Statistical Rethinking

Week 3: Multivariate Models & The Causal Terror

Richard McElreath

Posterior predictions

- Goal: Compute implied predictions for *observed* cases
  - Check model fit — golems do make mistakes
  - Find model failures, stimulate new ideas
- Always average over the posterior distribution
  - Using only MAP leads to overconfidence
  - Embrace the uncertainty

Distribution of residuals for each State

- negative residual: less divorce than expected
- positive residual: more divorce than expected

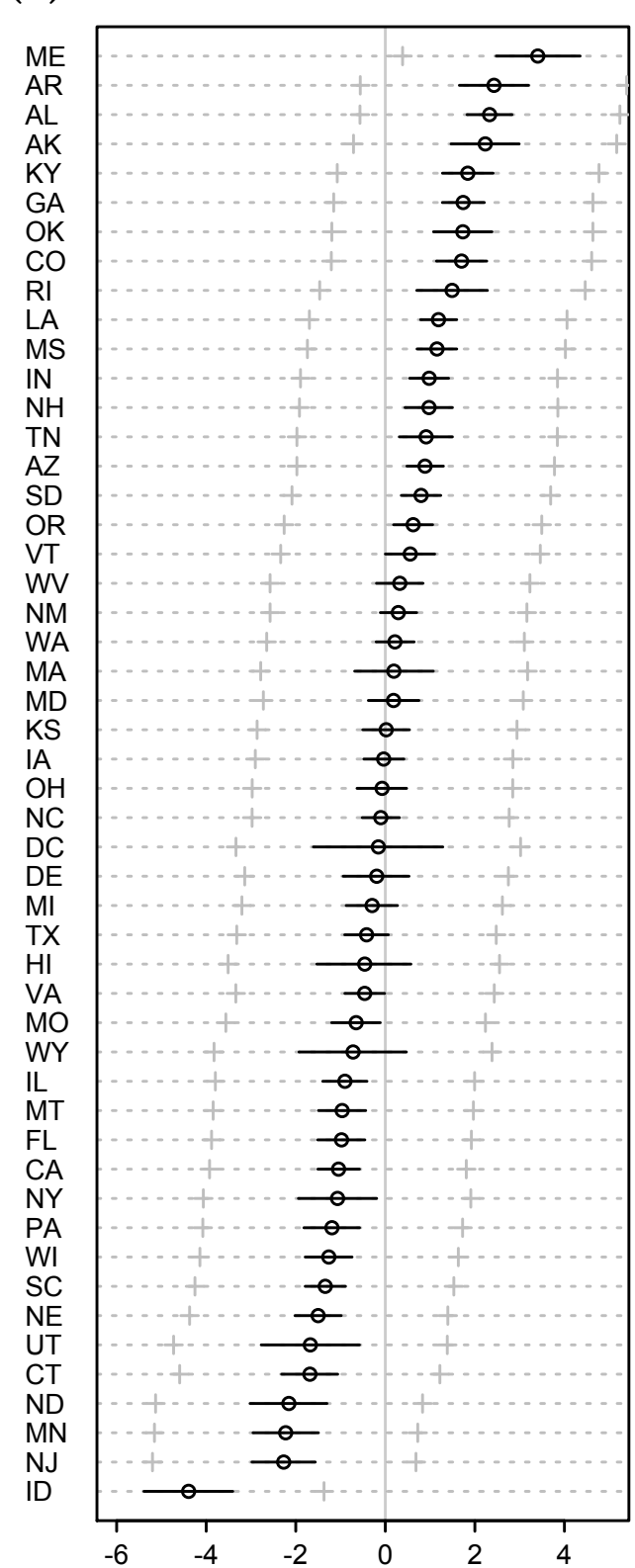
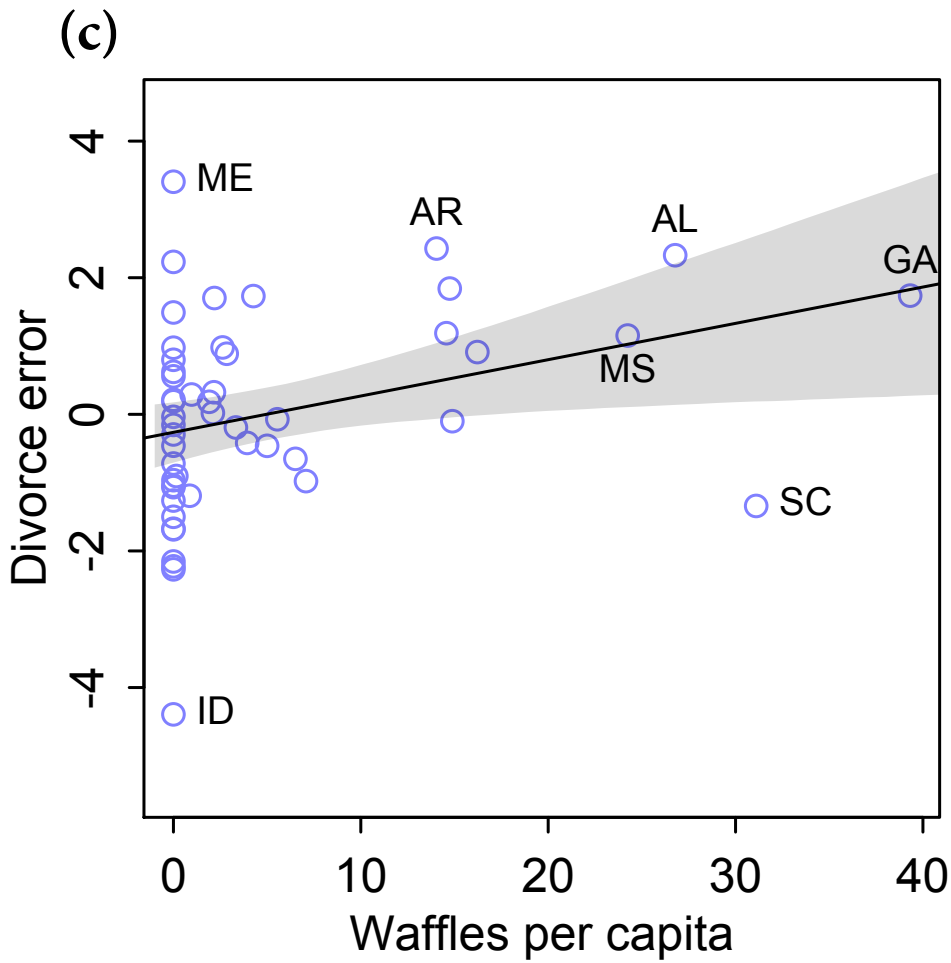


Figure 5.6


(c)
Divorce error
Waffles per capita
ME
AR
AL
GA
MS
SC
ID
4
2
0
-2
-4
0
10
20
30
40

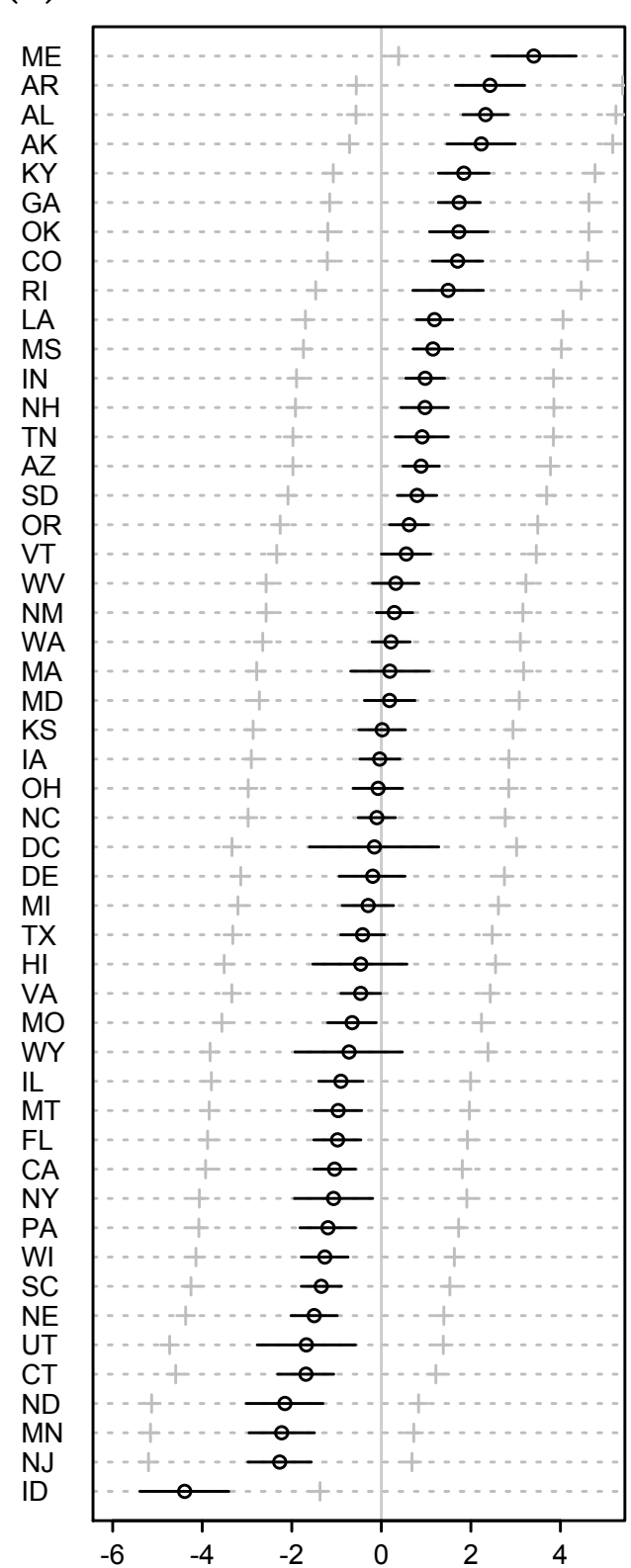

ME
AR
AL
AK
KY
GA
OK
CO
RI
LA
MS
IN
NH
TN
AZ
SD
OR
VT
WV
NM
WA
MA
MD
KS
IA
OH
NC
DC
DE
MI
TX
HI
VA
MO
WY
IL
MT
FL
CA
NY
PA
WI
SC
NE
UT
CT
ND
MN
NJ
ID
-6
-4
-2
0
2
4

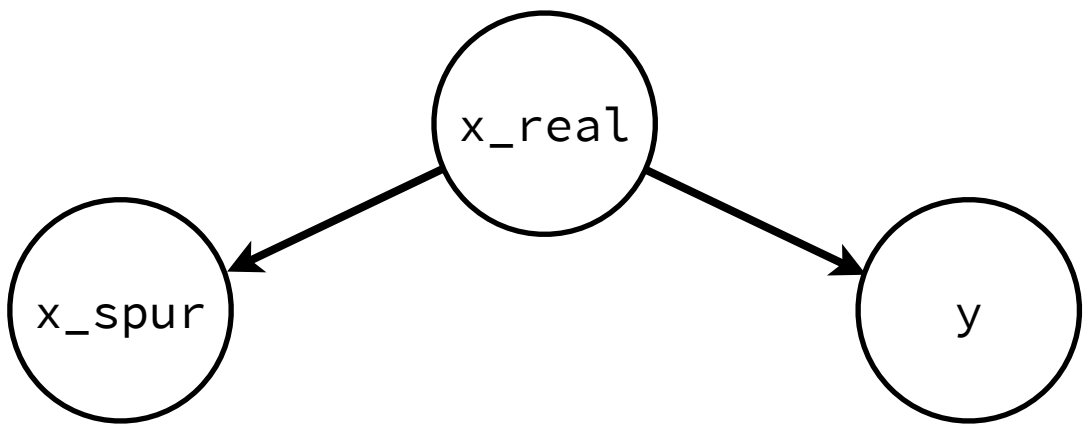
Figure 5.6

Synthetic spurious association

- Thinking generatively very useful
- Simulate spurious association to better understand how simple it is

