte

Eiland 2: 2 * zoveel bewoners als eiland 1; . . .

Wilt elk eiland bezoeken + elke bewoner een gelijke kans geven hem te zien

MAAR:

- houdt niet van vaste agenda en veel geplan;
- houdt niet van varen $\rightarrow$ enkel varen tussen aanpalende eilanden.

Markov chain Monte Carlo - The tale of King Markov

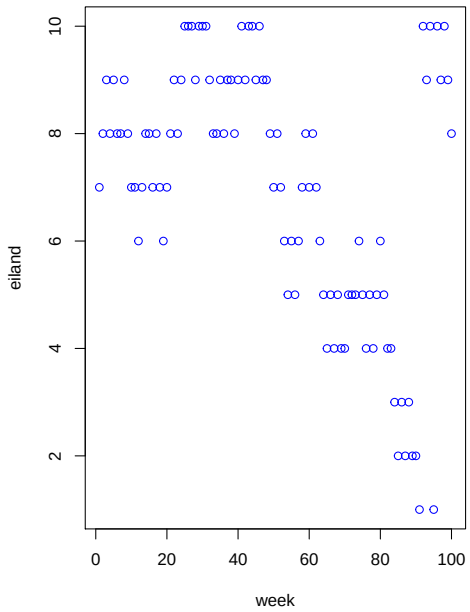
Dienaar *Metropolis* bedenkt een oplossing: Metropolis algoritme

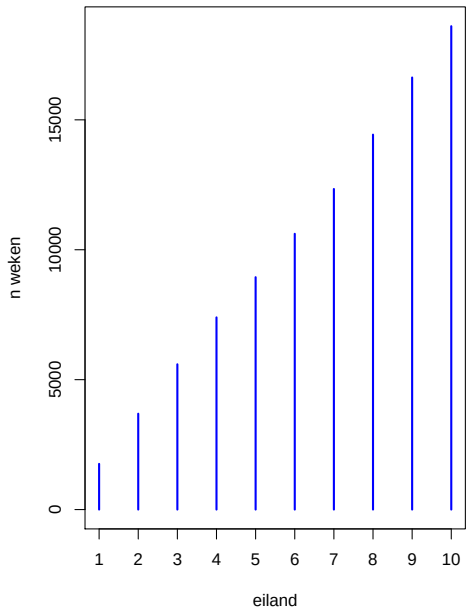
Elke week beslist de koning opnieuw (blijven of niet), volgende werkwijze:

- "flip a coin"
  - Kop = tegen de klok varen
  - Munt = met de klok varen
- verzamel n schelpen, n = relatieve grootte van voorgesteld eiland;
- verzamel n stenen, n = relatieve grootte huidig eiland;
- n schelpen $>$ n stenen $\rightarrow$ veranderen van eiland;
- n schelpen $<$ n stenen $\rightarrow$ alle schelpen + n_(stenen) - n_(schelpen) in een zak;
- hij kiest random een object uit de zak:
  - schelp $\rightarrow$ veranderen van eiland;
  - steen $\rightarrow$ blijven;

Markov chain Monte Carlo - The tale of King Markov

```
num_weken <- 1e5
eiland <- rep(0,num_weken)
huidig <- 7
for ( i in 1:num_weken ) {
  # huidig eiland vaststellen
  eiland[i] <- huidig
  # flip coin om een bestemming te bepalen
  bestemming <- huidig + sample( c(-1,1) , size=1 )
  # maken dat we binnen de 10 eilanden blijven
  if ( bestemming < 1 )  bestemming <- 10
  if ( bestemming > 10 ) bestemming <- 1
  # verplaatsen?
  prob_verplaatsen <- bestemming/huidig
  huidig <- ifelse( runif(1) < prob_verplaatsen , bestemming , huidig )
}
```





Markov chain Monte Carlo - bij Bayesiaans schatten

MCMC in Bayesiaanse analyses verloopt redelijk gelijkaardig

- ipv eilanden kiezen, gaat het over parameterwaarden kiezen
- proportioneel volgens de 'joint posterior' (de kans)
- ipv Metropolis algoritme verfijndere algoritmes: Gibbs sampling of Hamiltonian Monte Carlo
- gebruik makend van afzonderlijke 'samplers' (= software) zoals Stan

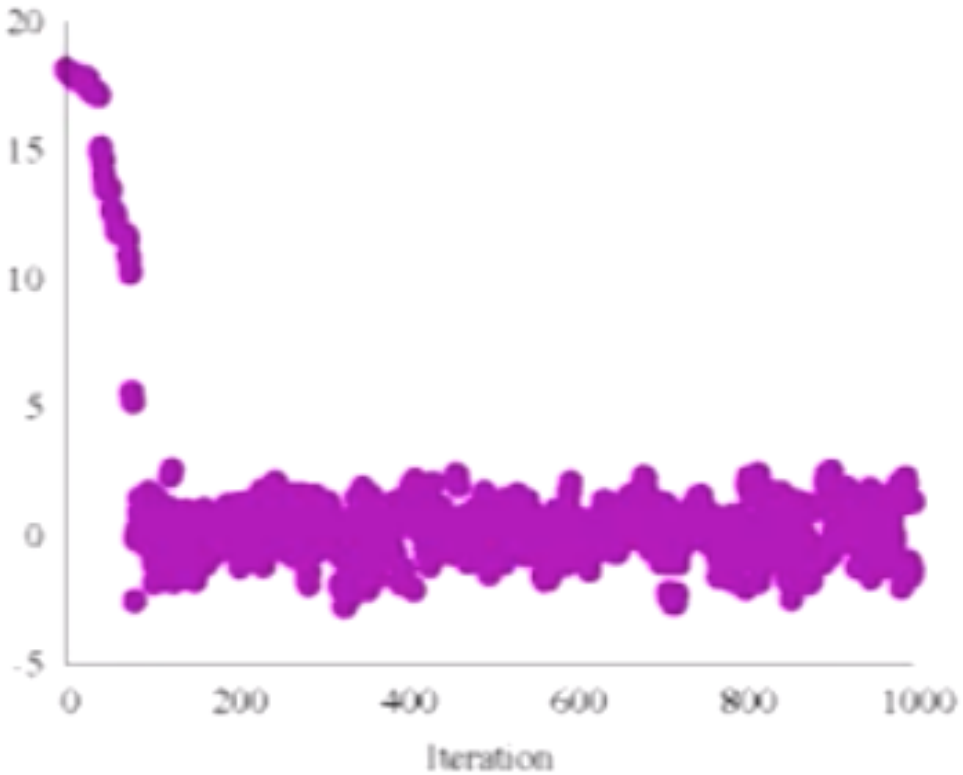
MCMC reist doorheen de 'multidimensionele parameterruimte'; stopt bij een set van parameterwaarden; berekent de 'joint posterior probability' van die set van parameterwaarden en reist verder (waarbij de richting en snelheid van de reis bepaald wordt door de densiteit (hoe hoger de densiteit, hoe trager de reis)) van die set van parameterwaarden

