Statistical Rethinking

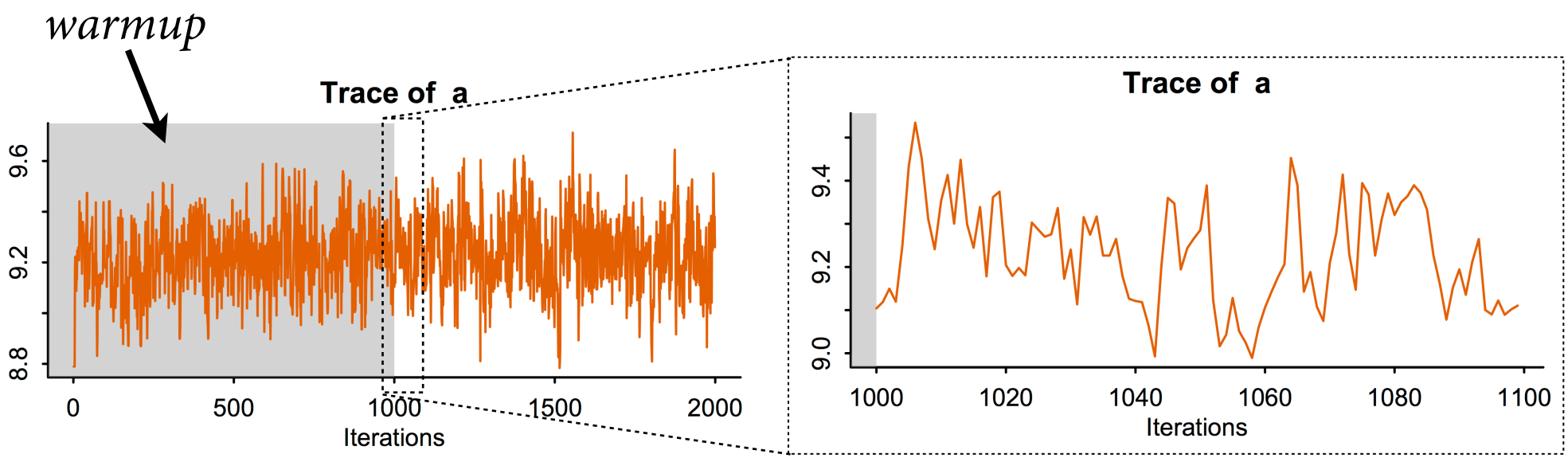
Week 6:
Markov Chains
& Maximum Entropy

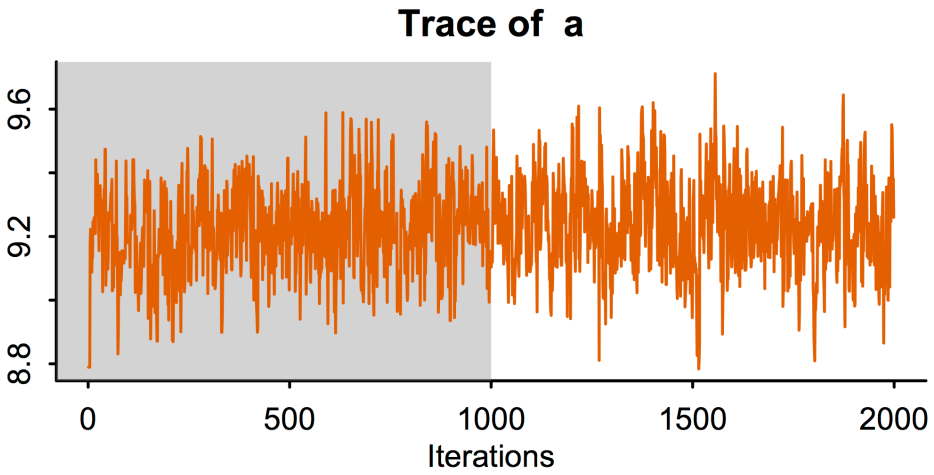
Richard McElreath

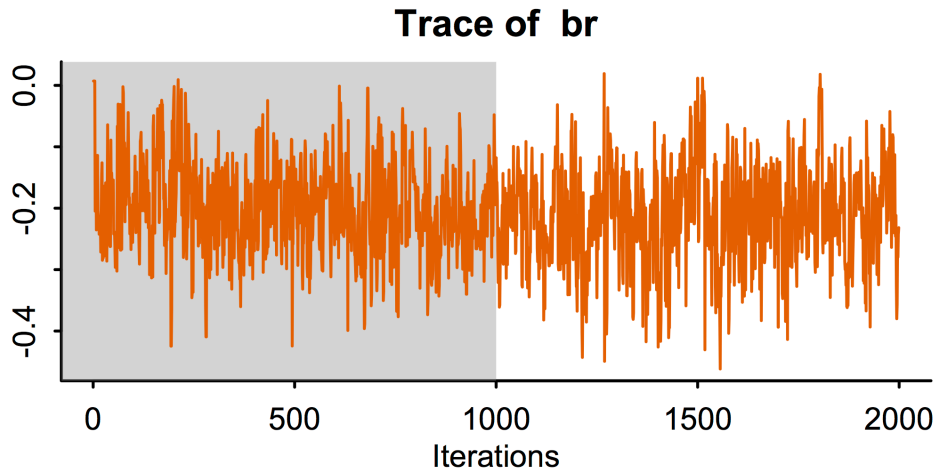
Check the chain

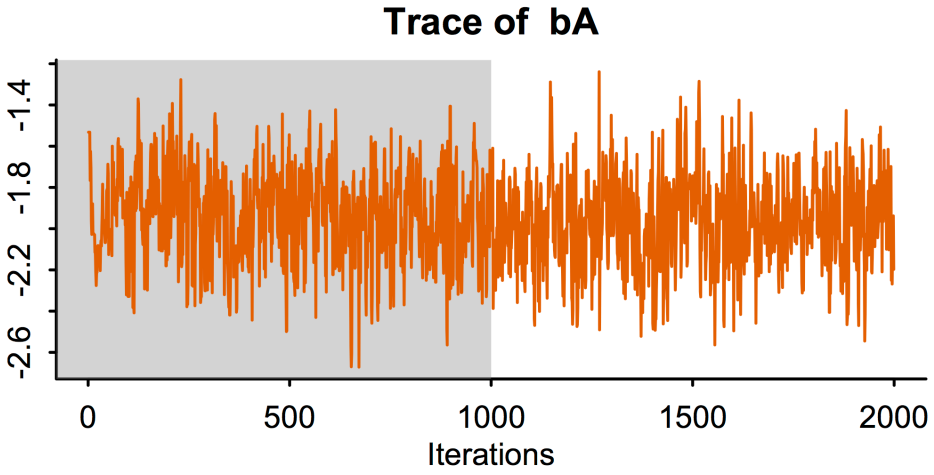
- Sometimes it doesn't work
- First and most important check: trace plot

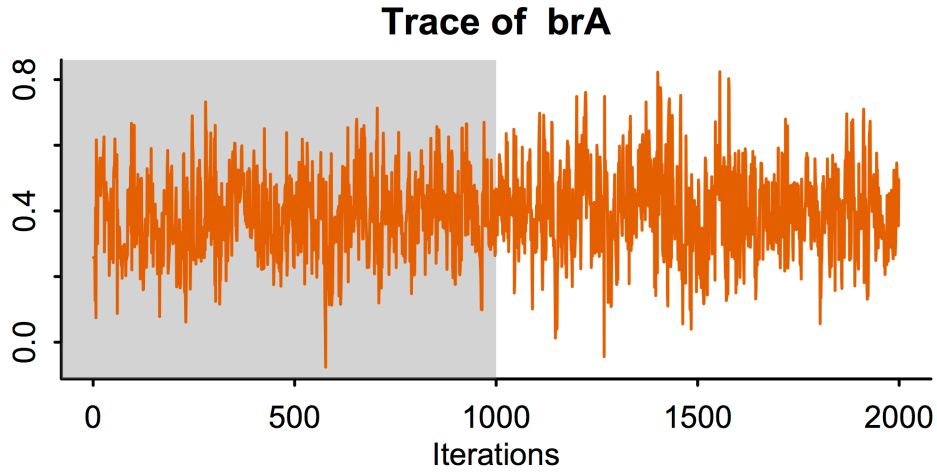
```
plot(m8.1stan)
```

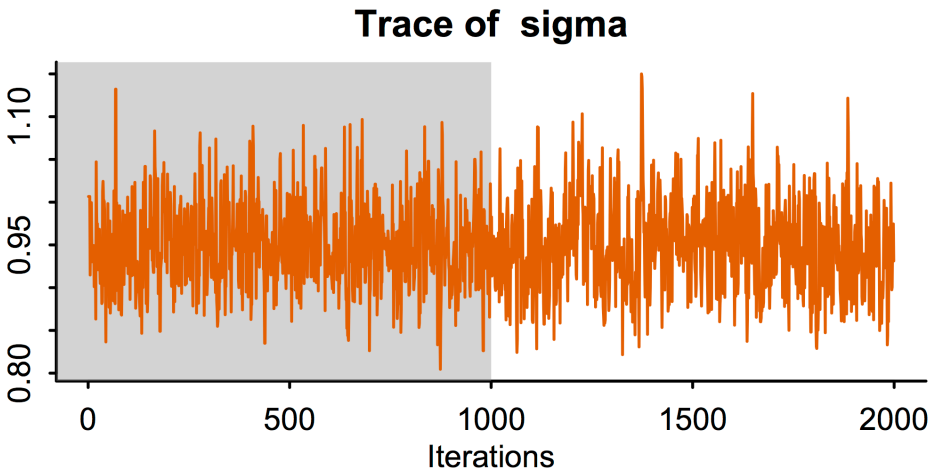






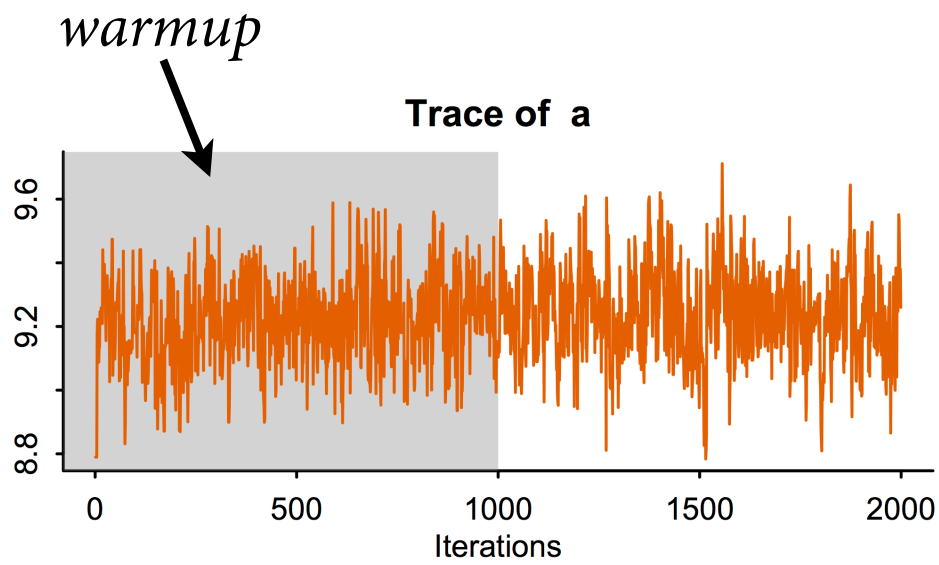






"Hairy caterpillar ocular inspection test"

Warmup



- What is “warmup”?
- Adaptation to posterior for efficient sampling
- Samples during warmup NOT from posterior
- Automatically discarded by precis/summary and other functions
- Warmup is NOT “burn in”

Convergence diagnostics

- n_eff: “effective” number of samples
  - n_eff/n < 0.1, be alarmed
- R-hat –> Run multiple chains!
  - R-hat: crudely, ratio of variance between chains to variance within chains
  - Should approach 1
- Both may mislead

```
precis(m8.1stan)
```

R code 8.6

```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
a      9.24   0.14       9.03       9.47   291    1
bR    -0.21   0.08      -0.32      -0.07   306    1
bA    -1.97   0.23      -2.31      -1.58   351    1
bAR    0.40   0.13       0.20       0.63   350    1
sigma  0.95   0.05       0.86       1.03   566    1
```

