Computational Statistics & Probability

Lab 5 - Multilevel Models

Fall 2022

```
#knitr::opts_chunk$set(echo = TRUE)
#answer_key <- FALSE
library(rethinking)
```

```
## Loading required package: rstan

## Loading required package: StanHeaders

##
## rstan version 2.26.13 (Stan version 2.26.1)

## 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)

## Loading required package: cmdstanr

## This is cmdstanr version 0.5.3

## - CmdStanR documentation and vignettes: mc-stan.org/cmdstanr

## - CmdStan path: /Users/neelesh/.cmdstan/cmdstan-2.30.1

## - CmdStan version: 2.30.1

##
## A newer version of CmdStan is available. See ?install_cmdstan() to install it.
## To disable this check set option or environment variable CMDSTANR_NO_VER_CHECK=TRUE.

## Loading required package: parallel

## rethinking (Version 2.21)

##
## Attaching package: 'rethinking'

## The following object is masked from 'package:rstan':
##
##     stan

## The following object is masked from 'package:stats':
##
##     rstudent
```

```
library(latex2exp)
library(DiagrammeR)
library(knitr)
```

1. Bangladesh Fertility Survey

In 1980, a typical Bengali woman could have 5 or more children in her lifetime. By the year 2000, a typical Bengali woman had only 2 or 3 children. An historical data set of 1934 Bengali women from the 1988 Bangladesh Fertility Survey, named `bangladesh` in the `rethinking` package, can be used to explore the adoption of contraception over this period within different districts of Bangladesh.

- `district`: the ID number of the administrative district each woman lives in
- `use.contraception`: an indicator variable (with values 0 or 1) denoting whether each woman uses contraception.

a) A large portion of time in applied statistics is devoted to inspecting and formatting your data, a task you have been spared thus far. But this data set has an issue with the cluster variable `district` that needs your attention. Load your data and inspect the cluster variable `district`. Can you spot the problem?

```
library(rethinking)
data(bangladesh)
d <- bangladesh
sort(unique(d$district))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
## [51] 51 52 53 55 56 57 58 59 60 61
```

```
library(rethinking)
data(bangladesh)
d <- bangladesh
# `district` is not contiguous: 54 is missing
sort(unique(d$district))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
## [51] 51 52 53 55 56 57 58 59 60 61
```

```
# The problem is that there are 60 districts, but the indices 1-53, 55-61 are
# used:  that is, '54' is missing.

# make `district` contiguous
d$district_id <- as.integer(as.factor(d$district))
sort(unique(d$district_id))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
## [51] 51 52 53 54 55 56 57 58 59 60
```

b) Correct the problem with `district` and save as a new variable, `district_id`.

```
# To use `district` as a contiguous index variable, we need to use the coercion
# function `as.factor` to return a sequence encoding the levels of
# `district`, then apply `as.integer` to ensure the data type of these levels
# are integers.  This is saved as a new variable, `district_id`:

d$district_id <- as.integer(as.factor(d$district))
sort(unique(d$district_id))
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
## [26] 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
## [51] 51 52 53 54 55 56 57 58 59 60
```

