AARHUS UNIVERSITY

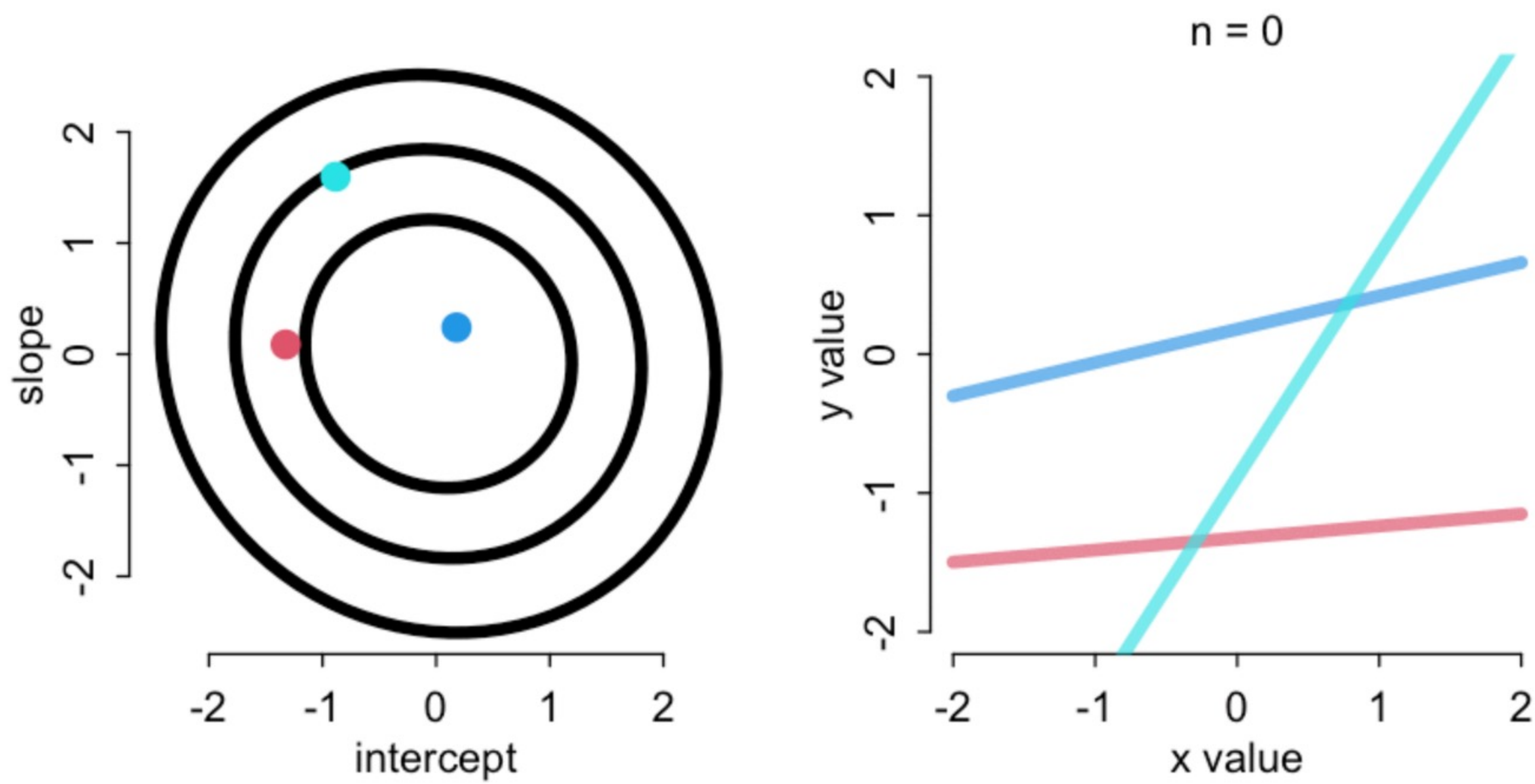
Methods 4 - 9

Chris Mathys

SOLIDUM PETIT IN PROFUNDIS · UNIVERSITAS ARHUSIENSIS ·

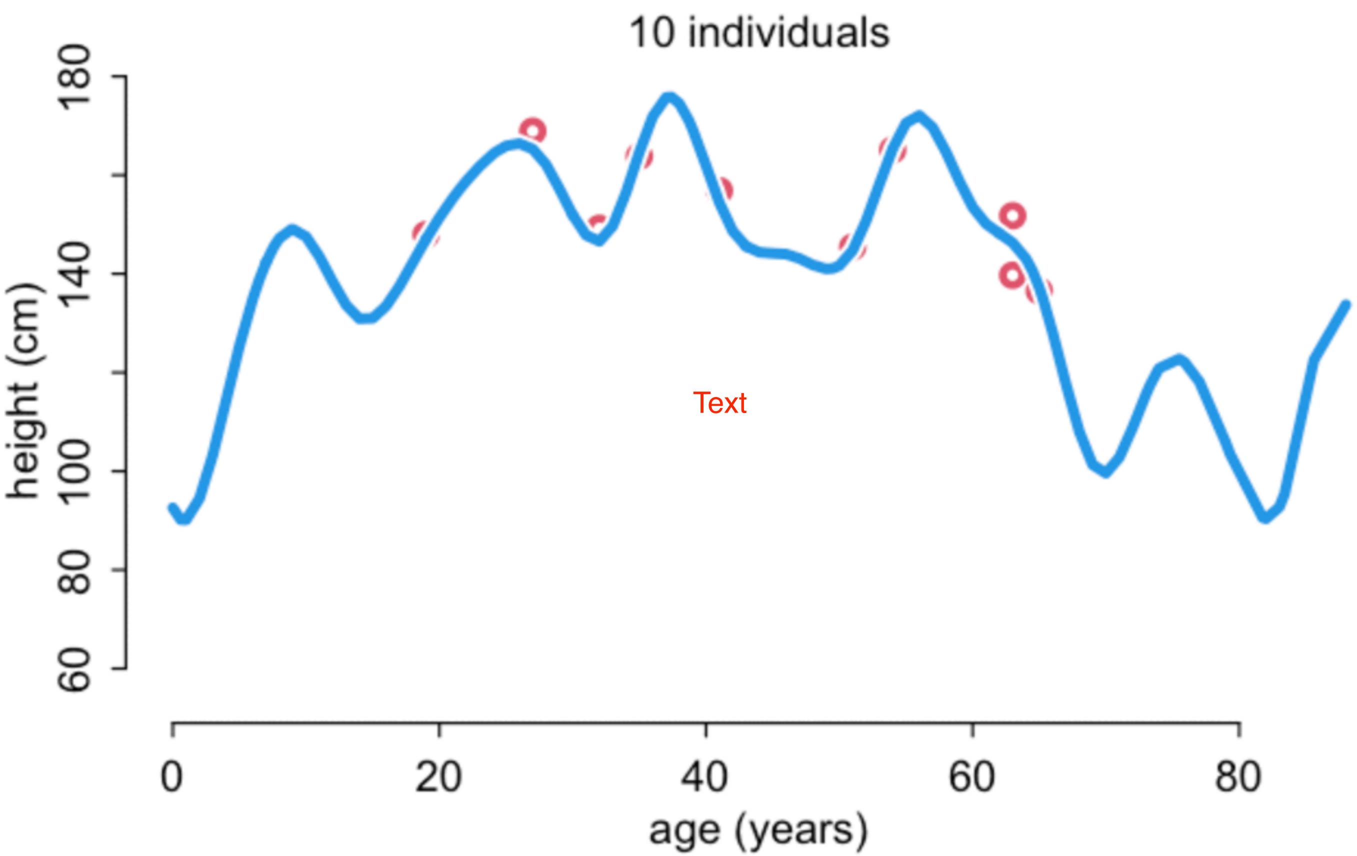
BSc Programme in Cognitive Science

Spring 2022



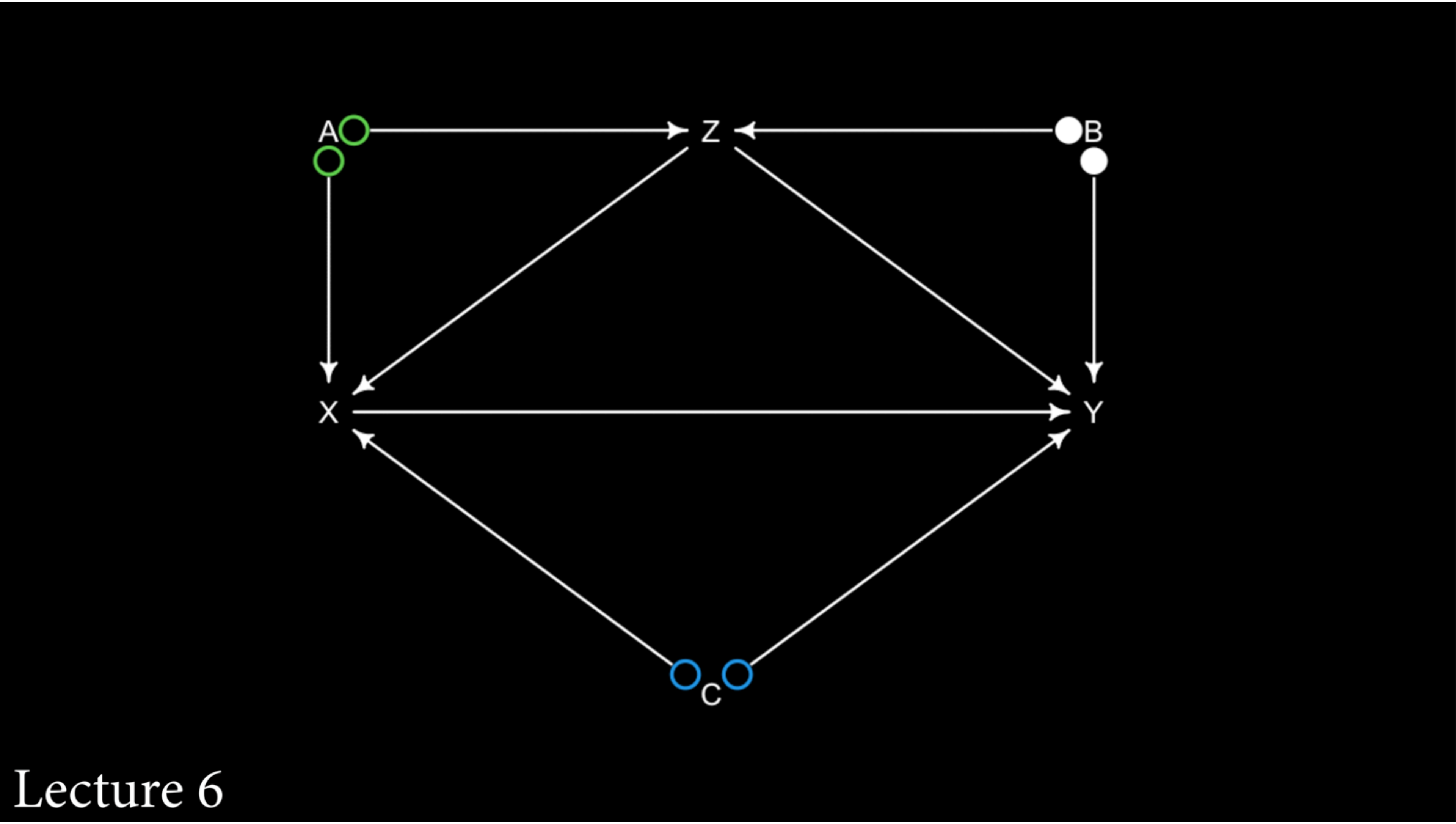
Lecture 3

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

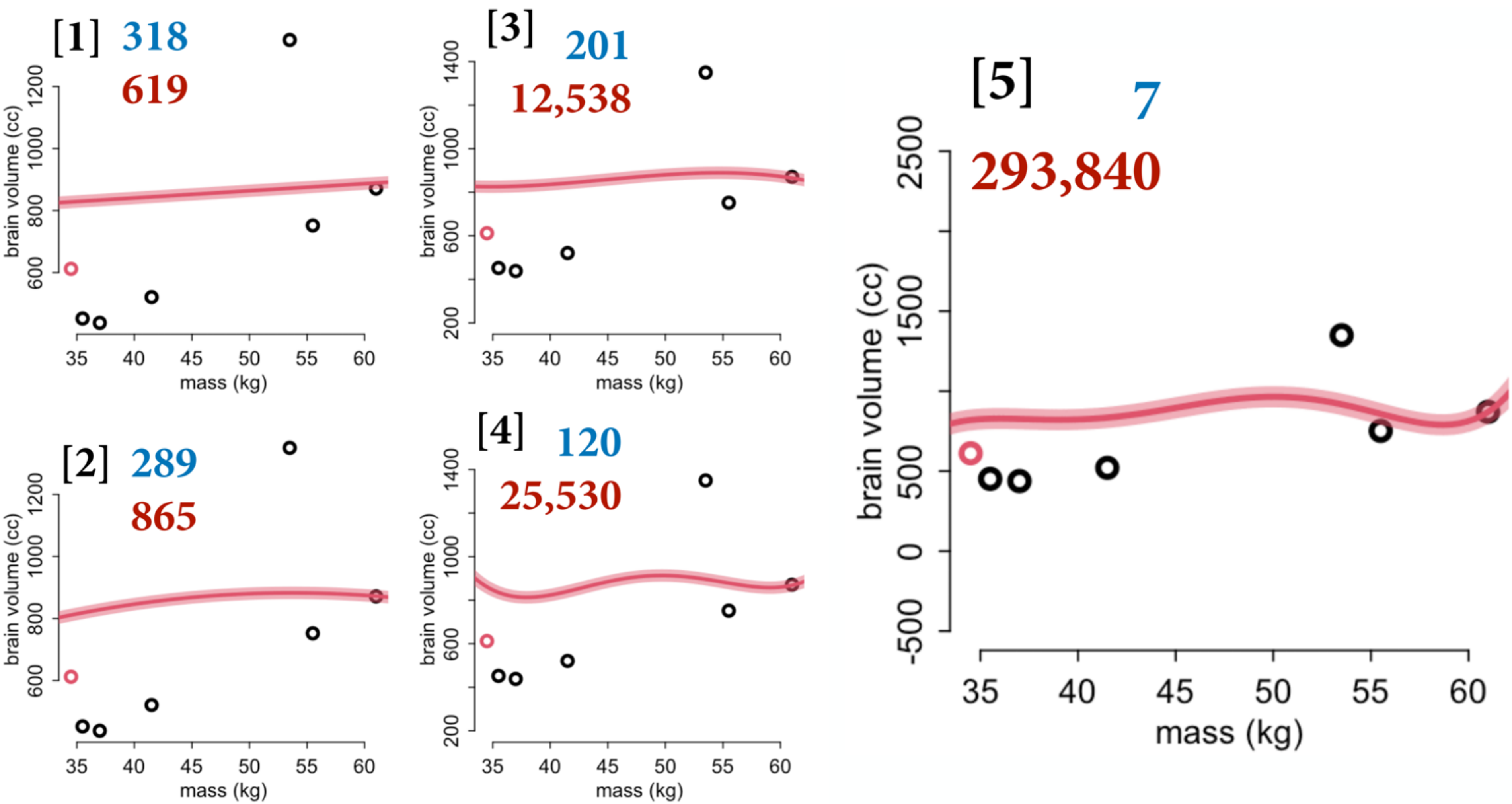


Lecture 4

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

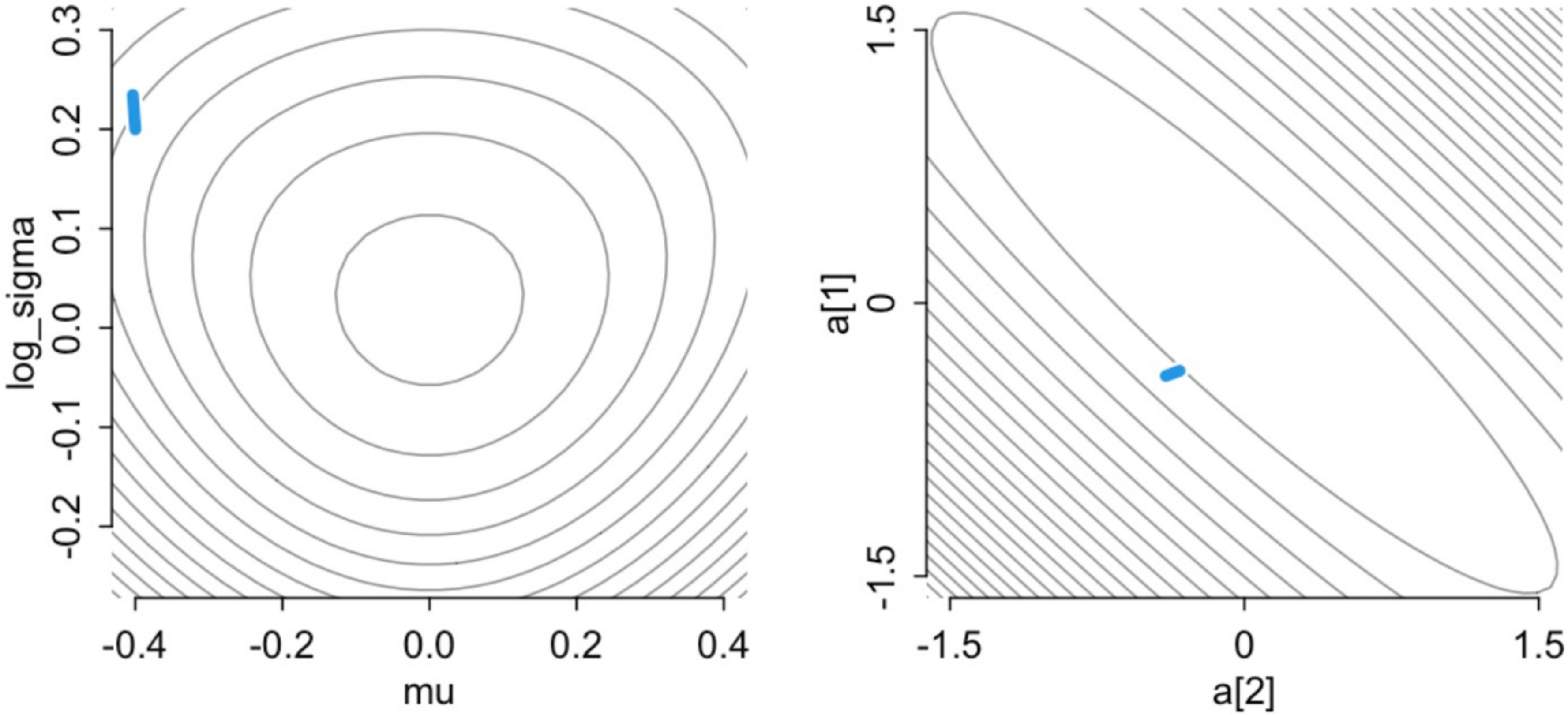


Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022



Lecture 7

Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022



Lecture 8

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

UC Berkeley Admissions

4526 graduate school applications for 1973 UC Berkeley

Stratified by department and gender of applicant

Evidence of gender discrimination?

	dept	admit	reject	applications	gender