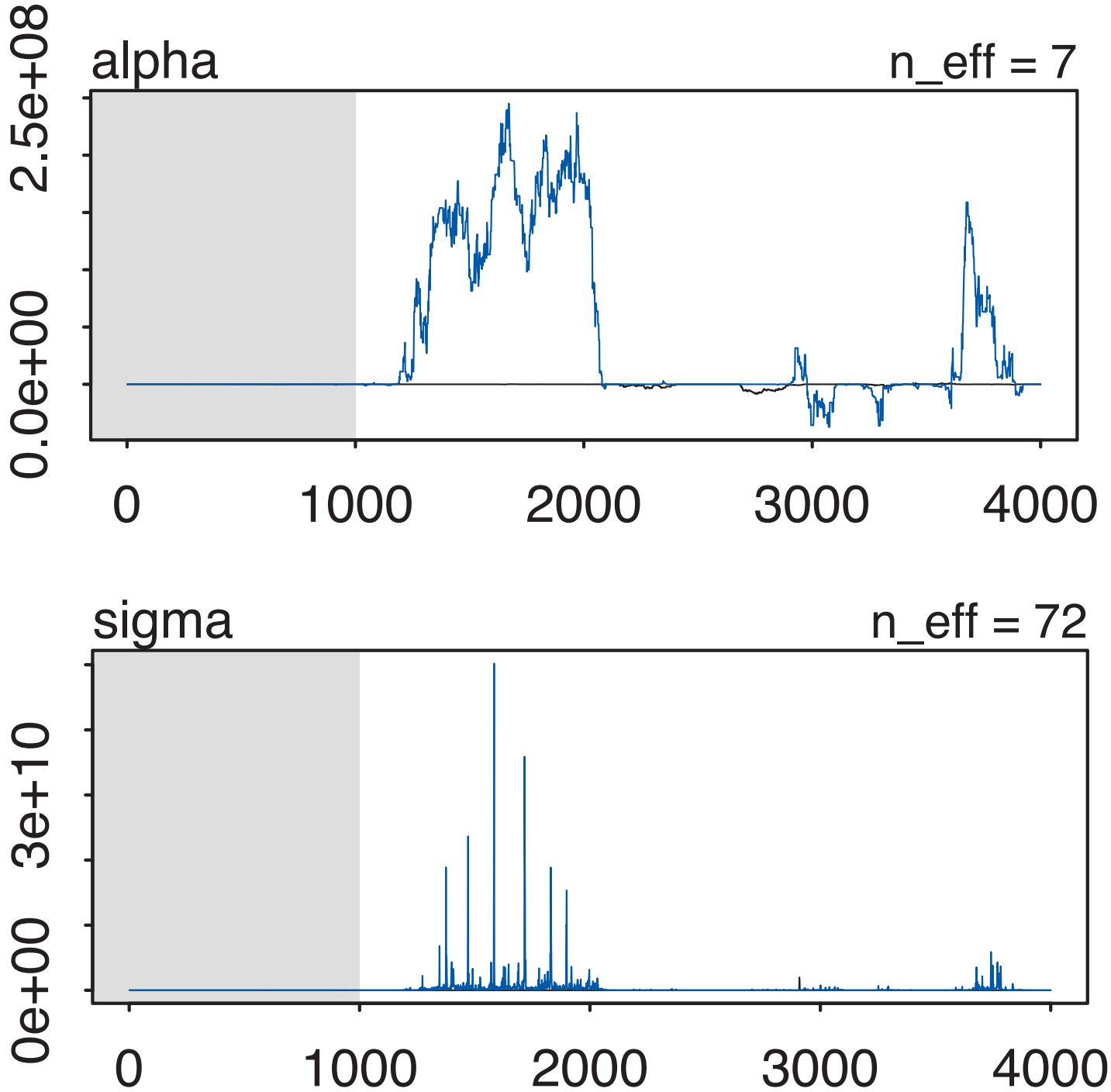
A wild chain

- Two observations: {–1,1}
- Estimate mean and standard deviation

R code 8.13

```
y <- c(-1,1)
m8.2 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- alpha
    ) ,
    data=list(y=y) , start=list(alpha=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
```

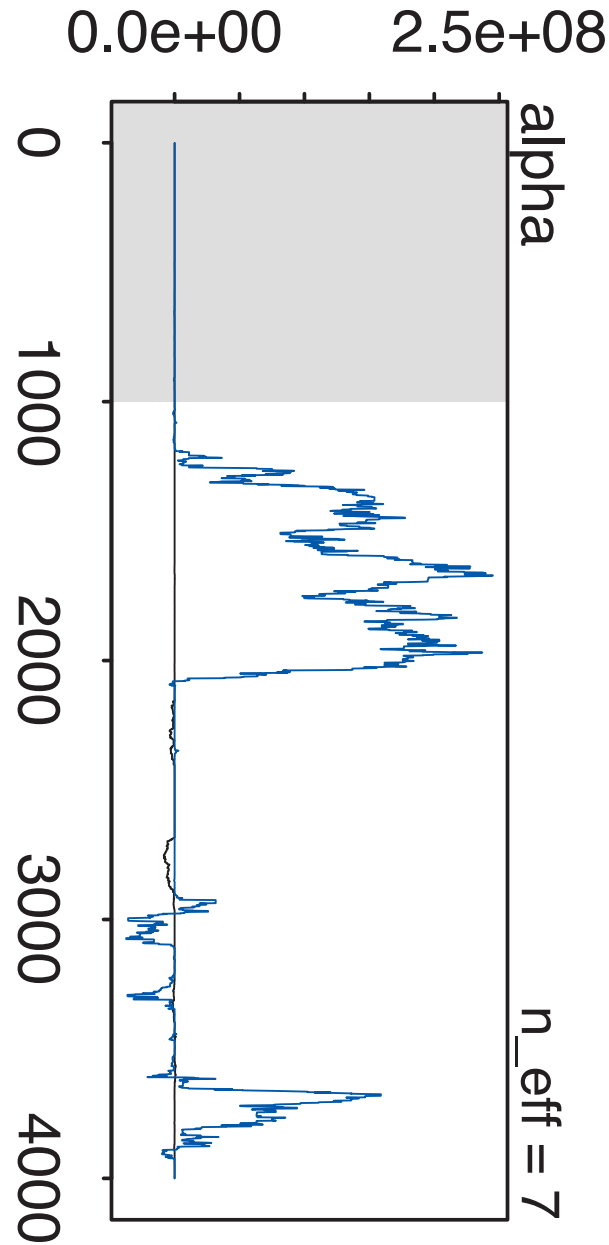
```
             Mean     StdDev   lower 0.89 upper 0.89 n_eff Rhat
alpha  21583691   54448550 -19611287.92  129922812     7 1.36
sigma 139399593 1147514738        29.06  185868167    72 1.02
```


alpha
n_eff = 7
2.5e+08
0.0e+00
0
1000
2000
3000
4000
sigma
n_eff = 72
3e+10
0e+00
0
1000
2000
3000
4000

A wild chain

- Problem is flat priors
  - Flat means flat forever
  - Little information in likelihood
  - Most probability is out to 30-million
  - King Monty's car keeps driving
  - Flat prior is *improper* (integrates to infinity)
- Fix with weakly informative priors

$$y_i \sim \text{Normal}(\mu, \sigma)$$
$$\mu = \alpha$$
$$\alpha \sim \text{Normal}(1, 10)$$
$$\sigma \sim \text{HalfCauchy}(0, 1)$$

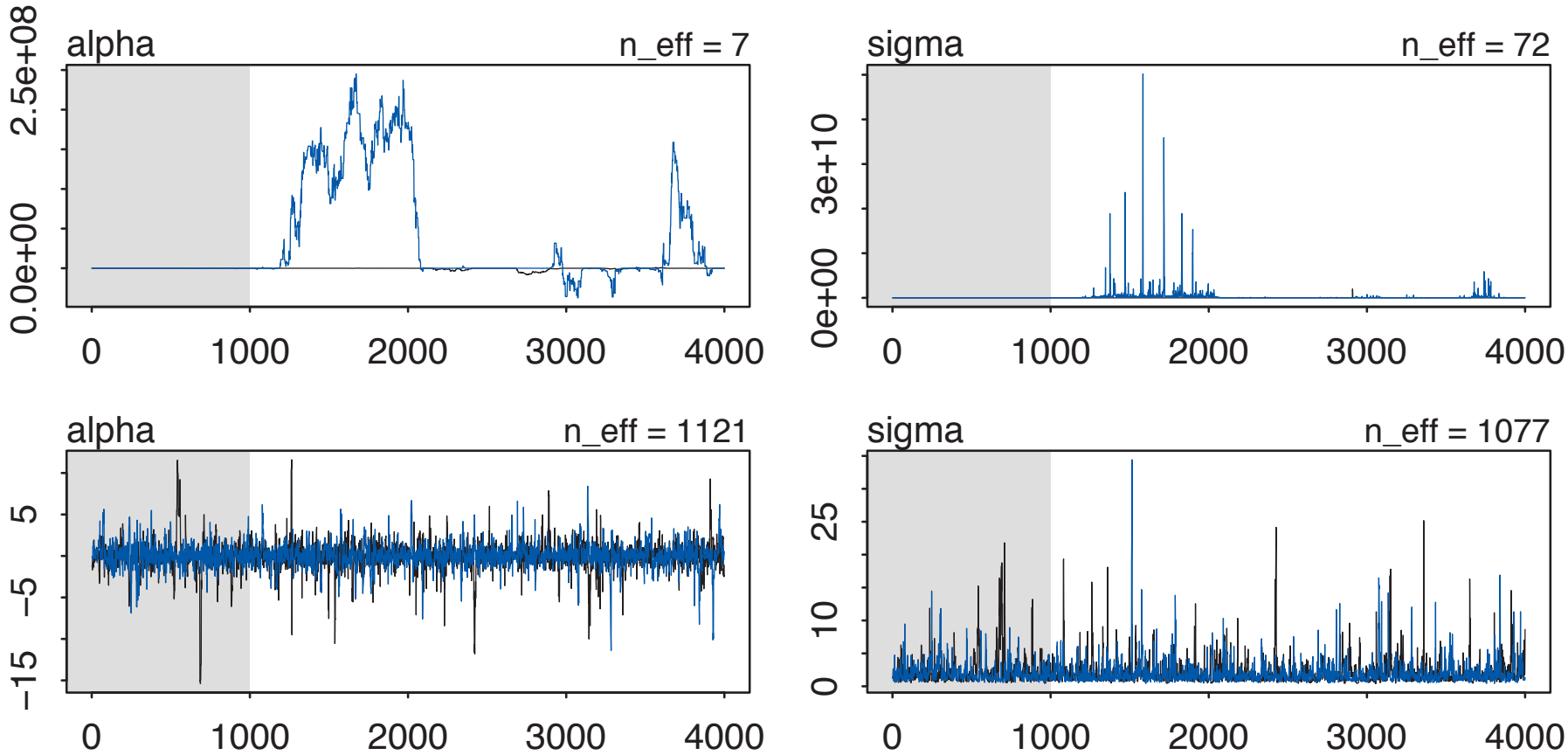


A wild chain

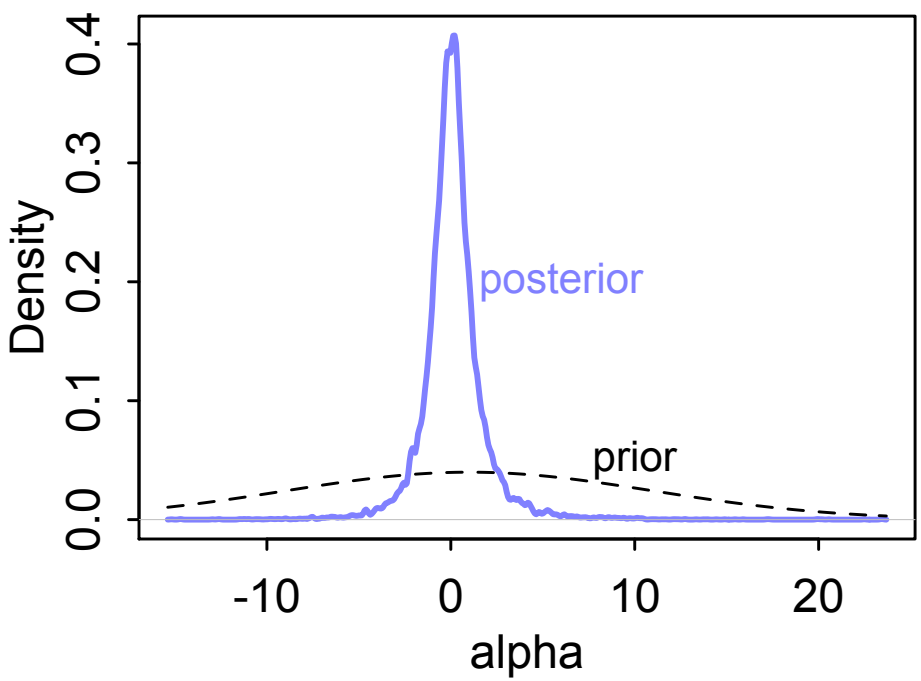
R code 8.15

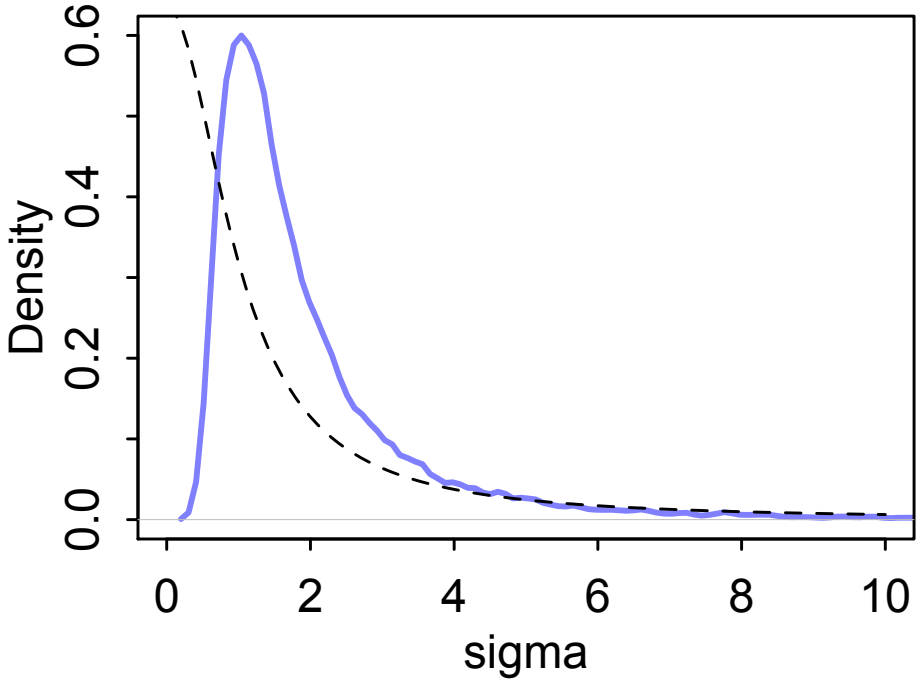
```
m8.3 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- alpha ,
        alpha ~ dnorm( 1 , 10 ) ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=list(y=y) , start=list(alpha=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
precis(m8.3)
```

```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
alpha -0.01   1.60      -1.98       2.37  1121    1
sigma  1.98   1.91       0.47       3.45  1077    1
```


alpha
n_eff = 7
0.0e+00
2.5e+08
0
1000
2000
3000
4000
sigma
n_eff = 72
0e+00
3e+10
0
1000
2000
3000
4000
alpha
n_eff = 1121
−15
−5
5
0
1000
2000
3000
4000
sigma
n_eff = 1077
0
10
25
0
1000
2000
3000
4000

A wild chain





Even with only 2 observations, these priors have no effect on inference! Except to allow you to make inferences...

