

# 18233038\_WatanabeMotohiro\_PS2

18233038\_WatanabeMotohiro

2023-11-15

## preparation

```
library(rethinking)

## Loading required package: cmdstanr
## This is cmdstanr version 0.6.1
## - CmdStanR documentation and vignettes: mc-stan.org/cmdstanr
## - Use set_cmdstan_path() to set the path to CmdStan
## - Use install_cmdstan() to install CmdStan
## Loading required package: posterior
## This is posterior version 1.5.0
##
## Attaching package: 'posterior'
## The following objects are masked from 'package:stats':
##
##     mad, sd, var
## The following objects are masked from 'package:base':
##
##     %in%, match
## Loading required package: parallel
## rethinking (Version 2.40)
##
## Attaching package: 'rethinking'
## The following object is masked from 'package:stats':
##
##     rstudent
data(foxes)
df <- foxes
```

## 5H1

```
m1 <- map(
  alist(
    weight ~ dnorm( mu , sigma ) ,
    mu <- a + b_area*area ,
```

```

a ~ dnorm(0,100),
b_area ~ dnorm(0,10),
sigma ~ dunif(0,50)
) , data=df )

m2 <- map(
  alist(
    weight ~ dnorm( mu , sigma ) ,
    mu <- a + b_group*groupsize ,
    a ~ dnorm(0,100),
    b_group ~ dnorm(0,10),
    sigma ~ dunif(0,50)
  ) , data=df )

precis(m1)

```

```

##           mean          sd      5.5%      94.5%
## a          4.45087277 0.39081774  3.8262705 5.0754750
## b_area     0.02485756 0.11838640 -0.1643468 0.2140619
## sigma      1.17868509 0.07738429  1.0550100 1.3023601

```

```

precis(m2)

##           mean          sd      5.5%      94.5%
## a          5.0678949 0.32488620  4.5486640 5.58712575
## b_group    -0.1238819 0.07052049 -0.2365873 -0.01117656
## sigma      1.1635303 0.07638932  1.0414454 1.28561518

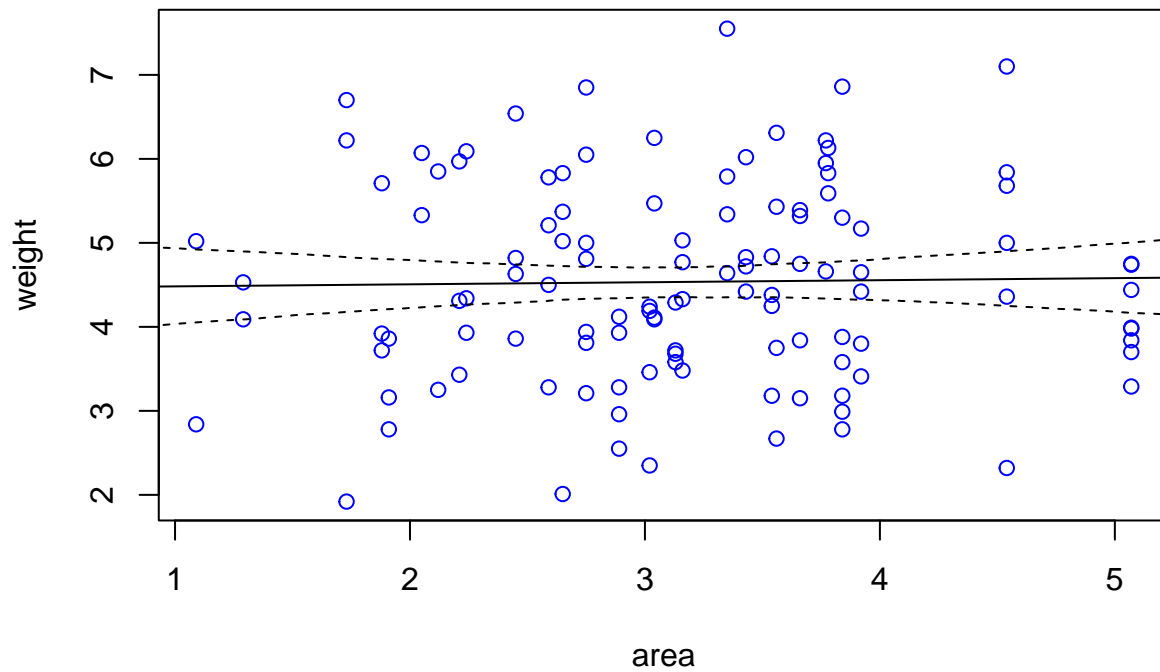
```