1	A	512	313	825	male
2	A	89	19	108	female
3	B	353	207	560	male
4	B	17	8	25	female
5	C	120	205	325	male
6	C	202	391	593	female
7	D	138	279	417	male
8	D	131	244	375	female
9	E	53	138	191	male
10	E	94	299	393	female
11	F	22	351	373	male
12	F	24	317	341	female

See ?UCBAdmissions for citation

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Admissions: Drawing the Owl

(1) Estimand(s)

(2) Scientific model(s)

(3) Statistical model(s)

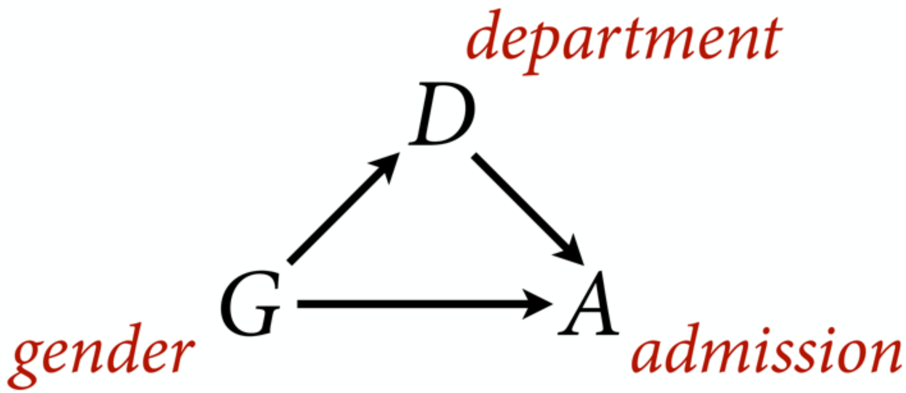
(4) Analyze

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Admissions

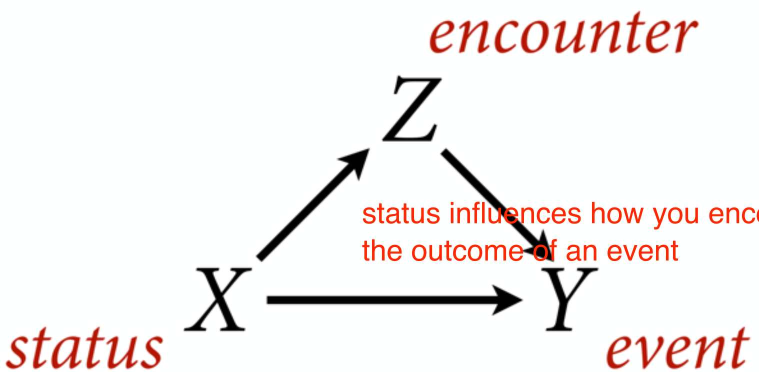
gender $G \longrightarrow A$ *admission*

Admissions



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

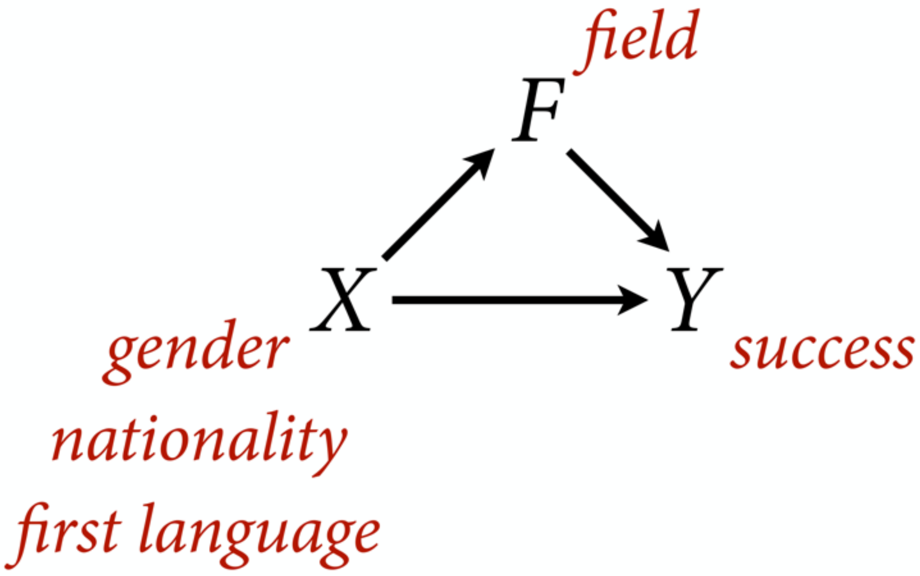
Encounters & discrimination



status influences how you encounter things, which together influence the outcome of an event

