Tiny Data, Approximate Bayesian Computation and the Socks of Karl Broman

Rasmus Bååth, Lund University
@rabaath || rasmus.baath@gmail.com
http://www.sumsar.net

Karl Broman
@kwbroman

Following

That the 1st 11 socks in the laundry are each distinct suggests there are a lot more socks.


THE DAWN OF
BIG DATA

Approximate

Bayesian

Computation

Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - Data

Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - ✓ Data

Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - ✓ Data
  - ⟶ A *generative* model

Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - ✓ Data
  - → A *generative* model
  - *Priors*. What information the model has before seeing the data.

Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - ✓ Data
  - ⟶ A *generative* model
  - *Priors*. What information the model has before seeing the data.
  - A *criterion* for when simulated data *matches* the actual data.

A Model of Picking out Socks from Your Washing Machine

n_pairs

```
n_pairs <- 9
```


ONE DOES NOT SIMPLY
MATCH ALL THE SOCKS

```
n_pairs <- 9

n_odd
```

```
n_pairs <- 9
n_odd   <- 5
```

```r
n_pairs <- 9
n_odd   <- 5
```


AEG
LAVAMAT

```
> socks
 [1]  1  1  2  2  3  3  4  4  5  5  6  6  7  7
8  8  9  9 10 11 12 13 14
```

```
> socks
 [1]  1  1  2  2  3  3  4  4  5  5  6  6  7  7
8  8  9  9 10 11 12 13 14
n_sock_types <- n_pairs + n_odd
```

```
> socks
 [1]  1  1  2  2  3  3  4  4  5  5  6  6  7  7
8  8  9  9 10 11 12 13 14
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
```

```
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
```

```r
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
```

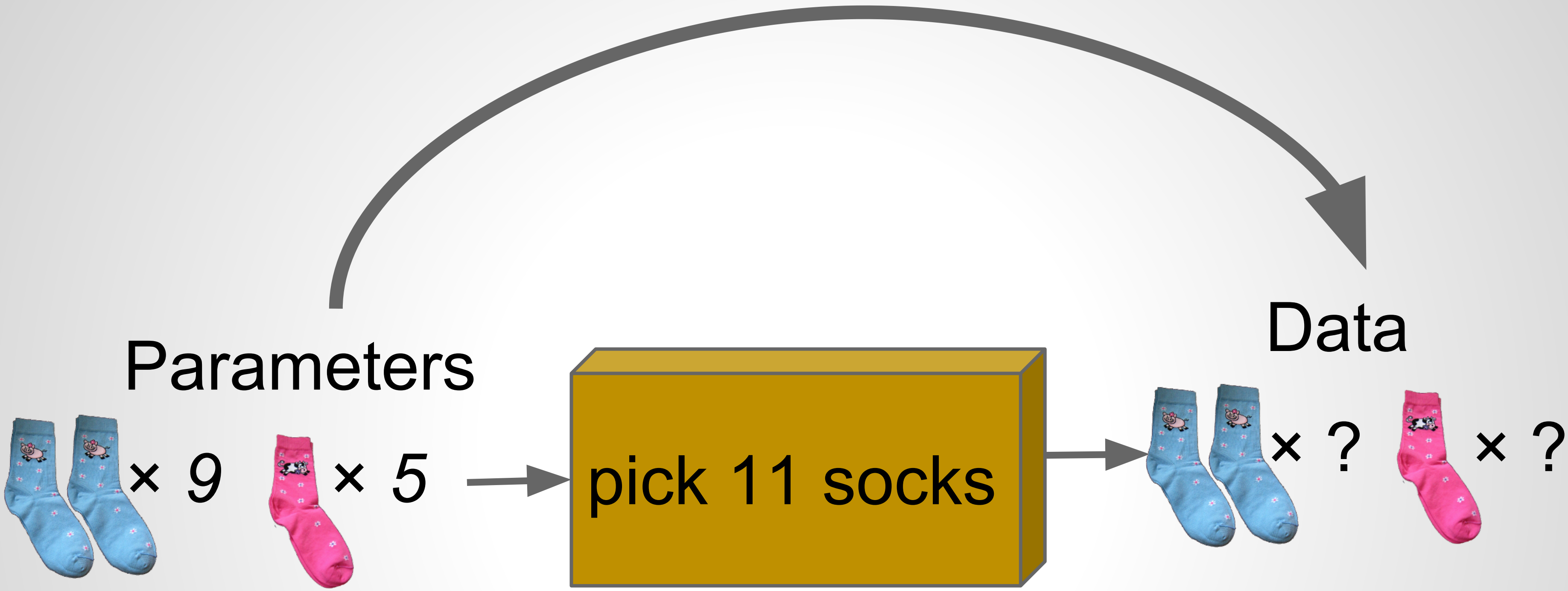
```
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
picked_socks <- sample(socks, 11)
```

```
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
picked_socks <- sample(socks, 11)
sock_counts <- table(picked_socks)
```

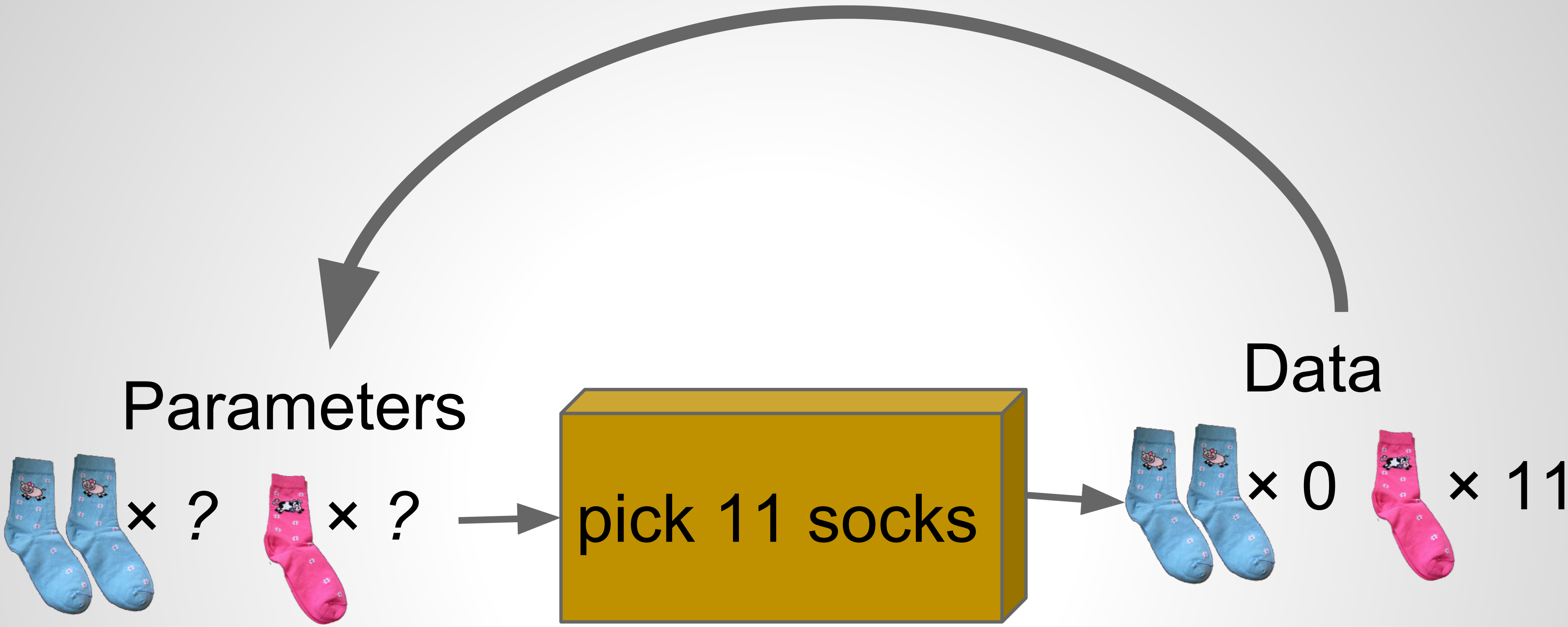
```
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
picked_socks <- sample(socks, 11)
sock_counts <- table(picked_socks)
> sock_counts
picked_socks
 1  3  4  5  7  8  9 10 11
 1  2  2  1  1  1  1  1  1
```

```
n_sock_types <- n_pairs + n_odd
socks <- rep(1:n_sock_types,
             rep( 2:1, c(n_pairs, n_odd) ))
picked_socks <- sample(socks, 11)
sock_counts <- table(picked_socks)

unique <- sum(sock_counts == 1)
pairs  <- sum(sock_counts == 2)
```



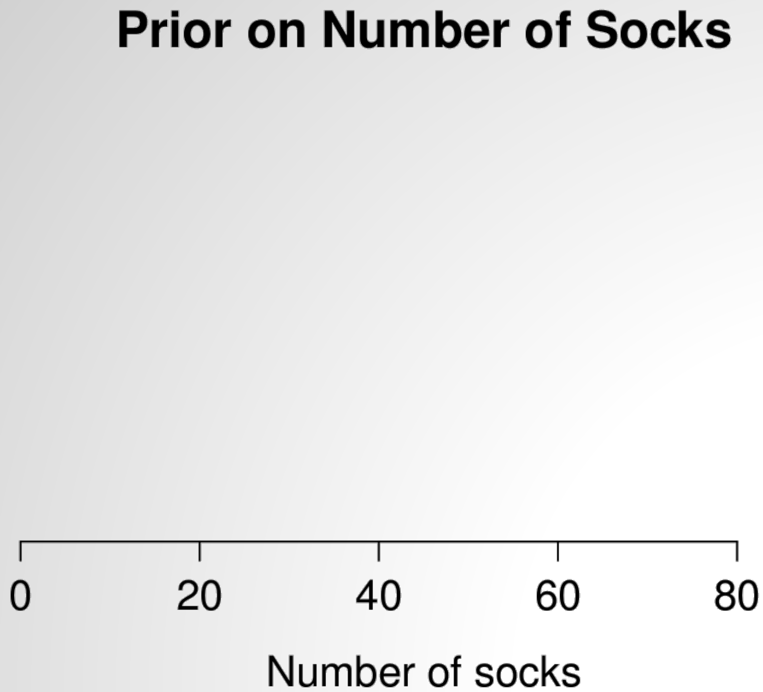

```
pick_socks(pairs = 9, odds = 5, n_pick = 11)
```

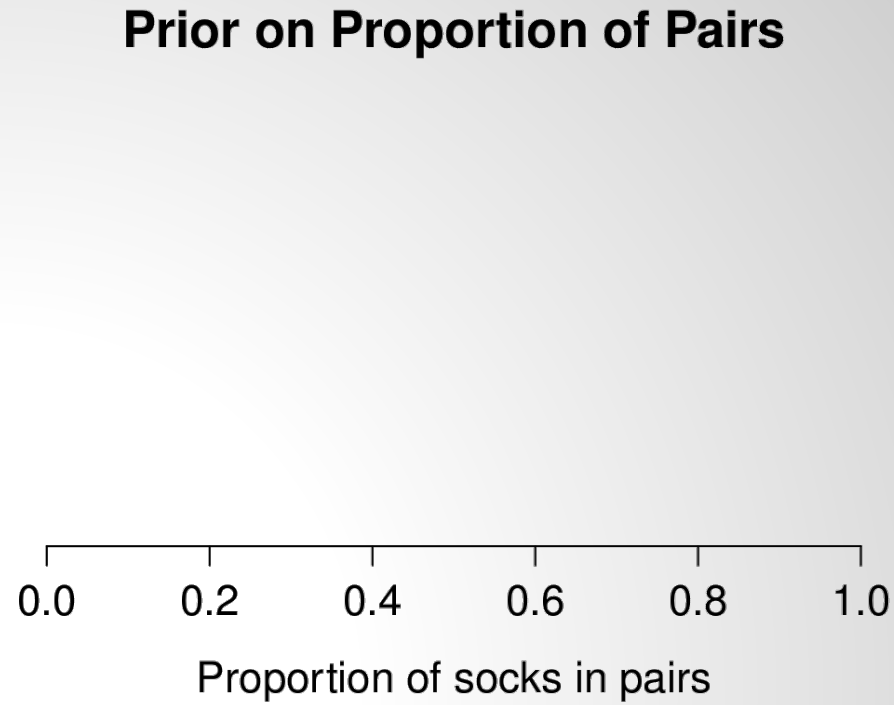


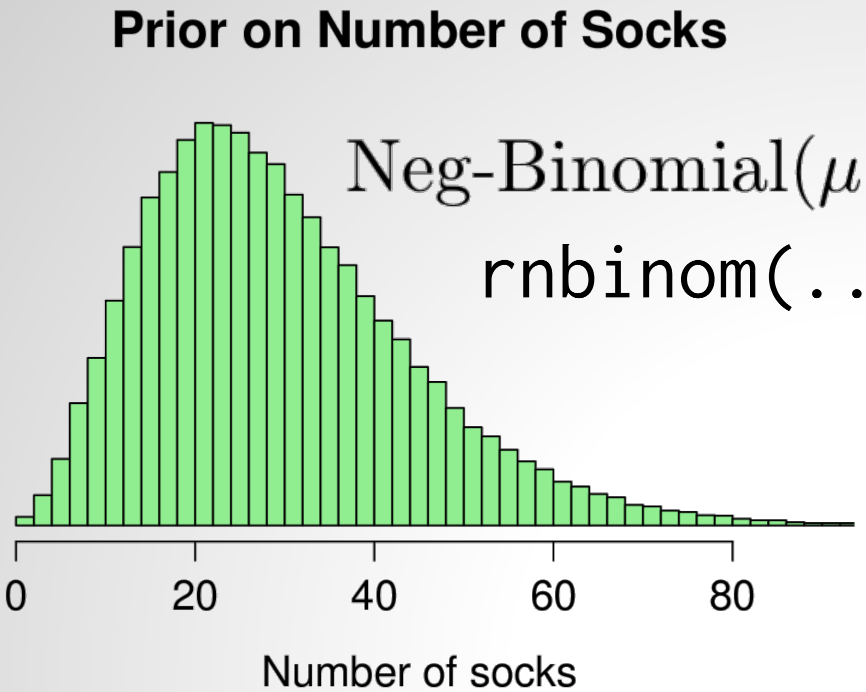

```
prob_socks(pairs = 0, odds = 11)
```

Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - ✓ Data
  - ✓ A *generative* model
  - → *Priors*. What information the model has before seeing the data.
  - A *criterion* for when simulated data *matches* the actual data.