MCMC visuele demo

https://chi-feng.github.io/mcmc-demo/app.html#HamiltonianMC,banana : Visuele demo

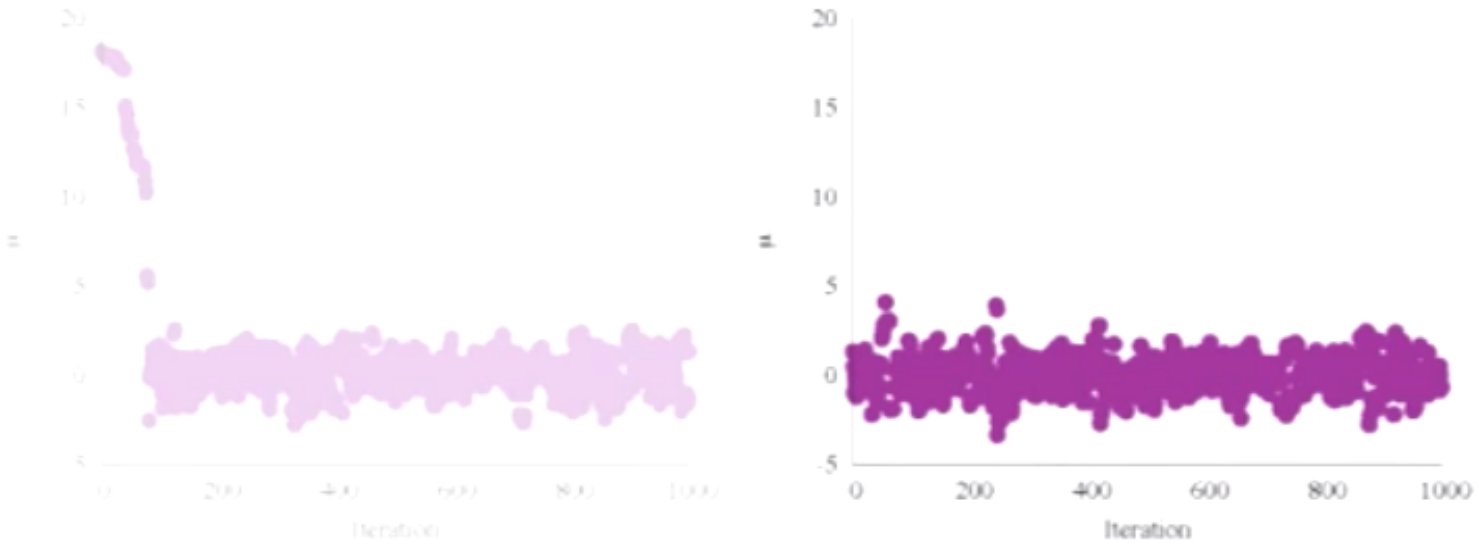
MCMC - 3 fases

Fase 1: Warm-up (soms ook burn-in)



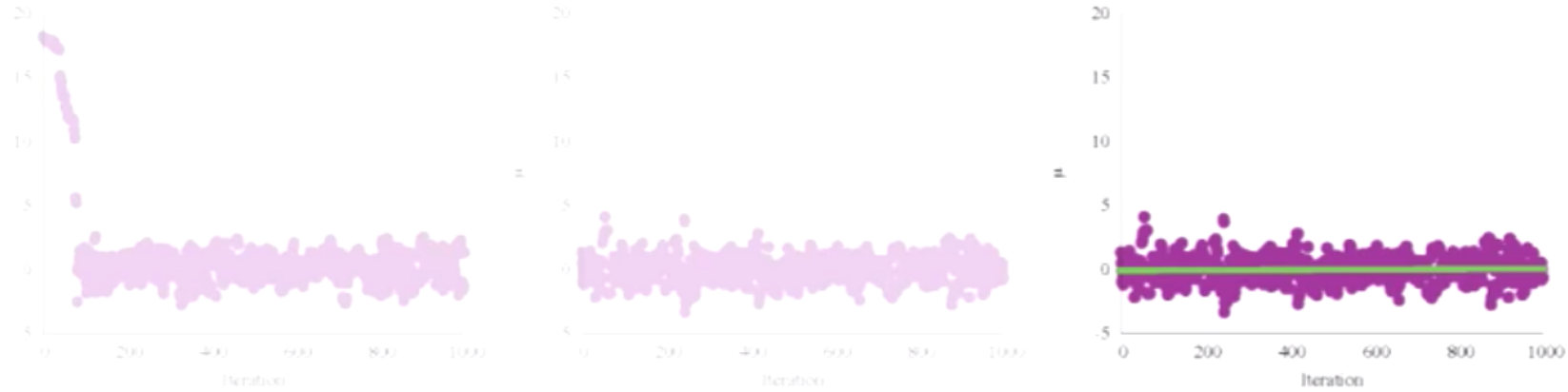
MCMC - 3 fases

Fase 2: Schatten



MCMC - 3 fases

Fase 3: Samenvatten/interpreteren



MCMC in R: Welkom Stan!

Stan is software die het HCM (Hamiltonian Monte-Carlo) algoritme implementeert.

Heeft een eigen taal en kan aangesproken worden via R.

Het pakket `brms` vergemakkelijkt het gebruik: bekende code om R modellen te definiëren als basis

MCMC in R: voorbeeld uit D-PAC

19 proefpersonen

10 vergelijkingen van teksten / proefpersoon

Hoe lang duurt het om een vergelijking te maken?

$$Duration_{ij} = \beta_0 * Cons + \mu_j + \epsilon_{ij}$$

```
library(brms)

load(file="Merge4.RData")
Data <- tbl_df(Merge4)

Data_Comp <- Data %>% group_by(ParticipantName,ID_Comp) %>%
  summarise(Duration=mean(GazeEventDuration,na.rm=T),
            Distance=mean(Est_distance,na.rm=T),
            Condition=first(Condition))

m1_Duration <- brm(Duration ~ 1+(1|ParticipantName),
                        data = Data_Comp,
                        warmup = 1000, iter = 5000, chains = 4, cores = 2,
                        seed = 1975)
```

R - brms: Waarvoor staat de code?

