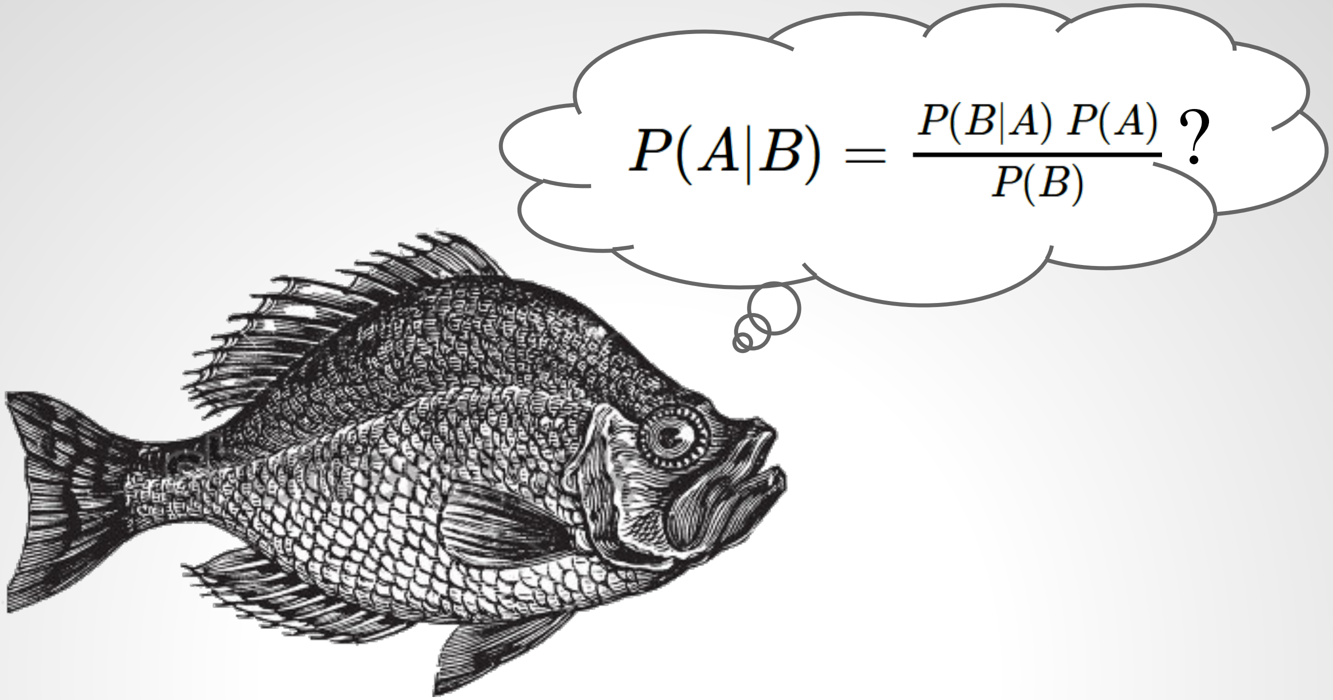
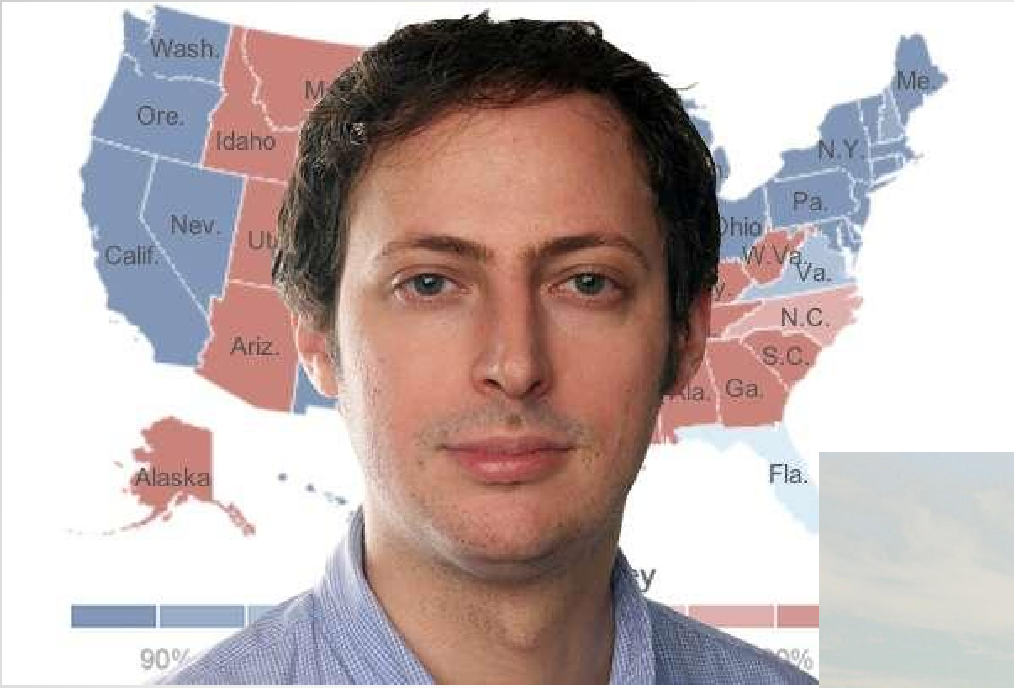
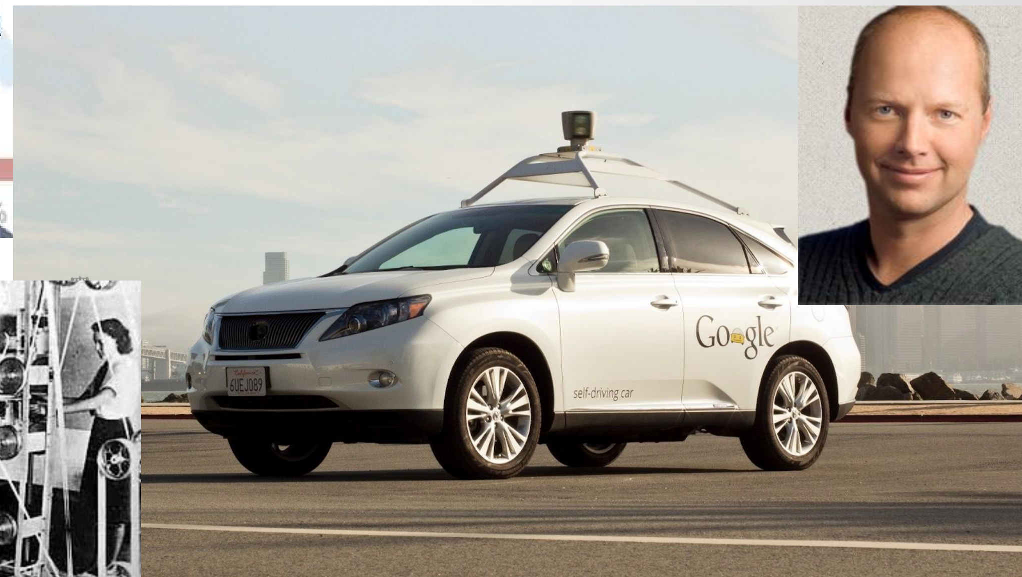
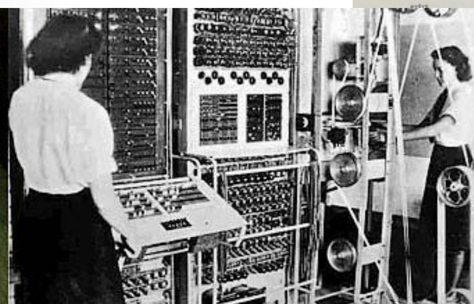


Introduction to Bayesian Data Analysis with R.

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

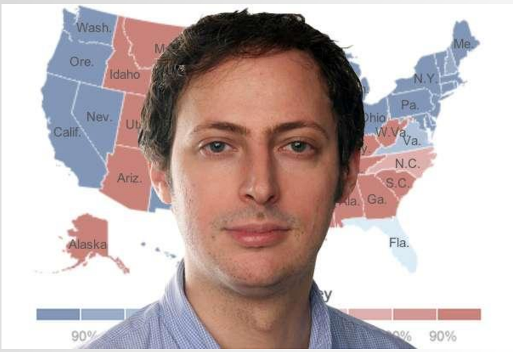


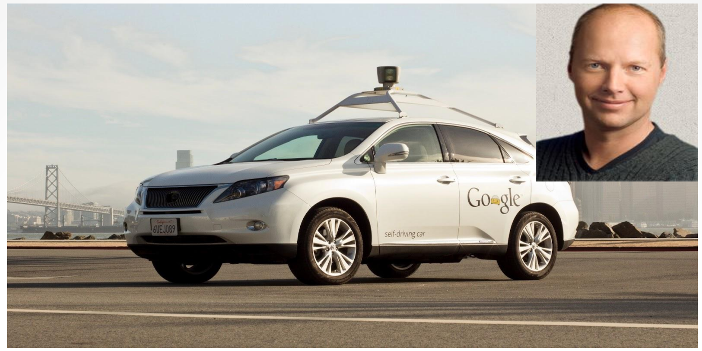


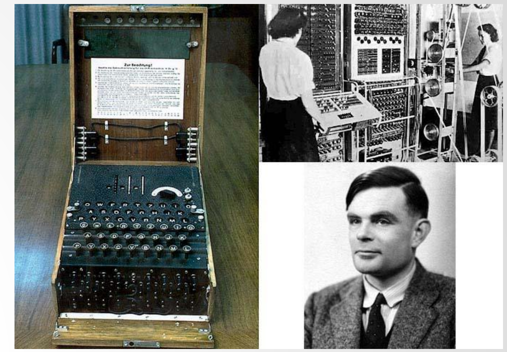


Source: I borrowed these three examples from a presentation. But which presentation I can't remember or find. I you know what presentation it could have been, please let me know and I will credit it here.

What do these have in common?







- Complex problems
- Large inherent uncertainty that needs to be quantified.
- Requires efficient integration of many sources of information.
- They all use Bayesian data analysis.

Bayesian data analysis is a great <u>tool</u>!

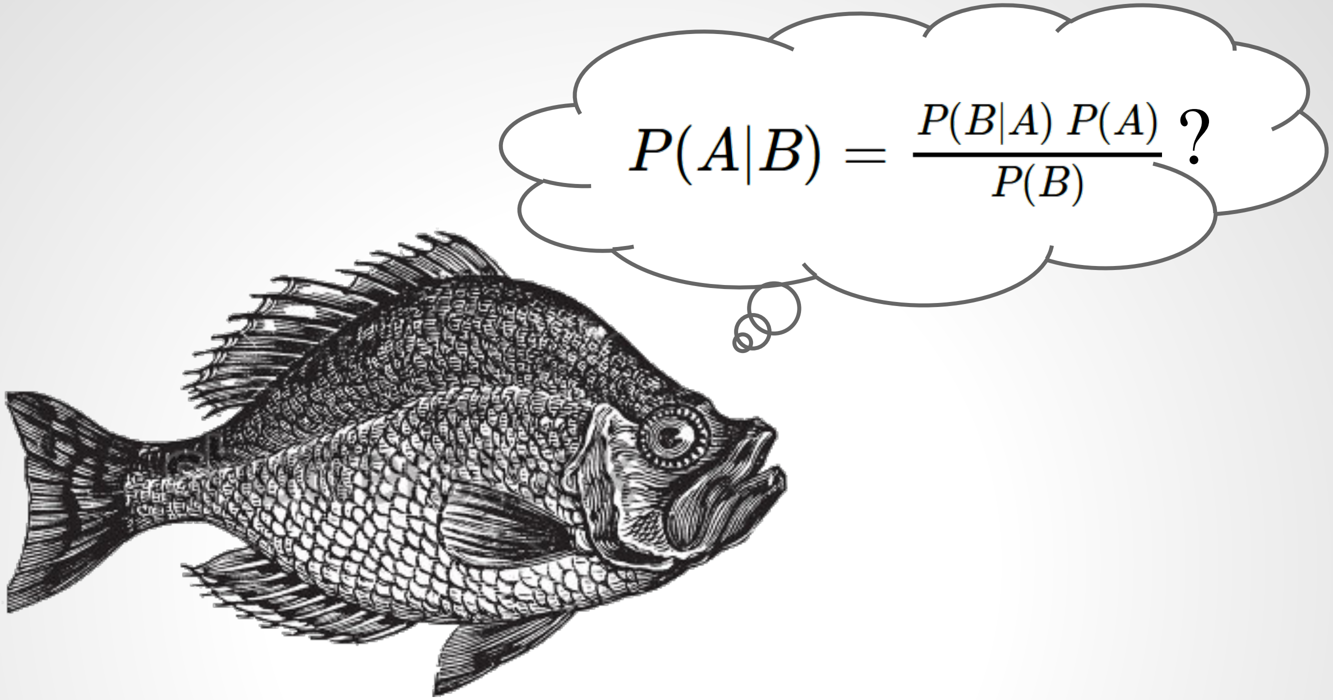
… and R is a great tool for doing Bayesian data analysis.

But if you google “Bayesian” you get philosophy:

Subjective vs Objective

Frequentism vs Bayesianism

p-values vs subjective probabilities



Bayesian data analysis
What? Why? How?

Overview of this tutorial

- **What** is Bayesian data analysis?

Prediction contest

- **Why** use Bayesian data analysis?

Exercises

- **How** to interpret and perform a Bayesian data analysis in R?

More Exercises

Why am I here?

- I use Bayesian methods in my research at Lund University where I also run a network for people interested in Bayes.

- I'm working on an R-package to make simple Bayesian analyses simple to run.

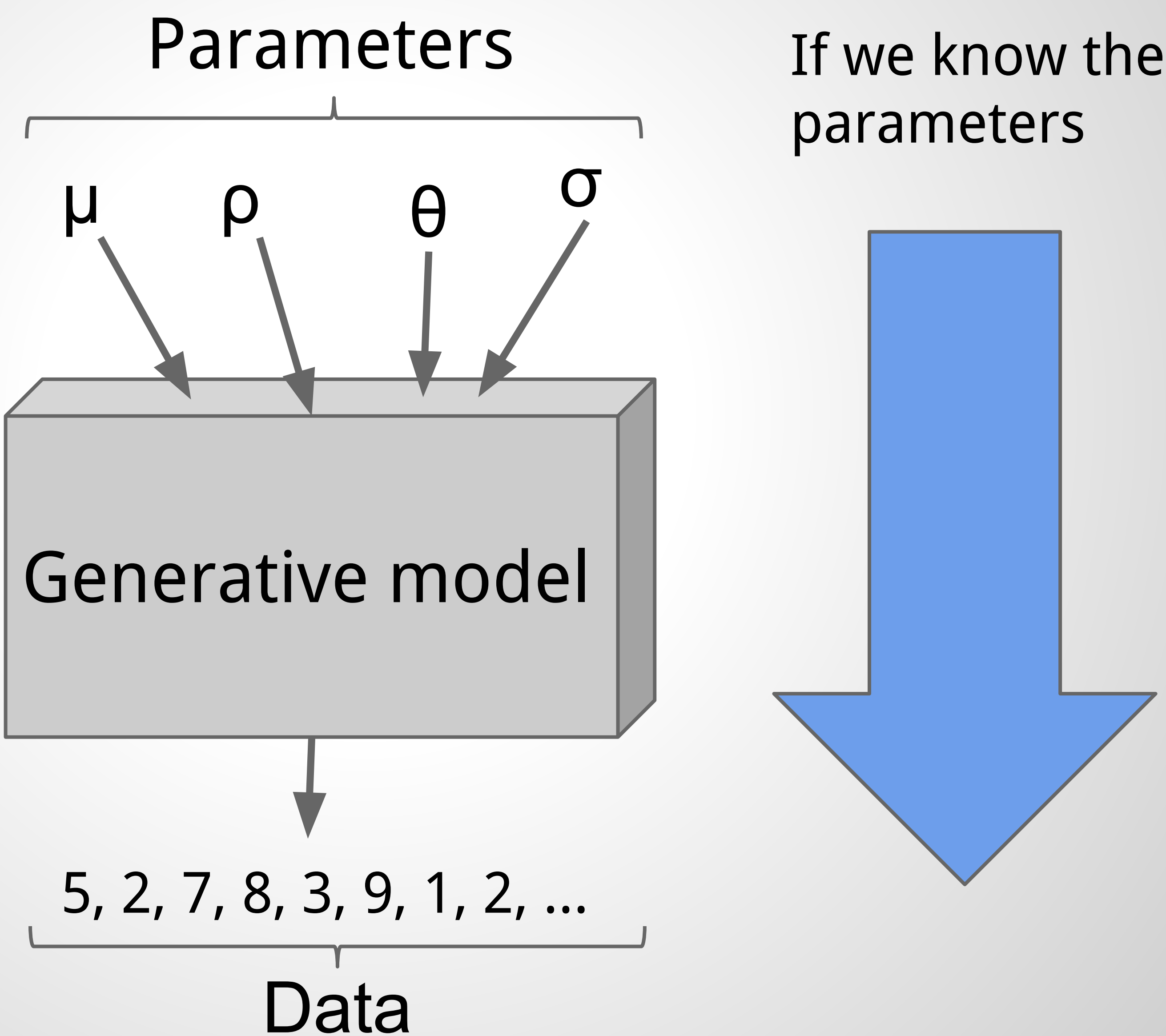
- I blog about Bayesian data analysis.

www.sumsar.net

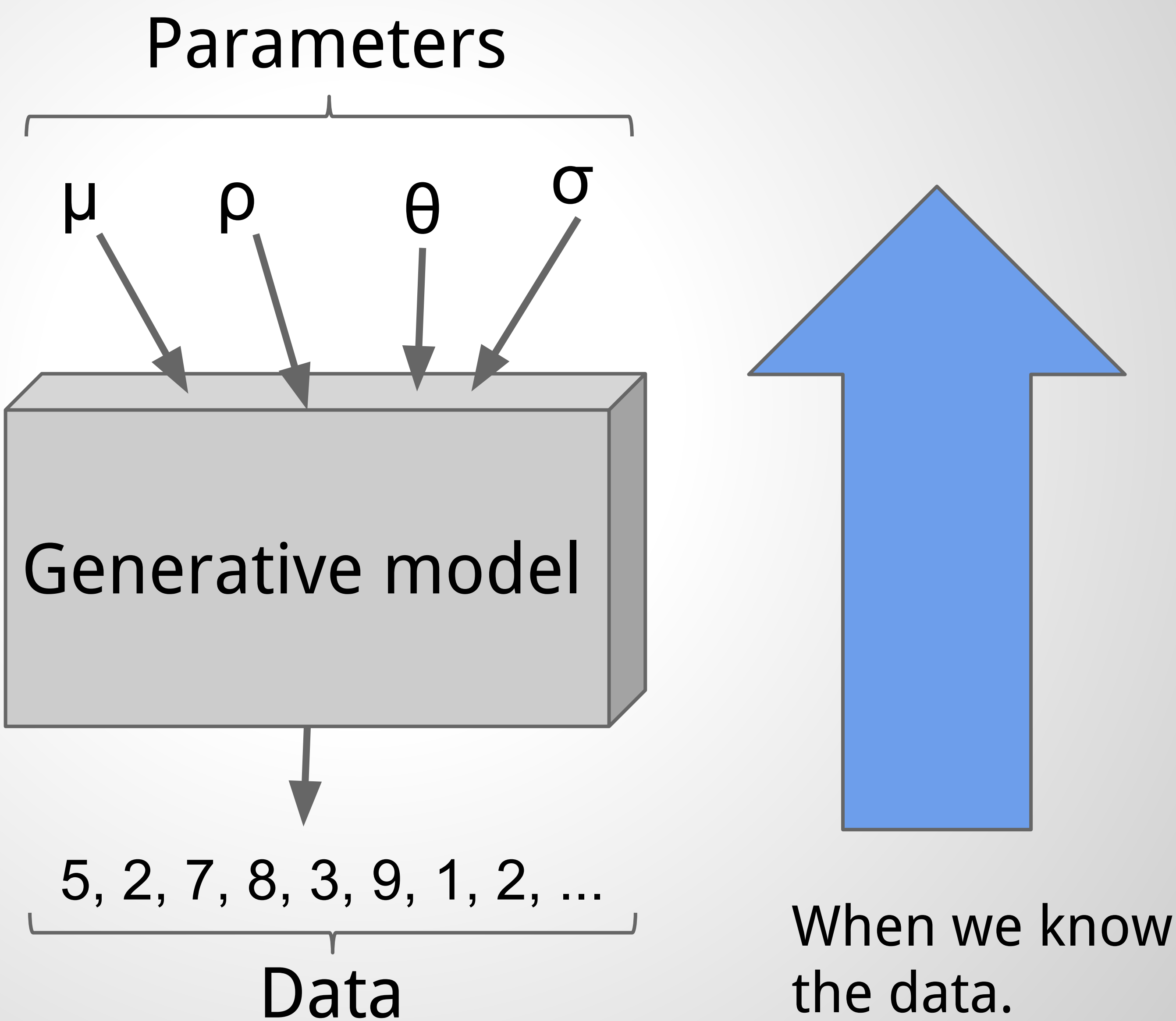
What is Bayesian data analysis?

