

# Chapter 5

# Example data

Divorce rate in the US Model:

$$d_i \sim N(\mu_i, \sigma), \\ \mu_i = \alpha + r_i \beta_r + a_i \beta_a$$

- $d_i$  - divorce rate by state
- $r_i$  - marriage rate by state
- $a_i$  - median age at marriage

Priors will be chosen non-informative.

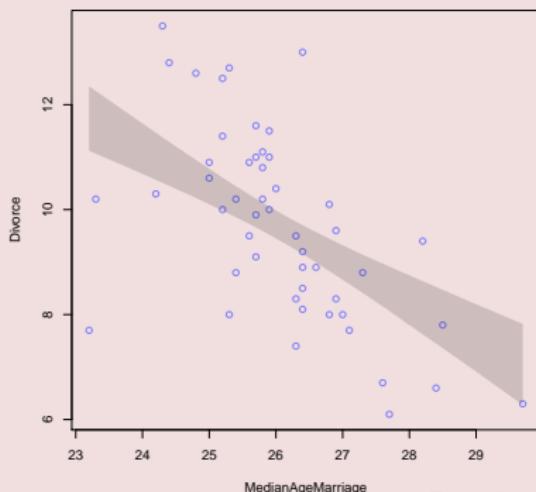
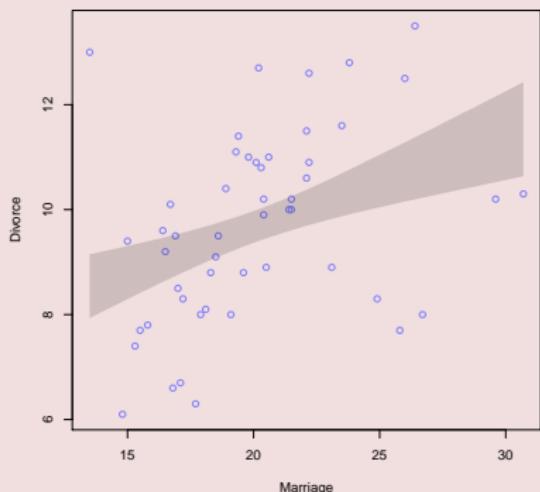
# Fitting the models separately

$$d_i \sim N(\mu_i, \sigma),$$

$$\mu_i = \alpha + r_i \beta_r$$

$$d_i \sim N(\mu_i, \sigma),$$

$$\mu_i = \alpha + a_i \beta_a$$



# fitting model

$$d_i \sim N(\mu_i, \sigma)$$

$$\mu_i = \alpha + r_i \beta_r + a_i \beta_a$$

$$\beta_r \sim N(0, 100)$$

$$\beta_a \sim N(0, 100)$$

$$\alpha \sim N(0, 100)$$

$$\sigma \sim U(0, 50)$$

```
library(rethinking)
model <- map(
  alist(
    Divorce ~ dnorm(mu, sigma),
    mu <- alpha + MedianAgeMarriage * beta_a
      + Marriage * beta_r,
    alpha ~ dnorm(0, 100),
    beta_a ~ dnorm(0, 100),
    beta_r ~ dnorm(0, 100),
    sigma ~ dunif(0, 50)),
  start = list(alpha = 0, beta_a = 0, beta_r = 0,
               sigma = 1),
  data = WaffleDivorce)
```

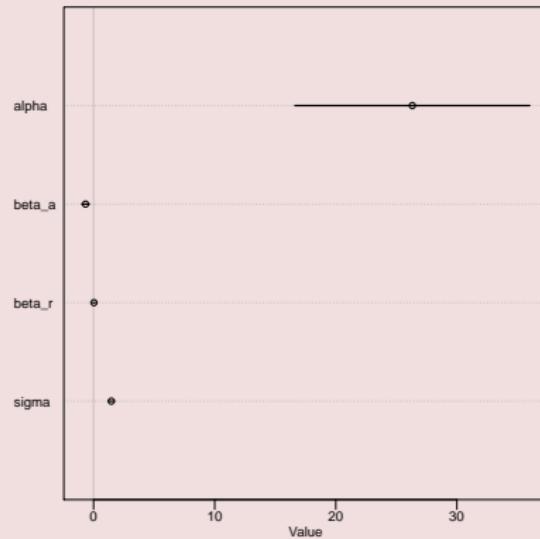
# Table output

```
precis(model)
```

	Mean	StdDev	5.5%	94.5%
alpha	36.69	7.41	24.85	48.53
beta_a	-0.99	0.24	-1.37	-0.61
beta_r	-0.06	0.08	-0.18	0.07
sigma	1.44	0.14	1.21	1.67

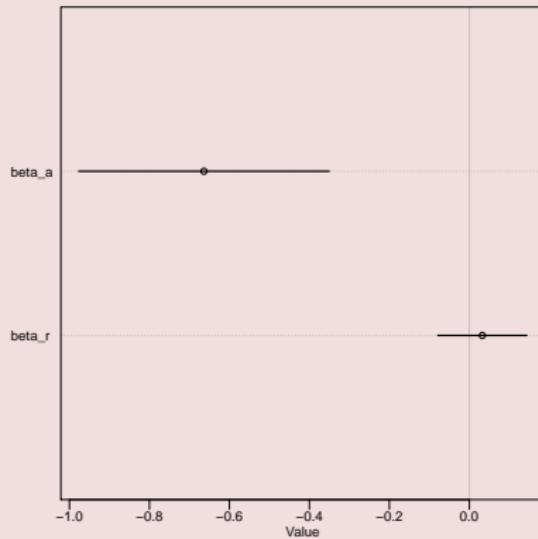
# Figure table output

```
plot(precis(model))
plot(precis(model), pars = c('beta_r', 'beta_a'))
```