`warmup = 1000` hoeveel van de iteraties als warm-up;

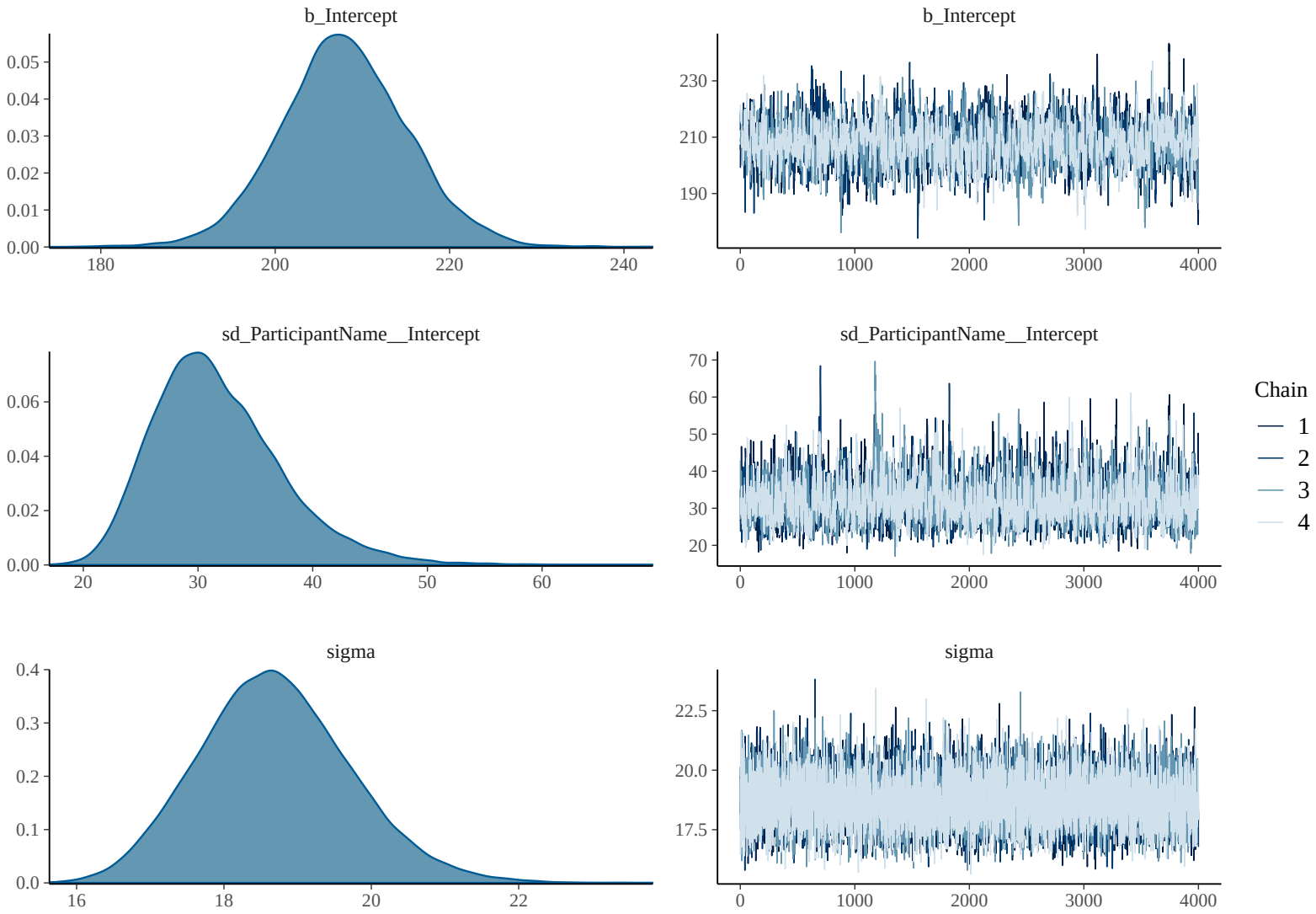
`iter = 5000` hoeveel iteraties in totaal;

`chains = 4` hoeveel keer er opnieuw gestart mag worden;

`cores = 2` hoeveel processoren in je pc mogen worden aangeschreven;

`seed = 1975` startpunt vastzetten.

R - brms: Hoe zien de resultaten eruit?