- It is when you use **probability** to represent **uncertainty** in **all parts** of a statistical model.
- A flexible extension of maximum likelihood.
- Potentially the most **information-efficient** method to fit a statistical model.
  (But potentially also the most computationally intensive method…)

Bayesian models as *generative* models

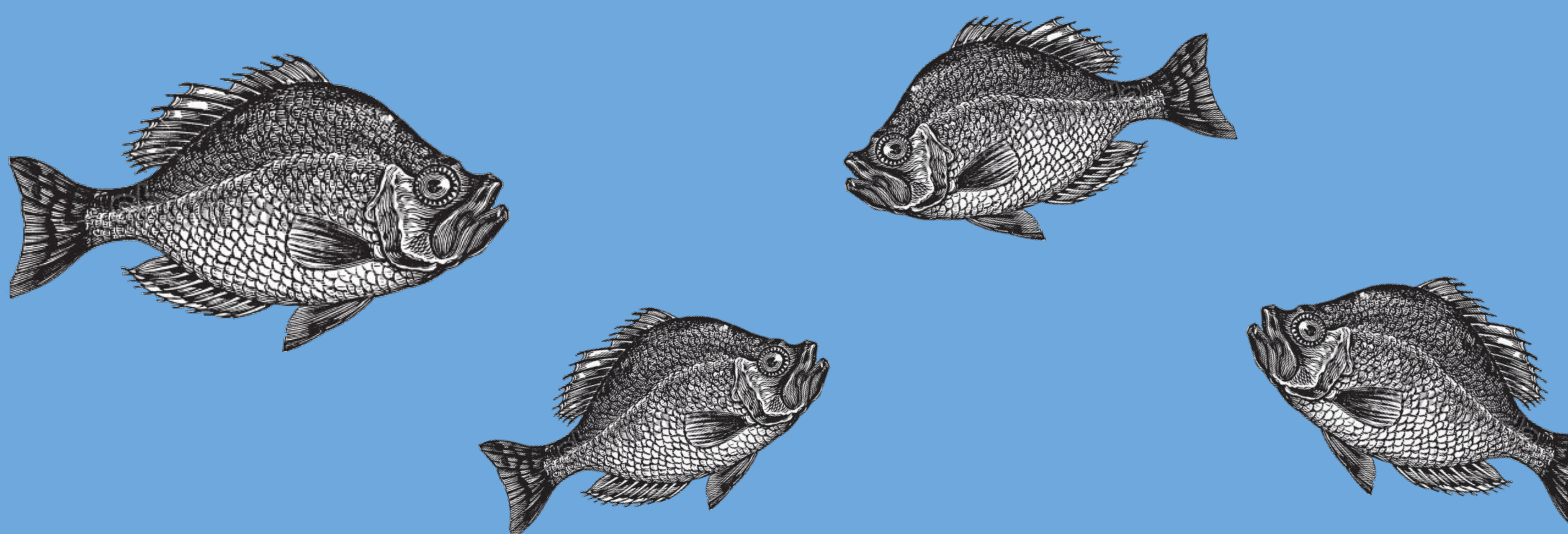


Bayesian models as *generative* models



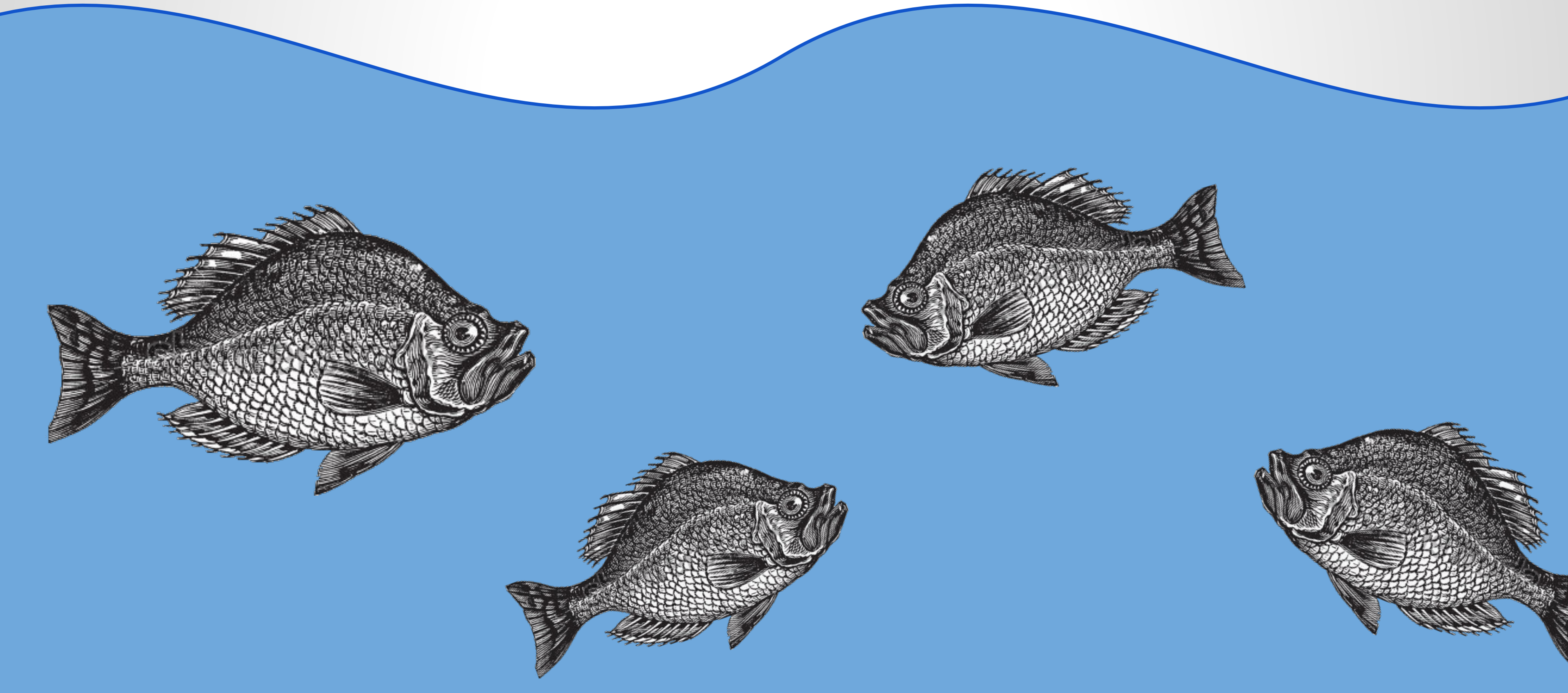
How many fish are in the lake?

- An actual problem in *Abundance estimation*. Use in, for example, wildlife management.
- Also other uses, for example, to estimate how many DKK 1,000 bills are in circulation.



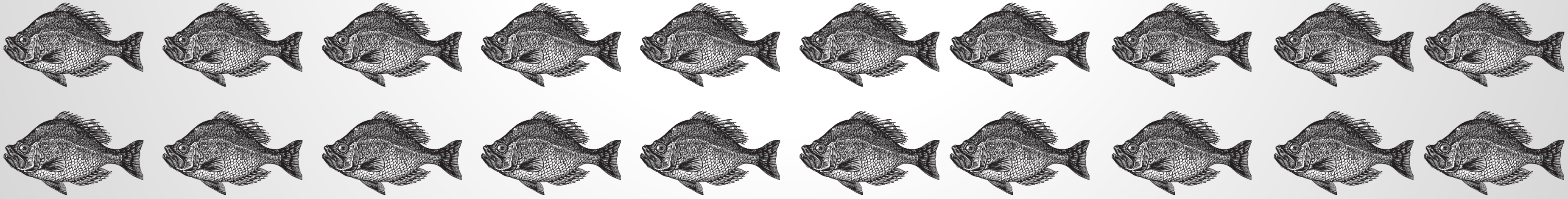
How many fish are in the lake?

- The problem: We can't catch them all.
- But we can catch some of them...



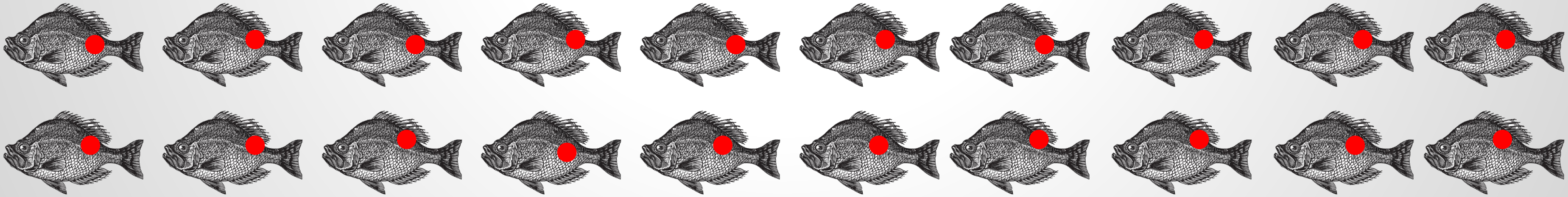
Mark and Re-capture

1. Catch a couple of fish.
2. Mark them and throw them back.



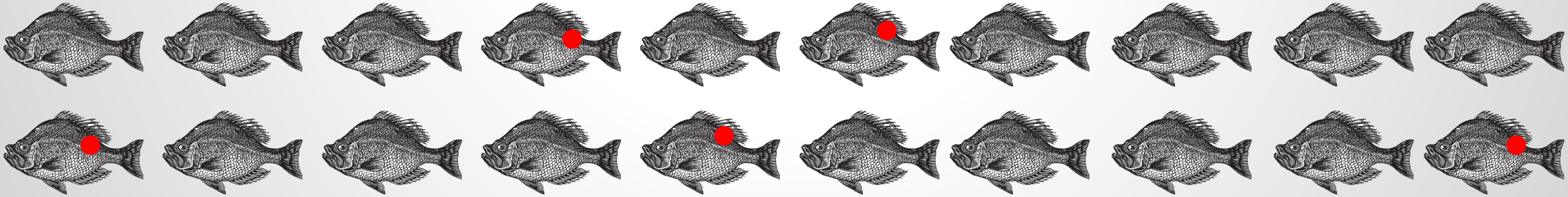
Mark and Re-capture

1. Catch a couple of fish.
2. Mark them and throw them back.



Mark and Re-capture

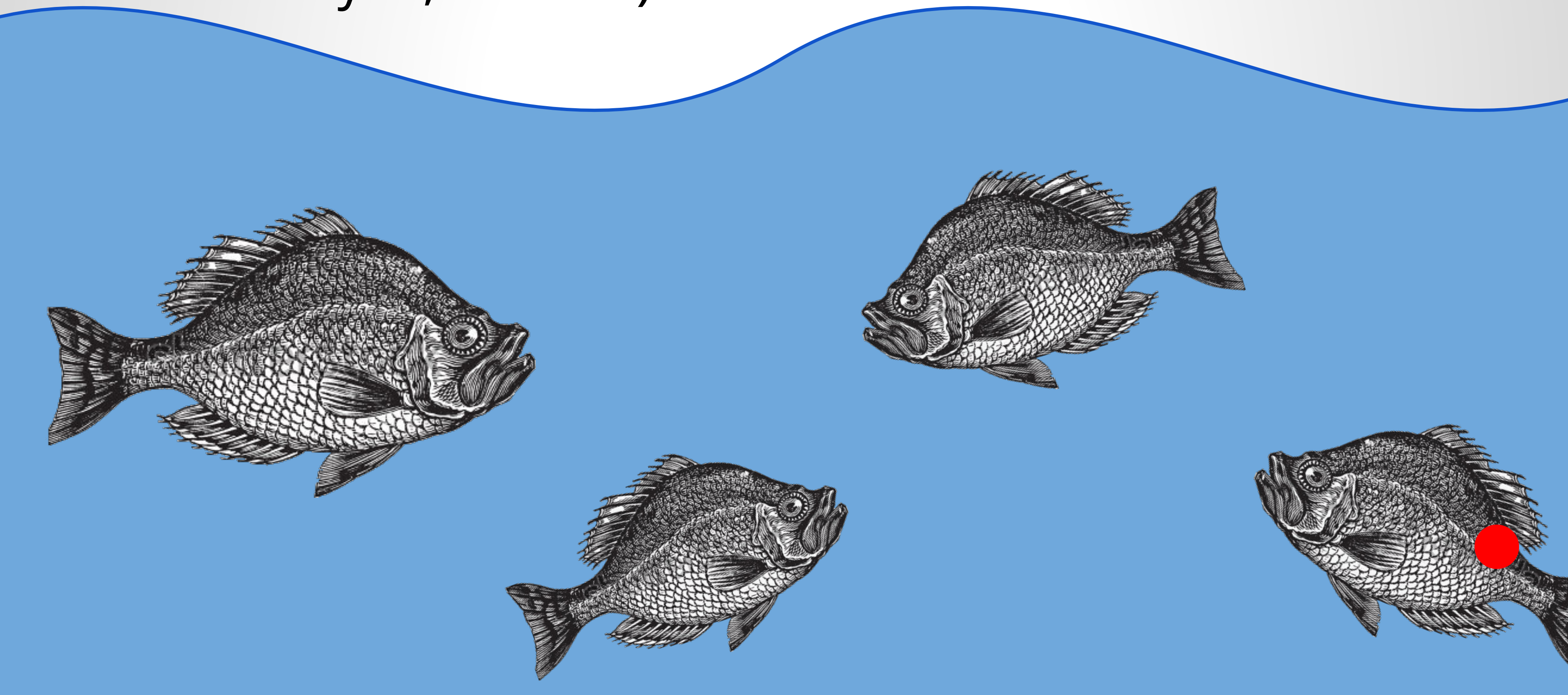
1. Catch a couple of fish.
2. Mark them and throw them back.
3. At a later point, catch a couple of fish again.
4. Count how many are marked.

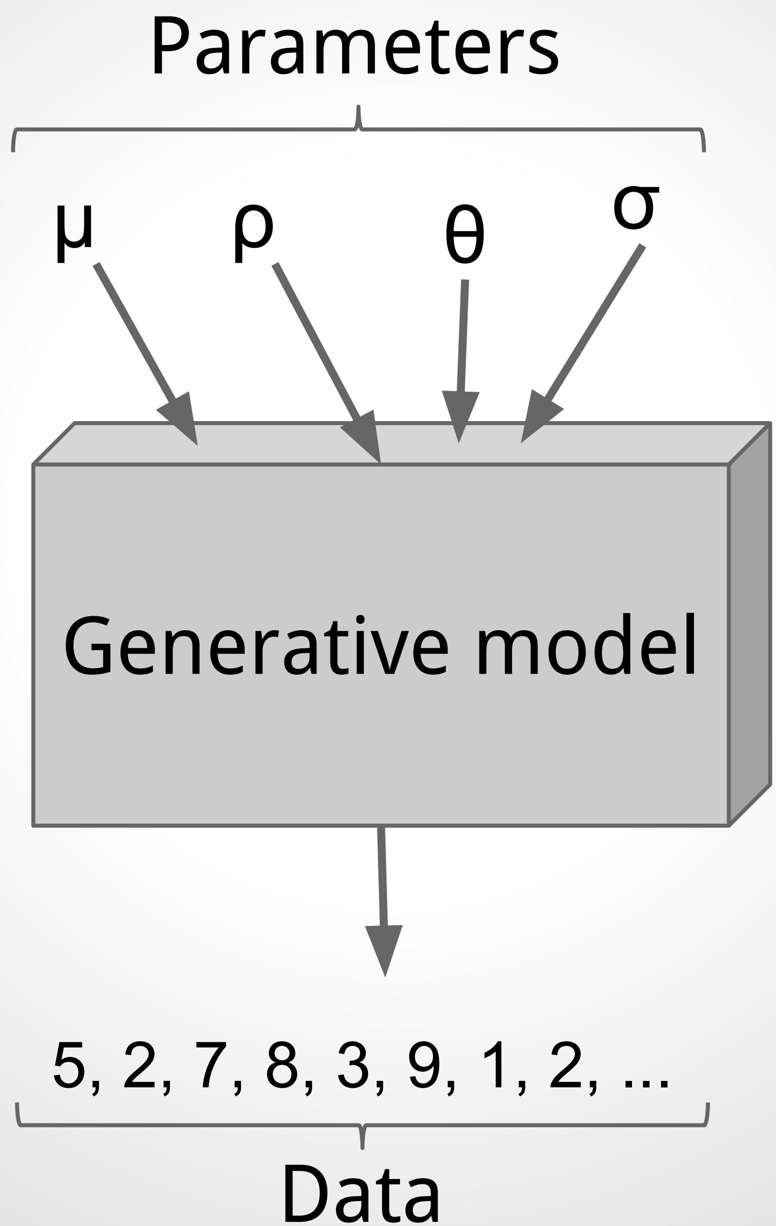


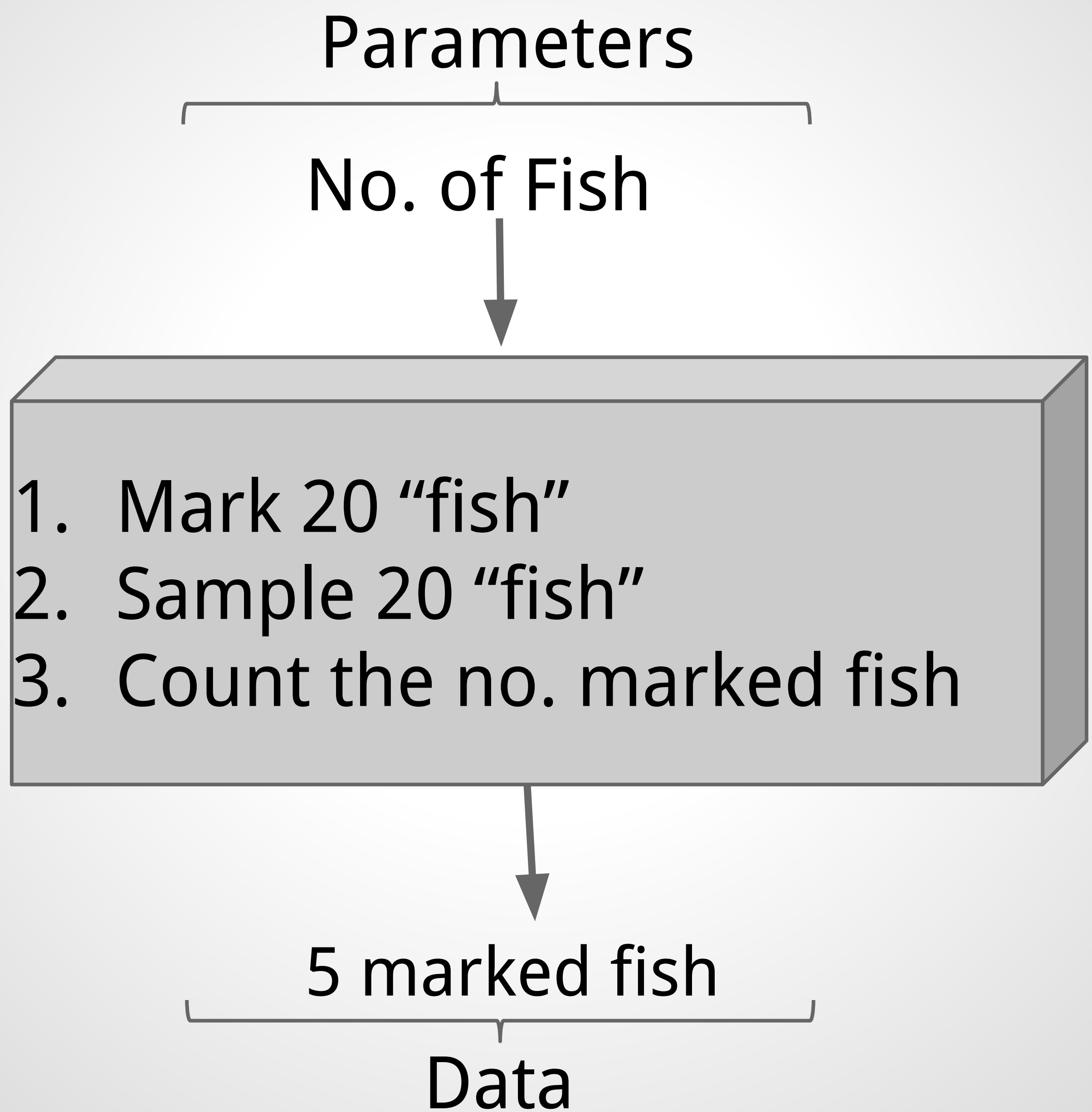
20 were marked and five out of the 20 that were caught the second time were marked.

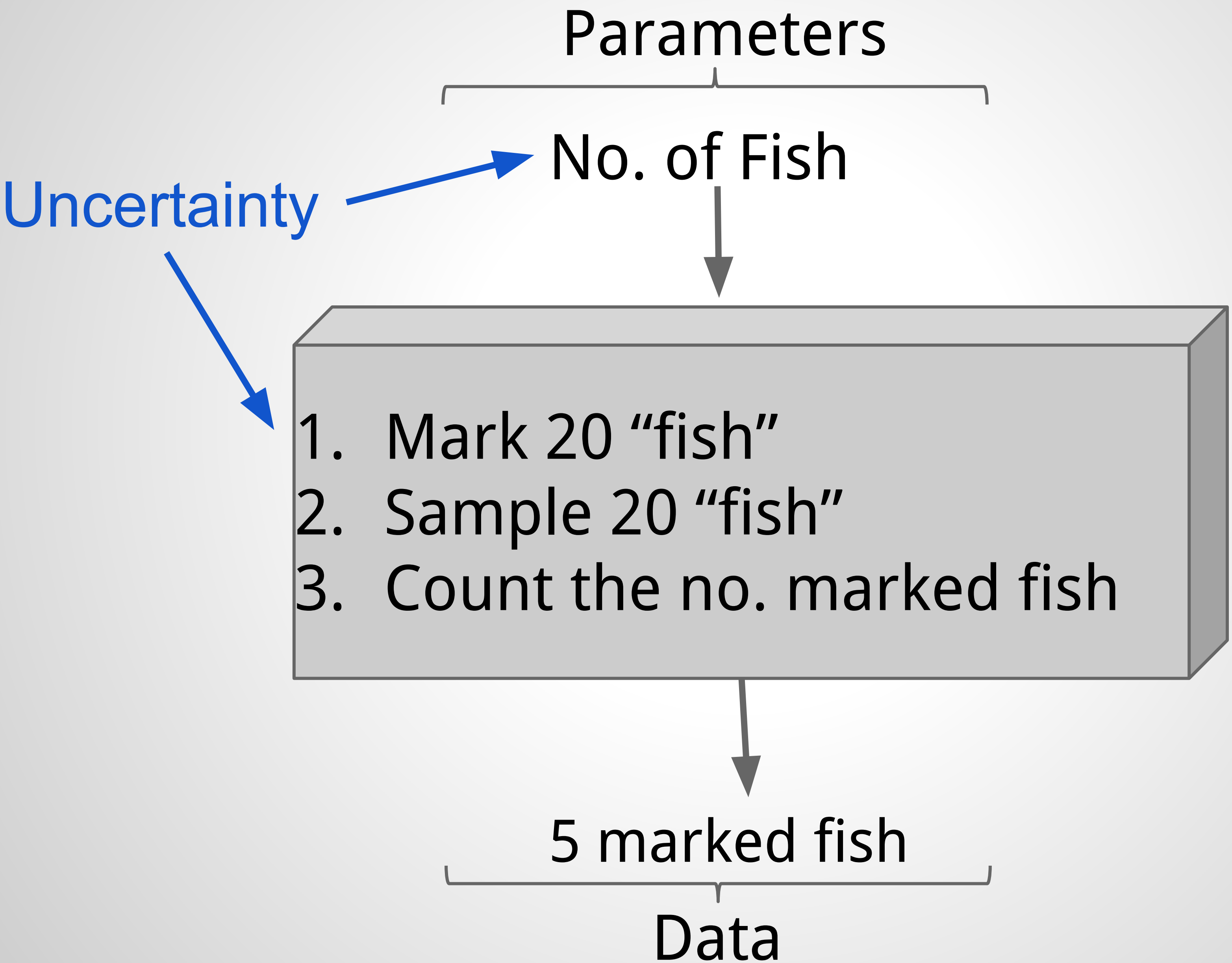
So, how many fish are in the lake?