# Figure table output

```
plot(precis(model))
plot(precis(model), pars = c('beta_r', 'beta_a'))
```



# Counterfactual plots

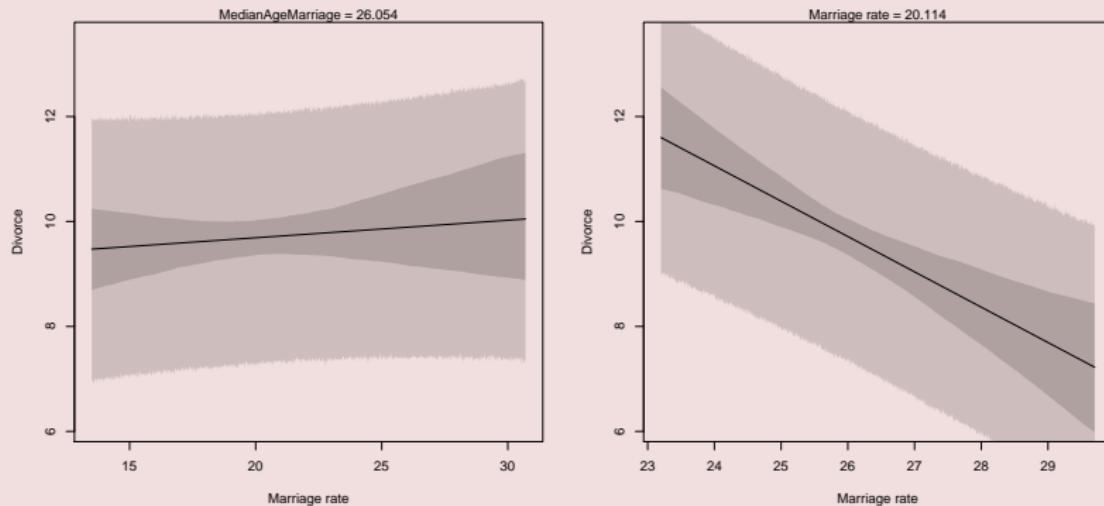
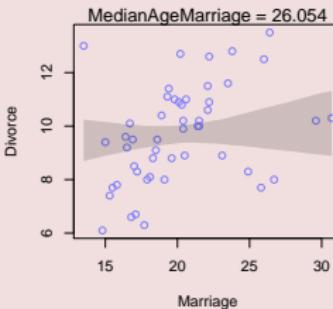
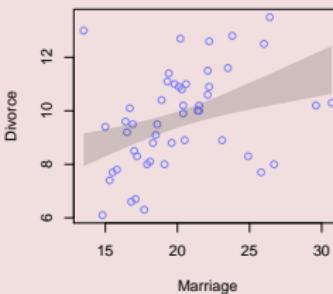
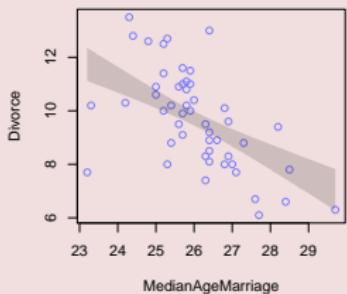


Figure: Keeping one variable fixed varying the other

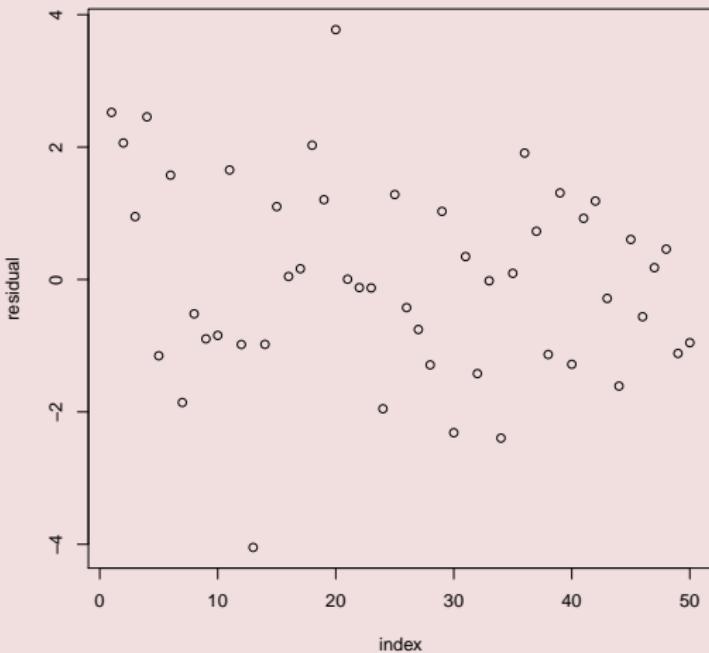
# Counterfactual plots vs one dim



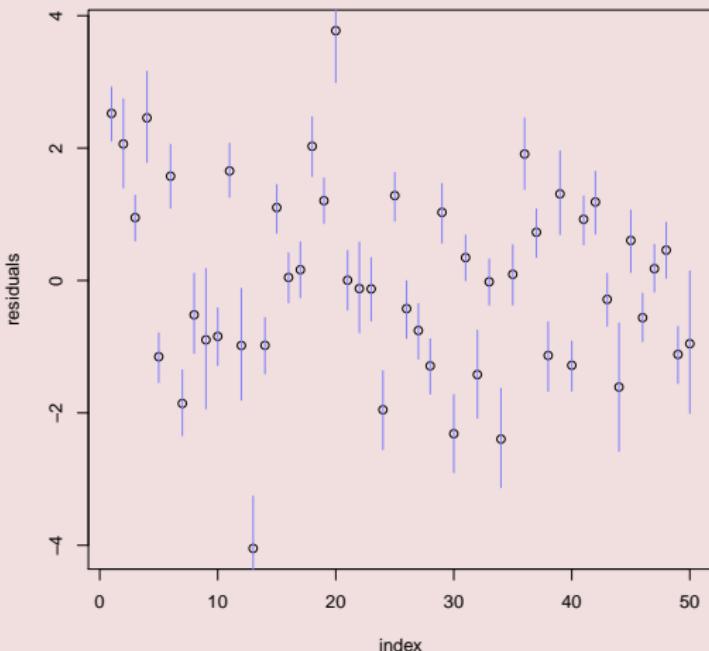
# Posterior prediction, residuals

- Check model fit.
- Find model failures, use to improve model.
- Non Bayesian  $y_i - \hat{\mu}_i$
- Bayesian  $pr(y_i - \mu_i | y_1, \dots, y_n)$  for  $i = 1, \dots, n$ .