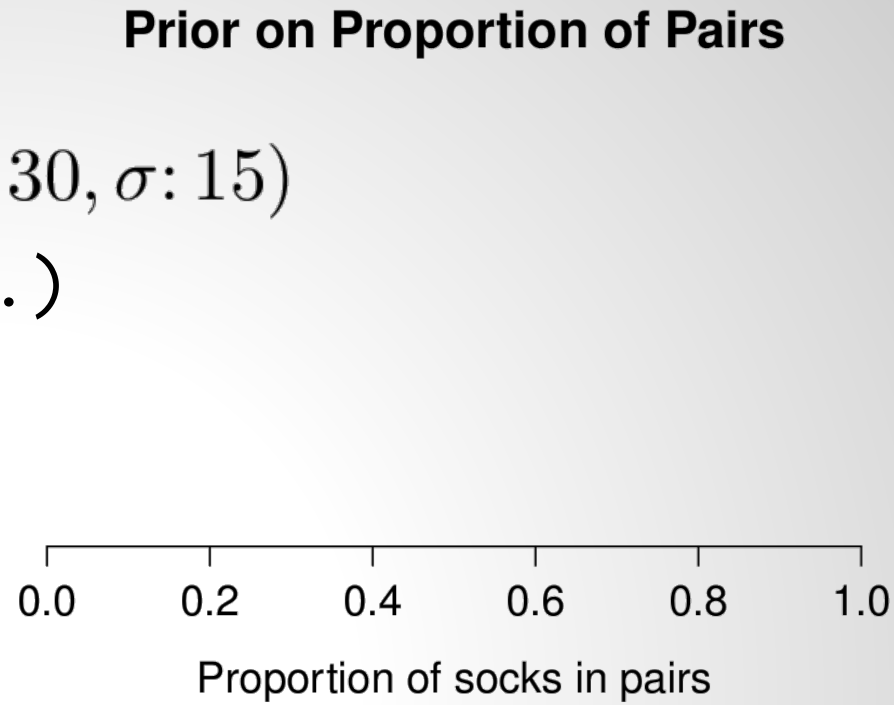

Prior on Number of Socks
0
20
40
60
80
Number of socks

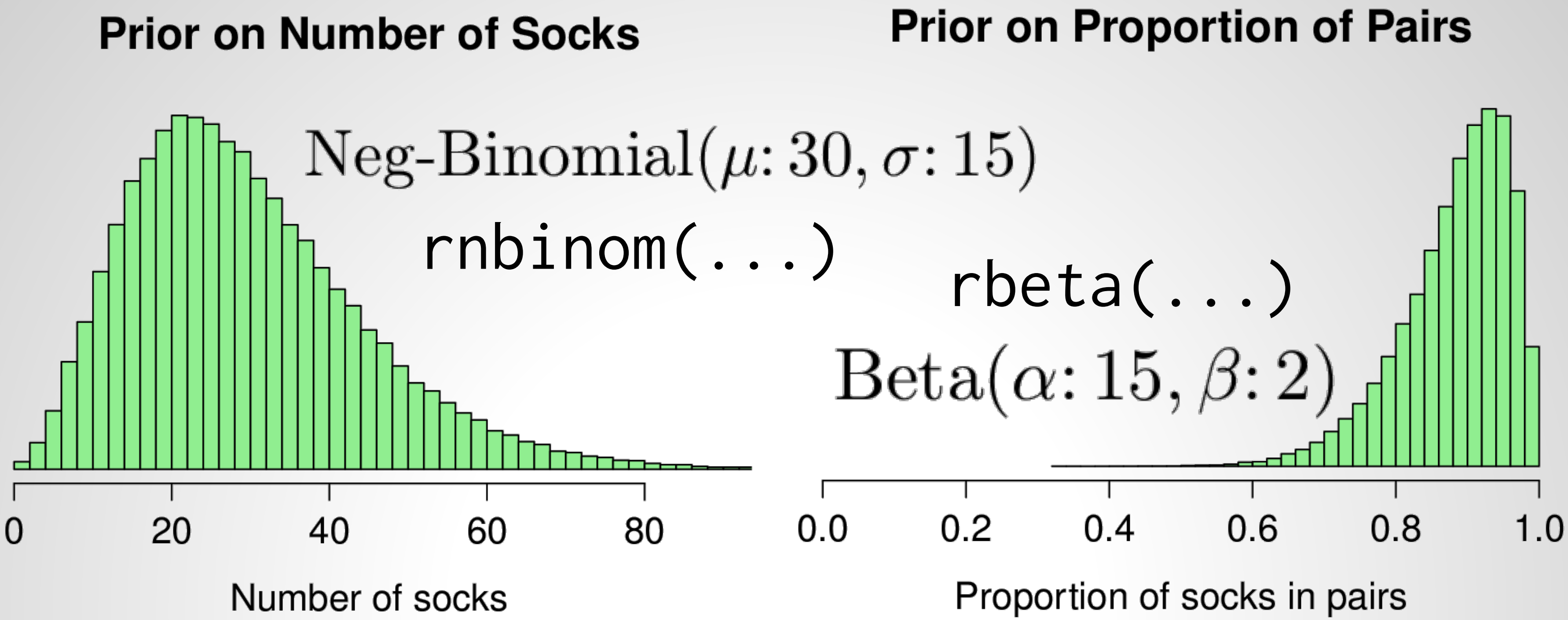

Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs

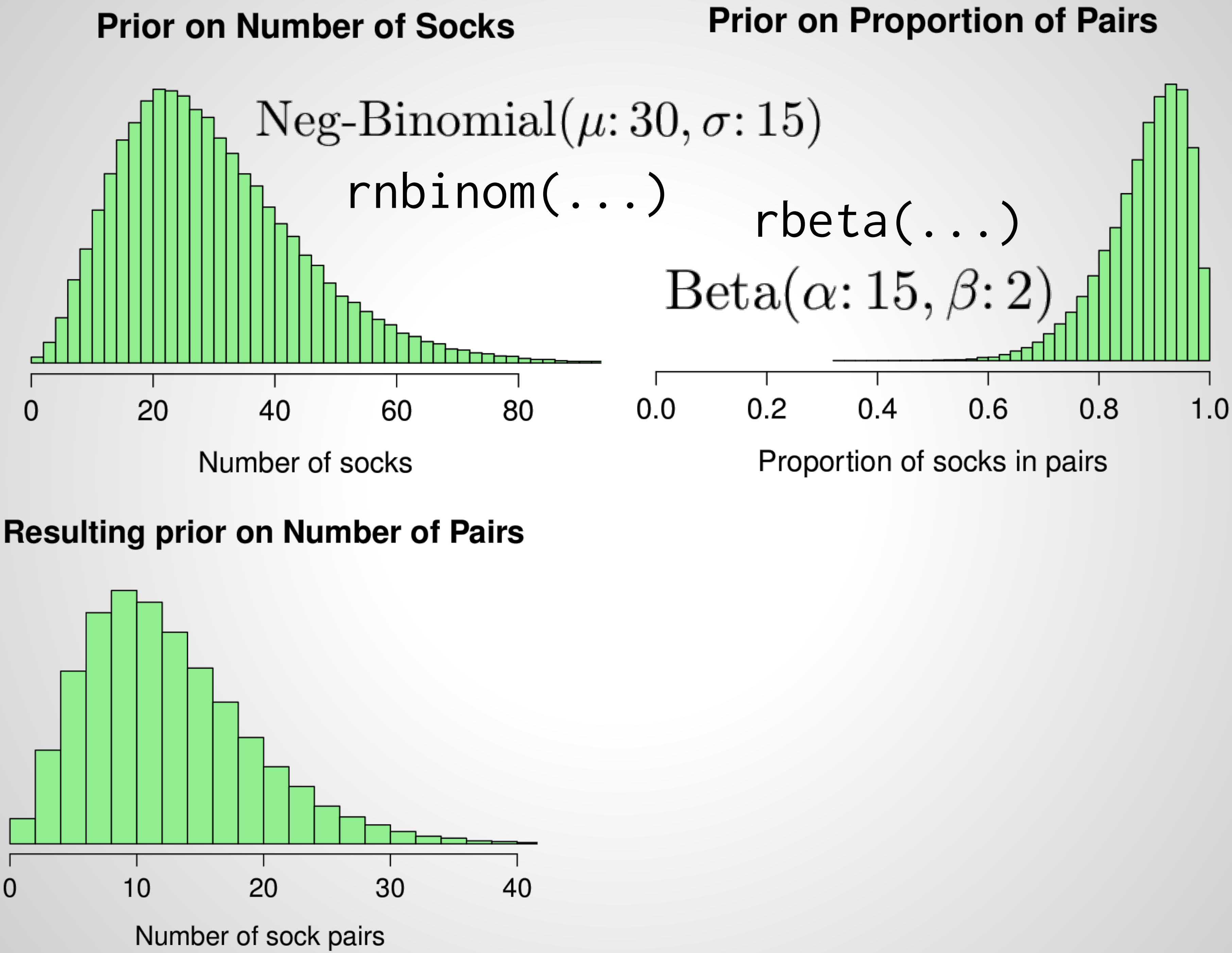


$\text{Neg-Binomial}(\mu\colon 30, \sigma\colon 15)$

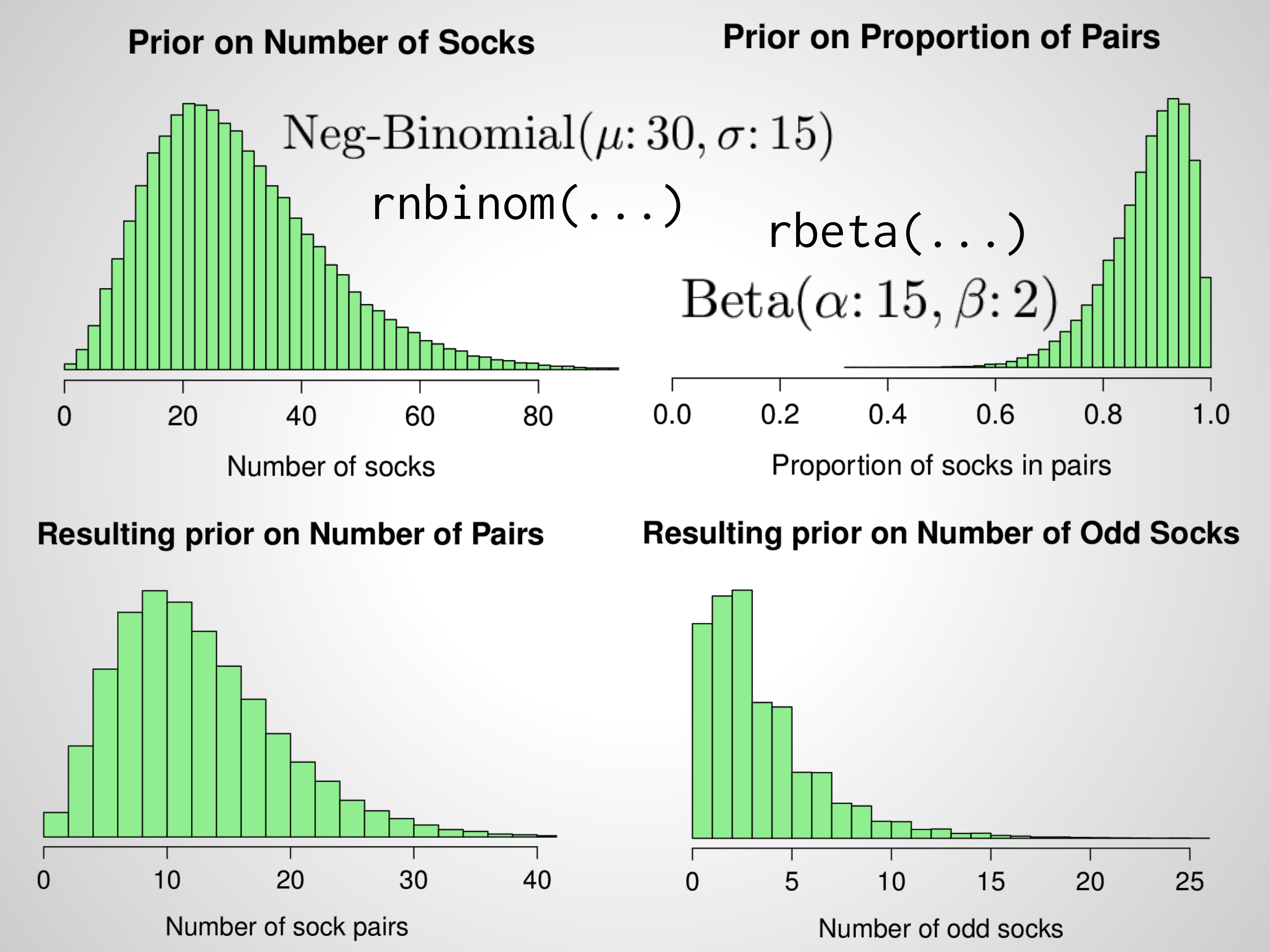
```
rnbinom(...)
```




Prior on Number of Socks
Prior on Proportion of Pairs
Neg-Binomial(μ: 30, σ: 15)
rnbinom(...)
rbeta(...)
Beta(α: 15, β: 2)
0
20
40
60
80
Number of socks
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs



Prior on Number of Socks
Neg-Binomial(μ: 30, σ: 15)
rnbinom(...)
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
rbeta(...)
Beta(α: 15, β: 2)
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
Resulting prior on Number of Pairs
0
10
20
30
40
Number of sock pairs


Prior on Number of Socks
Neg-Binomial(μ: 30, σ: 15)
rnbinom(...)
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
rbeta(...)
Beta(α: 15, β: 2)
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
Resulting prior on Number of Pairs
0
10
20
30
40
Number of sock pairs
Resulting prior on Number of Odd Socks
0
5
10
15
20
25
Number of odd socks

Approximate Bayesian Computation

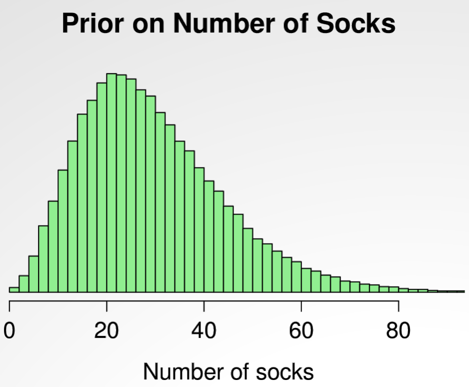
- A method of figuring out *unknowns* that requires:
  - ✓ Data
  - ✓ A *generative* model
  - ✓ *Priors*. What information the model has before seeing the data.
  - → A *criterion* for when simulated data *matches* the actual data.