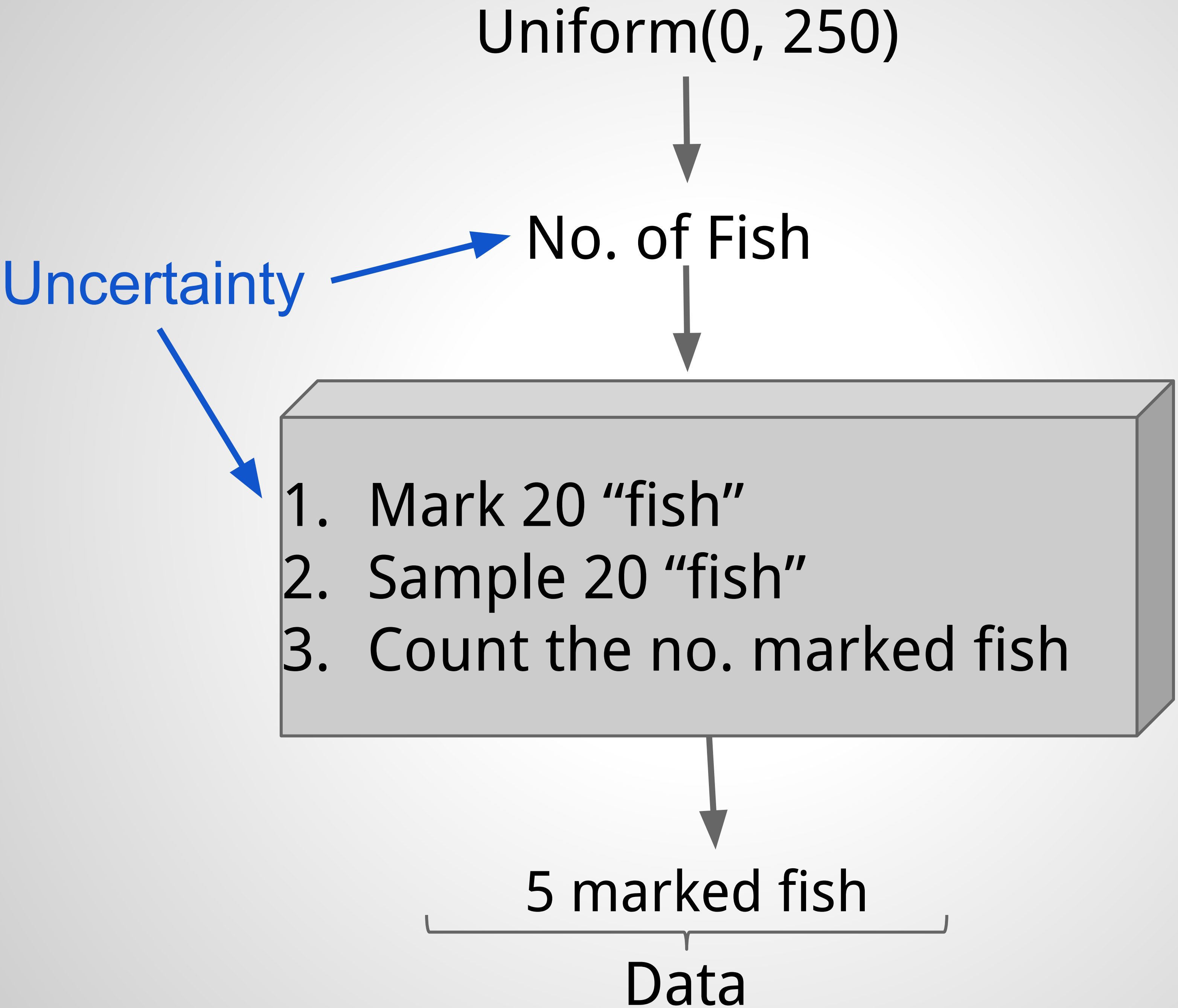
- What are the *probable* number of fish in the lake?
- We have almost already described the solution! (If we know about Bayesian Data Analysis, that is...)




Parameters
μ
ρ
θ
σ
Generative model
5, 2, 7, 8, 3, 9, 1, 2, ...
Data

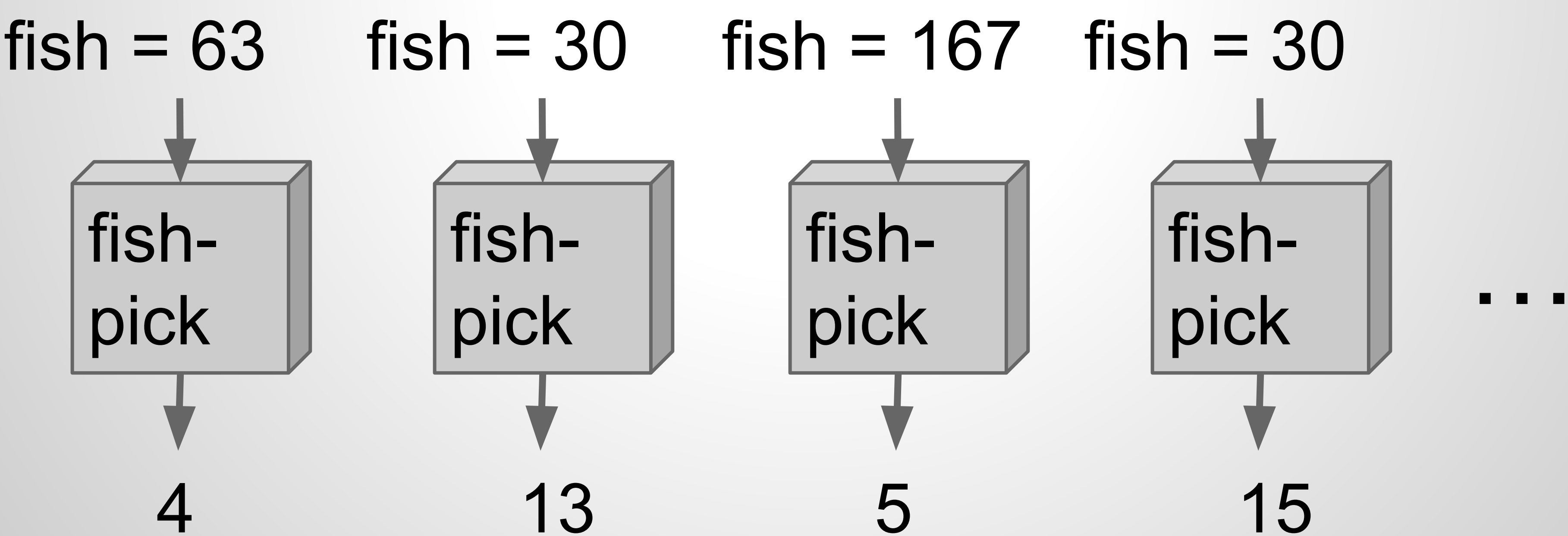

Parameters
No. of Fish
1. Mark 20 “fish”
2. Sample 20 “fish”
3. Count the no. marked fish
5 marked fish
Data


Parameters
No. of Fish
Uncertainty
1. Mark 20 “fish”
2. Sample 20 “fish”
3. Count the no. marked fish
5 marked fish
Data


Uniform(0, 250)
No. of Fish
Uncertainty
1. Mark 20 “fish”
2. Sample 20 “fish”
3. Count the no. marked fish
5 marked fish
Data

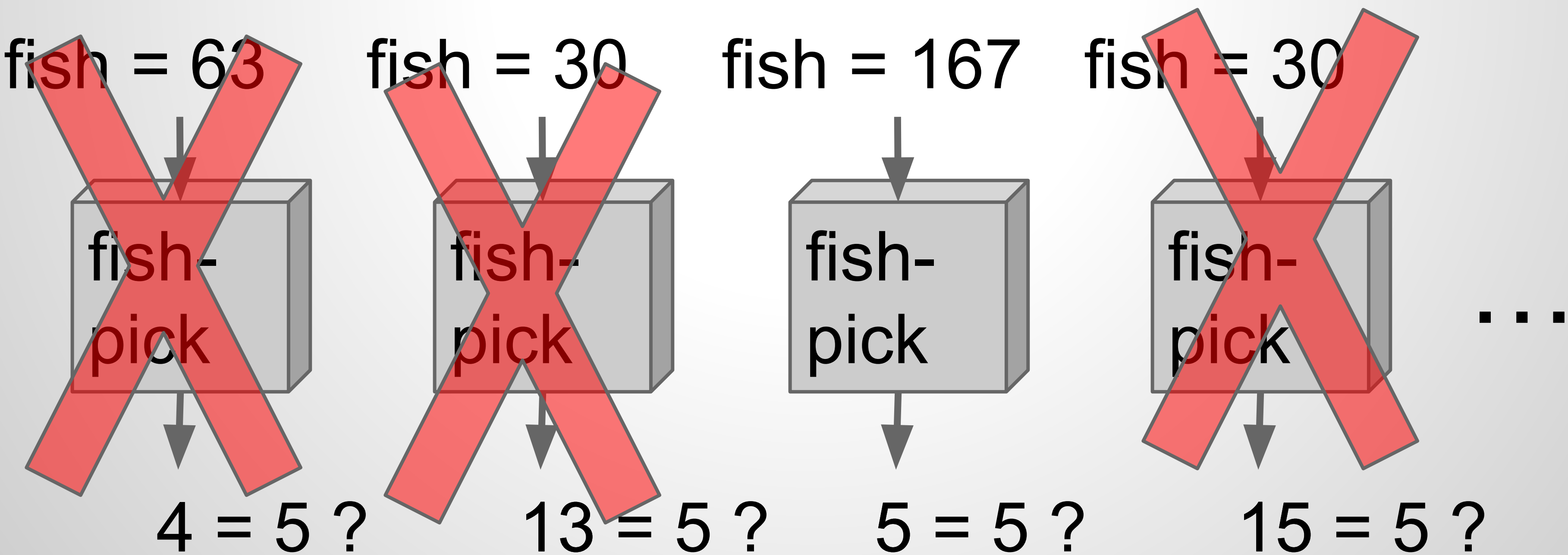
One simple way of fitting the model

1. Draw a large random sample from the “prior” probability distribution on the parameters. Here, for example:
   `no_fish: [63, 30, 167, 30, 164, 222, 225, 42, 122,…]`
2. Plug in each draw into the generative model which generates a vector of “fake” data. For example:



One simple way of fitting the model

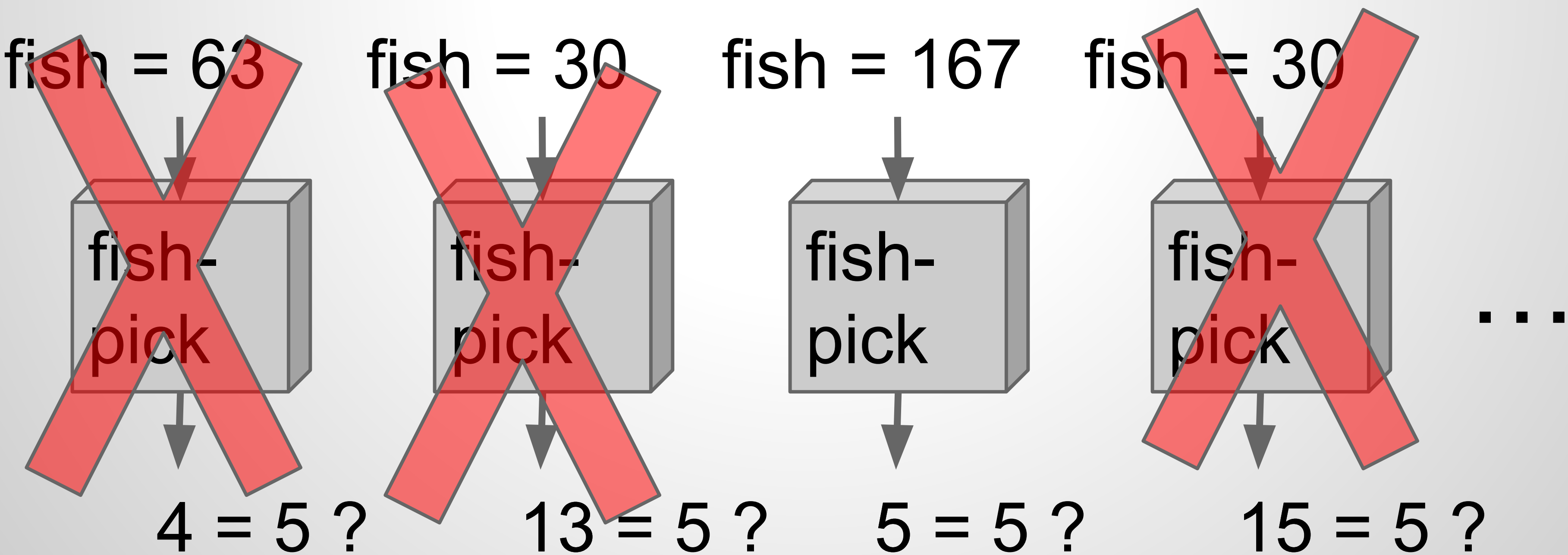
3. Keep only those parameter values that generated the data that was actually observed.



One simple way of fitting the model

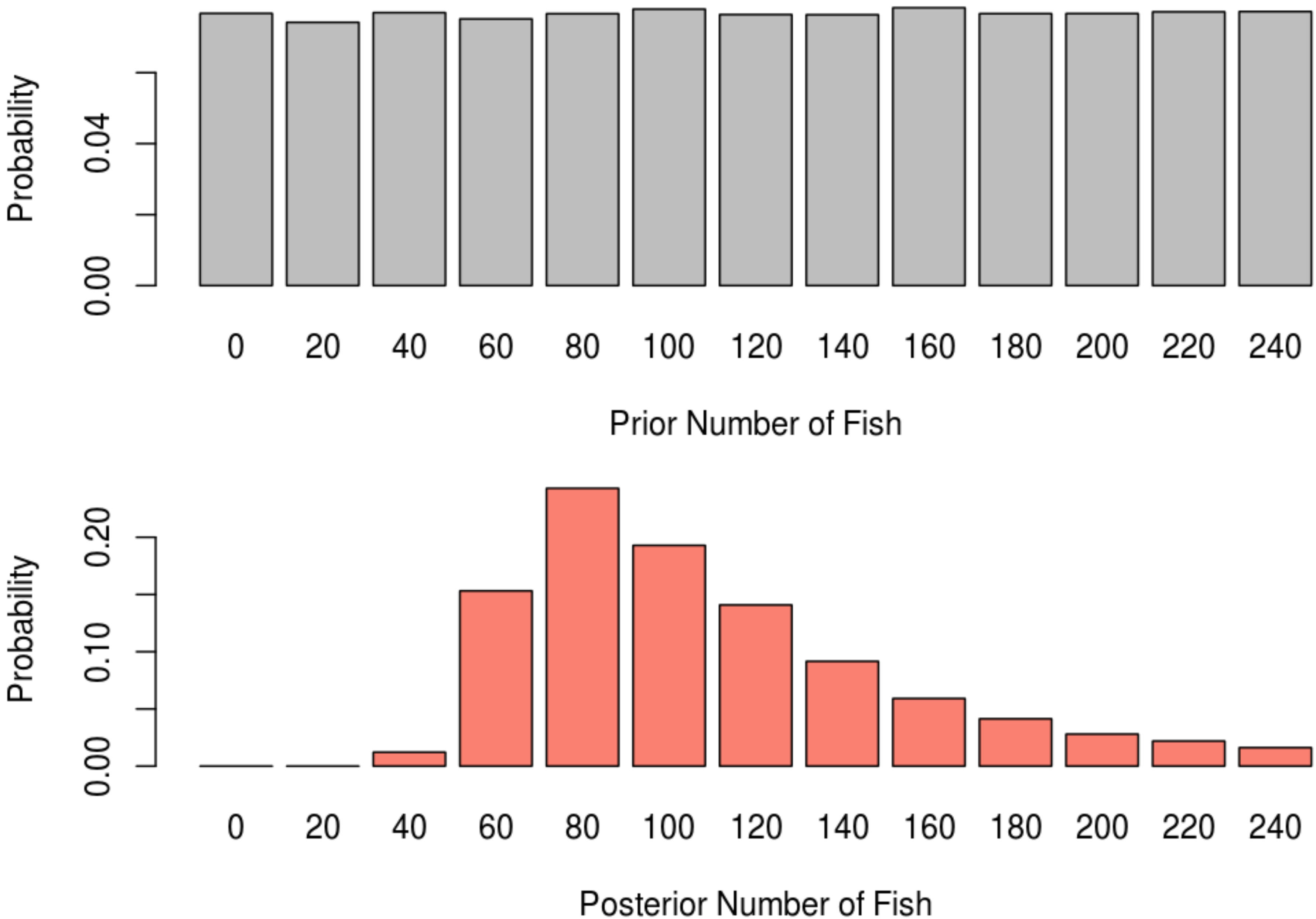
3. Keep only those parameter values that generated the data that was actually observed.
4. The distribution of the retained parameters now represent the probability that the data was produced by a certain parameter value. For example:

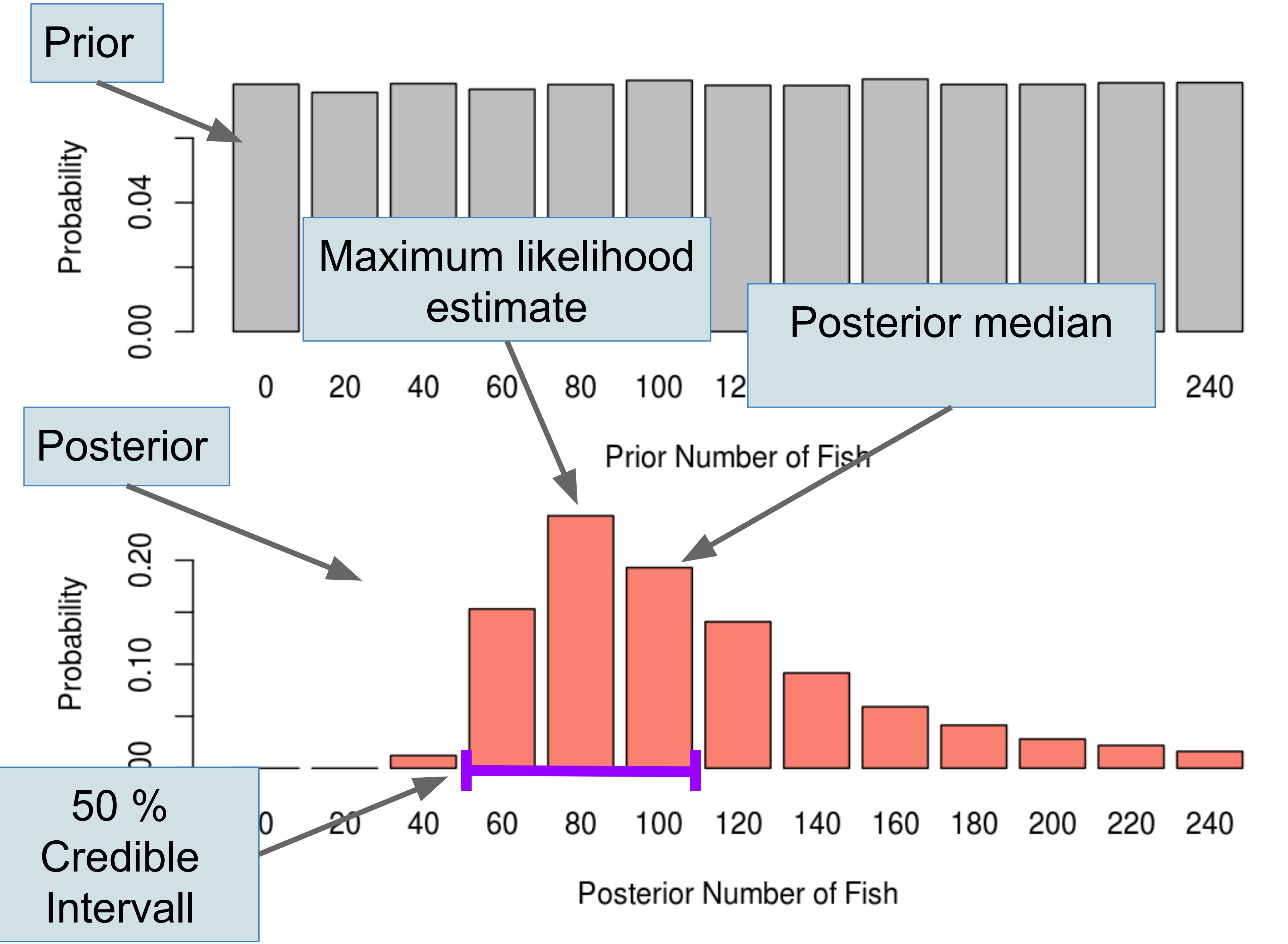
```
[167, 135, 148, 90, 162, 88, 98, 110, 176, … ]
```