# residual



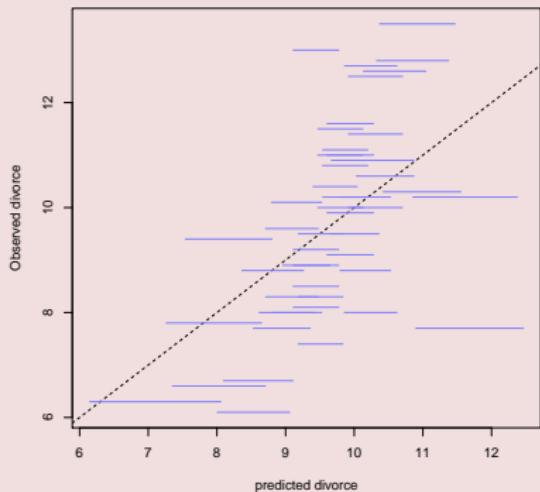
# residual



# residual

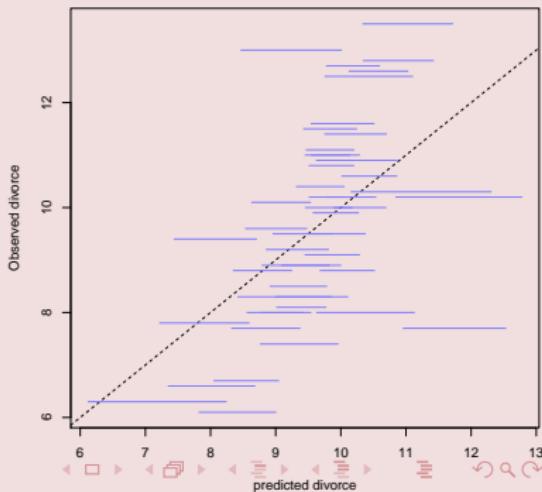
## simple model

$$d_i \sim N(\mu_i, \sigma), \\ \mu_i = \alpha + a_i \beta_a$$



## full model

$$d_i \sim N(\mu_i, \sigma), \\ \mu_i = \alpha + r_i \beta_r + a_i \beta_a$$



# milk and brains

Model:

$$k_i \sim N(\mu_i, \sigma),$$

$$\mu_i = \alpha + n_i \beta_n + \log(m_i) \beta_m$$

- $k_i$  - calories per gram milk.
- $m_i$  - weight of the ape.
- $n_i$  - neocortex percentage.

Priors see book.