(fig.1)area&weight

```

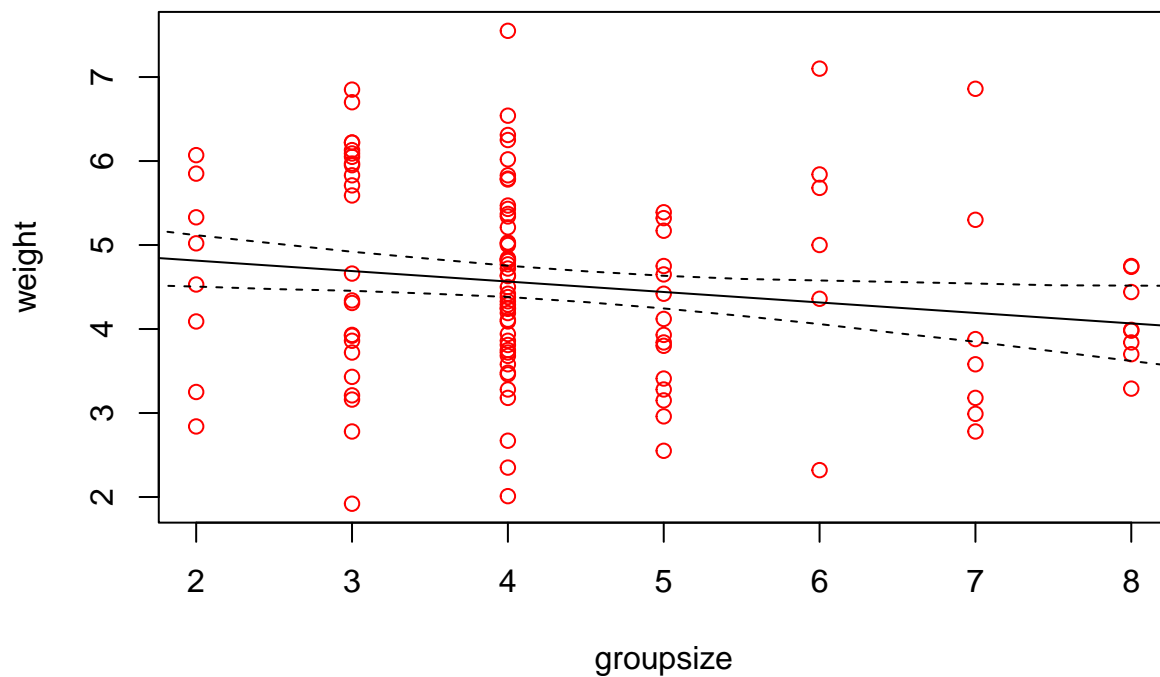
x.seq <- seq(from=0,to=6,by=0.025)
mu <- link( m1 , data=list(area=x.seq) )
mu.mean <- apply( mu , 2 , mean )
mu.ci <- apply( mu , 2 , PI )
plot( weight ~ area , data=df , col="blue" )
lines( x.seq , mu.mean )
lines( x.seq , mu.ci[1,] , lty=2 )
lines( x.seq , mu.ci[2,] , lty=2 )

```



(fig.2)groupsize&weight

```
x.seq <- seq(from=1,to=9,by=0.5)
mu <- link( m2 , data=list(groupsize=x.seq) )
mu.mean <- apply( mu , 2 , mean )
mu.ci <- apply( mu , 2 , PI )
plot( weight ~ groupsize , data=df , col="red" )
lines( x.seq , mu.mean )
lines( x.seq , mu.ci[1,] , lty=2 )
lines( x.seq , mu.ci[2,] , lty=2 )
```



Groupsize is more important for predicting fox body weight than area.