c) Turn now to predicting `contraception`, clustered by `district_id`. Fit both (1) a traditional fixed-effect model and (2) a varying-effects model.

```
set.seed(1776)
# trimmed data list
data_slim <- list(
    use_contraception = d$use.contraception,
    district_id = d$district_id )

# fixed effects model
m1_fixed <- ulam(
  alist(
    use_contraception ~ dbinom( 1 , p ),
    logit(p) <- a_district[district_id],
    a_district[district_id] ~ dnorm(0,10)
  ), data=data_slim, chains=1, refresh=0)     # set chains to 1 for speed
```

```
## Warning in '/var/folders/wx/1_76tj0s15gc4yxmndvw4l_00000gn/T/RtmpmrD1Jy/model-a702fb59453.stan', lin
##     of arrays by placing brackets after a variable name is deprecated and
##     will be removed in Stan 2.32.0. Instead use the array keyword before the
##     type. This can be changed automatically using the auto-format flag to
##     stanc
## Warning in '/var/folders/wx/1_76tj0s15gc4yxmndvw4l_00000gn/T/RtmpmrD1Jy/model-a702fb59453.stan', lin
##     of arrays by placing brackets after a variable name is deprecated and
##     will be removed in Stan 2.32.0. Instead use the array keyword before the
##     type. This can be changed automatically using the auto-format flag to
##     stanc

## Running MCMC with 1 chain, with 1 thread(s) per chain...
##
## Chain 1 finished in 3.7 seconds.
```

```
# varying effects model
m1_varying <- ulam(
  alist(
    use_contraception ~ dbinom( 1 , p ),
    logit(p) <- a+ a_district[district_id],
    a ~ dnorm(0,10),
    a_district[district_id] ~ dnorm(0,sigma),
    sigma ~ dcauchy(0,1)
  ), data=data_slim, chains=1, refresh=0)    # set chains to 1 for speed
```

```
## Warning in '/var/folders/wx/1_76tj0s15gc4yxmndvw4l_00000gn/T/RtmpmrD1Jy/model-a703d09a26e.stan', lin
##     of arrays by placing brackets after a variable name is deprecated and
##     will be removed in Stan 2.32.0. Instead use the array keyword before the
##     type. This can be changed automatically using the auto-format flag to
##     stanc
## Warning in '/var/folders/wx/1_76tj0s15gc4yxmndvw4l_00000gn/T/RtmpmrD1Jy/model-a703d09a26e.stan', lin
##     of arrays by placing brackets after a variable name is deprecated and
##     will be removed in Stan 2.32.0. Instead use the array keyword before the
##     type. This can be changed automatically using the auto-format flag to
```

```
##     stanc
## Running MCMC with 1 chain, with 1 thread(s) per chain...
## Chain 1 Informational Message: The current Metropolis proposal is about to be rejected because of the
## Chain 1 Exception: normal_lpdf: Scale parameter is 0, but must be positive! (in '/var/folders/wx/1_7
## Chain 1 If this warning occurs sporadically, such as for highly constrained variable types like cova
## Chain 1 but if this warning occurs often then your model may be either severely ill-conditioned or m
## Chain 1
## Chain 1 finished in 3.4 seconds.
```

d) What are the *predicted* probabilities of contraception use for each district? You can do this using `link`.

```
# First, create a vector `pred.dist` from the district ids:
set.seed(1776)
pred.dist <- list(district_id=1:60)

# Now we use `link` to get predictions from any linear model associated with the
# outcome variable:

pred_f <- link(m1_fixed, data=pred.dist)
pred_v <- link(m1_varying, data=pred.dist)

# For the fixed effect model and the varying effect model, a prediction for
# contraception use for each district is computed from 1000 samples from the
# respective posterior distributions.

# For example,

str(pred_f)
```

```
##  num [1:500, 1:60] 0.314 0.214 0.228 0.228 0.258 ...
```

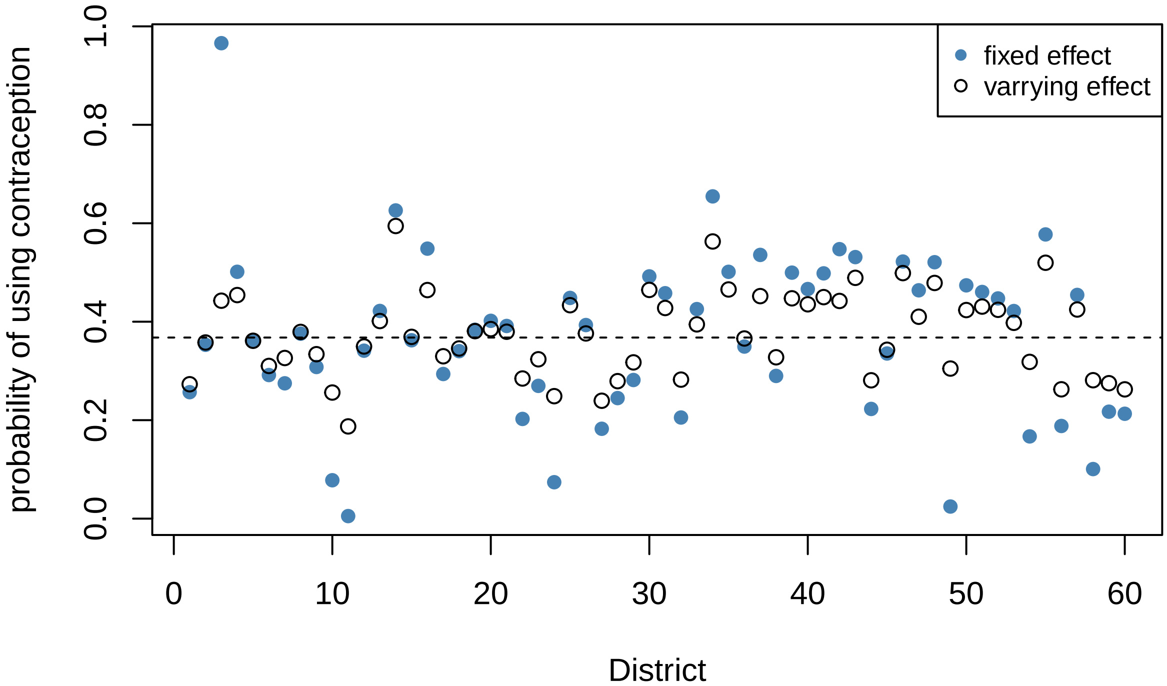
```
# yields 1000 probabilities sampled from the fixed effect posterior for each of
# the 60 districts.

# We now may follow the usual procedure to calculate the average probabilities
# of each district for the fixed-effect and varrying-effect model:

p_f.mean <- apply( pred_f, 2, mean)
p_v.mean <- apply( pred_v, 2, mean)
```