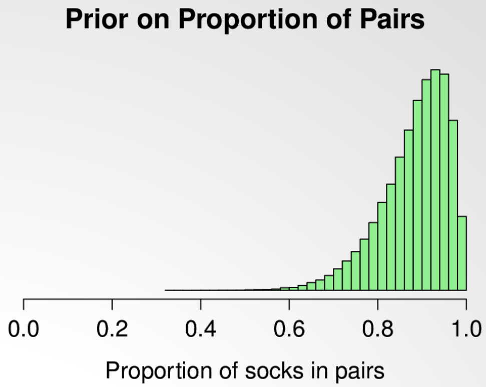
Approximate Bayesian Computation

- A method of figuring out *unknowns* that requires:
  - ✓ Data
  - ✓ A *generative* model
  - ✓ *Priors*. What information the model has before seeing the data.
  - ✓ A *criterion* for when simulated data *matches* the actual data.

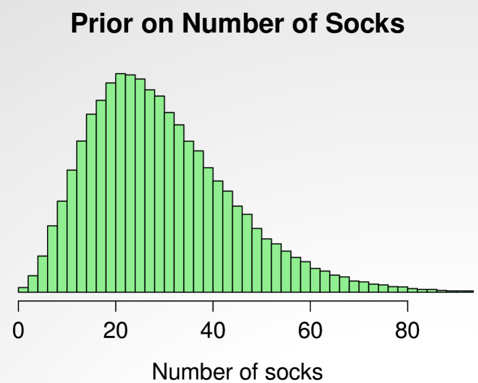
Let's do the ABC!

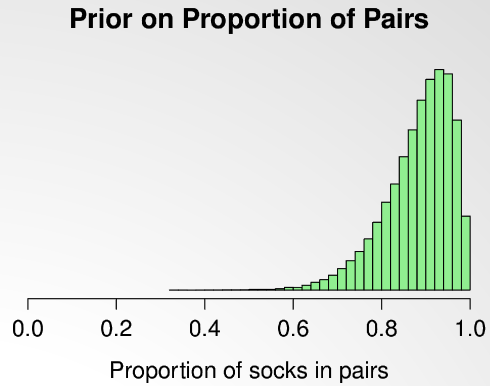
Priors


Prior on Number of Socks
0
20
40
60
80
Number of socks

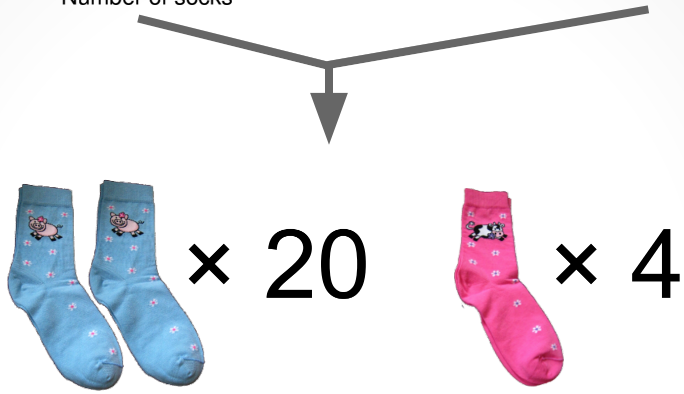

Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs

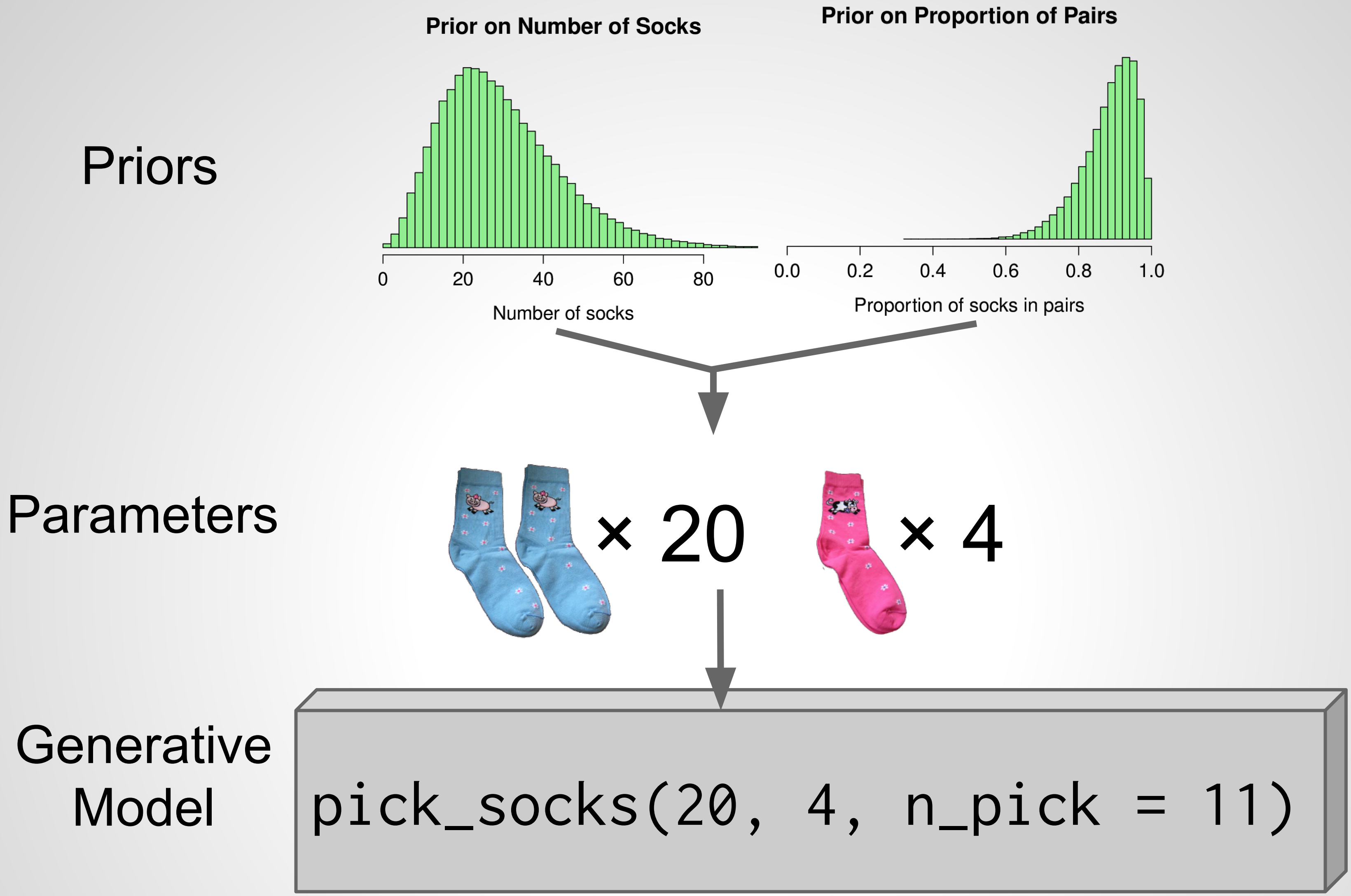
Priors

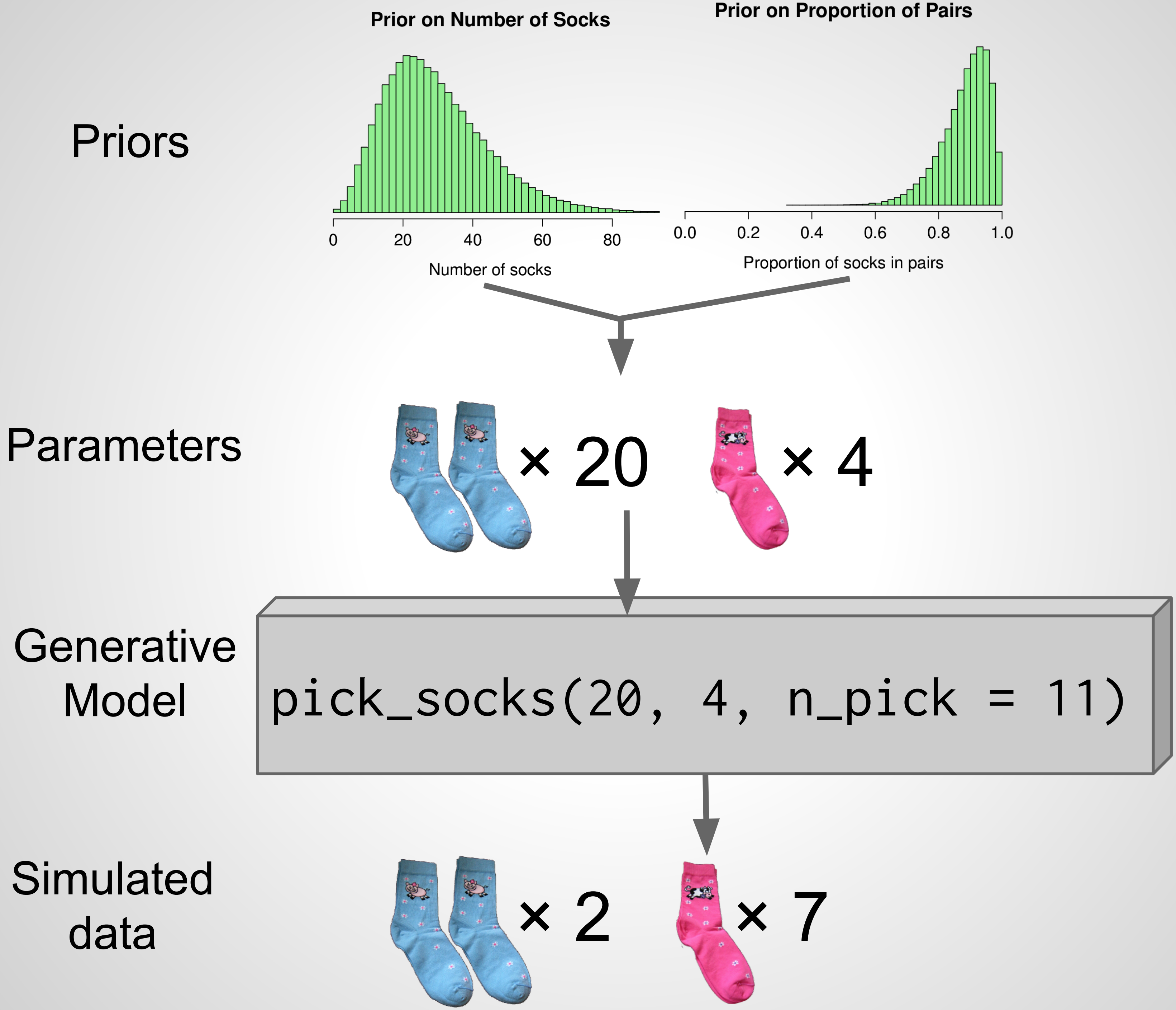


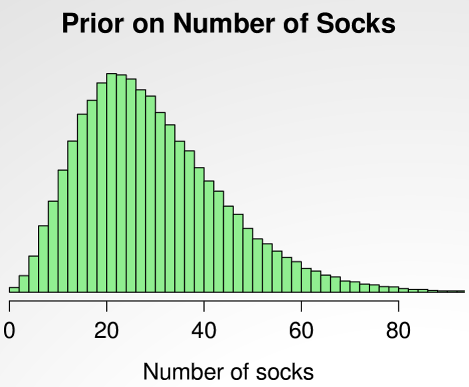


Parameters

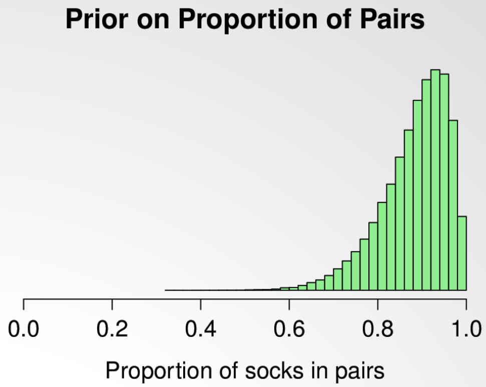



Priors
Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
Parameters
× 20
× 4
Generative
Model
pick_socks(20, 4, n_pick = 11)