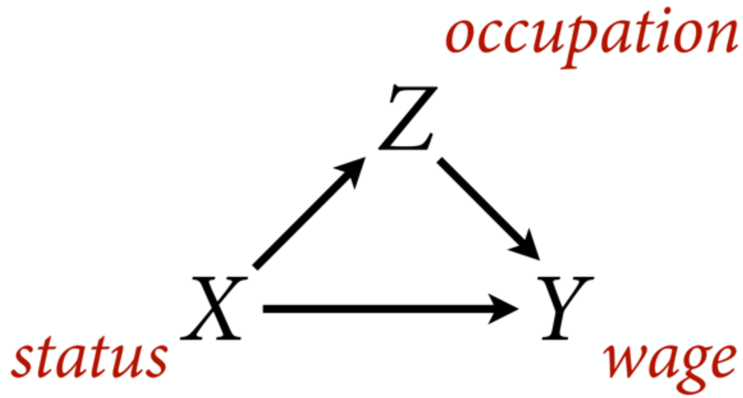
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Grants



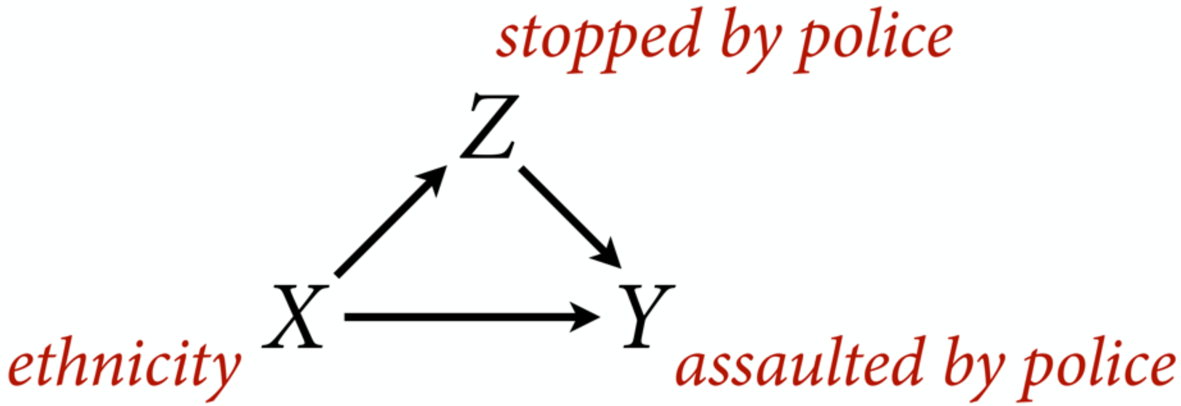
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Wage discrimination



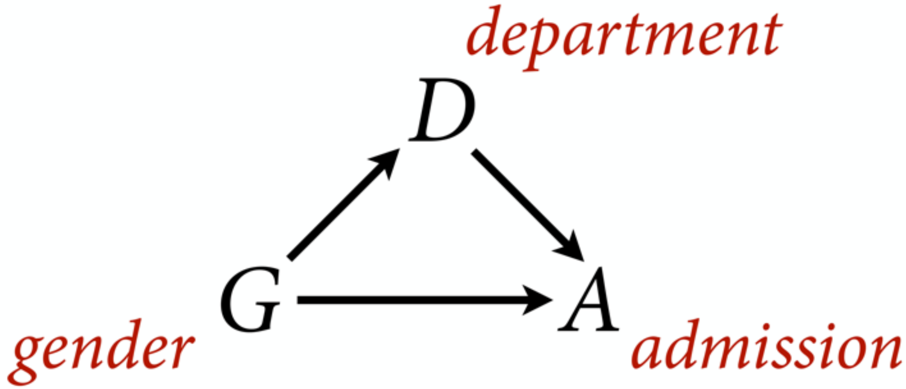
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Policing



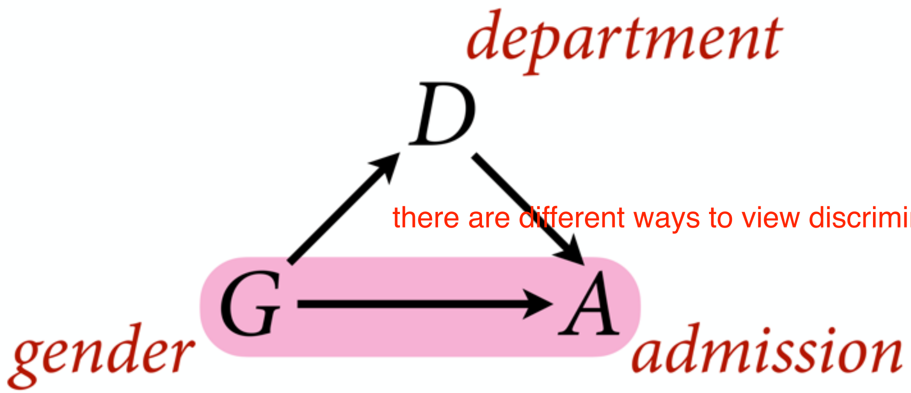
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Which path is "discrimination"?



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Which path is "discrimination"?

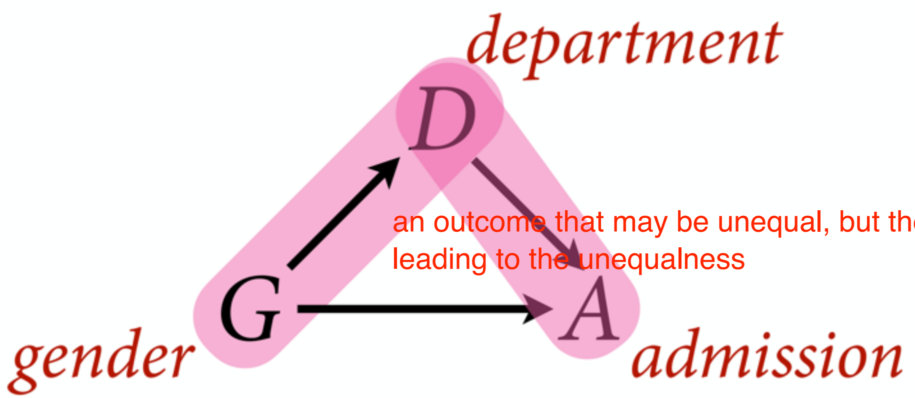


there are different ways to view discrimination.

Direct discrimination
(status-based or taste-based discrimination)

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Which path is “discrimination”?



an outcome that may be unequal, but there was no conscious decision leading to the unequalness

Indirect discrimination
(structural discrimination)

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Admissions: Drawing the Owl

(1) Estimand(s)

(2) Scientific model(s)

(3) Statistical model(s)

(4) Analyze

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

Constructed example:
We sample men and women.

How can choice of department create structural discrimination?

When departments vary in baseline admission rates.

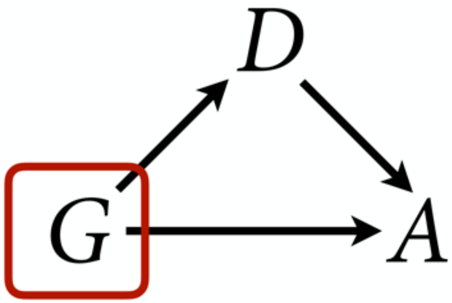
```
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.1,0.3,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```

```
> accept_rate
     [,1] [,2]
[1,]  0.1  0.1
[2,]  0.3  0.3
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

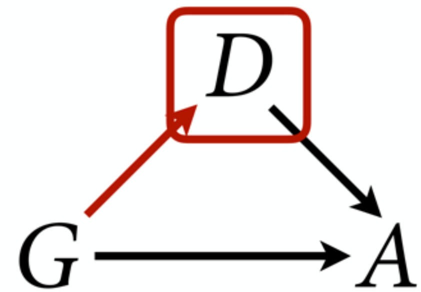
```
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.1,0.3,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```



Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

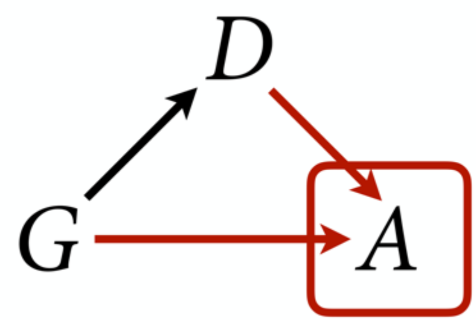
```
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.1,0.3,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

```r
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.1,0.3,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```




```
> accept_rate
     [,1] [,2]
[1,]  0.1  0.1
[2,]  0.3  0.3
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

```
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.1,0.3,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```

```
> table(G,D)
   D
G     1   2
  1 361 161
  2  99 379
```

```
> table(G,A)
   A
G     0   1
  1 421 101
  2 350 128
```

Accept rates
Gender 1: 19%
Gender 2: 27%

Text

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

```
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.05,0.2,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```

```
> table(G,D)
   D
G     1   2
  1 355 164
  2  95 386
```

```
> table(G,A)
   A
G     0   1
  1 473  46
  2 404  77
```

after running the simulation again:

Accept rates

Gender 1: 9%

Gender 2: 16%

```
> accept_rate
     [,1] [,2]
[1,] 0.05  0.1
[2,] 0.20  0.3
```

Text

now there is systematic discrimination against 1

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generative model

Is a start, but lots missing

Admission rate usually depends upon size of applicant pool, distribution of qualifications

In principle, should sample applicant pool and then sort to select admissions

Rates are conditional on structure of applicant pool

```
> table(G,D)
   D
G     1   2
  1 355 164
  2  95 386
```

```
> table(G,A)
   A
G     0   1
  1 473  46
  2 404  77
```

Accept rates
Gender 1: 9%
Gender 2: 16%

there are lots of problems with analyzing these kind of data.

Admissions: Drawing the Owl

(1) Estimand(s)

(2) Scientific model(s)

(3) Statistical model(s)

(4) Analyze

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generalized Linear Models

Linear Models: Expected value is additive ("linear") combination of parameters

$$Y_i \sim \text{Normal}(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta_X X_i + \beta_Z Z_i$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generalized Linear Models

Linear Models: Expected value is additive ("linear") combination of parameters

$$Y_i \sim \text{Normal}(\mu_i, \sigma)$$
$$\mu_i = \alpha + \beta_X X_i + \beta_Z Z_i$$

Generalized Linear Models: Expected value is some function of an additive combination of parameters

$$Y_i \sim \text{Bernoulli}(p_i)$$
$$f(p_i) = \alpha + \beta_X X_i + \beta_Z Z_i$$

Text

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Generalized Linear Models

Text

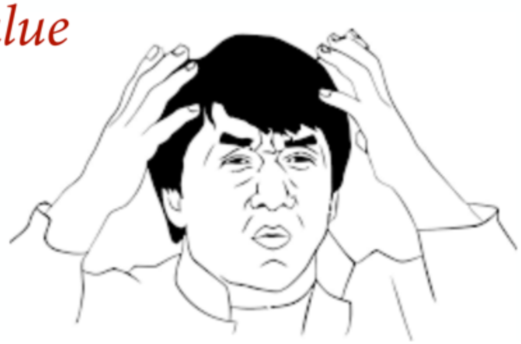
0/1 outcome

probability of event

$$Y_i \sim \text{Bernoulli}(p_i)$$

$$f(p_i) = \underbrace{\alpha + \beta_X X_i + \beta_Z Z_i}_{\text{can take any real value}}$$

f() maps probability scale to linear model scale



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Links and inverse links

the functions that links the outcome to the linear predictor = link function (in our case we use the inverse logit).

f is the link function

$$Y_i \sim \text{Bernoulli}(p_i)$$
$$f(p_i) = \alpha + \beta_X X_i + \beta_Z Z_i$$

Links parameters of distribution to linear model

f^{-1} is the inverse link function

$$p_i = f^{-1}(\alpha + \beta_X X_i + \beta_Z Z_i)$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Example inverse function

$$f(a) = a^2 = b$$

Text

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Example inverse function

$$f(a) = a^2 = b$$

Text

$$f^{-1}(b) = \sqrt{b} = a$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Logit link

