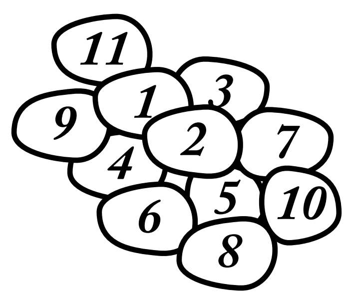
Statistical Rethinking
Winter 2019

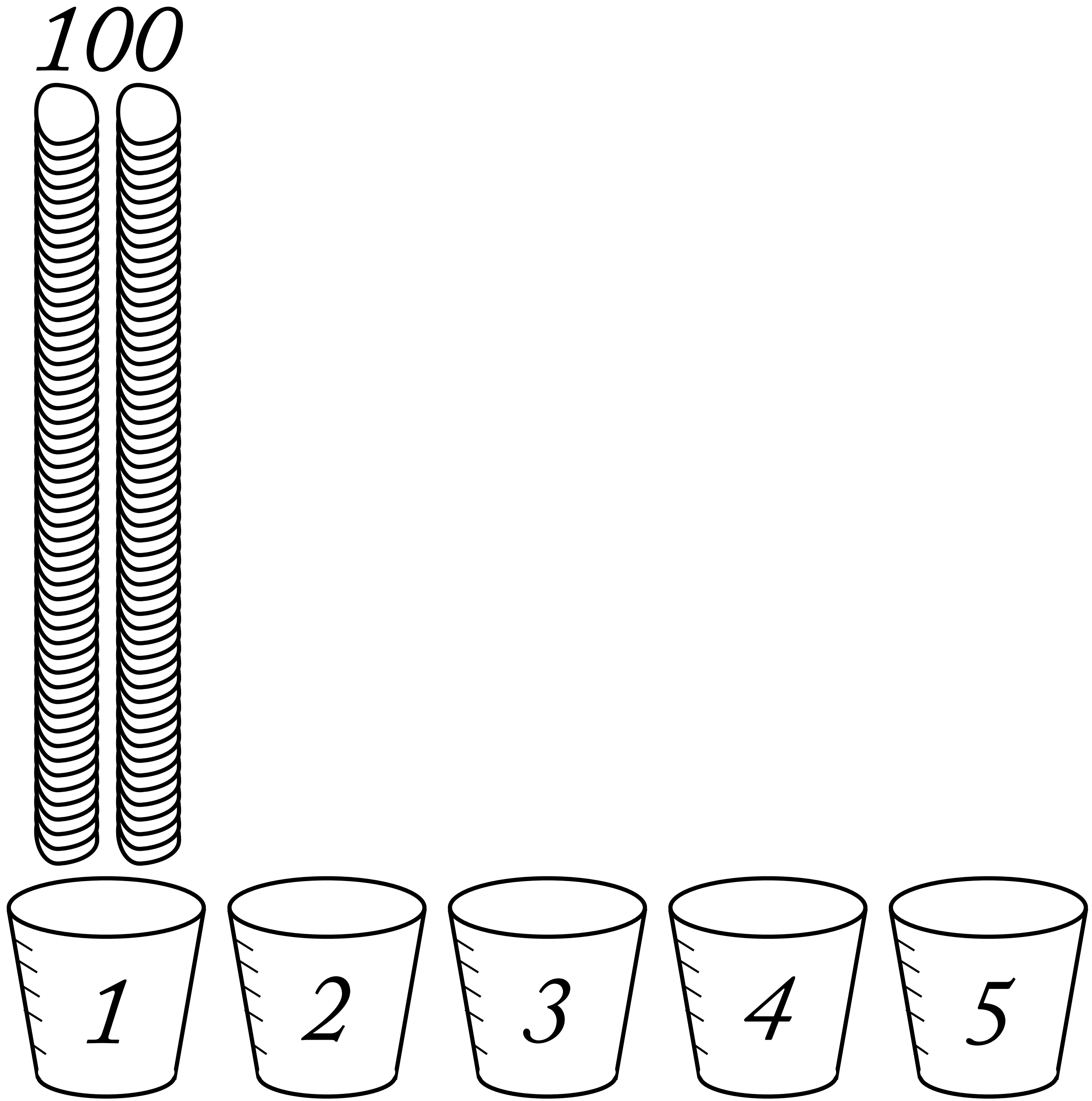
Lecture 11 / Week 6

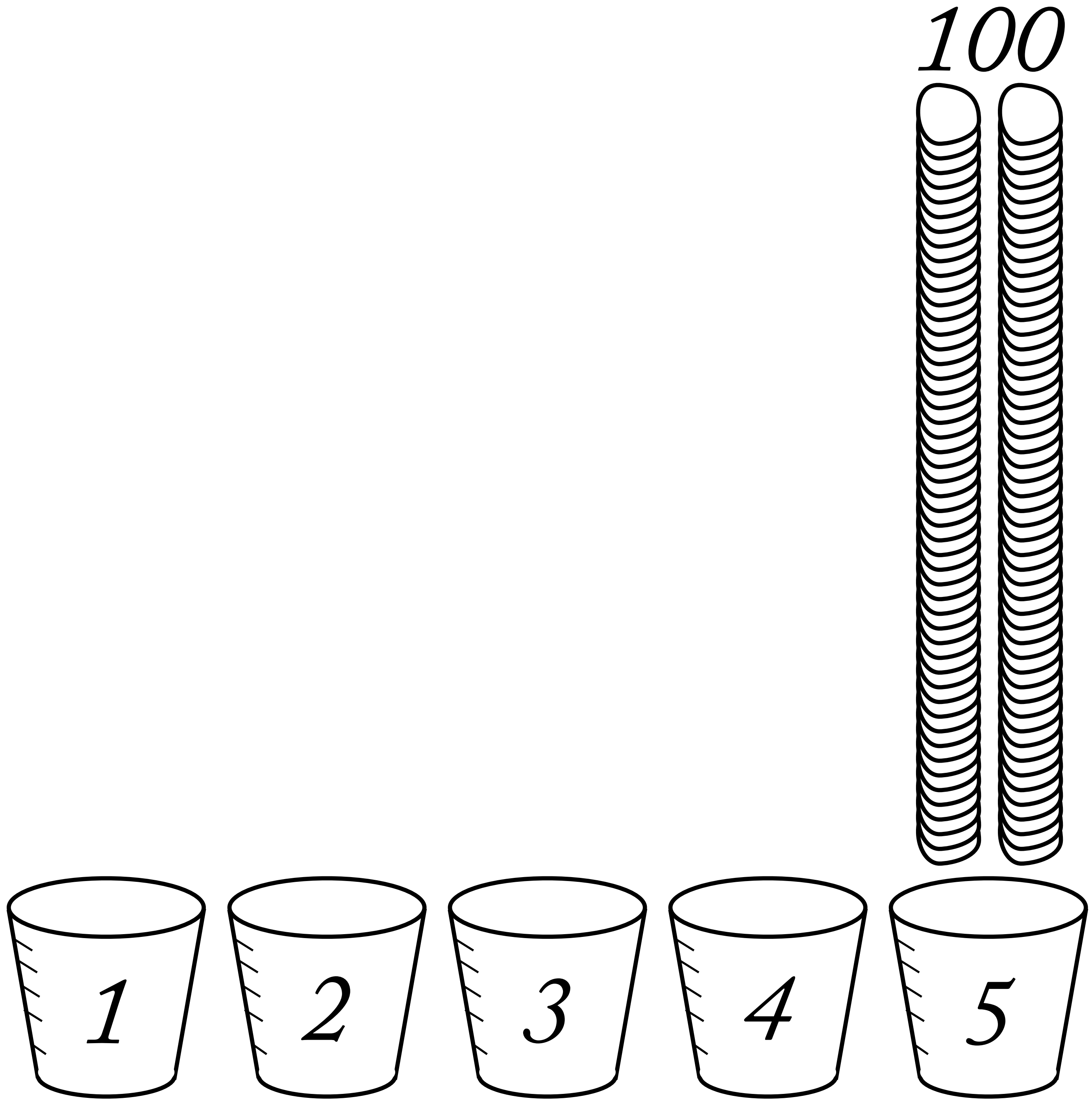
Big Entropy & The Generalized Linear Model

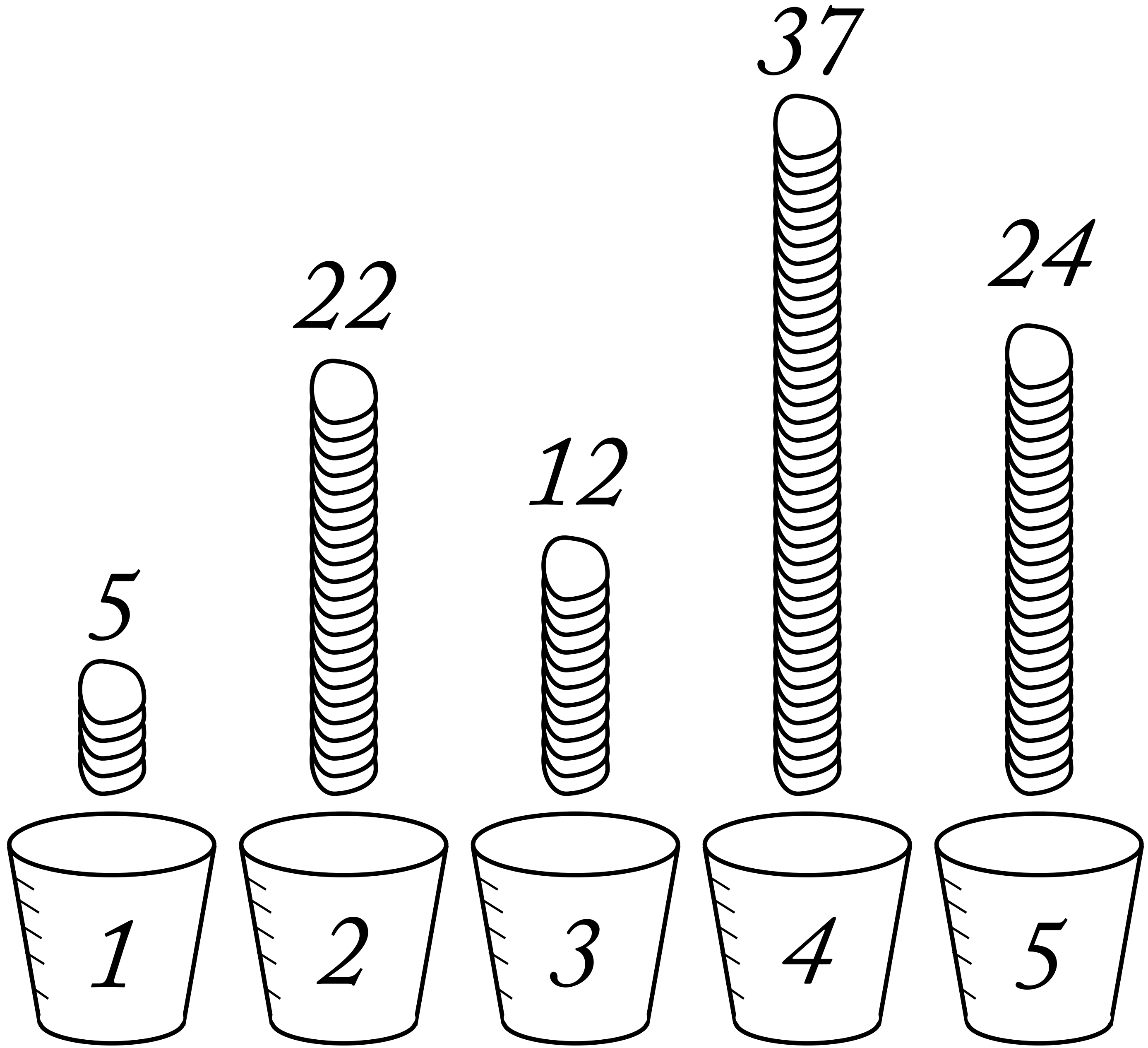


100 pebbles


1
2
3
4
5


100
1
2
3
4
5


100
1
2
3
4
5



5
22
12
37
24
1
2
3
4
5

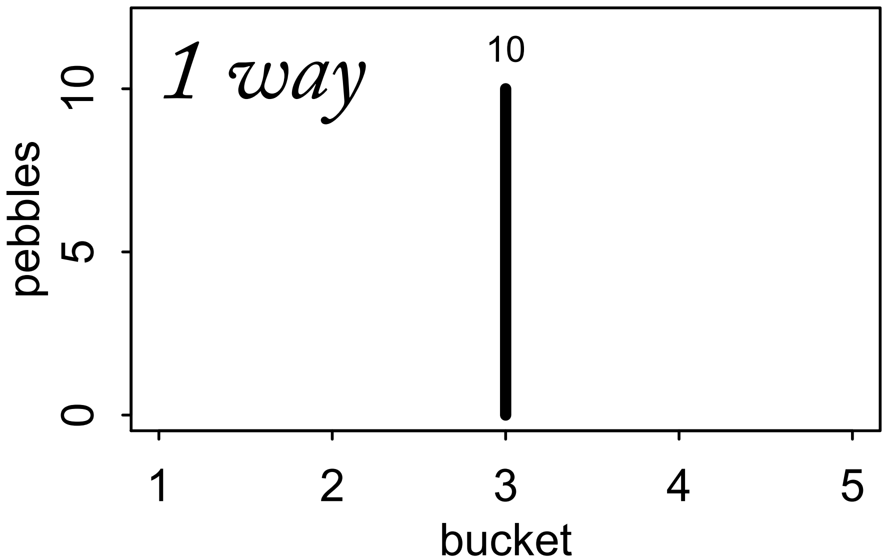
n_1 n_2 n_3 n_4 n_5

Number of ways:

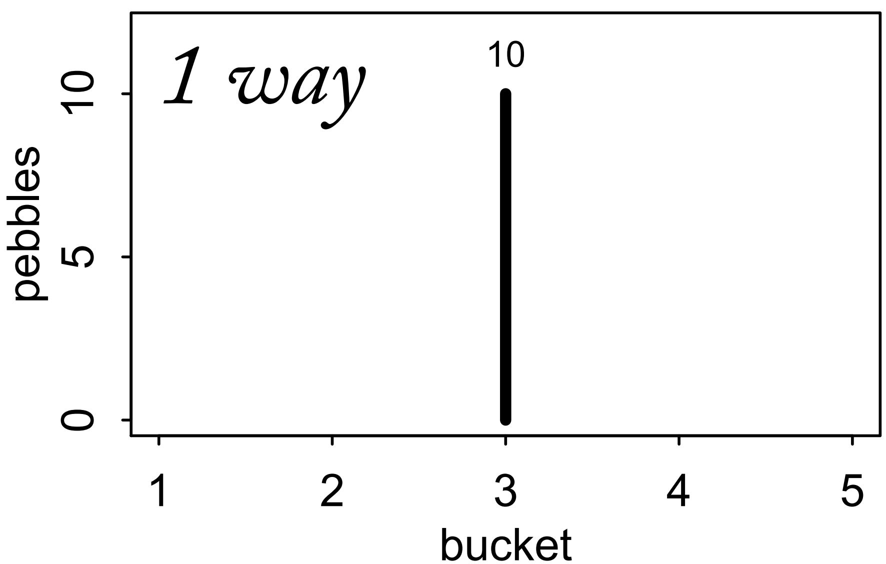
$$W = \frac{N!}{n_1!n_2!n_3!n_4!n_5!}$$

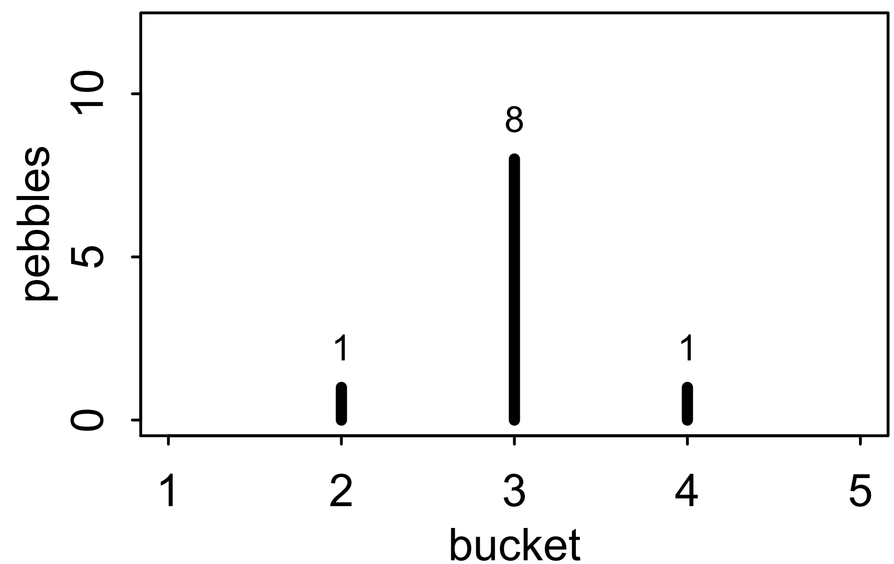
n_1 n_2 n_3 n_4 n_5

Suppose only 10 pebbles....

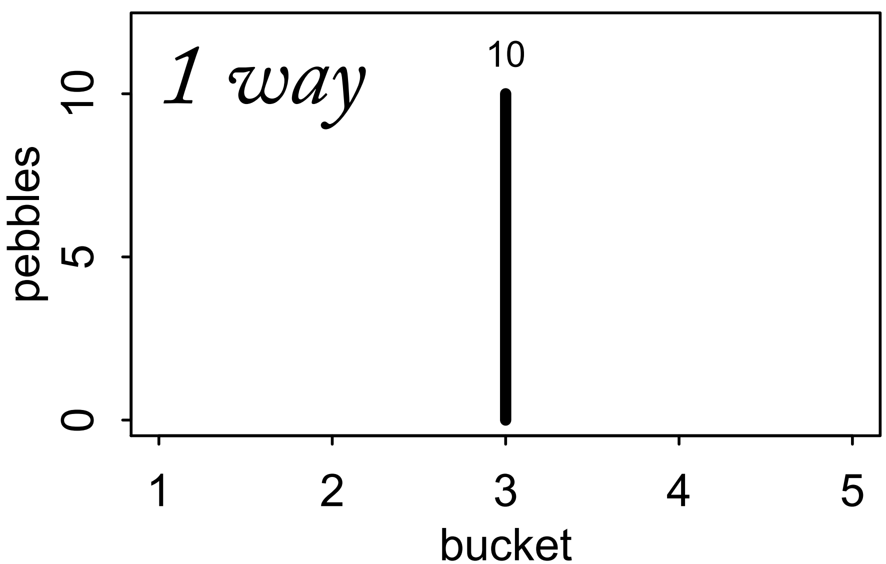


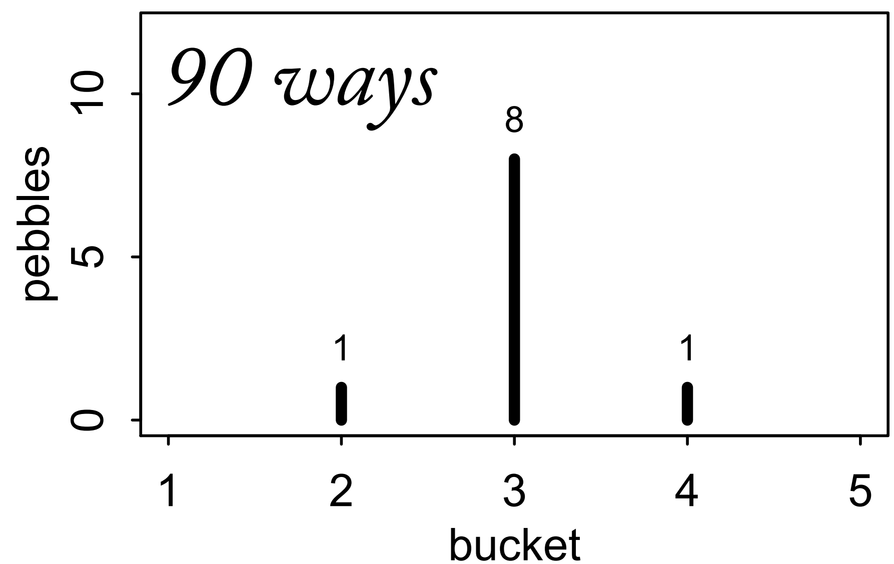
Suppose only 10 pebbles....