$$y_i \sim \text{Normal}(\mu, \sigma)$$
$$\mu = \alpha$$
$$\alpha \sim \text{Normal}(1, 10)$$
$$\sigma \sim \text{HalfCauchy}(0, 1)$$

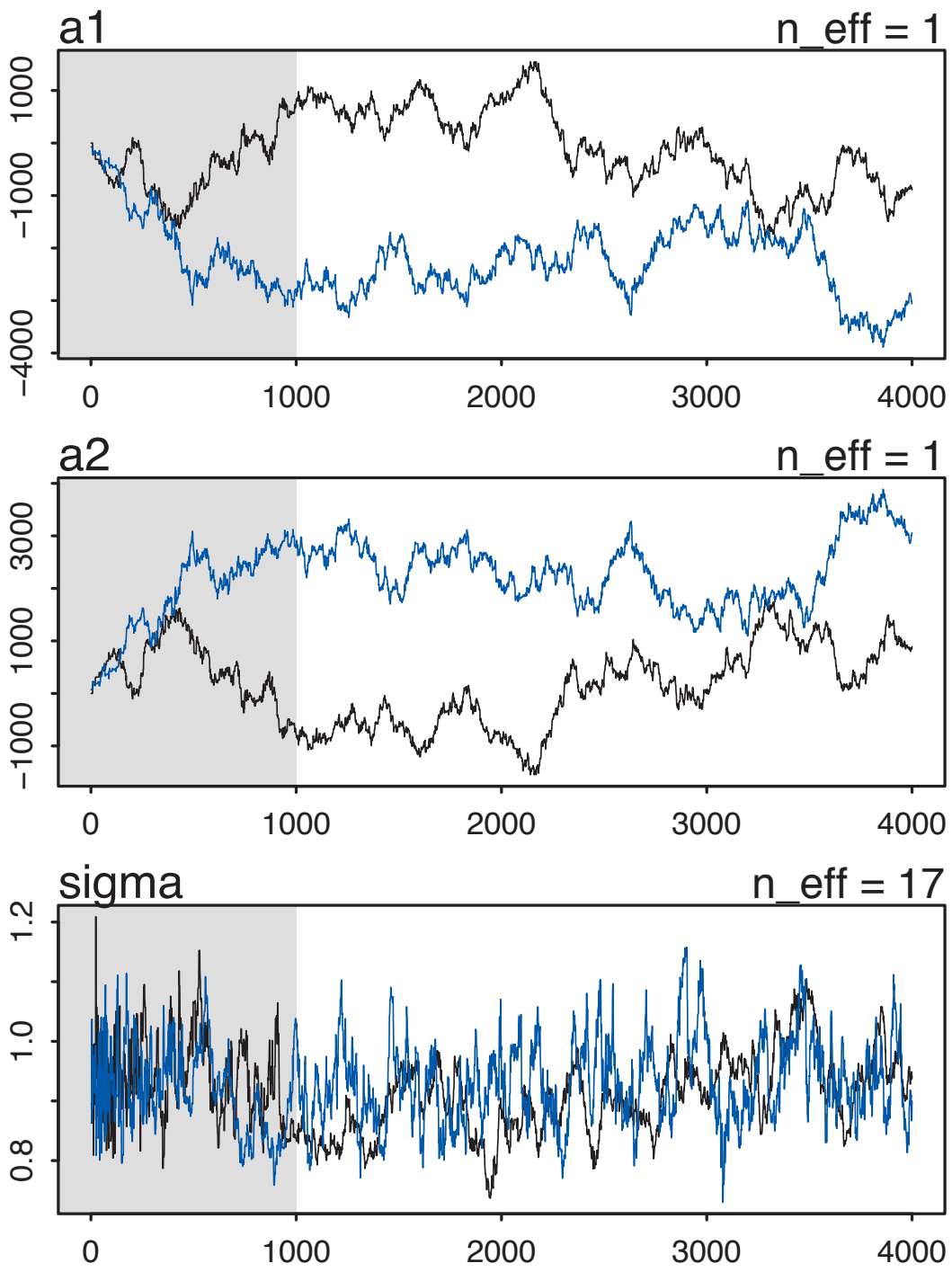
Unidentified

$$y_i \sim \text{Normal}(\mu, \sigma)$$
$$\mu = \alpha_1 + \alpha_2$$
$$\sigma \sim \text{HalfCauchy}(0, 1)$$

```
y <- rnorm( 100 , mean=0 , sd=1 )
```

```
m8.4 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- a1 + a2 ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=list(y=y) , start=list(a1=0,a2=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
precis(m8.4)
```

```
          Mean  StdDev lower 0.89 upper 0.89 n_eff Rhat
a1    -1194.76 1344.19   -2928.62    1053.52     1 2.83
a2     1194.81 1344.19   -1054.86    2927.39     1 2.83
sigma     0.92    0.07       0.81       1.02    17 1.13
```


a1
n_eff = 1
1000
−1000
−4000
0
1000
2000
3000
4000
a2
n_eff = 1
3000
1000
−1000
0
1000
2000
3000
4000
sigma
n_eff = 17
1.2
1.0
0.8
0
1000
2000
3000
4000

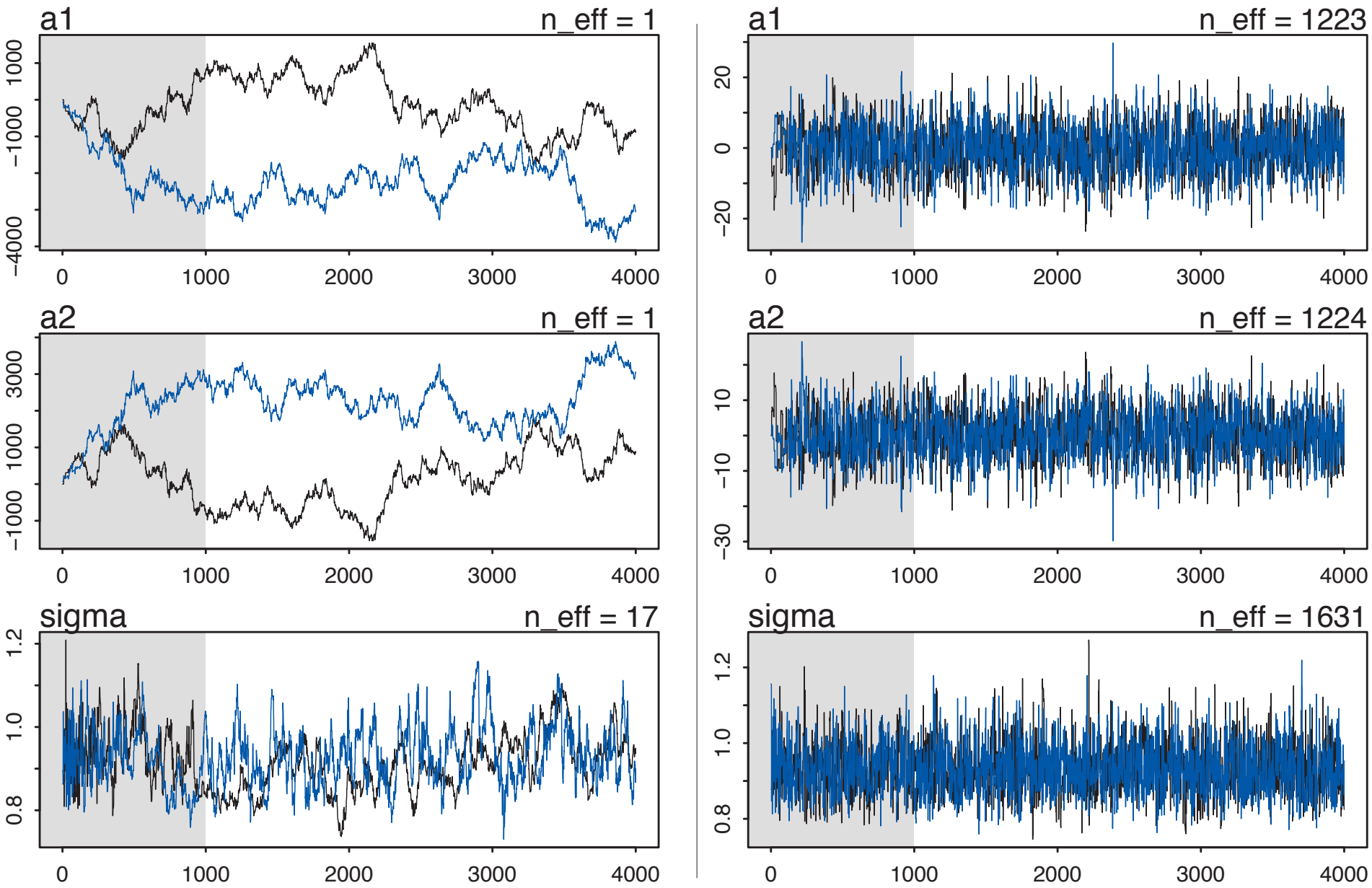
Unidentified

- Use weak priors, again

R code 8.19

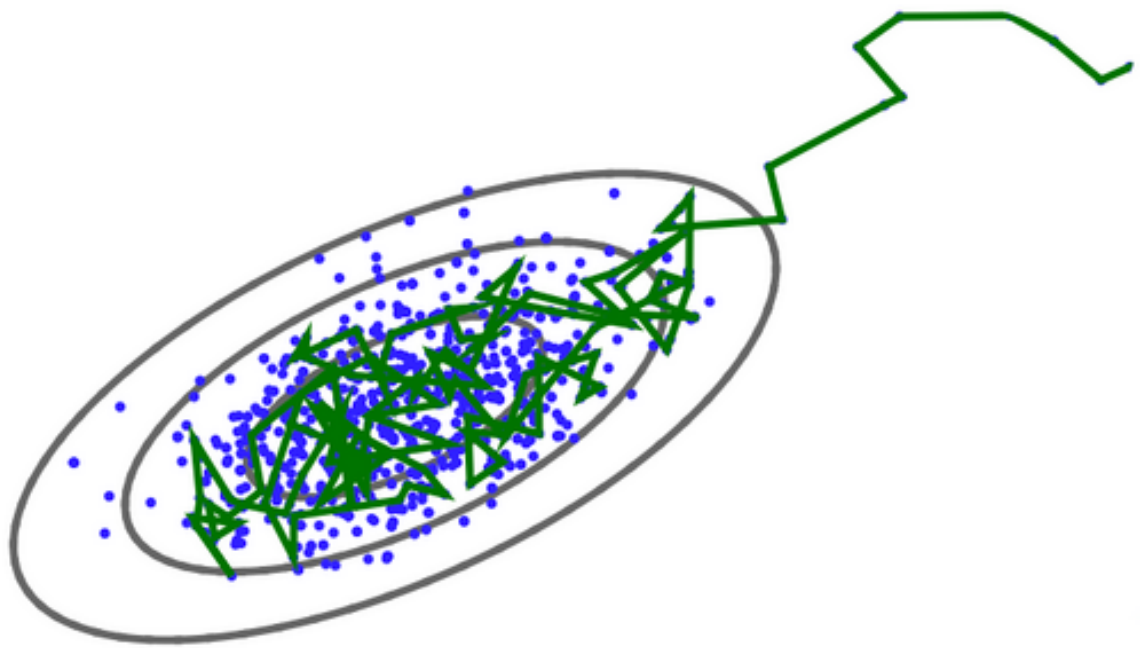
```
m8.5 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- a1 + a2 ,
        a1 ~ dnorm( 0 , 10 ) ,
        a2 ~ dnorm( 0 , 10 ) ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=list(y=y) , start=list(a1=0,a2=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
precis(m8.5)
```

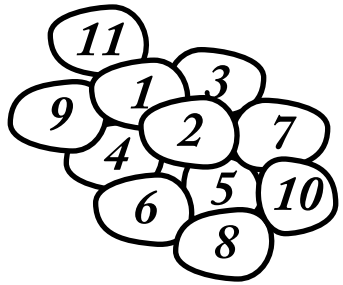
```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
a1    -0.23   6.97     -11.25      10.95  1223    1
a2     0.28   6.97     -10.85      11.36  1224    1
sigma  0.93   0.07       0.82       1.04  1631    1
```


a1
n_eff = 1
1000
−1000
−4000
0
1000
2000
3000
4000
a2
n_eff = 1
3000
1000
−1000
sigma
n_eff = 17
1.2
1.0
0.8
a1
n_eff = 1223
20
0
−20
a2
n_eff = 1224
10
−10
−30
sigma
n_eff = 1631