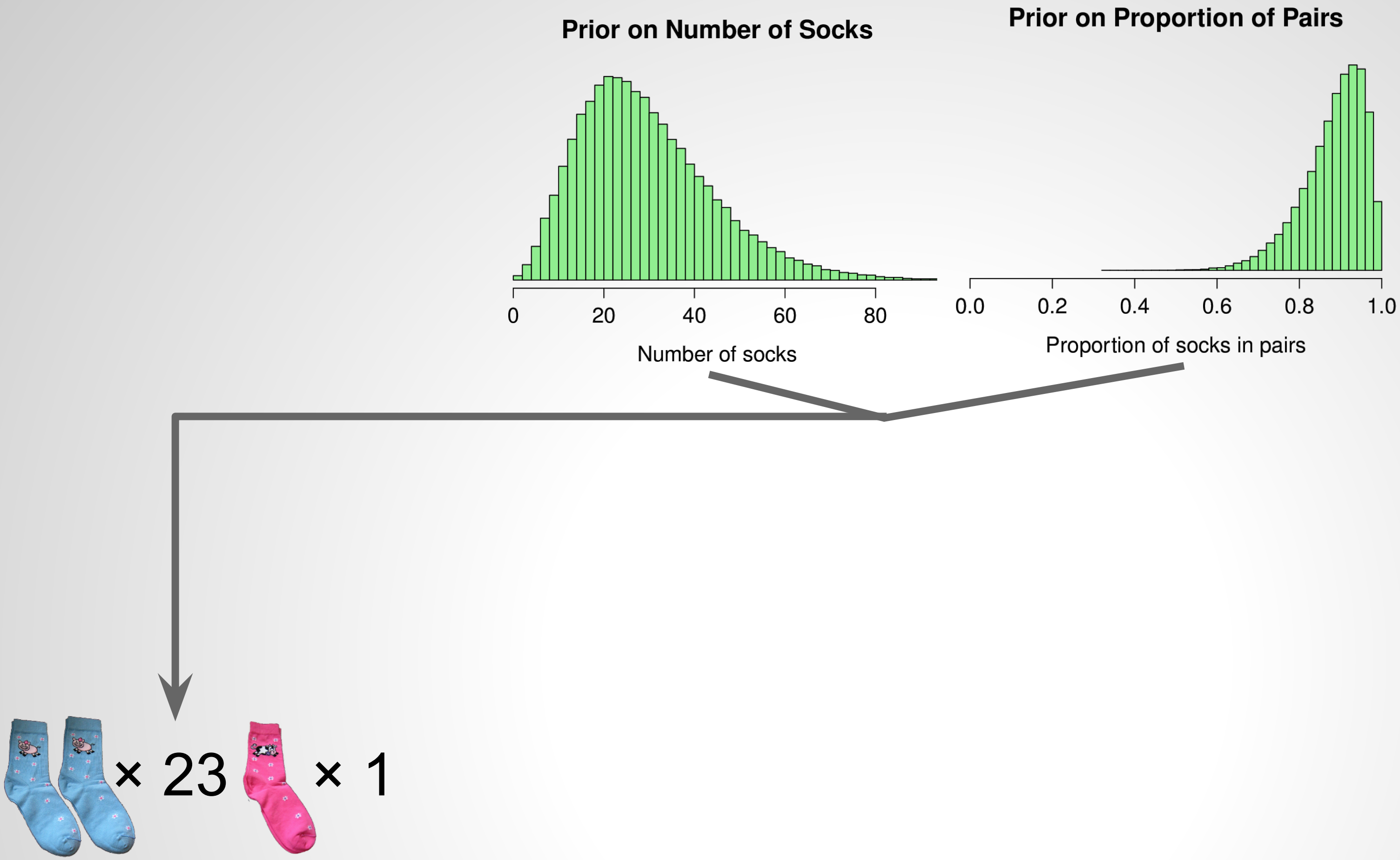

Priors
Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
Parameters
× 20
× 4
Generative Model
pick_socks(20, 4, n_pick = 11)
Simulated data
× 2
× 7

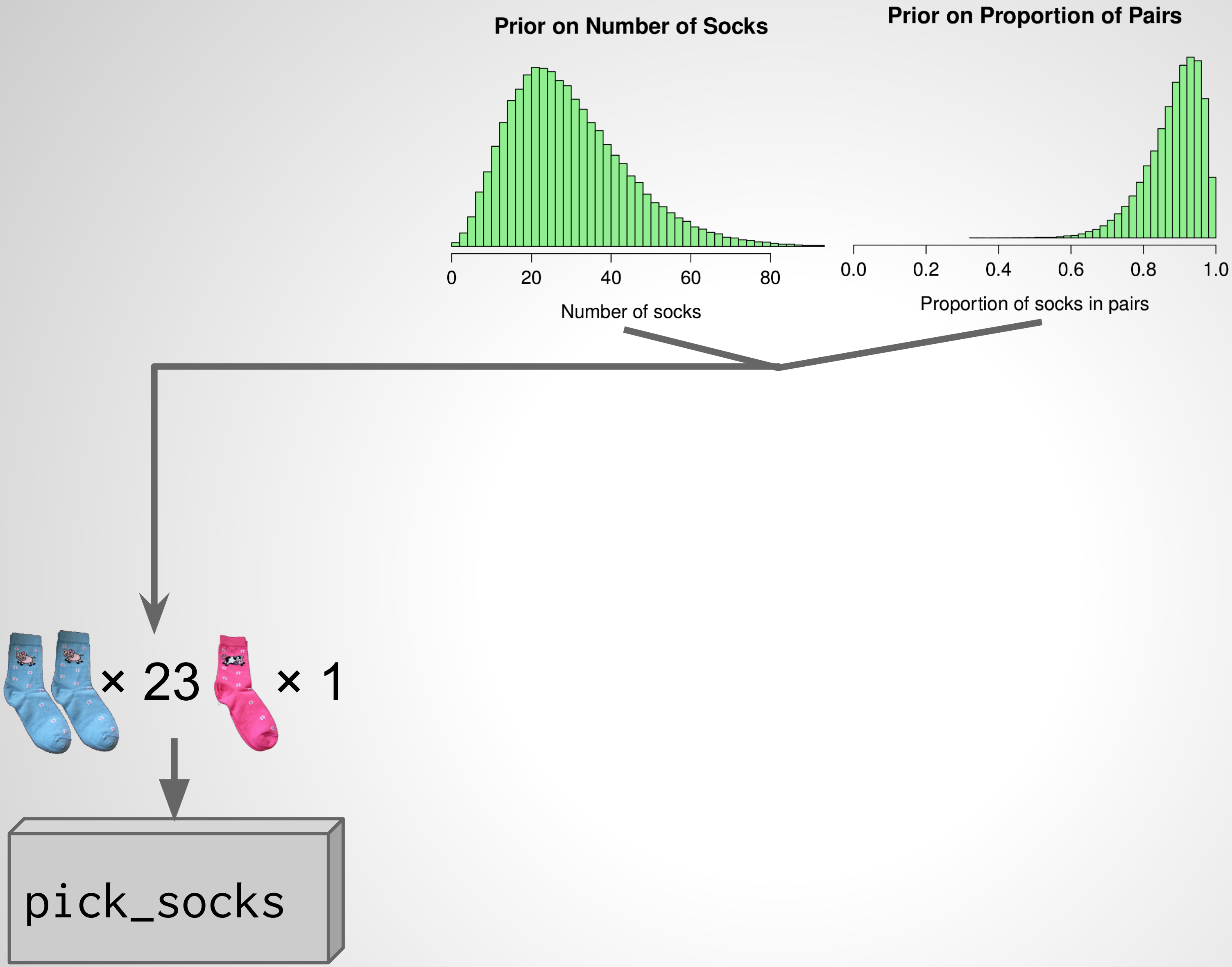


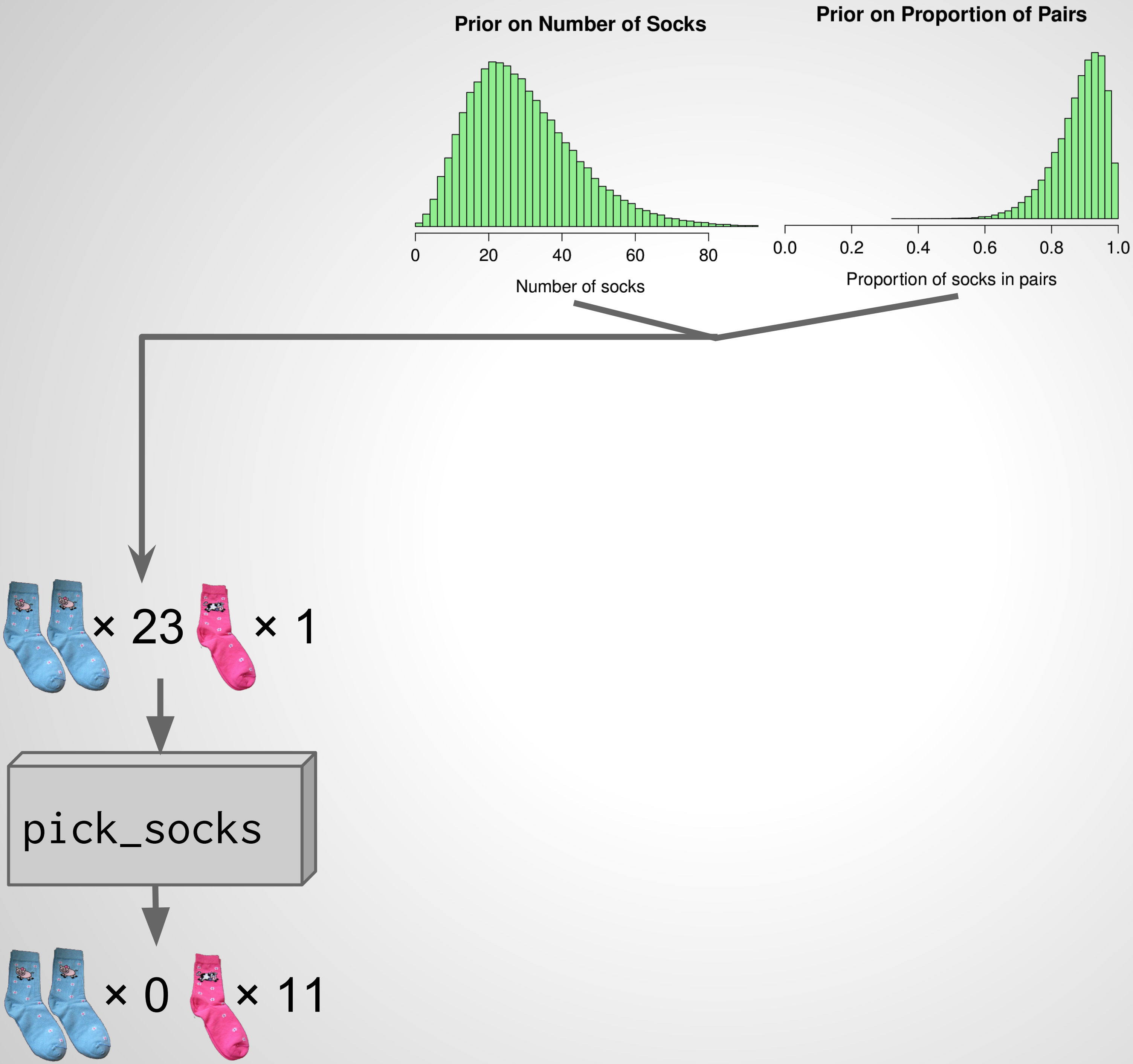
Prior on Number of Socks
0
20
40
60
80
Number of socks