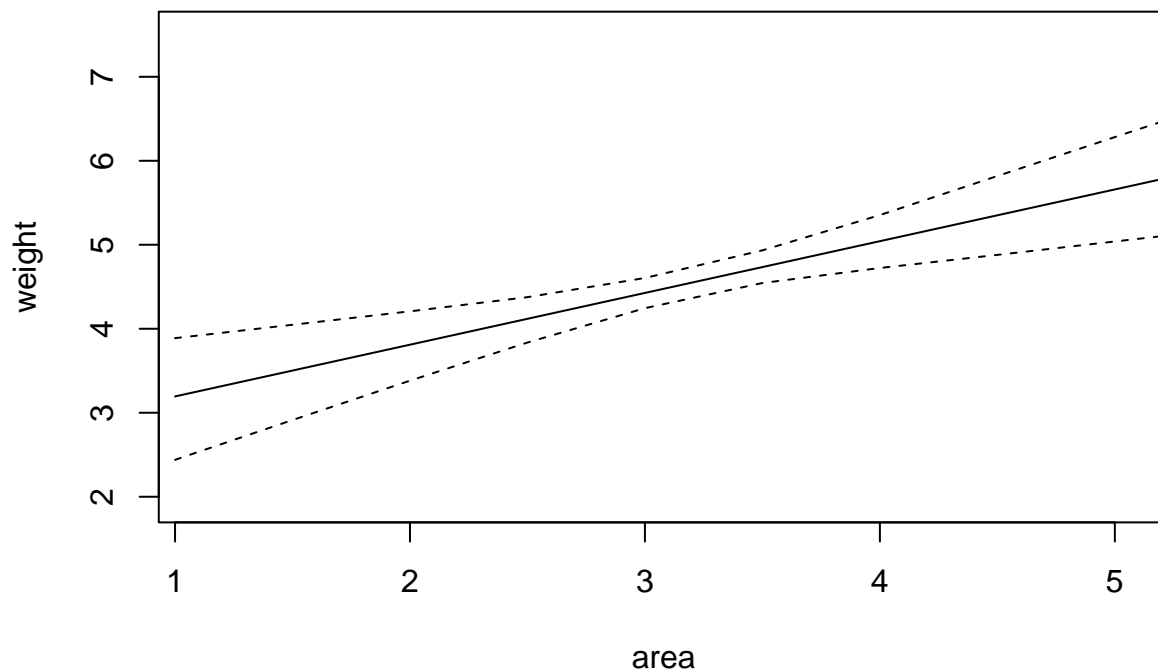
## 5H2

```
m3 <- map(
  alist(
    weight ~ dnorm( mu , sigma ) ,
    mu <- a + b_area*area + b_group*groupsize ,
    a ~ dnorm(0,100),
    c(b_area,b_group) ~ dnorm(0,10),
    sigma ~ dunif(0,50)
  ) , data=df )
precis(m3)
```

```
##           mean          sd      5.5%      94.5%
## a          4.4503659 0.37084848  3.8576784  5.0430534
## b_area     0.6178377 0.20009183  0.2980523  0.9376231
## b_group    -0.4324058 0.12074262 -0.6253758 -0.2394358
## sigma      1.1184507 0.07342969  1.0010959  1.2358056
```

(fig.3)groupsize fixed

```
area.seq <- seq(from=1,to=6,by=0.5)
pred.dat <- data.frame( area=area.seq , groupsize=mean(df$groupsize) )
mu <- link( m3 , data=pred.dat )
mu.mean <- apply( mu , 2 , mean )
mu.ci <- apply( mu , 2 , PI )
plot( weight ~ area , data=df , type="n" )
lines( area.seq , mu.mean )
lines( area.seq , mu.ci[1,] , lty=2 )
lines( area.seq , mu.ci[2,] , lty=2 )
```