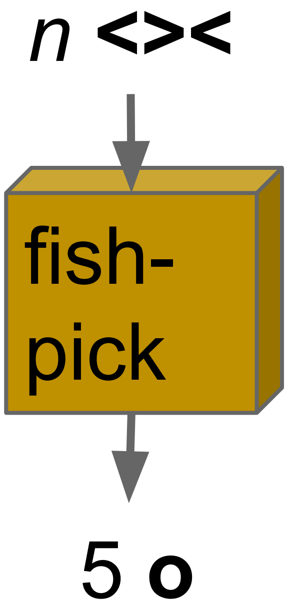
Time for a demonstration

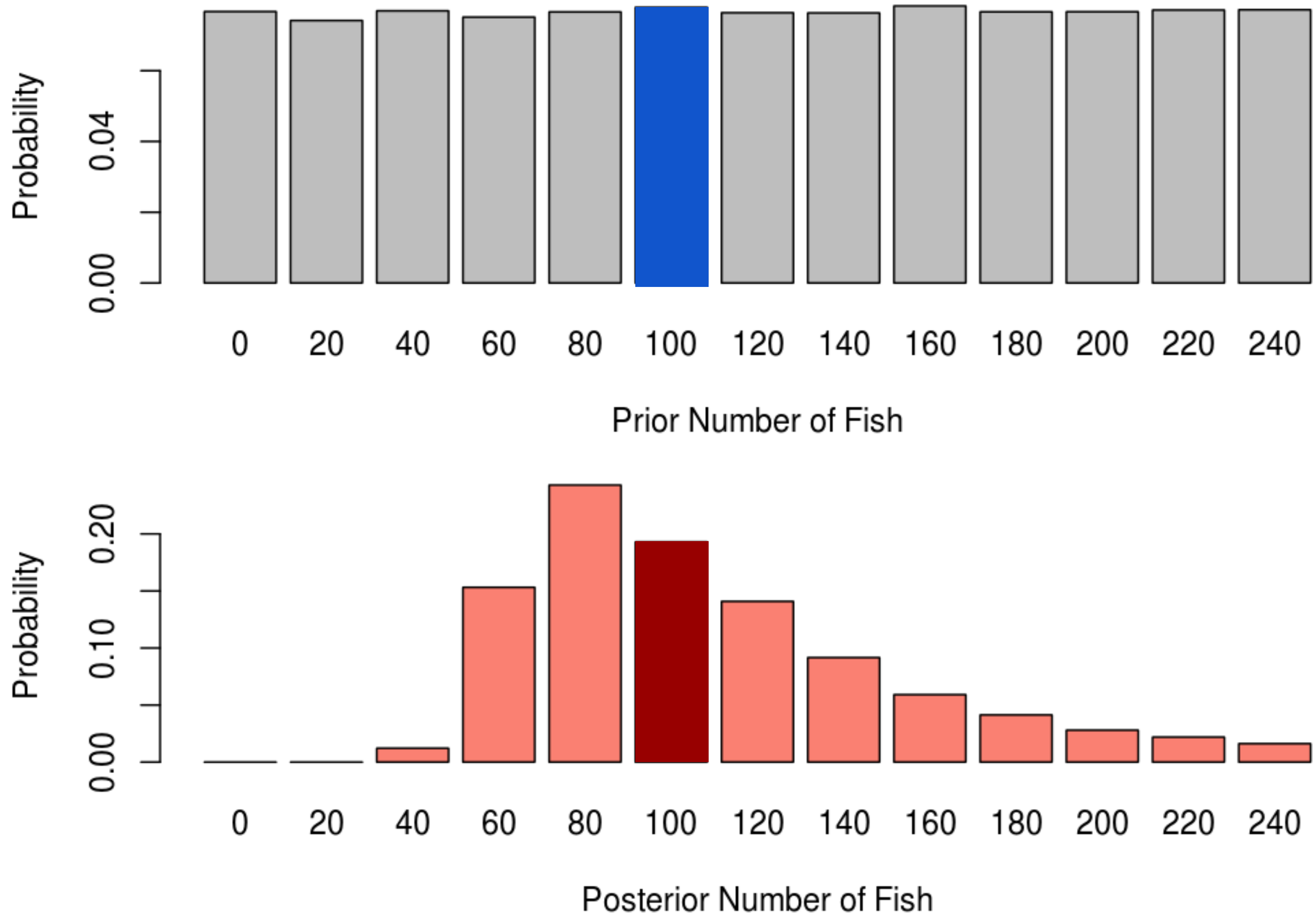
The script that was “live coded” can be found here:
http://rpubs.com/rasmusab/live_coding_user_2015_bayes_tutorial


Probability
0.00
0.04
0 20 40 60 80 100 120 140 160 180 200 220 240
Prior Number of Fish
Probability
0.00
0.10
0.20
0 20 40 60 80 100 120 140 160 180 200 220 240
Posterior Number of Fish

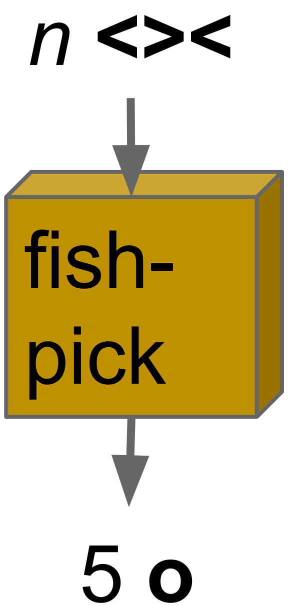

Prior
Probability
0.00
0.04
0
20
40
60
80
100
240
Maximum likelihood estimate
Posterior median
Prior Number of Fish
Posterior
Probability
0.10
0.20
20
40
60
80
100
120
140
160
180
200
220
240
50 % Credible Intervall
Posterior Number of Fish

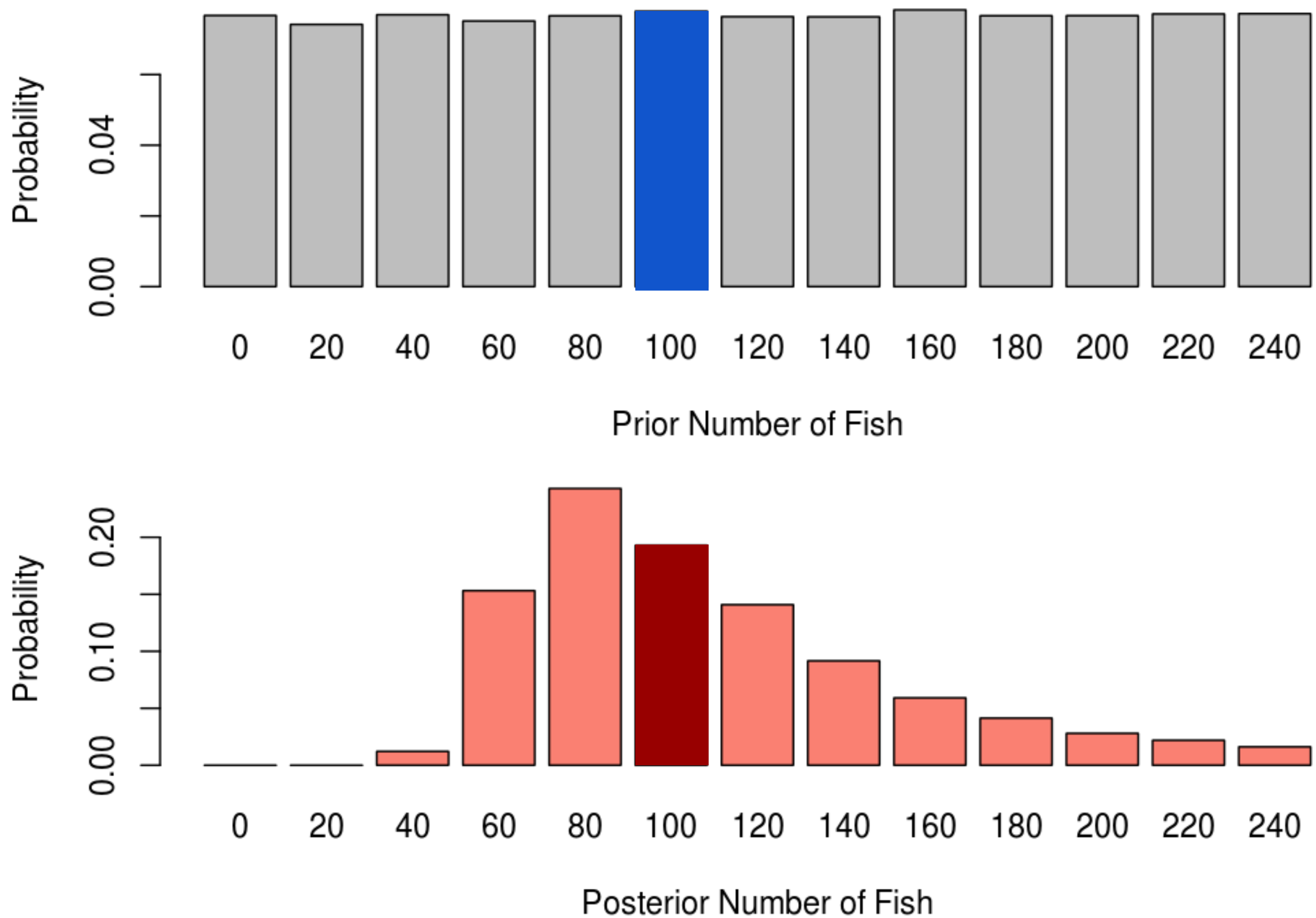
P(100 <>< | 5 o) ∝ P(100 <><) × P(5 o | 100 <><)

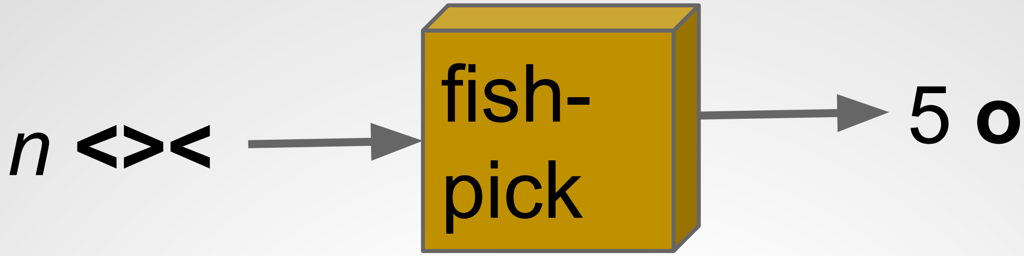




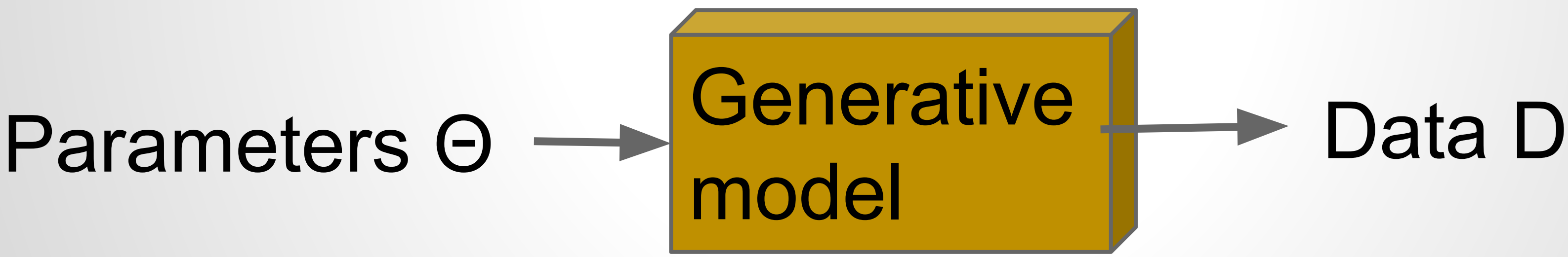
$$P(100 \text{ <>< } \mid 5\ \mathbf{o}) = \frac{P(100 \text{ <><}) \times P(5\ \mathbf{o} \mid 100 \text{ <><})}{\sum P(n \text{ <><}) \cdot P(5\ \mathbf{o} \mid n \text{ <><})}$$







$$P(100 <>< \mid 5\ \mathbf{o}) = \frac{P(100 <><) \times P(5\ \mathbf{o} \mid 100 <><)}{\sum P(n <><) \cdot P(5\ \mathbf{o} \mid n <><)}$$

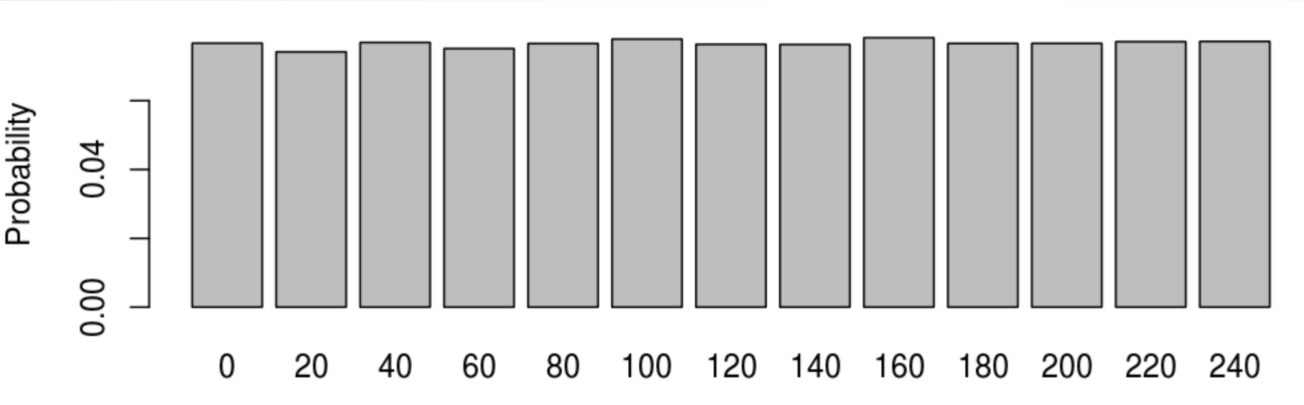


$$P(\Theta \mid D) = \frac{P(\Theta) \cdot P(D \mid \Theta)}{\sum P(\Theta) \cdot P(D \mid \Theta)}$$

Bayes theorem

What have we done?

- We have specified prior information

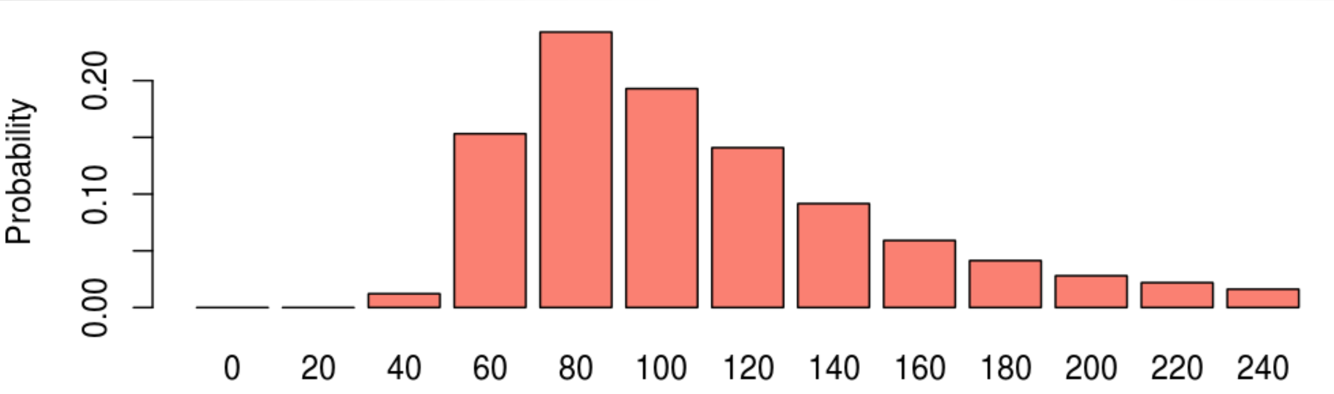


P(Θ)

- A generative model

P(D | Θ)

- And have calculated the probability of of different parameter values



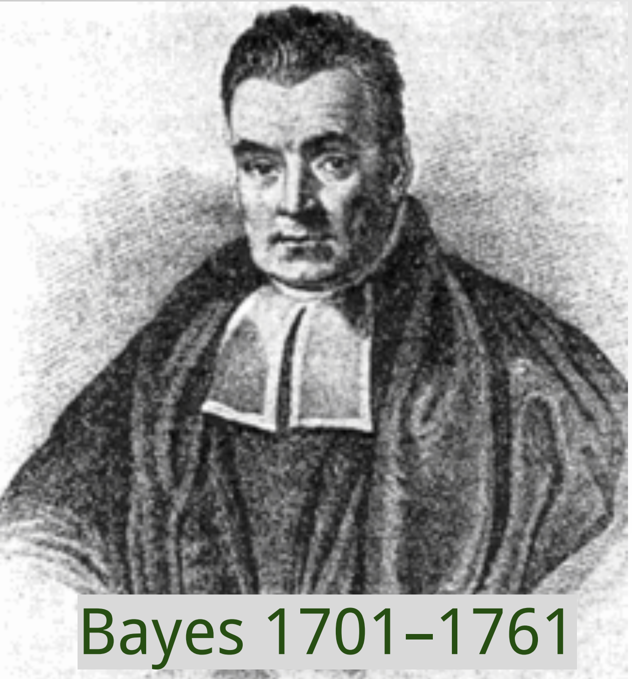
P(Θ | D)

What have we done?

- In this example we used a capture-recapture model with *one* parameter.
- But the general method works on *any* generative model and with *any* number of parameters.
- The specific computational method we used only works in rare cases...

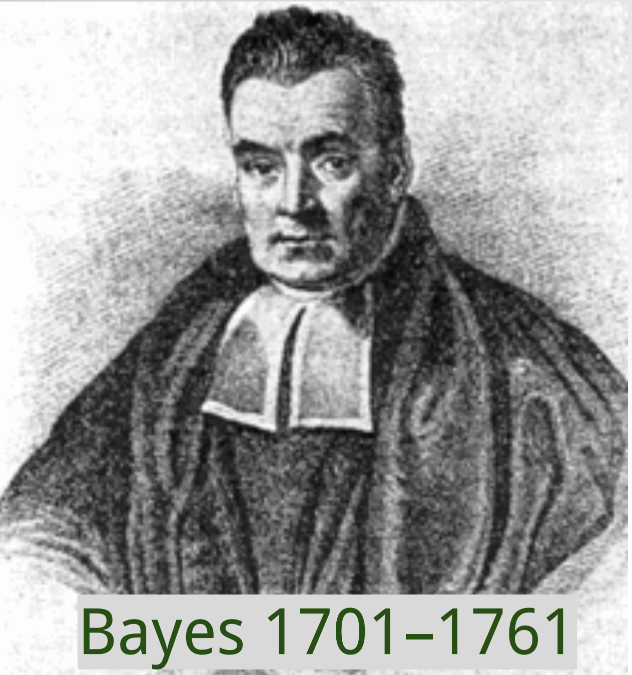
What is not Bayesian data analysis?

- A category of models
- Subjective
- Not necessarily the most computationally efficient method of fitting a model.
- Anything new.


Bayes 1701–1761


Laplace 1749–1827

Inverse Probability



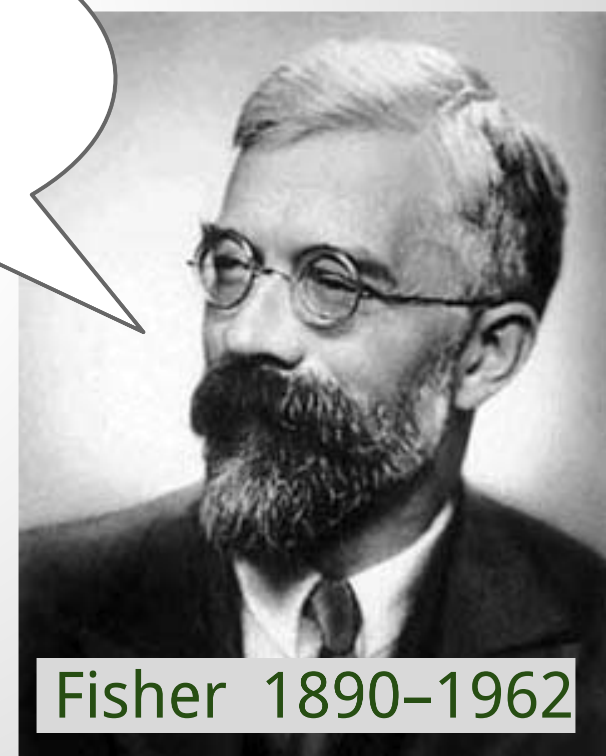
Bayes 1701–1761

“Bayesian data analysis” is not the best of names…

“Probabilistic modeling” would be better!

Bayesians!