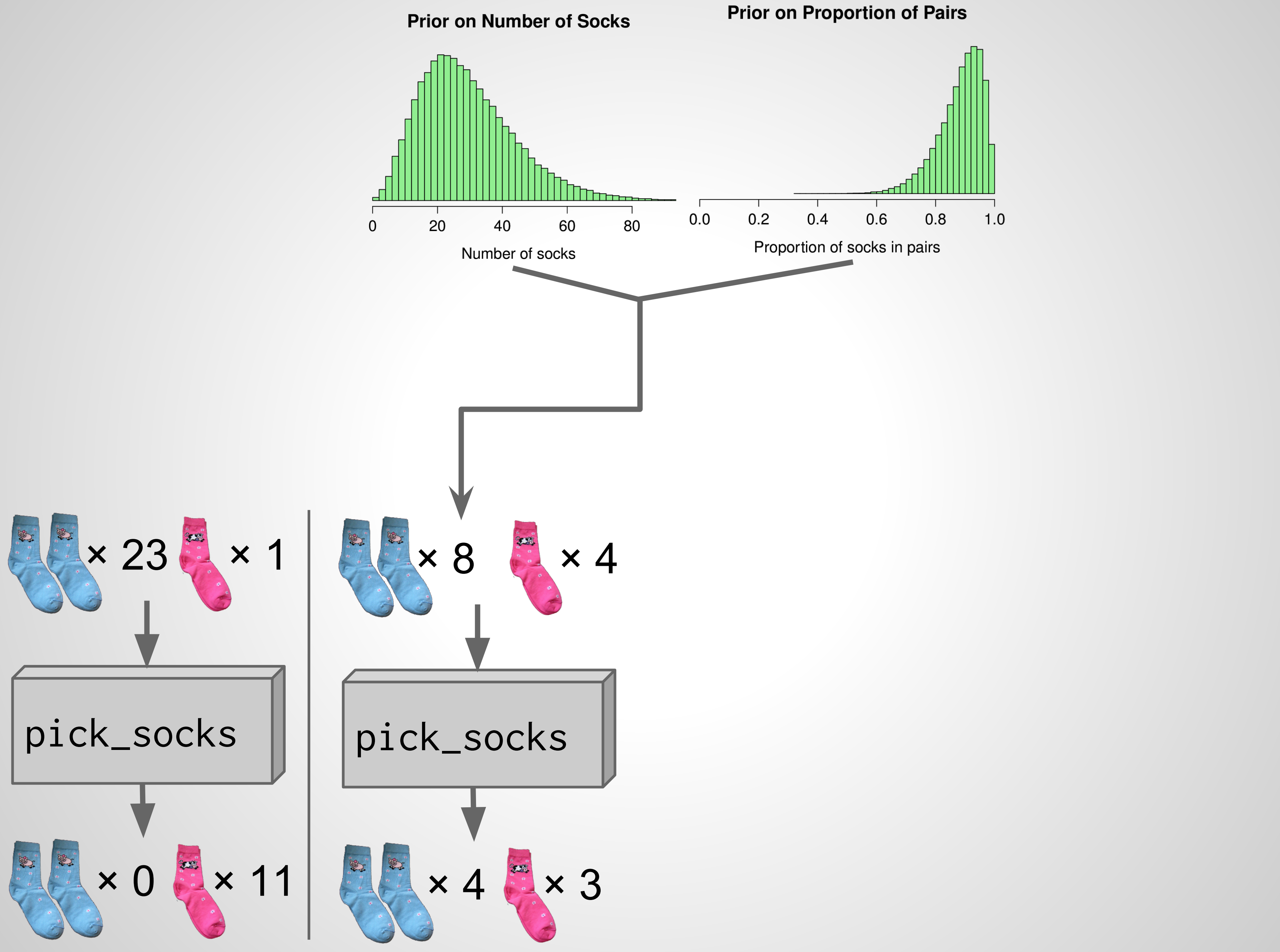


Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs

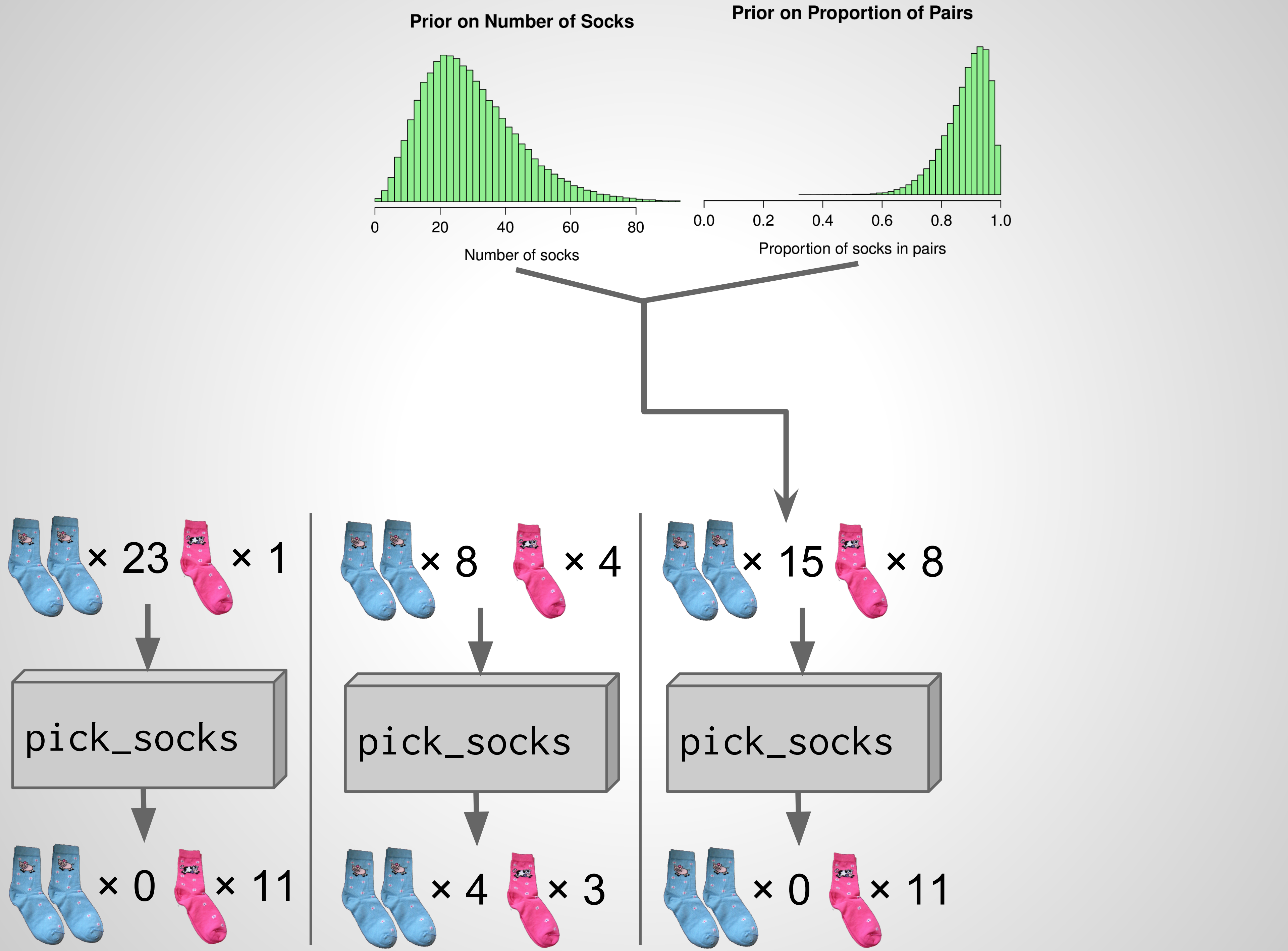

Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1

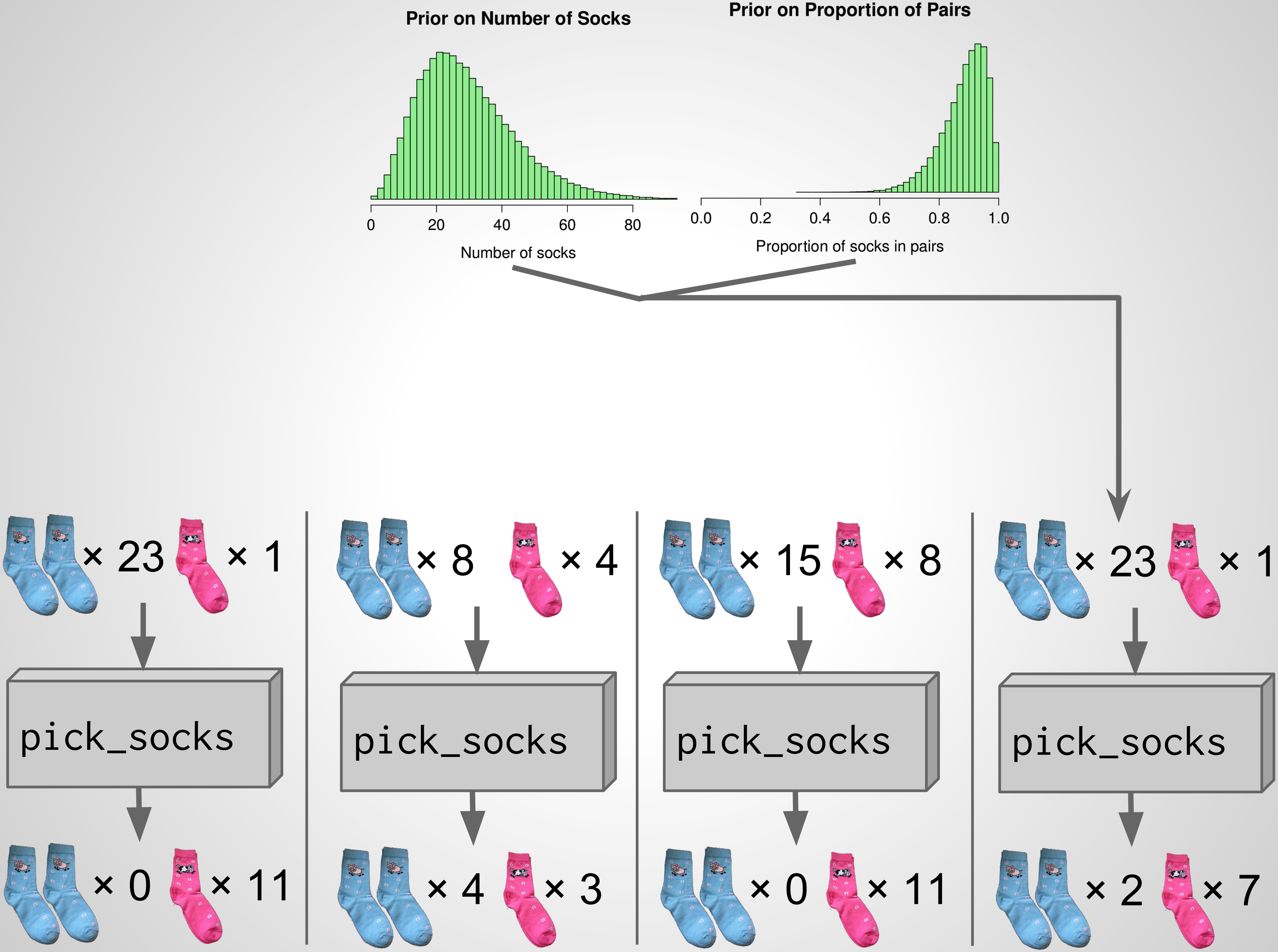

Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks


Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks
× 0
× 11



Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks
× 0
× 11
× 8
× 4
pick_socks
× 4
× 3

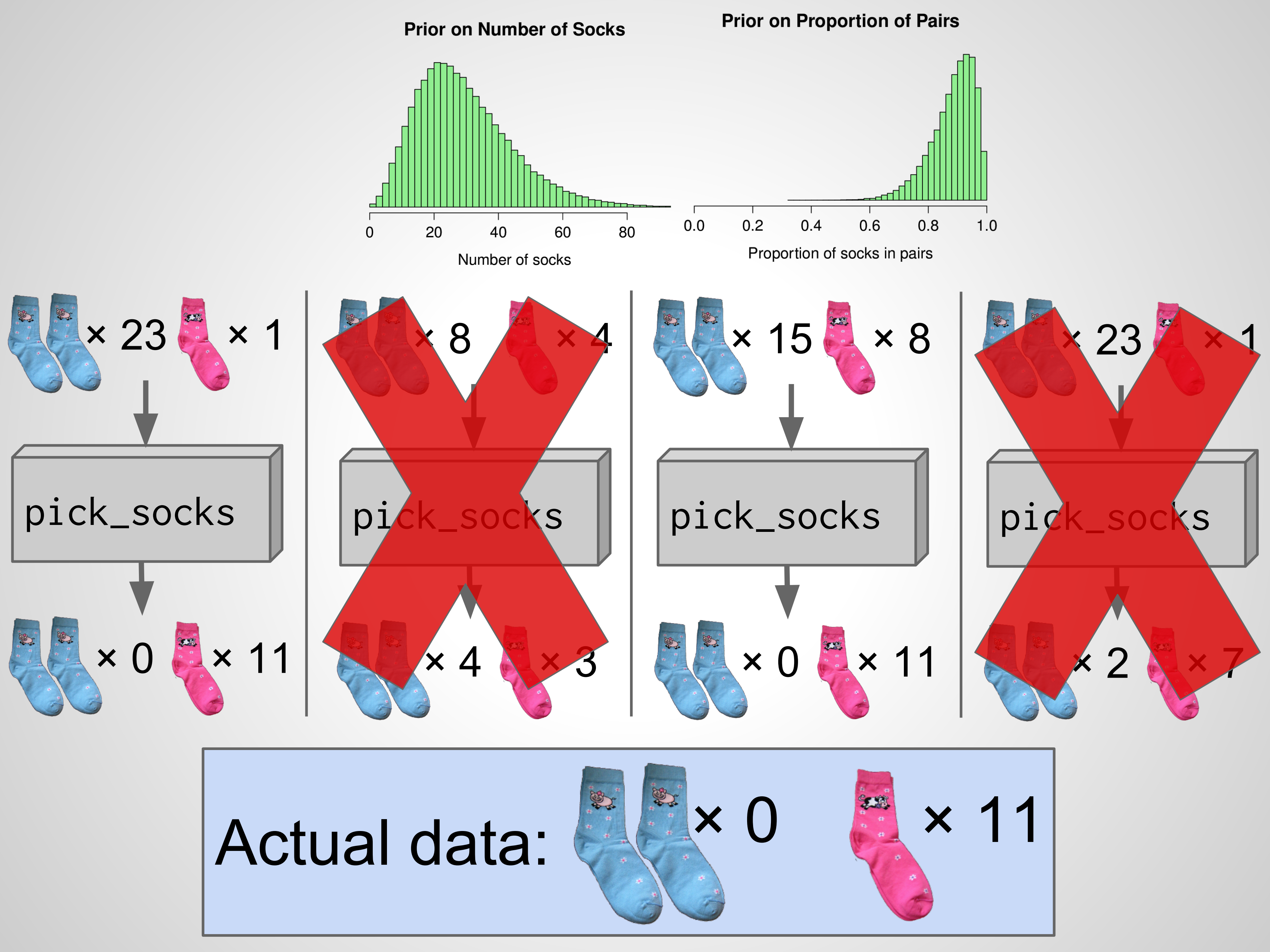

Prior on Number of Socks
Prior on Proportion of Pairs
0 20 40 60 80
Number of socks
0.0 0.2 0.4 0.6 0.8 1.0
Proportion of socks in pairs
× 23
× 1
× 8
× 4
× 15
× 8
pick_socks
pick_socks
pick_socks
× 0
× 11
× 4
× 3
× 0
× 11



Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks
× 0
× 11
× 8
× 4
pick_socks
× 4
× 3
× 15
× 8
pick_socks
× 0
× 11
× 23
× 1
pick_socks
× 2
× 7

Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks
× 0
× 11
× 8
× 4
pick_socks
× 4
× 3
× 15
× 8
pick_socks
× 0
× 11
× 23
× 1
pick_socks
× 2
× 7
Actual data:
× 0
× 11

Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks
× 0
× 11
× 8
× 4
pick_socks
× 4
× 3
× 15
× 8
pick_socks
× 0
× 11
× 23
× 1
pick_socks
× 2
× 7
Actual data:
× 0
× 11