Homework

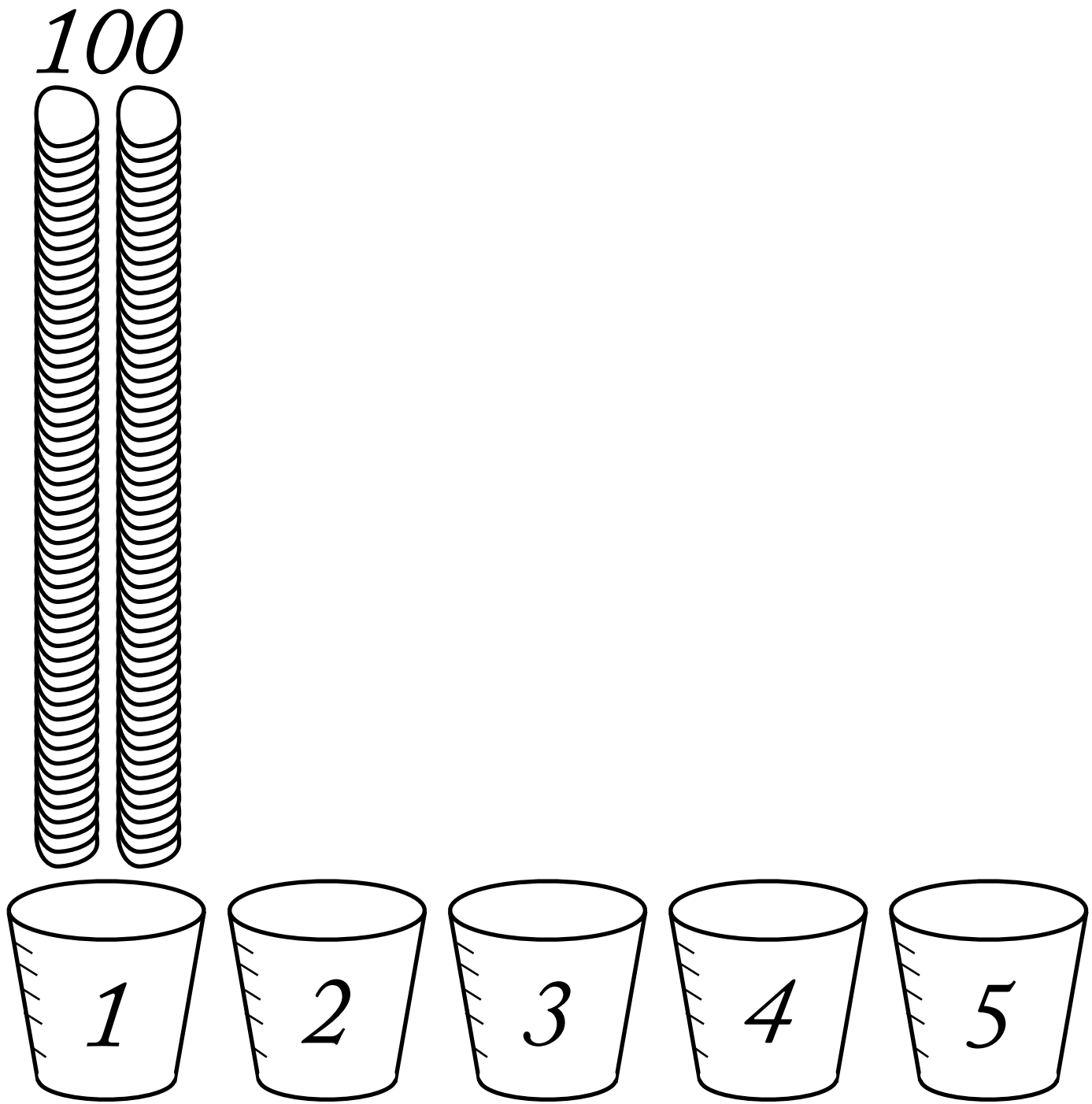
- Problems 8H1, 8H2, 8H3
- Next week: Generalized Linear Models (GLMs), Chapters 9, 10, 11
- Next next week: Holiday break, resume in 2018

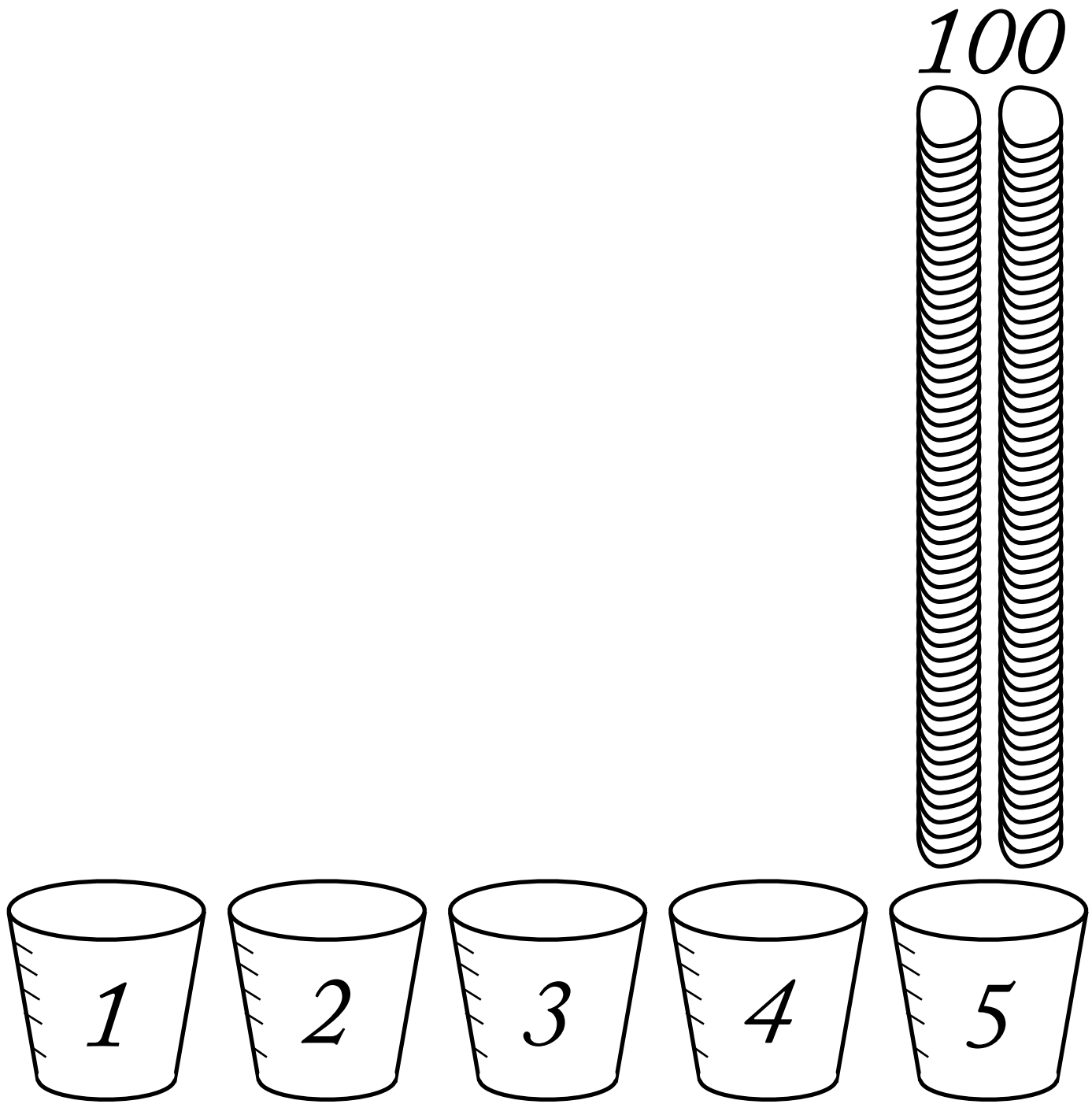


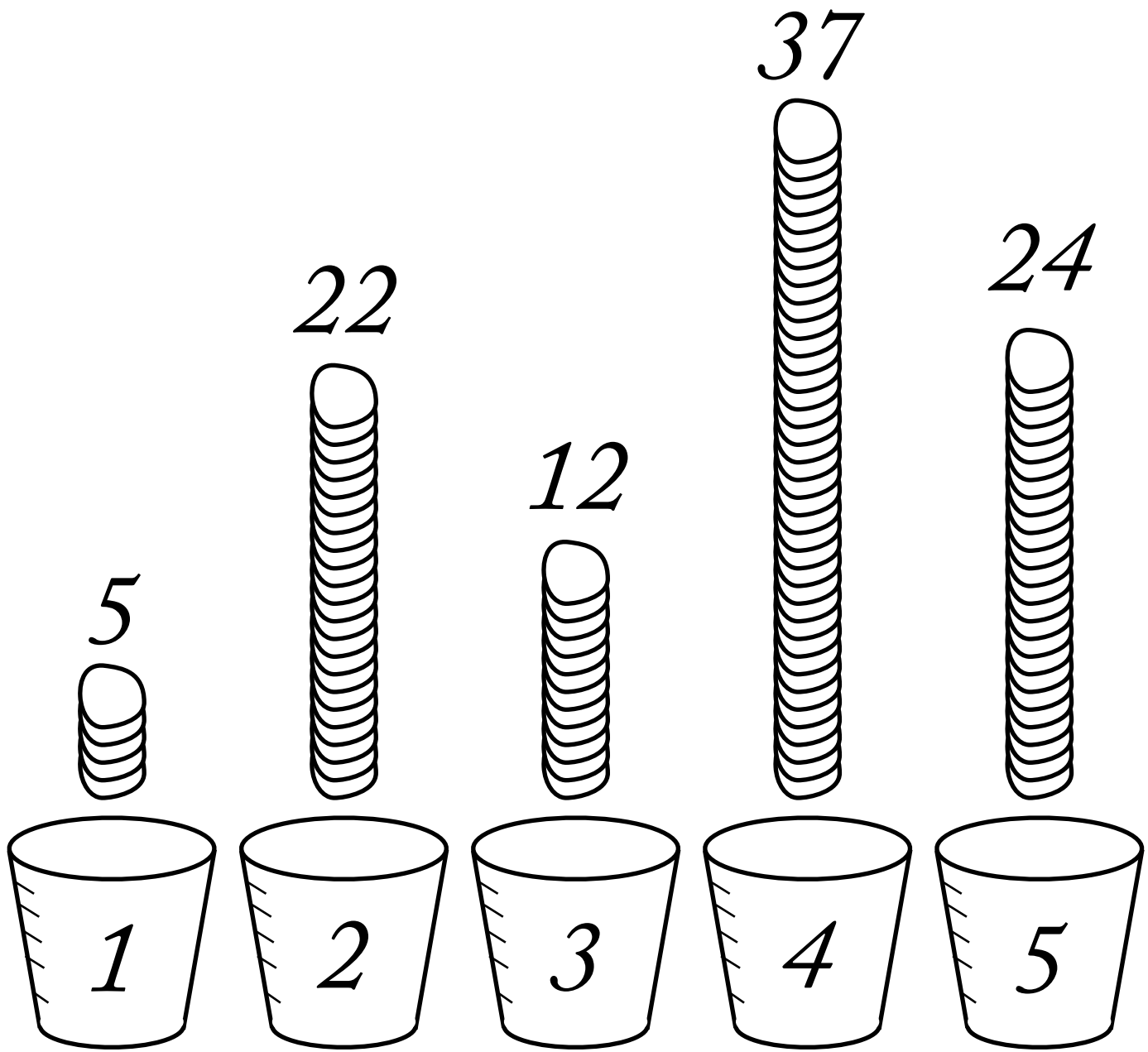


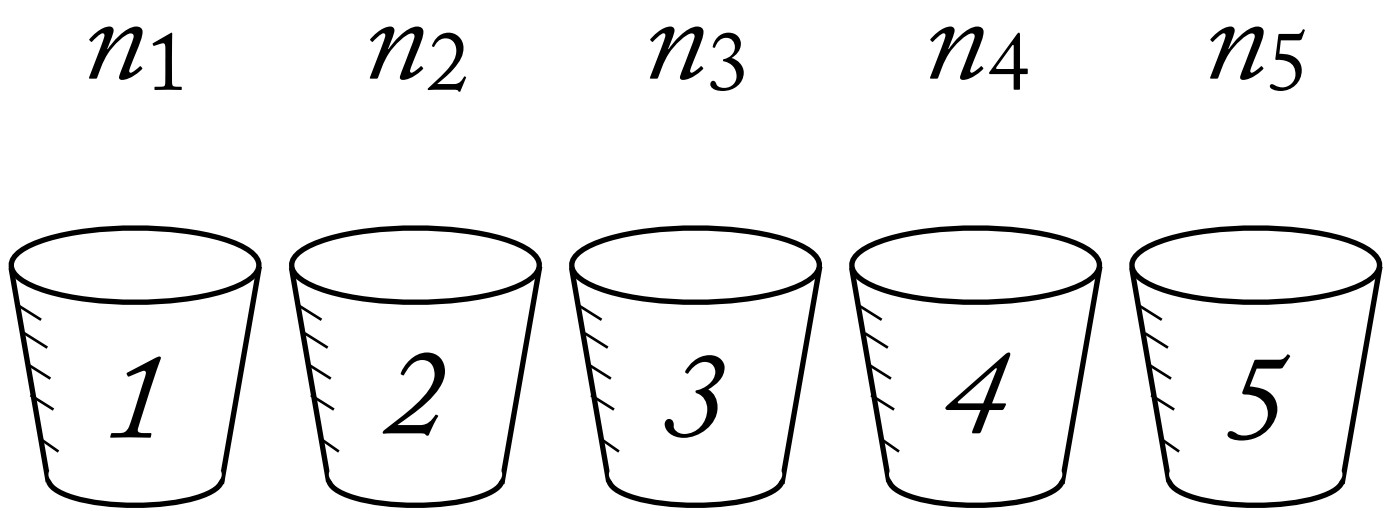
100 pebbles


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

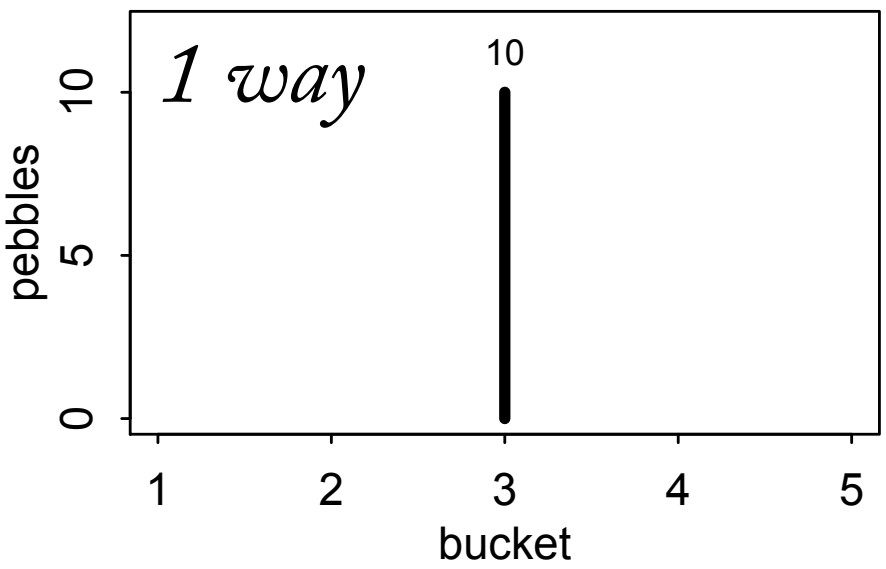

n_1
n_2
n_3
n_4
n_5
1
2
3
4
5

Number of ways:

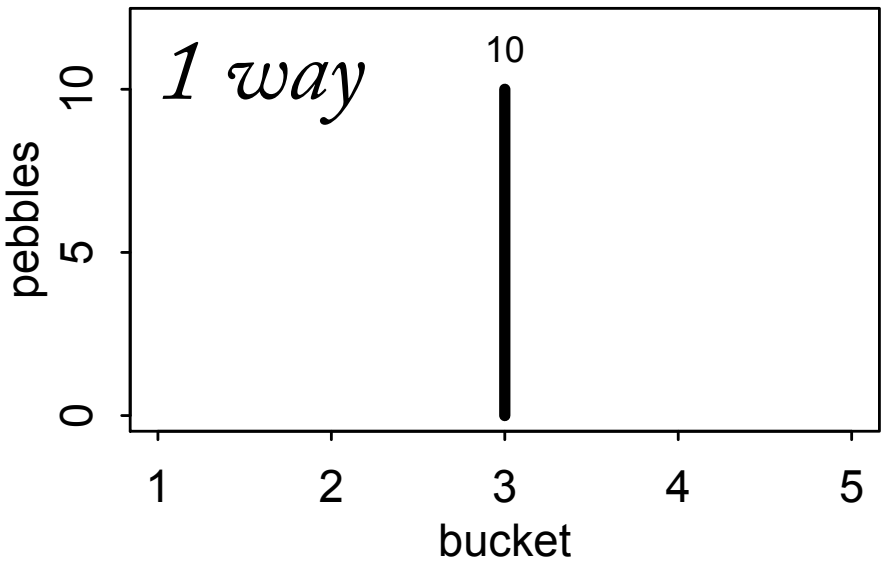
$$W = \frac{N!}{n_1! n_2! n_3! n_4! n_5!}$$

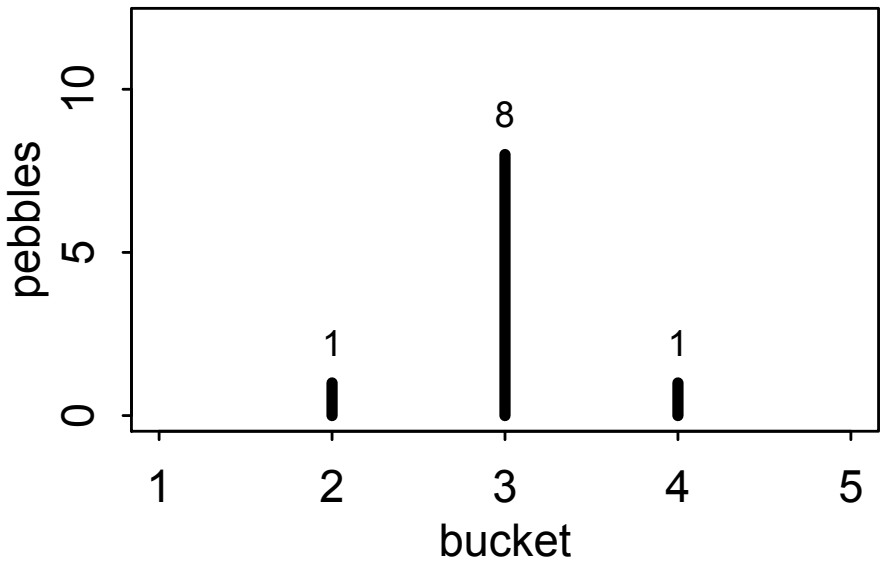
n_1 n_2 n_3 n_4 n_5

Suppose only 10 pebbles....

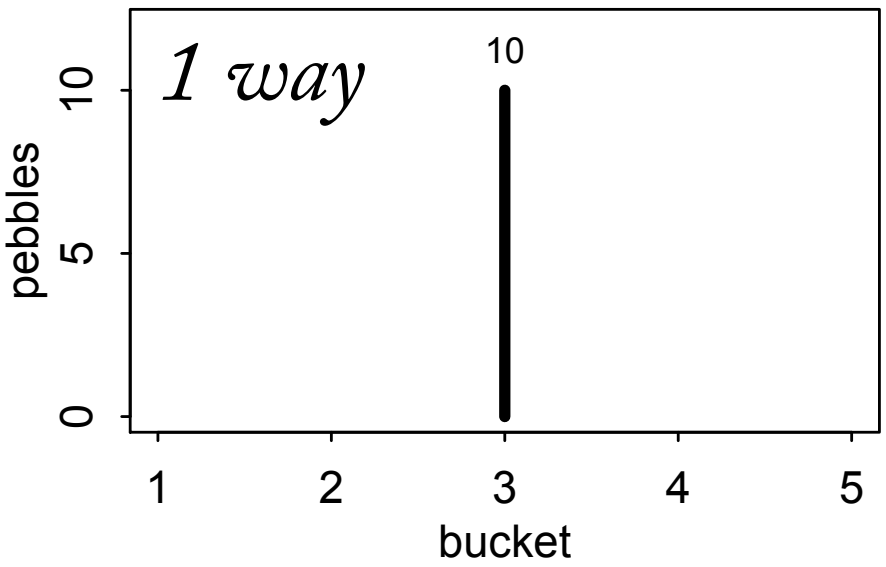


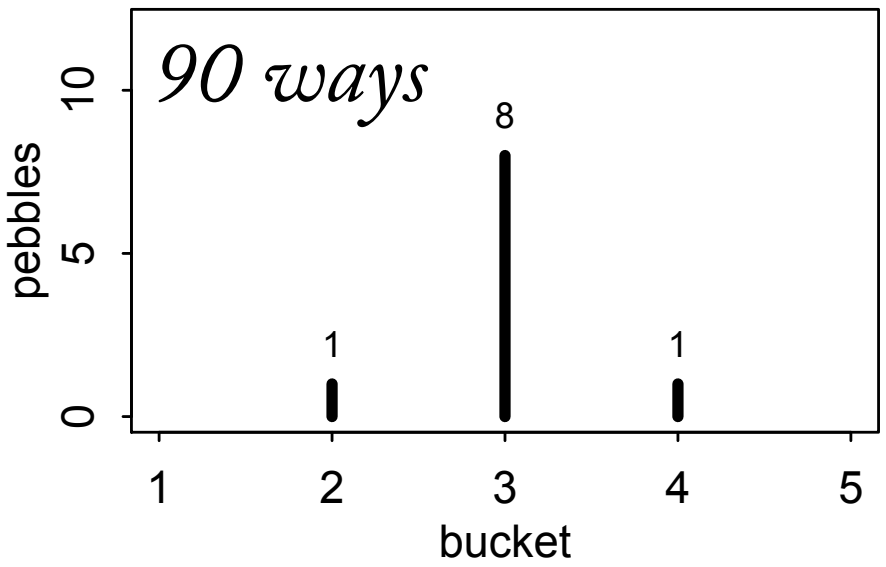
Suppose only 10 pebbles....



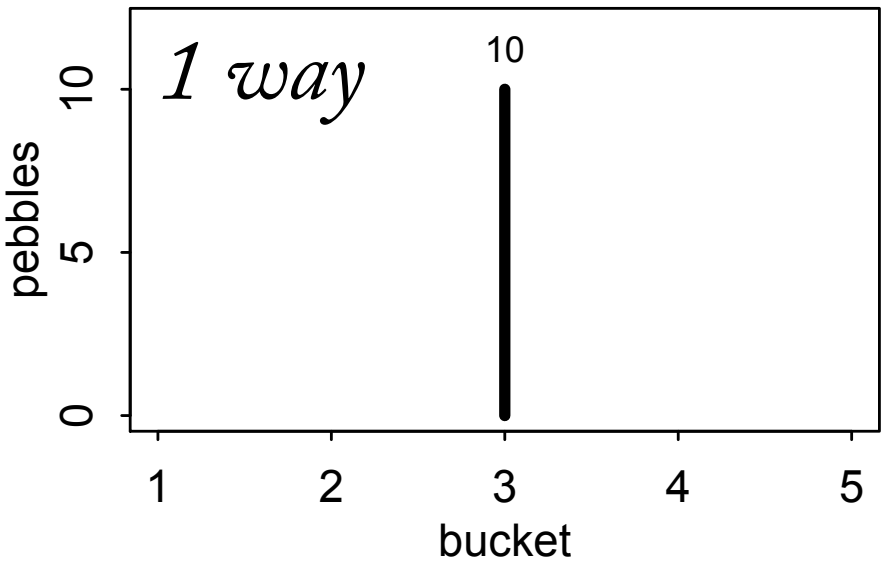


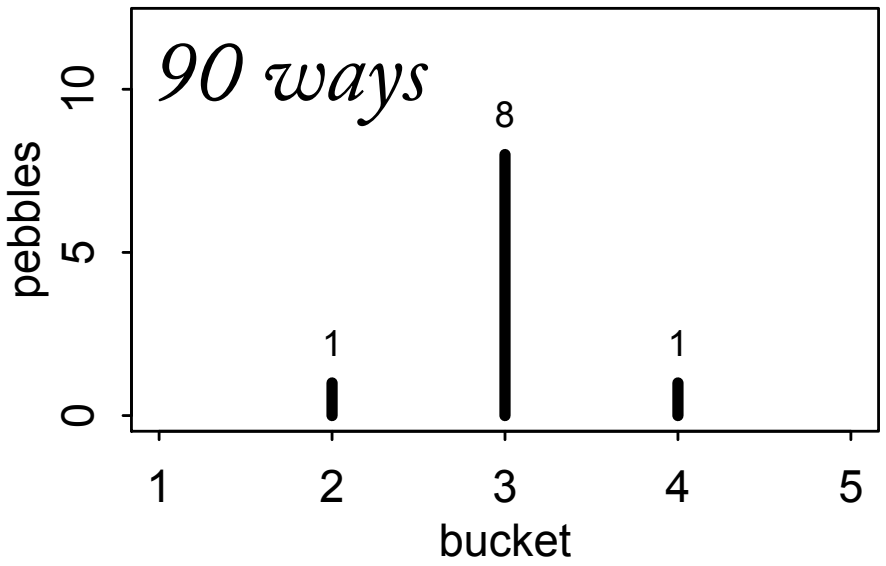
Suppose only 10 pebbles....

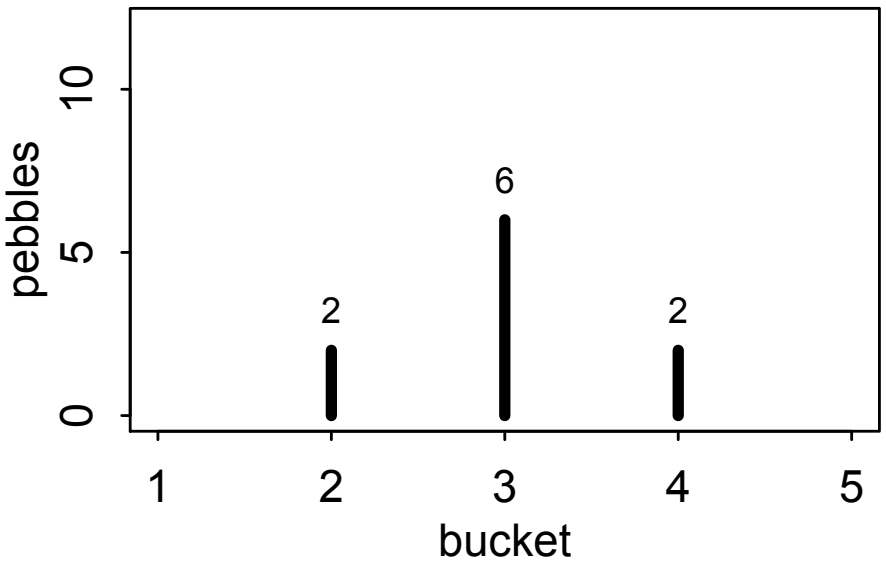




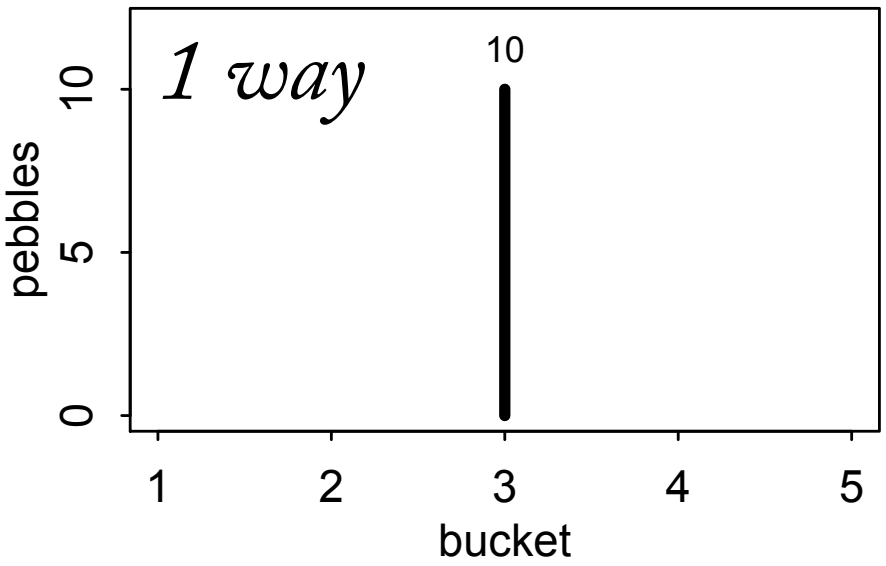
Suppose only 10 pebbles....