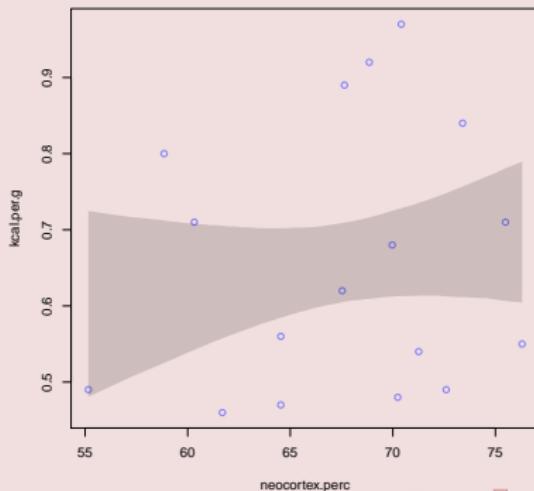


0.89 kcal/g  
68% neocortex

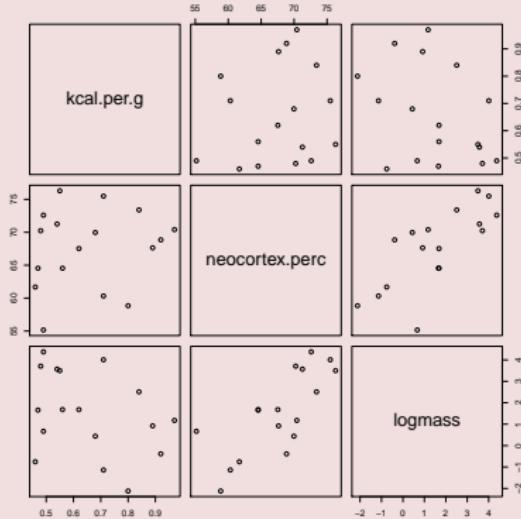
# simple model

simple model

$$k_i \sim N(\mu_i, \sigma), \\ \mu_i = \alpha + n_i \beta_n$$



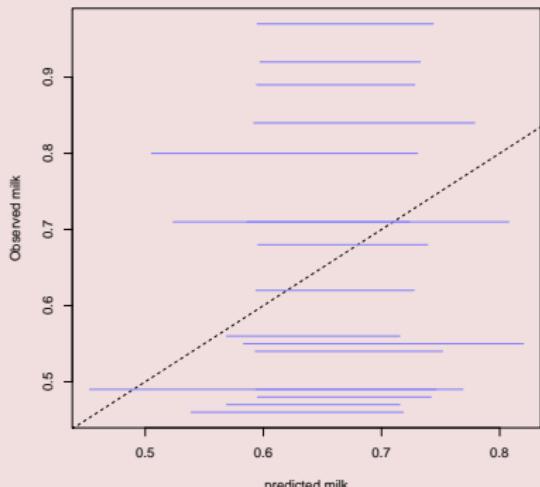
# milk data



# residual

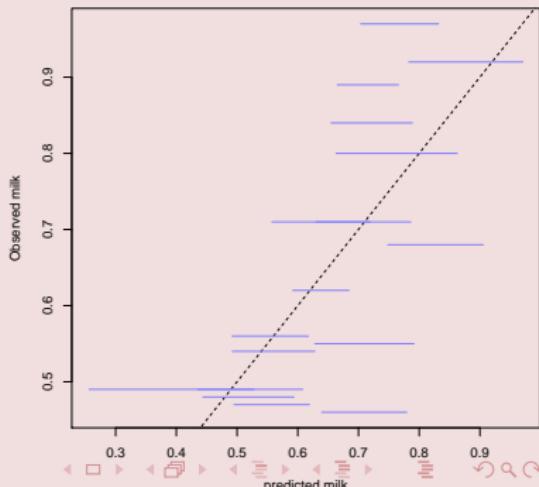
## simple model

$$k_i \sim N(\mu_i, \sigma), \\ \mu_i = \alpha + n_i \beta_n$$

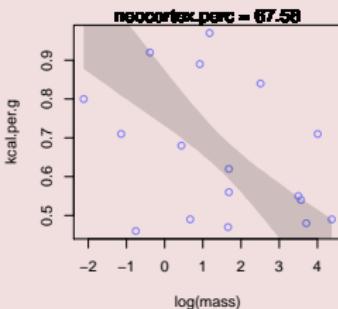
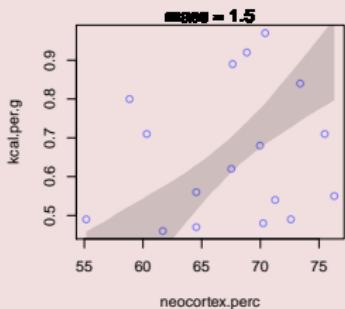
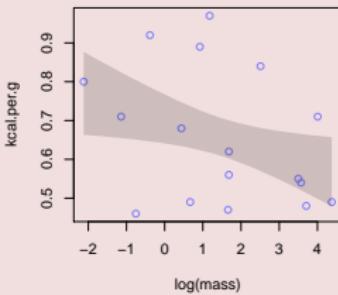
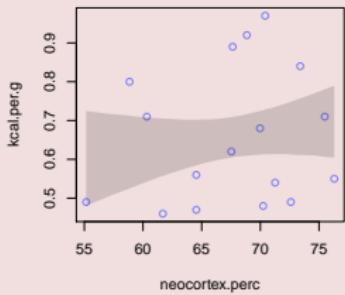


## full model

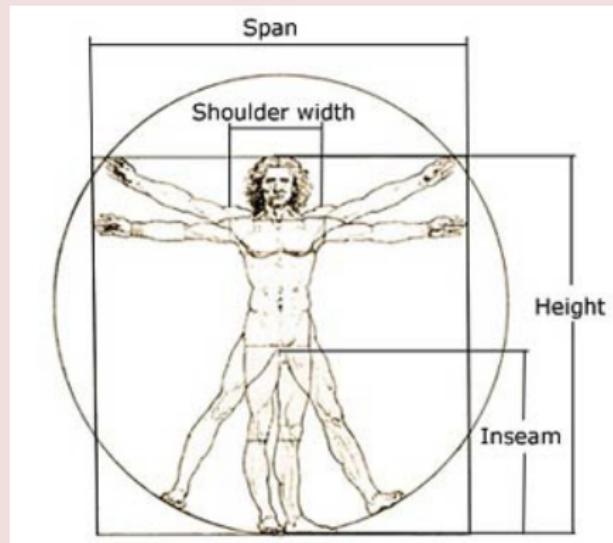
$$d_i \sim N(\mu_i, \sigma), \\ \mu_i = \alpha + n_i \beta_n + \log(m_i) \beta_m$$



# counter milk



# Body proportion



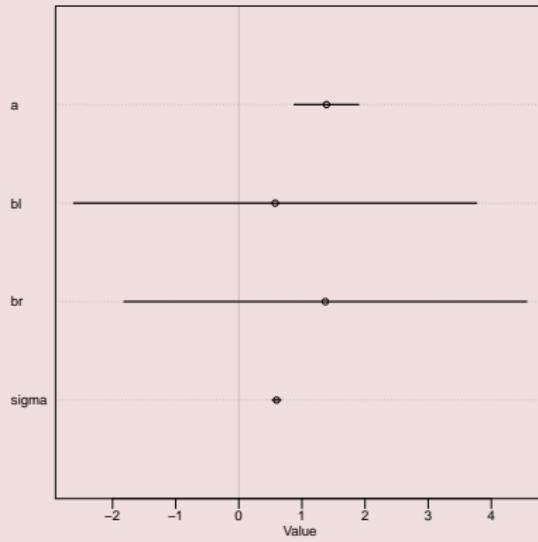
## the leg model

Building and estimating a Bayesian model:

```
model <- map(
  alist(
    height ~ dnorm(mu, sigma),
    mu <- alpha + leg_left * beta1 + leg_right * beta2,
    alpha ~ dnorm(10, 100),
    beta1 ~ dnorm(2, 10),
    beta2 ~ dnorm(2, 10),
    sigma ~ duniform(0, 10)
  ),
  data = d)
precis(model)
```

	Mean	StdDev	5.5%	94.5%
alpha	1.60	0.34	1.06	2.13
beta1	2.31	2.34	-1.43	6.05
beta2	-0.44	2.35	-4.19	3.32
sigma	0.65	0.05	0.58	0.73