R - brms: Hoe zien de resultaten eruit?

```
summary(m1_Duration)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: Duration ~ 1 + (1 | ParticipantName)
##    Data: Data_Comp (Number of observations: 190)
## Samples: 4 chains, each with iter = 5000; warmup = 1000; thin = 1;
##          total post-warmup samples = 16000
##
## Group-Level Effects:
## ~ParticipantName (Number of levels: 19)
##               Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
## sd(Intercept)    31.76      5.67    22.83    44.82       2226 1.00
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
## Intercept   207.90      7.35   193.60   222.51       1736 1.00
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
## sigma    18.70      1.03    16.81    20.85       9724 1.00
##
## Samples were drawn using sampling(NUTS). For each parameter, Eff.Sample
## is a crude measure of effective sample size, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

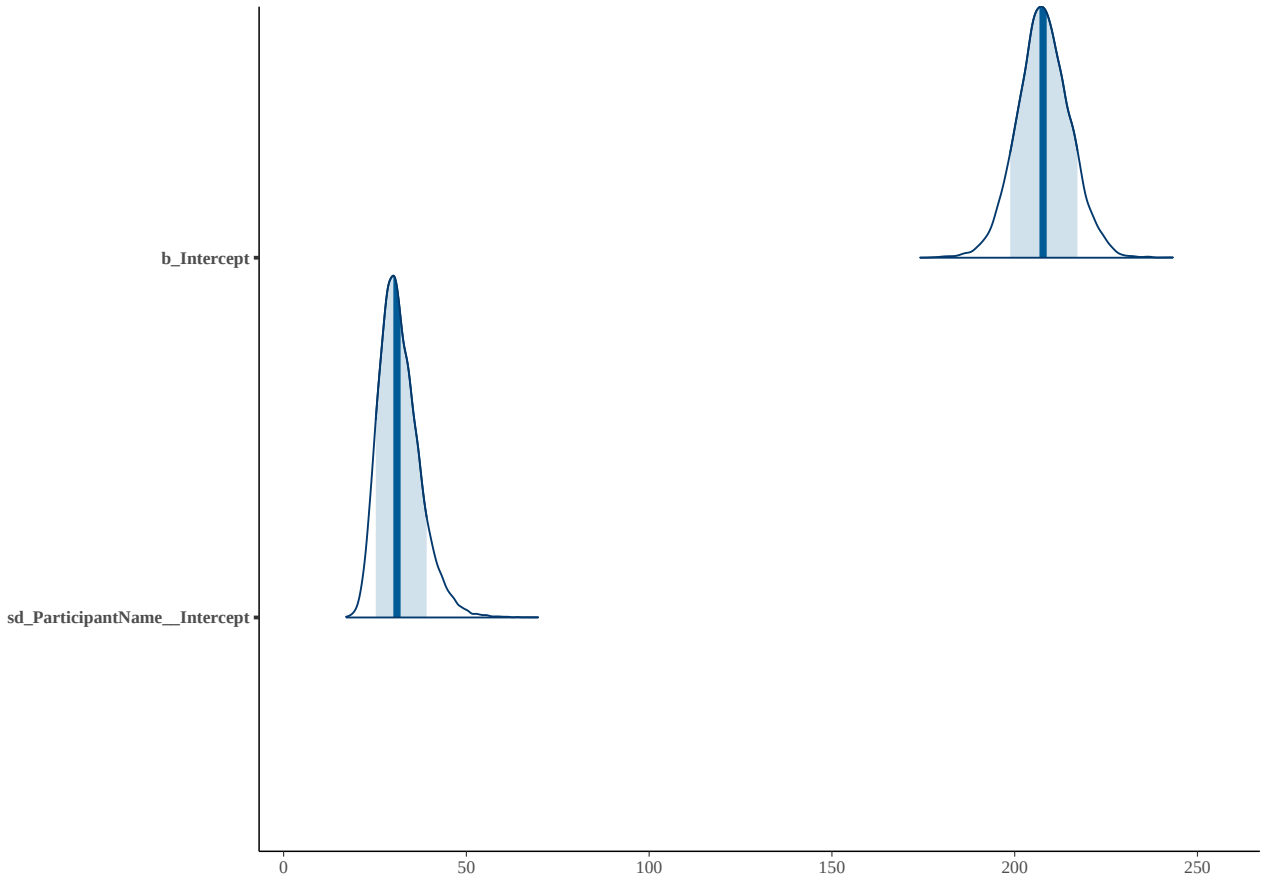
VAR. TSS EN PERSONEN

GEM. # SEC.

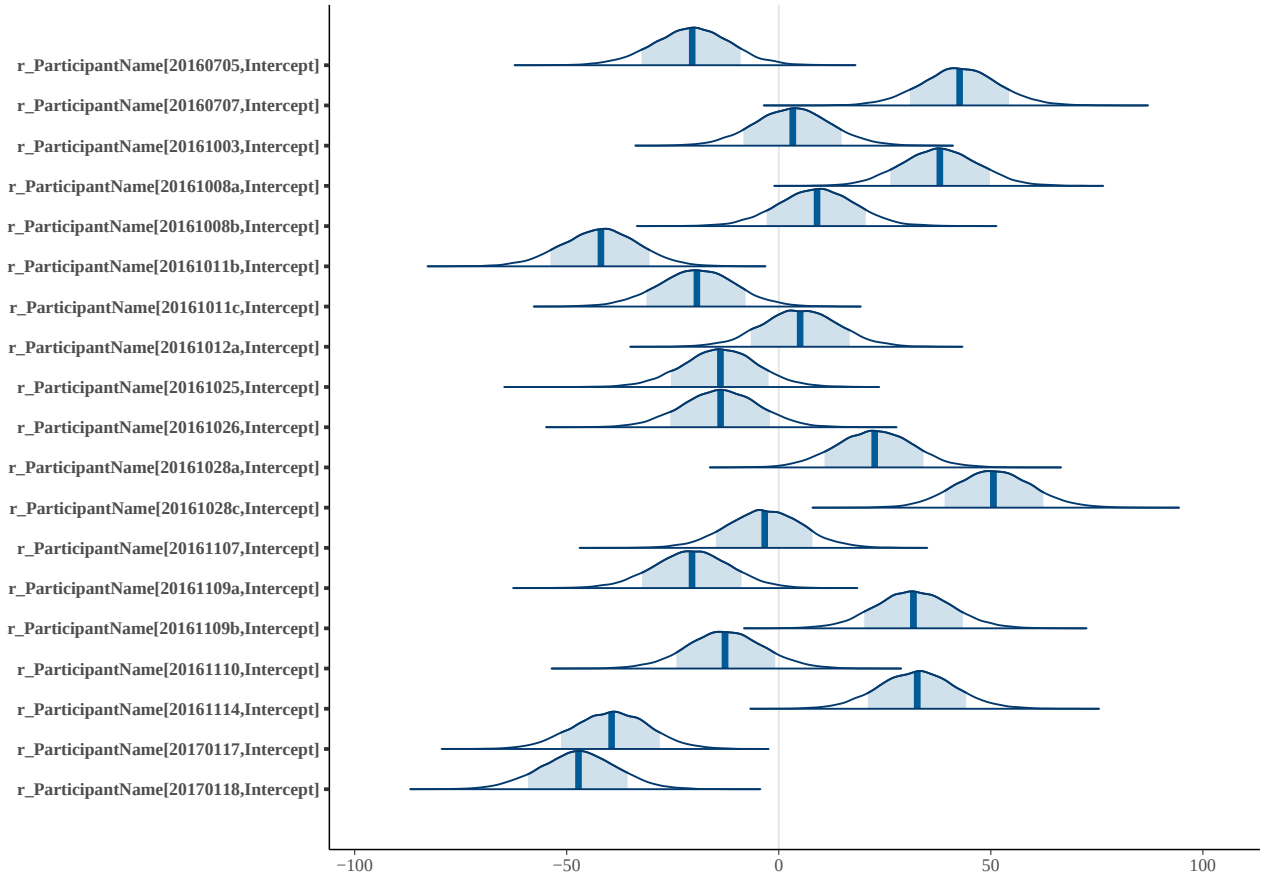
RESIDUELE VAR.

R - brms: Hoe zien de resultaten eruit?