R code 5.15

```
N <- 100                          # number of cases
x_real <- rnorm( N )              # x_real as Gaussian with mean 0 and stddev 1
x_spur <- rnorm( N , x_real )     # x_spur as Gaussian with mean=x_real
y <- rnorm( N , x_real )          # y as Gaussian with mean=x_real
d <- data.frame(y,x_real,x_spur) # bind all together in data frame
```



Masked association

- Sometimes association between outcome and predictor masked by another variable
- Need both variables to see influence of either
- Tends to arise when
  - Another predictor associated with outcome *in opposite direction*
  - Both predictors associated with one another
- Noise in predictors can also mask association

Milk and Brain

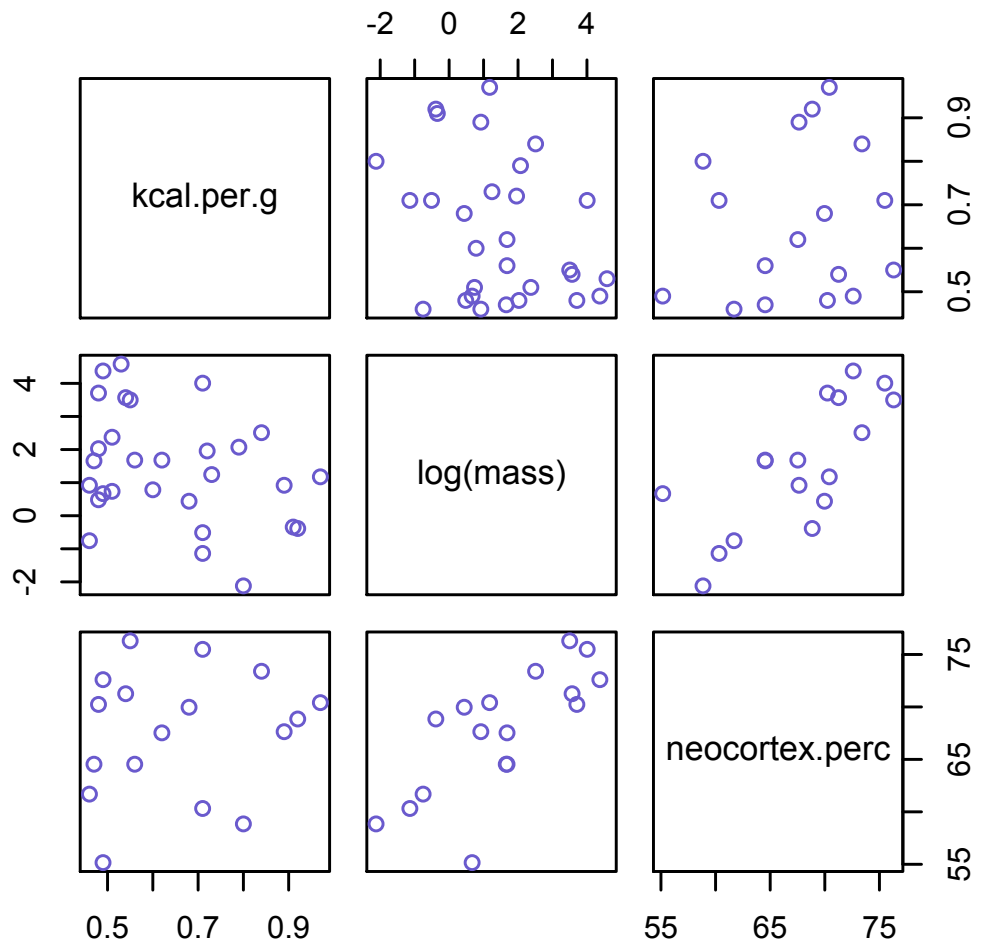
Eulemur fulvus
0.49 *kcal/g*
55% *neocortex*

Homo sapiens
0.71 *kcal/g*
75% *neocortex*

Cebus apella
0.89 *kcal/g*
68% *neocortex*

Masked influence

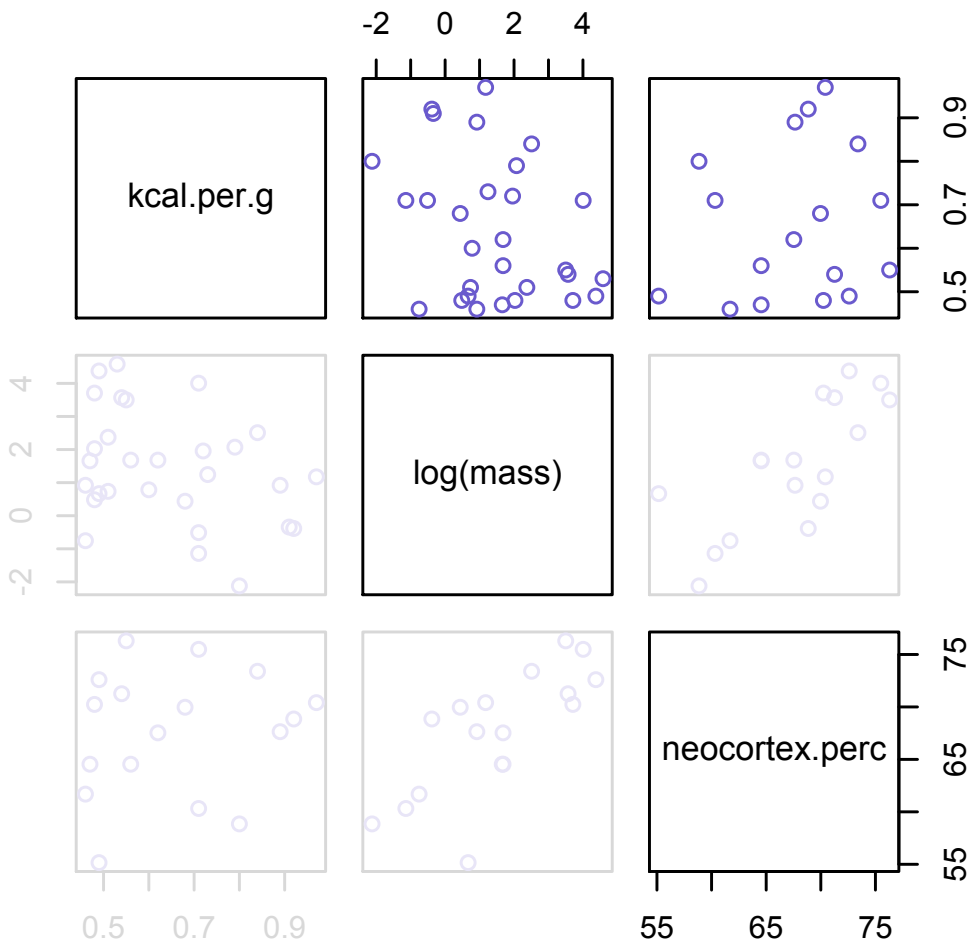
- Primate milk data




```
library(rethinking)
data(milk)
d <- milk
pairs(~kcal.per.g+log(mass)
    +neocortex.perc , data=d)
```

Masked influence

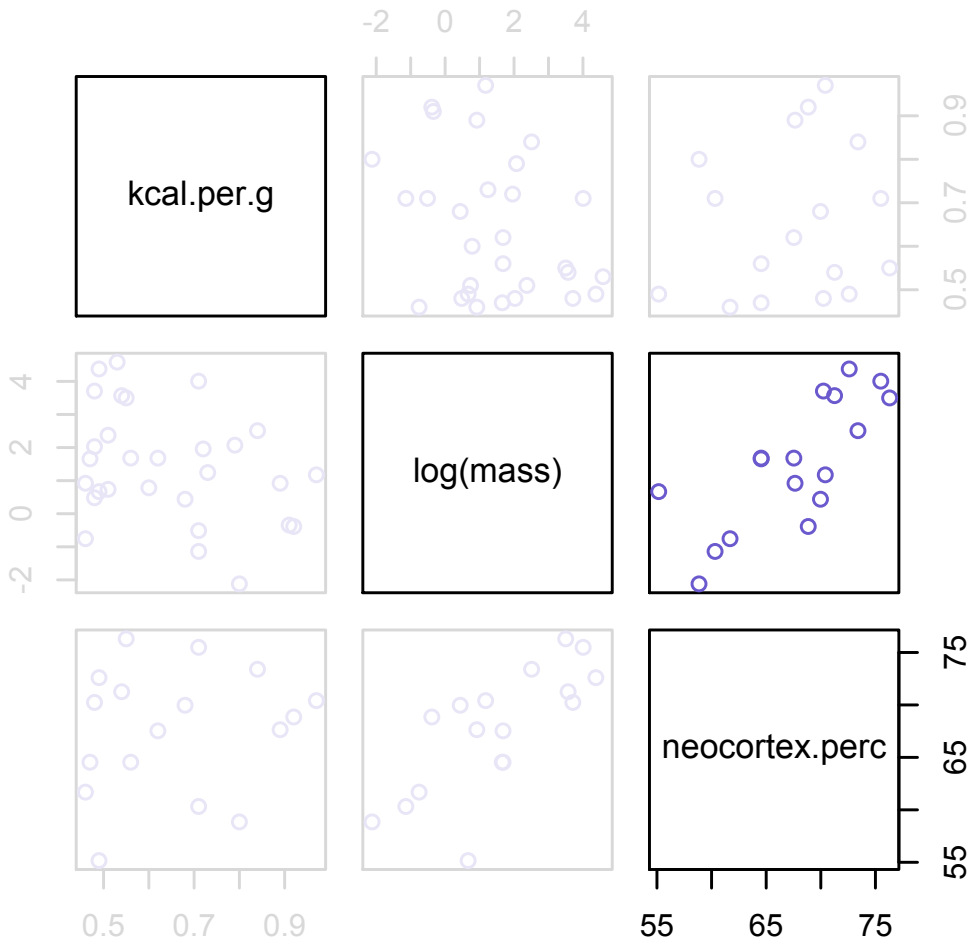
- Primate milk data




```
library(rethinking)
data(milk)
d <- milk
pairs(~kcal.per.g+log(mass)
    +neocortex.perc , data=d)
```

Masked influence

- Primate milk data




```
library(rethinking)
data(milk)
d <- milk
pairs(~kcal.per.g+log(mass)
    +neocortex.perc , data=d)
```

Complete cases

- Missing values in primate milk data

R code 5.18

```
d$neocortex.perc
```

```
 [1] 55.16    NA    NA    NA    NA 64.54 64.54 67.64    NA 68.85 58.85 61.69
[13] 60.32    NA    NA 69.97    NA 70.41    NA 73.40    NA 67.53    NA 71.26
[25] 72.60    NA 70.24 76.30 75.49
```

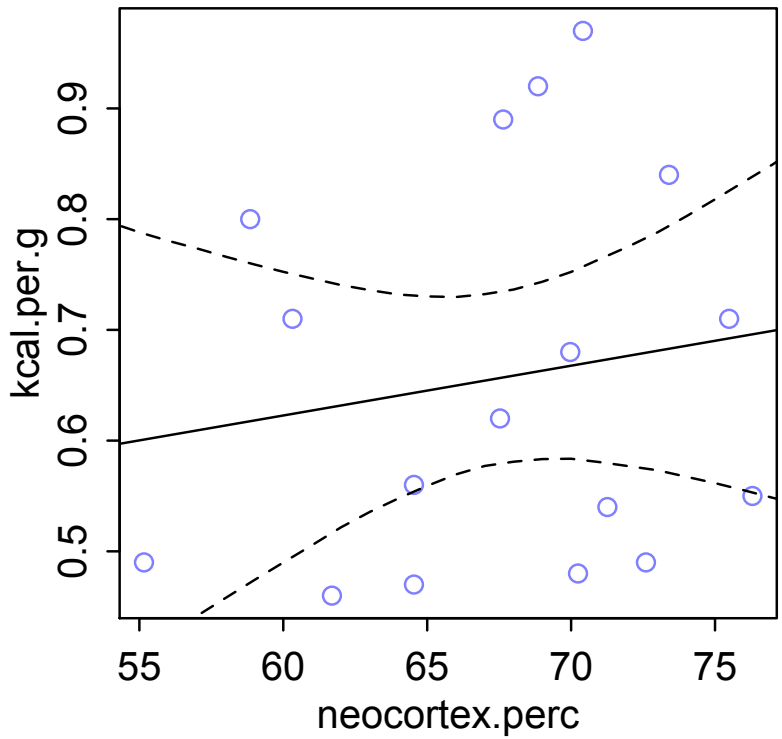
- Drop cases (species) with missing values