# Fitting the distribution



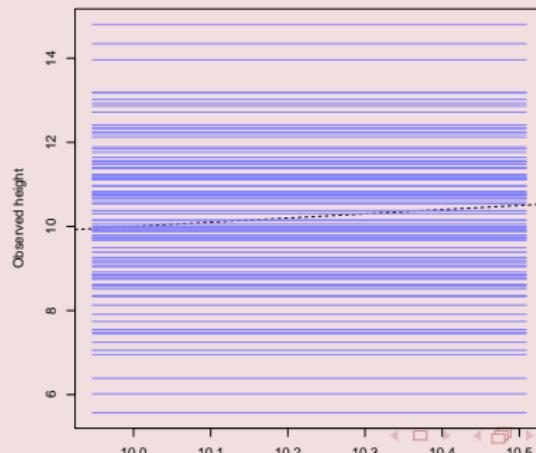
# Prediction

Using only simple model

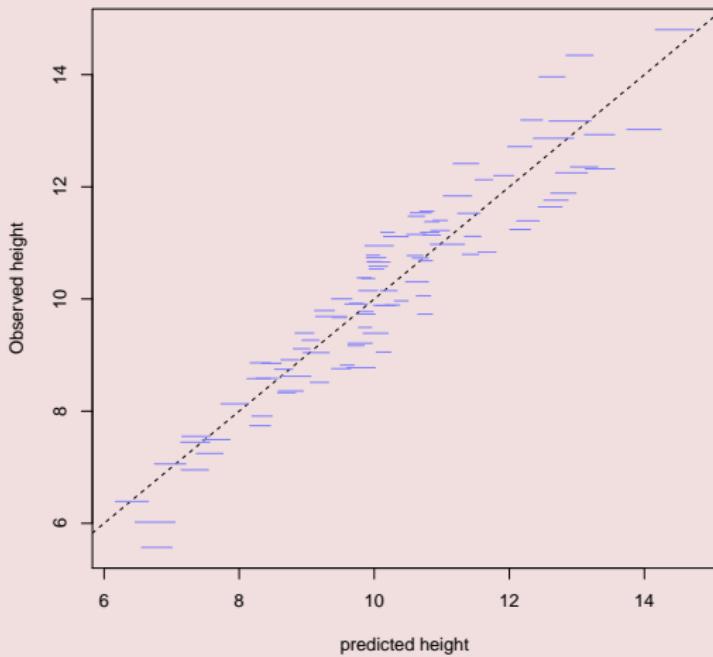
$$h_i \sim N(\alpha, \sigma),$$

$$\alpha \sim N(10, 100)$$

$$\sigma \sim U[0, 10]$$



# Prediction

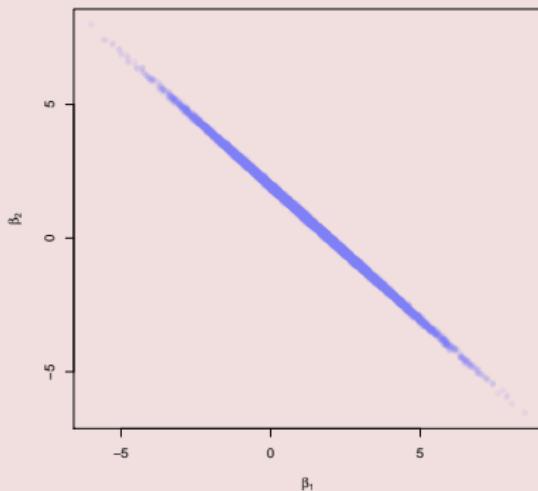


# Fitting the distribution

- Note that  
 $\text{left\_leg} \approx \text{right\_leg}$ .

# Fitting the distribution

- Note that  $\text{left\_leg} \approx \text{right\_leg}$ .
- Thus the  $\beta_1$  and  $\beta_2$  are very dependent.

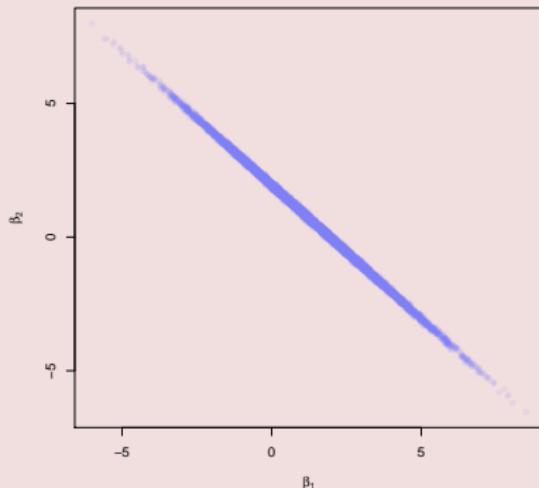


# Fitting the distribution

- Note that  $\text{left\_leg} \approx \text{right\_leg}$ .
- Thus the  $\beta_1$  and  $\beta_2$  are very dependent.
- The model is almost:

$$\text{height}_i \sim N(\mu_i, \sigma),$$

$$\mu_i = \alpha + \text{leg}_i(\beta_1 + \beta_2)$$



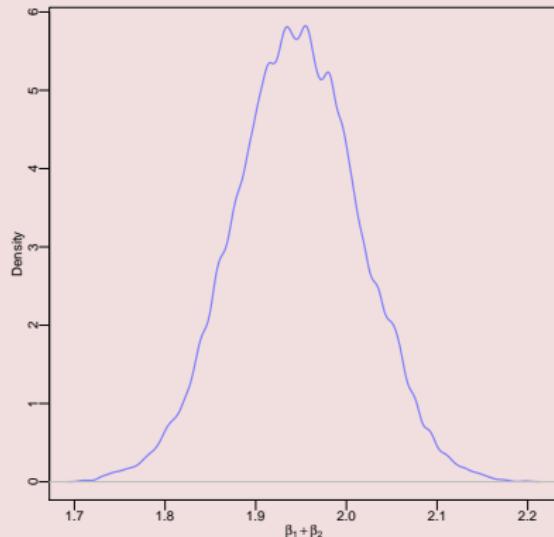
# Fitting the distribution

- Note that  $\text{left\_leg} \approx \text{right\_leg}$ .
- Thus the  $\beta_1$  and  $\beta_2$  are very dependent.
- The model is almost:

$$\text{height}_i \sim N(\mu_i, \sigma),$$

$$\mu_i = \alpha + \text{leg}_i(\beta_1 + \beta_2)$$

- Examining posterior distribution of  $\beta_1 + \beta_2$



# Category model

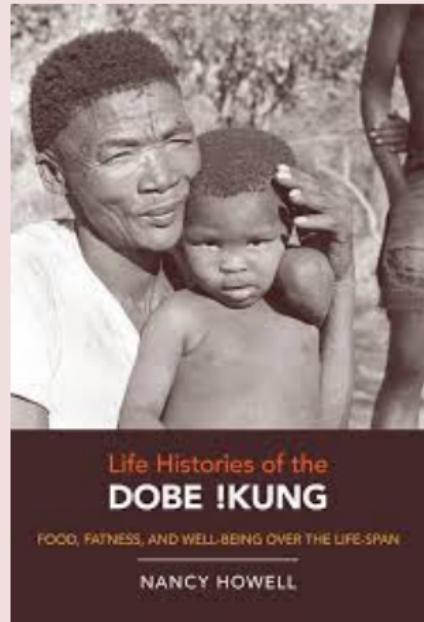
- Binary variables (or dummy variables) allow each category to have its own unique intercept.

$$h_i \sim N(\mu_i, \sigma)$$

$$\mu_i = \alpha + m\beta_m$$

- For simple model, better interpretation we think the categories have the same standard deviation.

# height data again



```
library(rethinking)  
data(Howell1)  
head(Howell1)
```

	height	male
1	151.765	1
2	139.700	0
3	136.525	0
4	156.845	1
5	145.415	0
6	163.830	1

# difference in gender

```
precis(model)
```

	Mean	StdDev	5.5%	94.5%
alpha	149.52	0.40	148.87	150.16
beta	10.84	0.59	9.90	11.79
sigma	5.52	0.21	5.19	5.85

# Table output

```
post <- extract.samples(model)
alpha.male <- post$alpha + post$beta
PI(alpha.male)
```

5%	94%
159.6768	161.0520

# The two models

difference prior

$$h_i \sim N(\mu_i, \sigma)$$

$$\mu_i = \alpha + \beta m_i$$

$$\alpha \sim N(178, 100)$$

$$\beta \sim N(0, 10)$$

$$\sigma \sim U(0, 50)$$

independent prior

$$h_i \sim N(\mu_i, \sigma)$$

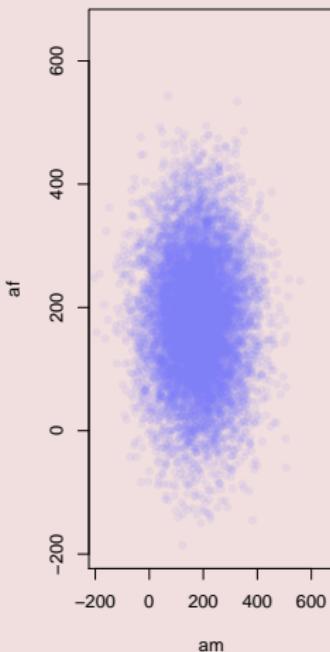
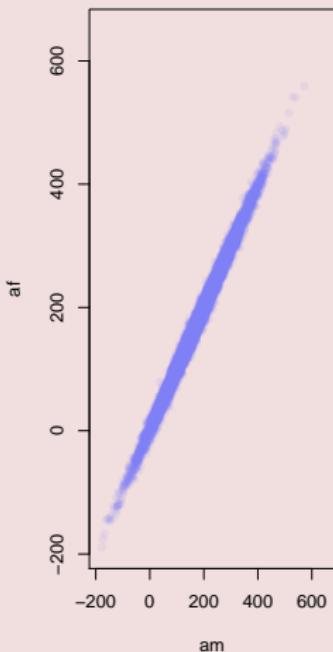
$$\mu_i = \alpha_f(1 - m_i) + \alpha_m m_i$$

$$\alpha_m \sim N(178, 100)$$

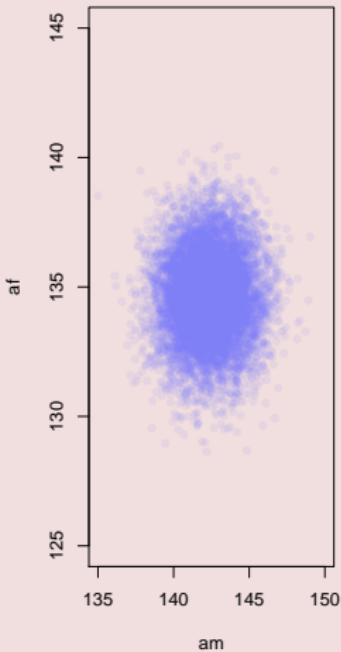
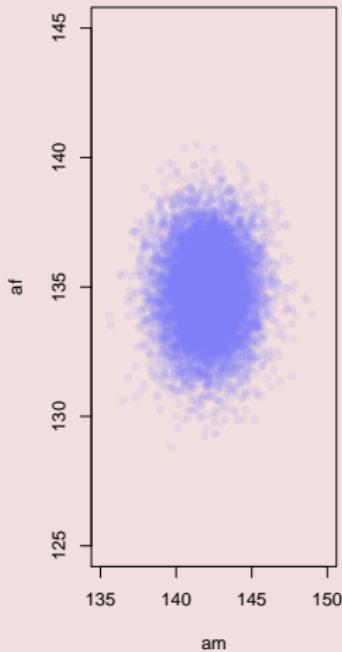
$$\alpha_f \sim N(178, 100)$$