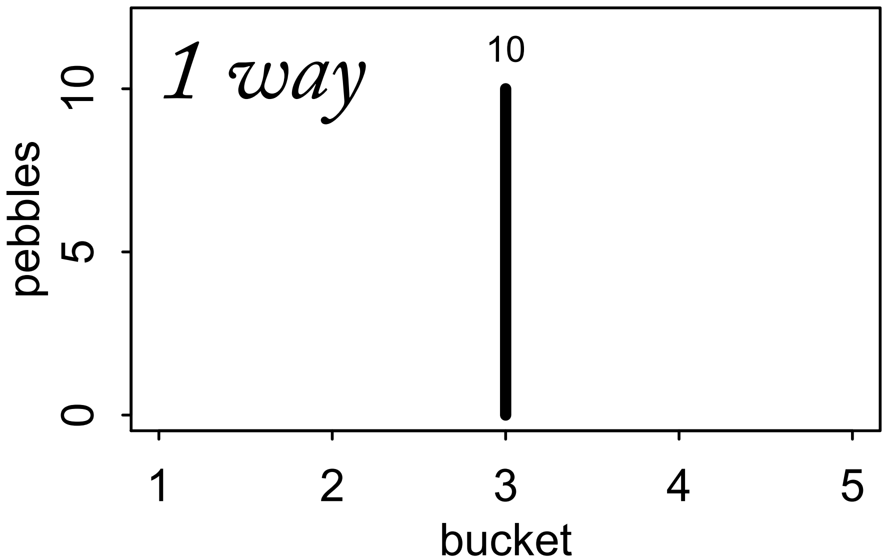


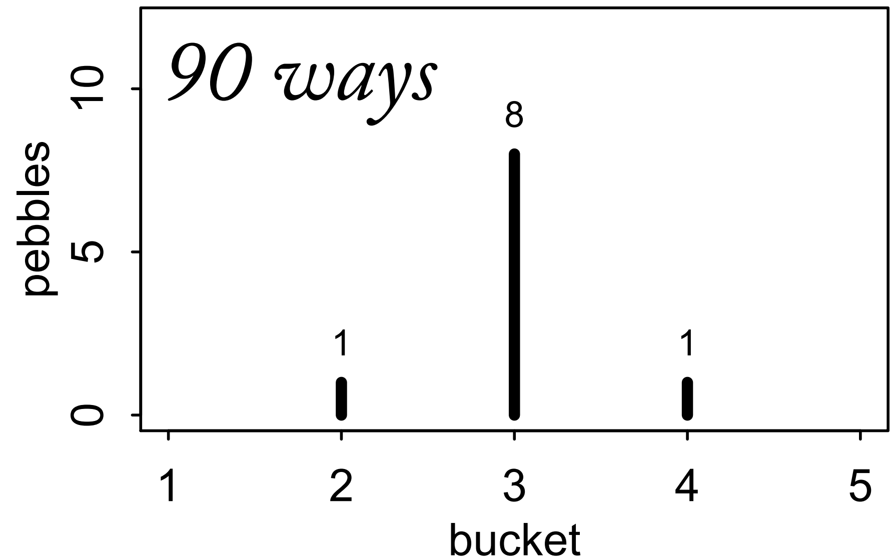
Suppose only 10 pebbles....

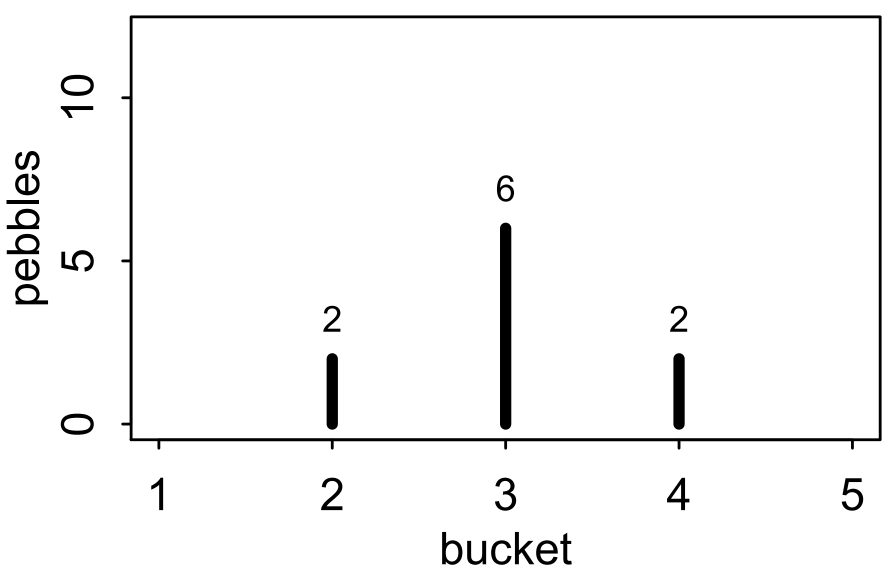




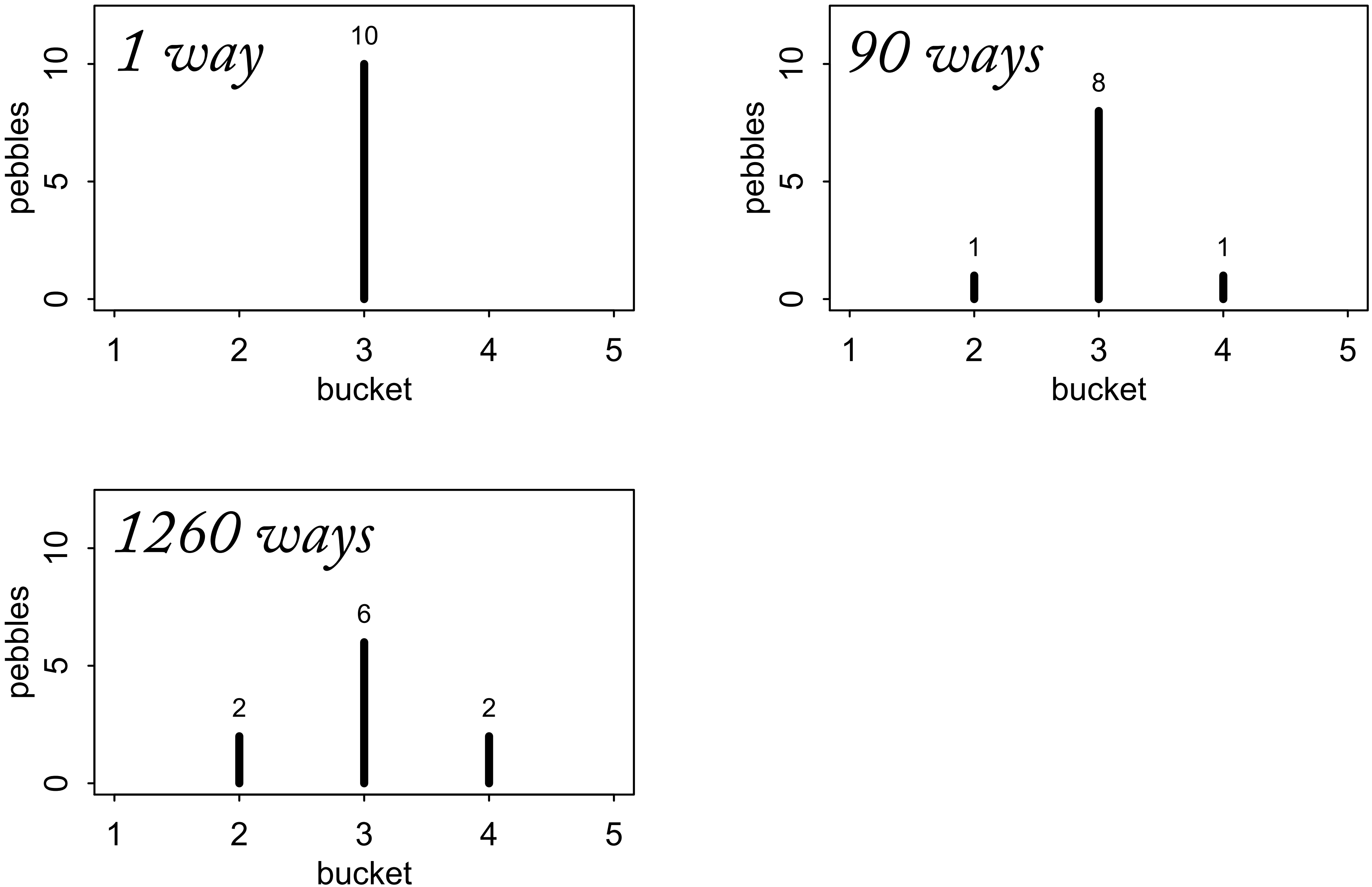
Suppose only 10 pebbles....



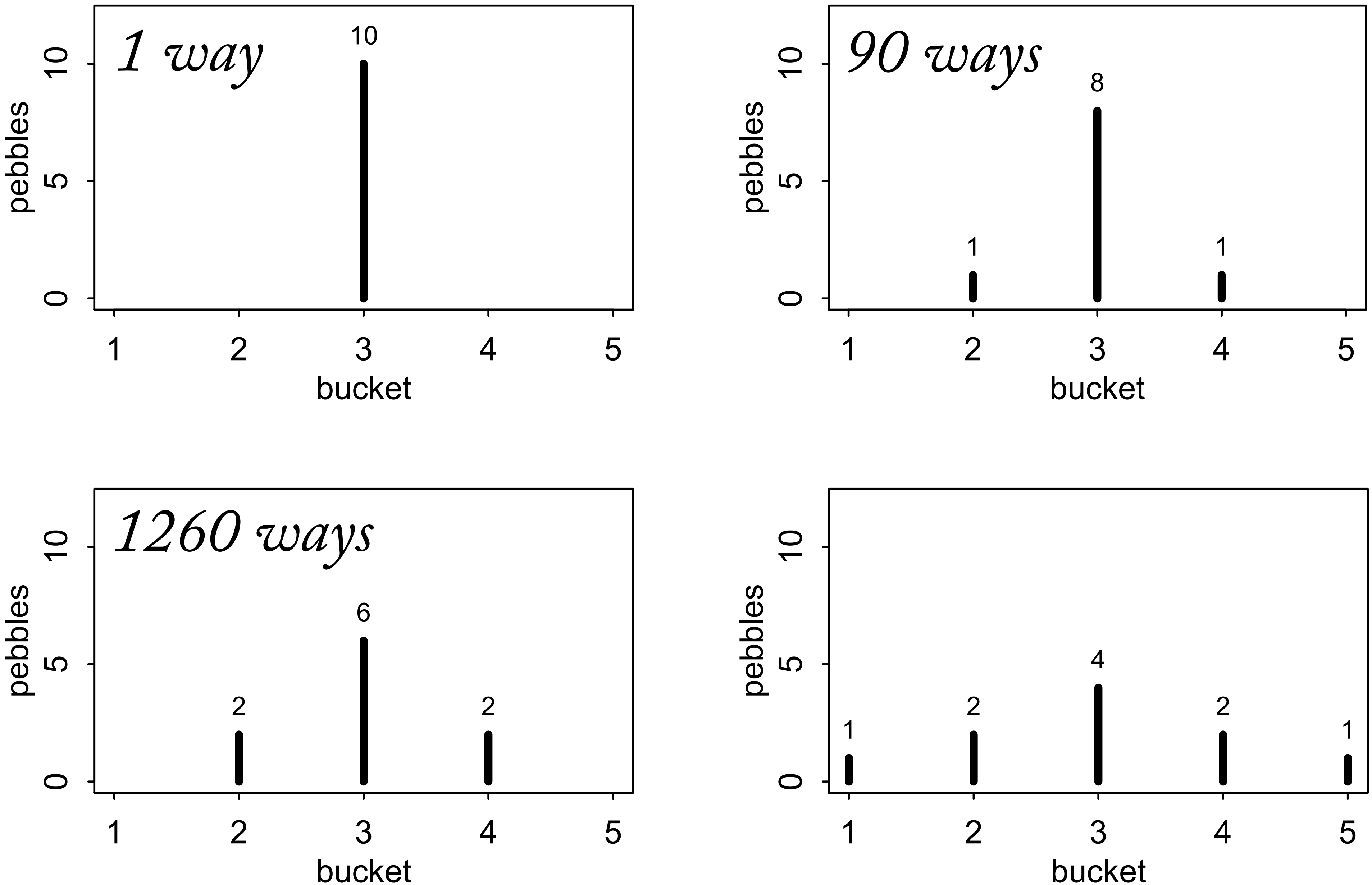




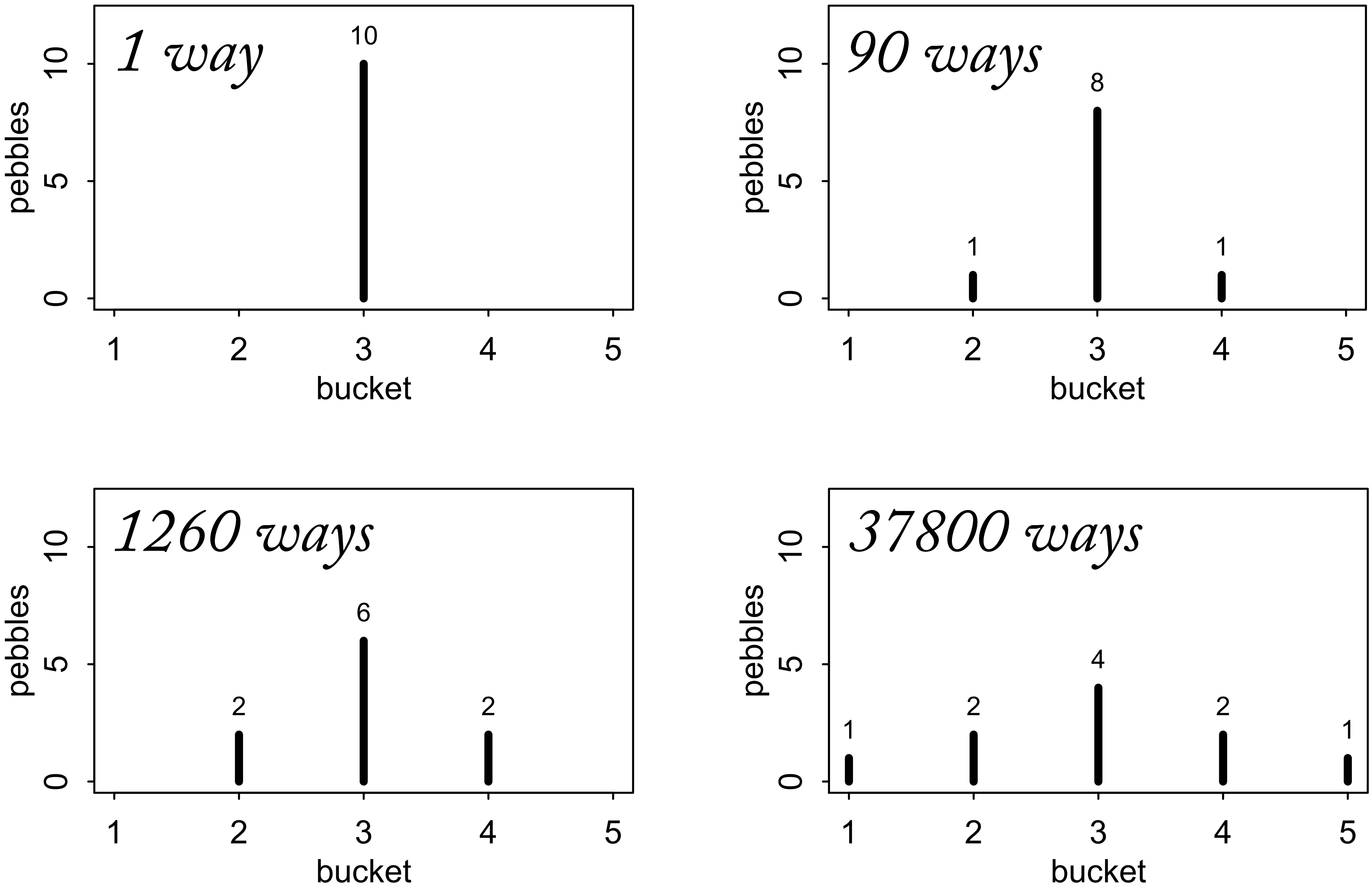
Suppose only 10 pebbles....