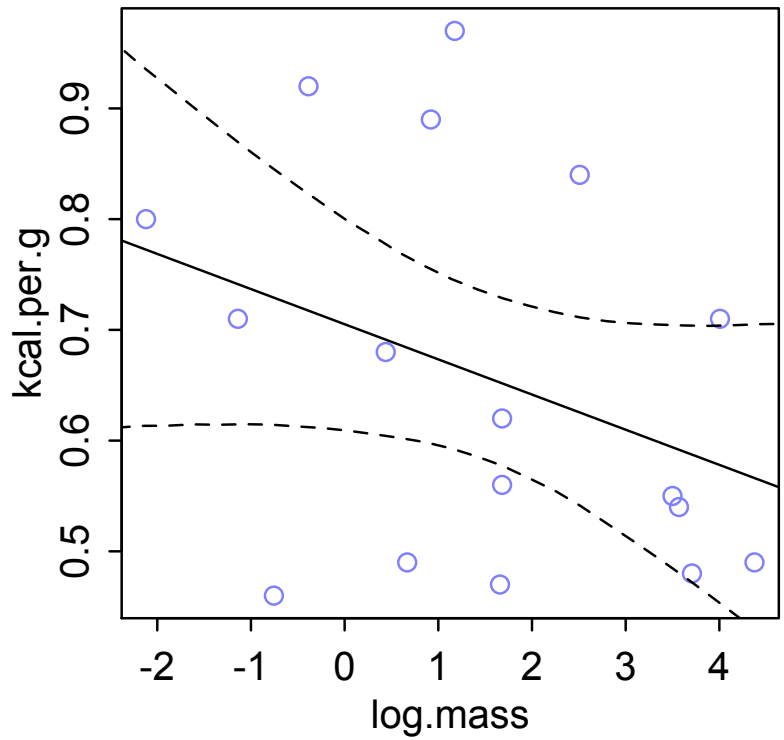
R code 5.19

```
dcc <- d[ complete.cases(d) , ]
```

- Much later, see how to "impute" missing values, so can use all the data

Bivariate models






```
kcal.per.g ~ dnorm(mu,sigma),
mu <- a + bp*neocortex.perc
```

```
kcal.per.g ~ dnorm(mu,sigma),
mu <- a + bm*log(mass)
```

Figure 5.7

Multivariate model

$$
\begin{aligned}
k_i &\sim \text{Normal}(\mu_i, \sigma) \\
\mu_i &= \alpha + \beta_n n_i + \beta_m \log(m_i) \\
\alpha &\sim \text{Normal}(0, 100) \\
\beta_n &\sim \text{Normal}(0, 1) \\
\beta_m &\sim \text{Normal}(0, 1) \\
\sigma &\sim \text{Uniform}(0, 10)
\end{aligned}
$$

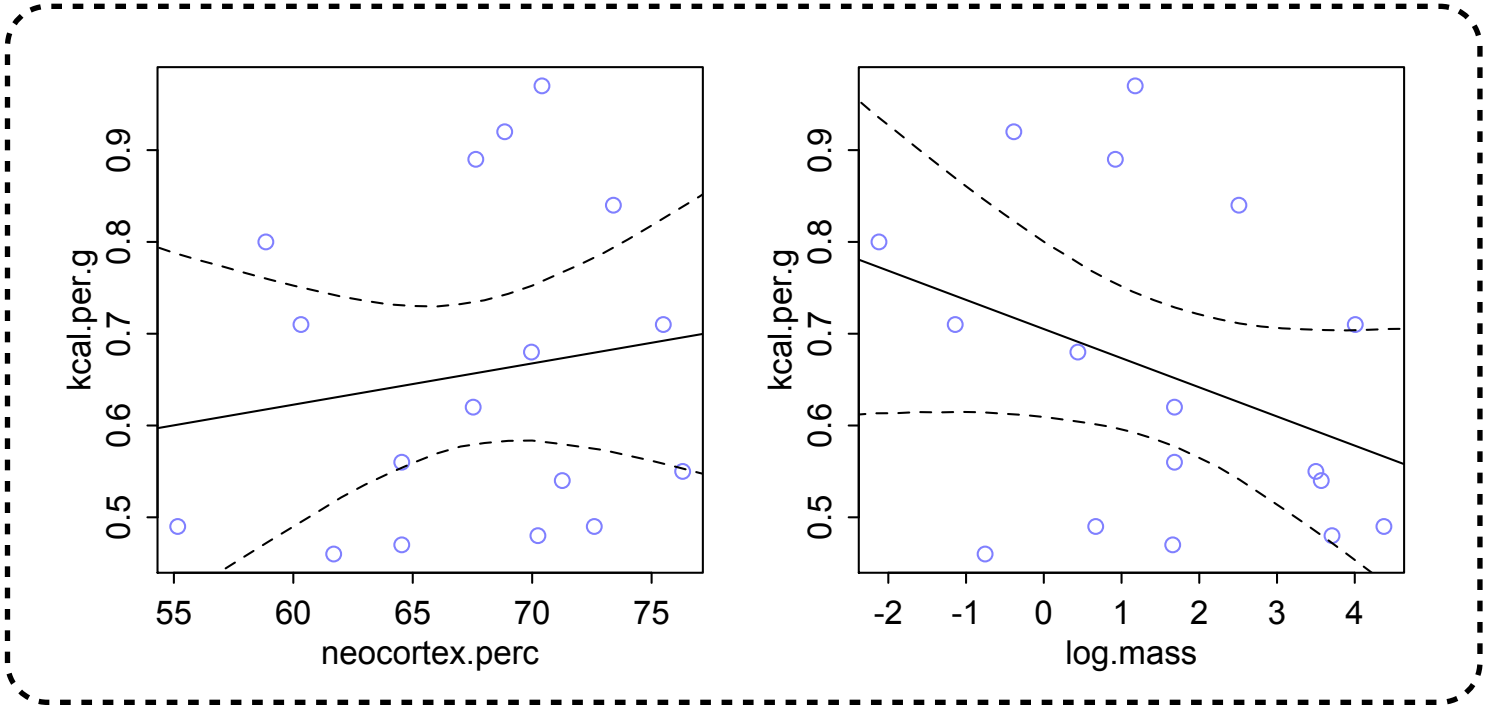
Multivariate model

R code 5.26

```
m5.7 <- map(
    alist(
        kcal.per.g ~ dnorm( mu , sigma ) ,
        mu <- a + bn*neocortex.perc + bm*log.mass ,
        a ~ dnorm( 0 , 100 ) ,
        bn ~ dnorm( 0 , 1 ) ,
        bm ~ dnorm( 0 , 1 ) ,
        sigma ~ dunif( 0 , 1 )
    ) ,
    data=dcc )
precis(m5.7)
```

```
       Mean StdDev  5.5% 94.5%
a     -1.09   0.47 -1.83 -0.34
bn     0.03   0.01  0.02  0.04
bm    -0.10   0.02 -0.13 -0.06
sigma  0.11   0.02  0.08  0.15
```

Bivariate



Multivariate

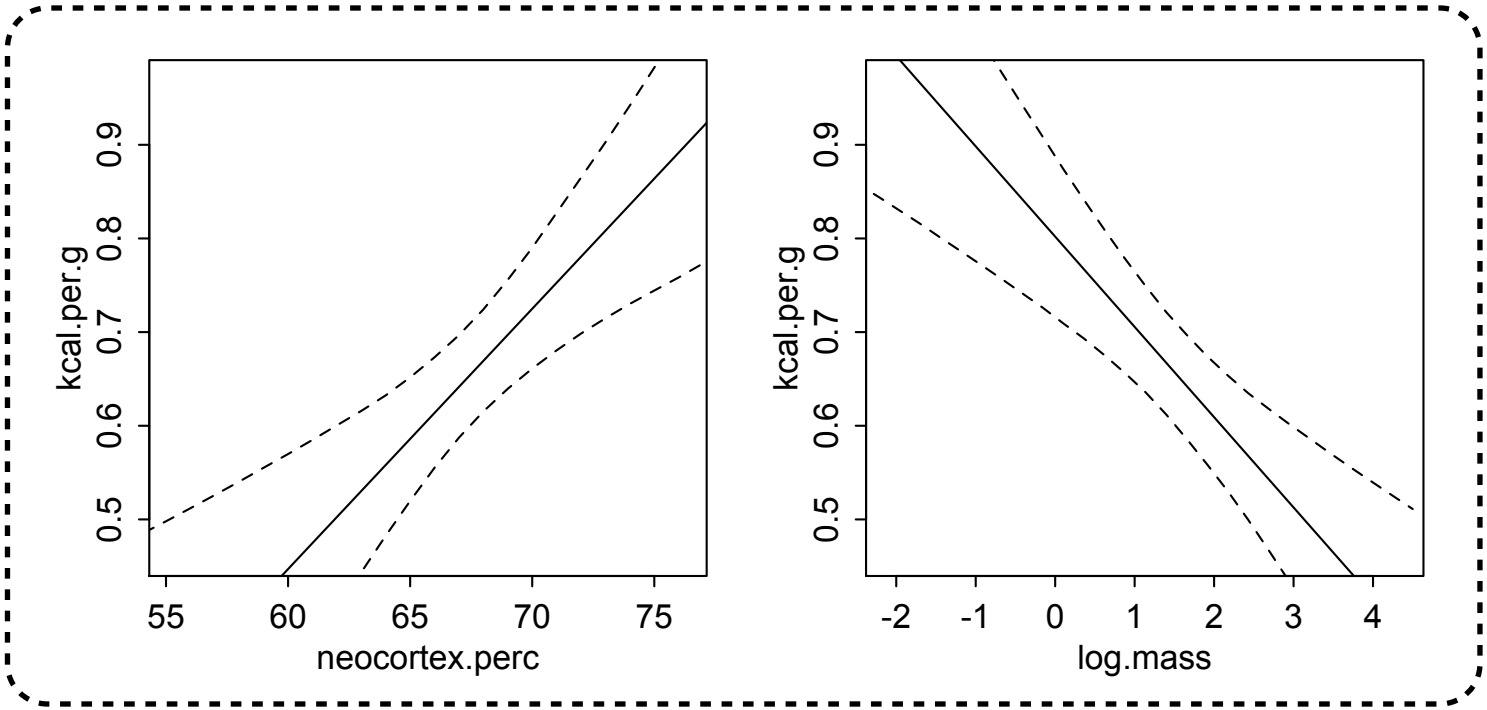


Figure 5.7

```
mean.log.mass <- mean( log(dcc$mass) )
np.seq <- 0:100
pred.data <- list(
    neocortex.perc=np.seq,
    log.mass=mean.log.mass
)

mu <- link( m5.7 , data=pred.data )
mu.mean <- apply( mu , 2 , mean )
mu.PI <- apply( mu , 2 , PI )

plot( kcal.per.g ~ neocortex.perc , data=dcc , type="n" )
lines( np.seq , mu.mean )
lines( np.seq , mu.PI[1,] , lty=2 )
lines( np.seq , mu.PI[2,] , lty=2 )
```