```
m1_Duration_posterior <- as.matrix(m1_Duration)
mcmc_areas(m1_Duration_posterior,
           pars = c("b_Intercept",
                    "sd_ParticipantName__Intercept"),
                    prob = 0.8)
```



R - brms: Hoe zien de resultaten eruit?



R - brms: een voorspeller toevoegen

Duurt het langer om moeilijke vergelijkingen te maken?

$$Duration_{ij} = \beta_0 * Cons + \beta_1 * Distance + \mu_j + \epsilon_{ij}$$

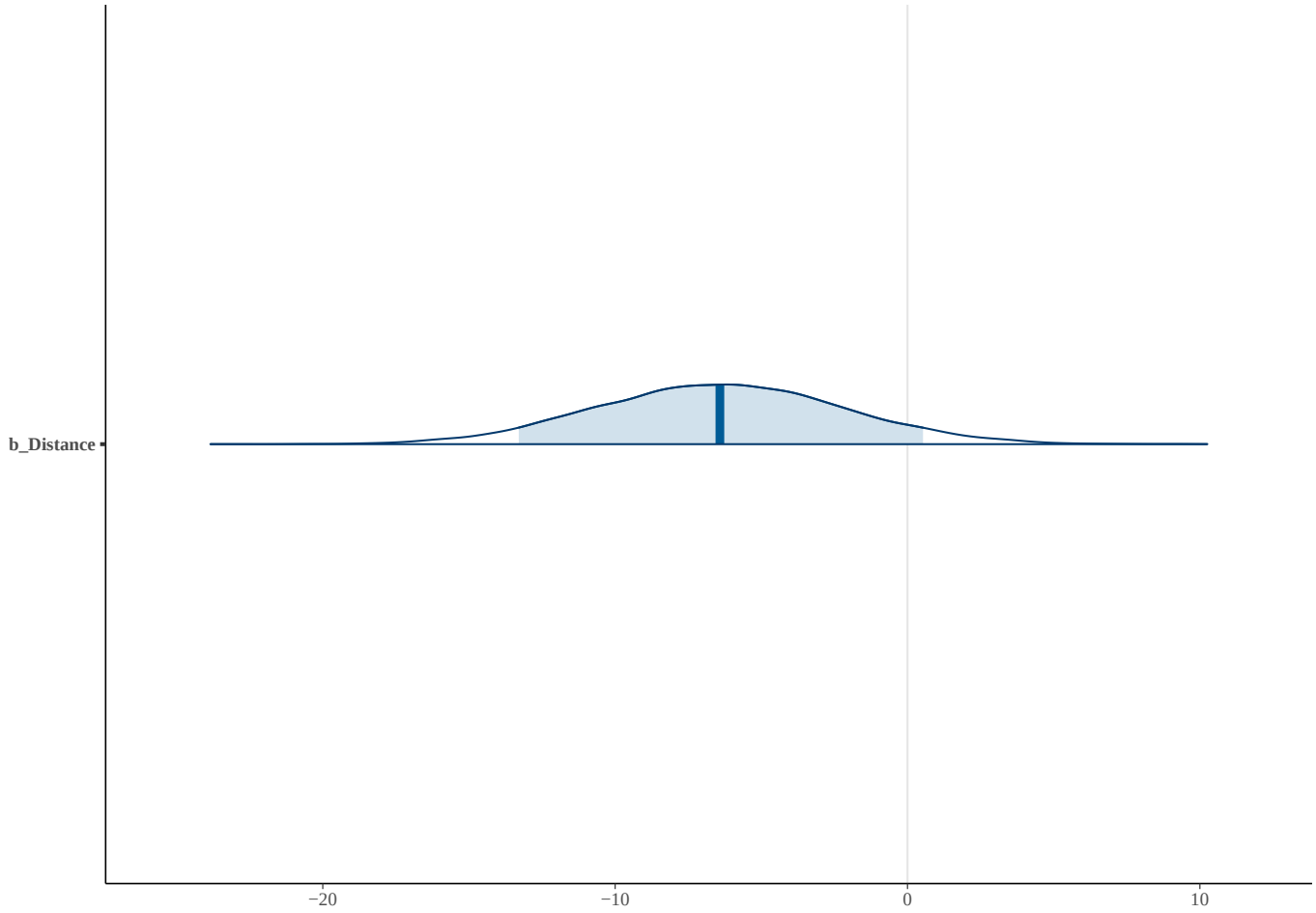
```
m2_Duration <- brm(Duration ~ Distance + (1|ParticipantName),
                   data = Data_Comp,
                   warmup = 1000, iter = 5000,
                   chains = 4, cores = 2,
                   seed = 1975)
```

R - brms: een voorspeller toevoegen

```
summary(m2_Duration)
```

```
##  Family: gaussian
##   Links: mu = identity; sigma = identity
## Formula: Duration ~ Distance + (1 | ParticipantName)
##    Data: Data_Comp (Number of observations: 190)
## Samples: 4 chains, each with iter = 5000; warmup = 1000; thin = 1;
##          total post-warmup samples = 16000
##
## Group-Level Effects:
## ~ParticipantName (Number of levels: 19)
##               Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
## sd(Intercept)    31.75      5.70    22.85    45.10       2289 1.00
##
## Population-Level Effects:
##           Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
## Intercept   212.83      7.83   197.30   227.78       1844 1.00
## Distance     -6.43      4.22   -14.61     1.83      10320 1.00
##
## Family Specific Parameters:
##       Estimate Est.Error l-95% CI u-95% CI Eff.Sample Rhat
## sigma    18.61      1.02    16.73    20.71       8944 1.00
##
## Samples were drawn using sampling(NUTS). For each parameter, Eff.Sample
## is a crude measure of effective sample size, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

R - brms: een voorspeller toevoegen



Conclusie??

ROPE...

Region Of Practical Equivalence = een bereik van parameterwaarden die we gelijk stellen aan het ontbreken van een effect

Kruschke (2018):

- ROPE definiëren -0.1 tot 0.1 in gestandaardiseerde termen
- is de helft van een klein effect volgens de vuistregels van Cohen (1988)
- parameterwaarden binnen die regio gelijk zetten aan equivalent aan het ontbreken van een effect

De HPDI+ROPE regel

Kruschke (2018) formuleert een *HPDI+ROPE-rule*:

- valt het 95% HPDI volledig buiten de ROPE $\rightarrow$ H_0 verwerpen
  - de 95% meest waarschijnlijke parameterwaarden zijn van een praktisch relevante effectgrootte
- valt het 95% HDI volledig in de ROPE $\rightarrow$ H_0 aanvaarden
  - de 95% meest waarschijnlijke parameterwaarden wijken niet af van een praktisch irrelevante effectgrootte
- valt het 95% HDI deels in de ROPE $\rightarrow$ 'onbeslist'
  - de 95% meest waarschijnlijke parameterwaarden zijn mogelijks van een praktisch relevante effectgrootte

HPDI+ROPE in R

```
library(sjstats)
kable(equi_test(m2_Duration))
```

term	decision	inside.rope	hdi.low	hdi.high
b_Intercept (*)	reject	0.000	197.27091	227.721668
b_Distance (*)	undecided	23.005	-14.57061	1.838754
sigma (*)	reject	0.000	16.69527	20.646825