e) Plot the predicted proportions of women in each district using contraception, for both the fixed effects model and varrying effects model.

```
plot( 1:60 , p_f.mean, col="steelblue", pch=16, xlab="District",
      ylab="probability of using contraception")
points(1:60, p_v.mean )
abline( h=logistic(coef( m1_varying)[1]) ,
        lty=2 )  # plots line for fixed `a`
legend("topright",c("fixed effect","varrying effect"),
       cex=.8,col=c("steelblue","black"),pch=c(16,1))
```




```
# DISCUSSION: For each district there are two estimated probabilities of contraception
# use, one generated by the fixed effects model `p_f` (solid blue circles) and one
# generated by the varying effects model `p_v` (open black circles). The dashed line
# is the average proportion of women using contraception in the entire population
# of all 60 districts.

# For each district, observe that the estimated probabilities of the varying effects
# model (black circles) are *always* closer to the population average (dashed line)
# than the fixed effect estimates.  Some of these differences are extreme:

# maximum absolute difference:
max(abs(p_f.mean - p_v.mean))
```

```
## [1] 0.5231038
```

```
# which district has this max difference
which.max(abs(p_f.mean - p_v.mean))
```

```
## [1] 3
```

```
# District 3 has a fixed estimate of 0.956, P(C=1) = 0.956
p_f.mean[3]
```

```
## [1] 0.9657917
```

```
# whereas that district has a vary-effects estimate of 0.451
p_v.mean[3]
```

```
## [1] 0.4426879
```

```
# Conversely, the fixed effect model predicts that no one in District 11 uses
# contraception
min(p_f.mean)
```

```
## [1] 0.005192211
```

```
which.min(p_f.mean)

## [1] 11

# yet the varying effect model predicts estimates the probability to be about
# 31%:
p_v.mean[49]

## [1] 0.30485

# Why are the estimates for District 3 and District 11 so different? In District
# 3 there are only two women:
sum(d$district_id == 3)

## [1] 2

# both of whom use contraception:
d[d$district_id==3, ]

##     woman district use.contraception living.children age.centered urban
## 138   138        3                 1               4      -3.5599     1
## 139   139        3                 1               1      -8.5599     1
##     district_id
## 138           3
## 139           3

# whereas in District 11 none of the 21 women use contraception:
d[d$district_id==11, ]

##     woman district use.contraception living.children age.centered urban
## 365   365       11                 0               1      -9.5599     0
## 366   366       11                 0               1      -8.5599     0
## 367   367       11                 0               2      -5.5599     0
## 368   368       11                 0               2      18.4400     0
## 369   369       11                 0               1      -8.5599     0
## 370   370       11                 0               1      -9.5599     0
## 371   371       11                 0               1     -12.5590     0
## 372   372       11                 0               1       3.4400     0
## 373   373       11                 0               1      -8.5599     0
## 374   374       11                 0               4      19.4400     0
## 375   375       11                 0               1      -3.5599     0
## 376   376       11                 0               2      -5.5599     0
## 377   377       11                 0               2       2.4400     0
## 378   378       11                 0               2       0.4400     0
## 379   379       11                 0               1     -11.5590     0
## 380   380       11                 0               3      -2.5599     0
## 381   381       11                 0               4       2.4400     0
## 382   382       11                 0               2      -8.5599     0
## 383   383       11                 0               1      -8.5599     0
## 384   384       11                 0               1      -6.5600     0
## 385   385       11                 0               4      18.4400     0
##     district_id
## 365          11
## 366          11
## 367          11
## 368          11
## 369          11
```

```
## 370          11
## 371          11
## 372          11
## 373          11
## 374          11
## 375          11
## 376          11
## 377          11
## 378          11
## 379          11
## 380          11
## 381          11
## 382          11
## 383          11
## 384          11
## 385          11
```