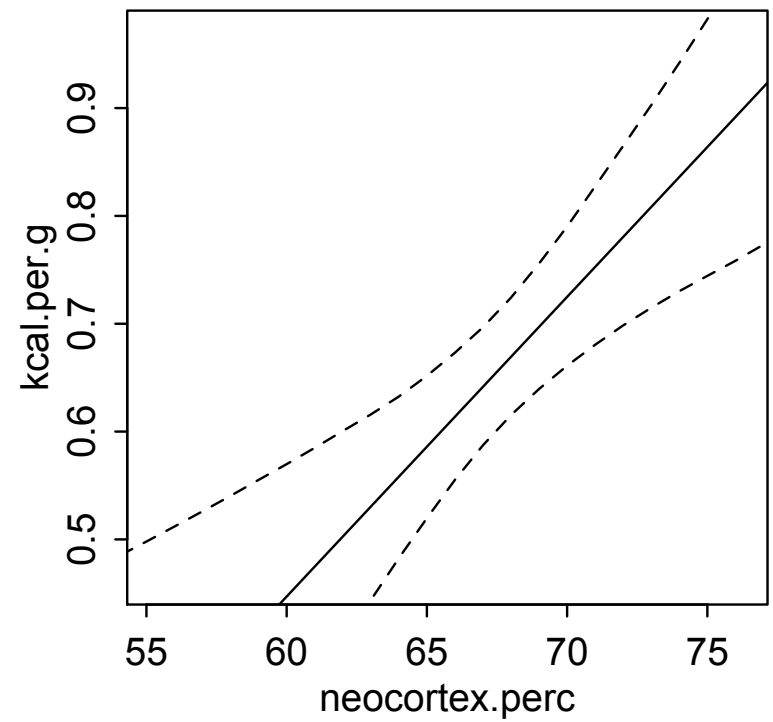
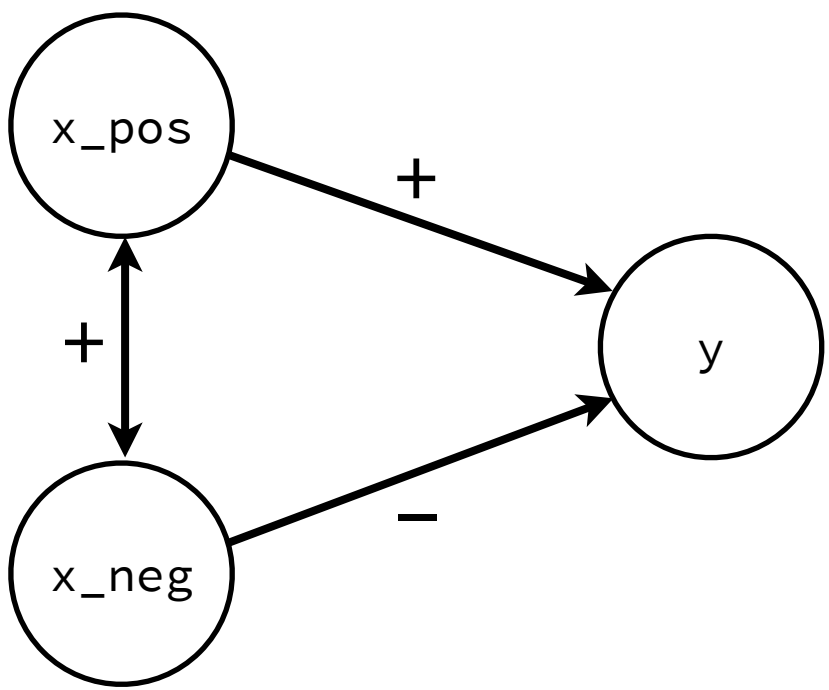


Figure 5.7

Synthetic masked association

R code 5.28

```
N <- 100                          # number of cases
rho <- 0.7                        # correlation btw x_pos and x_neg
x_pos <- rnorm( N )               # x_pos as Gaussian
x_neg <- rnorm( N , rho*x_pos ,   # x_neg correlated with x_pos
    sqrt(1-rho^2) )
y <- rnorm( N , x_pos - x_neg )   # y equally associated with x_pos, x_neg
d <- data.frame(y,x_pos,x_neg)    # bind all together in data frame
```



Regression as a wicked oracle

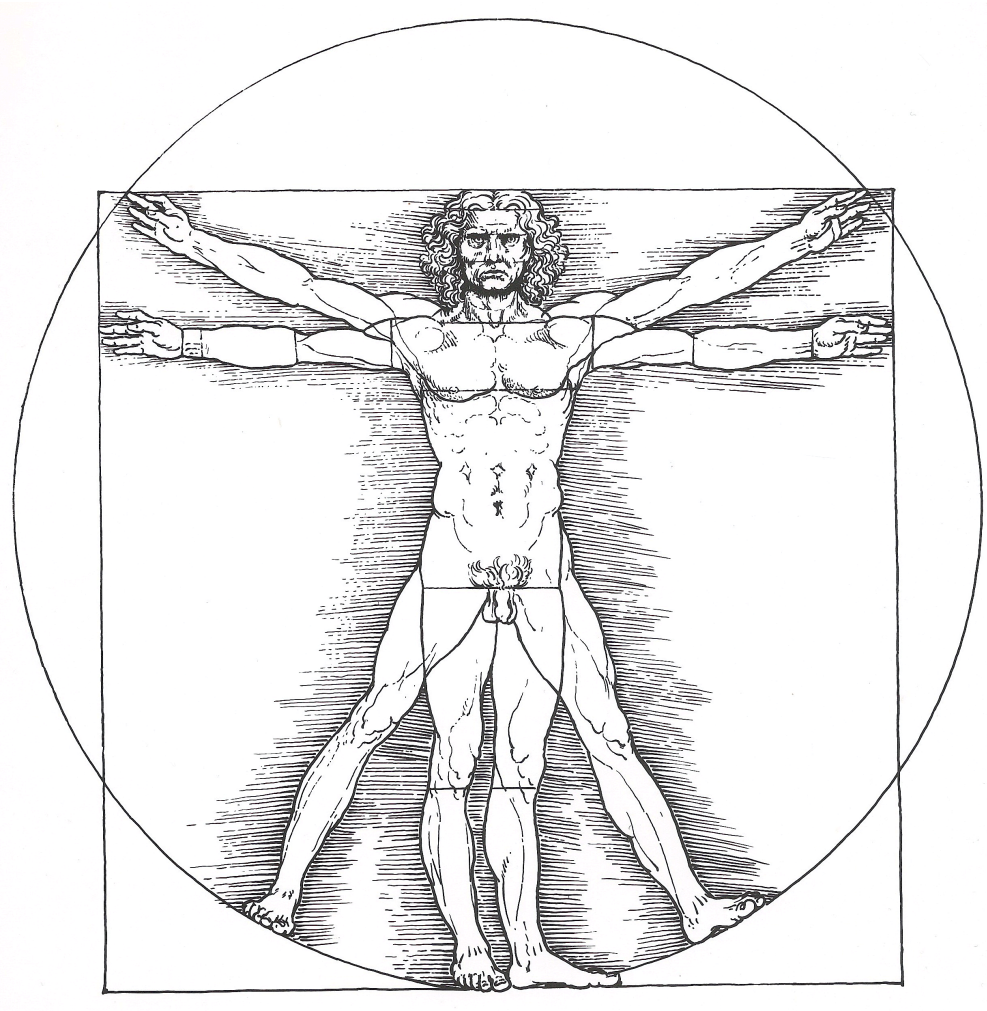
- Regression automatically focuses on the most informative cases
- Cases that don't help are automatically ignored
- But not kind — ask carefully

Why not just add everything?

- Could just add all available predictors to model
- Almost always a bad idea
  - **Multicollinearity**
  - Confounding **colliders**
  - Loss of interpretability
  - Loss of precision
  - Overfitting

Multicollinear legs

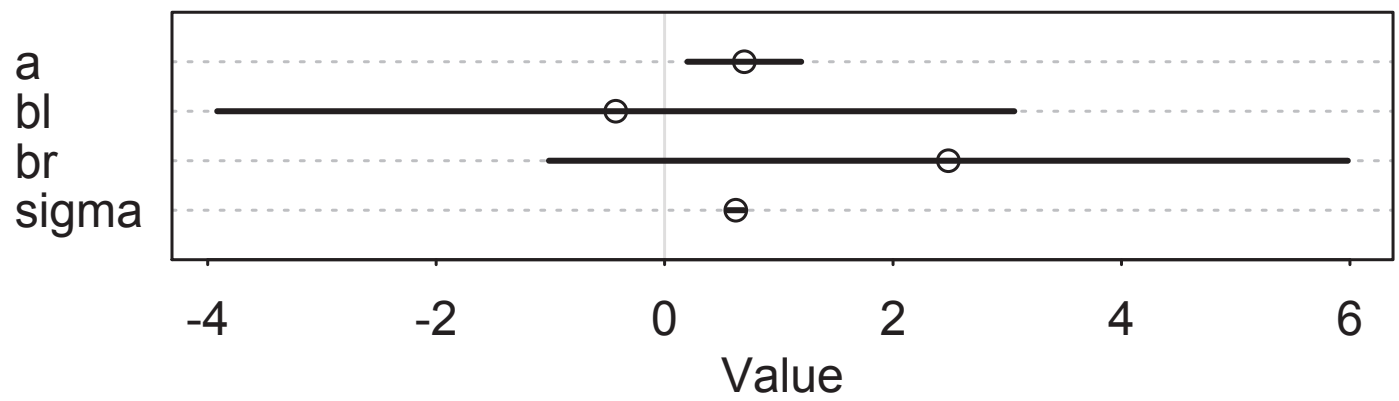
	height	leg_left	leg_right
1	15.384202	7.115039	7.139183
2	12.176479	5.718942	5.729024
3	9.634356	4.278725	4.275795
4	7.671892	3.158348	3.166970
5	8.592127	3.518352	3.543422
6	7.747036	3.397380	3.384179
7	9.623175	4.601825	4.603800
8	7.735412	3.852066	3.848137
9	12.083202	5.502614	5.521156
10	11.080817	4.847354	4.790418
11	11.631615	5.017371	4.996615
12	6.477359	3.023023	3.036469
13	8.870094	3.708882	3.764201
14	12.703396	6.073339	6.076483
15	11.416840	5.444431	5.441192
16	10.758823	5.286965	5.297677
17	11.464688	5.596979	5.604316
18	9.747457	4.003333	4.012955
19	12.211823	6.092597	6.100131
20	12.671249	6.184386	6.193254



R code 5.30

```
m5.8 <- map(
    alist(
        height ~ dnorm( mu , sigma ) ,
        mu <- a + bl*leg_left + br*leg_right ,
        a ~ dnorm( 10 , 100 ) ,
        bl ~ dnorm( 2 , 10 ) ,
        br ~ dnorm( 2 , 10 ) ,
        sigma ~ dunif( 0 , 10 )
    ) ,
    data=d )
precis(m5.8)
```

```
       Mean StdDev  5.5% 94.5%
a      0.70   0.31  0.20  1.20
bl    -0.43   2.18 -3.92  3.06
br     2.48   2.19 -1.01  5.98
sigma  0.62   0.04  0.55  0.69
```



Multicollinear legs

- Q: What is value of learning left/right leg, once we already know right/left leg?
- A: Almost nothing, on average.