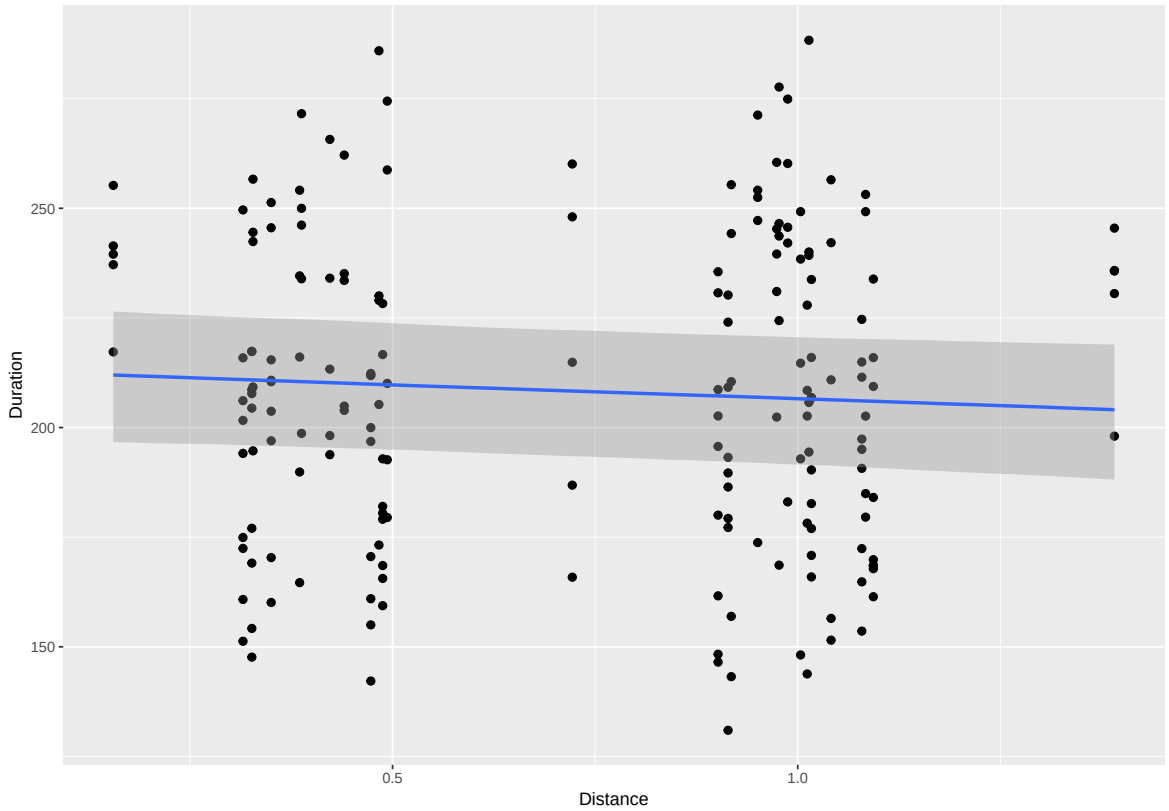
Non-linear model. . .

```
Data_Comp$Distance2 <- Data_Comp$Distance^2
m3_Duration <- brm(Duration ~ Distance + Distance2 + (1|ParticipantName),
                   data = Data_Comp,
                   warmup = 1000, iter = 5000,
                   chains = 4, cores = 2,
                   seed = 1975)
```

term	decision	inside.rope	hdi.low	hdi.high
b_Intercept (*)	reject	0.000	191.18620	233.95122
b_Distance (*)	undecided	11.223	-54.39140	47.64402
b_Distance2 (*)	undecided	16.532	-36.92919	33.39698
sigma (*)	reject	0.000	16.72675	20.74505

Grafiek van de voorspelde scores

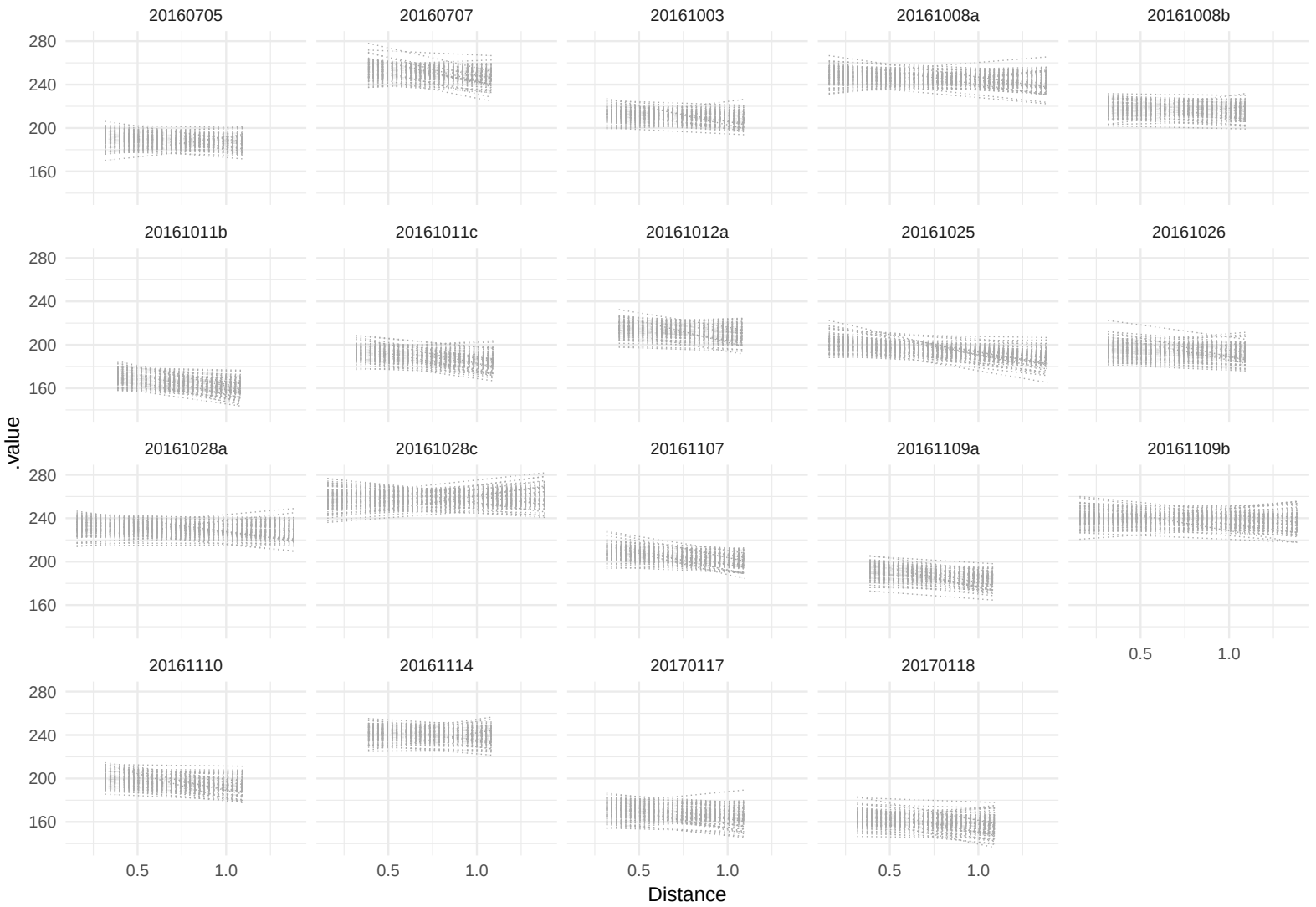
```
library(tidybayes)
XX <- marginal_effects(m2_Duration)
plot(XX,points= T)
```