Laplace 1749–1827



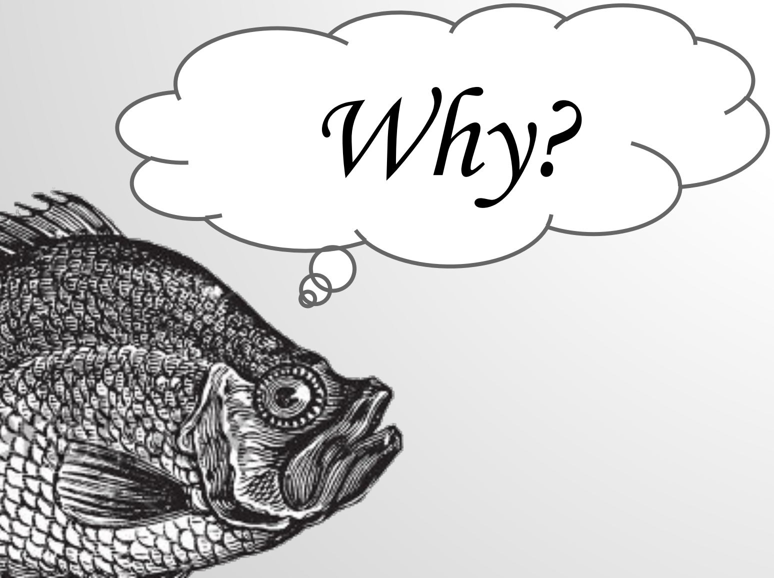
Fisher 1890–1962

UseR! 2015 prediction competition

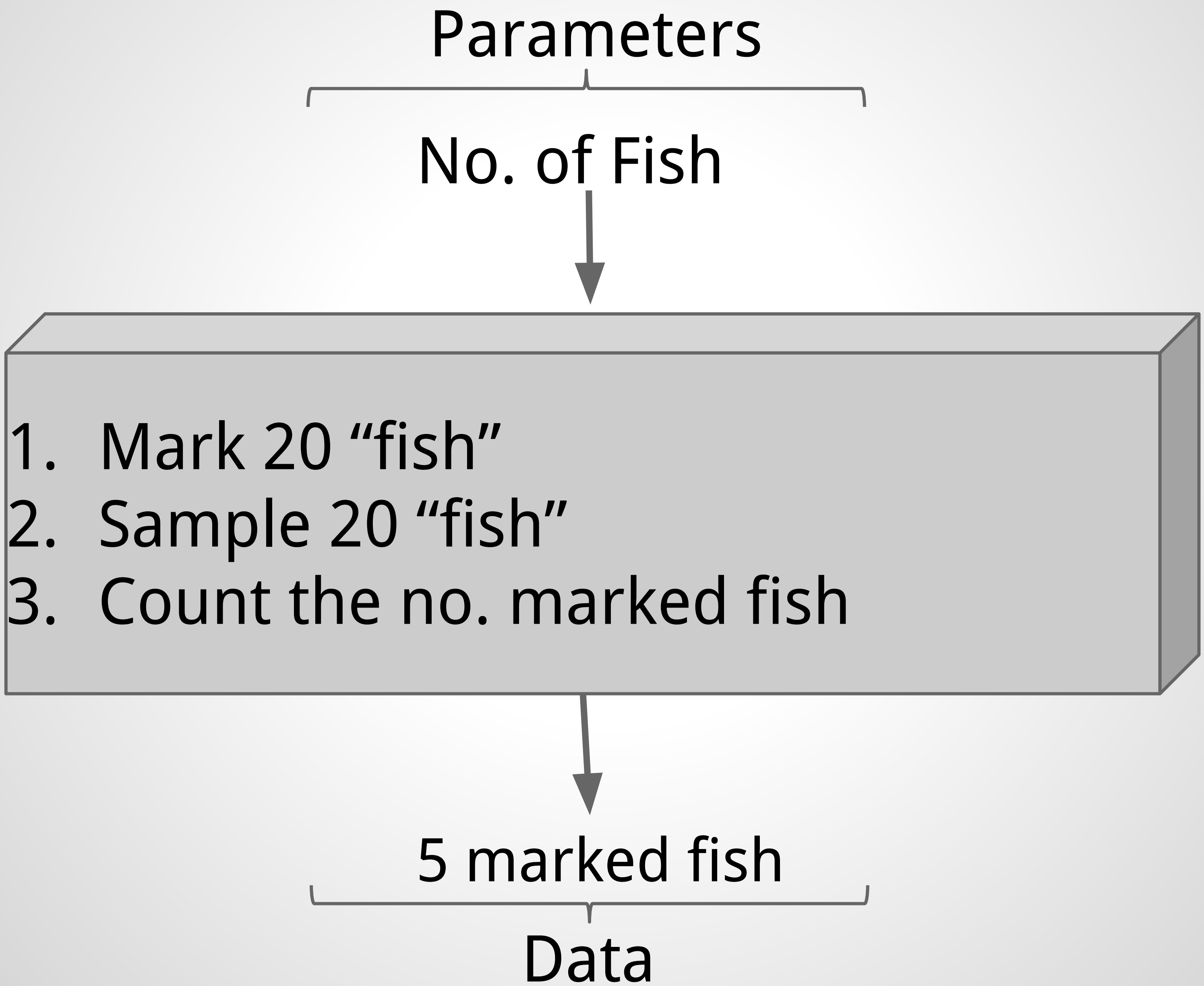
http://bit.ly/1LuF64m
20 minutes

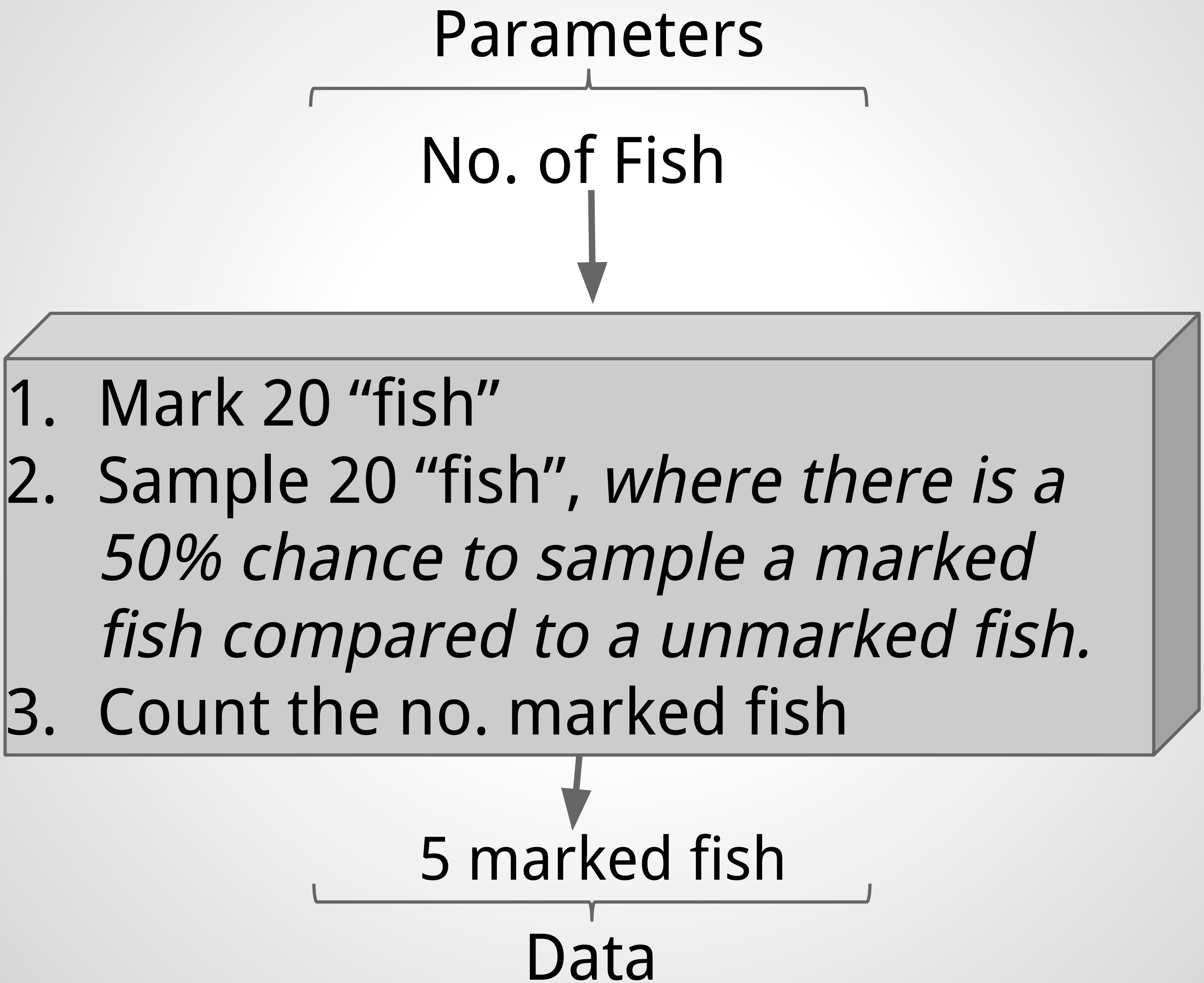
Why use Bayesian data analysis?

- You have great flexibility when building models, and can focus on that, rather than computational issues.



“Marked fish get shy! It is half as likely to catch a marked fish compared to a fish that has not been marked.”


Parameters
No. of Fish
1. Mark 20 “fish”
2. Sample 20 “fish”
3. Count the no. marked fish
5 marked fish
Data


Parameters
No. of Fish
1. Mark 20 “fish”
2. Sample 20 “fish”, *where there is a 50% chance to sample a marked fish compared to a unmarked fish.*
3. Count the no. marked fish
5 marked fish
Data

Demo

The script that was "live coded" can be found here:
http://rpubs.com/rasmusab/live_coding_user_2015_bayes_tutorial

Why use Bayesian data analysis?

- You have great flexibility when building models, and can focus on that, rather than computational issues.
- You can include information sources in addition to the data, for example, expert opinion.

“There has always been plenty of fish in the lake.
Around 200, I would say!”

Demo

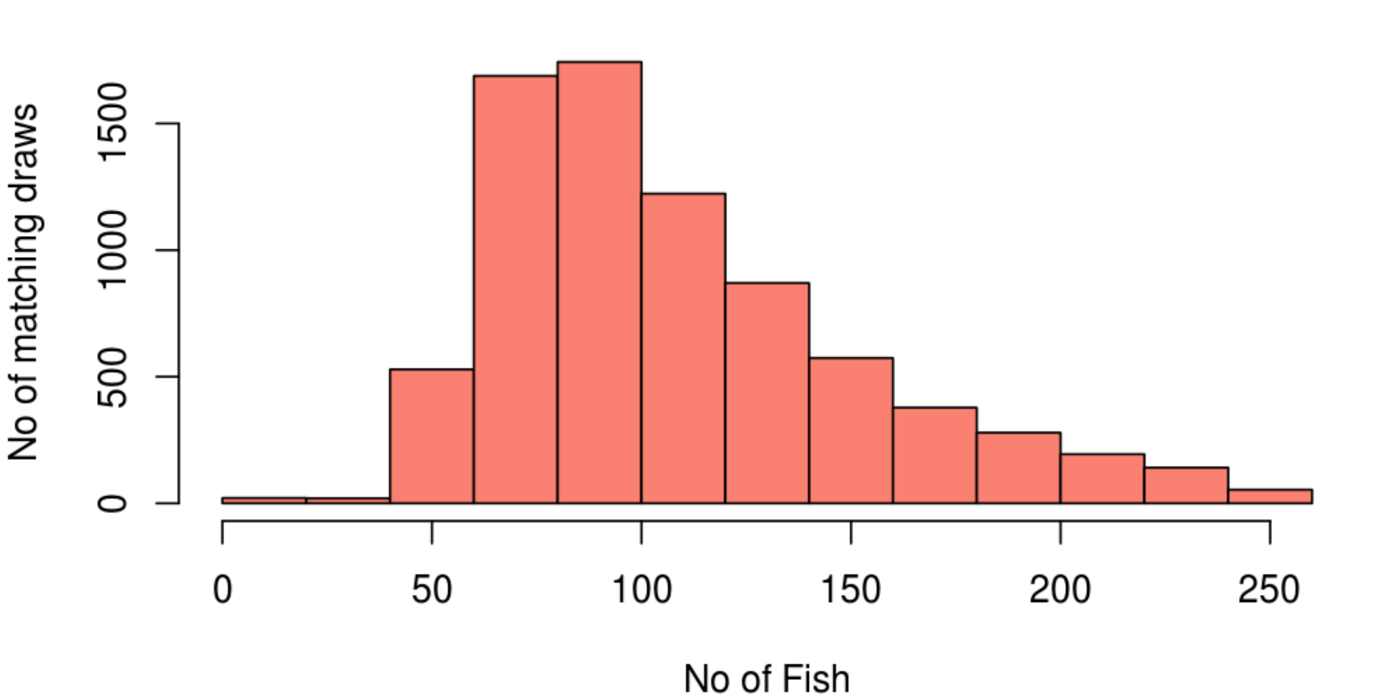
The script that was “live coded” can be found here:
http://rpubs.com/rasmusab/live_coding_user_2015_bayes_tutorial

“If you’re not using a informative prior, you’re leaving money on the table.”

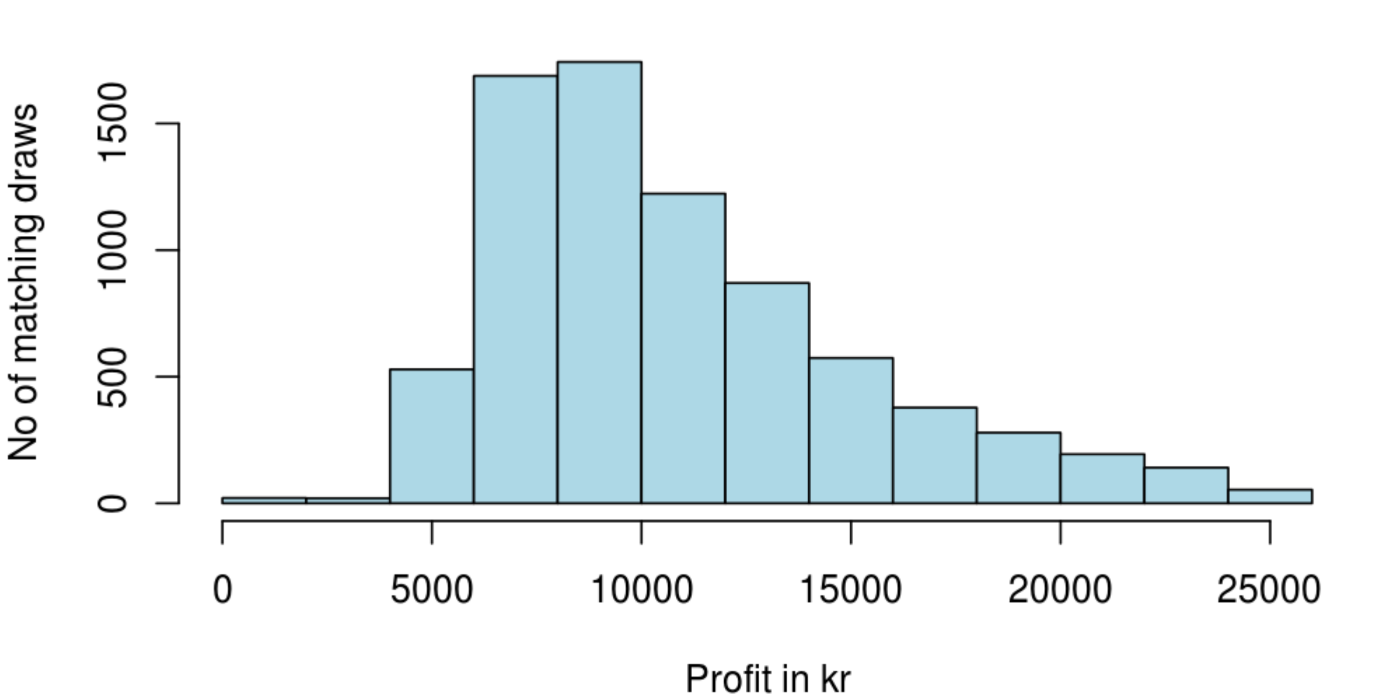
- Robert Weiss, UCLA, Los Angeles.

Why use Bayesian data analysis?

- You have great flexibility when building models, and can focus on that, rather than computational issues.
- You can include information sources in addition to the data, for example, expert opinion.
- The result of a Bayesian analysis retains the uncertainty of the estimated parameters, which is *very* useful in decision analysis.



draw_id	no_fish
1	90
2	162
3	202
4	98
5	176
...	...



draw_id	no_fish	no_fish * 100
1	90	9000
2	162	16200
3	202	20200
4	98	9800
5	176	17600
...	...	...

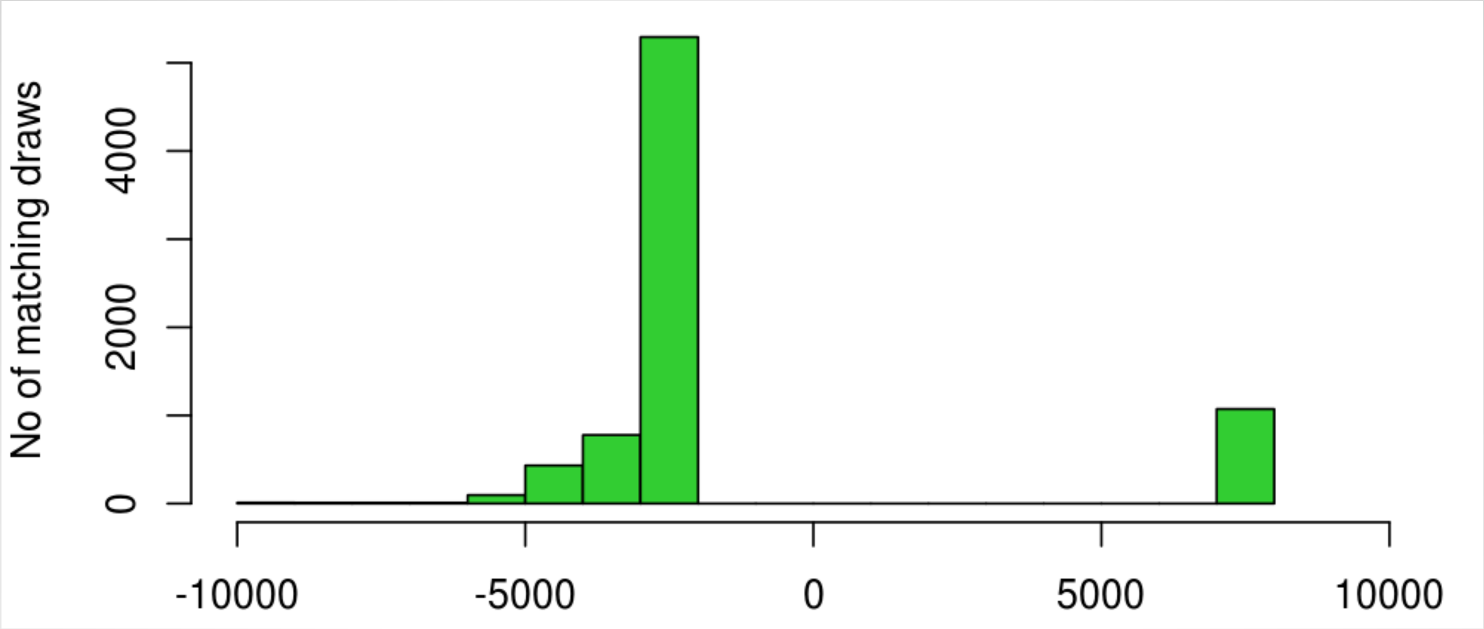
“If there are less than 50 fish in the lake, they wont last the season. It will cost 10 000 kr to plant new fish into the lake!

draw_id	no_fish	catch 80 fish x 100 kr
1	90	8000
2	162	8000
3	202	8000
4	98	8000
5	71	7100
...	...	...

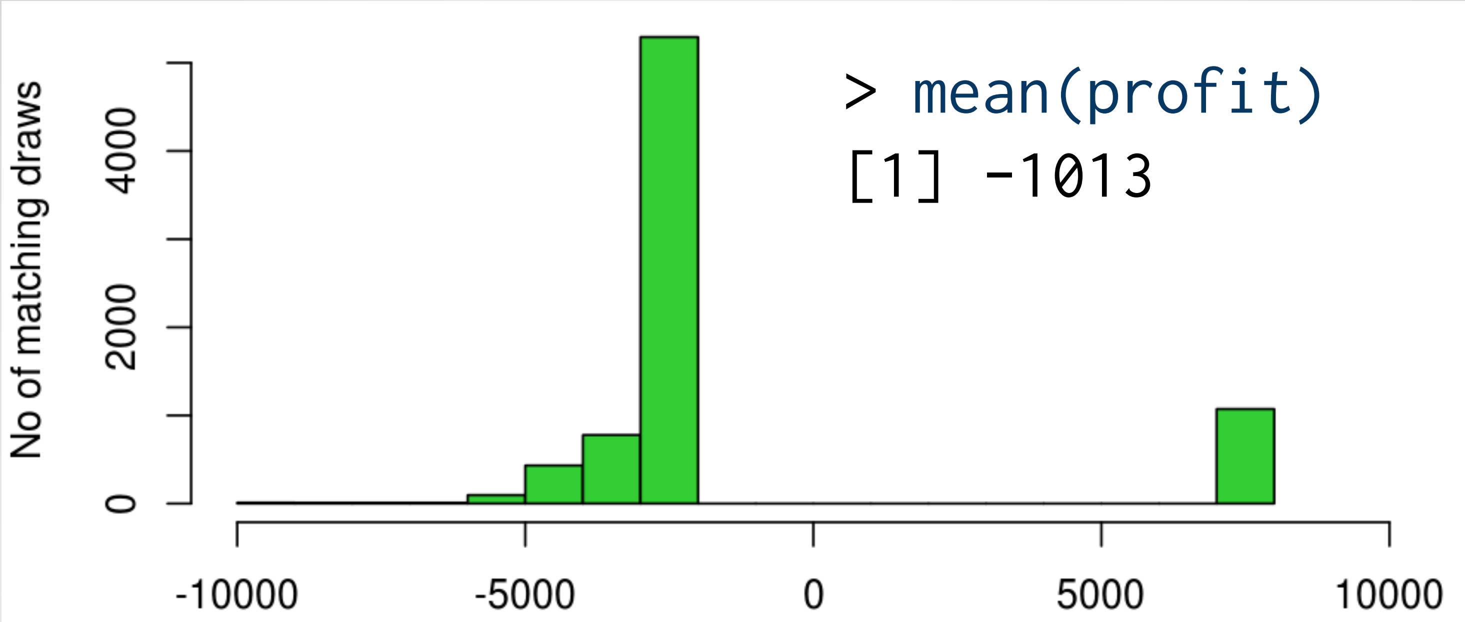
draw_id	no_fish	catch 80 fish x 100 kr	fish left	repopulation cost	Profit
1	90	8000	10	10,000	-2000
2	162	8000	82	0	8000
3	202	8000	122	0	8000
4	98	8000	18	10,000	-2000
5	71	7100	0	10,000	-2900
...	...	...	...	...	...

```
profit <- min(no_fish, 80) * 100 -
            (no_fish - 80 < 50) * 10000
```

draw_id	no_fish	catch 80 fish x 100 kr	fish left	repopulation cost	Profit
1	90	8000	10	10,000	-2000
2	162	8000	82	0	8000
3	202	8000	122	0	8000
4	98	8000	18	10,000	-2000
5	71	7100	0	10,000	-2900
...	...	...	...	...	...