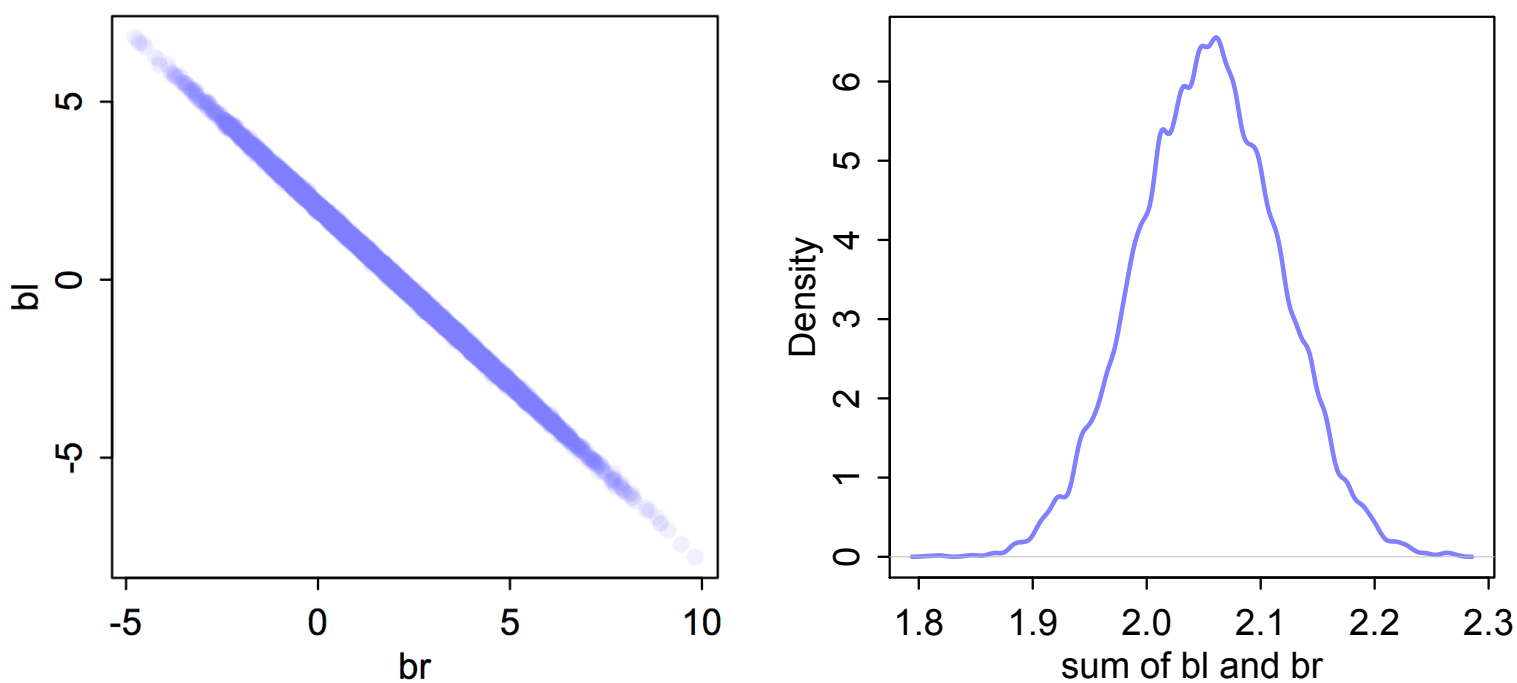


Figure 5.8

Multicollinear legs

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta_1 x_i + \beta_2 x_i = \alpha + (\beta_1 + \beta_2) x_i$$

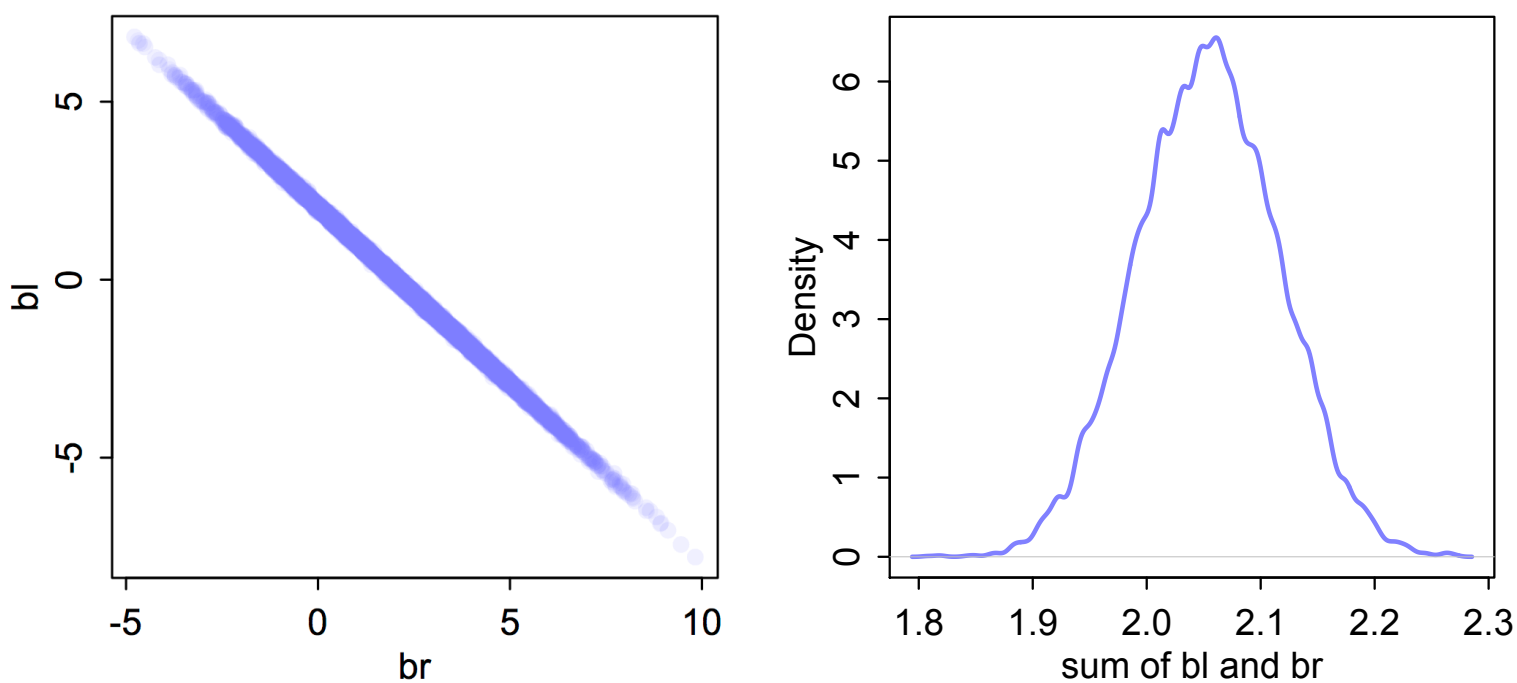


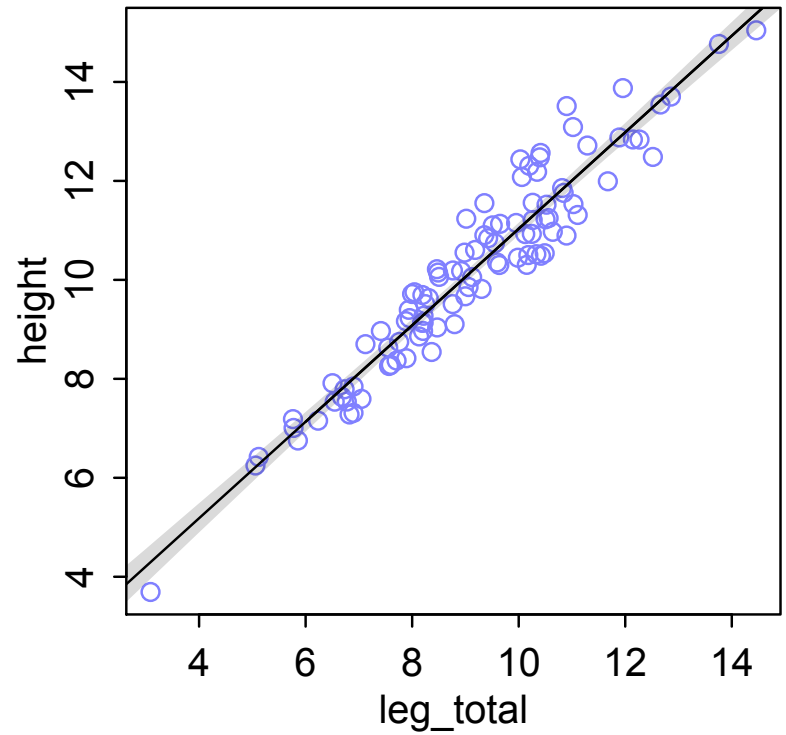
Figure 5.8

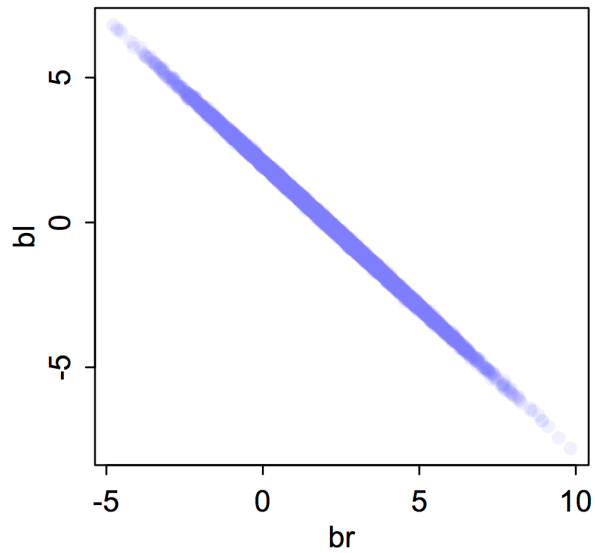
Model did what you asked!

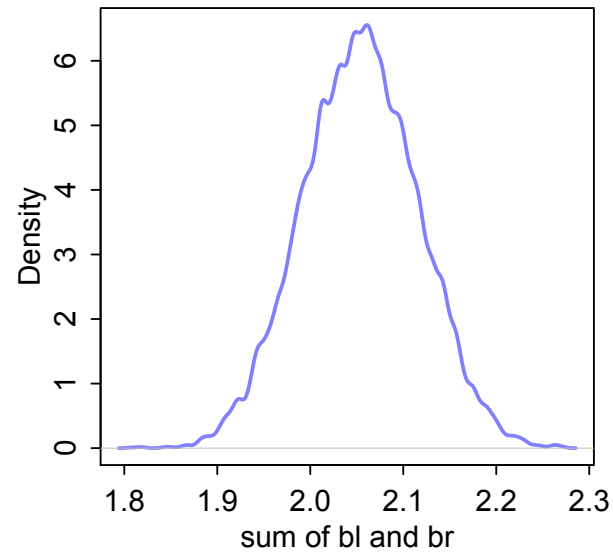
```
d$leg_total <- d$leg_left + d$leg_right
plot( height ~ leg_total , d , col=rangi2 )

leg_list <- seq(from=1,to=15,length.out=30)
leg_dat <- list(leg_left=leg_list,
                leg_right=leg_list)
mu <- link( m5.8 , data=leg_dat )
mu.mean <- apply( mu , 2 , mean )
mu.PI <- apply( mu , 2 , PI )

lines( leg_list*2 , mu.mean )
shade( mu.PI , leg_list*2 )
```