Bernoulli/Binomial models most often use logit link

$$Y_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha + \beta_X X_i + \beta_Z Z_i$$

$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i}$$

$$p_i = \text{logit}^{-1}(\alpha + \beta_X X_i + \beta_Z Z_i)$$

Text

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Logit link

Bernoulli/Binomial models most often use logit link

$$Y_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha + \beta_X X_i + \beta_Z Z_i$$

$$\text{logit}(p_i) = \log \boxed{\frac{p_i}{1 - p_i}}$$

"log odds" *odds*

$$p_i = \text{logit}^{-1}(\alpha + \beta_X X_i + \beta_Z Z_i)$$

"logistic"

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

From link to inverse link

$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i} = q_i$$

$$\text{logit}^{-1}(q_i) = ?$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

From link to inverse link

$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i} = q_i$$

$$\text{logit}^{-1}(q_i) = ? = p_i$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

From link to inverse link

$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i} = q_i$$

$$\text{logit}^{-1}(q_i) = \,? = p_i$$

$$\log \frac{p_i}{1 - p_i} = q_i$$

$$\boxed{p_i = \frac{\exp(q_i)}{1 + \exp(q_i)}} \quad \text{logit}^{-1}$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

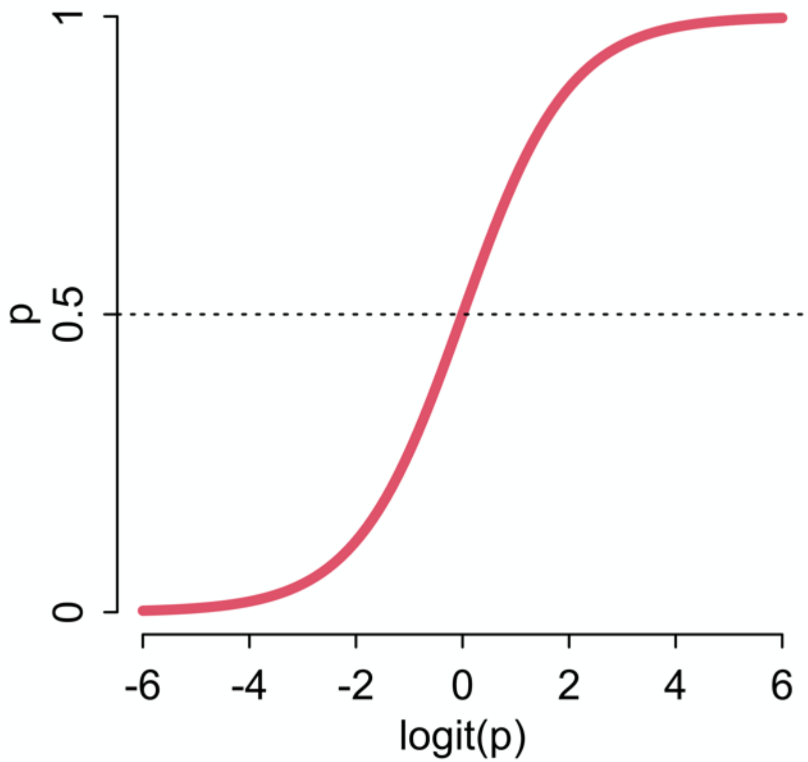
logit link is a harsh transform

"log-odds scale": The value of the linear model

logit(p)=0, p=0.5

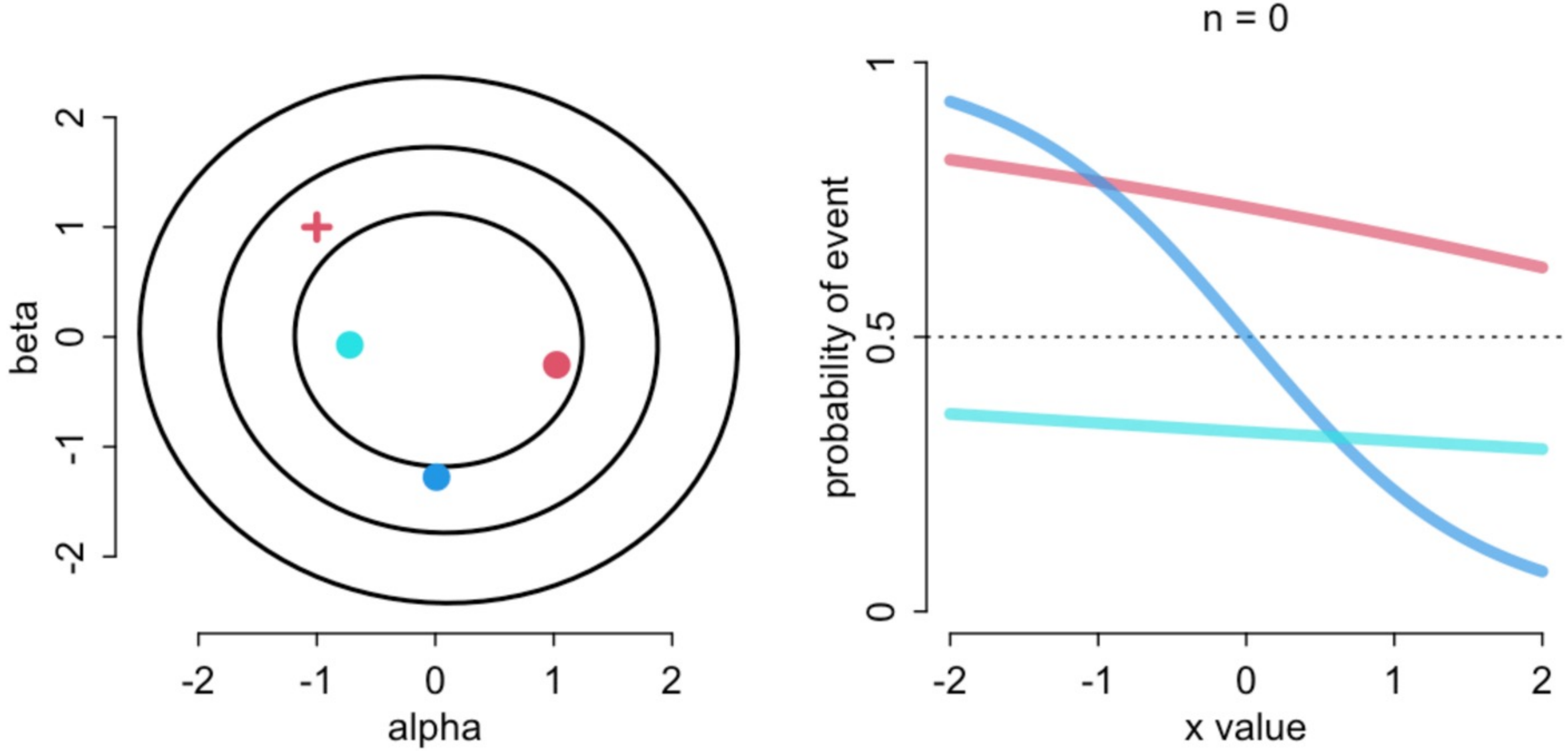
logit(p)=6, p=always

logit(p)=–6, p=never



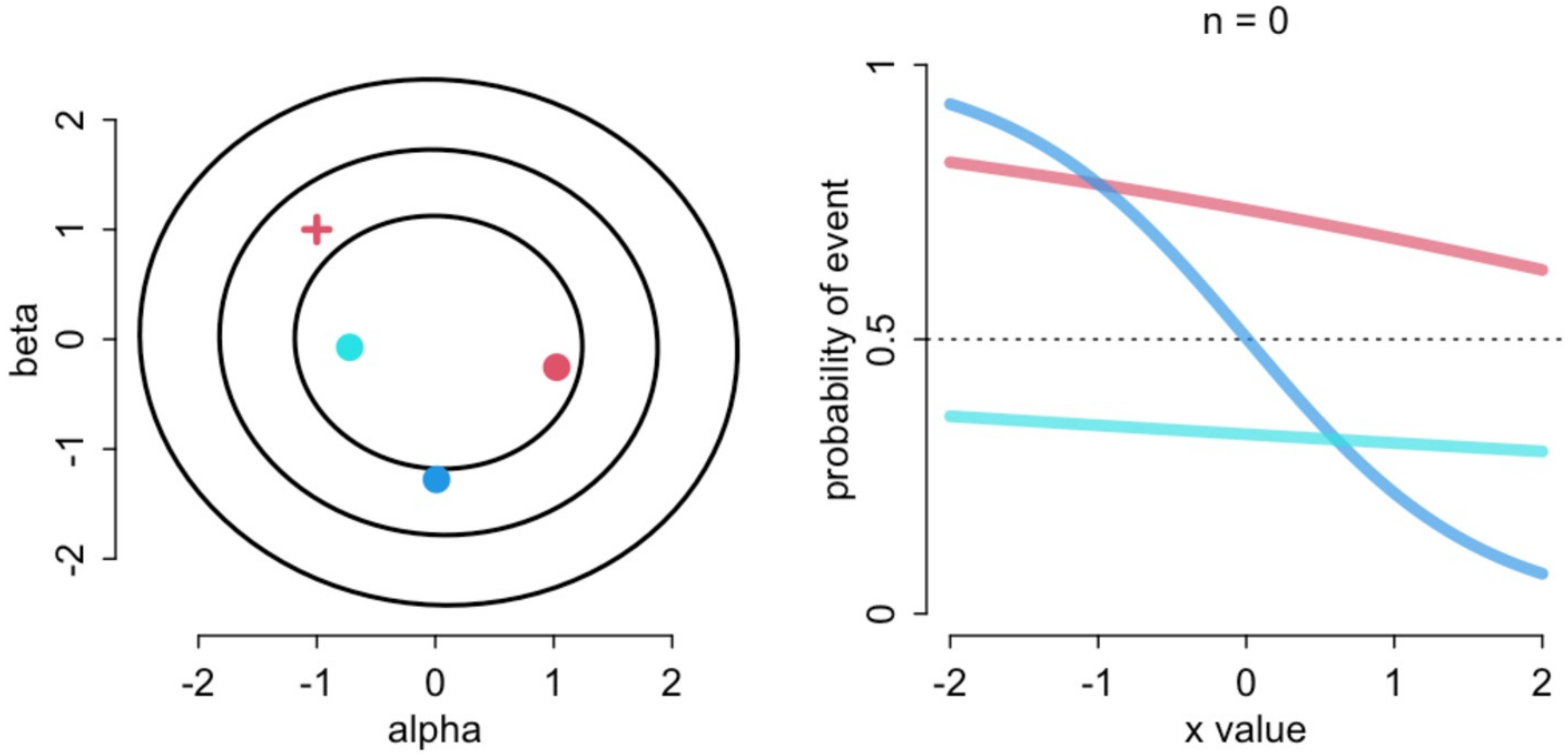
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