$$\sigma \sim U(0, 50)$$

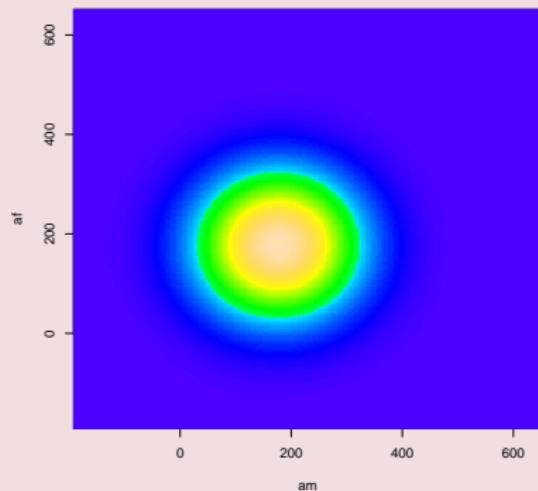
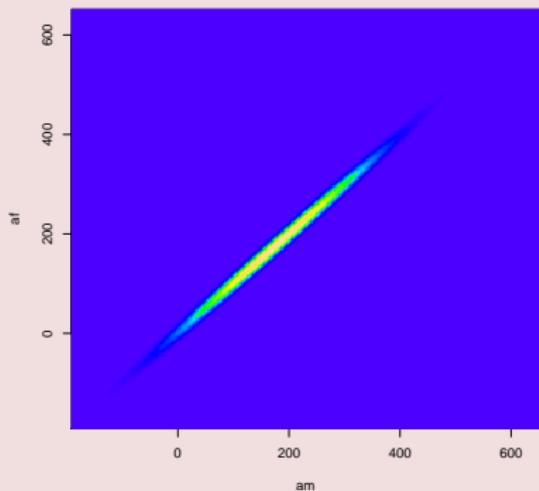
# Sampling from prior



# Sampling from posterior



# The two priors



**Figure:** The two priors on  $\alpha_f$  and  $\alpha_m$

# The likelihood

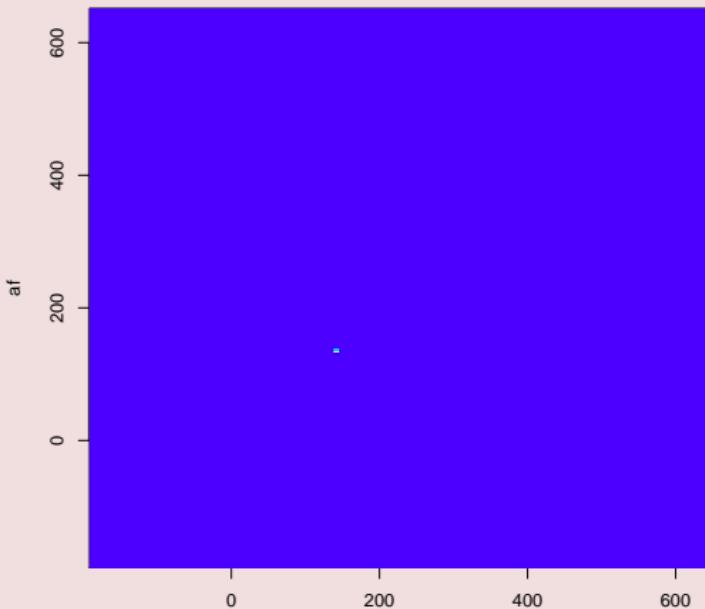


Figure:  $p(h_1, \dots, h_m | \alpha_m, \alpha_f, \sigma = \hat{\sigma})$

# Two posteriors

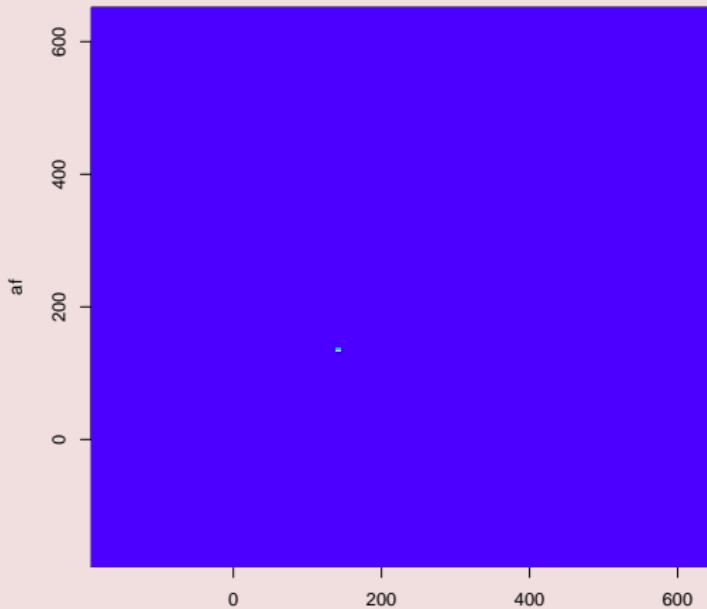
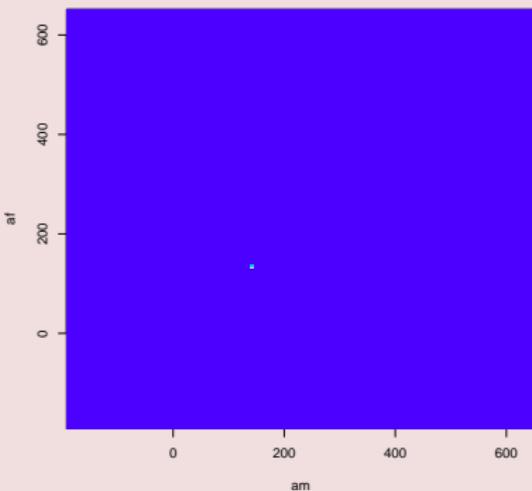
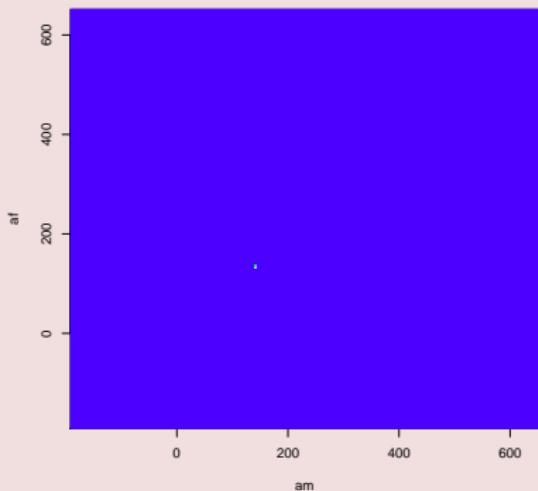