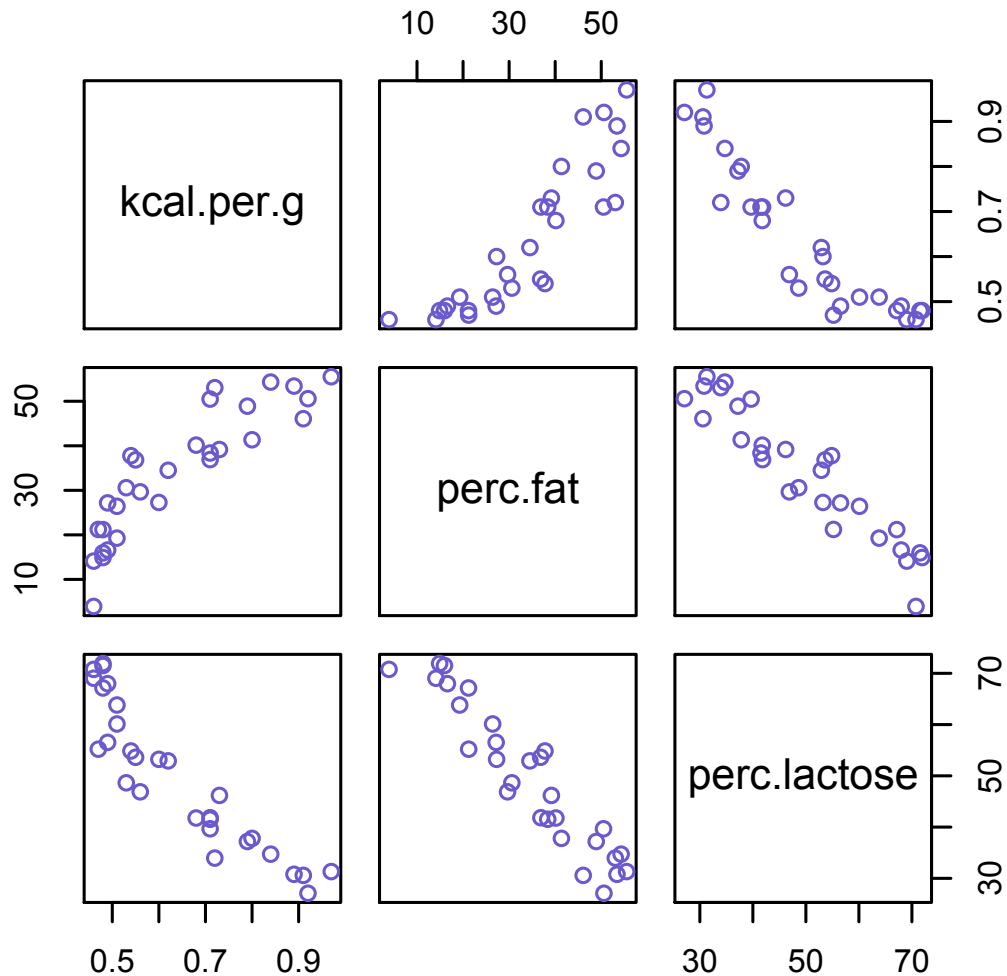






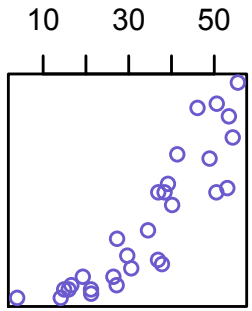
Correlated predictors

- *Multicollinearity*: strong correlations among prediction variables

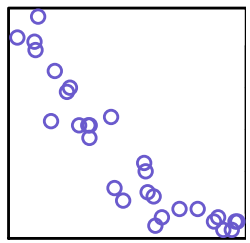


Correlated predictors

- `perc.fat` or `perc.lactose` alone: Strong association with `kcal.per.g`



	Mean	StdDev	5.5%	94.5%
a	0.301	0.036	0.244	0.358
bf	0.010	0.001	0.008	0.012
sigma	0.073	0.010	0.058	0.089



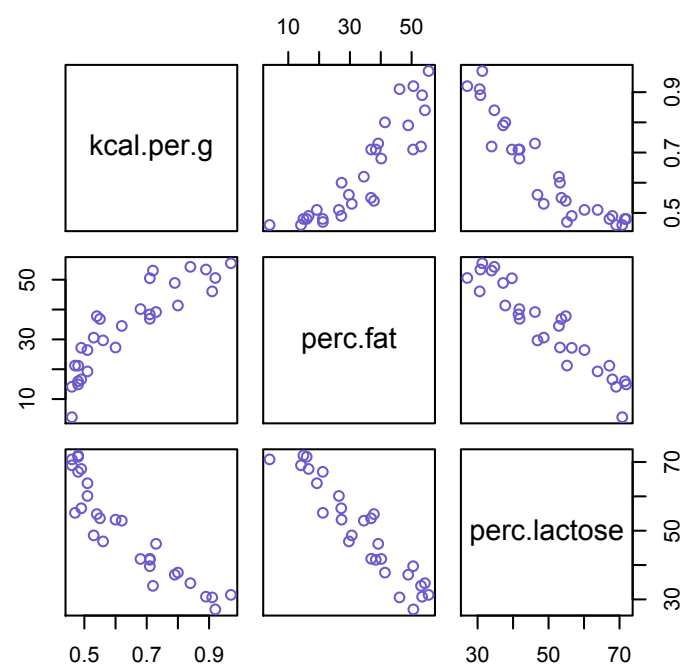
	Mean	StdDev	5.5%	94.5%
a	1.166	0.043	1.098	1.235
bl	-0.011	0.001	-0.012	-0.009
sigma	0.062	0.008	0.049	0.075

- Together: Both reduced association?

R code 5.37

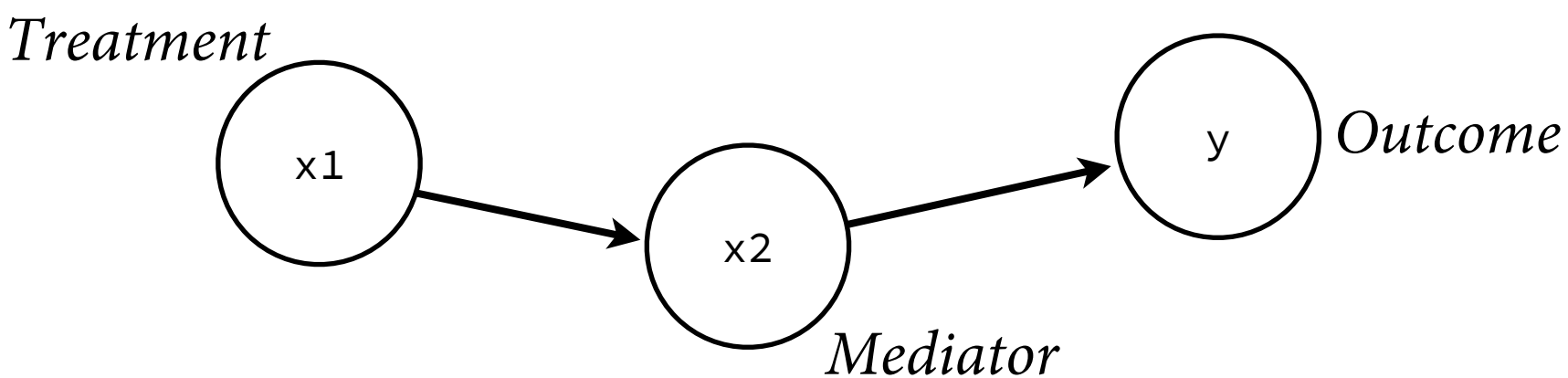
```
m5.12 <- map(
    alist(
        kcal.per.g ~ dnorm( mu , sigma ) ,
        mu <- a + bf*perc.fat + bl*perc.lactose ,
        a ~ dnorm( 0.6 , 10 ) ,
        bf ~ dnorm( 0 , 1 ) ,
        bl ~ dnorm( 0 , 1 ) ,
        sigma ~ dunif( 0 , 10 )
    ) ,
    data=d )
precis( m5.12 , digits=3 )
```

```
         Mean StdDev   5.5%  94.5%
a       1.007  0.200  0.688  1.327
bf      0.002  0.002 -0.002  0.006
bl     -0.009  0.002 -0.013 -0.005
sigma   0.061  0.008  0.048  0.074
```

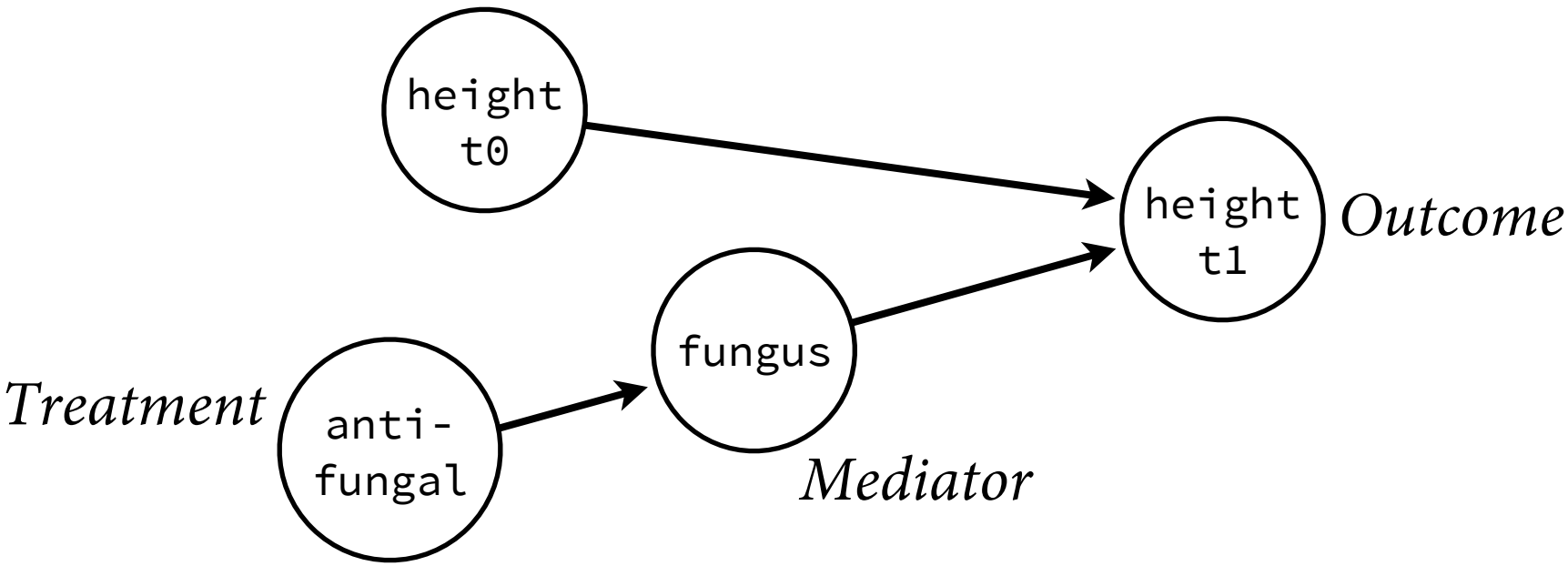


Post-treatment bias

- Headline: Thoughtlessly adding predictors is a bad idea.
- Another danger: *Post-treatment bias*
  Controlling for consequence of treatment statistically knocks out treatment