$$\text{logit}(p_i) = \alpha + \beta x_i$$



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

$$\text{logit}(p_i) = \alpha + \beta x_i$$



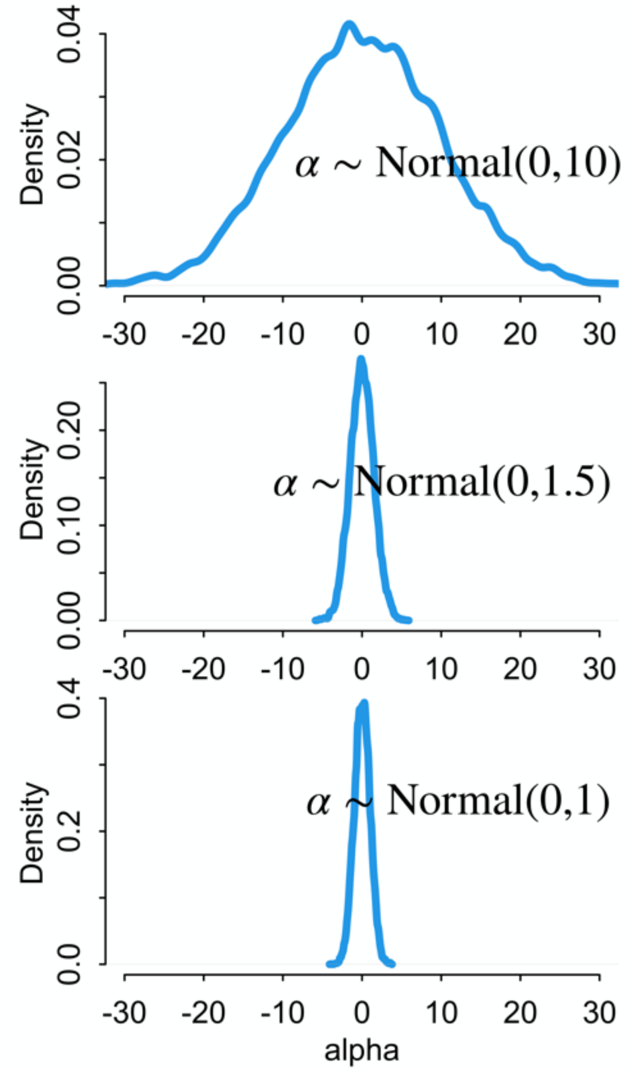
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Logistic priors

$$\text{logit}(p_i) = \alpha$$

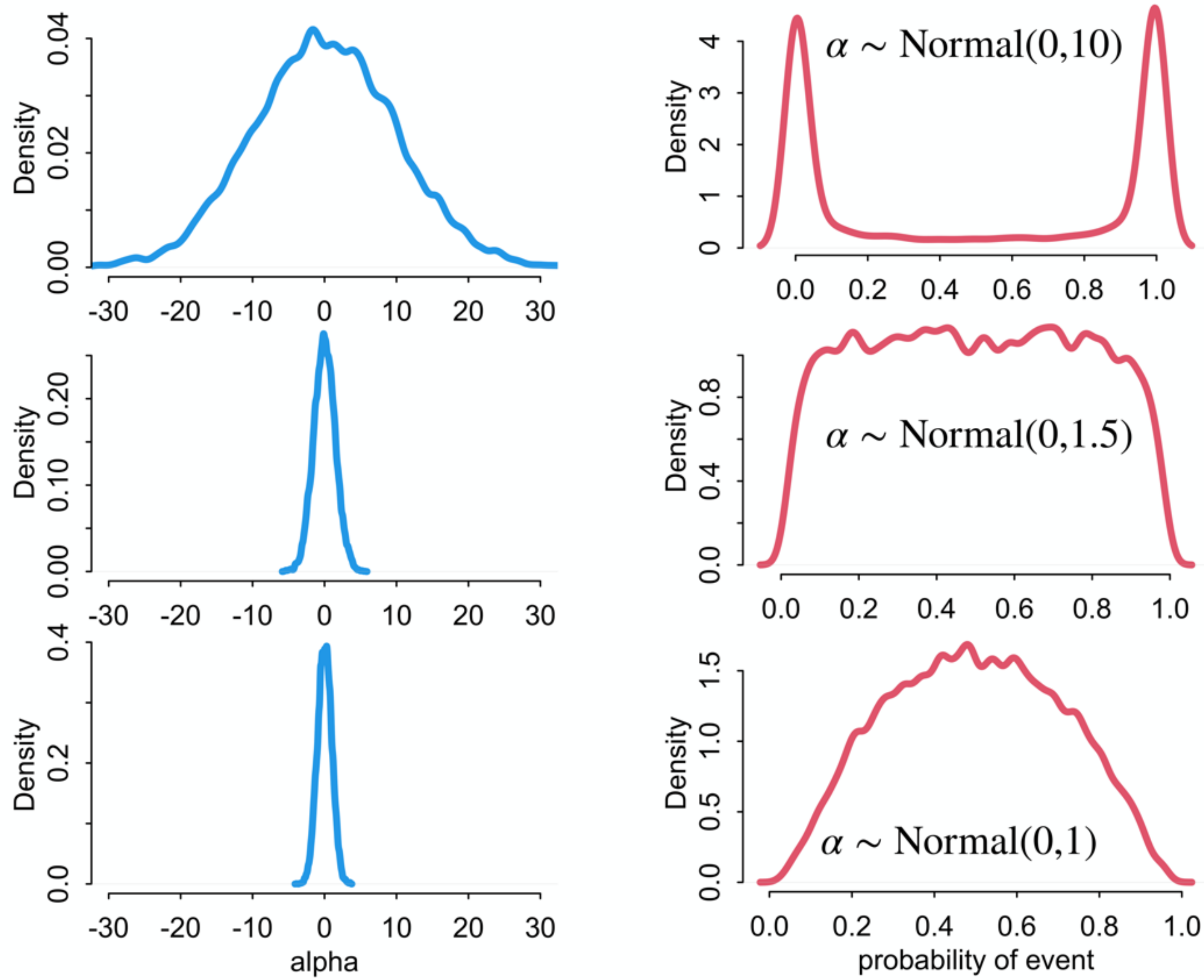
The logit link compresses parameter distributions

Anything above +4 = almost always
Anything below –4 = almost never



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

$$\text{logit}(p_i) = \alpha$$

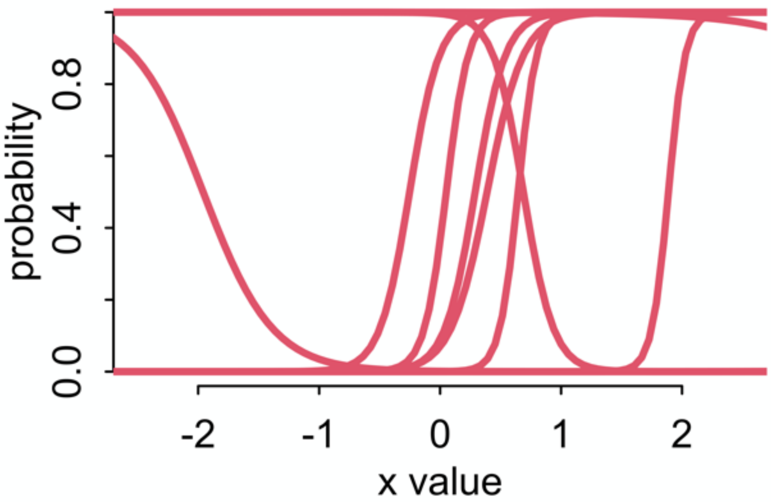


Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022

$$\text{logit}(p_i) = \alpha + \beta x_i$$

$$\alpha \sim \text{Normal}(0,10)$$
$$\beta \sim \text{Normal}(0,10)$$

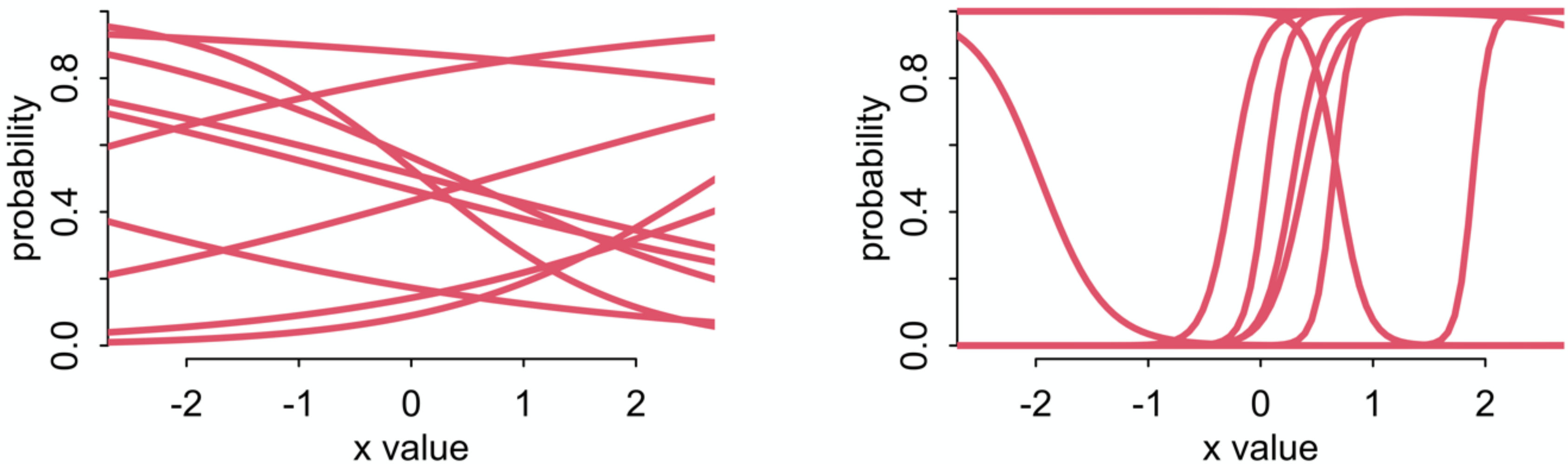
```
a <- rnorm(1e4,0,10)
b <- rnorm(1e4,0,10)

xseq <- seq(from=-3,to=3,len=100)
p <- sapply( xseq , function(x)
inv_logit(a+b*x) )

plot( NULL , xlim=c(-2.5,2.5) , ylim=c(0,1) ,
xlab="x value" , ylab="probability" )
for ( i in 1:10 ) lines( xseq , p[i,] , lwd=3 ,
col=2 )
```



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

$\alpha \sim \text{Normal}(0,1.5)$
$\beta \sim \text{Normal}(0,0.5)$

$\alpha \sim \text{Normal}(0,10)$
$\beta \sim \text{Normal}(0,10)$



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

A statistical model

```r
# generative model, basic mediator scenario

N <- 1000 # number of applicants
# even gender distribution
G <- sample( 1:2 , size=N , replace=TRUE )
# gender 1 tends to apply to department 1, 2 to 2
D <- rbern( N , ifelse( G==1 , 0.3 , 0.8 ) ) + 1
# matrix of acceptance rates [dept,gender]
accept_rate <- matrix( c(0.05,0.2,0.1,0.3) , nrow=2 )
# simulate acceptance
A <- rbern( N , accept_rate[D,G] )
```

$$\begin{tikzcd} & D \arrow[dr] & \\ G \arrow[ur] \arrow[rr] & & A \end{tikzcd}$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Estimand: Total effect of G

$$A_i \sim \text{Bernoulli}(p_i)$$

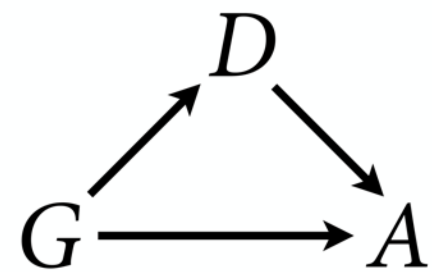
$$\text{logit}(p_i) = \alpha[G_i]$$

Genders

$$\alpha = [\alpha_1, \alpha_2]$$

$$\Pr(A_i = 1) = p_i$$

$$p_i = \frac{\exp(\alpha[G_i])}{1 + \exp(\alpha[G_i])}$$

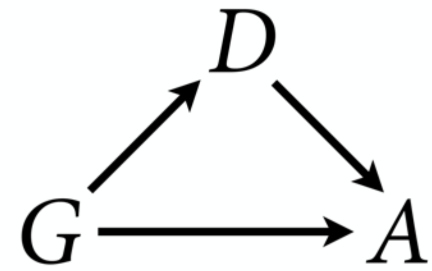


Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Estimand: Direct effect of G

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

$$\begin{array}{cc} & \textit{Departments} \\ \alpha = & \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} \\ \alpha_{2,1} & \alpha_{2,2} \end{bmatrix} \textit{Genders} \end{array}$$



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i]$$
$$\alpha_j \sim \text{Normal}(0,1)$$

Direct effect(s)

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$
$$\alpha_{j,k} \sim \text{Normal}(0,1)$$