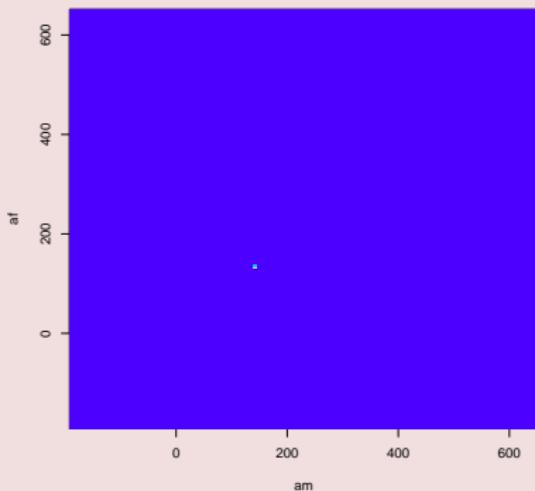
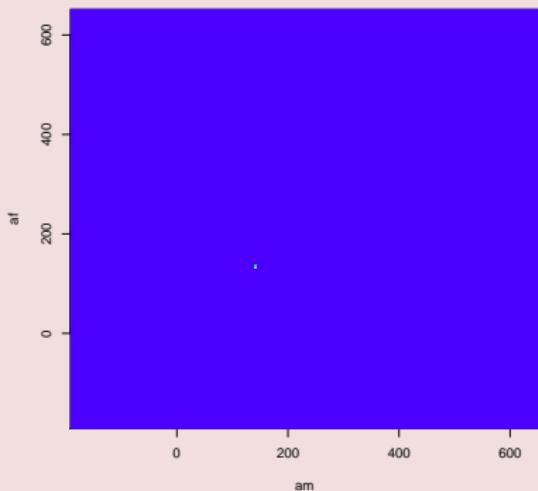
Figure:  $p(h_1, \dots, h_m | \alpha_m, \alpha_f, \sigma = \hat{\sigma})$

# The two posteriors



**Figure:** The two posteriors

# The two posteriors



**Figure:** The two posteriors

# Several categories

- For  $k$  categories, we need  $k - 1$  dummy variables.

# Building category variables

```
data(milk)
unique(milk$clade)
```

```
[1] Strepsirrhine      New World Monkey Old World Monkey Ape
Levels: Ape New World Monkey Old World Monkey Strepsirrhine
```

# Building category variables

```
milk$d.NMM <- ifelse(milk$clade == "NewWorldMonkey", 1, 0)
milk$d.OMM <- ifelse(milk$clade == "OldWorldMonkey", 1, 0)
milk$d.S   <- ifelse(milk$clade == "Strepsirrhine", 1, 0)
head(milk[,c("clade", "d.NMM", "d.OMM", "d.S")])
```

	clade	d.NMM	d.OMM	d.S
1	Strepsirrhine	0	0	1
6	New World Monkey	1	0	0
7	New World Monkey	1	0	0
8	New World Monkey	1	0	0
10	New World Monkey	1	0	0
11	New World Monkey	1	0	0
12	New World Monkey	1	0	0
13	New World Monkey	1	0	0
16	Old World Monkey	0	1	0
18	Old World Monkey	0	1	0

# Building category variables by intercept

$$kcal_i \sim N(\mu_i, \sigma)$$

$$\mu_i = \alpha_{j[i]}$$

$$\alpha_j \sim N(0.6, 10), j = 1, 2, 3, \dots, J$$

$$\sigma \sim U(0, 10)$$

- $j[i]$  means persons  $i$  category.
- Example if person  $i = 1$  belongs to category 2, then  $\mu_1 = \alpha_2$ .

# Building category variables

```
milk$clade_id <- coerce_index(milk$clade)
head(milk[,c("clade", "clade_id")], 5)
```

	clade	clade_id
1	Strepsirrhine	4
6	New World Monkey	2
7	New World Monkey	2
8	New World Monkey	2
10	New World Monkey	2

# Building category variables

```
data(milk)
milk <- milk[complete.cases(milk),]
milk$clade_id <- coerce_index(milk$clade)
model <- map(
  alist(
    kcal.per.g ~ dnorm(mu, sigma),
    mu <- alpha[clade_id],
    alpha[clade_id] ~ dnorm(0.6, 10),
    sigma ~ dunif(0, 10)
  ),
  data = milk)
precis(model, depth = 2)
```

	Mean	StdDev	5.5%	94.5%
alpha[1]	0.57	0.05	0.48	0.65
alpha[2]	0.69	0.05	0.61	0.77
alpha[3]	0.83	0.08	0.71	0.95
alpha[4]	0.49	0.13	0.28	0.70
sigma	0.13	0.02	0.10	0.17