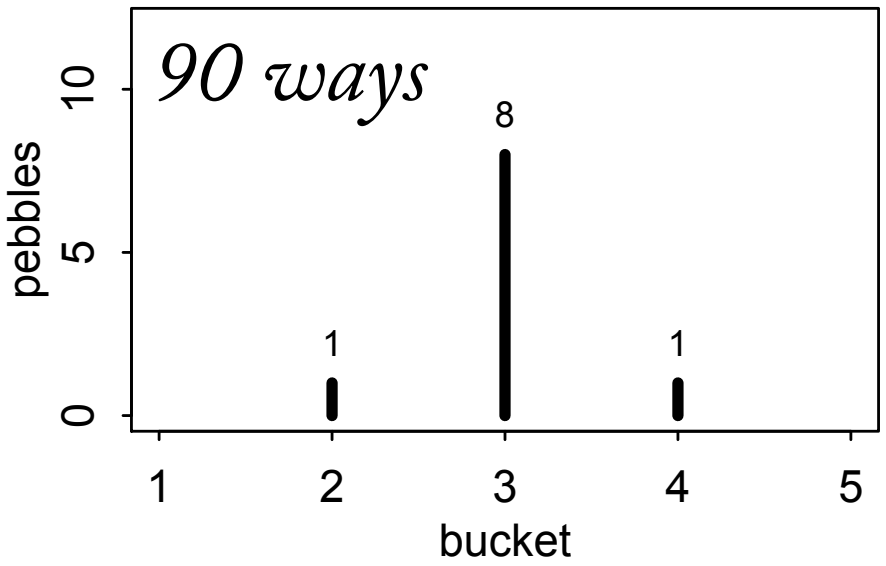


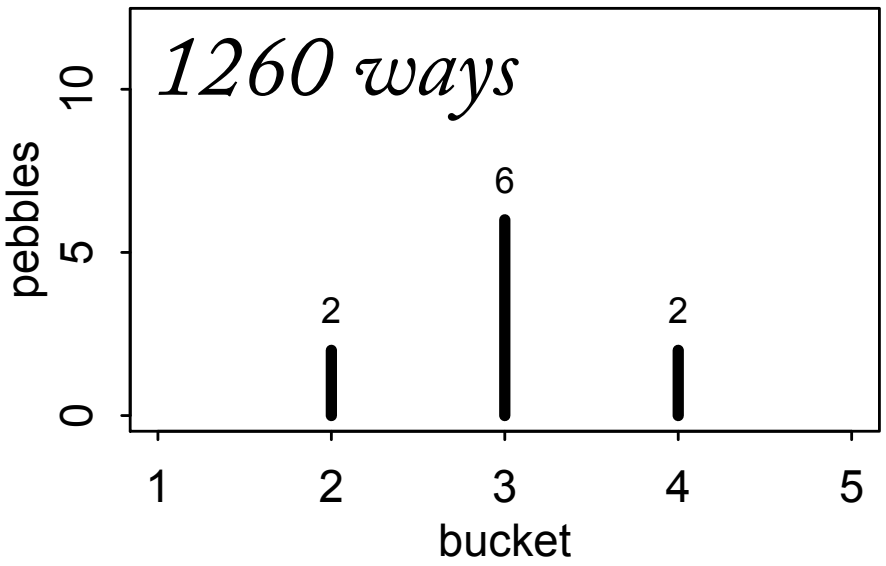




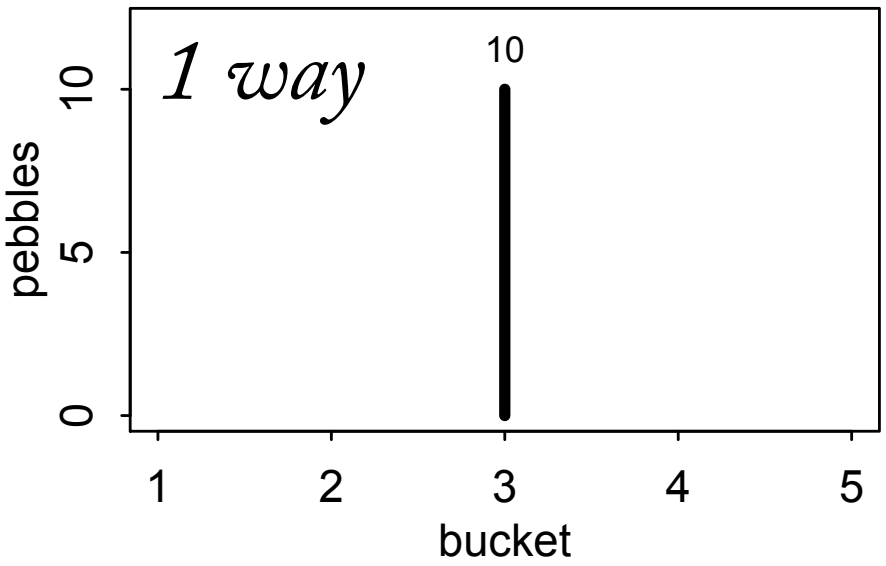
Suppose only 10 pebbles....

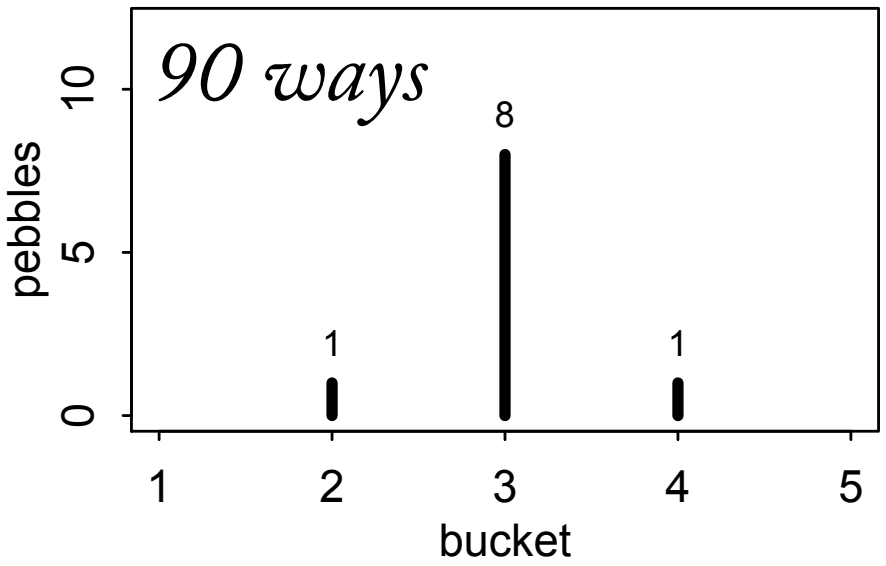


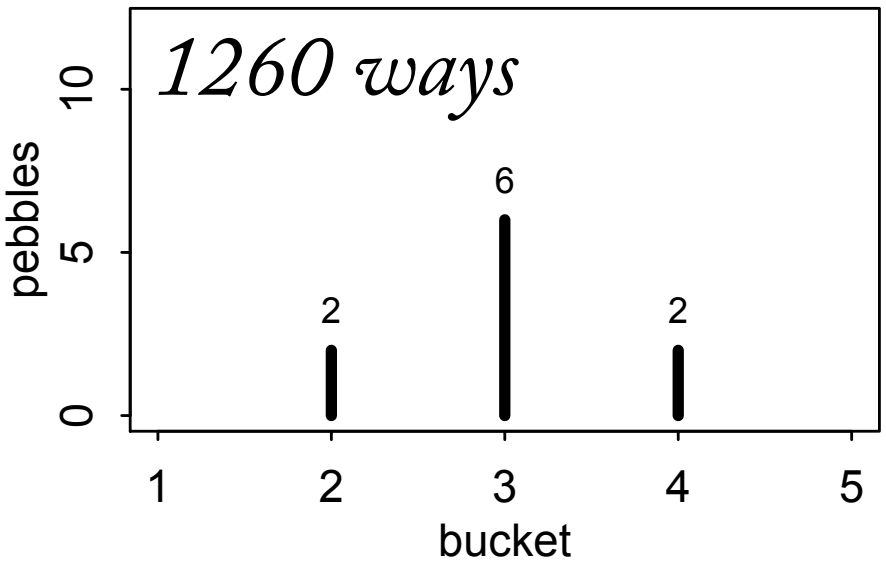


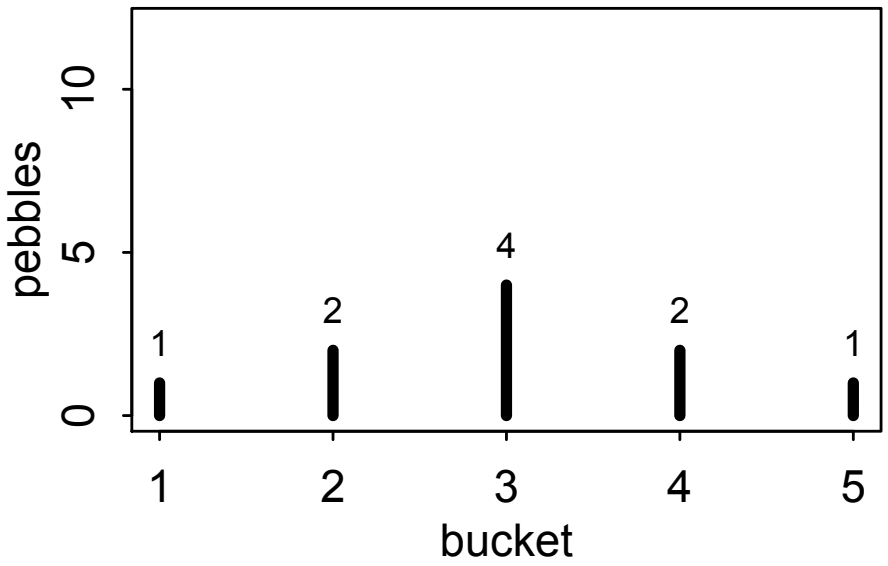


Suppose only 10 pebbles....