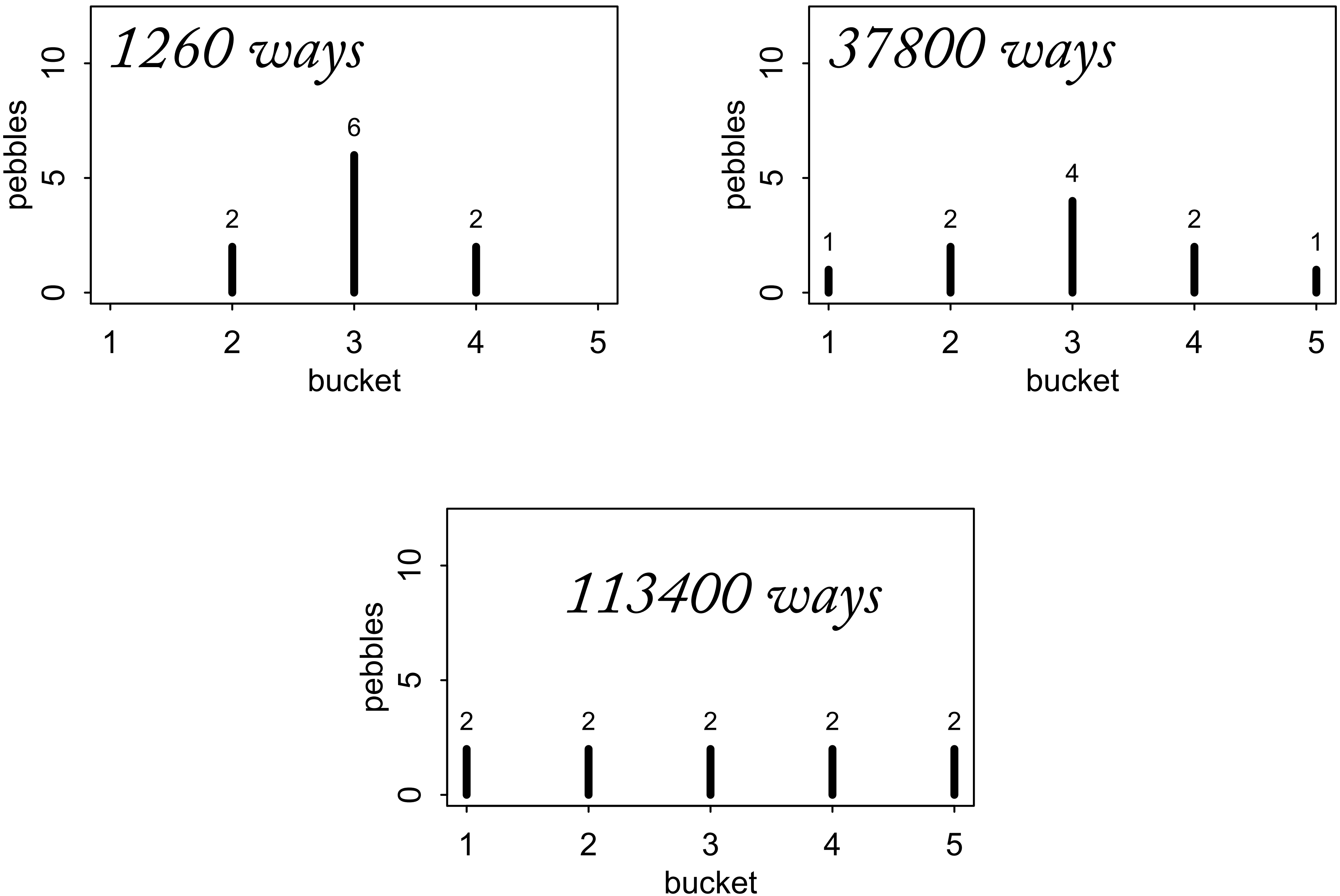


Suppose only 10 pebbles....



Suppose only 10 pebbles....





1260 ways
37800 ways
113400 ways
pebbles
bucket

For large N:

$$\frac{1}{N}\log W \approx -\sum_i \frac{n_i}{N}\log\left(\frac{n_i}{N}\right)$$

n_1 n_2 n_3 n_4 n_5

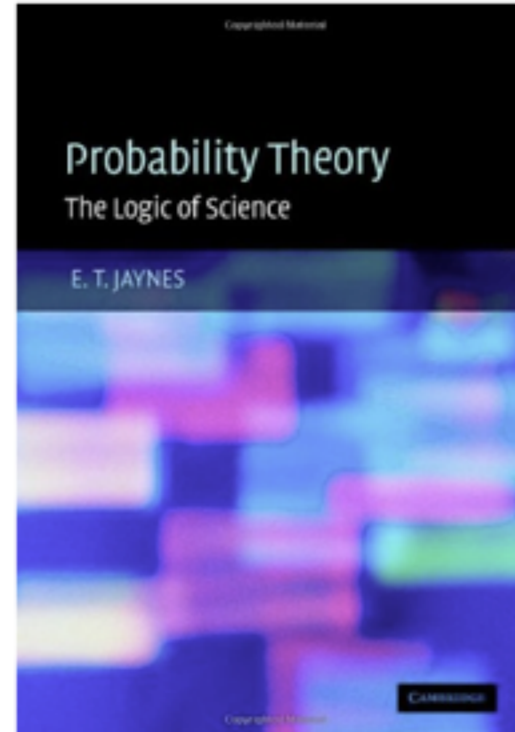
For large N:

$$\frac{1}{N}\log W \approx -\sum_i \frac{n_i}{N}\log\left(\frac{n_i}{N}\right) = -\sum_i p_i \log p_i$$

n_1 n_2 n_3 n_4 n_5

Maximum entropy

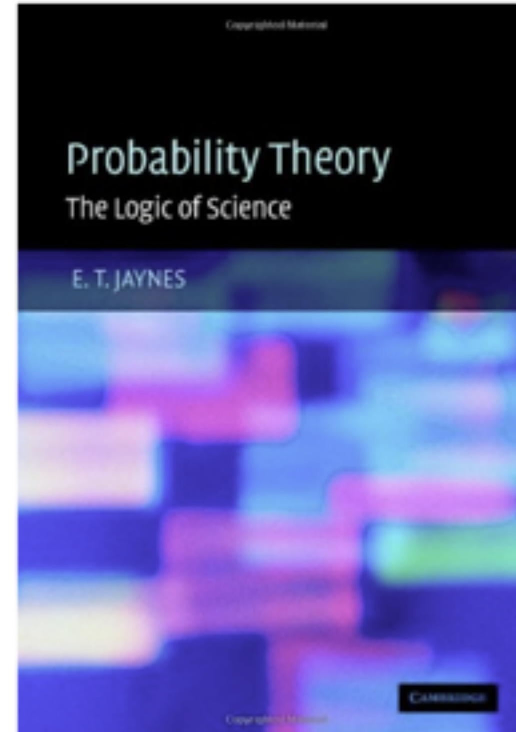
- Due to Edwin T. Jaynes (1922–1998)
- The maxent principle:
  - *Distribution with largest entropy is distribution most consistent with stated assumptions*
  - *Can happen the largest number of ways*



E. T. Jaynes (1922–1998)

Maximum entropy

- Due to Edwin T. Jaynes (1922–1998)
- The maxent principle:
  - *Distribution with largest entropy is distribution most consistent with stated assumptions*
  - *Can happen the largest number of ways*
- For parameters, provides way to understand priors
- For observations, way to understand likelihood
- Also reproduces Bayesian updating as special case (*minimum cross-entropy*)



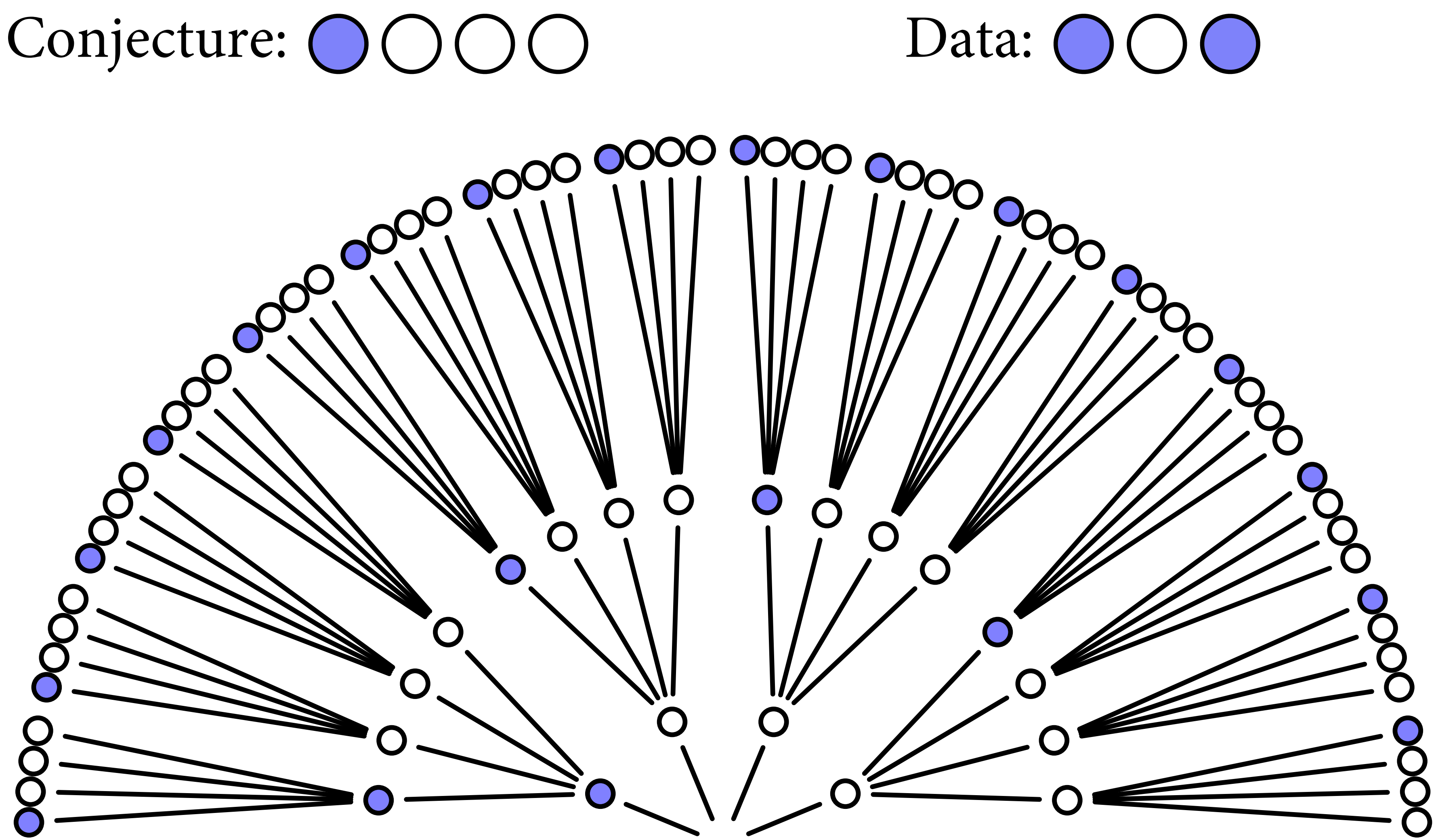
E. T. Jaynes (1922–1998)

Maximum entropy

- Ye olde information entropy:

$$H(p) = -\sum_i p_i \log p_i$$

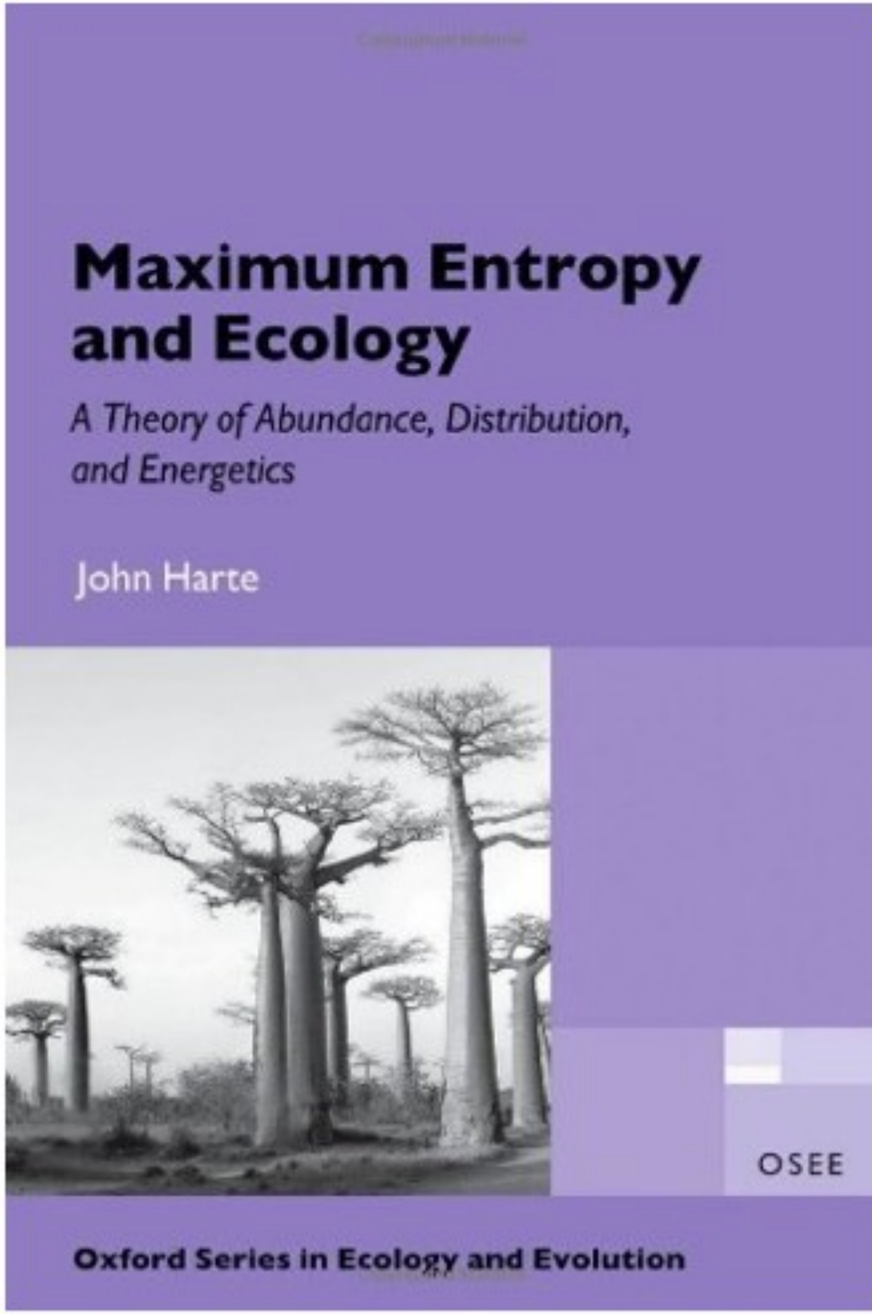
- Q: What kind of distribution maximizes this quantity?
- A: Flattest distribution still consistent with **constraints**. This is the distribution that can happen the most unique ways.
- Whatever does happen, bound to be one of those ways.



Counting paths produces flattest distribution consistent with stated constraints.

Maximum entropy

Constraints	Maxent distribution
Real value in interval	Uniform
Real value, finite variance	Gaussian
Binary events, fixed probability	Binomial
Non-negative real, has mean	Exponential



Generalized Linear Models

- Goal: Connect linear model to outcome variable
- Still geocentric!
- Strategy:
  1. Pick an outcome distribution
  2. Model its parameters using links to linear models
  3. Compute posterior
- Can model multivariate relationships and non-linear responses
- Building blocks of multilevel models

Generalized Linear Models

- How to pick a data distribution
    - Mostly *exponential family*
    - Arise from natural processes
    - M*aximum entropy* interpretations
    - Select from first principles
    - Resist ***histomancy***: Superstitious practice of picking likelihoods by gazing at a histogram

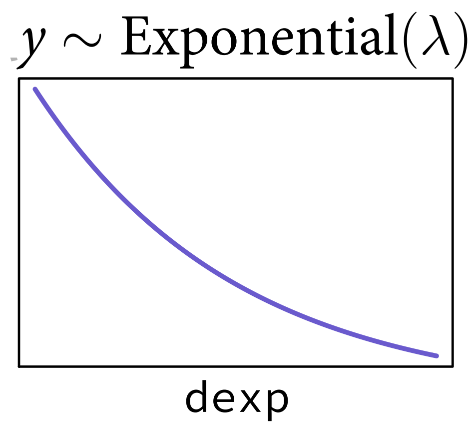


Figure 9.5

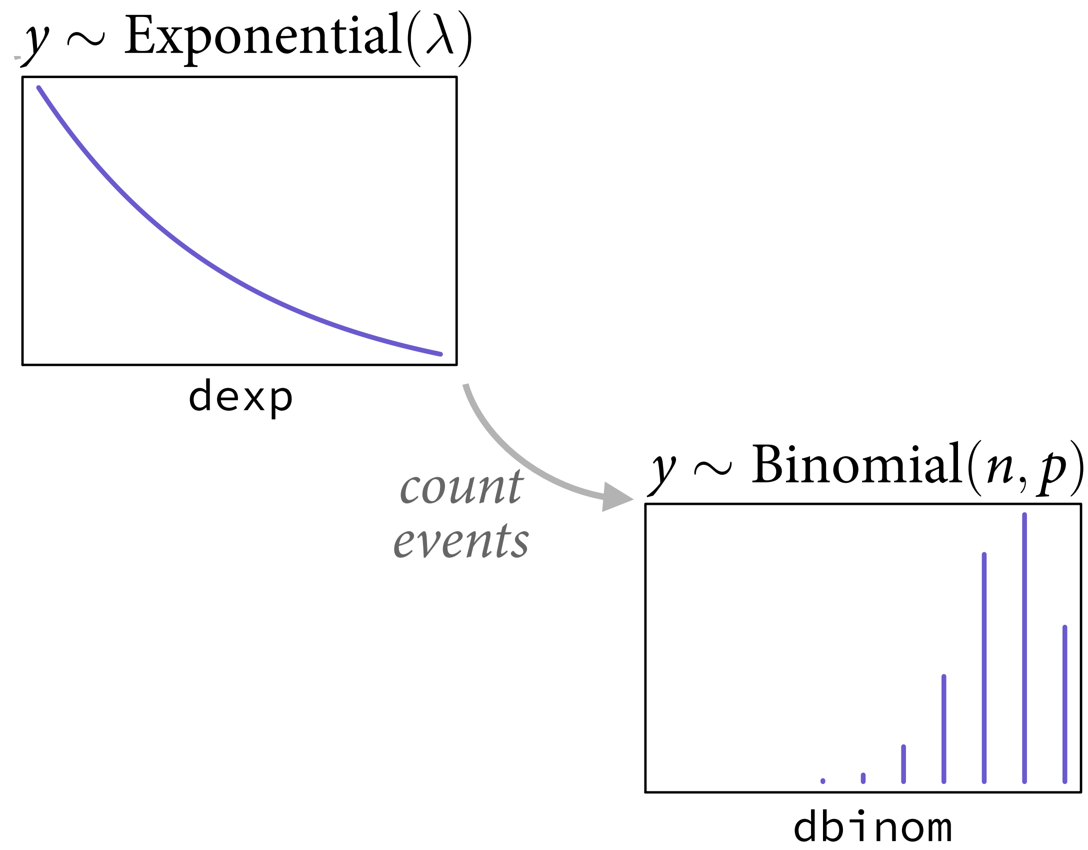

$y \sim \text{Exponential}(\lambda)$
dexp
count events
$y \sim \text{Binomial}(n, p)$
dbinom