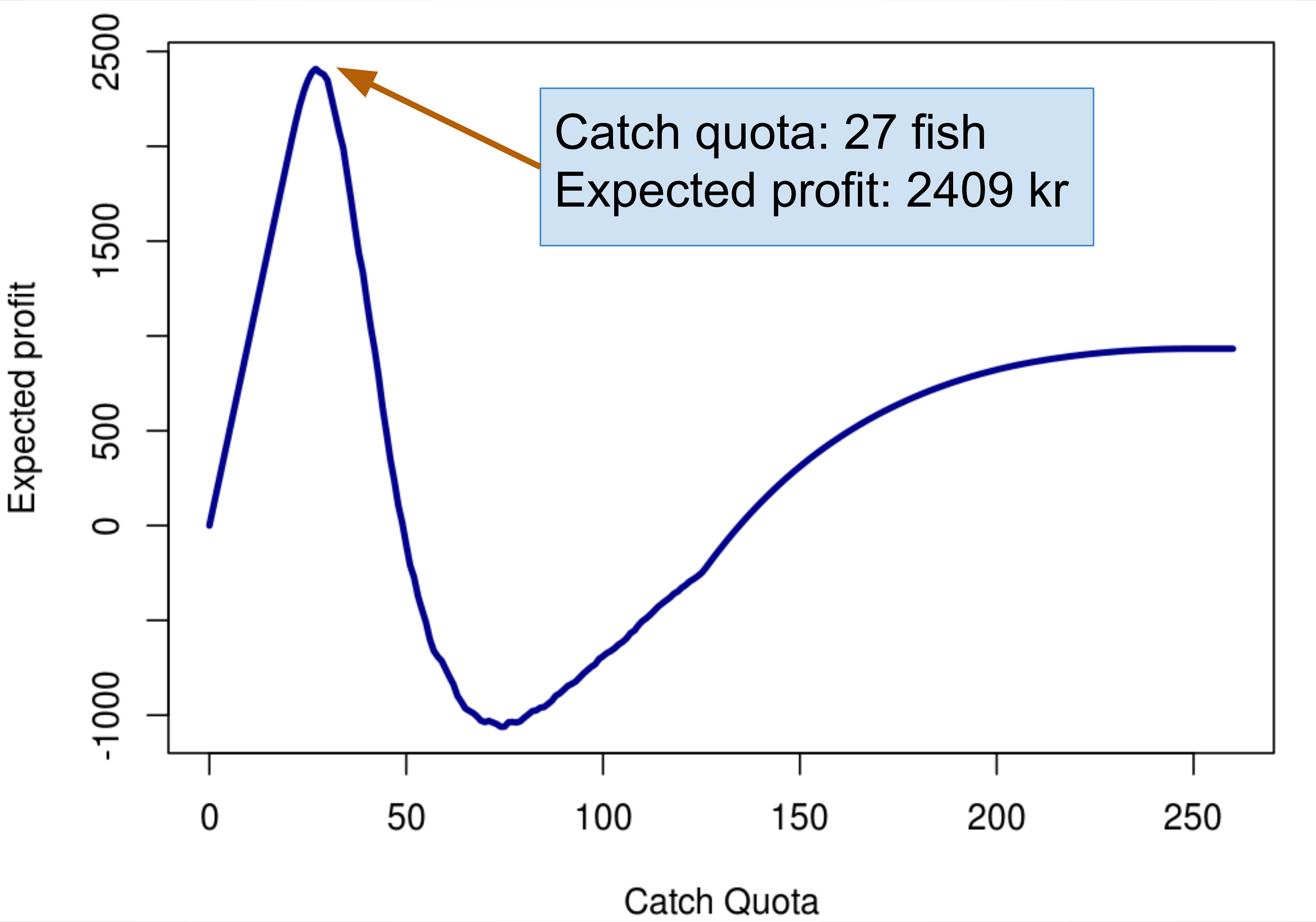


draw_id	no_fish	catch 80 fish x 100 kr	fish left	repopulation cost	Profit
1	90	8000	10	10,000	-2000
2	162	8000	82	0	8000
3	202	8000	122	0	8000
4	98	8000	18	10,000	-2000
5	71	7100	0	10,000	-2900
...	...	...	...	...	...



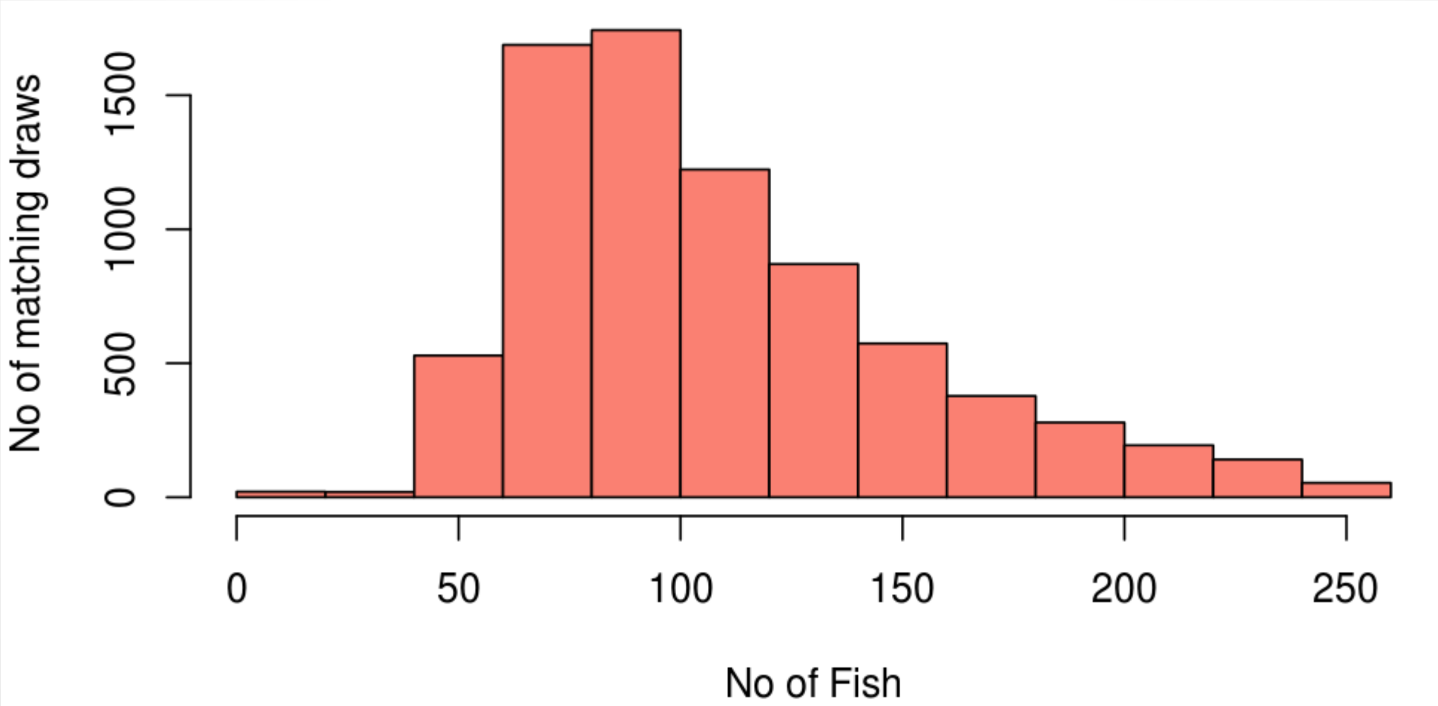
draw_id	no_fish	catch 80 fish x 100 kr	fish left	repopulation cost	Profit
1	90	8000	10	10,000	-2000
2	162	8000	82	0	8000
3	202	8000	122	0	8000
4	98	8000	18	10,000	-2000
5	71	7100	0	10,000	-2900
...	...	...	...	...	...

What's the optimal catch quota?



80

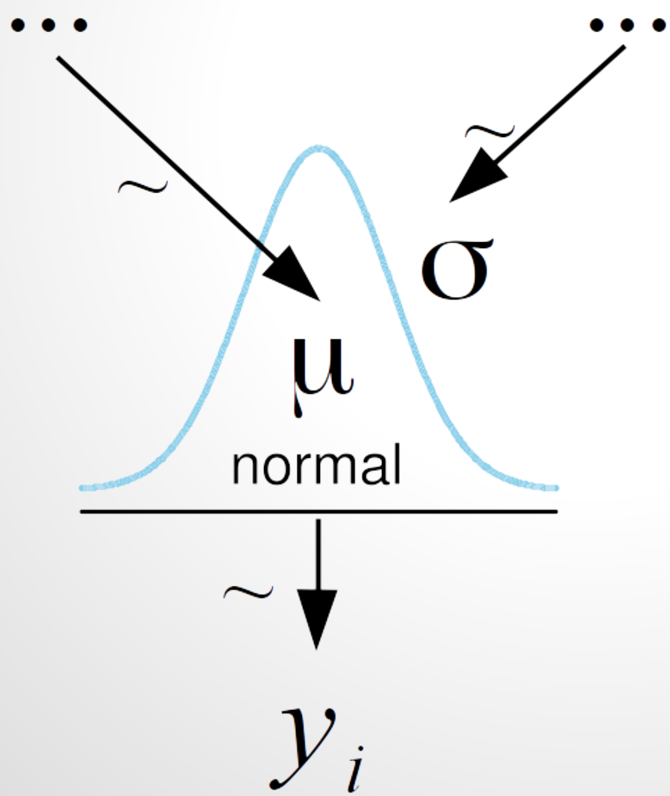
vs



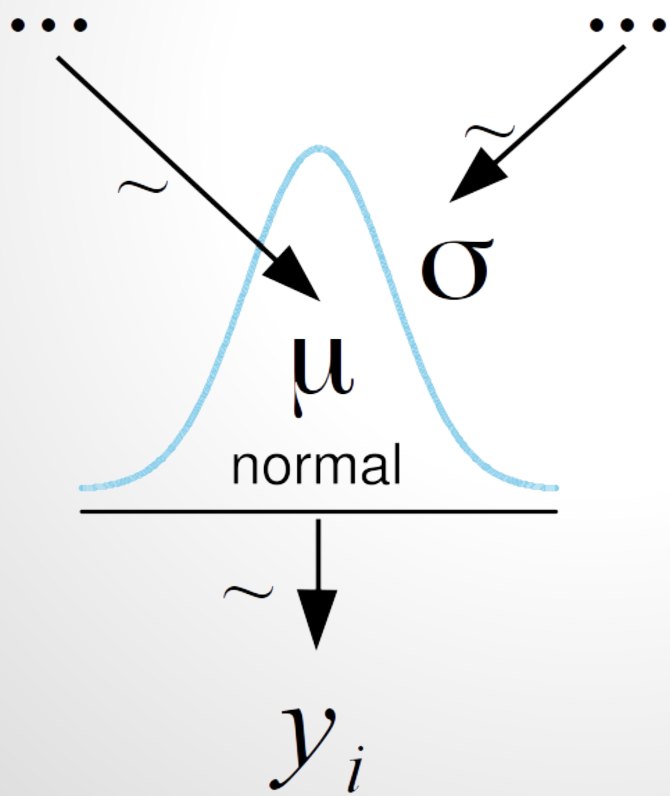
No of matching draws
0
500
1000
1500
0
50
100
150
200
250
No of Fish

Why use Bayesian data analysis?

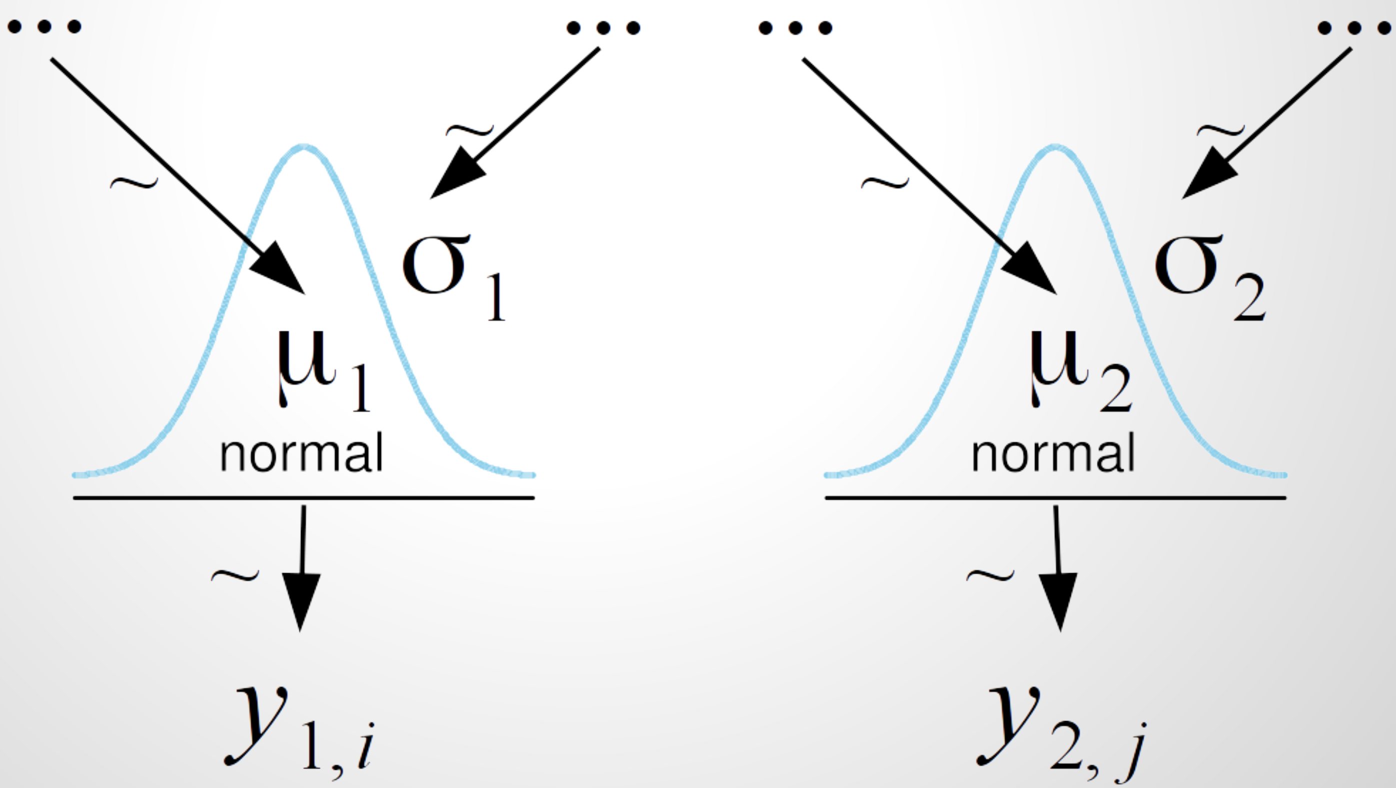
- You have great flexibility when building models, and can focus on that, rather than computational issues.
- You can include information sources in addition to the data, for example, expert opinion.
- The result of a Bayesian analysis retains the uncertainty of the estimated parameters, which is *very* useful in decision analysis.
- You probably are already...

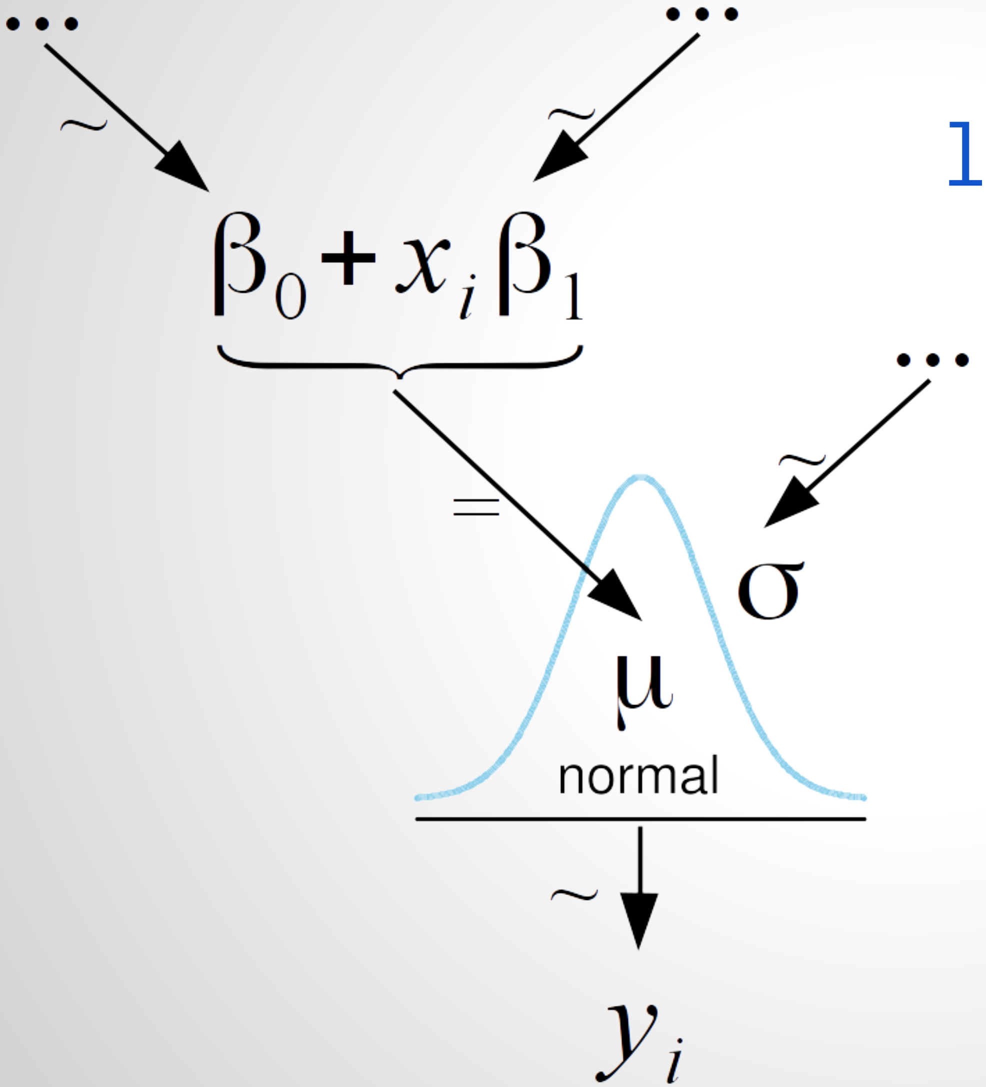

...
...
~
~
σ
μ
normal
~
y_i

`t.test(y)`



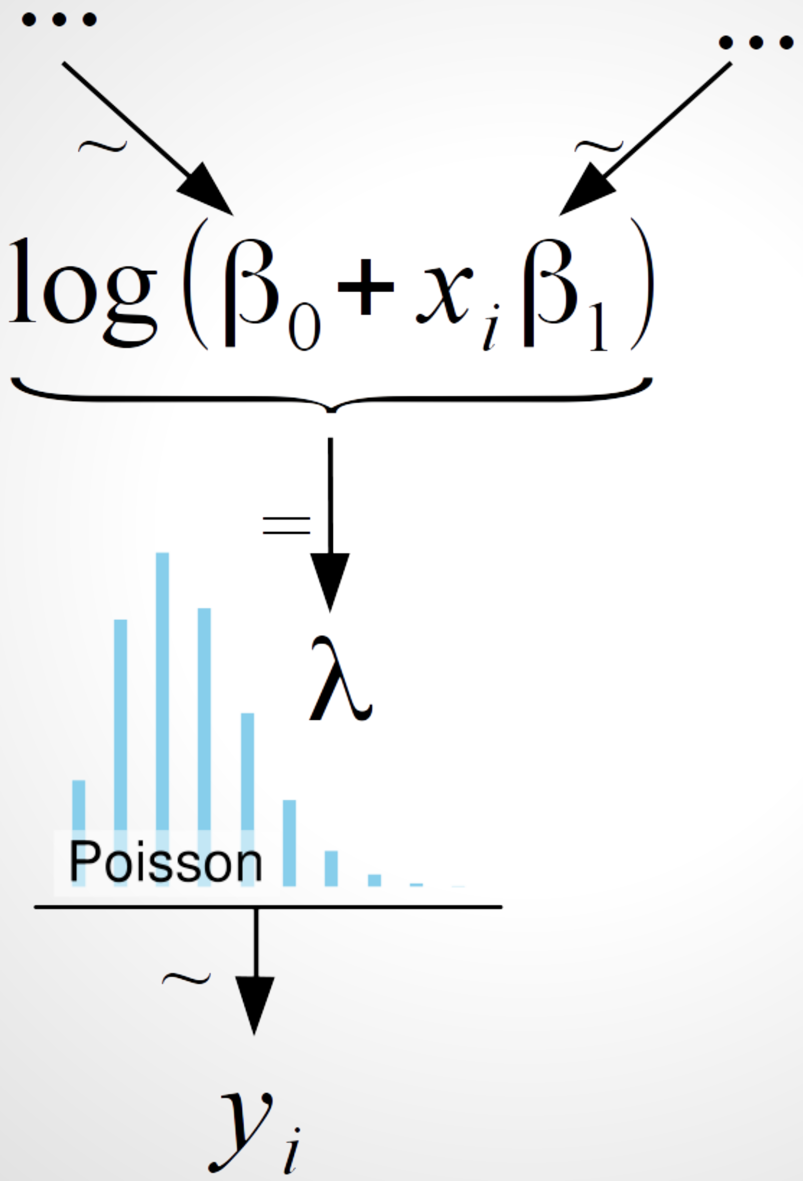
```
t.test(y1, y2)
```






```
lm(y ~ 1 + x)
```

glm(y ~ 1 + x, family = “poisson”)



Why not use Bayesian data analysis?

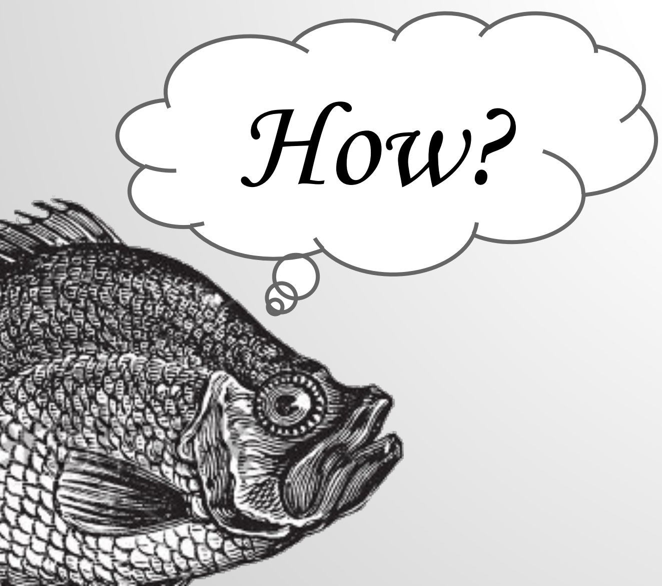
- Everything is just working fine as it is.
- I’m not that interested in uncertainty.
- It’s too computationally demanding.