Random slopes model. . .

```
m4_Duration <- brm(Duration ~ Distance + (1+Distance|ParticipantName),
                   data = Data_Comp,
                   warmup = 1000, iter = 5000,
                   chains = 4, cores = 2,
                   seed = 1975)
```

Mogelijke regressielijnen per persoon



Categorische voorspeller toevoegen

```
m5_Duration <- brm(Duration ~ Distance + Condition + (1+Distance|ParticipantName),
                   data = Data_Comp, sample_prior = T,
                   warmup = 1000, iter = 5000,
                   chains = 4, cores = 2,
                   seed = 1975)
```

Hypothesetoetsen

Hoeveel evidentie is er dat vergelijkingen uit Conditie 3 meer tijd in beslag namen dan vergelijkingen in Conditie 2?

```
hypothesis(m5_Duration, "ConditionCondition2 < ConditionCondition3")
```

```
## Hypothesis Tests for class b:
##                 Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 (ConditionConditi... < 0   -19.24      18.6     -Inf     11.2       5.97
##   Post.Prob Star
## 1      0.86
## ---
## '*': The expected value under the hypothesis lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

Hoeveel evidentie is er dat vergelijkingen uit Conditie 2 meer tijd in beslag namen dan vergelijkingen in Conditie 1?

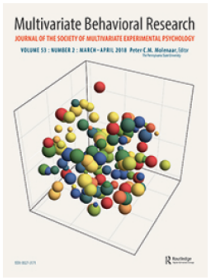
```
hypothesis(m5_Duration, "ConditionCondition2 > 0")
```