Figure 9.5

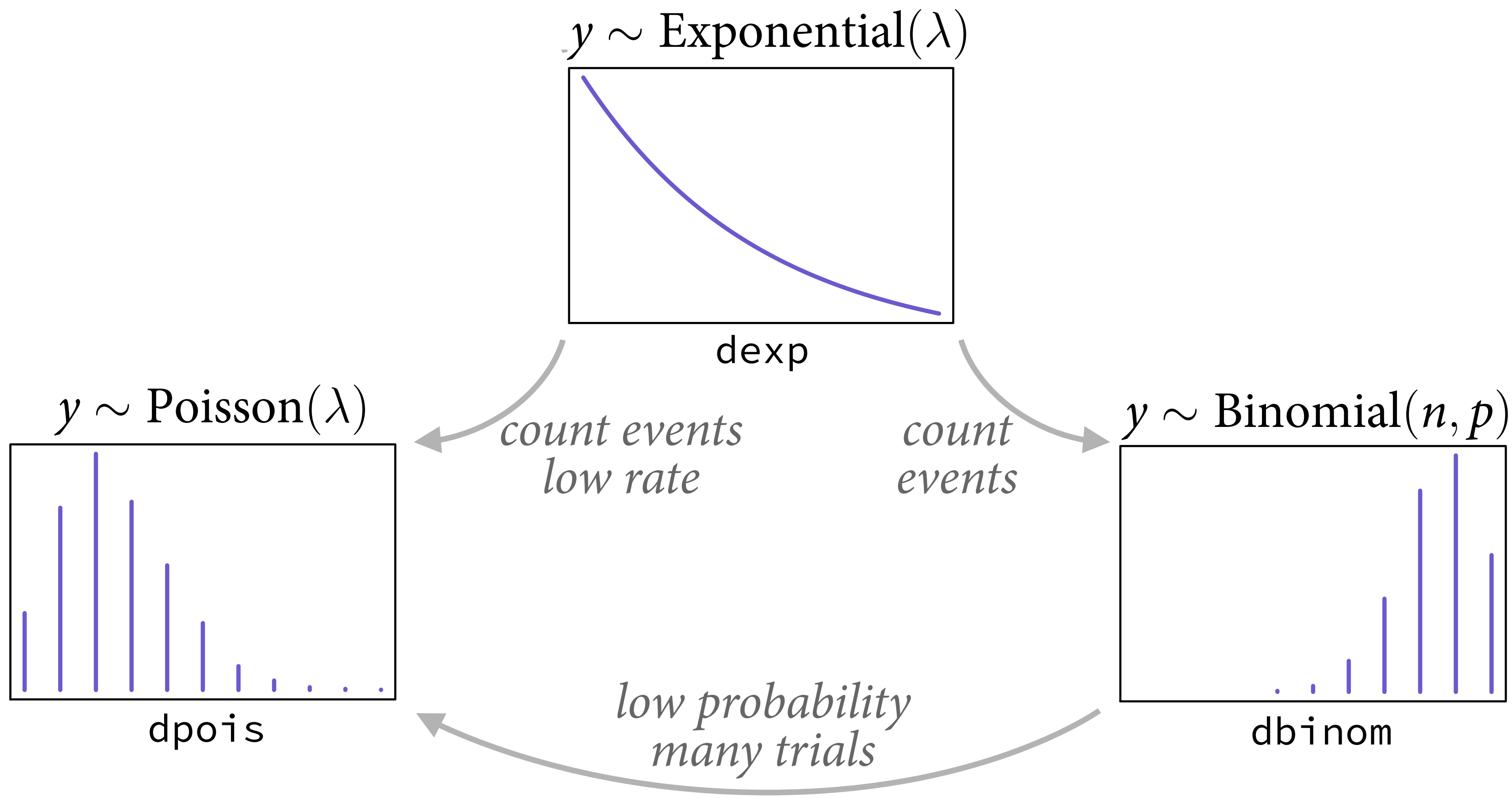
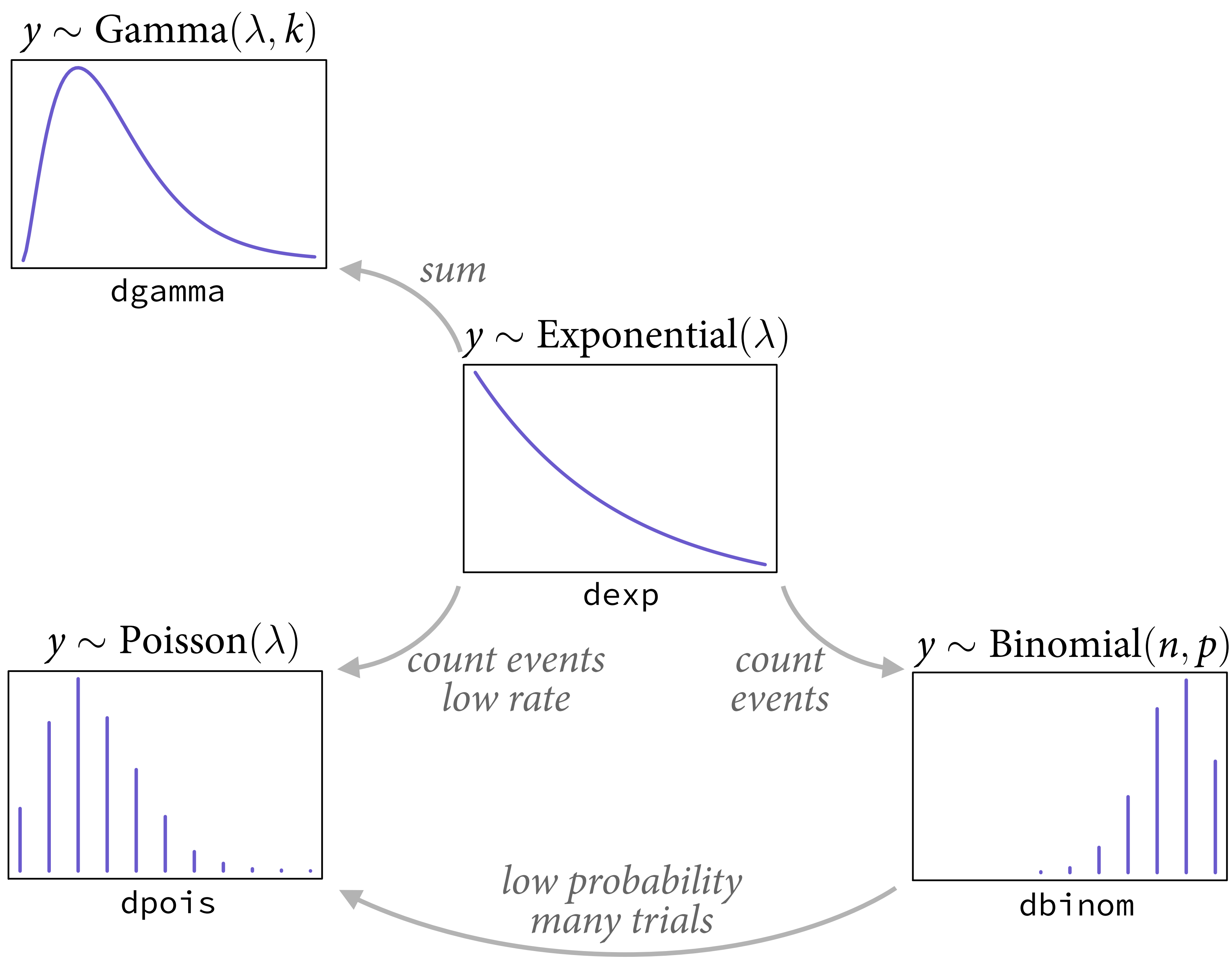

$y \sim \text{Exponential}(\lambda)$
dexp
$y \sim \text{Poisson}(\lambda)$
dpois
$y \sim \text{Binomial}(n, p)$
dbinom
count events
low rate
count
events
low probability
many trials

Figure 9.5



$y \sim \text{Gamma}(\lambda, k)$
dgamma
sum
$y \sim \text{Exponential}(\lambda)$
dexp
$y \sim \text{Poisson}(\lambda)$
count events
low rate
count
events
$y \sim \text{Binomial}(n, p)$
dpois
low probability
many trials
dbinom

Figure 9.5

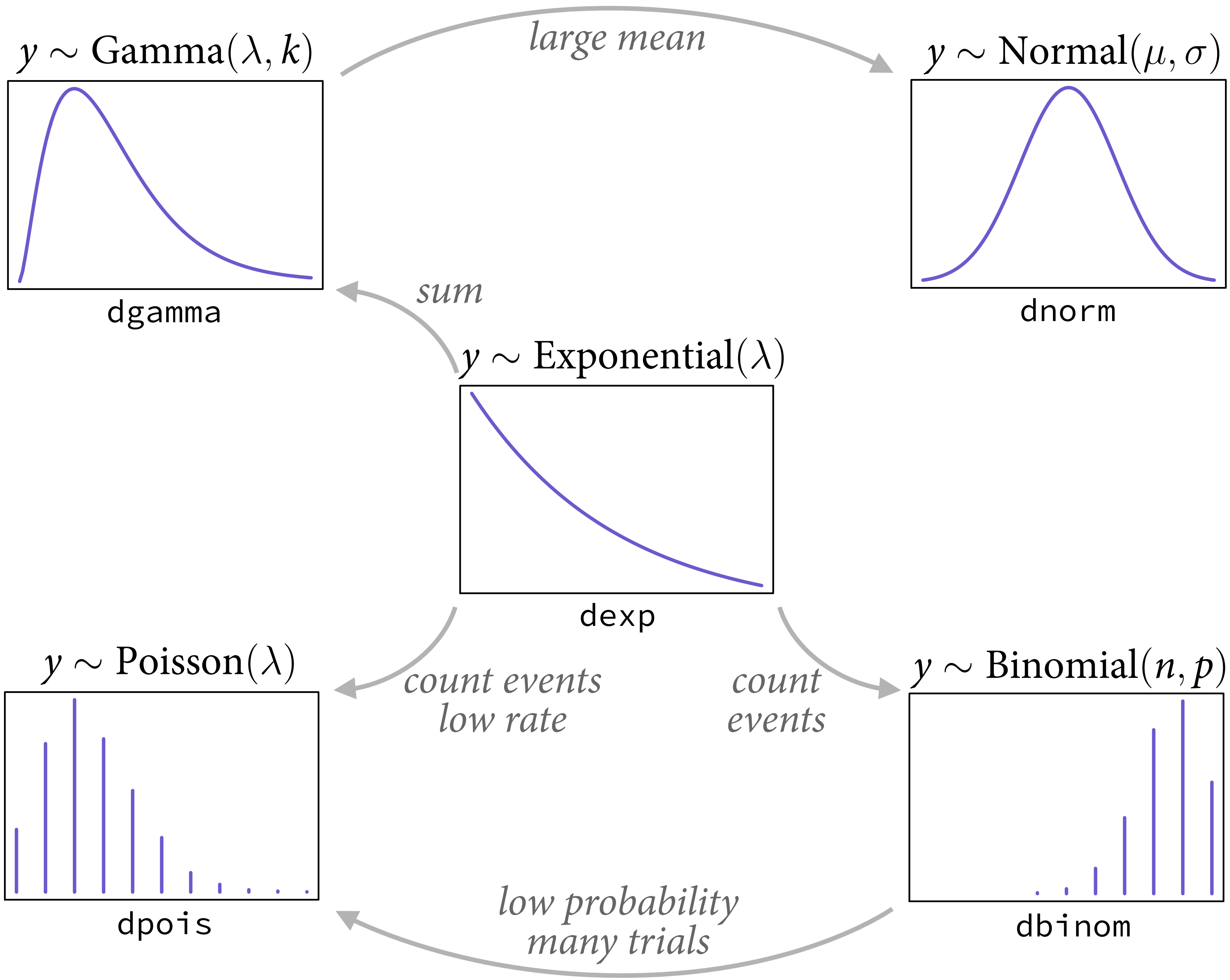
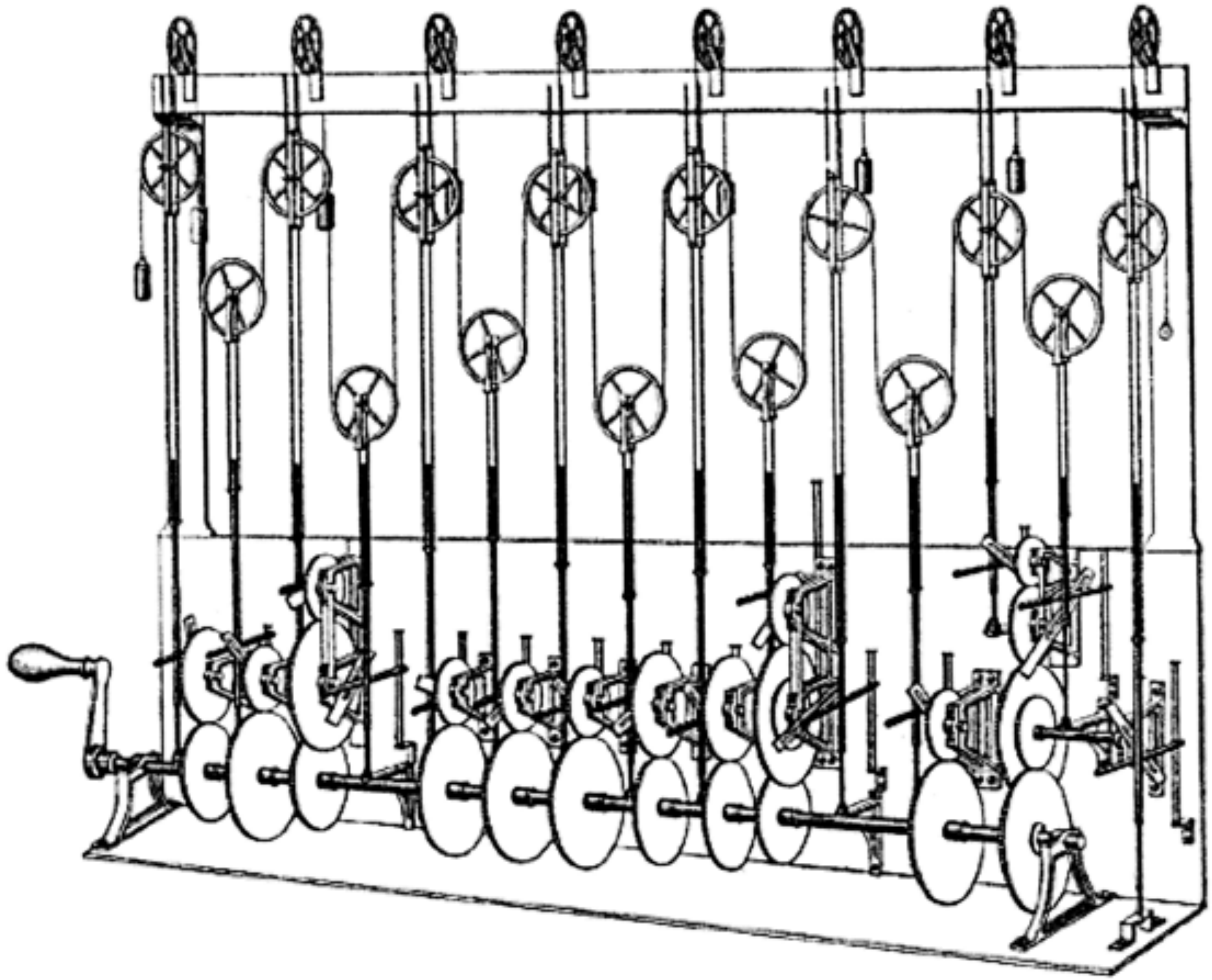

$y \sim \text{Gamma}(\lambda, k)$
large mean
$y \sim \text{Normal}(\mu, \sigma)$
dgamma
sum
dnorm
$y \sim \text{Exponential}(\lambda)$
dexp
$y \sim \text{Poisson}(\lambda)$
count events
low rate
count
events
$y \sim \text{Binomial}(n, p)$
dpois
dbinom
low probability
many trials

Figure 9.5



Tide prediction machine, 1879 design,
by William Thomson (Lord Kelvin, 1824–1907)

Generalized Linear Models

- (1) Pick an outcome distribution
    - Distances and durations: exponential, gamma (*survival* or *event history*)
    - Counts: Poisson, binomial, multinomial, geometric
    - Monsters: Ranks and ordered categories
    - Mixtures: Beta-binomial, gamma-Poisson, zero-inflated processes, occupancy models