Prior on Number of Socks
0
20
40
60
80
Number of socks
Prior on Proportion of Pairs
0.0
0.2
0.4
0.6
0.8
1.0
Proportion of socks in pairs
× 23
× 1
pick_socks
× 0
× 11
× 8
× 4
pick_socks
× 4
× 3
× 15
× 8
pick_socks
× 0
× 11
× 23
× 1
pick_socks
× 2
× 7
Actual data:
× 0
× 11

```
> head(sock_sim)
   n_pairs n_odd n_socks prop_pairs
       12     4      28     0.8391
       11     4      26     0.8802
       19     2      40     0.9604
       11     4      26     0.8699
       24     5      53     0.9192
```

```
> head(sock_sim)
  n_pairs n_odd n_socks prop_pairs pairs unique
       12     4      28     0.8391     1      9
       11     4      26     0.8802     0     11
       19     2      40     0.9604     1      9
       11     4      26     0.8699     3      5
       24     5      53     0.9192     0     11
```

```
> head(sock_sim)
  n_pairs n_odd n_socks prop_pairs pairs unique
       12     4      28     0.8391     1      9
       11     4      26     0.8802     0     11
       19     2      40     0.9604     1      9
       11     4      26     0.8699     3      5
       24     5      53     0.9192     0     11

posterior <- subset(sock_sim, unique == 11)
```

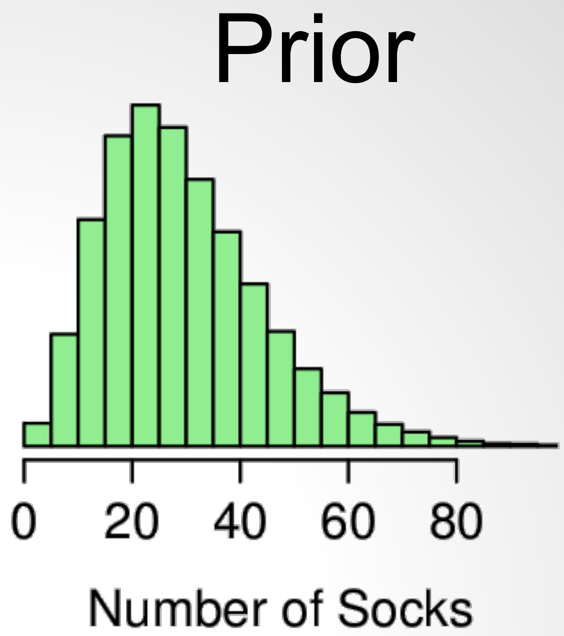
```
> head(sock_sim)
  n_pairs n_odd n_socks prop_pairs pairs unique
      12     4      28     0.8391     1      9
      11     4      26     0.8802     0     11
      19     2      40     0.9604     1      9
      11     4      26     0.8699     3      5
      24     5      53     0.9192     0     11
```

```
posterior <- subset(sock_sim, unique == 11)
> head(posterior)
  n_pairs n_odd n_socks prop_pairs pairs unique
      25     9      59     0.8626     0     11
      24    21      69     0.6980     0     11
      20    20      60     0.6580     0     11
      11     4      26     0.8802     0     11
```

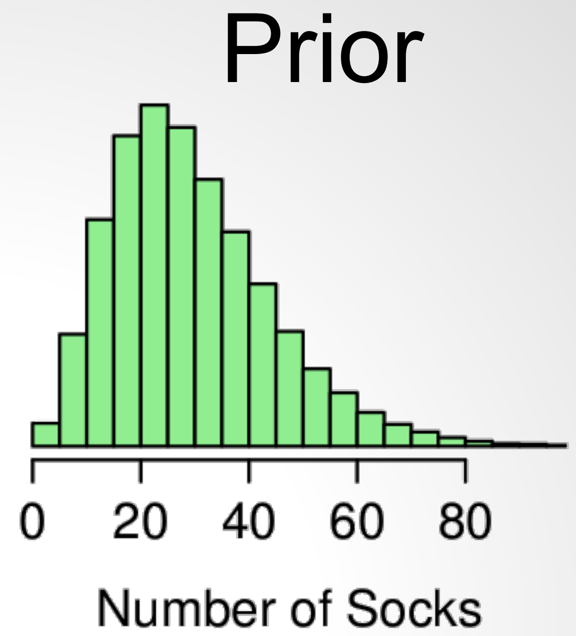
```
> head(sock_sim)
  n_pairs n_odd n_socks prop_pairs pairs unique
       12     4      28     0.8391     1      9
       11     4      26     0.8802     0     11
       19     2      40     0.9604     1      9
       11     4      26     0.8699     3      5
       24     5      53     0.9192     0     11

posterior <- subset(sock_sim, unique == 11)
> head(posterior)
  n_pairs n_odd n_socks prop_pairs pairs unique
       25     9      59     0.8626     0     11
       24    21      69     0.6980     0     11
       20    20      60     0.6580     0     11
       11     4      26     0.8802     0     11
```

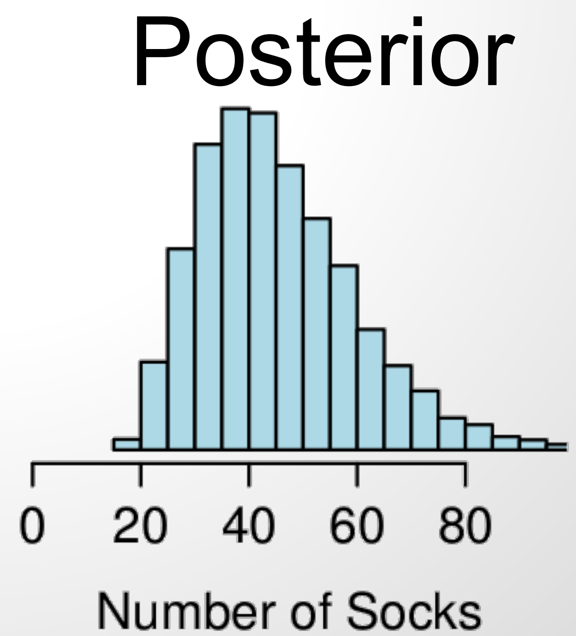
```
hist(sock_sim$n_socks)
```



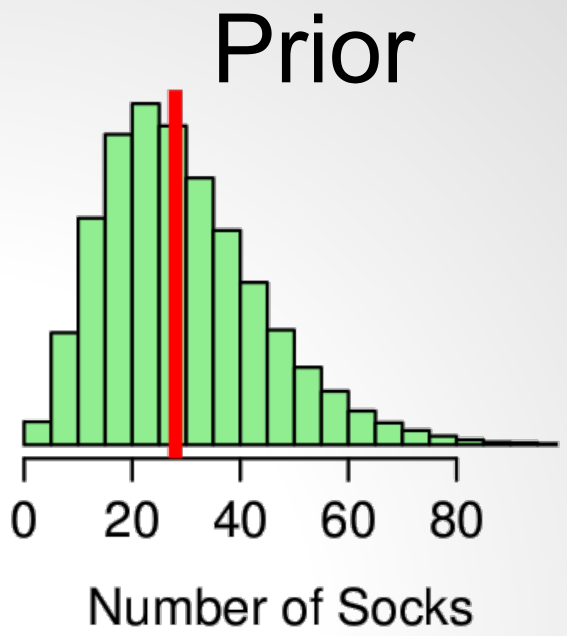
```
hist(sock_sim$n_socks)
```



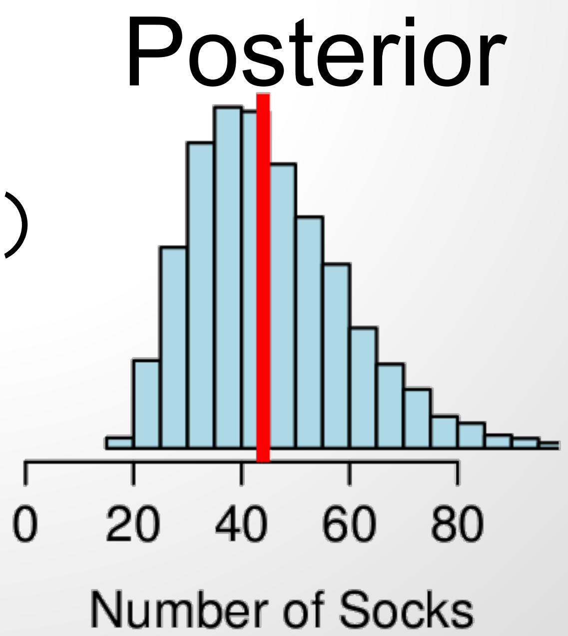

```
hist(posterior$n_socks)
```



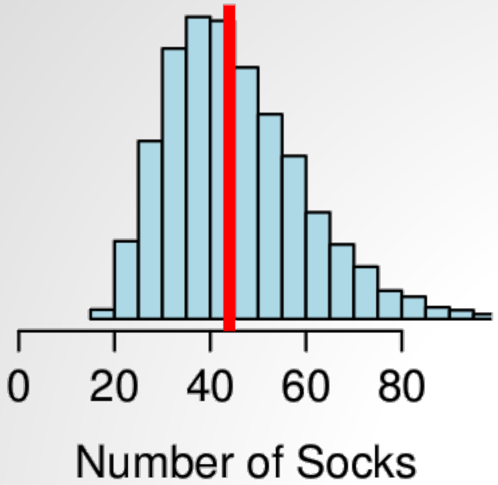
```
hist(sock_sim$n_socks)
median(sock_sim$n_socks)
```



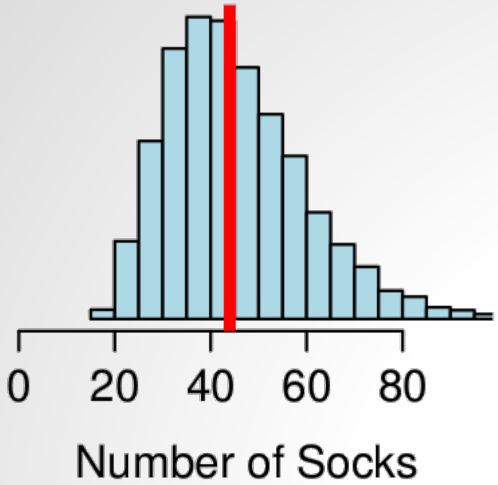

```
hist(posterior$n_socks)
median(posterior$n_socks)
```