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i]$$
$$\alpha_j \sim \text{Normal}(0,1)$$

```
dat_sim <- list( A=A , D=D , G=G )

m1 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

Direct effect(s)

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$
$$\alpha_{j,k} \sim \text{Normal}(0,1)$$

```
dat_sim <- list( A=A , D=D , G=G )

m2 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

```
dat_sim <- list( A=A , D=D , G=G )

m1 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

```
precis(m1,depth=2)
```

```
      mean   sd  5.5% 94.5% n_eff Rhat4
a[1] -1.80 0.13 -2.01 -1.60  1549     1
a[2] -1.09 0.10 -1.25 -0.93  1159     1
```

Direct effect(s)

```
dat_sim <- list( A=A , D=D , G=G )

m2 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

```
dat_sim <- list( A=A , D=D , G=G )

m1 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

```
precis(m1,depth=2)
```

	mean	sd	5.5%	94.5%	n_eff	Rhat4
a[1]	-1.80	0.13	-2.01	-1.60	1549	1
a[2]	-1.09	0.10	-1.25	-0.93	1159	1

Direct effect(s)

```
dat_sim <- list( A=A , D=D , G=G )

m2 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

```
precis(m2,depth=3)
```

	mean	sd	5.5%	94.5%	n_eff	Rhat4
a[1,1]	-2.31	0.18	-2.60	-2.04	2529	1
a[1,2]	-0.92	0.19	-1.23	-0.62	2216	1
a[2,1]	-1.93	0.31	-2.45	-1.44	2214	1
a[2,2]	-0.93	0.11	-1.11	-0.75	2055	1

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

```
dat_sim <- list( A=A , D=D , G=G )

m1 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

```
precis(m1,depth=2)
```

```
      mean   sd  5.5% 94.5% n_eff Rhat4
a[1] -1.80 0.13 -2.01 -1.60  1549     1
a[2] -1.09 0.10 -1.25 -0.93  1159     1
```

Direct effect(s)

```
dat_sim <- list( A=A , D=D , G=G )

m2 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

```
precis(m2,depth=3)
```

```
        mean   sd  5.5% 94.5% n_eff Rhat4
a[1,1] -2.31 0.18 -2.60 -2.04  2529     1
a[1,2] -0.92 0.19 -1.23 -0.62  2216     1
a[2,1] -1.93 0.31 -2.45 -1.44  2214     1
a[2,2] -0.93 0.11 -1.11 -0.75  2055     1
```

to apply this in probability space, we apply the inv_logit func

```
> inv_logit(coef(m2))
    a[1,1]     a[1,2]     a[2,1]     a[2,2]
0.06296434 0.21109945 0.08253890 0.20003819
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Admissions: Drawing the Owl

(1) Estimand(s)

(2) Scientific model(s)

(3) Statistical model(s)

(4) Analyze

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

UC Berkeley Admissions

4526 graduate school applications for 1973 UC Berkeley

Stratified by department and gender of applicant

Evidence of gender discrimination?

	dept	admit	reject	applications	gender
1	A	512	313	825	male
2	A	89	19	108	female
3	B	353	207	560	male
4	B	17	8	25	female
5	C	120	205	325	male
6	C	202	391	593	female
7	D	138	279	417	male
8	D	131	244	375	female
9	E	53	138	191	male
10	E	94	299	393	female
11	F	22	351	373	male
12	F	24	317	341	female

See ?UCBAdmissions for citation

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Logistic & Binomial Regression

Logistic regression:
Binary [0,1] outcome and logit link

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

Binomial regression:
Count [0,N] outcome and logit link

$$A_i \sim \text{Binomial}(N_i, p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

Completely equivalent for inference

what is completely equivalent?
binomial regression: model on the count of 1's and 0's: disregarding the individuals

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

LOOOOOOOONG

	A	G	D
1	0	2	1
2	0	1	1
3	1	2	2
4	1	2	2
5	0	2	2
6	0	1	1
7	0	2	2
8	0	2	2
9	0	2	2
10	0	2	2
11	0	2	2
12	1	2	2
13	0	2	2
14	0	1	1
15	0	2	2
16	0	1	2
17	0	1	1
18	0	1	1
19	0	1	1
20	0	1	1

```r
dat_sim2 <- aggregate( A ~ G + D , dat_sim , sum )
dat_sim2$N <- aggregate( A ~ G + D , dat_sim , length )$A
```

the count approach from previous slide

	G	D	A	N
1	1	1	30	355
2	2	1	10	92
3	1	2	38	135
4	2	2	117	418

Aggregated

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Logistic & Binomial Regression

```
m2 <- ulam(
    alist(
        A ~ bernoulli(p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

$$A_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

```
m2_bin <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim2 , chains=4 , cores=4 )
```

$$A_i \sim \text{Binomial}(N_i, p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

Completely equivalent for inference

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Logistic & Binomial Regression

```
m2 <- ulam(
    alist(
        A ~ binomial(1,p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim , chains=4 , cores=4 )
```

$$A_i \sim \text{Binomial}(1, p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

```
m2_bin <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat_sim2 , chains=4 , cores=4 )
```

$$A_i \sim \text{Binomial}(N_i, p_i)$$
$$\text{logit}(p_i) = \alpha[G_i, D_i]$$

Completely equivalent for inference

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

```
data(UCBadmit)
d <- UCBadmit

dat <- list(
    A = d$admit,
    N = d$applications,
    G = ifelse(d$applicant.gender=="female",1,2),
    D = as.integer(d$dept)
)

# total effect gender
mG <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat , chains=4 , cores=4 )
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

```r
data(UCBadmit)
d <- UCBadmit

dat <- list(
    A = d$admit,
    N = d$applications,
    G = ifelse(d$applicant.gender=="female",1,2),
    D = as.integer(d$dept)
)

# total effect gender
mG <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat , chains=4 , cores=4 )
```

Direct effect(s)

```r
# direct effects
mGD <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat , chains=4 , cores=4 )
```

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

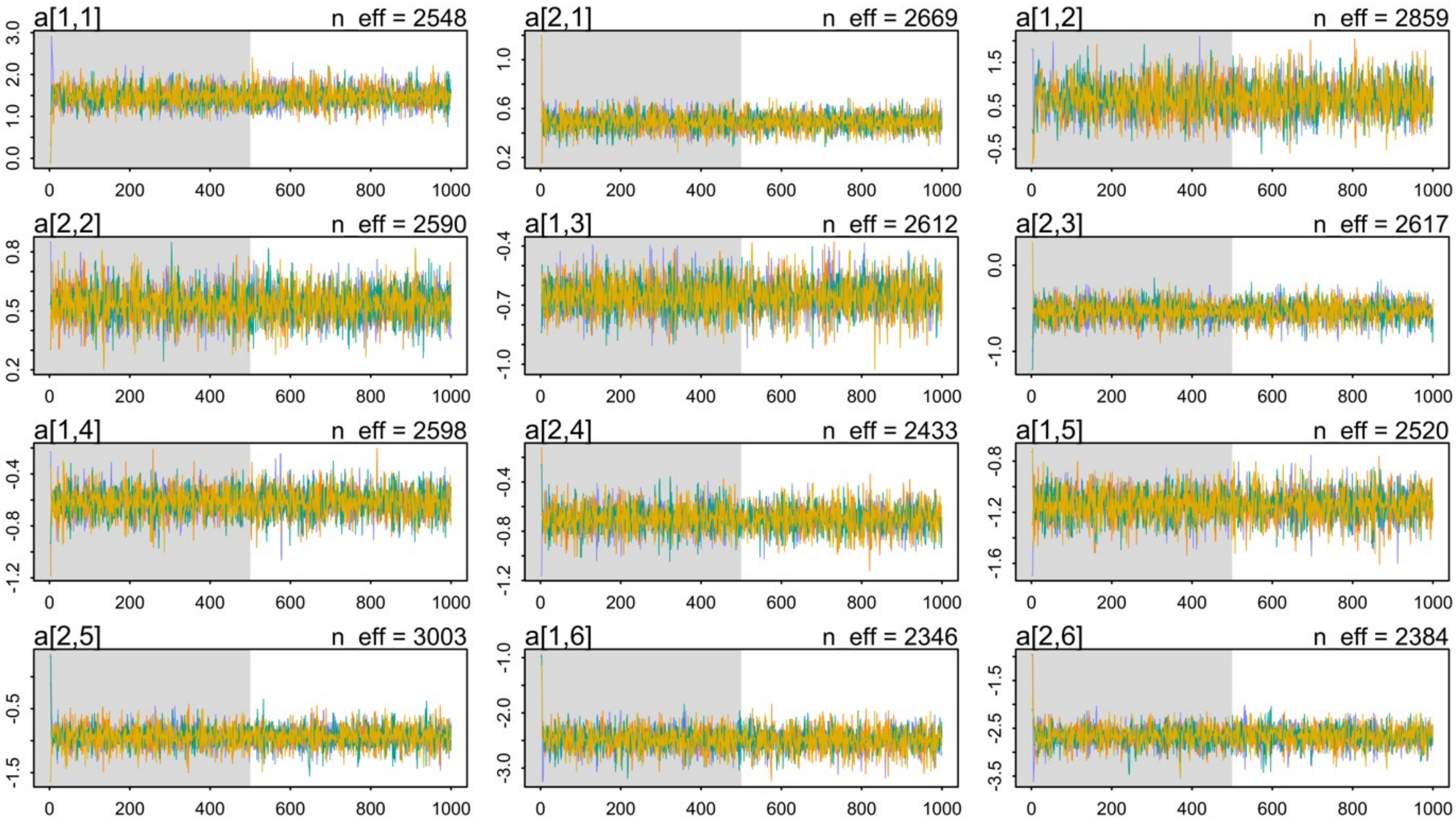
```
# total effect gender
mG <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G],
        a[G] ~ normal(0,1)
    ), data=dat , chains=4 , cores=4 )
```