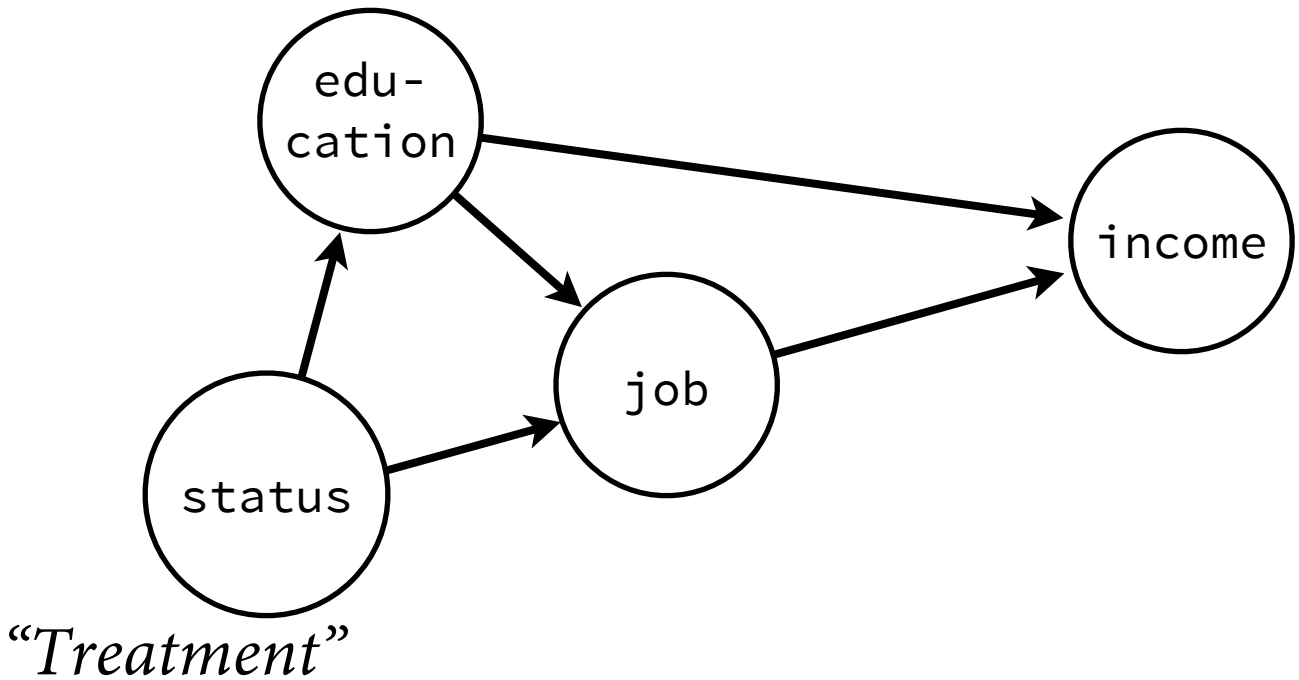
Post-treatment bias



Section 5.3.3, page 150

Post-treatment bias

Observational studies harder

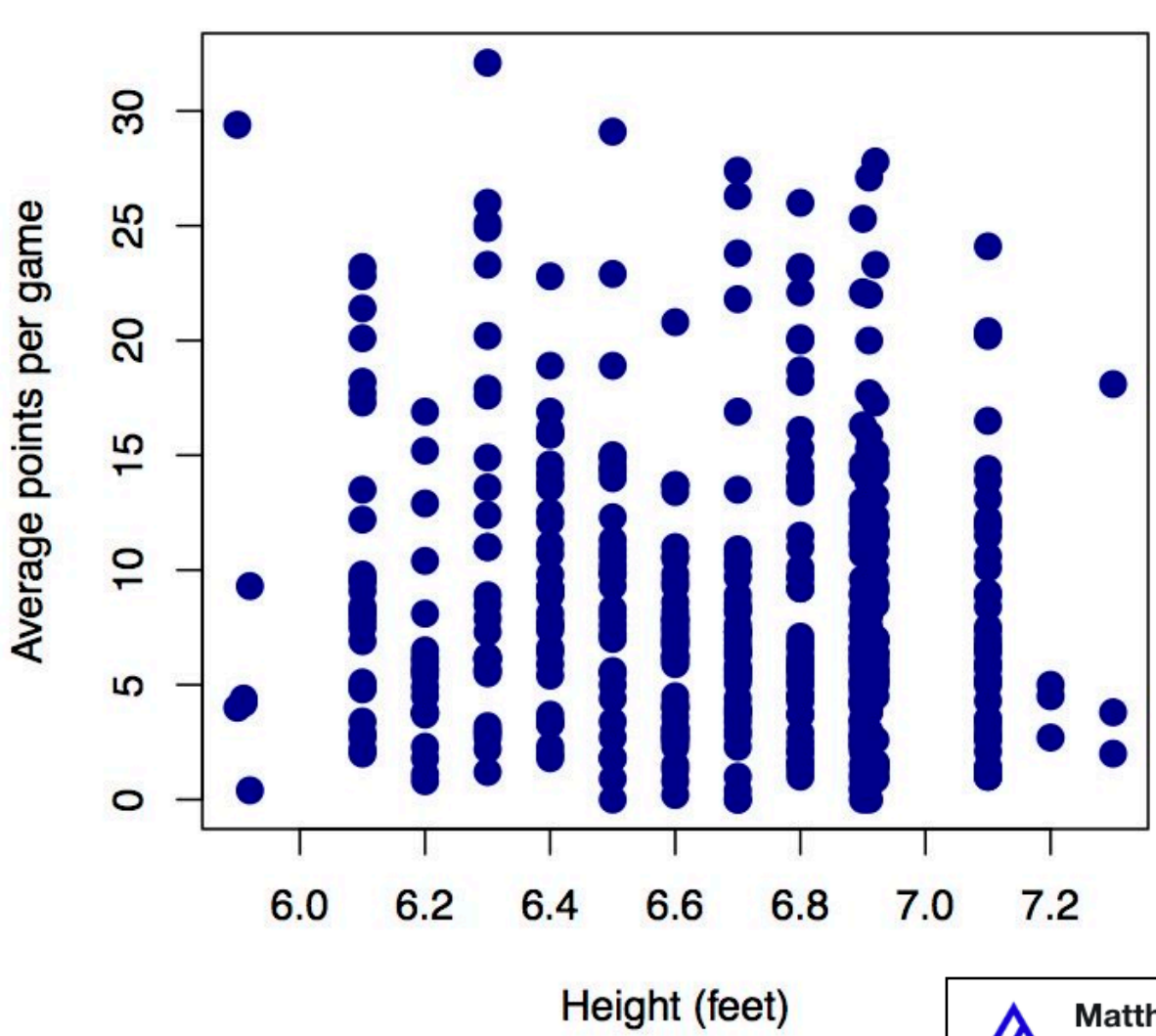


Beware the Collider

- ***Collider***: A variable X that is influenced by two other variables, Y and Z
- Want to know Z ~ Y
- Don't condition on X (or anything X causes)

- Common trap: Selection on X forces conditioning on X

Are taller people better at basketball?



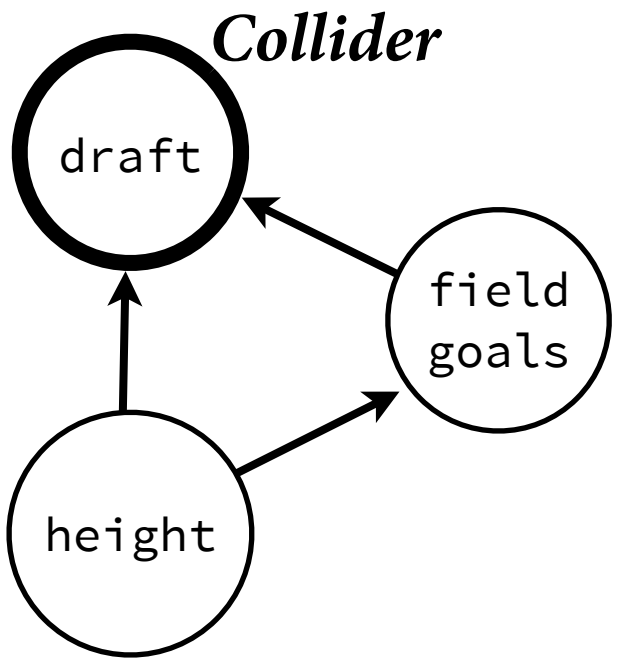
473 NBA players, 2016-2017 season

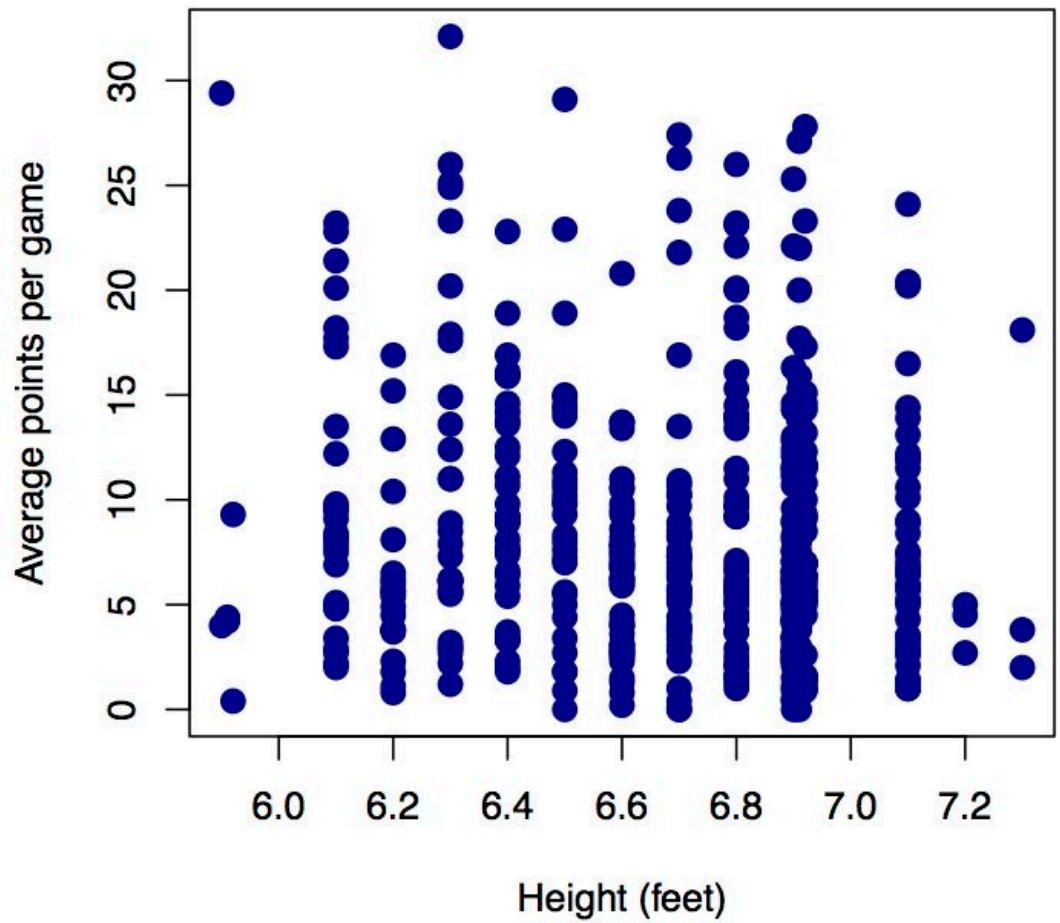
Matthew Hahn
@3rdreviewer
Following

You can be a professional basketball player, no matter how tall you are!
No correlation between height and scoring success in the NBA:

Beware the Collider

Are taller people better at basketball?