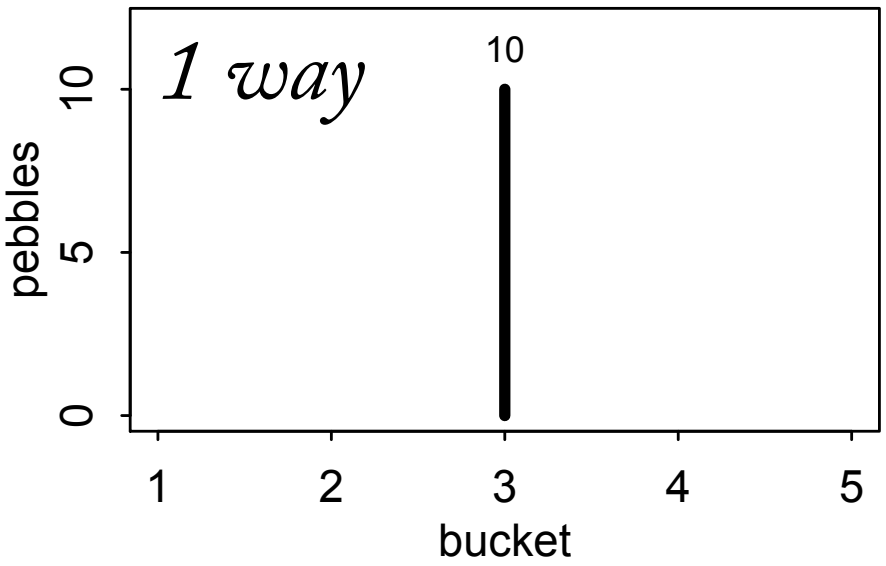


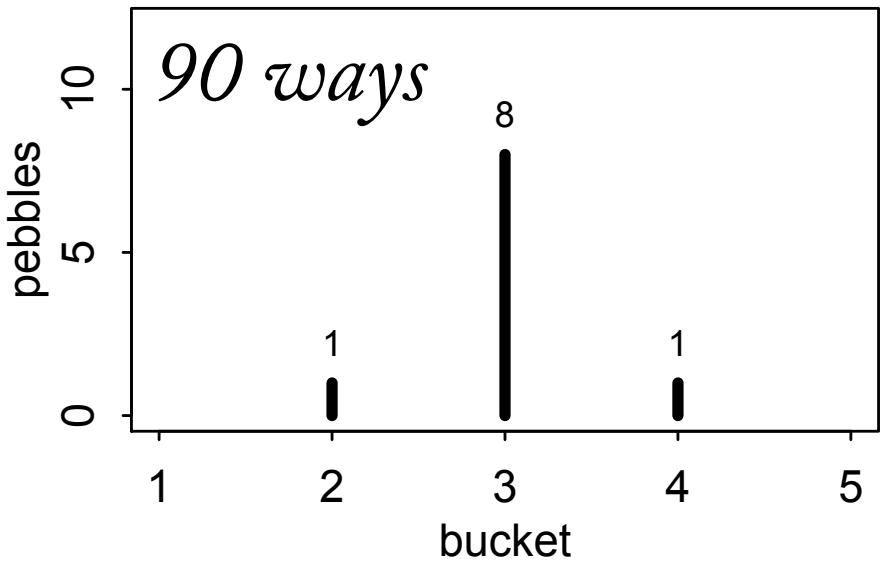


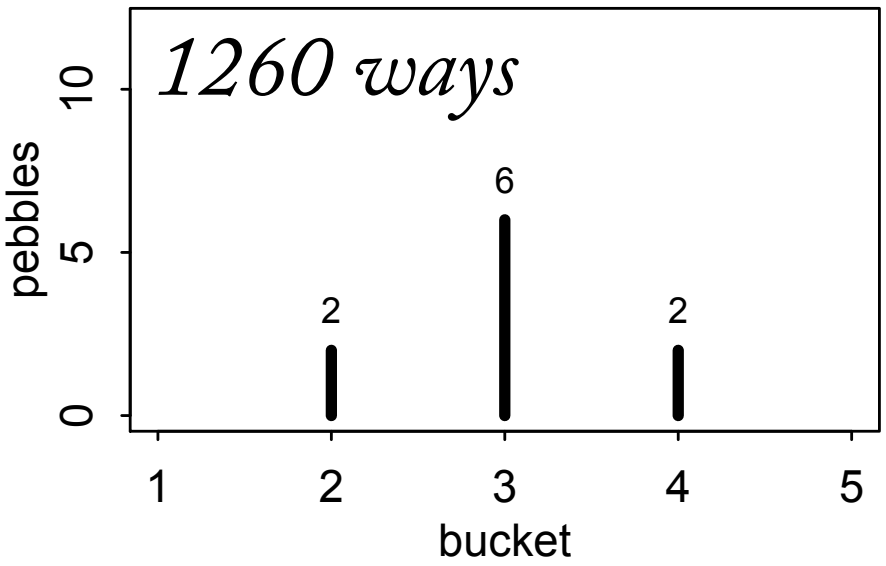


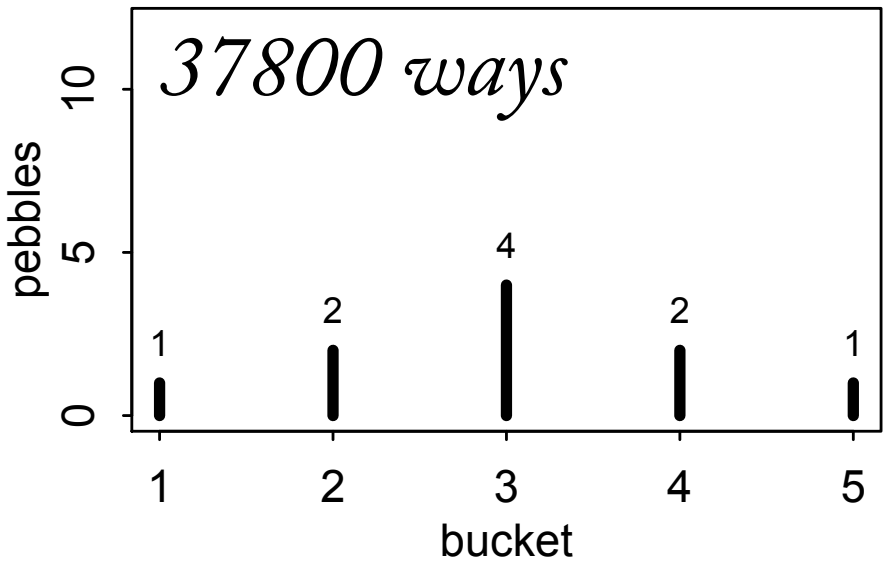


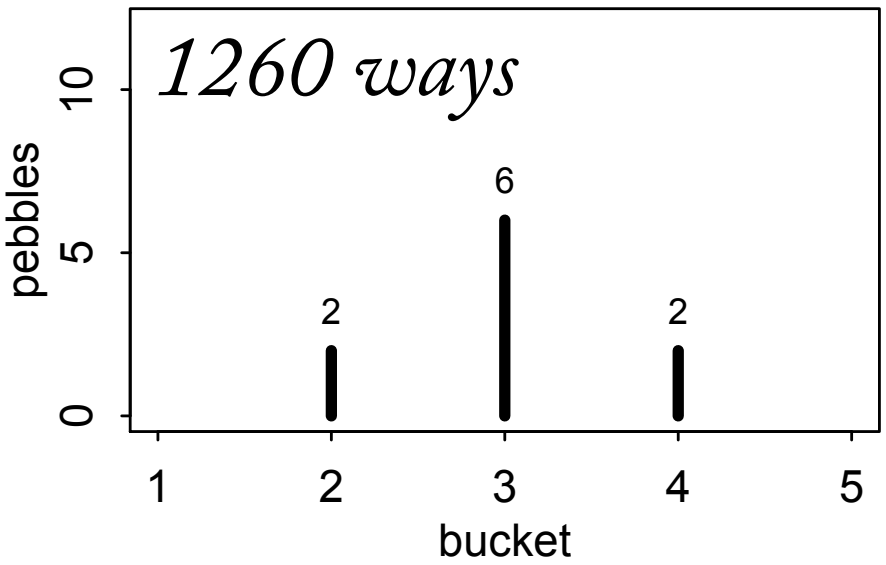
Suppose only 10 pebbles....

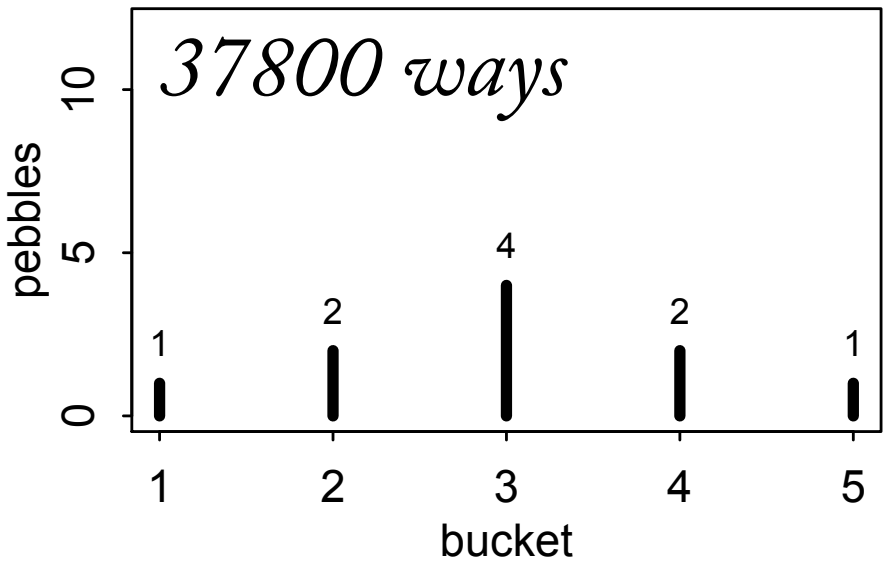


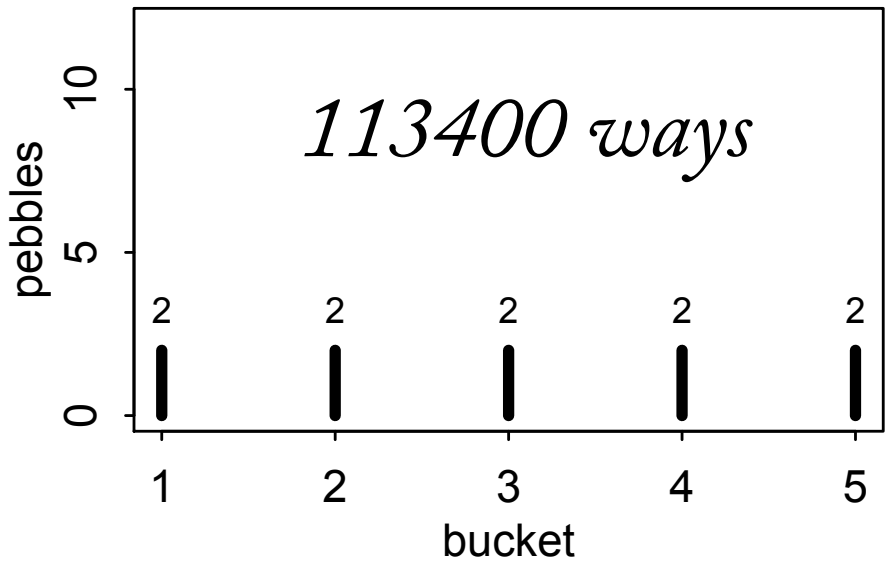






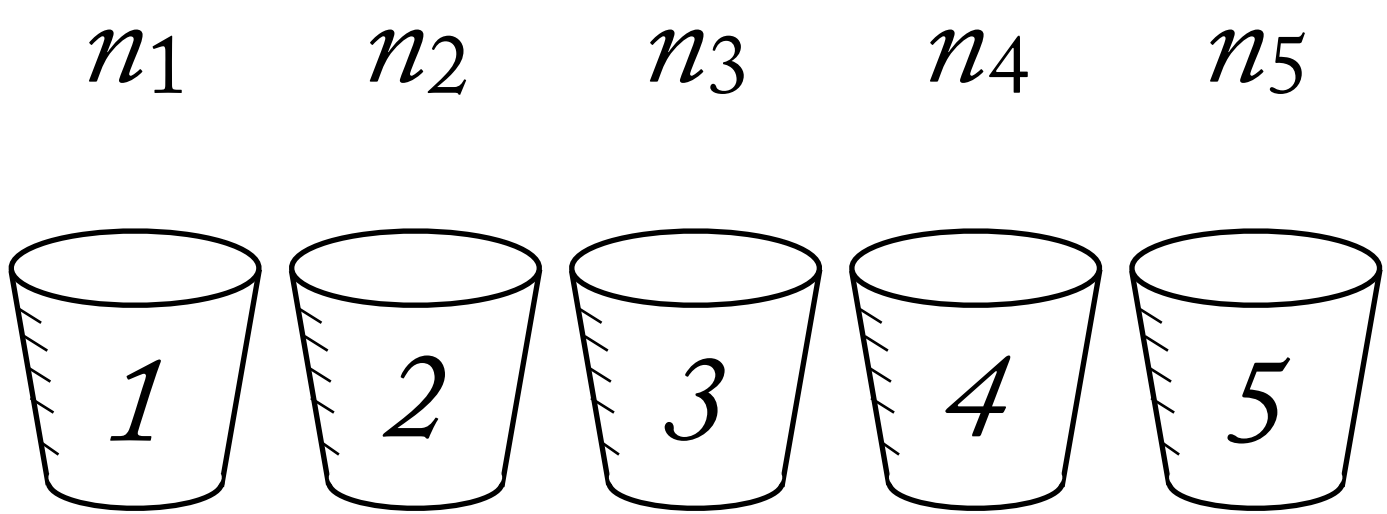

1260 ways
pebbles
0
5
10
2
6
2
1
2
3
4
5
bucket


37800 ways
pebbles
0
5
10
1
2
4
2
1
1
2
3
4
5
bucket


113400 ways
pebbles
0
5
10
2
2
2
2
2
1
2
3
4
5
bucket

For large N:

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



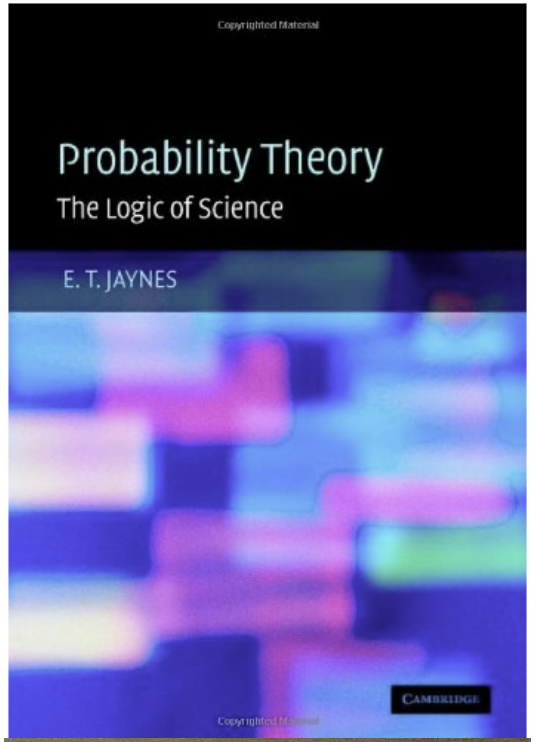
For large N:

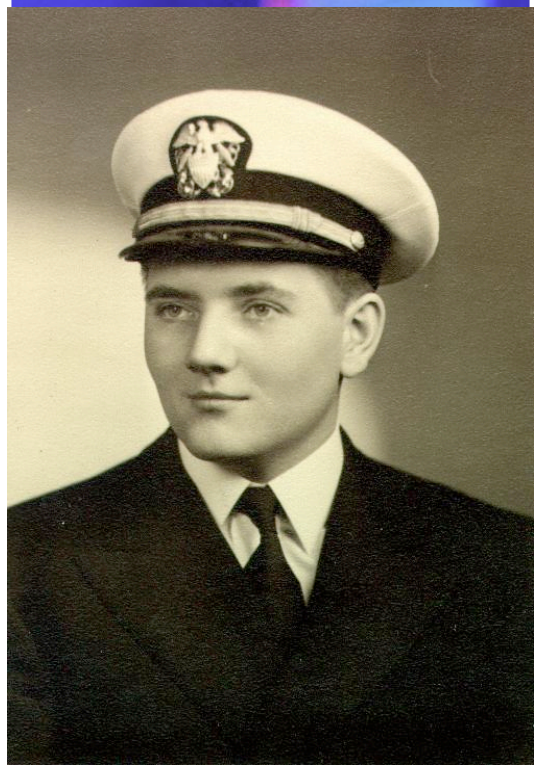
$$\frac{1}{N}\log W \approx -\sum_i \frac{n_i}{N}\log\left(\frac{n_i}{N}\right) = -\sum_i p_i \log p_i$$

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*

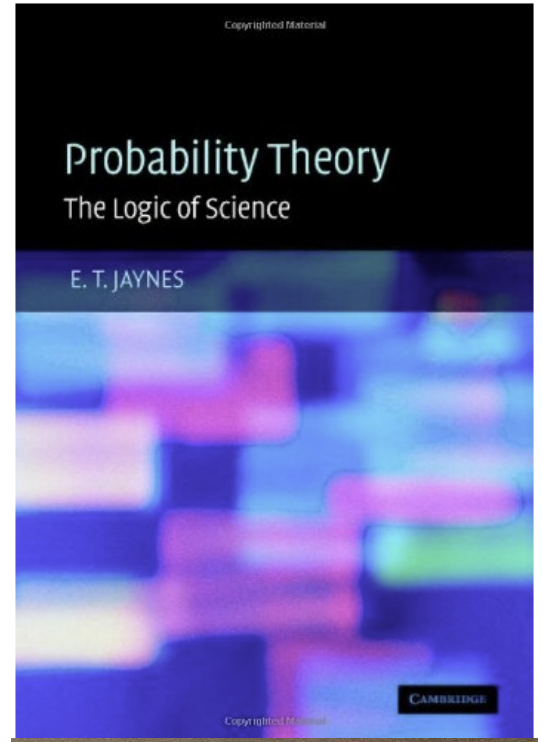


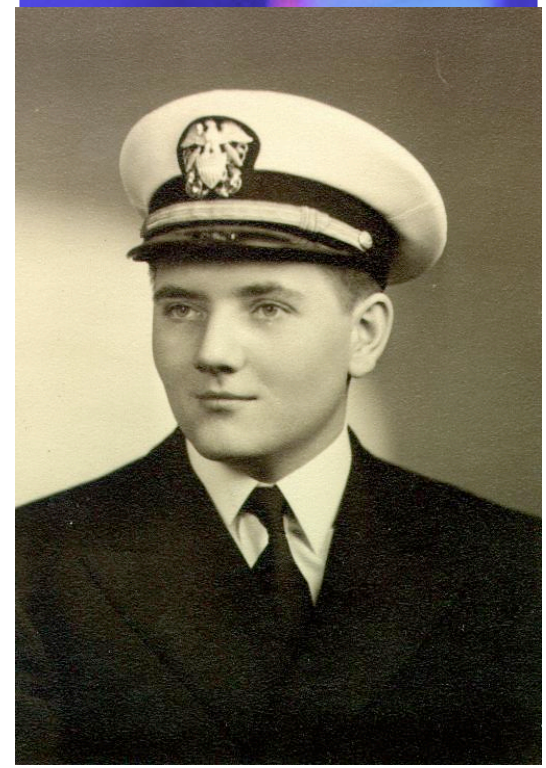


E. T. Jaynes (1922–1998)

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 construct priors
- For observations, way to construct likelihood
- Also reproduces Bayesian updating as special case (*minimum cross-entropy*)