```
# direct effects
mGD <- ulam(
    alist(
        A ~ binomial(N,p),
        logit(p) <- a[G,D],
        matrix[G,D]:a ~ normal(0,1)
    ), data=dat , chains=4 , cores=4 )
```

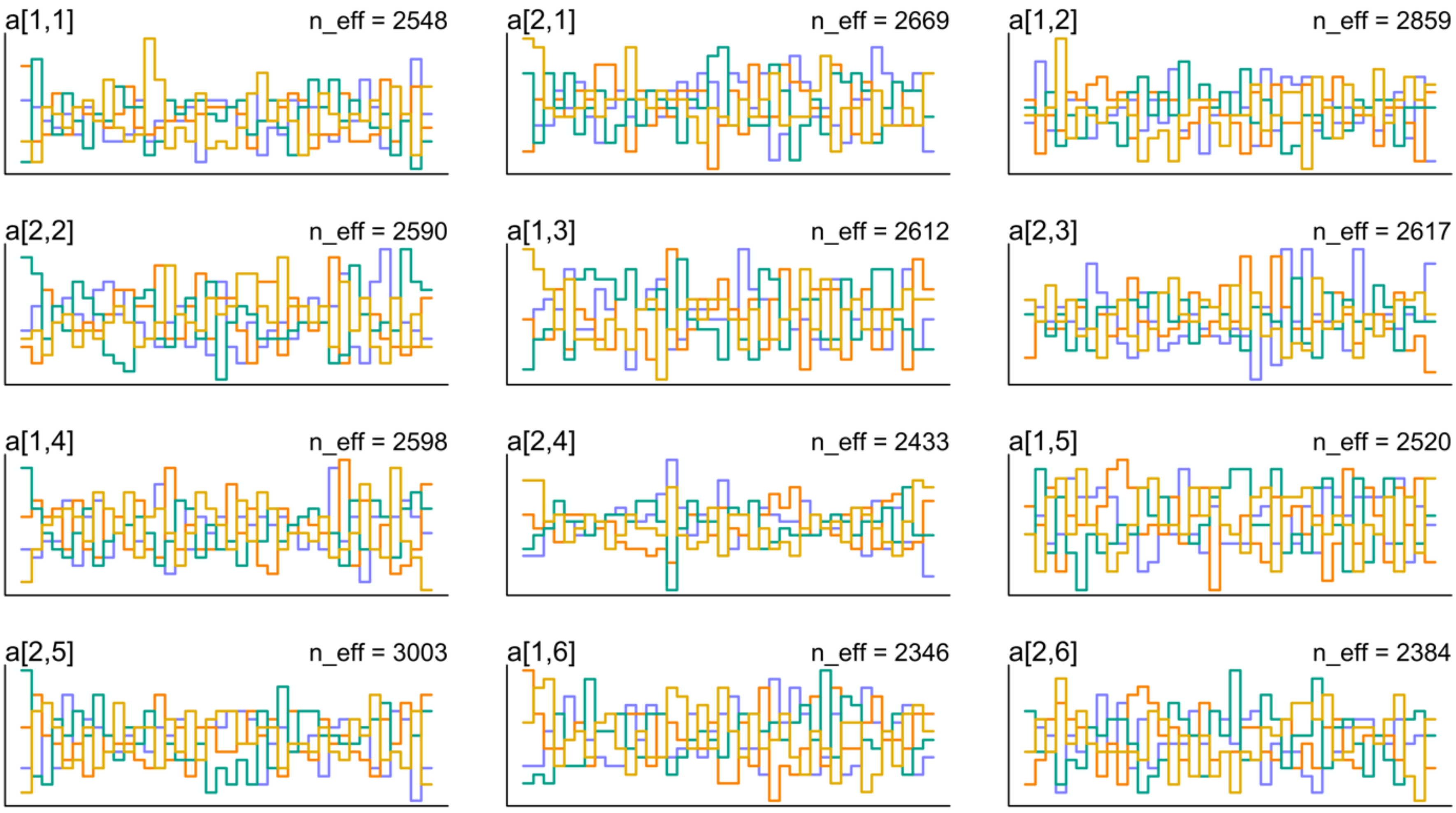
	mean	sd	5.5%	94.5%	n_eff	Rhat4
a[1]	-0.83	0.05	-0.91	-0.75	1487	1
a[2]	-0.22	0.04	-0.28	-0.16	1499	1

	mean	sd	5.5%	94.5%	n_eff	Rhat4
a[1,1]	1.48	0.24	1.11	1.87	2548	1
a[1,2]	0.66	0.40	0.03	1.32	2859	1
a[1,3]	-0.65	0.08	-0.79	-0.52	2612	1
a[1,4]	-0.62	0.11	-0.79	-0.45	2598	1
a[1,5]	-1.15	0.12	-1.34	-0.96	2520	1
a[1,6]	-2.50	0.20	-2.81	-2.18	2346	1
a[2,1]	0.49	0.07	0.38	0.60	2669	1
a[2,2]	0.53	0.08	0.40	0.67	2590	1
a[2,3]	-0.53	0.11	-0.72	-0.35	2617	1
a[2,4]	-0.70	0.10	-0.87	-0.54	2433	1
a[2,5]	-0.94	0.16	-1.20	-0.69	3003	1
a[2,6]	-2.67	0.21	-3.00	-2.34	2384	1

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022



Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022

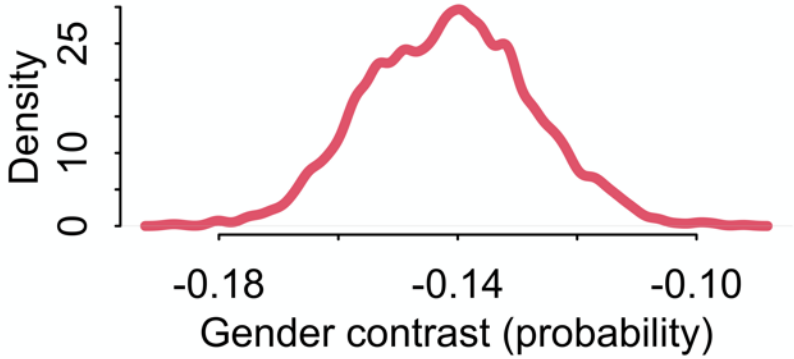


Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Total effect

```
post1 <- extract.samples(mG)
PrA_G1 <- inv_logit( post1$a[,1] )
PrA_G2 <- inv_logit( post1$a[,2] )
diff_prob <- PrA_G1 - PrA_G2
dens(diff_prob,lwd=4,col=2,xlab="Gender
contrast (probability)")
```

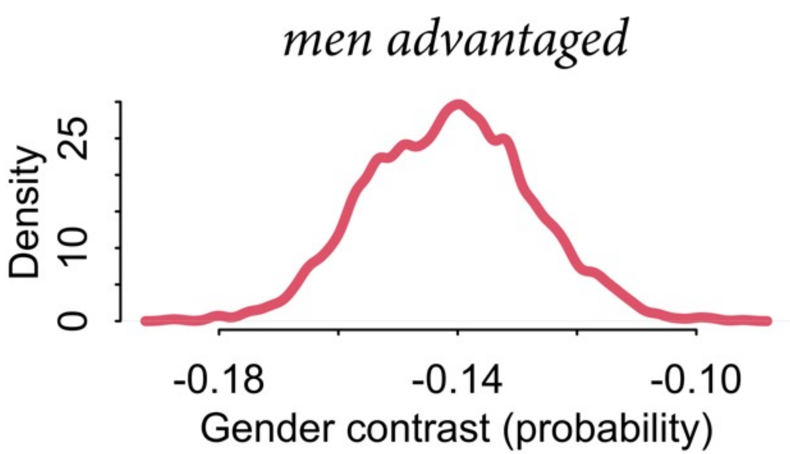
men advantaged



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

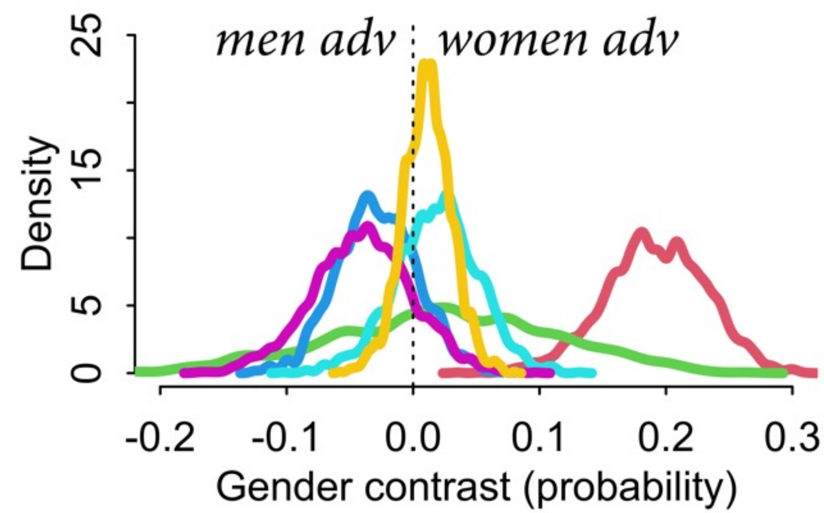
Total effect

```
post1 <- extract.samples(mG)
PrA_G1 <- inv_logit( post1$a[,1] )
PrA_G2 <- inv_logit( post1$a[,2] )
diff_prob <- PrA_G1 - PrA_G2
dens(diff_prob,lwd=4,col=2,xlab="Gender
contrast (probability)")
```



Direct effect(s)

```
post2 <- extract.samples(mGD)
PrA <- inv_logit( post2$a )
diff_prob_D_ <- sapply( 1:6 , function(i) PrA[,
1,i] - PrA[,2,i] )

plot(NULL,xlim=c(-0.2,0.3),ylim=c(0,25),xlab="G
ender contrast (probability)",ylab="Density")
for ( i in 1:6 ) dens( diff_prob_D_[,i] , lwd=4
, col=1+i , add=TRUE )
```

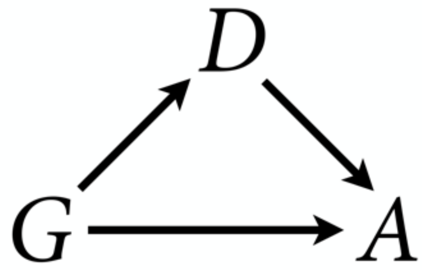


Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022

What is the **average direct effect** of gender across departments?

Depends upon distribution of applications, probability woman/man applies to each department

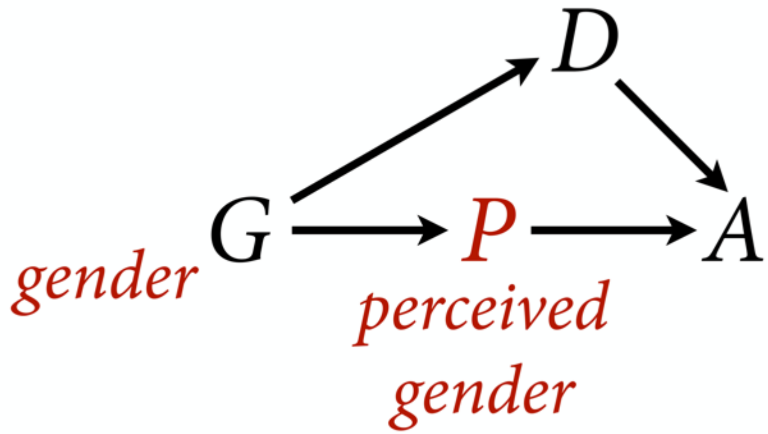
What is the invention actually?



What is the average **direct** effect of gender across departments?

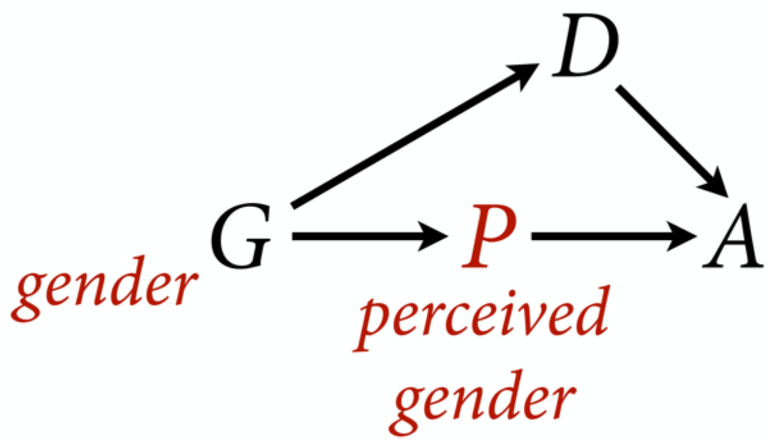
Depends upon distribution of applications, probability woman/man applies to each department

What is the invention actually?



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

What is the invention actually?

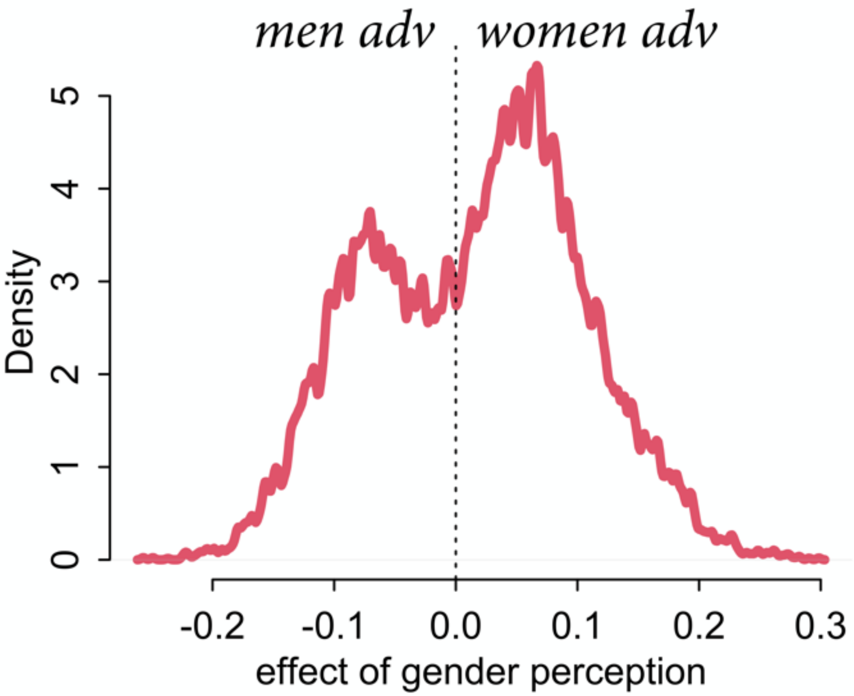


To calculate causal effect of *P*, must average (marginalize) over departments

Easy to do it as a simulation

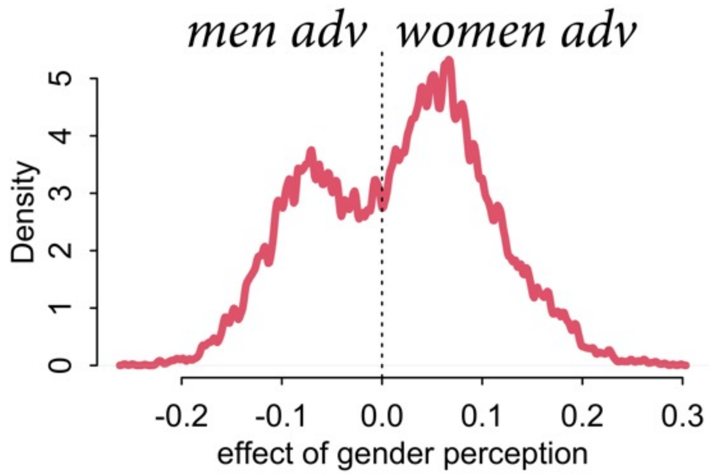
Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