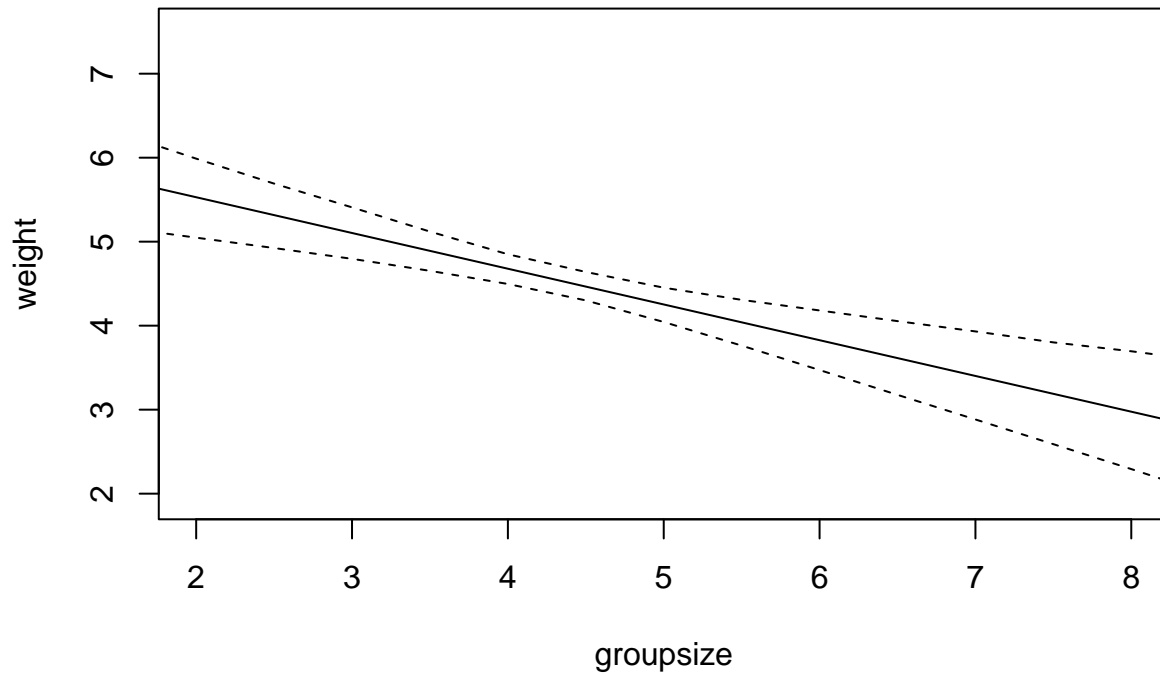
(fig.4)area fixed

```
gs.seq <- seq(from=1,to=9,by=0.5)
pred.dat <- data.frame( area=mean(df$area) , groupsize=gs.seq )
```

```

mu <- link( m3 , data=pred.dat )
mu.mean <- apply( mu , 2 , mean )
mu.ci <- apply( mu , 2 , PI )
plot( weight ~ groupsize , data=df , type="n" )
lines( gs.seq , mu.mean )
lines( gs.seq , mu.ci[1,] , lty=2 )
lines( gs.seq , mu.ci[2,] , lty=2 )

```



Importance of area appeared because of collinearity with groupsize. And, importance of groupsize still is clear.

### 5H3

```

m4 <- map(
  alist(
    weight ~ dnorm( mu , sigma ) ,
    mu <- a + b_food*avgfood + b_group*groupsize ,
    a ~ dnorm(0,100),
    c(b_food,b_group) ~ dnorm(0,10),
    sigma ~ dunif(0,50)
  ) , data=df )

m5 <- map(
  alist(
    weight ~ dnorm( mu , sigma ) ,
    mu <- a + b_food*avgfood + b_group*groupsize + b_area*area ,
    a ~ dnorm(0,100),
    c(b_food,b_group,b_area) ~ dnorm(0,10),
    sigma ~ dunif(0,50)
  ) , data=df )

```

(a)

```
precis(m4)
```

```
##           mean      sd      5.5%      94.5%
## a          4.1371741 0.4307945  3.4486813  4.8256669
## b_food     3.7707499 1.2041804  1.8462370  5.6952628
## b_group    -0.5620558 0.1554305 -0.8104637 -0.3136479
## sigma      1.1166282 0.0733130  0.9994599  1.2337966
```

Putting avgfood into model is favorable because coefficients of avgfood is estimated higher than area.

(b)

```
precis(m5)
```

```
##           mean      sd      5.5%      94.5%
## a          4.0719103 0.42792512  3.388003309  4.7558173
## b_food     2.4570830 1.43770749  0.159348760  4.7548173
## b_group    -0.6039466 0.15584600 -0.853018647 -0.3548746
## b_area      0.3896126 0.23847218  0.008487991  0.7707372
## sigma      1.1043540 0.07250498  0.988477060  1.2202310
```

This model may has multicollinearity problem because vast area can be thought to have much food. So, result of m5 may be biased.