E. T. Jaynes (1922–1998)

Maximum entropy

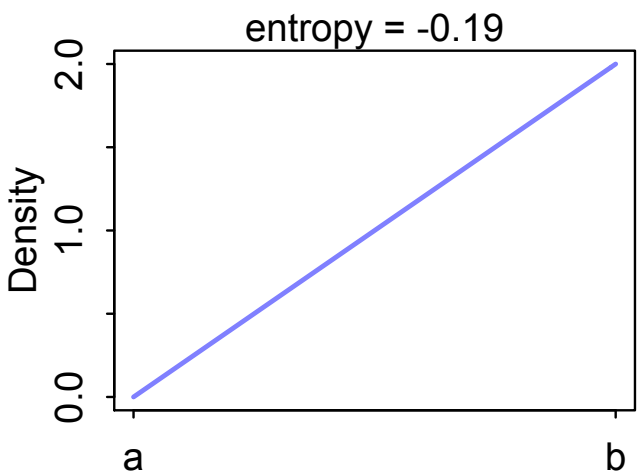
- Ye olde information entropy:

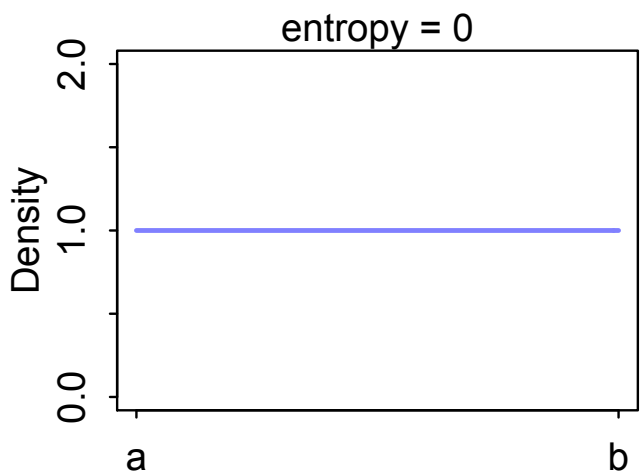
$$H(p) = -\sum_i p_i \log p_i$$

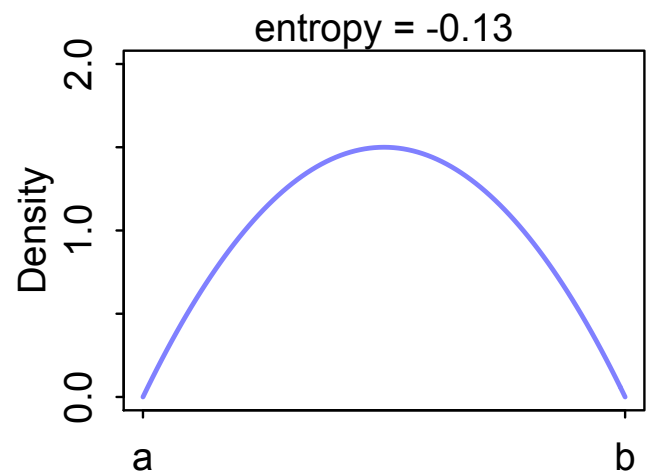
- Q: What kind of distribution maximizes this quantity?
- A: Flattest distribution still consistent with **constraints**. This is the distribution that can happen the most unique ways.
- Whatever does happen, bound to be one of those ways.

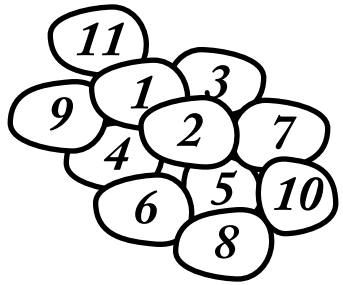
Uniform distribution

- Constraints: bounded between *a* and *b*
- Maxent distribution is uniform, because flattest
- What if there are other constraints, such that flat is impossible?









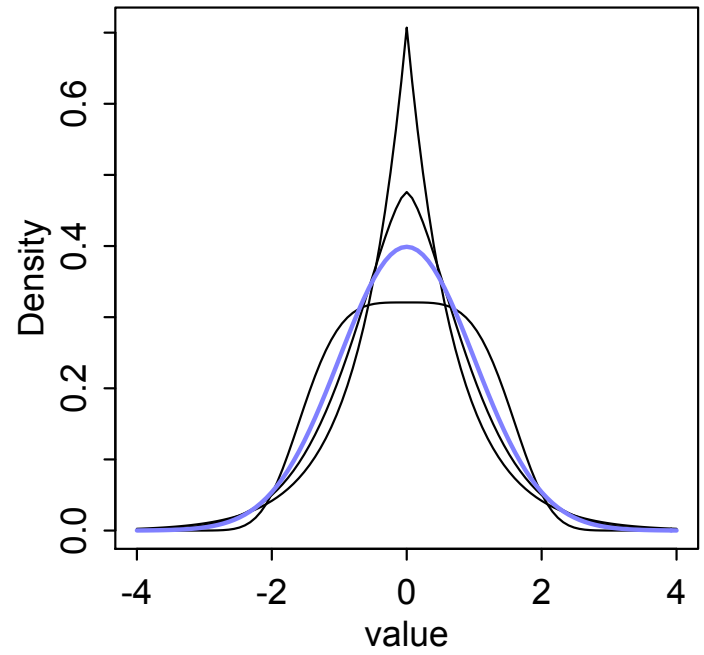
100 pebbles

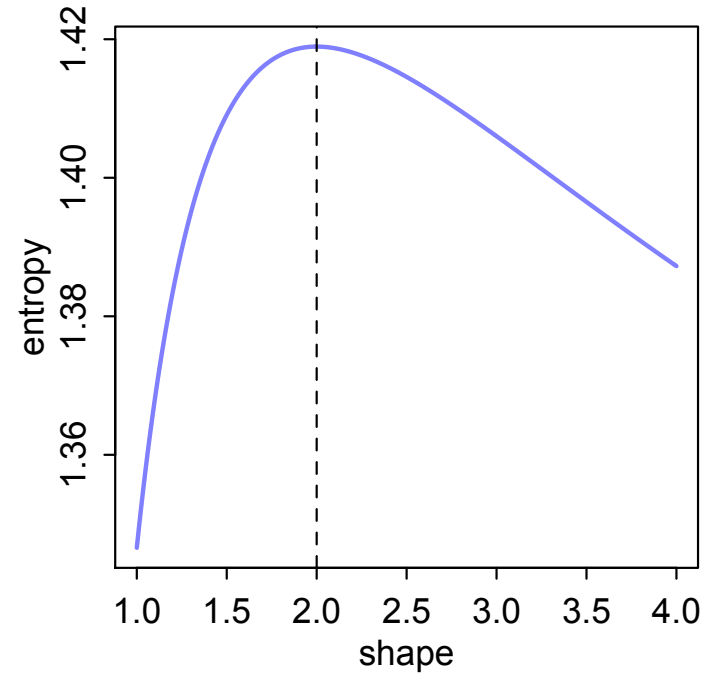

1
2
3
4
5

Constraint:
variance must equal 1

Revisit Gaussian

- Constraints: unbounded real, finite variance
- Add up fluctuations, distribution of sums converges to Gaussian
  - Why? Vastly many more ways to realize Gaussian than another shape.
  - Flattest distribution with given variance
  - Ergo, Gaussian has maxent for all continuous, unbounded distributions with finite variance

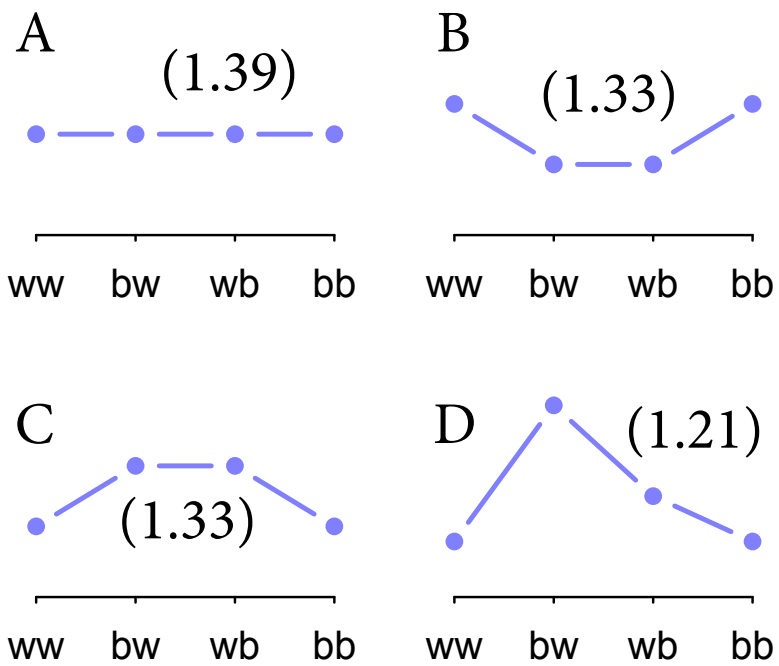




Revisit binomial

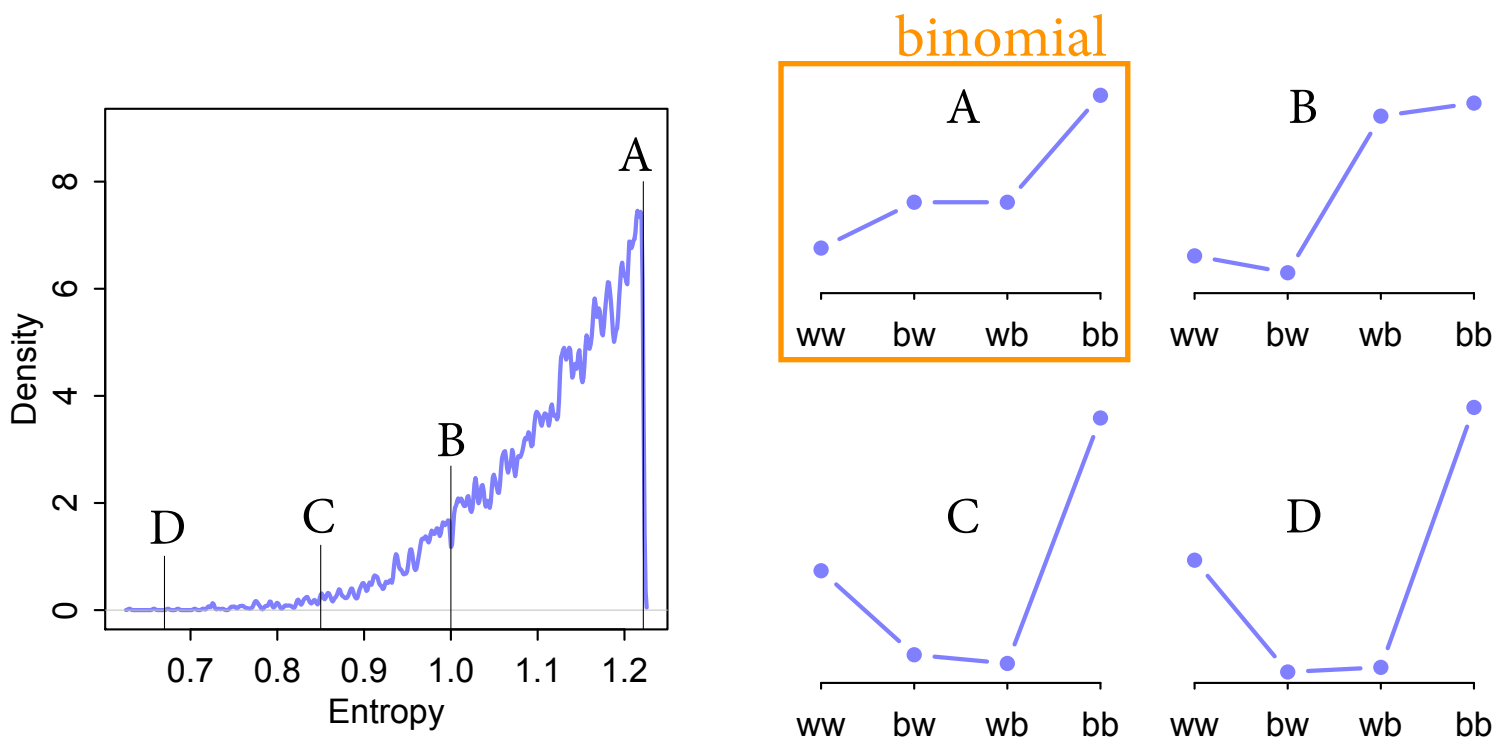
- Constraints: binary outcomes, constant expected value across trials
- Maxent now is binomial

Distribution	ww	bw	wb	bb
A	1/4	1/4	1/4	1/4
B	2/6	1/6	1/6	2/6
C	1/6	2/6	2/6	1/6
D	1/8	4/8	2/8	1/8



Revisit binomial

- Constraints: binary outcomes, constant expected value across trials
- Maxent now is binomial
- e.g.: 2 trials, expected value 1.4



Generalized Linear Models

- Goal: Connect linear model to outcome variable
- 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