```
# number of applicatons to simulate
total_apps <- sum(dat$N)

# number of applications per department
apps_per_dept <- sapply( 1:6 , function(i)
sum(dat$N[dat$D==i]) )

# simulate as if all apps from women
p_G1 <- link(m2,data=list(
    D=rep(1:6,times=apps_per_dept),
    N=rep(1,total_apps),
    G=rep(1,total_apps)))

# simulate as if all apps from men
p_G2 <- link(m2,data=list(
    D=rep(1:6,times=apps_per_dept),
    N=rep(1,total_apps),
    G=rep(2,total_apps)))

# summarize
dens( p_G1 - p_G2 , lwd=4 , col=2 ,
xlab="effect of gender perception" )
```

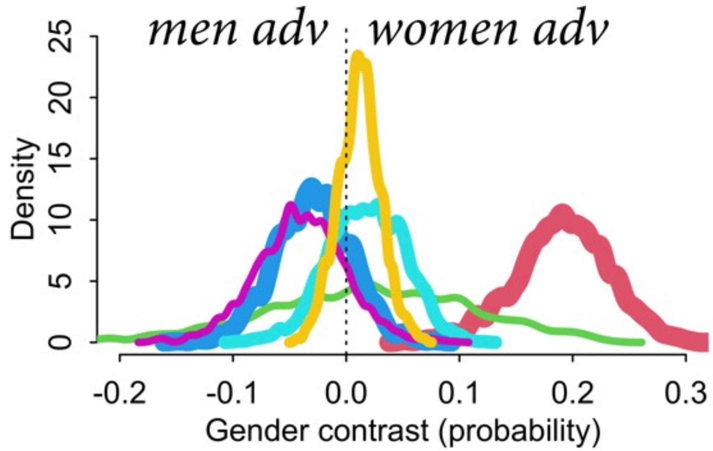


Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022

```
# simulate as if all apps from women
p_G1 <- link(m2,data=list(
    D=rep(1:6,times=apps_per_dept),
    N=rep(1,total_apps),
    G=rep(1,total_apps)))

# simulate as if all apps from men
p_G2 <- link(m2,data=list(
    D=rep(1:6,times=apps_per_dept),
    N=rep(1,total_apps),
    G=rep(2,total_apps)))
```




```
# show each dept density with weight as in
population
w <- xtabs( dat$N ~ dat$D ) / sum(dat$N)

plot(NULL,xlim=c(-0.2,0.3),ylim=c(0,25),xlab="
Gender contrast (probability)",ylab="Density")
for ( i in 1:6 ) dens( diff_prob_D_[,i] ,
lwd=2+20*w[i] , col=1+i , add=TRUE )
abline(v=0,lty=3)
```



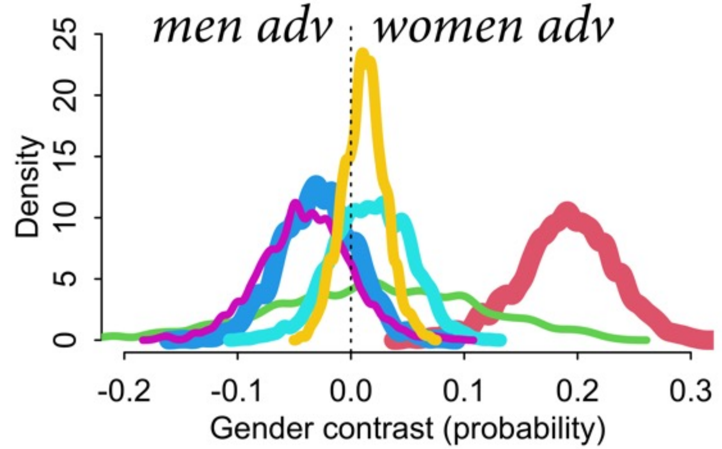
Source: Richard McElreath, https://github.com/rmcelreath/stat_rethinking_2022

Post-stratification

Description, prediction & causal inference often require post-stratification

Post-stratification: Re-weighting estimates for target population

At a different university, distribution of applications will differ => predicted consequence of intervention different



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

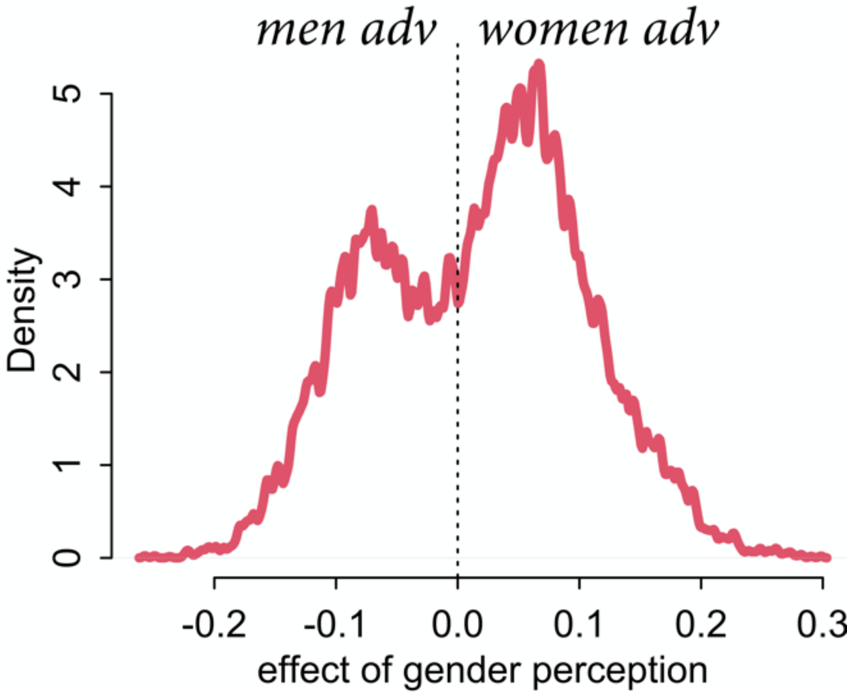
Admissions so far

Evidence for gender discrimination?

No overall evidence for, but:

(1) Distribution of applications can be a consequence of discrimination (data do not speak to this)

(2) Confounds are likely



Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022

Binomial Generalized Linear Models

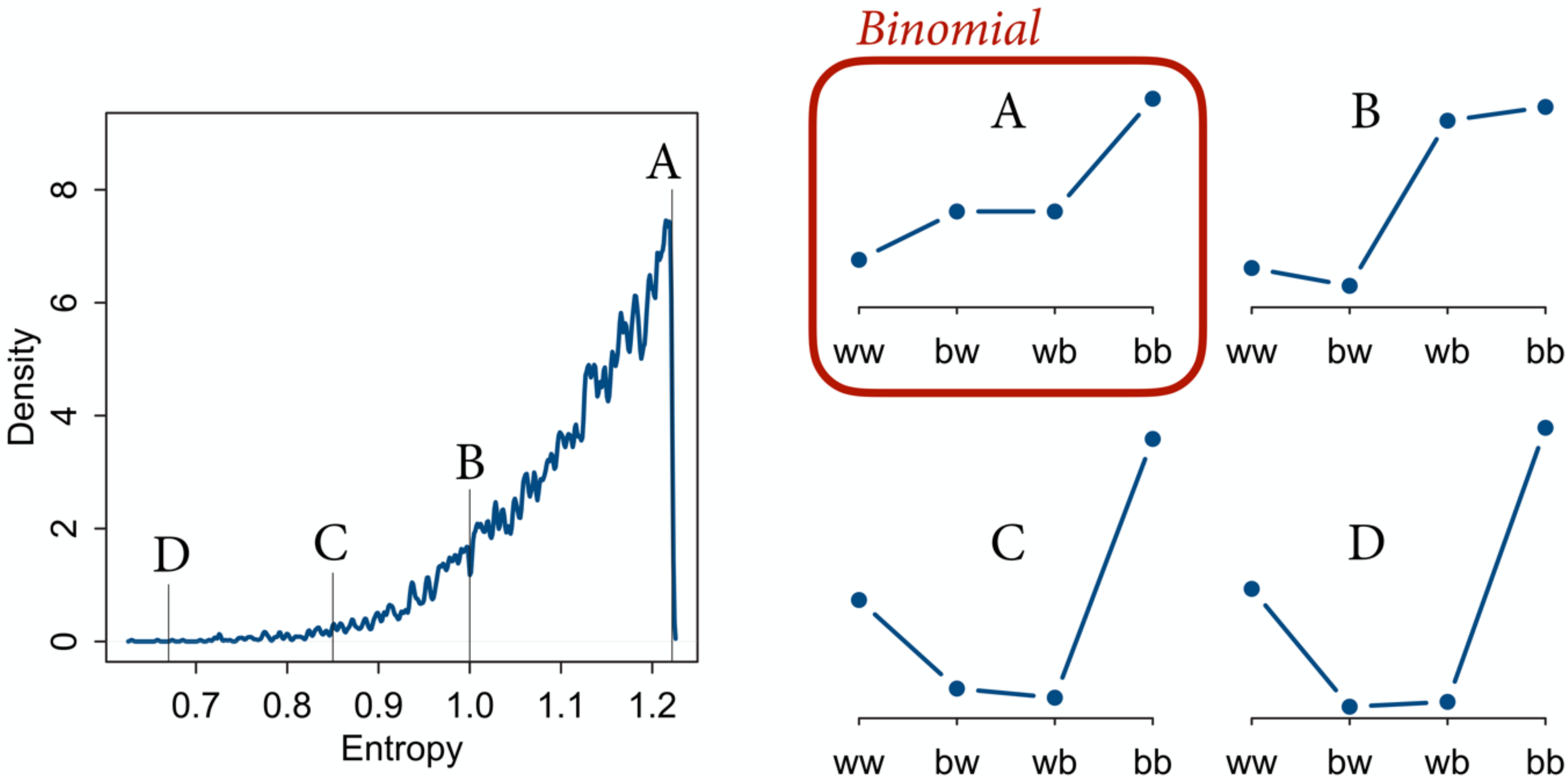
Outcome is a count with a maximum

Maximum needn't be known, sometimes target of inference (population size estimation)

logit link is not only option

Is maximum entropy distribution for count with constant expectation

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022



Page 310

Source: Richard McElraeth, https://github.com/rmcelreath/stat_rethinking_2022