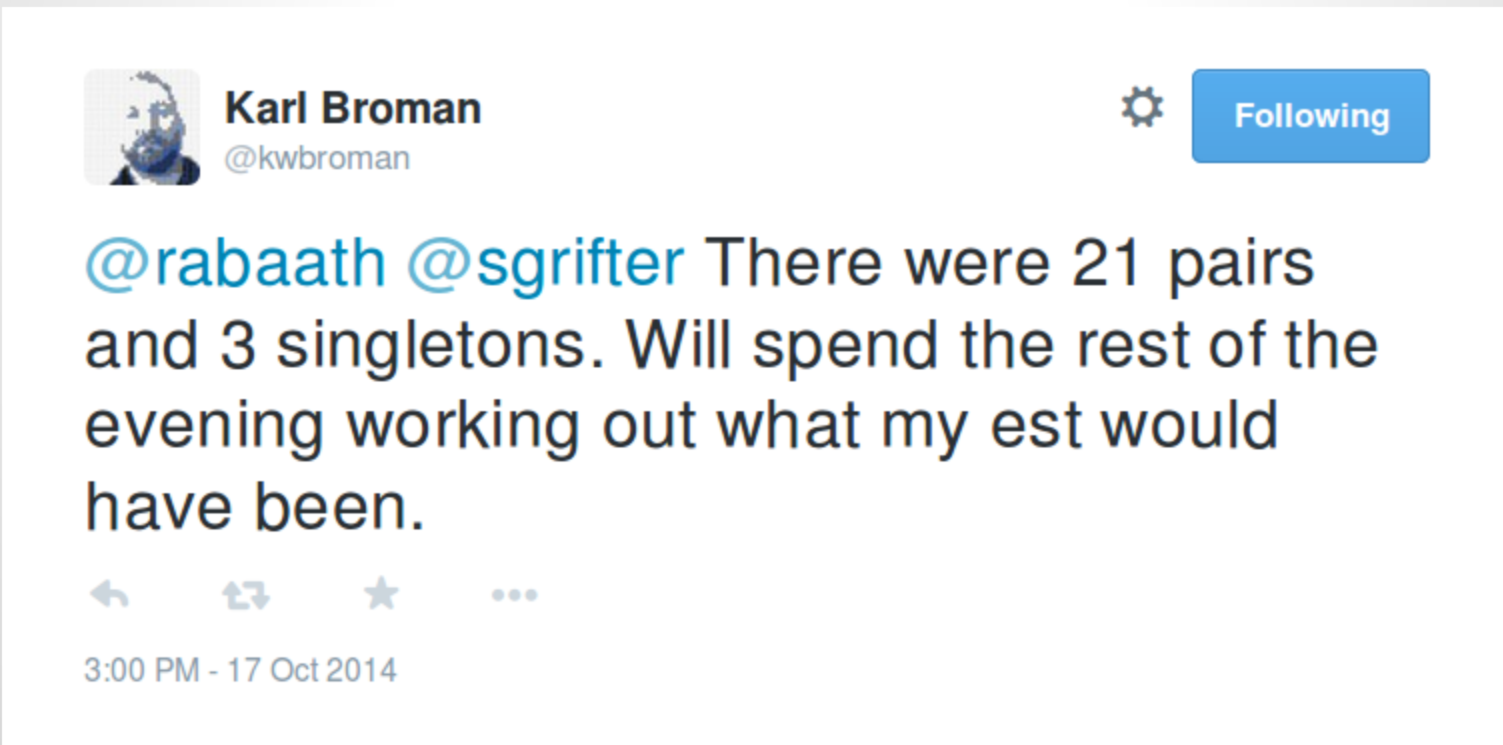


Our best guess: 44



Our best guess: 44





Karl Broman
@kwbroman

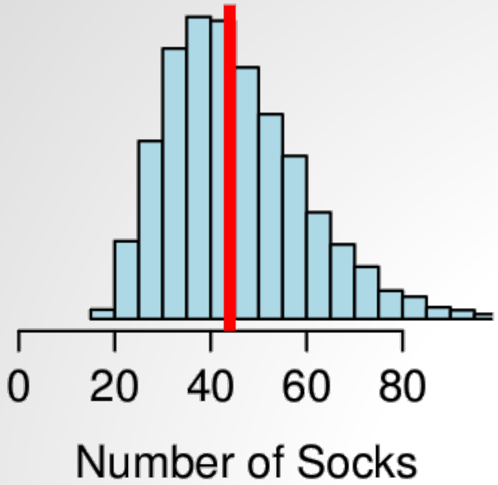
Following

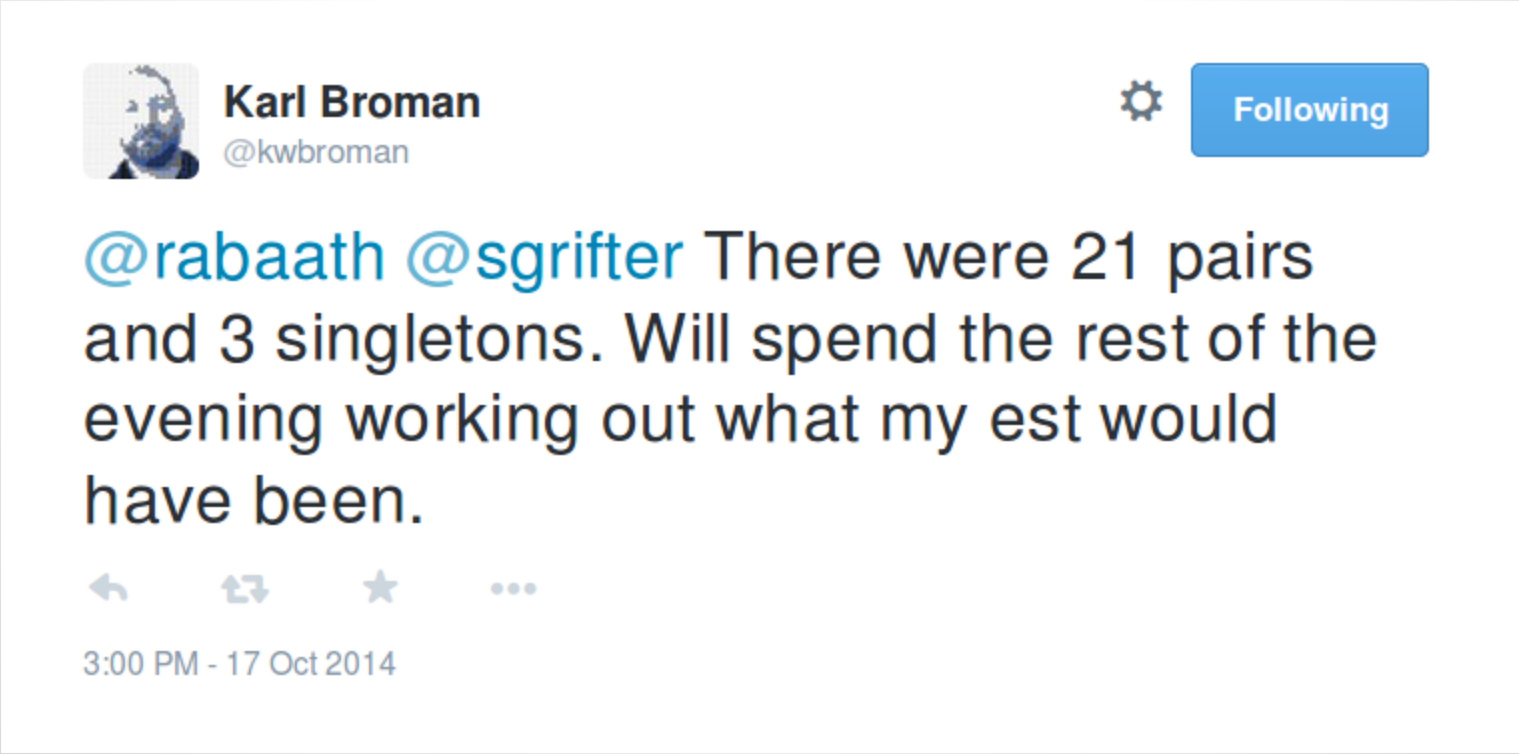
@rabaath @sgrifter There were 21 pairs and 3 singletons. Will spend the rest of the evening working out what my est would have been.

3:00 PM - 17 Oct 2014

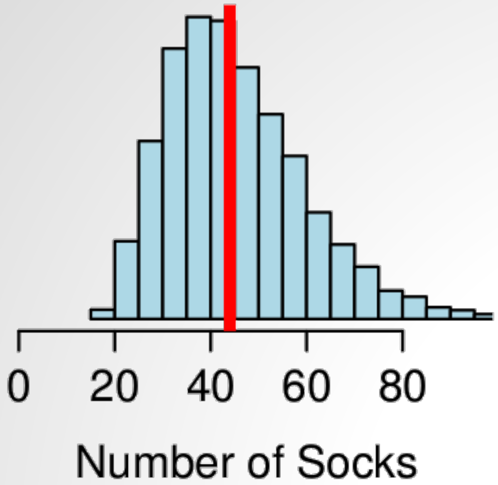
Our best guess: 44

Actual number of socks:
21 x 2 + 3 = 45





Our best guess: 44



Actual number of socks:
$21 \times 2 + 3 = 45$

Error: 

Karl Broman
@kwbroman

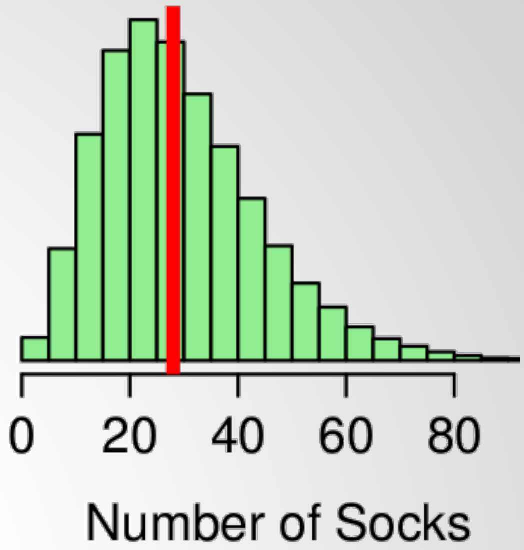
Following

@rabaath @sgrifter There were 21 pairs and 3 singletons. Will spend the rest of the evening working out what my est would have been.

3:00 PM - 17 Oct 2014

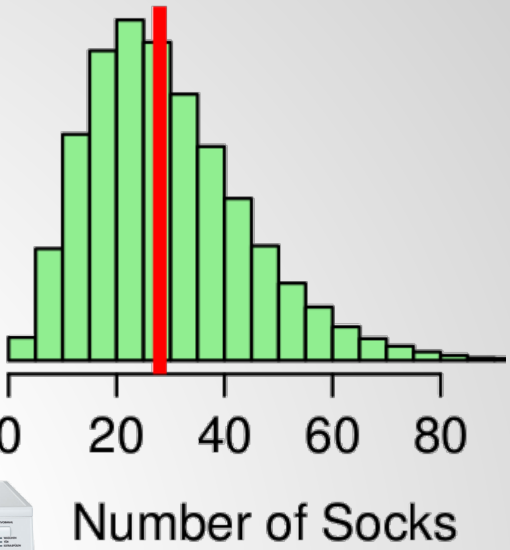
So, what have we done?

- We have specified prior information



So, what have we done?

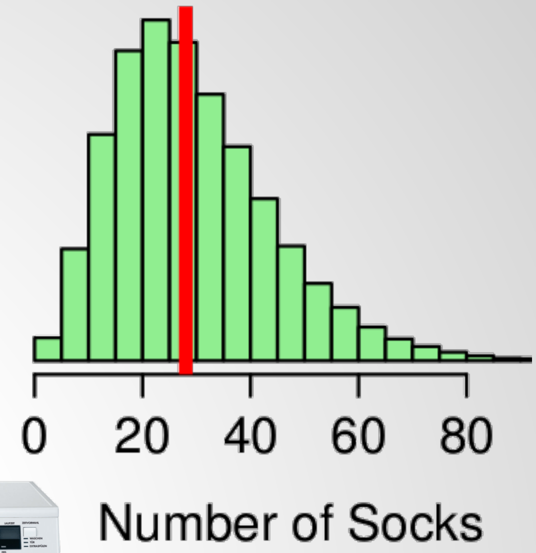
- We have specified prior information



- A generative model

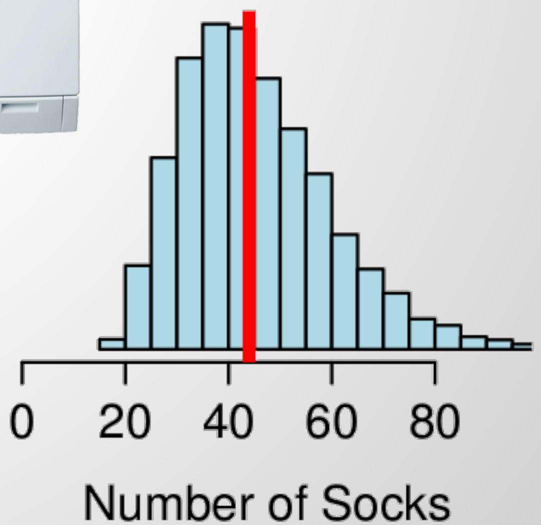
So, what have we done?

- We have specified prior information



- A generative model

- And got out the probability of different parameter values using ABC.

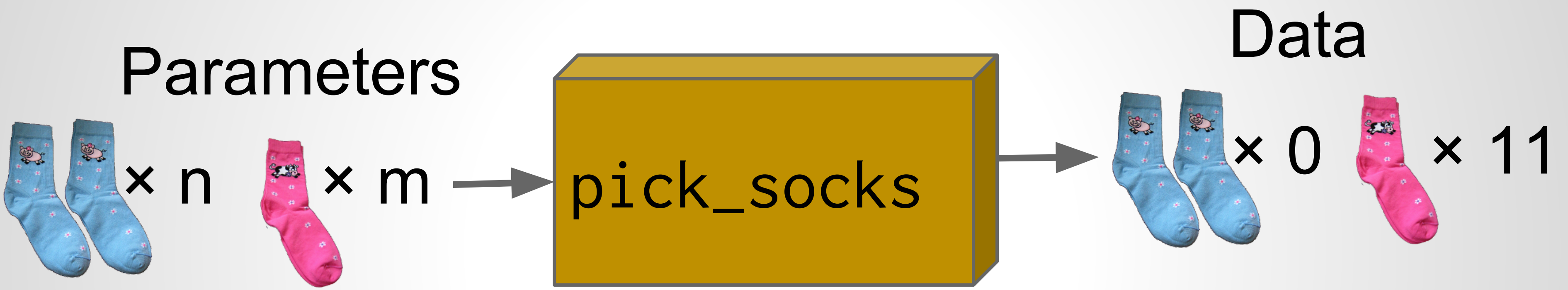


So, what have we done?

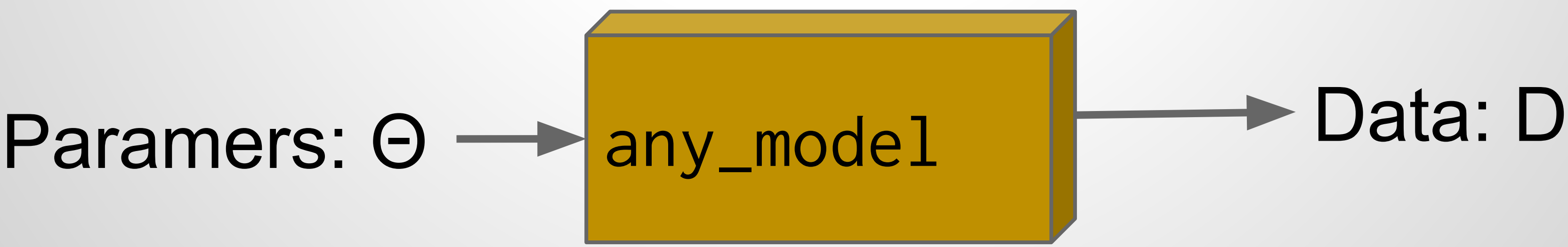
- The example we used was about socks in Karl Broman's washing machine.

So, what have we done?

- The example we used was about socks in Karl Broman's washing machine.



- But the general method works on *any* generative model.



In Conclusion

Approximate Bayesian Computation is

+ Information efficient

In Conclusion

Approximate Bayesian Computation is

+ Information efficient
+ Principled

In Conclusion

Approximate Bayesian Computation is

+ Information efficient
+ Principled
+ Very easy to code up in R

```
sock_sim <- t(replicate(100000, {
  n_socks <- rnbinom(1, mu = 30, size = -30^2 / (30 - 15^2) )
  prop_pairs <- rbeta(1, shape1 = 15, shape2 = 2)
  n_pairs <- round(floor(n_socks / 2) * prop_pairs)
  n_odd <- n_socks - n_pairs * 2

  n_sock_types <- n_pairs + n_odd
  socks <- rep(seq_len(n_sock_types), rep( 2:1, c(n_pairs, n_odd) ))
  picked_socks <- sample(socks, size =  min(11, n_socks))
  sock_counts <- table(picked_socks)

  c(unique = sum(sock_counts == 1), pairs = sum(sock_counts == 2),
    n_socks = n_socks, prop_pairs = prop_pairs)
}))

post_samples <- sock_sim[sock_sim[, "unique"] == 11 &
                         sock_sim[, "pairs" ] == 0 , ]
```

```r
sock_sim <- t(replicate(100000, {
  n_socks <- rnbinom(1, mu = 30, size = -30^2 / (30 - 15^2) )
  prop_pairs <- rbeta(1, shape1 = 15, shape2 = 2)
  n_pairs <- round(floor(n_socks / 2) * prop_pairs)
  n_odd <- n_socks - n_pairs * 2

  n_sock_types <- n_pairs + n_odd
  socks <- rep(seq_len(n_sock_types), rep( 2:1, c(n_pairs, n_odd) ))
  picked_socks <- sample(socks, size =  min(11, n_socks))
  sock_counts <- table(picked_socks)

  c(unique = sum(sock_counts == 1), pairs = sum(sock_counts == 2),
    n_socks = n_socks, prop_pairs = prop_pairs)
}))

post_samples <- sock_sim[sock_sim[, "unique"] == 11 &
                         sock_sim[, "pairs" ] == 0 , ]
```

library(abc) library(EasyABC)

In Conclusion

Approximate Bayesian Computation is

- + Information efficient
- + Principled
- + Very easy to code up in R

In Conclusion

Approximate Bayesian Computation is

+ Information efficient
+ Principled
+ Very easy to code up in R
- So very slooow.