Generalized Linear Models

- (2) Model parameters with a *link*

same units

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

Generalized Linear Models

same units

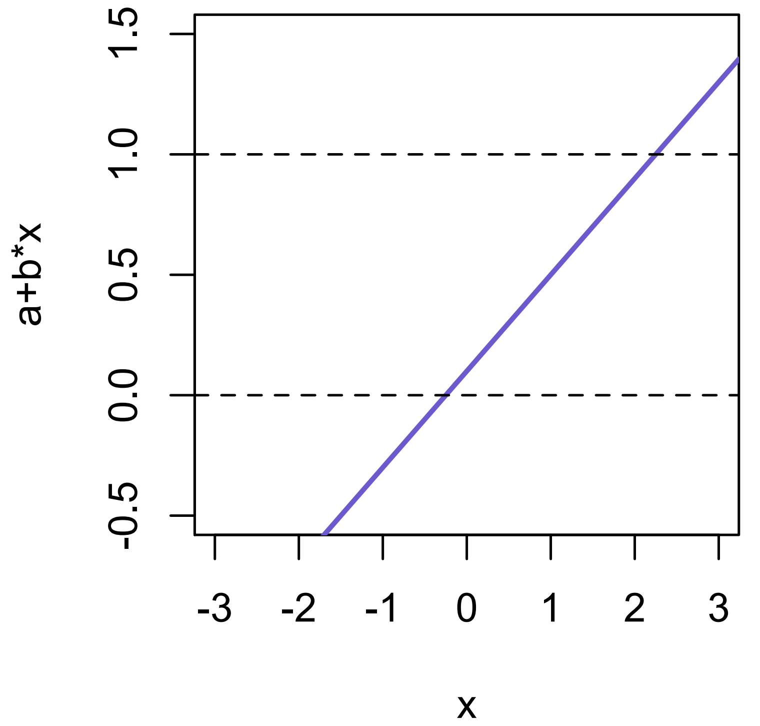
$$y_i \sim \text{Normal}(\mu_i, \sigma),$$
$$\mu_i = \alpha + \beta x_i.$$

count

$$y_i \sim \text{Binomial}(n_i, p_i)$$

probability

$$p_i \; ? \; \alpha + \beta x_i$$



Generalized Linear Models

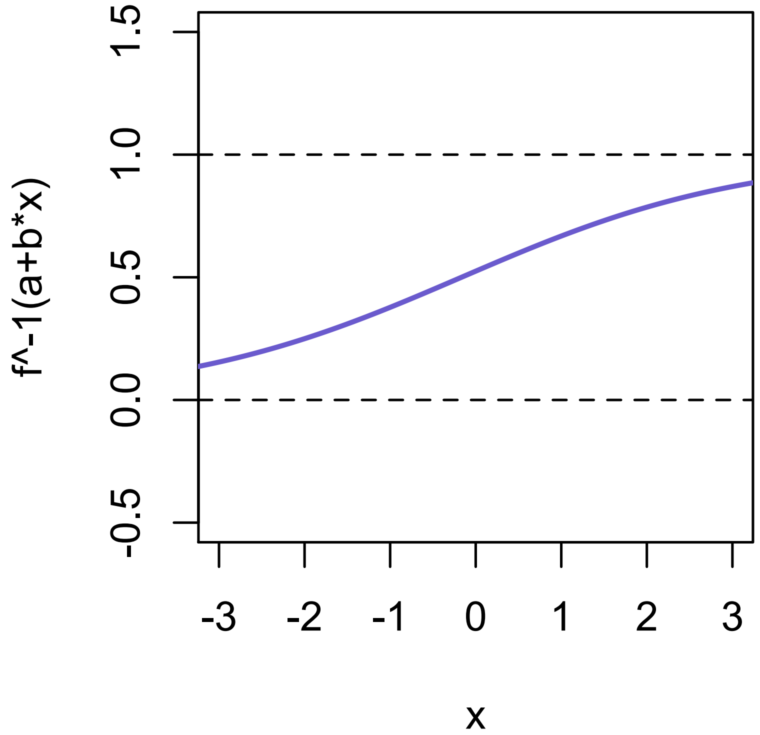
same units

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

count

$$y_i \sim \text{Binomial}(n_i, p_i)$$
$$f(p_i) = \alpha + \beta x_i$$

link function

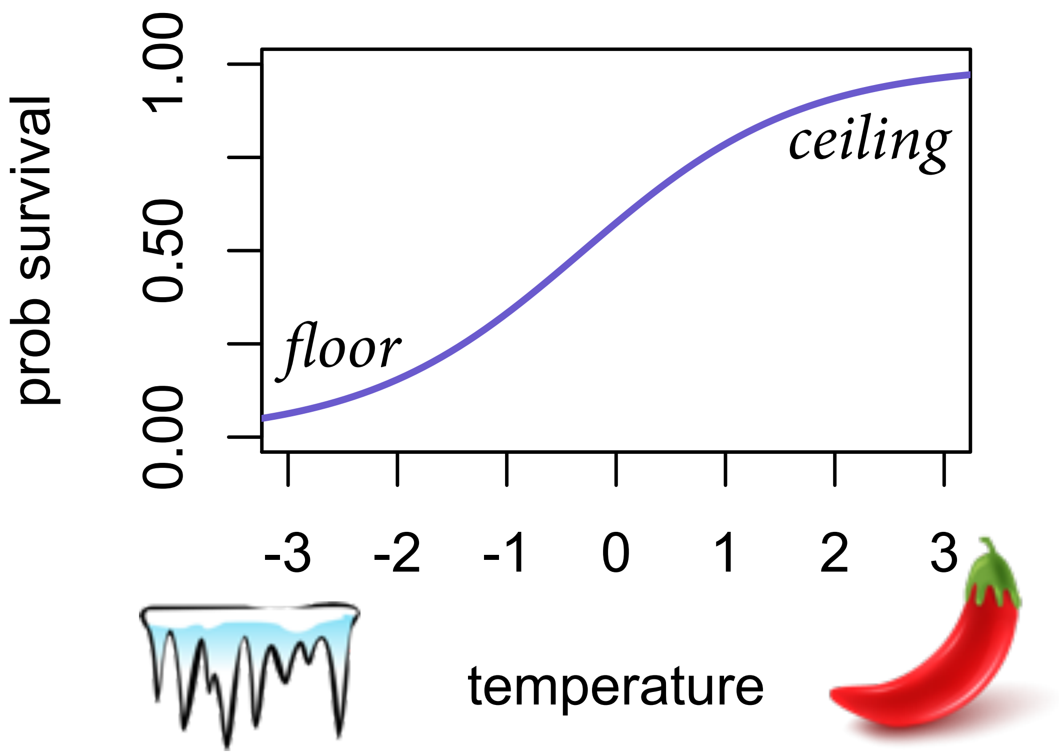


Generalized Linear Models

- (3) Compute posterior
  - Search is harder
  - Interpretation is harder
  - Links matter
  - Quadratic approximation often works, but not always
  - Safer to rely on MCMC

Everything interacts

- There are *floor* and *ceiling* effects



Everything interacts

- Linear regression:

$$\mu = \alpha + \beta x \qquad \partial \mu / \partial x = \beta.$$

- Logistic regression:

$$p = \frac{\exp(\alpha + \beta x)}{1 + \exp(\alpha + \beta x)} \qquad \frac{\partial p}{\partial x} = \frac{\beta}{2\big(1 + \cosh(\alpha + \beta x)\big)}$$

Binomial distribution

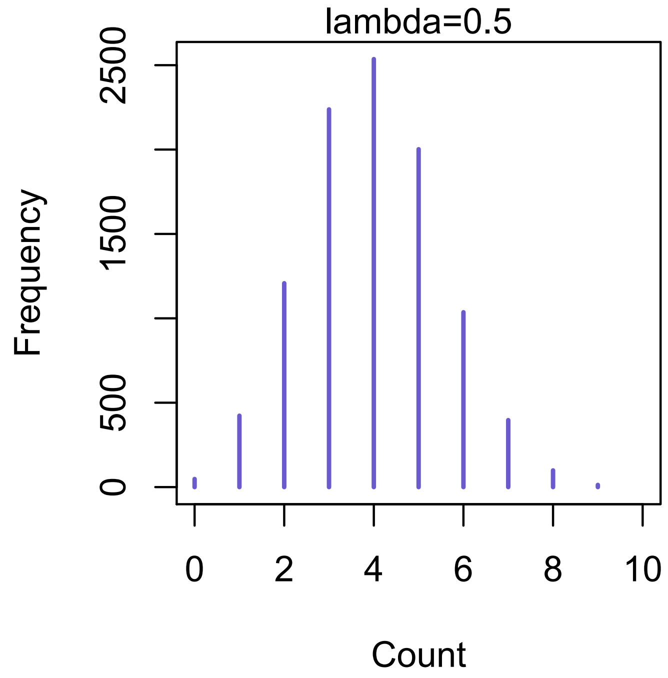
- Counts of a specific event out of *n* possibilities
- Constant expected value
- Maxent: Binomial

$$y \sim \text{Binomial}(n, p)$$

count “successes”

number of trials

probability of success



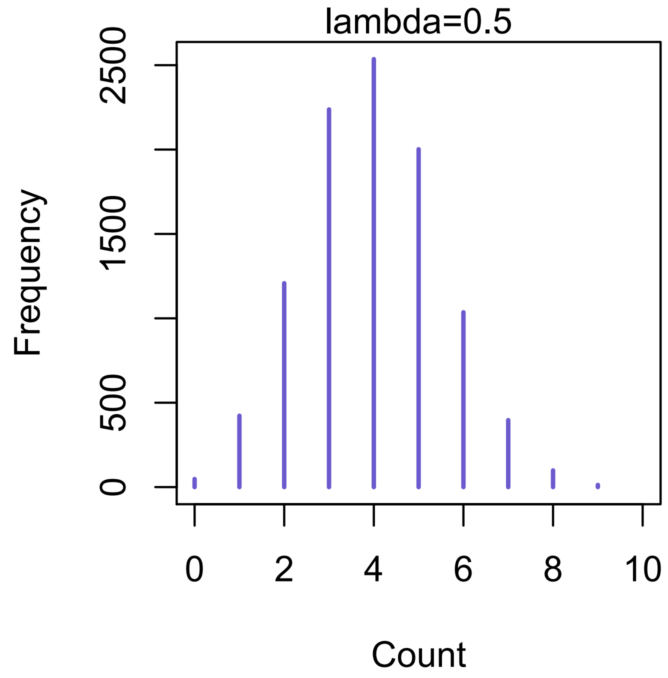
Binomial distribution

- Counts of a specific event out of n possibilities
- Constant expected value
- Maxent: Binomial

$$y \sim \text{Binomial}(n, p)$$

$$\text{E}(y) = np$$
$$\text{var}(y) = np(1-p)$$

Mean and variance not independent



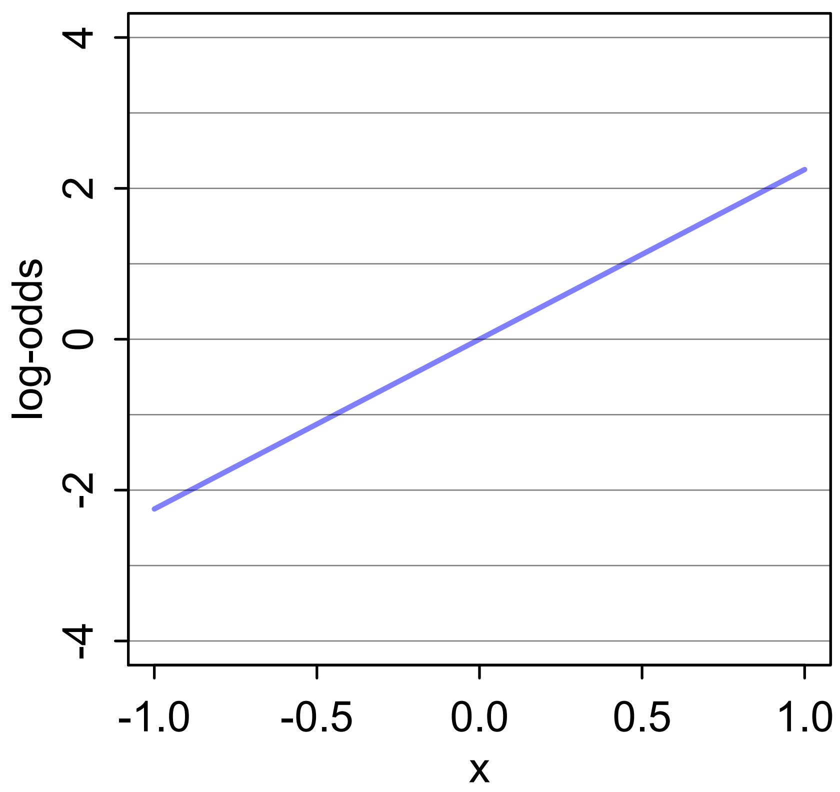
Need a link

$$y \sim \text{Binomial}(n, p)$$

- y and p on different scales
- y: count
- p: probability
- Want to model p as function of predictor variables
- Must bound it to [0,1] interval

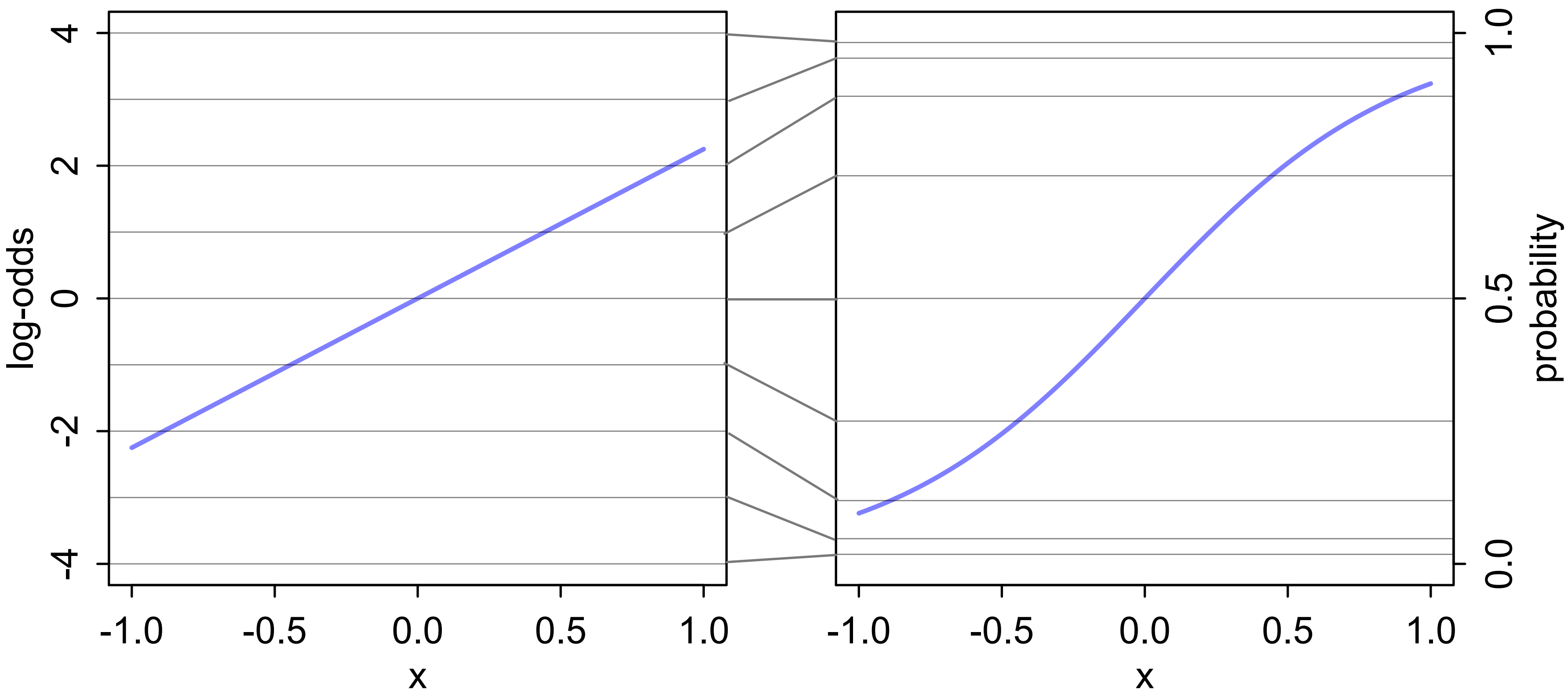
Logit link

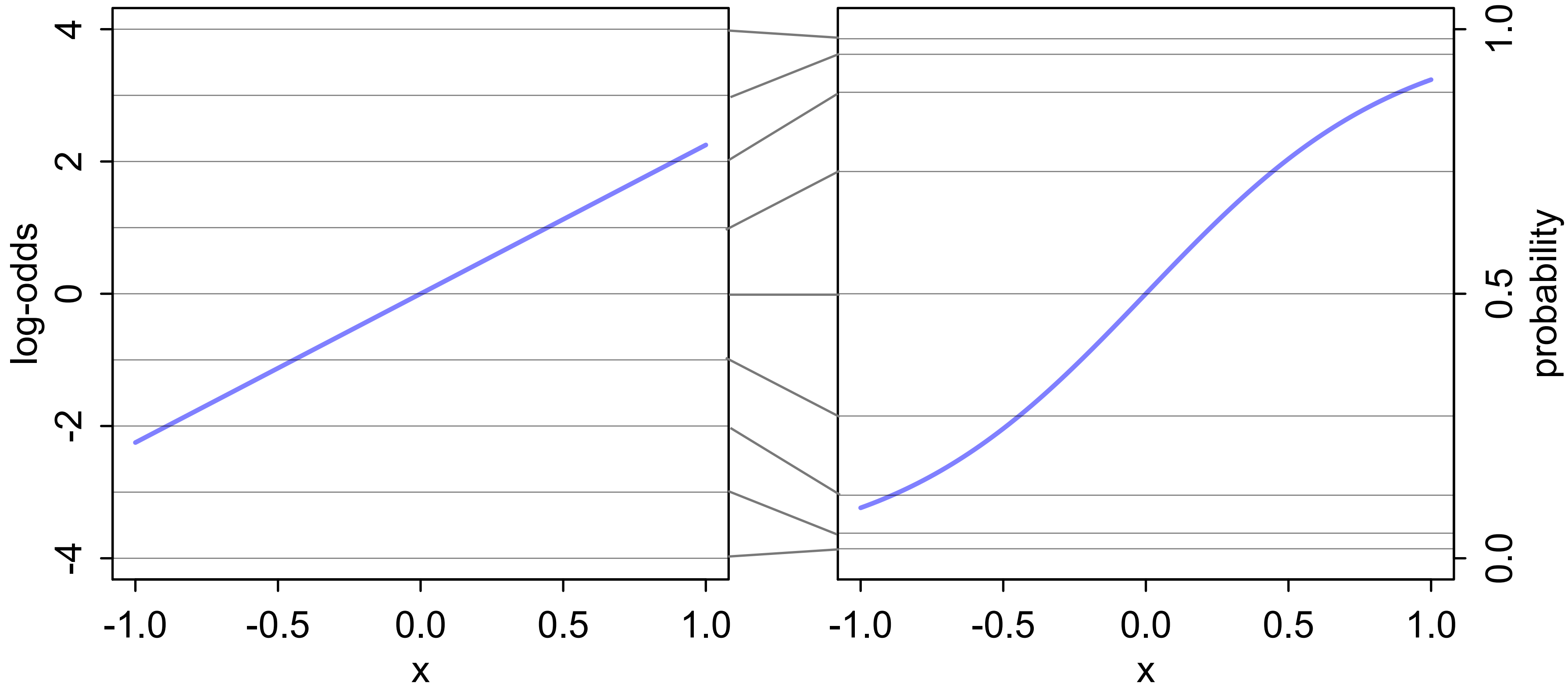
- Goal: map linear model to [0,1]



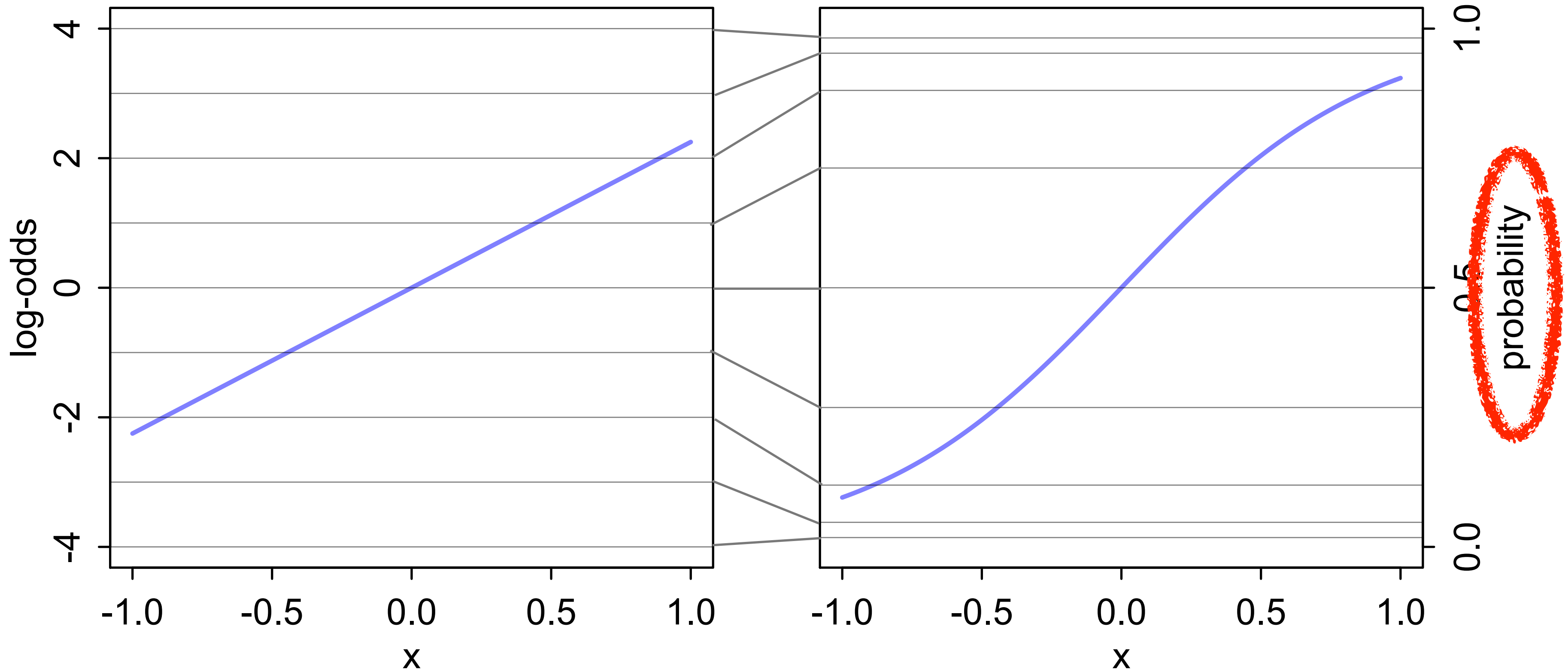
Logit link

- Goal: map linear model to [0,1]