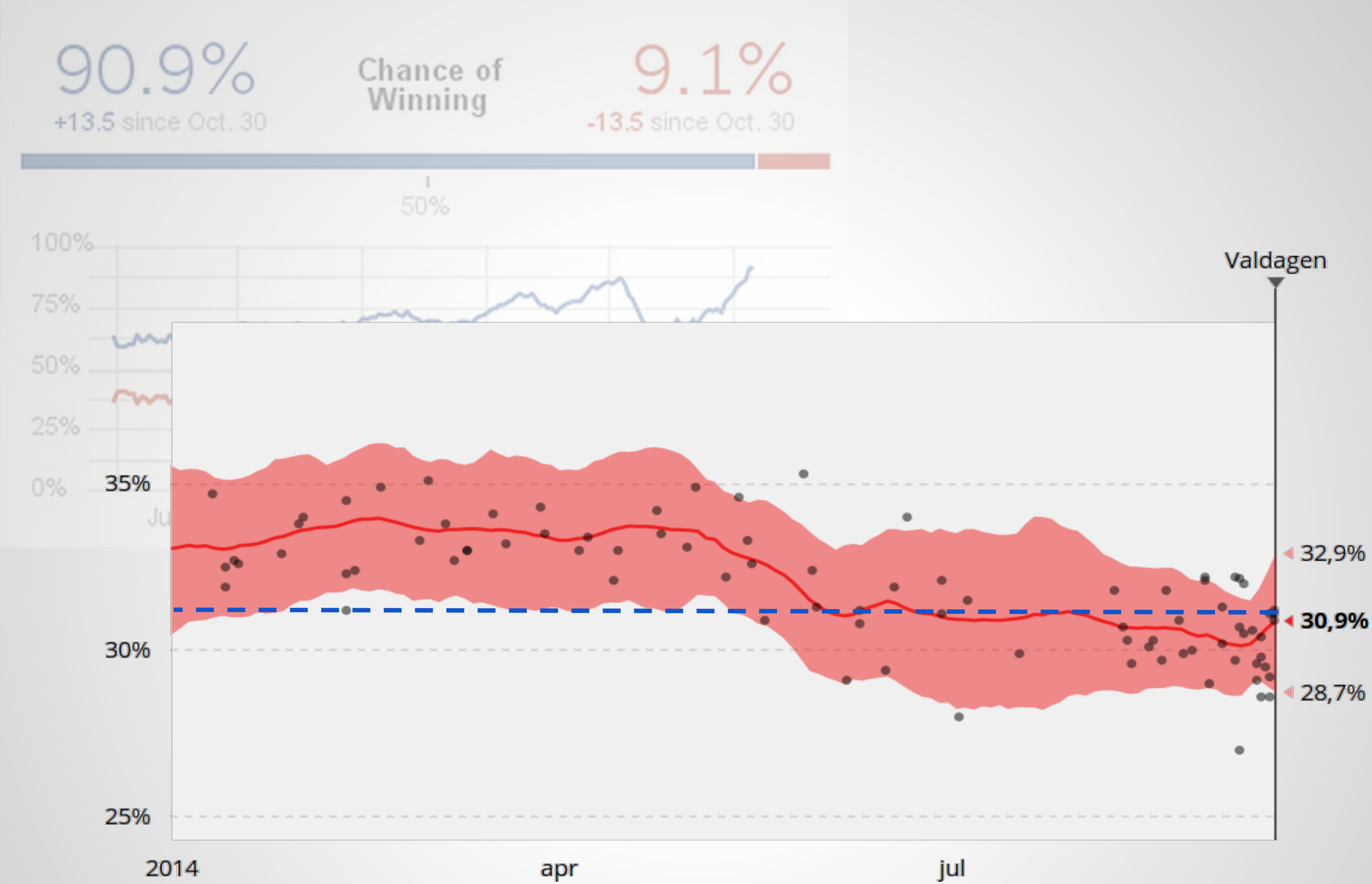
Exercise 1

Bayesian A/B testing for
Swedish Fish Incorporated
http://bit.ly/1SSCAaj

How to interpret and perform a Bayesian data analysis in R?

- Interpreting the result of an Bayesian data analysis is usually straight forward.

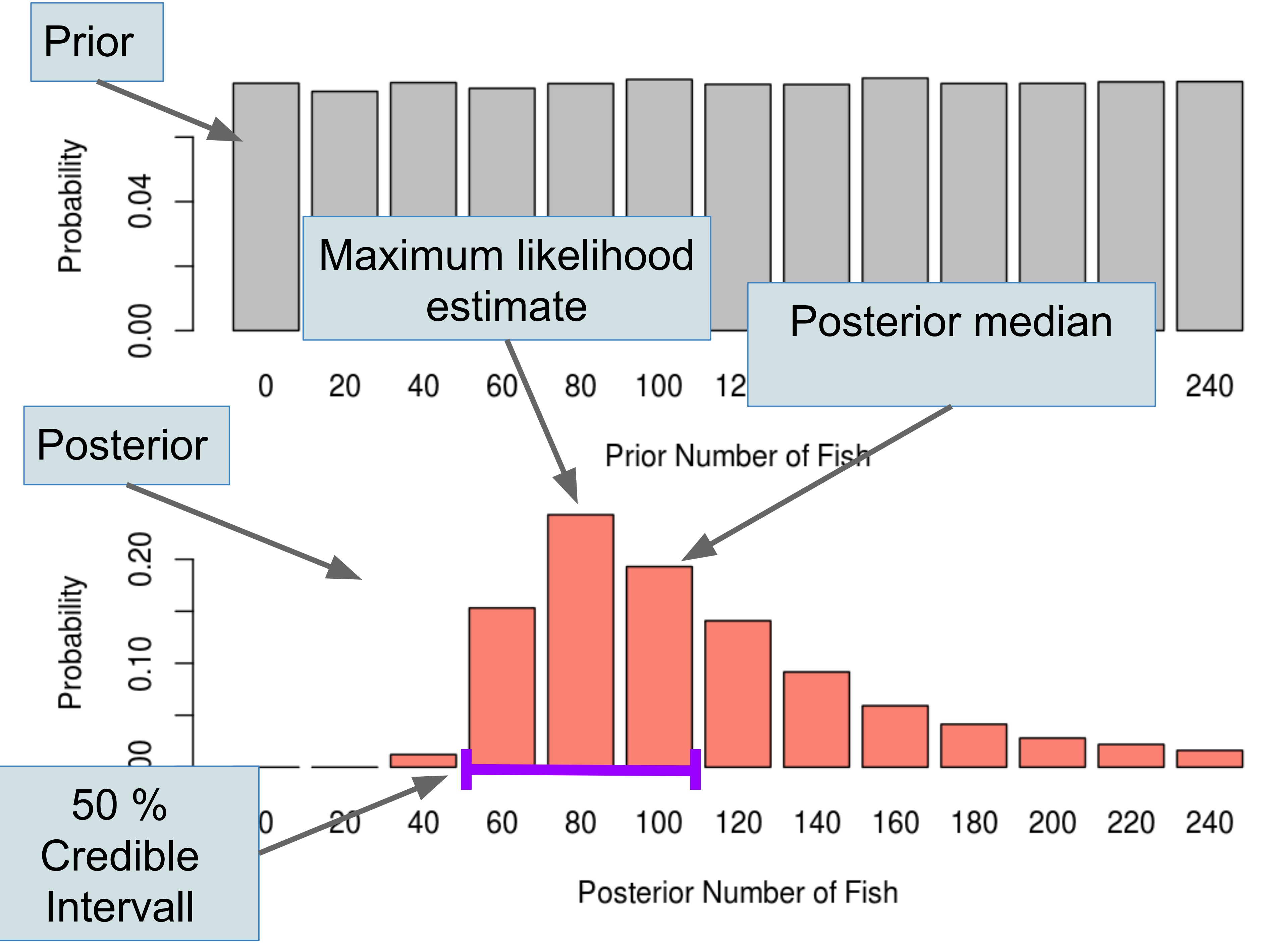




With 95% probability the support of the voters lie within this band.

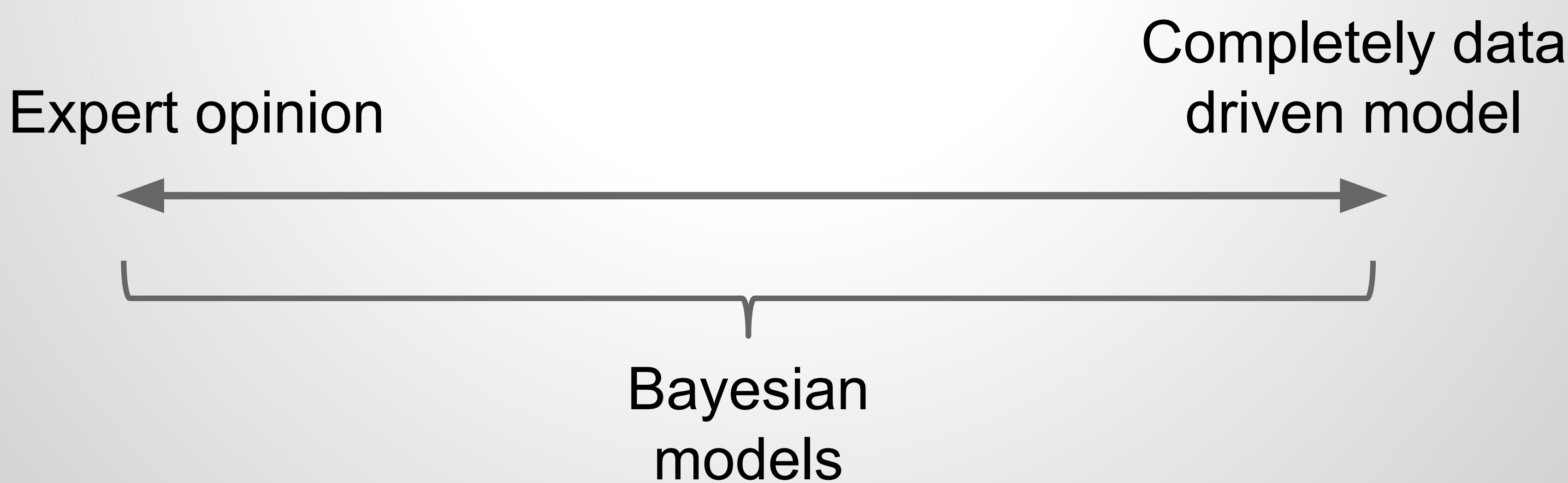
How to interpret and perform a Bayesian data analysis in R?

- Interpreting the result of an Bayesian data analysis is usually straight forward.
- But if you scratch the surface there is a lot of Bayesian jargon!


Prior
Probability
0.00
0.04
0
20
40
60
80
100
12
240
Maximum likelihood estimate
Posterior median
Prior Number of Fish
Posterior
Probability
0.10
0.20
20
40
60
80
100
120
140
160
180
200
220
240
50 % Credible Intervall
Posterior Number of Fish

More Bayesian Jargon

- Priors
  - Objective priors
  - Subjective priors
  - Informative priors
  - Improper priors
  - Conjugate priors



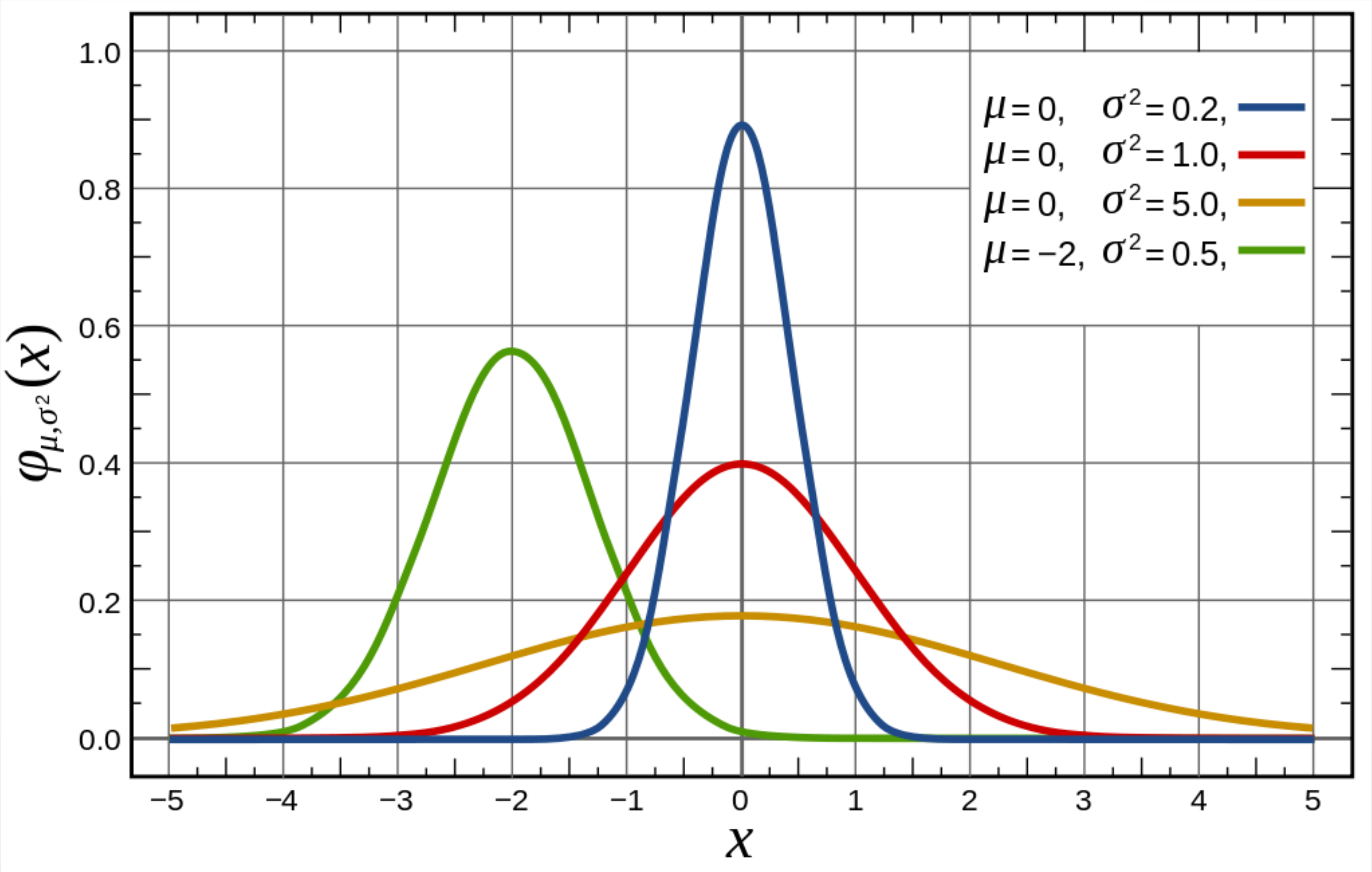
More Bayesian Jargon: Distributions!

More Bayesian Jargon: Distributions!

- The usual suspects: The Normal

x ~ Normal(μ, σ)

```
x <- rnorm(n_draw, mu, sd)
```

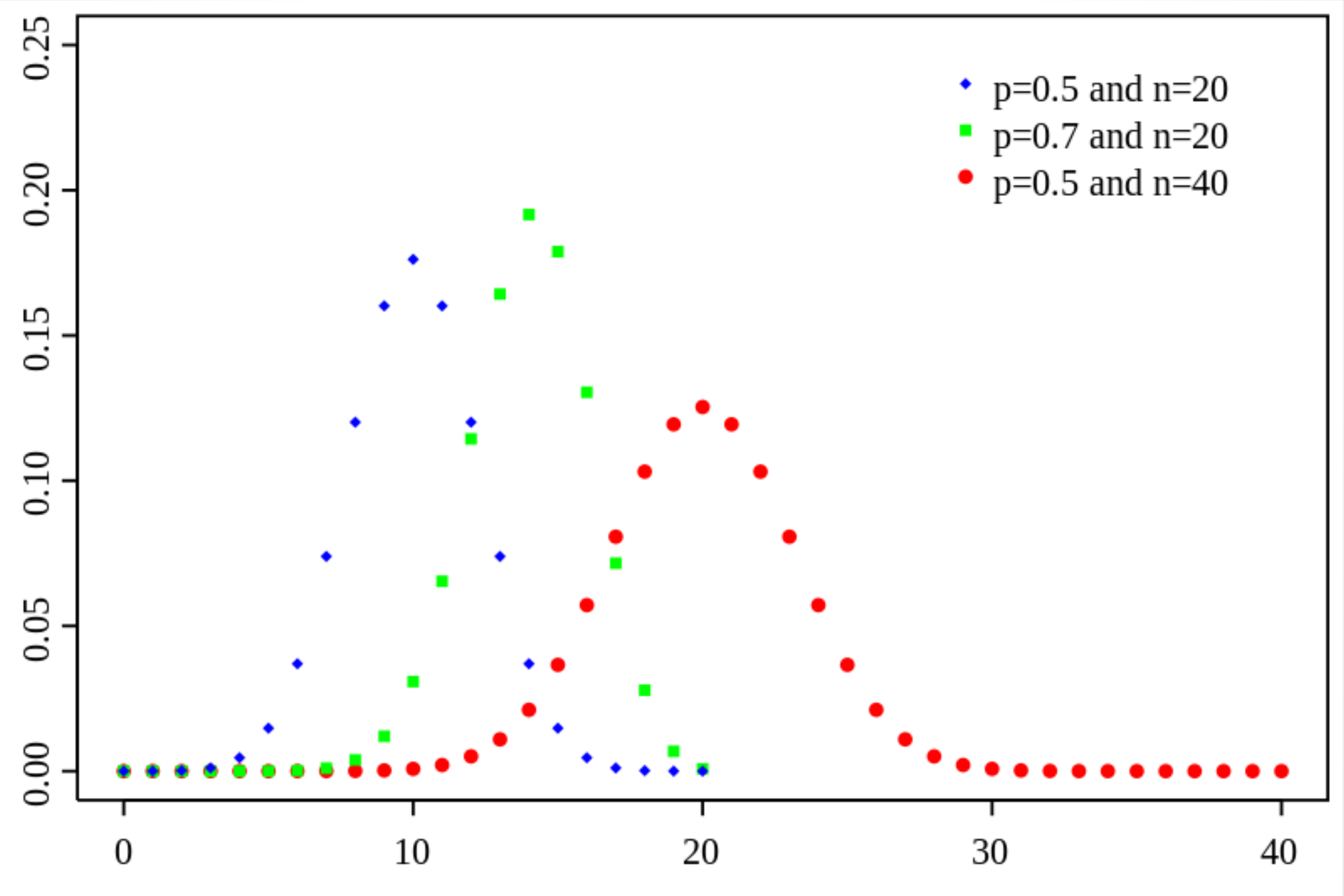


More Bayesian Jargon: Distributions!

- The usual suspects: The Binomial

x ~ Binomial(p, n)

```
x <- rbinom(n_draw, size, prob)
```

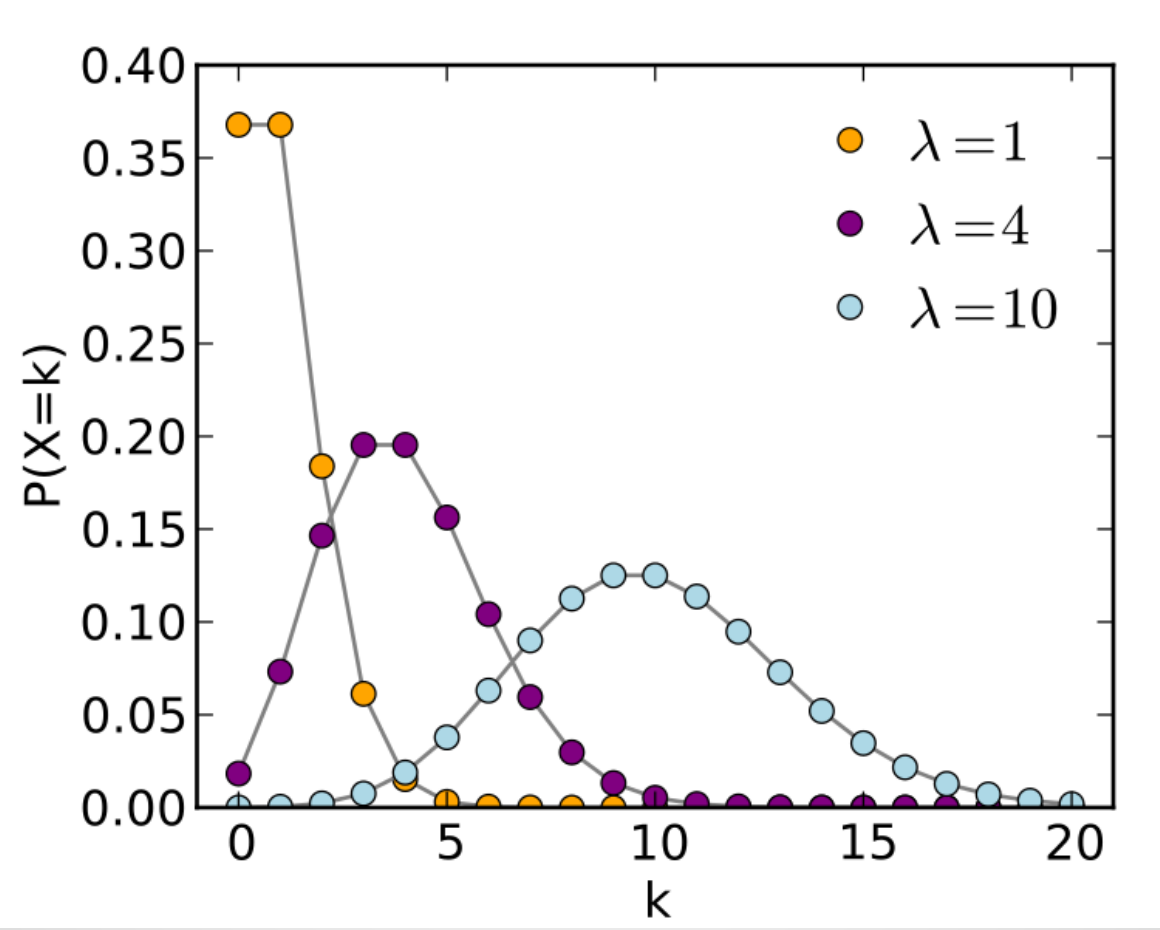


More Bayesian Jargon: Distributions!

