SBP in Young Adults

Simon Bond

2024-09-20

# Set up

First load in the data (I slightly edited it) and load some libraries

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(nlme)

##   
## Attaching package: 'nlme'  
##   
## The following object is masked from 'package:dplyr':  
##   
## collapse

library(rstan)

## Loading required package: StanHeaders  
##   
## rstan version 2.32.6 (Stan version 2.32.2)  
##   
## For execution on a local, multicore CPU with excess RAM we recommend calling  
## options(mc.cores = parallel::detectCores()).  
## To avoid recompilation of unchanged Stan programs, we recommend calling  
## rstan\_options(auto\_write = TRUE)  
## For within-chain threading using `reduce\_sum()` or `map\_rect()` Stan functions,  
## change `threads\_per\_chain` option:  
## rstan\_options(threads\_per\_chain = 1)  
##   
##   
## Attaching package: 'rstan'  
##   
## The following object is masked from 'package:tidyr':  
##   
## extract

library(bayesplot)

## This is bayesplot version 1.11.1  
## - Online documentation and vignettes at mc-stan.org/bayesplot  
## - bayesplot theme set to bayesplot::theme\_default()  
## \* Does \_not\_ affect other ggplot2 plots  
## \* See ?bayesplot\_theme\_set for details on theme setting

df <- read.csv(file="Central SBP in young ISH\_v2.csv") %>%   
 cctu::clean\_names() %>%   
 mutate(se=sd/sqrt(n),  
 weights=se^2  
 )  
summary(df)

## study n mean sd   
## Length:13 Min. : 6.0 Min. : 94.0 Min. : 3.000   
## Class :character 1st Qu.: 21.0 1st Qu.:116.0 1st Qu.: 6.000   
## Mode :character Median : 33.0 Median :120.0 Median : 7.000   
## Mean : 136.5 Mean :120.5 Mean : 7.154   
## 3rd Qu.: 57.0 3rd Qu.:124.0 3rd Qu.: 9.000   
## Max. :1308.0 Max. :141.0 Max. :13.000   
## group se weights   
## Length:13 Min. :0.1936 Min. :0.03746   
## Class :character 1st Qu.:0.9045 1st Qu.:0.81818   
## Mode :character Median :1.2452 Median :1.55046   
## Mean :1.2585 Mean :1.91801   
## 3rd Qu.:1.6977 3rd Qu.:2.88235   
## Max. :2.4495 Max. :6.00000

# Basic analysis

This start with the assumption that in the hypertensive studies, and in the optimal group. The standard deviation is , and is assumed here to be a known fixed parameter. Also there is no heterogeniety between studies, akin to assuming you simply took seperate samples with identicial treatments, from and identical population, for each study.

A basic frequentest analysis can be done that uses the fixed standard deviation.

## Generalized least squares fit by REML  
## Model: mean ~ group   
## Data: df   
## AIC BIC logLik  
## 450.3497 451.1455 -223.1749  
##   
## Variance function:  
## Structure: fixed weights  
## Formula: ~weights   
##   
## Coefficients:  
## Value Std.Error t-value p-value  
## (Intercept) 120.1999 0.3061368 392.6346 0  
## groupoptimal -26.1999 0.3621899 -72.3375 0  
##   
## Correlation:   
## (Intr)  
## groupoptimal -0.845  
##   
## Standardized residuals:  
## Min Q1 Med Q3 Max   
## -5.9359510 -4.6431626 -0.1605387 3.8001014 13.6168627   
##   
## Residual standard error: 1   
## Degrees of freedom: 13 total; 11 residual

# Extensions

We want to relax the assumptions made above.

1. acknowledge there is sample variability in the standard deviations. IN fact theory says
2. Allow the location to vary randomly between studies by
3. Allow also the between-patient SD to vary between studies

These need Bayesian MCMC software to fit. The R code is shown below, which in turn relies on individual files, contained in the folder ( m0.stan, m1.stan, m2.stan, m4.stan).

df\_stan <- list(  
 N = nrow(df),  
 y=df$mean,  
 x=1\*(df$group=="optimal"),  
 se=df$se,  
 sample=df$n  
)  
  
  
fit0 <- stan(  
 file = "m0.stan", # Stan program  
 data = df\_stan, # named list of data  
 chains = 4, # number of Markov chains  
 warmup = 1000, # number of warmup iterations per chain  
 iter = 2000, # total number of iterations per chain  
 cores = 2, # number of cores (could use one per chain)  
 refresh = 100 # no progress shown  
)  
   
fit1 <- stan(  
 file = "m1.stan", # Stan program  
 data = df\_stan, # named list of data  
 chains = 4, # number of Markov chains  
 warmup = 1000, # number of warmup iterations per chain  
 iter = 2000, # total number of iterations per chain  
 cores = 2, # number of cores (could use one per chain)  
 refresh = 100 # no progress shown  
)  
save.image("fits.Rdata")  
load("fits.Rdata")  
  
fit2 <- stan(  
 file = "m2.stan", # Stan program  
 data = df\_stan, # named list of data  
 chains = 4, # number of Markov chains  
 warmup = 1000, # number of warmup iterations per chain  
 iter = 2000, # total number of iterations per chain  
 cores = 2, # number of cores (could use one per chain)  
 refresh = 100 # no progress shown  
)  
  
summary(fit2, pars=c("mu","delta","sigma\_within","sigma\_between"))  
  
renv::install("bayesplot")  
library(bayesplot)  
mcmc\_areas(fit3,pars=my\_pars)  
mcmc\_areas(fit3,pars="sigma\_within")  
mcmc\_areas(fit3,pars="delta")  
traceplot(fit3, pars=my\_pars)  
summary(fit3, c(my\_pars,"sigma\_within\_se"))  
summary(fit2, pars=c("mu","delta","sigma\_within","sigma\_between"))  
  