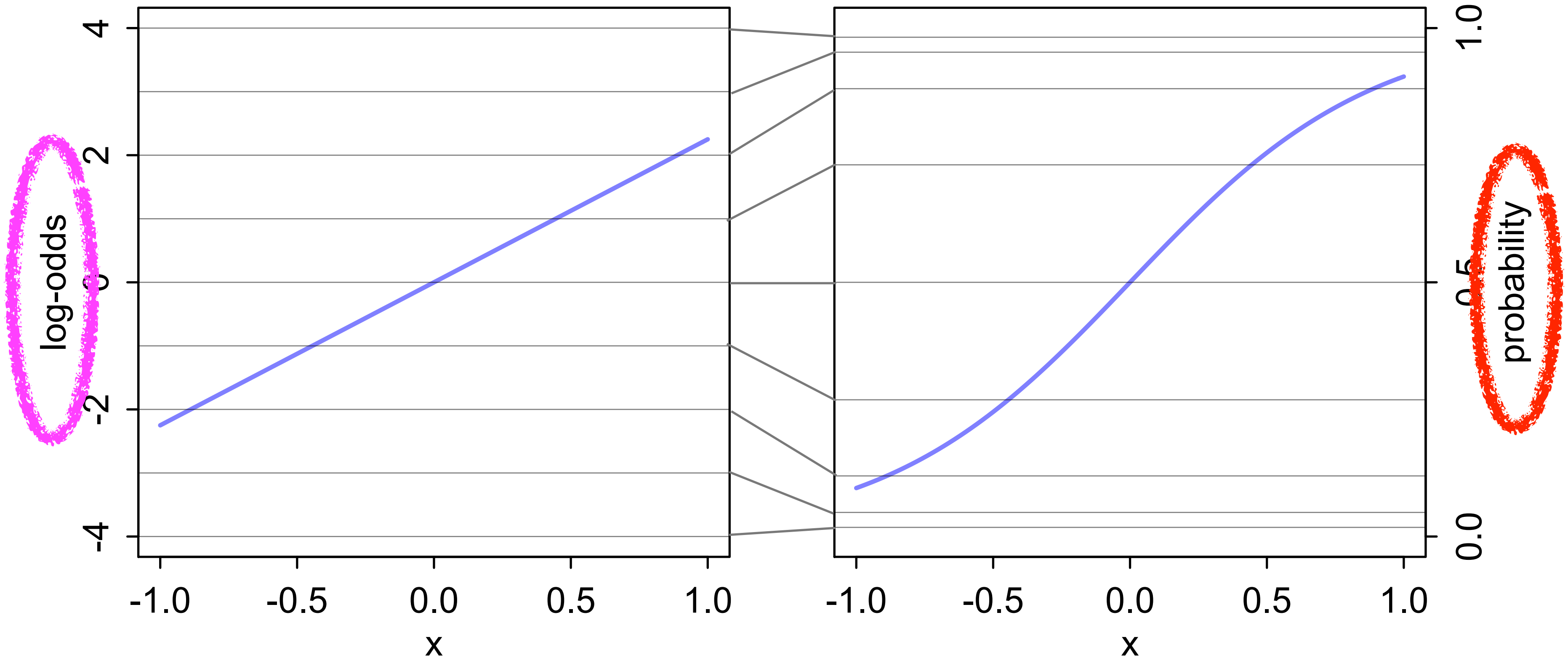




$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$

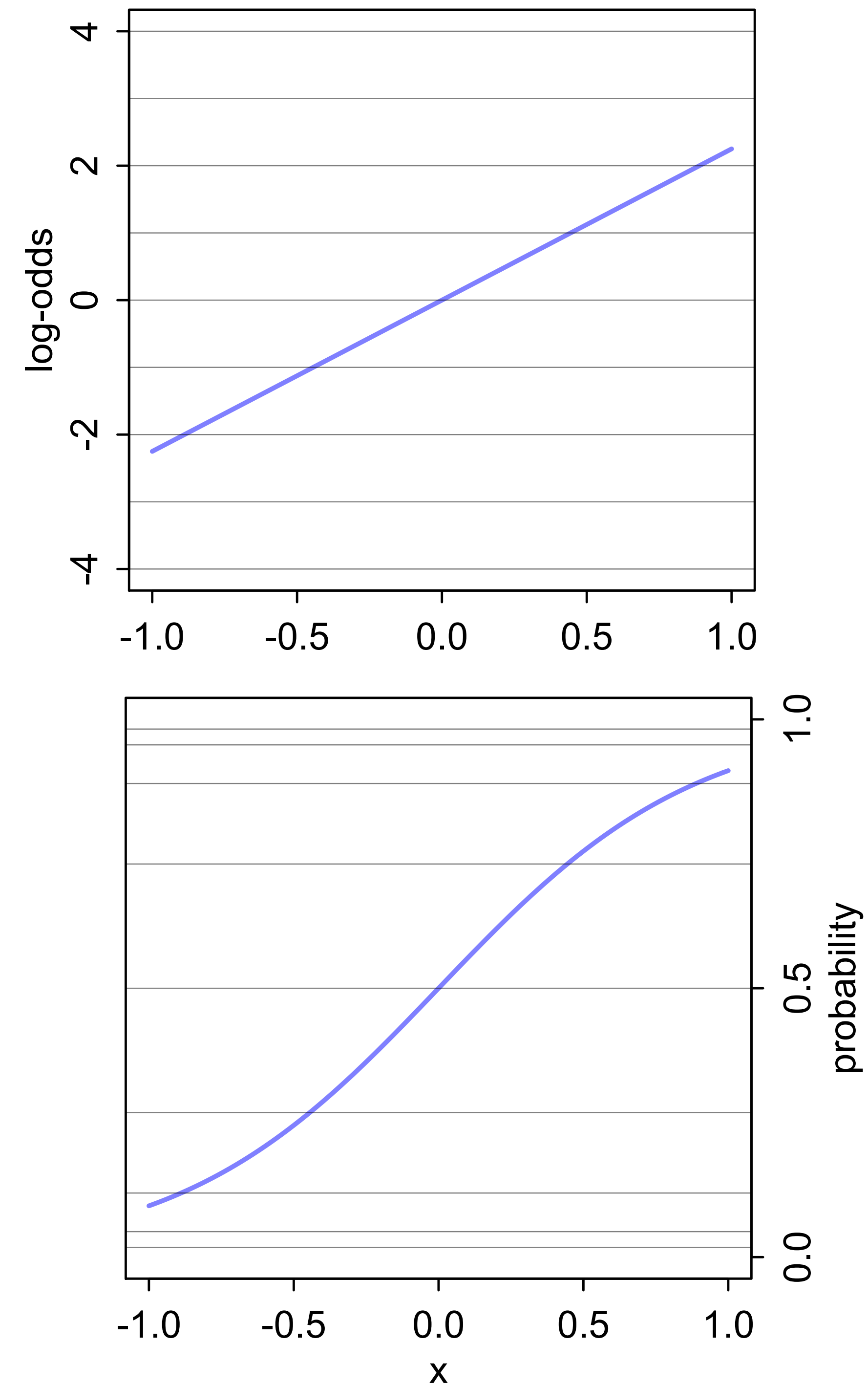


$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$



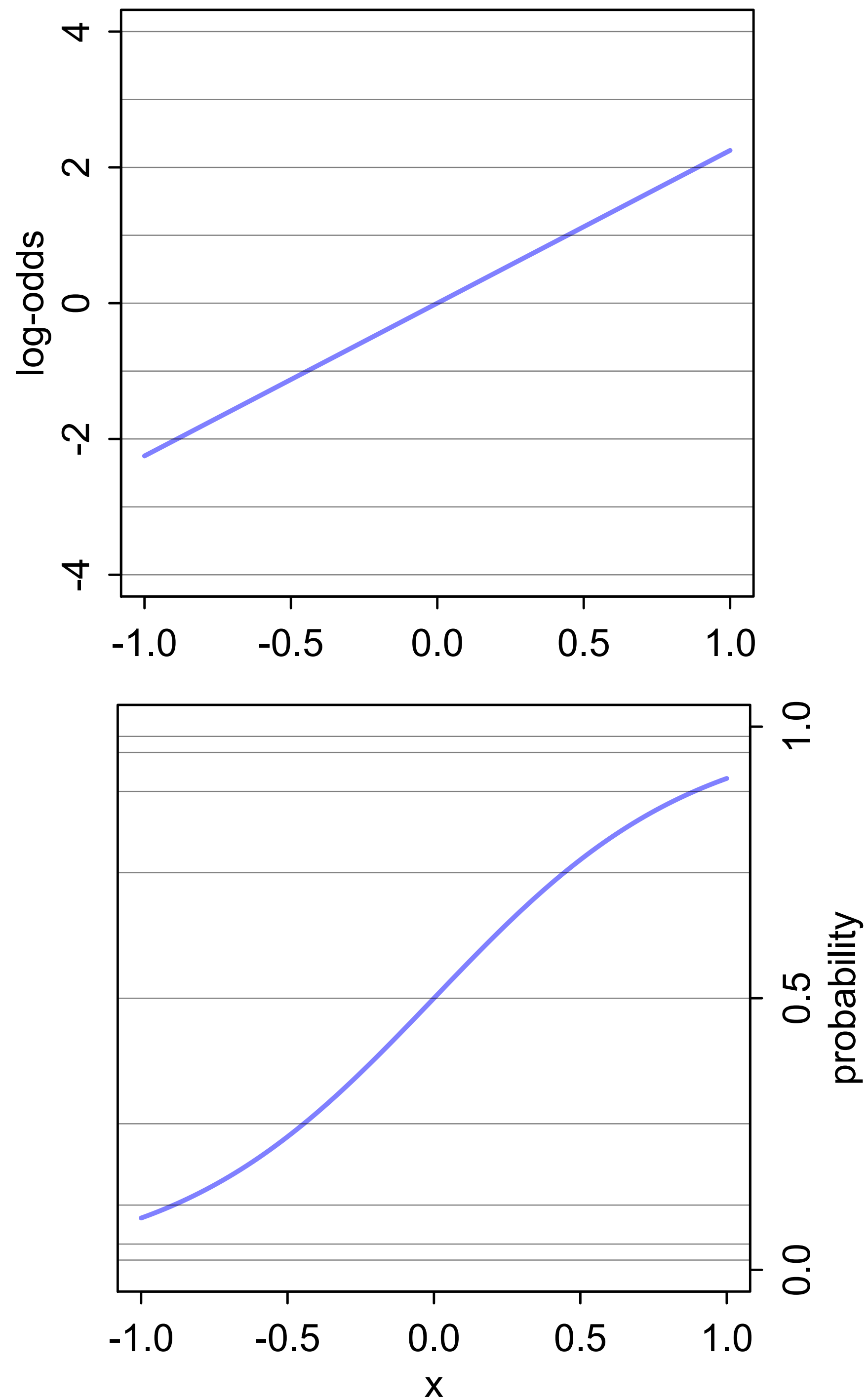
$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$

$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$



$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$

$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i} = \alpha + \beta x_i$$



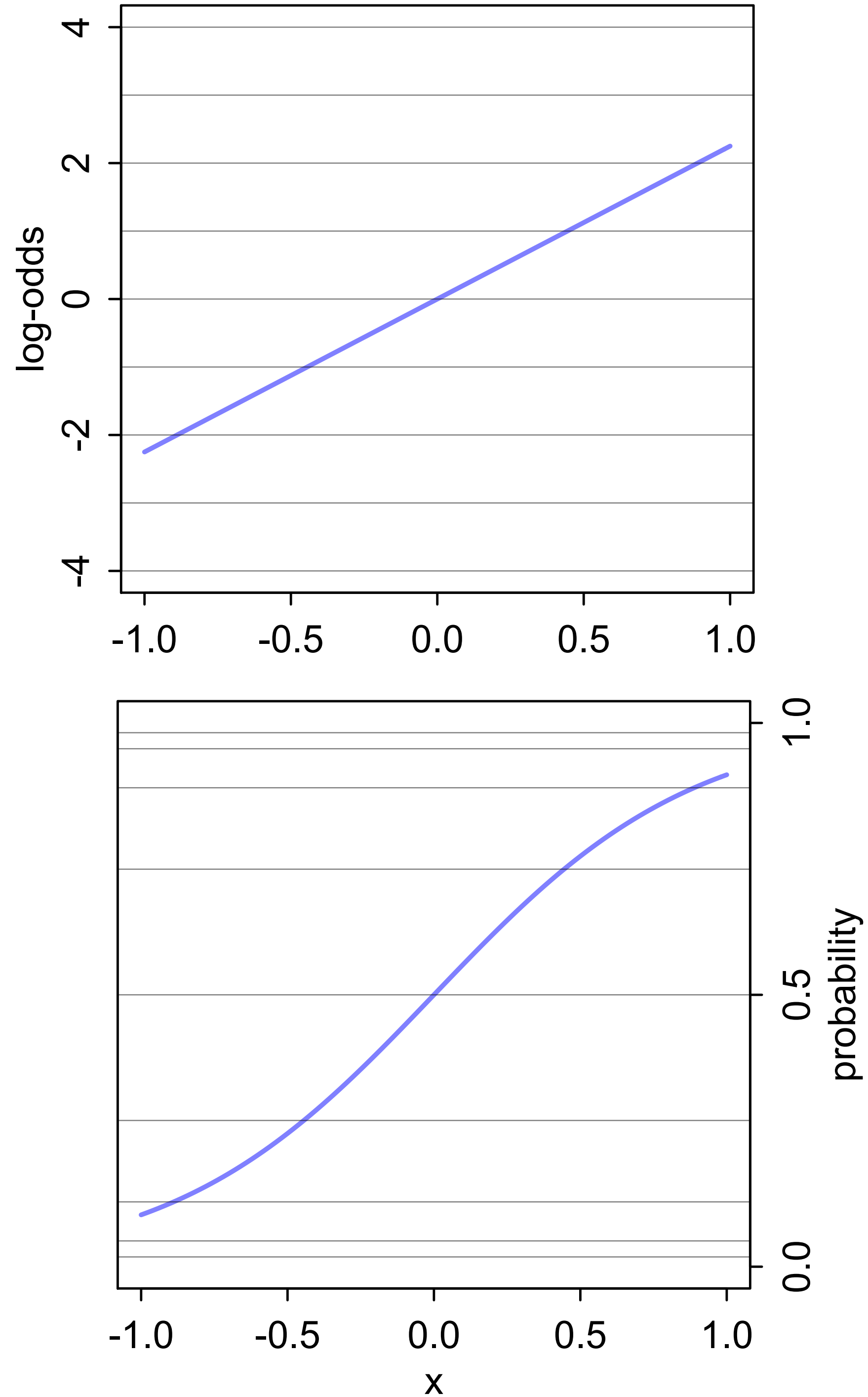
$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$

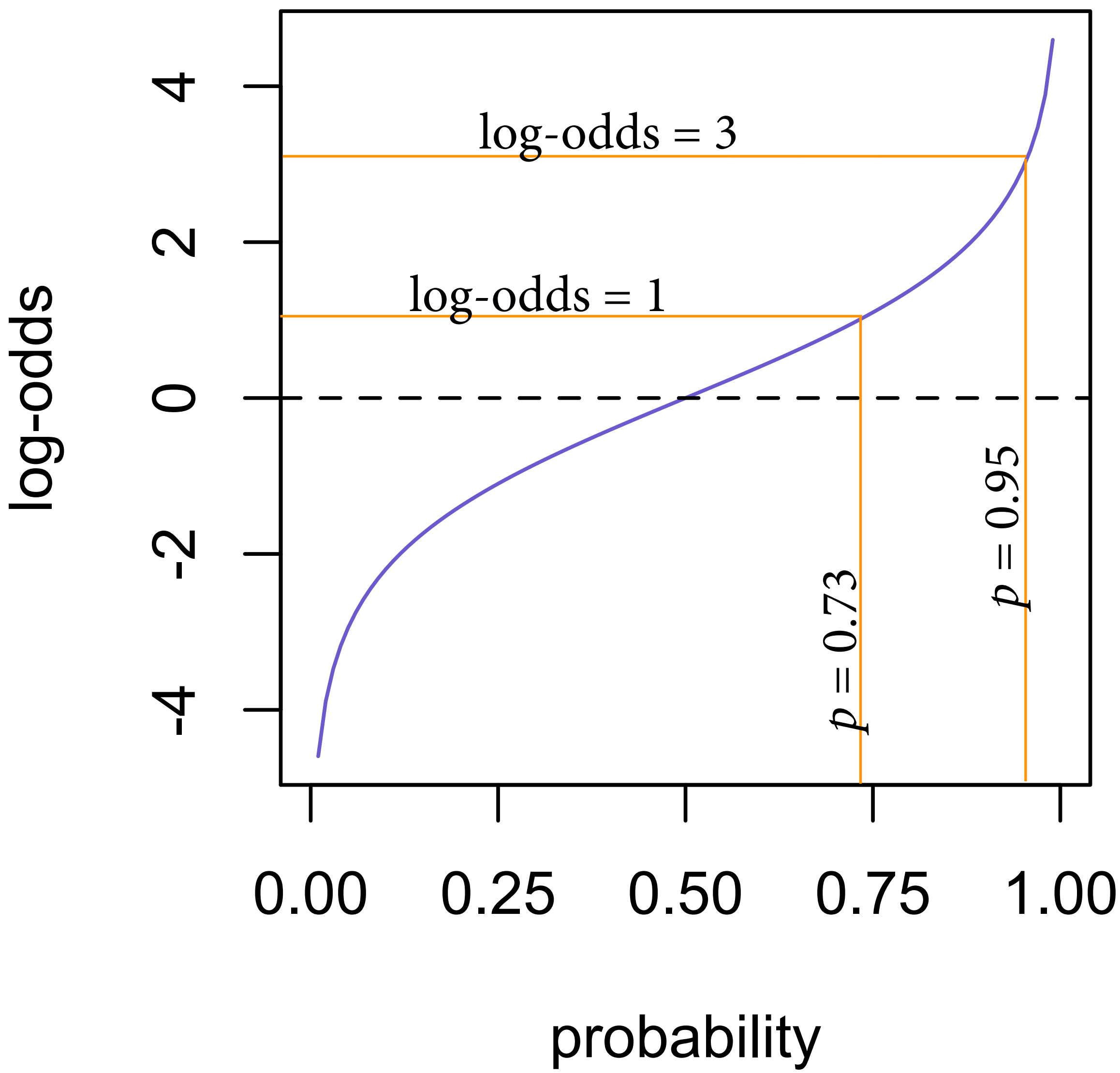
$$\text{logit}(p_i) = \log \frac{p_i}{1 - p_i} = \alpha + \beta x_i$$