```
## Hypothesis Tests for class b:
##                 Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 (ConditionConditi... > 0    23.84     16.51    -3.07      Inf      13.41
##   Post.Prob Star
## 1      0.93
## ---
## '*': The expected value under the hypothesis lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.
```

Soorten priors?

- uninformed priors;
- weakly informed priors;
- informed priors;

Must read . . .



Multivariate Behavioral Research

Routledge
Taylor & Francis Group

ISSN: 0027-3171 (Print) 1532-7906 (Online) Journal homepage: https://www.tandfonline.com/loi/hmbr20

Bayesian PTSD-Trajectory Analysis with Informed Priors Based on a Systematic Literature Search and Expert Elicitation

Rens van de Schoot, Marit Sijbrandij, Sarah Depaoli, Sonja D. Winter, Miranda Olff & Nancy E. van Loey

To cite this article: Rens van de Schoot, Marit Sijbrandij, Sarah Depaoli, Sonja D. Winter, Miranda Olff & Nancy E. van Loey (2018) Bayesian PTSD-Trajectory Analysis with Informed Priors Based on a Systematic Literature Search and Expert Elicitation, Multivariate Behavioral Research, 53:2, 267-291, DOI: 10.1080/00273171.2017.1412293

To link to this article: https://doi.org/10.1080/00273171.2017.1412293

Artikel Rens van de Schoot et. al (2018)

Bij 'kleinere datasets' spelen priors een belangrijke rol!

Nood aan 'geïnformeerde' priors!

Wekwijze:

- Stap 1: systematische literatuurstudie;
- Stap 2: expert opinies eliciteren;

Kwaliteitscontrole bij Bayesiaanse analyses?

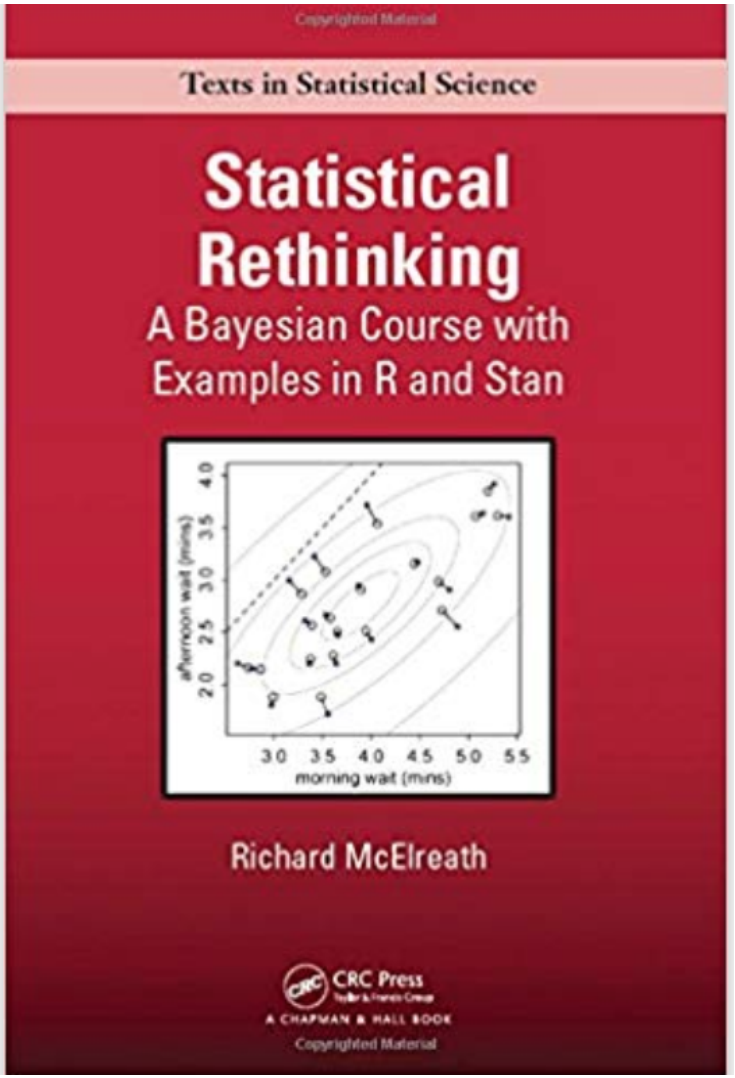
WAMBS - checklist:

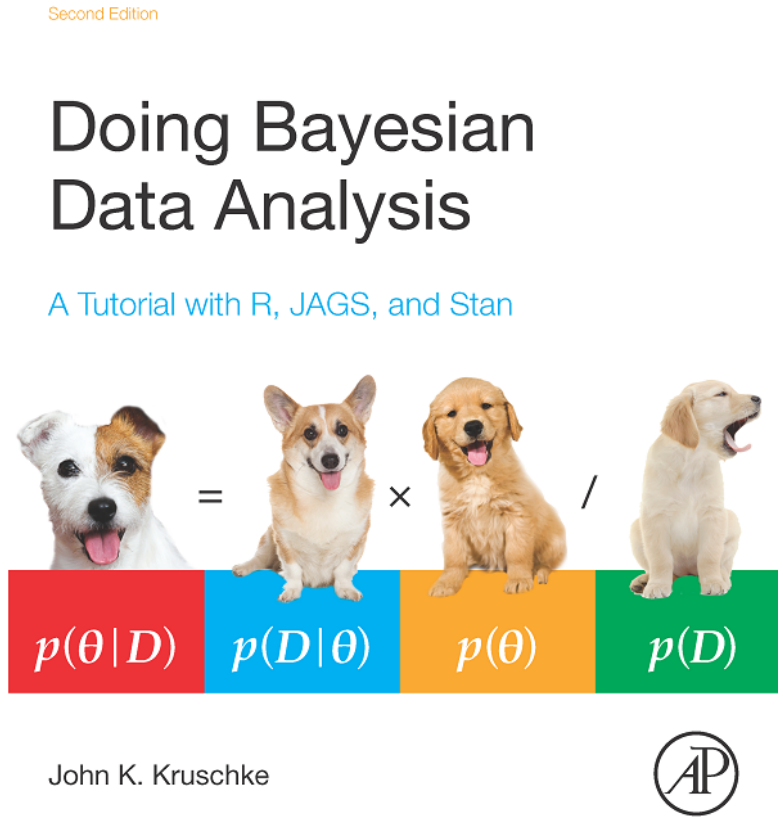
1. Do you understand the priors?
2. Does the trace-plot exhibit convergence?
3. Does convergence remain after doubling the iterations?
4. Does the histogram have enough information?
5. Do the chains exhibit a strong degree of autocorrelation?
6. Does the posterior distribution make substantive sense?
7. Do different specifications of the multivariate priors influence the results?
8. Is there a notable effect of the prior when compared with noninformative priors?
9. Are the results stable from a sensitivity analysis?
10. Is the Bayesian way of interpreting and reporting model results used?

Packages

- brms → de vertaler tussen R-taal en Stan
- bayesplot → zoals de naam het al zegt, om mooie plotjes te maken
- tidybayes → laat toe om allerlei informatie uit de posterior te halen en als data object (tidy format) weg te schrijven
- sjstats → bevat functies om resultaten samen te vatten, bv. voor de ROPE HPDI regel

To read

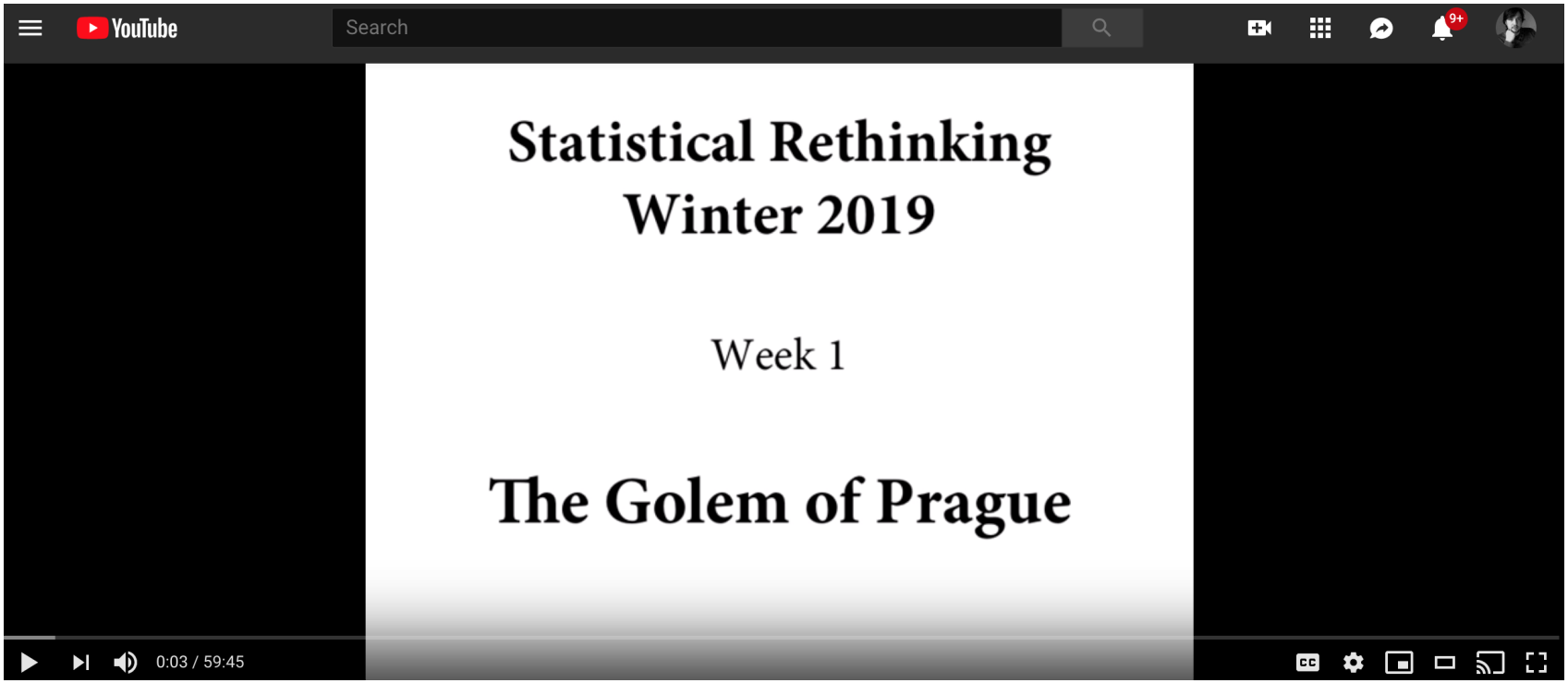

Copyrighted Material
Texts in Statistical Science
Statistical Rethinking
A Bayesian Course with Examples in R and Stan
afternoon wait (mins)
morning wait (mins)
Richard McElreath
CRC Press
A CHAPMAN & HALL BOOK
Copyrighted Material


Second Edition
Doing Bayesian Data Analysis
A Tutorial with R, JAGS, and Stan
p(θ|D)
p(D|θ)
p(θ)
p(D)
John K. Kruschke

Some url's (out of the many)

https://rdrr.io/cran/brms/: info brms pakket
https://mjskay.github.io/tidybayes/index.html: info tidybayes pakket
https://www.rensvandeschoot.com/bayesian-analysis-informed-priors/: site van rens vandeschoot
http://www.sumsar.net/: site van Rasmus Bååth

To see


YouTube
Search
Statistical Rethinking
Winter 2019
Week 1
The Golem of Prague
0:03 / 59:45

Tot slot: een oproep!