  
  
fit4 <- stan(  
 file = "m4.stan", # Stan program  
 data = df\_stan, # named list of data  
 chains = 4, # number of Markov chains  
 warmup = 2000, # number of warmup iterations per chain  
 iter = 10000, # total number of iterations per chain  
 cores = 2, # number of cores (could use one per chain)  
 refresh = 500 # no progress shown  
)  
  
save.image("fits.Rdata")

We ran the code earliear (each fit takes abotu 5 mins) saved the results, which is now re-loaded

In turn this

1. Replicates the analysis above
2. Acknowledges sampling variability in the standard deviations
3. Adds in between-study heterogeneity in the location
4. Adds in between-study heterogeneity in the variance

load("fits.Rdata")  
  
  
summary(fit0, pars=c("mu","delta"))$summary %>% knitr::kable(digits = 2)

|  | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% | n\_eff | Rhat |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| mu | 120.2 | 0.01 | 0.31 | 119.63 | 119.99 | 120.20 | 120.41 | 120.82 | 1205.53 | 1 |
| delta | -26.2 | 0.01 | 0.37 | -26.92 | -26.46 | -26.19 | -25.95 | -25.51 | 1213.21 | 1 |

summary(fit1, pars=c("mu","delta","sigma\_within"))$summary%>% knitr::kable(digits = 2)

|  | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% | n\_eff | Rhat |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| mu | 121.44 | 0.01 | 0.39 | 120.67 | 121.17 | 121.43 | 121.70 | 122.20 | 1501.60 | 1 |
| delta | -27.44 | 0.01 | 0.46 | -28.33 | -27.75 | -27.44 | -27.13 | -26.52 | 1509.98 | 1 |
| sigma\_within | 8.37 | 0.00 | 0.14 | 8.10 | 8.27 | 8.36 | 8.46 | 8.64 | 2268.22 | 1 |

summary(fit2, pars=c("mu","delta","sigma\_within", "sigma\_between"))$summary%>% knitr::kable(digits = 2)

|  | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% | n\_eff | Rhat |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| mu | 122.68 | 0.17 | 3.13 | 116.59 | 120.72 | 122.72 | 124.54 | 129.02 | 358.25 | 1.01 |
| delta | -28.38 | 0.33 | 10.79 | -49.03 | -35.17 | -28.52 | -21.66 | -7.11 | 1051.24 | 1.00 |
| sigma\_within | 7.59 | 0.00 | 0.13 | 7.33 | 7.50 | 7.58 | 7.67 | 7.84 | 1870.97 | 1.00 |
| sigma\_between | 9.99 | 0.08 | 2.57 | 6.29 | 8.17 | 9.50 | 11.31 | 16.24 | 1120.60 | 1.00 |

summary(fit4, pars=c("mu","delta","sigma\_within","sigma\_between","sigma\_within\_se"))$summary%>% knitr::kable(digits = 2)

|  | mean | se\_mean | sd | 2.5% | 25% | 50% | 75% | 97.5% | n\_eff | Rhat |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| mu | 122.66 | 0.05 | 2.91 | 116.91 | 120.81 | 122.65 | 124.54 | 128.39 | 3880.29 | 1 |
| delta | -28.79 | 0.09 | 10.51 | -49.80 | -35.30 | -28.79 | -22.20 | -7.83 | 13261.37 | 1 |
| sigma\_within | 7.94 | 0.00 | 0.71 | 6.66 | 7.47 | 7.89 | 8.36 | 9.51 | 20448.95 | 1 |
| sigma\_between | 9.77 | 0.02 | 2.53 | 6.19 | 8.01 | 9.32 | 11.04 | 16.00 | 17494.11 | 1 |
| sigma\_within\_se | 2.23 | 0.00 | 0.62 | 1.33 | 1.81 | 2.13 | 2.54 | 3.70 | 18298.92 | 1 |

# Comments

These all assume a flat prior for all the parameters, except for the last model for sigma\_within. This is the lazy unthinking working assumption. The last model, was tripped up by this allowing unrealistic large values, and so a gamma distribution prior was added, with an expectation of 10, and 95% CI of 3.3 – 20.5, otherwise convergence issue arose.

There is evidence to support heterogeniety being present for both the location and variance.

The difference between the groups is statistically significant (for the frequentist first model). The other models are Bayesian, and so p-values don’t exist, but give credible intervals well away from zero. They are all centred around 26-28 point estimates for the difference, the optimal group being lower. The credible intervals widen, but nonethelss exclude 0, for the models that incorporate the heterogeniety in location, which has a standard deviation of 7/9 (models 2/4), comparable in size to the between-patient SD.