Solve for p_i:

$$p_i = \frac{\exp(\alpha + \beta x_i)}{1 + \exp(\alpha + \beta x_i)}$$

inverse-link is logistic



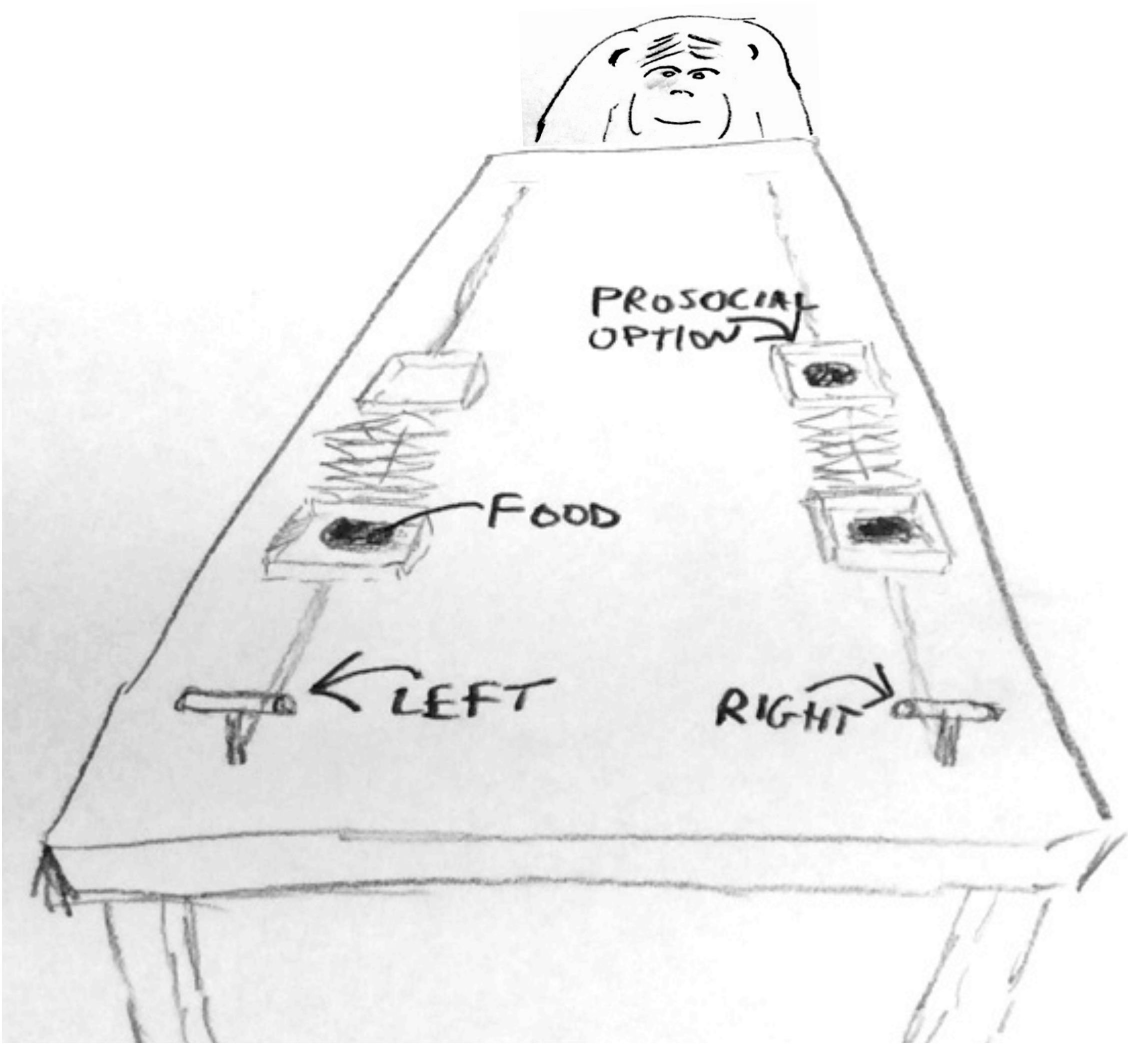

log-odds = 3
log-odds = 1
p = 0.73
p = 0.95
log-odds
4
2
0
-2
-4
0.00
0.25
0.50
0.75
1.00
probability

Logit link

$$y_i \sim \text{Binomial}(n, p_i)$$
$$\text{logit}(p_i) = \alpha + \beta x_i$$

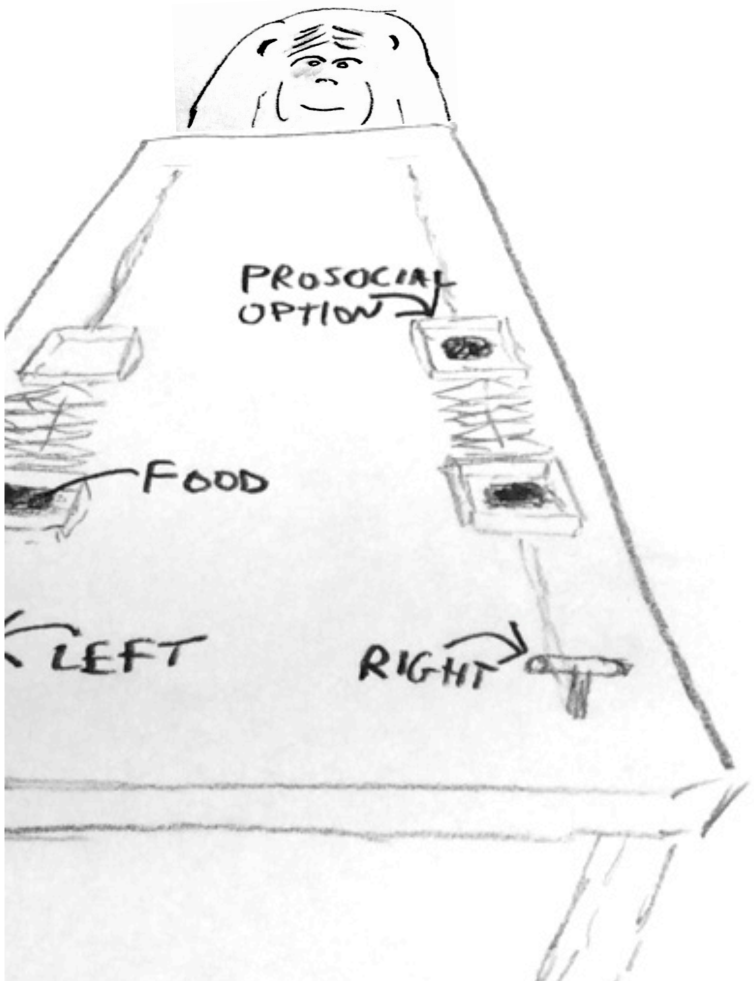
- Where does this thing come from?
- Several good answers:
    - “Natural” link inside probability formula
    - log-odds is fundamental parameter
    - See Overthinking box, pages 313–314
- Other links sometimes justified
    - Probit (common in economics)
    - Complementary-log-log (cloglog)
- If you have a real scientific model, link is automatic

Prosocial chimpanzees



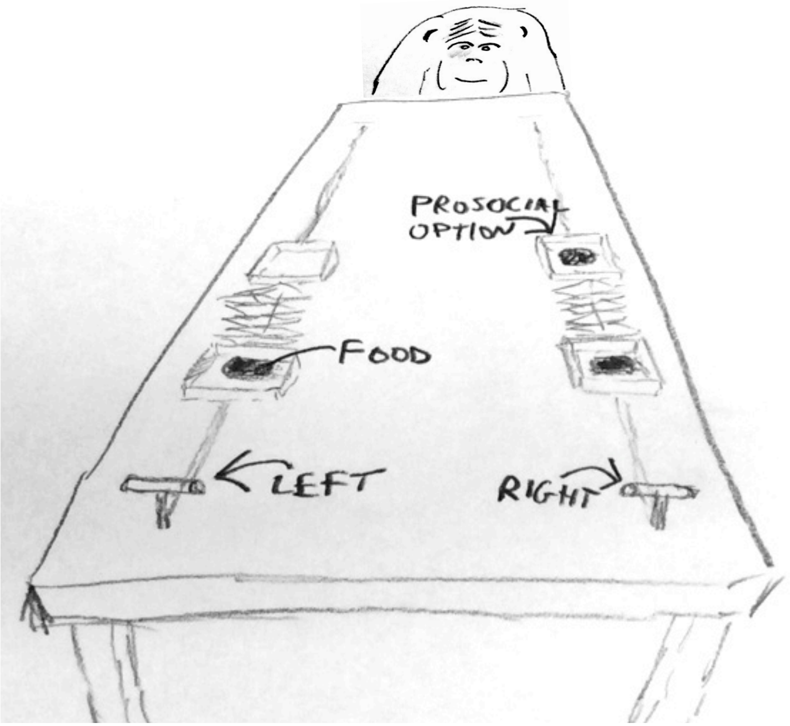
Prosocial chimpanzees

- Two *conditions*: (1) partner, (2) alone
- Two *options*: (1) prosocial, (2) asocial
- Two *outcomes*: (1) left lever, (2) right lever
- Want to predict *outcome* as function of *condition* and which side *option* is on
    - Do chimps prefer *left* lever when *partner* present and *prosocial* on left? => interaction!



Prosocial chimpanzees

- Coding treatments:
  - (1) right/no-partner
  - (2) left/no-partner
  - (3) right/partner
  - (4) left/partner



$$
\begin{aligned}
L_i &\sim \text{Binomial}(1, p_i) \\
\text{logit}(p_i) &= \alpha_{\text{ACTOR}[i]} + \beta_{\text{TREATMENT}[i]} \\
\alpha_j &\sim \text{to be determined} \\
\beta_k &\sim \text{to be determined}
\end{aligned}
$$

Binomial(1,*p*) often called *logistic regression*

Same as Bernoulli(*p*)

Logit link priors

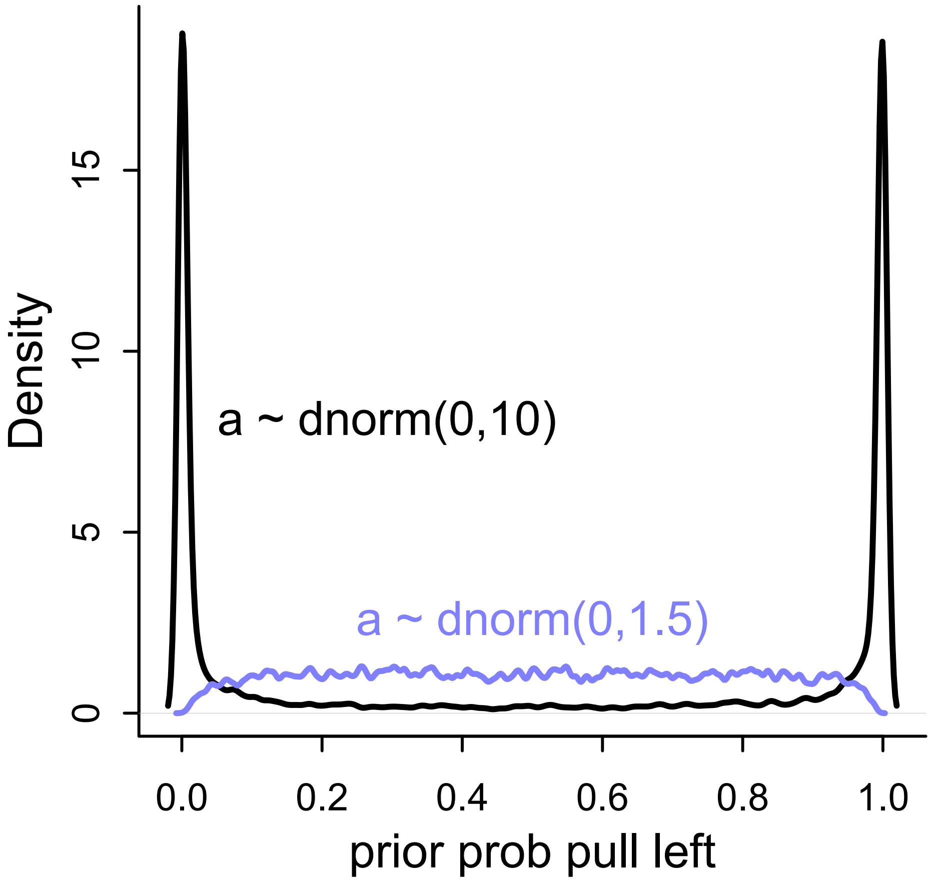
- Prior on logit scale not same shape as prior on probability scale
- Use prior simulation to understand

$$
\begin{aligned}
L_i &\sim \text{Binomial}(1, p_i) \\
\text{logit}(p_i) &= \alpha \\
\alpha &\sim \text{Normal}(0, \omega)
\end{aligned}
$$

Logit link priors

$$L_i \sim \text{Binomial}(1, p_i)$$
$$\text{logit}(p_i) = \alpha$$
$$\alpha \sim \text{Normal}(0, \omega)$$

```
m11.1 <- quap(
    alist(
        pulled_left ~ dbinom( 1 , p ) ,
        logit(p) <- a ,
        a ~ dnorm( 0 , 10 )
    ) , data=d )
```



Density
0
5
10
15
a ~ dnorm(0,10)
a ~ dnorm(0,1.5)
0.0
0.2
0.4
0.6
0.8
1.0
prior prob pull left

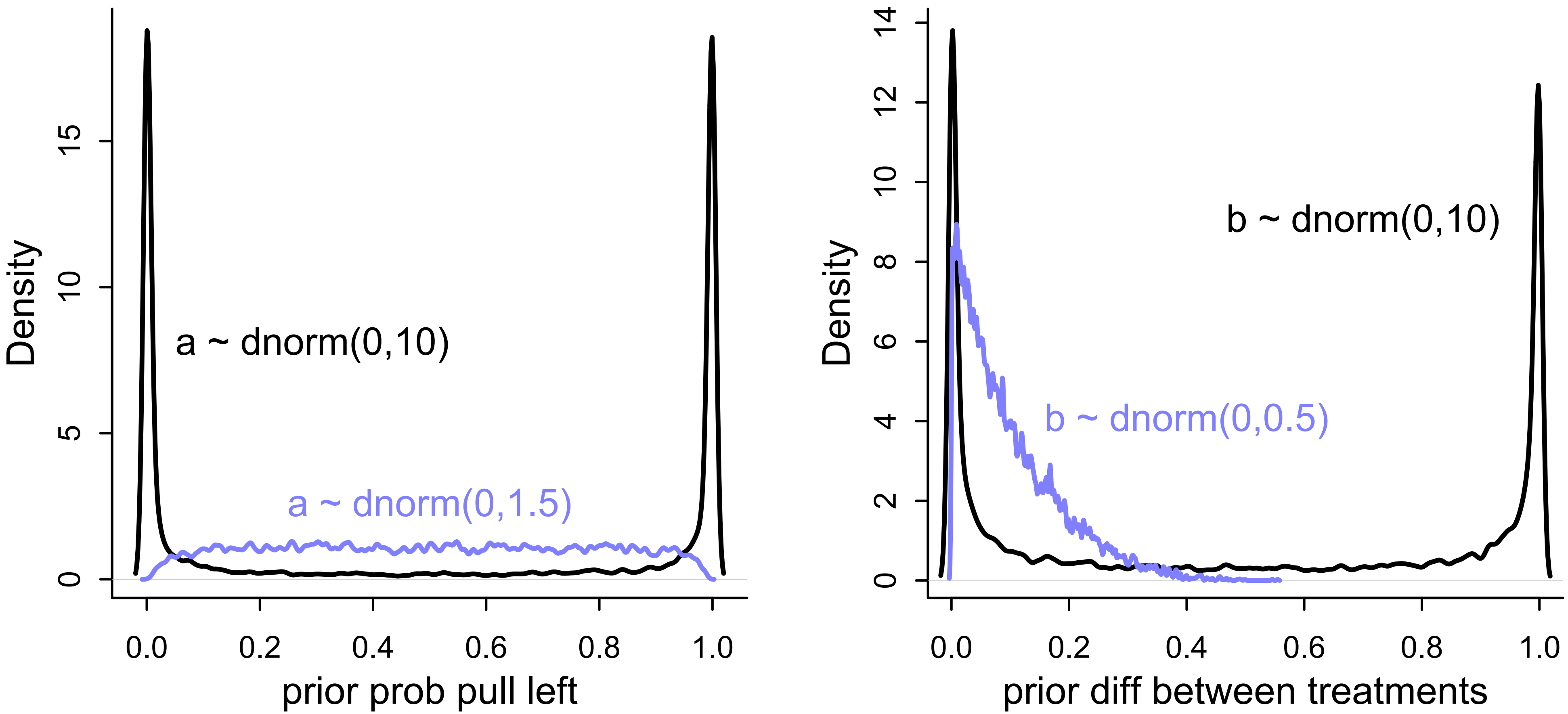
Figure 11.3

Logit link priors

- What about slopes?