In Conclusion

Approximate Bayesian Computation is

- \+ Information efficient
- \+ Principled
- \+ Very easy to code up in R
- \- So very slooow.

What's wrong with the model?

✉ : rasmus.baath@gmail.com

⌂ : http://www.sumsar.net

: @rabaath

This talk was sponsored by:

By appointment to UseR! 2015, Aalborg.

Rlsberg

COPENHAGEN

1847

PREMIUM BEER

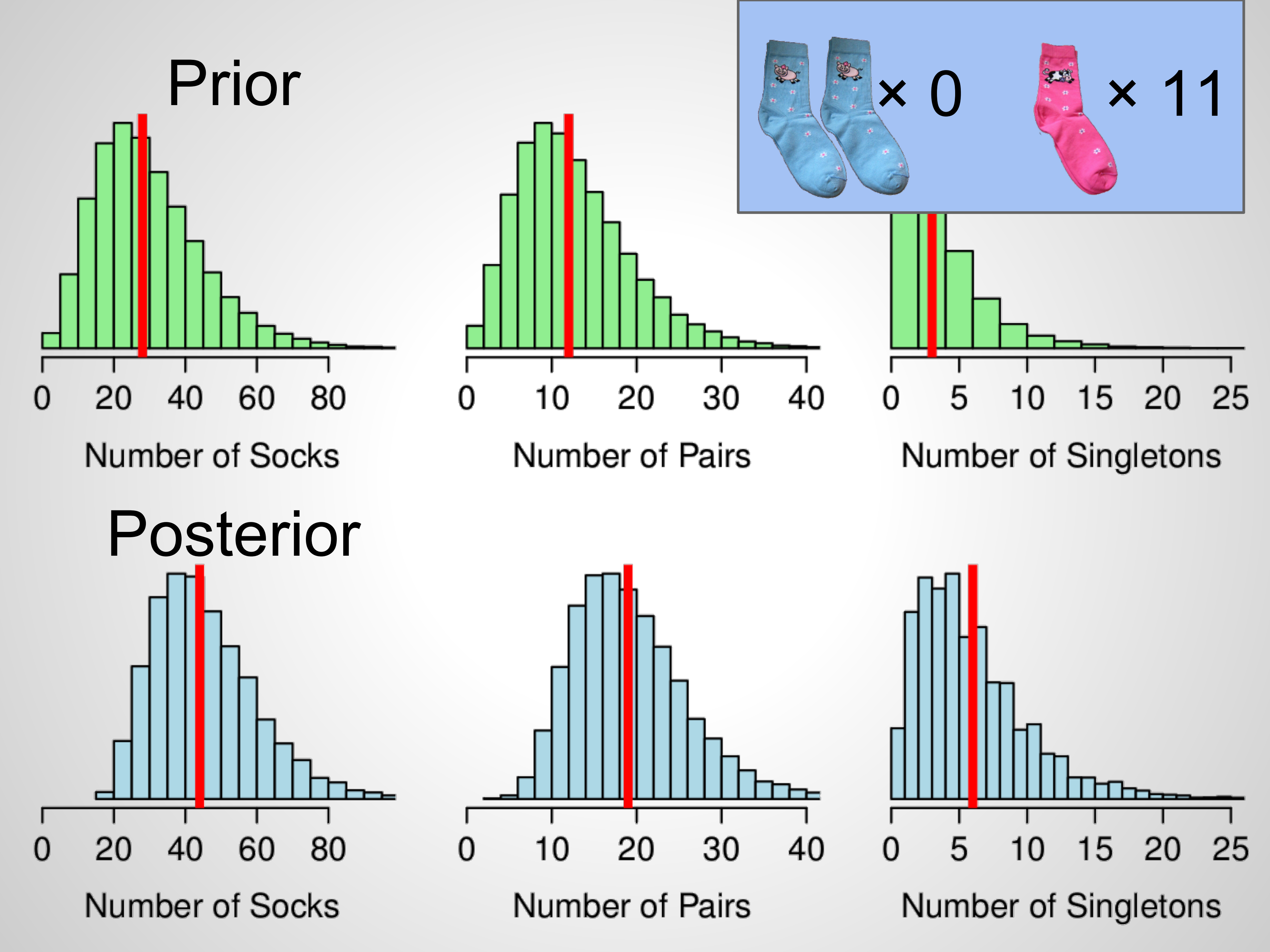
Do we really need priors?

Maximum Likelihood

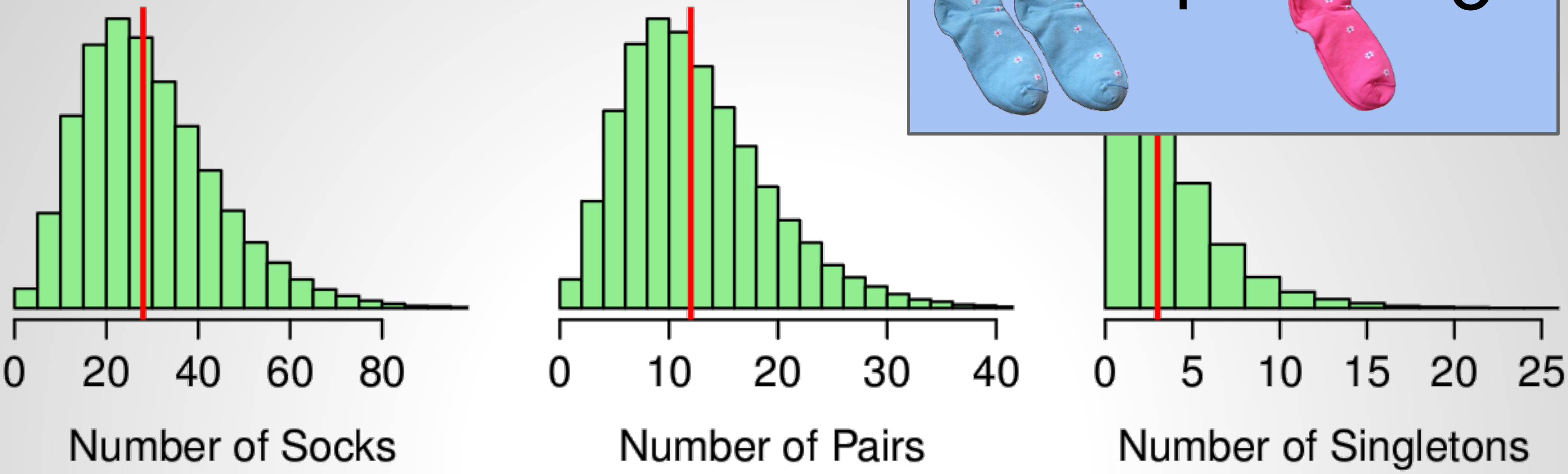
∞


THIS SOCK ESTIMATE
IS TOO DAMN HIGH!
memegenerator.net

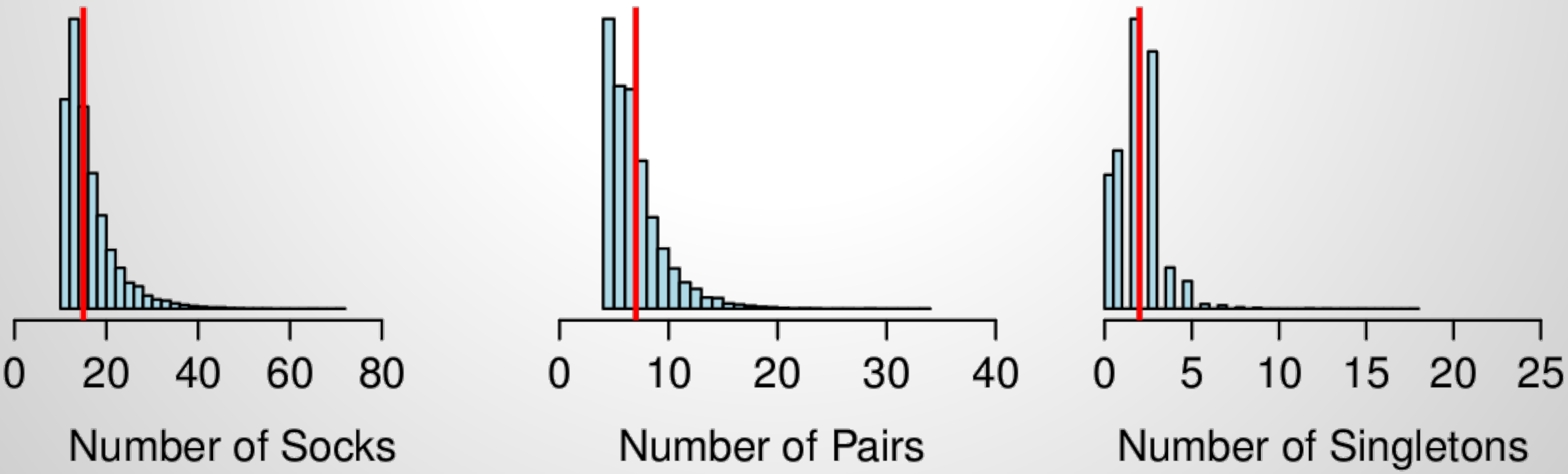
Was the data really necessary?


Prior
× 0
× 11
0 20 40 60 80
Number of Socks
0 10 20 30 40
Number of Pairs
0 5 10 15 20 25
Number of Singletons
Posterior
0 20 40 60 80
Number of Socks
0 10 20 30 40
Number of Pairs
0 5 10 15 20 25
Number of Singletons

Prior



Posterior



The Annals of Statistics
1984, Vol. 12, No. 4, 1151–1172

SPECIAL INVITED PAPER

BAYESIANLY JUSTIFIABLE AND RELEVANT FREQUENCY CALCULATIONS FOR THE APPLIED STATISTICIAN[1]

BY DONALD B. RUBIN

University of Chicago

A common reaction among applied statisticians is that the Bayesian statistician's energies in an applied problem must be directed at the a priori elicitation of *one* model specification from which an optimal design and all inferences follow automatically by applying Bayes's theorem to calculate conditional distributions of unknowns given knowns. I feel, however, that the applied Bayesian statistician's tool-kit should be more extensive and include tools that may be usefully labeled frequency calculations. Three types of Bayesianly justifiable and relevant frequency calculations are presented using examples to convey their use for the applied statistician.

1. Introduction. My purpose here is to discuss three important uses of frequency calculations for the applied Bayesian statistician: (1) for understanding, communicating and scientifically validating Bayesian statements, (2) for examining operating characteristics of Bayesian inferences derived from general

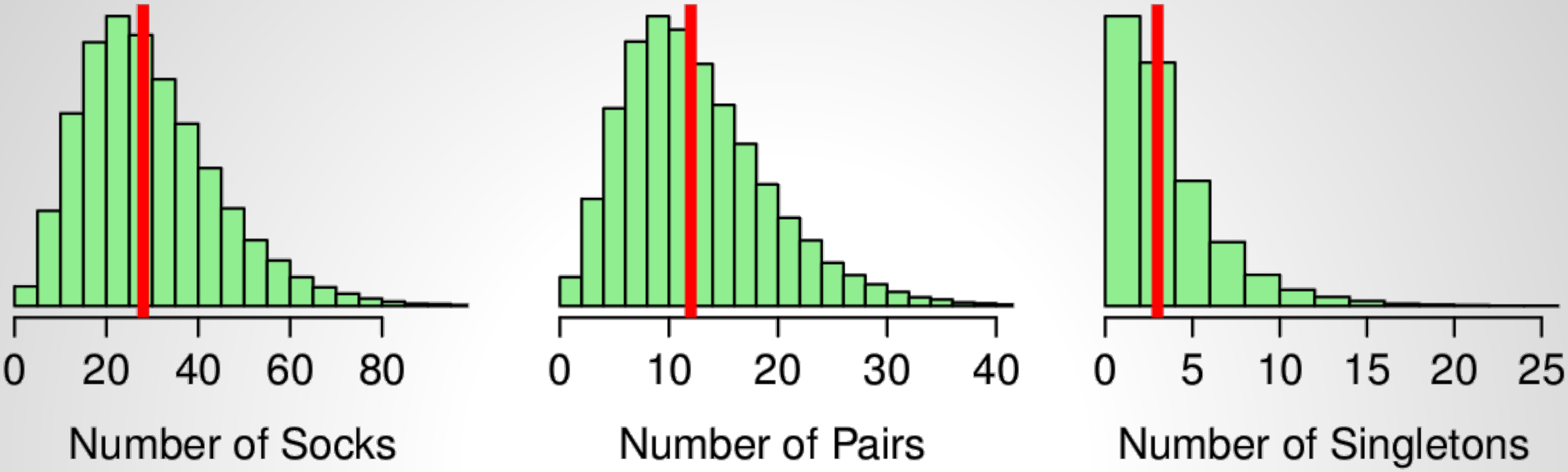

Location Of Socks
Neatly laid out in dresser
God knows

Prior Sock Distributions

× ?

× ?

Prior



Posterior