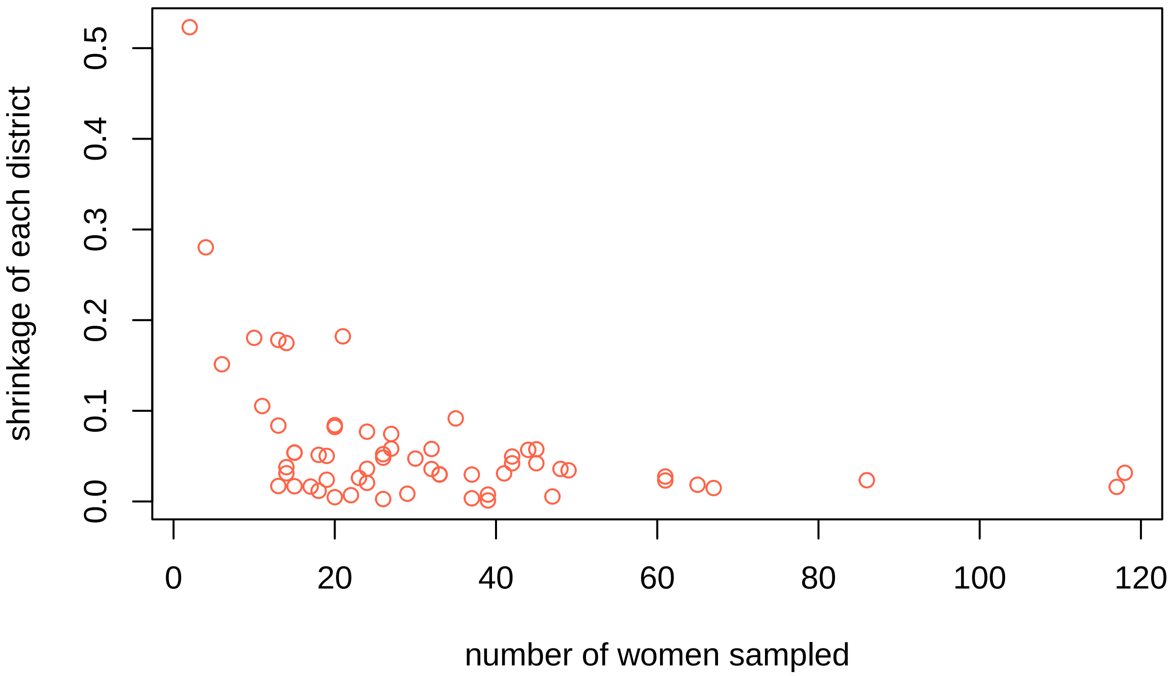
```r
# The varying effect model was able to *pool* information from other districts
# to give a better estimate than "all" or "none", which the fixed effects model
# essentially does. Further, the effect of *shrinkage* the varying effect model
# introduces depends on the number of data points within each District.  The
# model is more aggressive in the case of District 3 (2 observations) than
# it is with District 11 (21 observations).

# To visualize this relationship between the degree of shinkage and the number
# of women in a district, let's first compute the number of women sampled from
# each district in the dataset:

n_by_district <- sapply( 1:60 ,
    function(indx) length(d$district_id[d$district_id==indx]) )

# Next, compute shrinkage for each district:
shrinkage <- abs( p_f.mean - p_v.mean )
```

```r
# compute the number of women sampled in each district
plot( n_by_district , shrinkage , col="tomato" ,
    xlab="number of women sampled" , ylab="shrinkage of each district" )
```



shrinkage of each district
0.0
0.1
0.2
0.3
0.4
0.5
0
20
40
60
80
100
120
number of women sampled