R code 11.7

```
m11.2 <- quap(
    alist(
        pulled_left ~ dbinom( 1 , p ) ,
        logit(p) <- a + b[treatment] ,
        a ~ dnorm( 0 , 1.5 ),
        b[treatment] ~ dnorm( 0 , 10 )
    ) , data=d )
```



Density
prior prob pull left
a ~ dnorm(0,10)
a ~ dnorm(0,1.5)
0.0
0.2
0.4
0.6
0.8
1.0
0
5
10
15
Density
prior diff between treatments
b ~ dnorm(0,10)
b ~ dnorm(0,0.5)
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
12
14

Figure 11.3

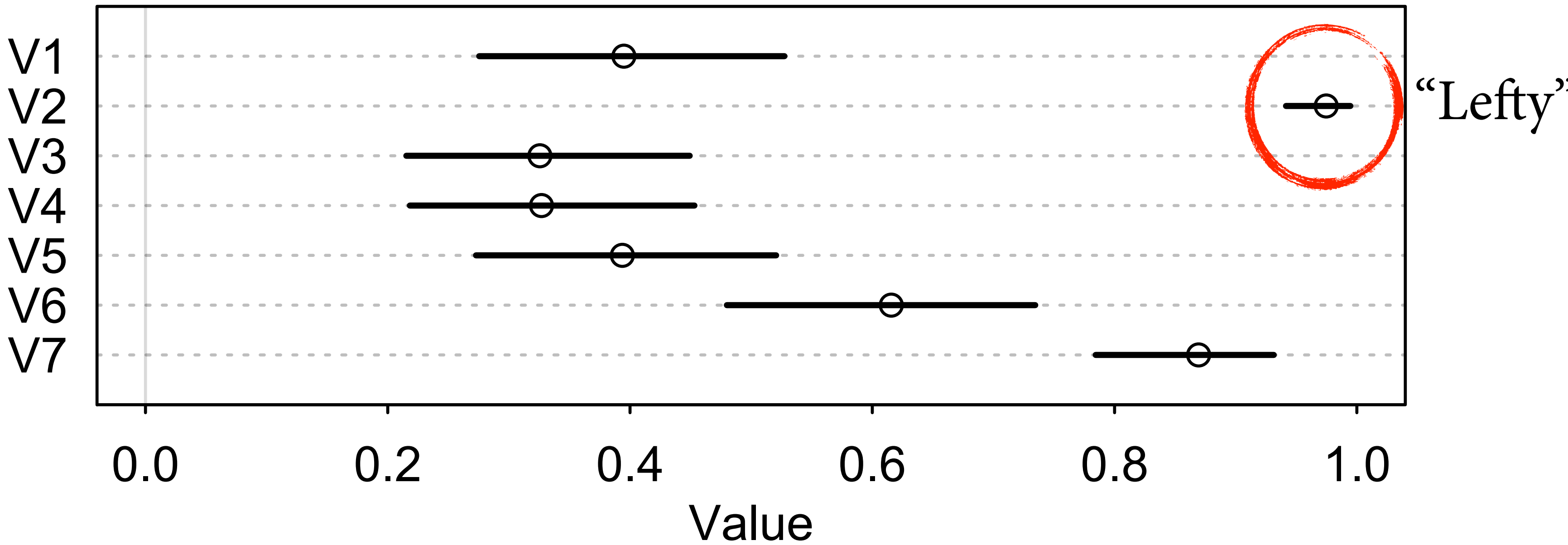
```
# particles in 11-dimensional space
m11.4 <- ulam(
    alist(
        pulled_left ~ dbinom( 1 , p ) ,
        logit(p) <- a[actor] + b[treatment] ,
        a[actor] ~ dnorm( 0 , 1.5 ),
        b[treatment] ~ dnorm( 0 , 0.5 )
    ) ,
    data=dat_list , chains=4 )
precis( m11.4 , depth=2 )
```

			mean	sd	5.5%	94.5%	n_eff	Rhat
Chimpanzees		a[1]	-0.44	0.34	-0.97	0.11	736	1
		a[2]	3.90	0.77	2.78	5.22	921	1
		a[3]	-0.75	0.34	-1.29	-0.20	886	1
		a[4]	-0.74	0.34	-1.28	-0.19	770	1
		a[5]	-0.44	0.34	-0.98	0.08	832	1
		a[6]	0.48	0.34	-0.08	1.02	854	1
		a[7]	1.96	0.41	1.29	2.61	847	1
Treatments	RN	b[1]	-0.05	0.29	-0.51	0.42	781	1
	LN	b[2]	0.48	0.29	0.03	0.94	657	1
	RP	b[3]	-0.39	0.28	-0.83	0.07	669	1
	LP	b[4]	0.36	0.29	-0.11	0.81	732	1

Individual differences

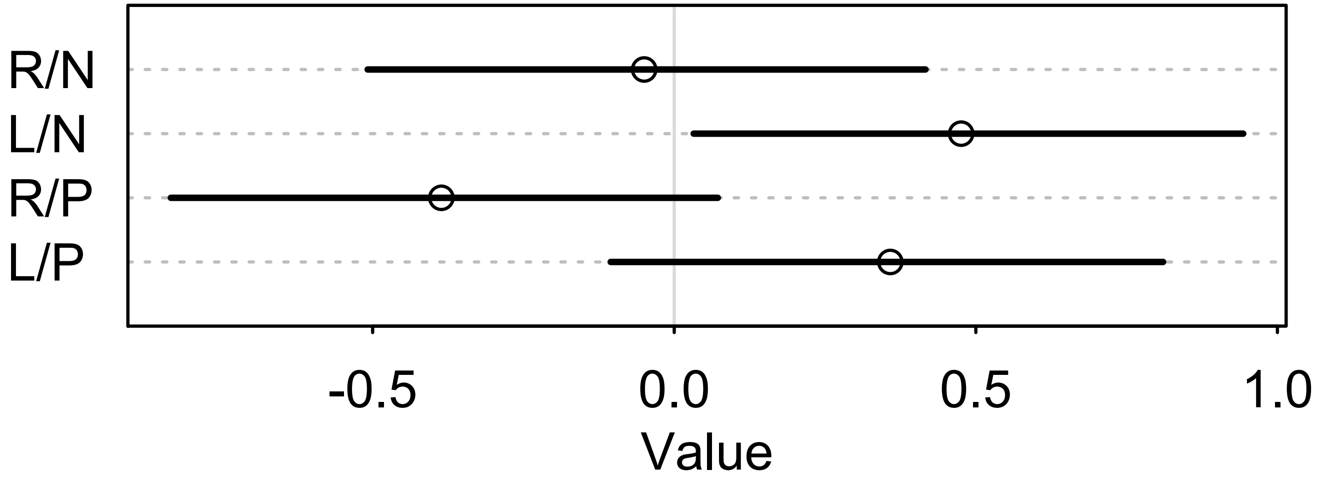
R code 11.11

```
post <- extract.samples(m11.4)
p_left <- inv_logit( post$a )
plot( precis( as.data.frame(p_left) ) , xlim=c(0,1) )
```



Treatments

```
labs <- c("R/N","L/N","R/P","L/P")
plot( precis( m11.4 , depth=2 , pars="b" ) , labels=labs )
```



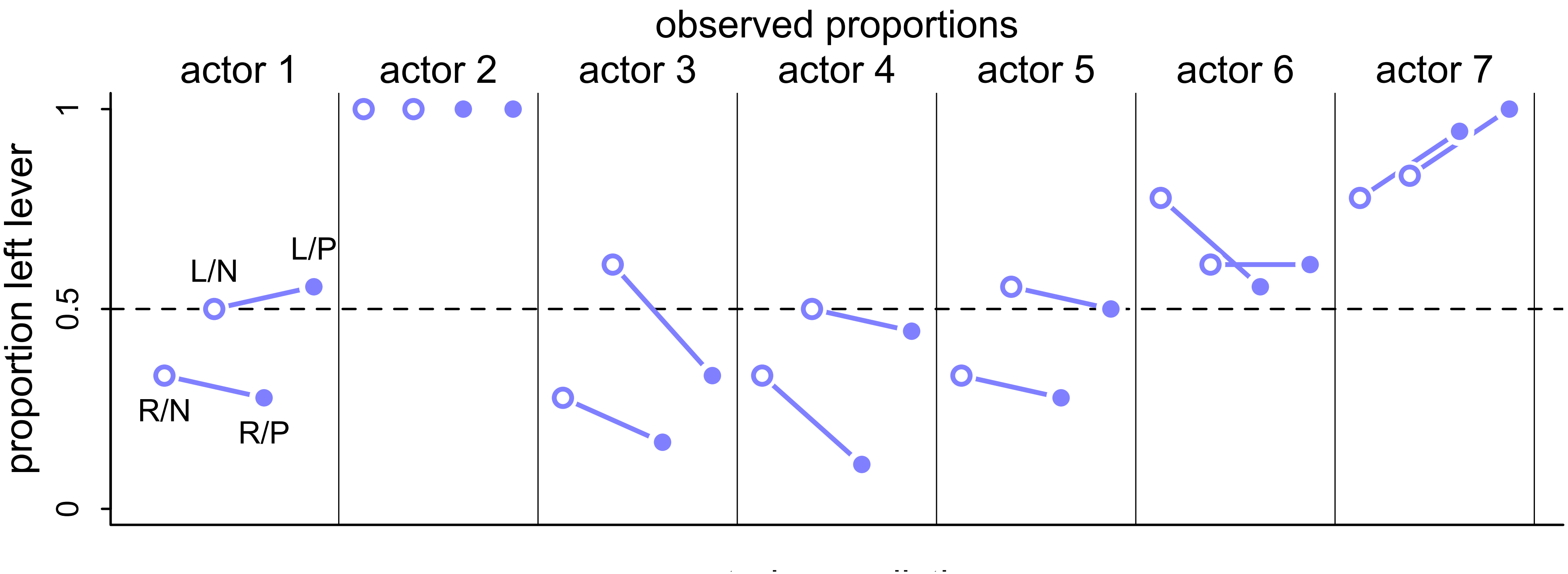
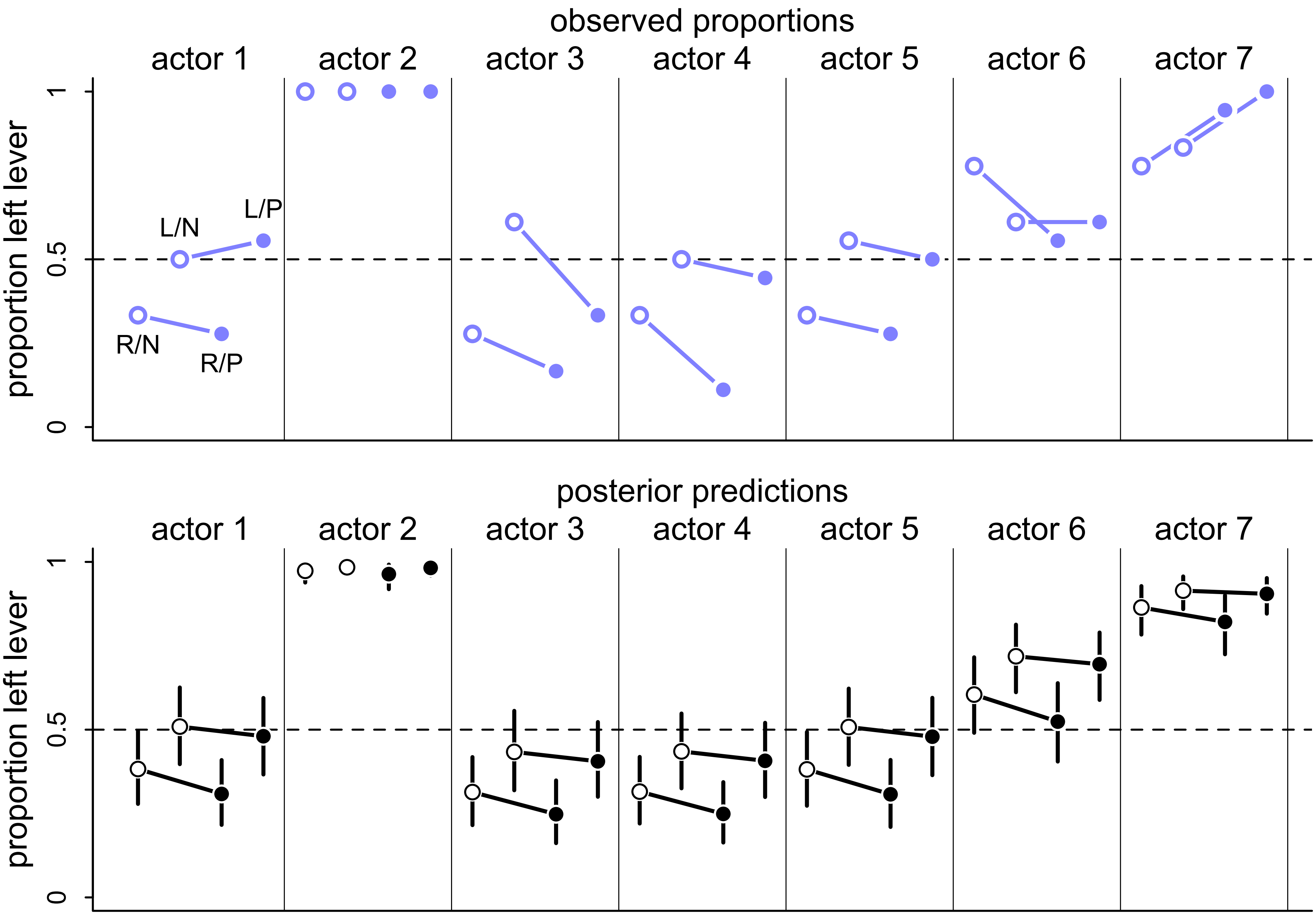

observed proportions
actor 1
actor 2
actor 3
actor 4
actor 5
actor 6
actor 7
proportion left lever
1
0.5
0
L/N
L/P
R/N
R/P

Figure 11.4



observed proportions
actor 1
actor 2
actor 3
actor 4
actor 5
actor 6
actor 7
proportion left lever
0
0.5
1
L/N
L/P
R/N
R/P
posterior predictions
actor 1
actor 2
actor 3
actor 4
actor 5
actor 6
actor 7
proportion left lever
0
0.5
1

Figure 11.4

Comparing no-interaction

```
m11.5 <- ulam(
    alist(
        pulled_left ~ dbinom( 1 , p ) ,
        logit(p) <- a[actor] + bs[side] + bc[cond] ,
        a[actor] ~ dnorm( 0 , 1.5 ),
        bs[side] ~ dnorm( 0 , 0.5 ),
        bc[cond] ~ dnorm( 0 , 0.5 )
    ) ,
    data=dat_list2 , chains=4 , log_lik=TRUE )
```

R code 11.19

```
compare( m11.5 , m11.4 , func=LOO )
```

```
        LOO pLOO dLOO weight    SE  dSE
m11.5 531.2  7.9  0.0   0.66 19.17   NA
m11.4 532.6  8.7  1.4   0.34 19.01 1.28
```

Relative and absolute effects

- Parameters on *relative* effect scale
- Predictions on *absolute* effect scale
- Proportional odds: Relative effect measure

R code 11.22

```
post <- extract.samples(m11.4)
mean( exp(post$b[,4]-post$b[,2]) )
```

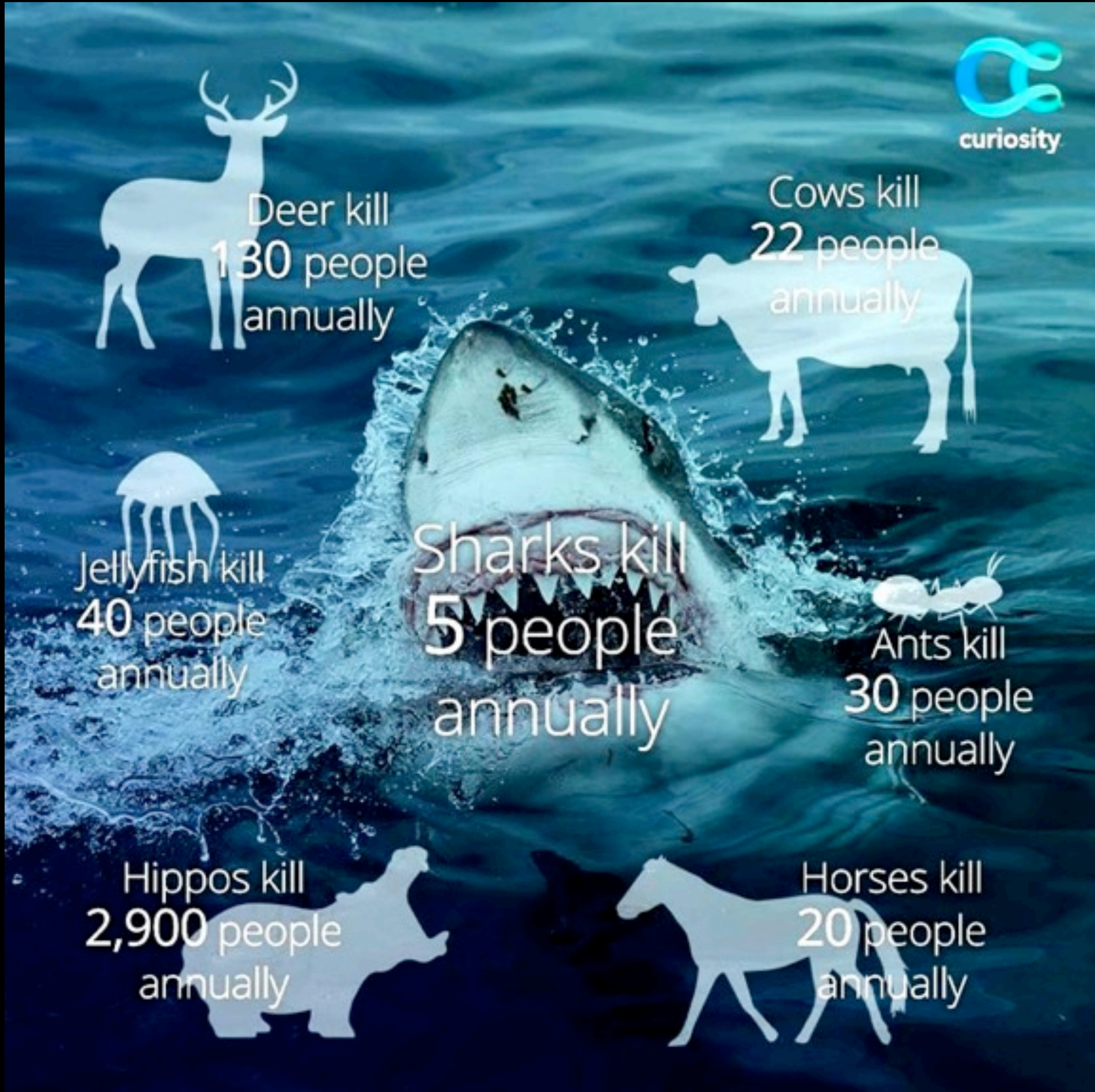
```
[1] 0.9206479
```

Relative and absolute effects

- Parameters on *relative* effect scale
- Predictions on *absolute* effect scale
- Using relative effects may exaggerate importance of predictor
  - Good for scaring people, getting published
  - Not so good for public health, scientific progress
  - But needed for causal inference

relative shark

absolute penguin



curiosity
Deer kill 130 people annually
Cows kill 22 people annually
Jellyfish kill 40 people annually
Sharks kill 5 people annually
Ants kill 30 people annually
Hippos kill 2,900 people annually
Horses kill 20 people annually

Risk communication

- Many people mistake relative risk for absolute risk
- Example:
  - 1/1000 women develop blood clots
  - 3/1000 women on birth control develop blood clots
  - => 200% increase in blood clots!
  - Change in probability is only 0.002
  - Pregnancy much more dangerous than blood clots