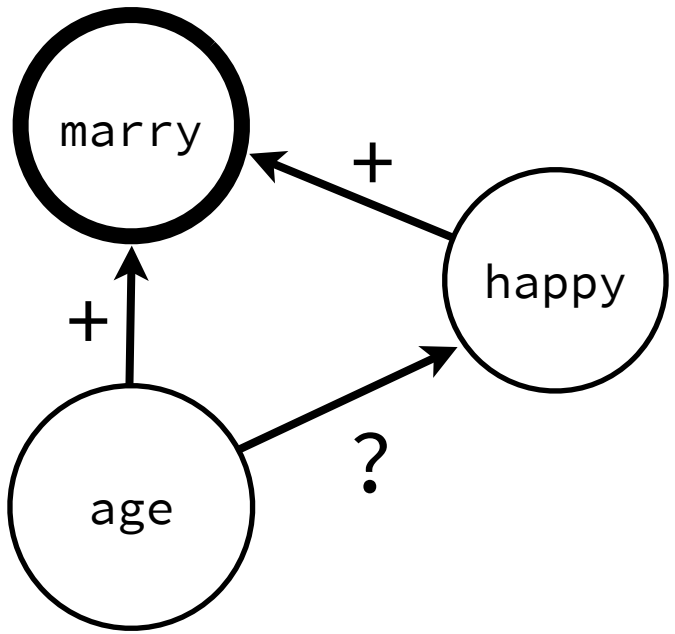




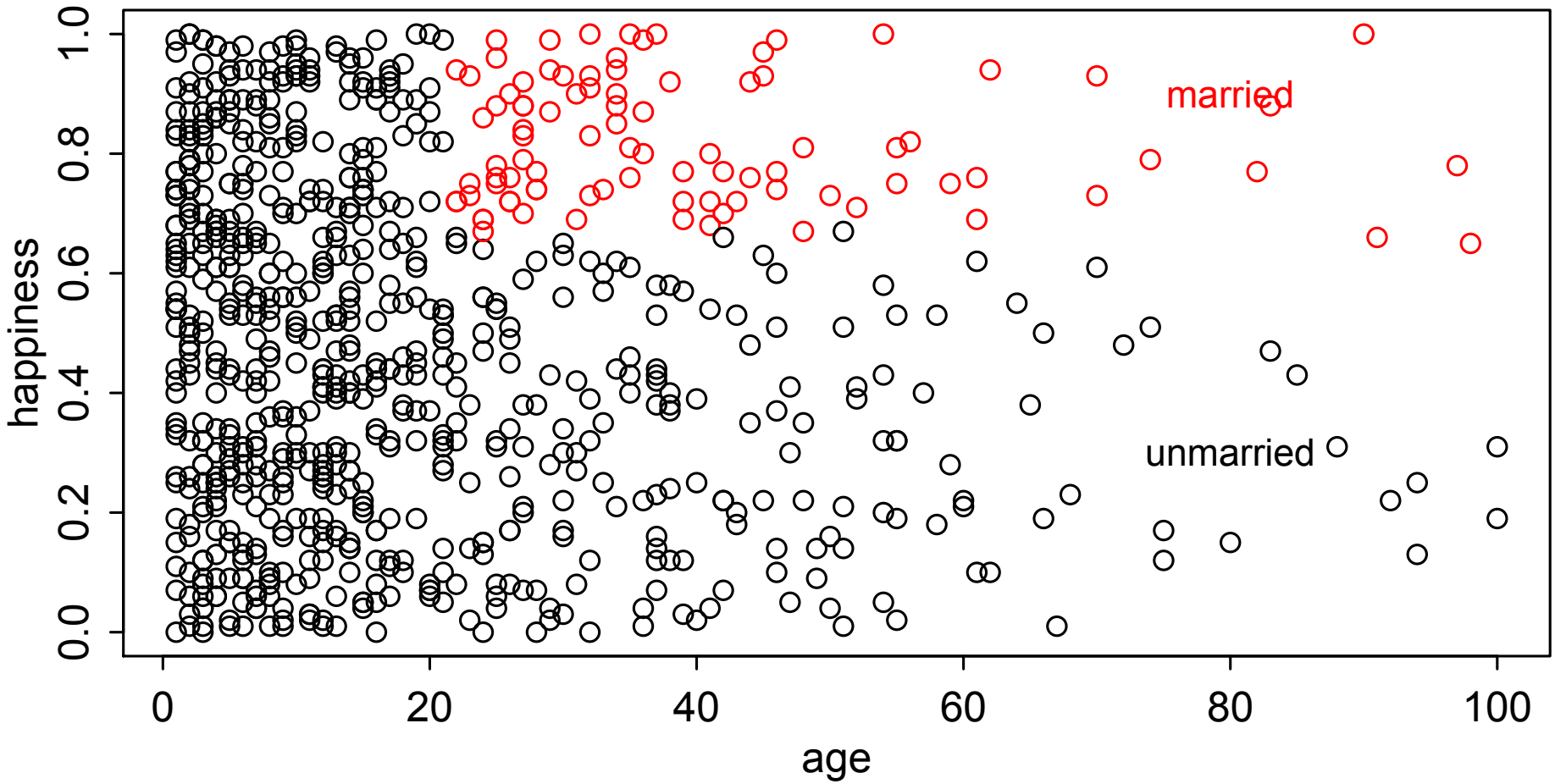
Beware the Collider

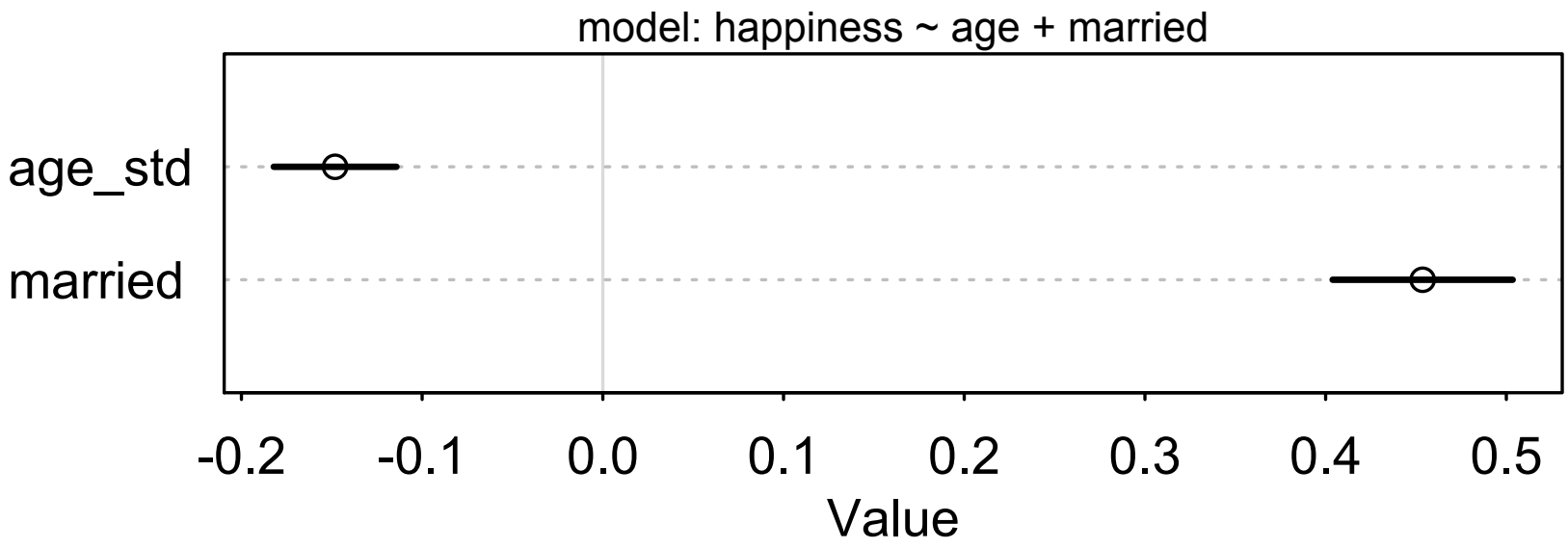
Also happens with model specification.

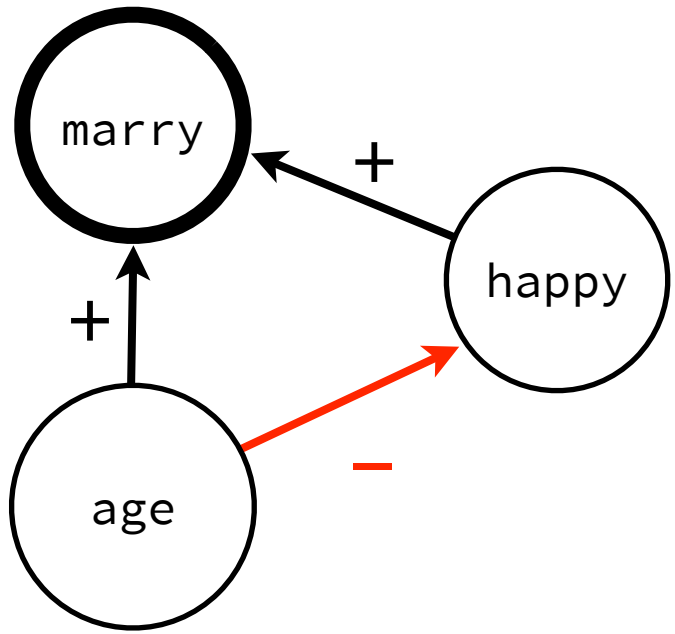
Are older people less happy? Should we control for marriage status?



No relationship between age & happiness.
5 happiest people get married each year.
What happens when we control for marriage status?



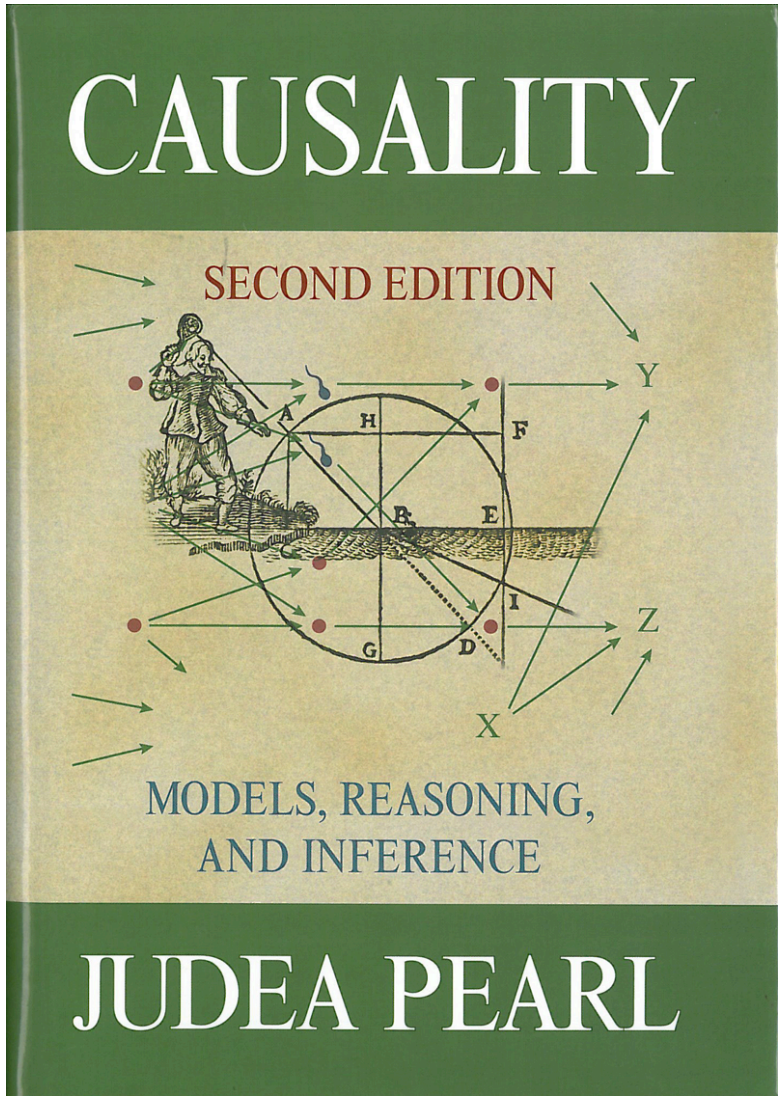

model: happiness ~ age + married
age_std
married
-0.2
-0.1
0.0
0.1
0.2
0.3
0.4
0.5
Value


marry
happy
age
+
+
−


model: happiness ~ age + married
age_std
married
-0.2
-0.1
0.0
0.1
0.2
0.3
0.4
0.5
Value
model: happiness ~ age
age_std
-0.2
-0.1
0.0
0.1
0.2
0.3
0.4
0.5
Value

Additional Nightmares

- Causal inference is very hard™
  - Residual confounding looms
  - Always model dependent
  - Real interventions change many variables at once
  - Complex systems –> everything "causes" everything
- No secret weapon



Next week

- Homework: 5H1, 5H2, 5H3
- Please put your name in the file
- Next week, we are in the big lecture hall downstairs
- Next week, Chapter 6
  - Sailing between
    (1) the whirlpool of *underfitting*
    (2) the many-headed monster of *overfitting*