- The usual suspects: The Poisson

$x \sim \text{Poisson}(\lambda)$

```
x <- rpois(n_draw, lambda)
```

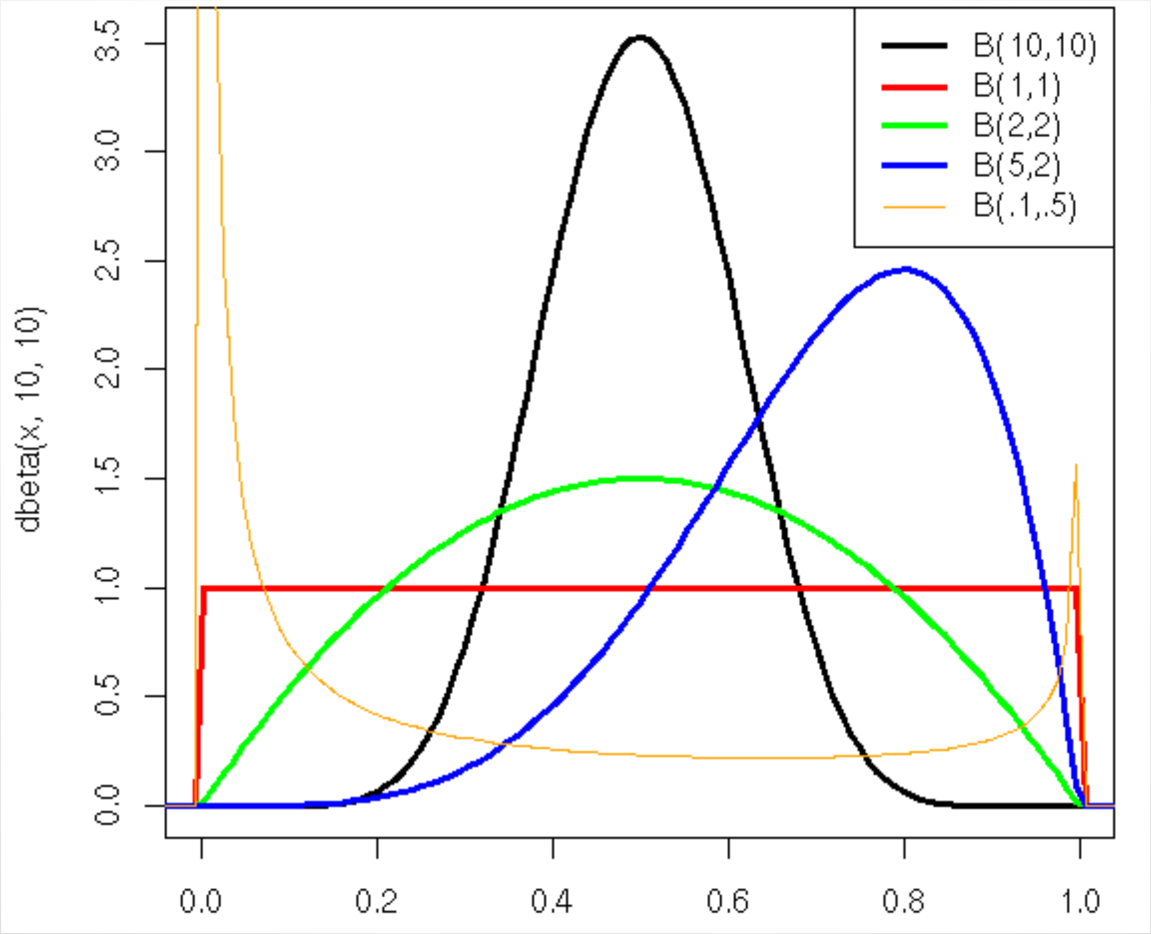


More Bayesian Jargon: Distributions!

- Less common beasts: The Beta

x ~ Beta(α, β)

```
x <- rbeta(n_draw, shape1, shape2)
```

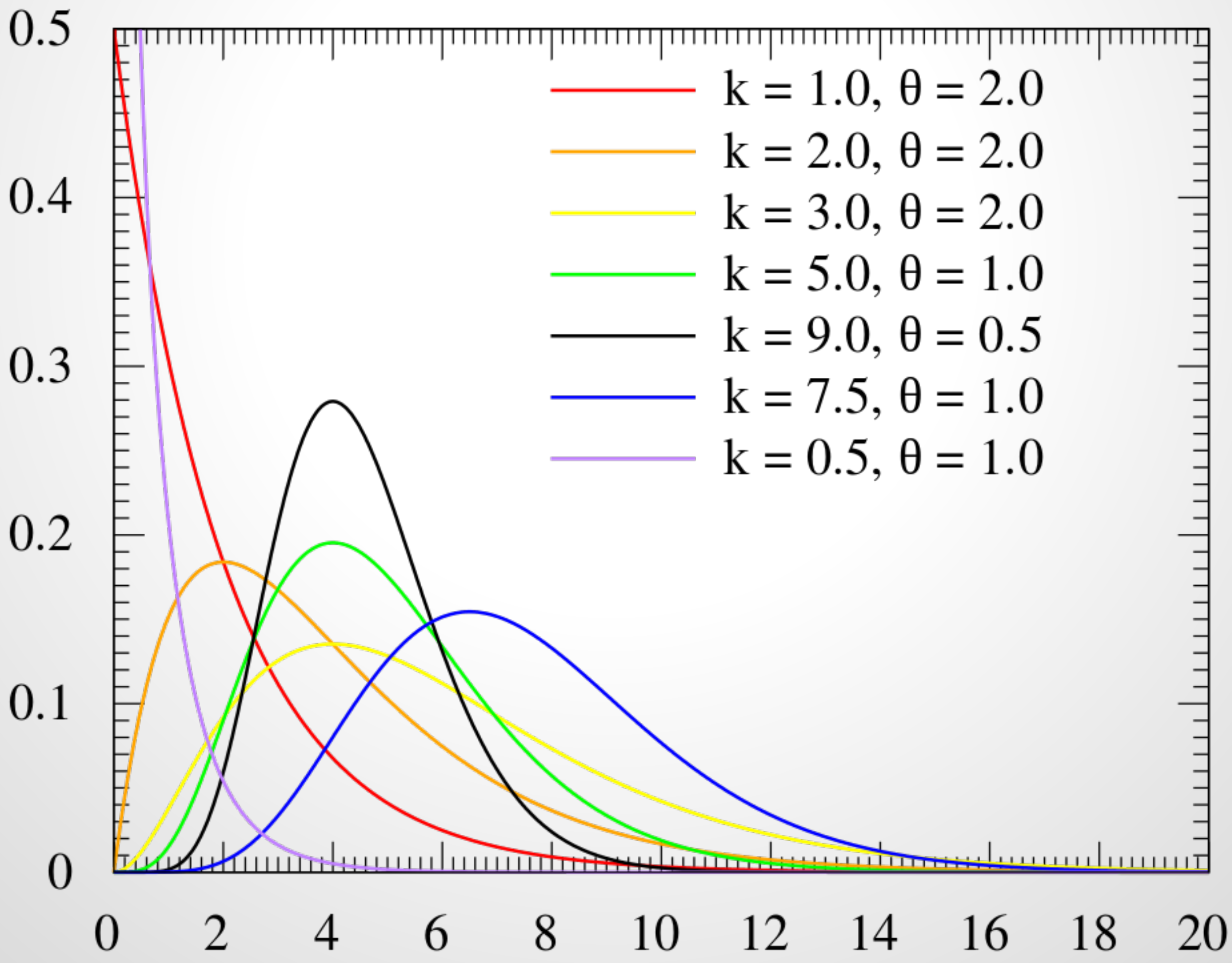


More Bayesian Jargon: Distributions!

- Less common beasts: The Gamma

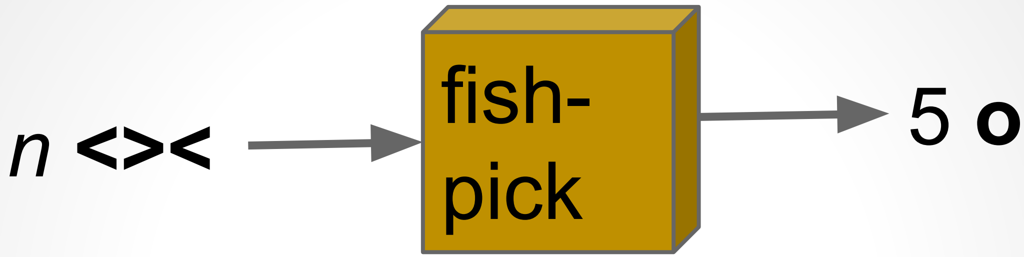
x ~ Gamma(k, θ)

```
x <- rgamma(n_draw, shape, scale)
```



More Bayesian Jargon: Distributions!

- Less common beasts: The Hypergeometric



- Fisher's noncentral hypergeometric distribution

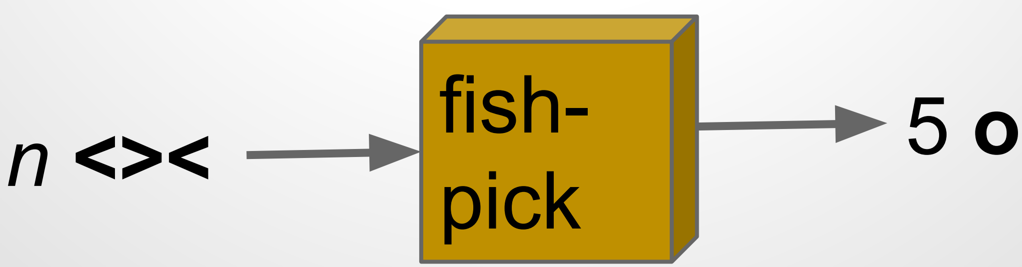
- When it comes to distributions, Wikipedia is your friend!

More Bayesian Jargon

- Samples, samples, samples.
  Prior samples:
  `[63, 30, 167, 30, 164, 222, 225, 42, 122, … ]`
  Posterior samples:
  `[167, 135, 148, 90, 162, 88, 98, 110, 176, … ]`
- Methods to generate posterior samples:
  - Approximate Bayesian Computation (ABC)
  - Markov Chain Monte Carlo (MCMC)
    - Metropolis-Hastings
    - Gibbs Sampling
    - Hamiltonian monte carlo
- Other methods
  - Conjugate models
  - Laplace Approximation
  - Etc. Etc. Etc.

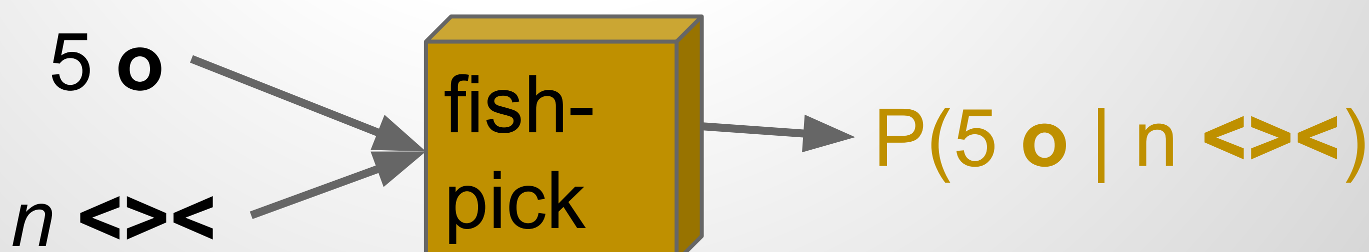
Faster Bayesian computation

- We have been doing *approximate Bayesian computation*, which is the most general and **slowest** method for fitting a Bayesian model.
- Faster methods have in common that:
  - They require that the *likelihood* that the generative model will generate any given data can be *calculated*.



Faster Bayesian computation

- We have been doing *approximate Bayesian computation*, which is the most general and **slowest** method for fitting a Bayesian model.
- Faster methods have in common that:
  - They require that the *likelihood* that the generative model will generate any given data can be *calculated*.

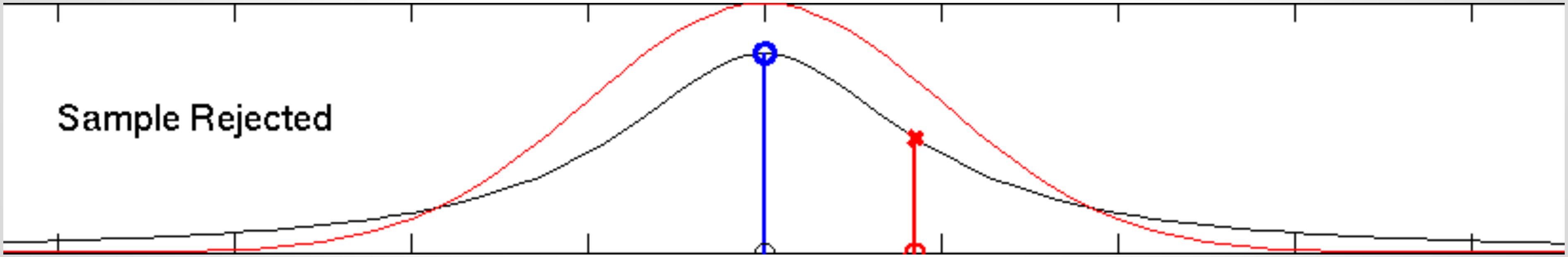


Faster Bayesian computation

- We have been doing *approximate Bayesian computation*, which is the most general and **slowest** method for fitting a Bayesian model.
- Faster methods have in common that:
  - They require that the *likelihood* that the generative model will generate any given data can be *calculated*.
  - They explore the parameter space in a smarter way.
  - What you get are samples *as if* you would have done the analysis using approximate Bayesian computation.

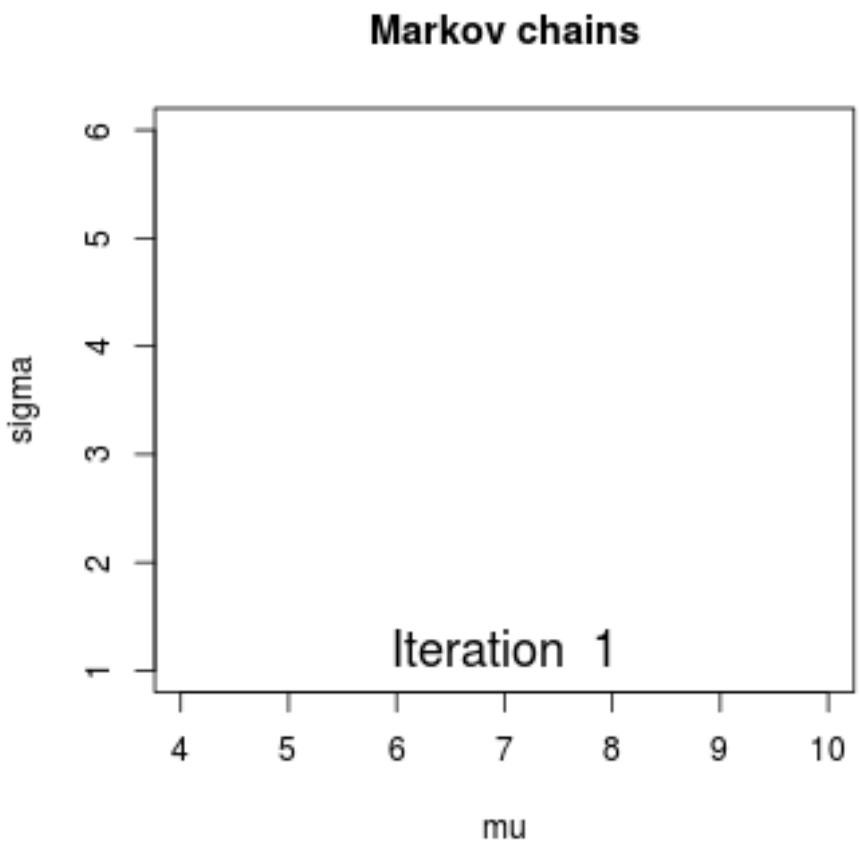
MCMC: The Metropolis-Hasting algorithm

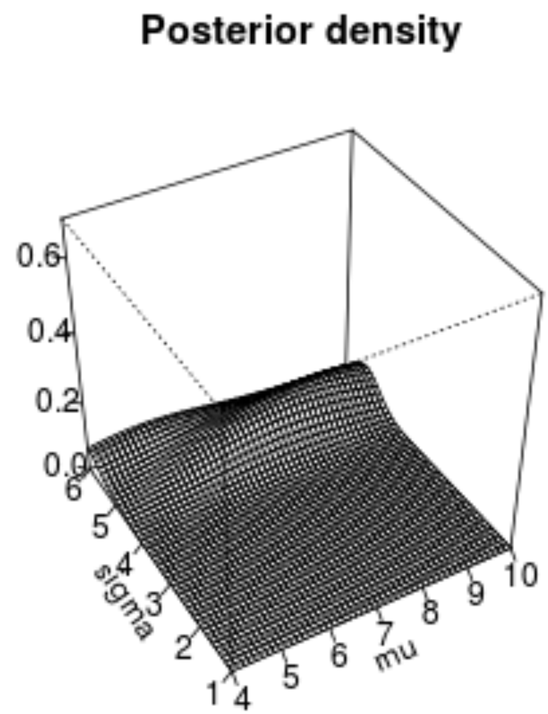
- The “classic” MCMC algorithm. Performs a random walk in the parameter space, and will stay at a parameter value proportional to its posterior probability.



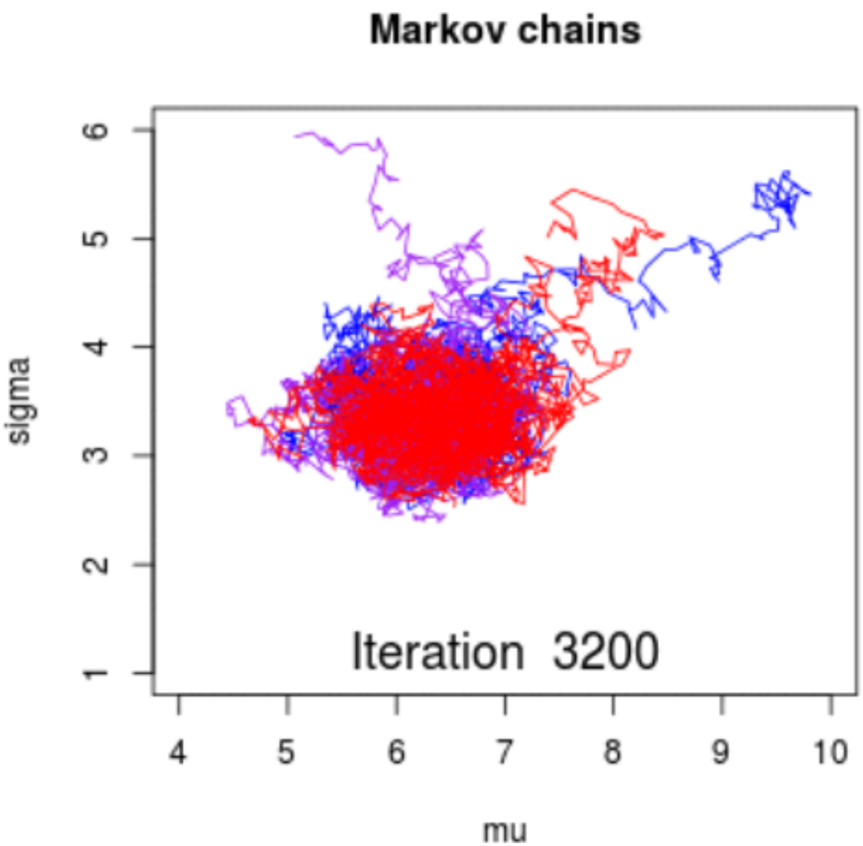
Source: https://theclevermachine.wordpress.com/tag/metropolis-hastings-sampling/

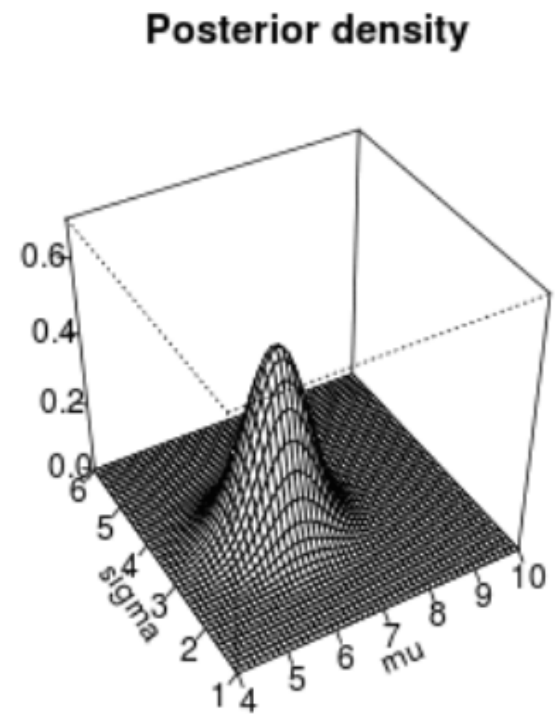
- A good R implementation can be found in the MCMCpack package as the function `MCMCmetrop1R(fun, theta.init, ...)`





Source: http://mbjoseph.github.io/blog/2013/09/08/metropolis/

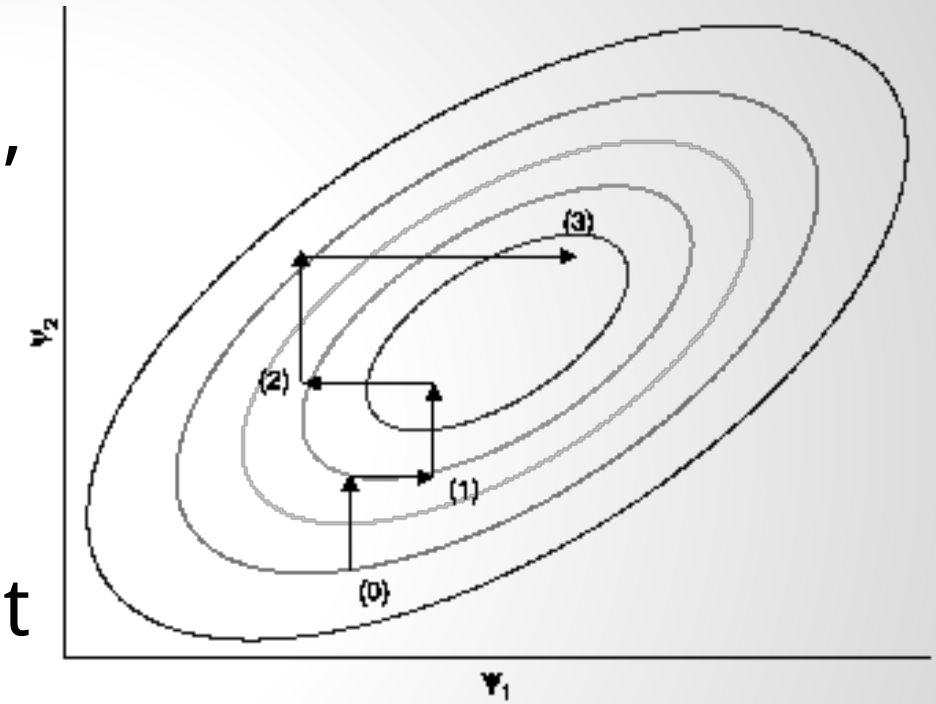




Source: http://mbjoseph.github.io/blog/2013/09/08/metropolis/

MCMC: Gibbs sampling and JAGS

- Similar to Metropolis, but moves by one parameter at a time.
- Can be much more efficient, but usually required custom built sampling schemes.
- Unless you use JAGS!



JAGS

JAGS: Just Another Gibbs Sampler

- A cross-platform implementation of the BUGS language, an R-like probabilistic programming language.
- It builds a custom Gibbs sampler for you.
- Created by Martyn Plummer, member of the R core group.
- Made to be called from R.

JAGS: Just Another Gibbs Sampler

- JAGS is tailor made for building generative models.
- A minimal JAGS program:

JAGS code

```
model {
   n <- 30
   p ~ dunif(0, 1)
   x ~ dbinom(p, n)
}
```

R code

```
n <- 30
p <- runif(1, 0, 1)
x <- rbinom(1, p, n)
```

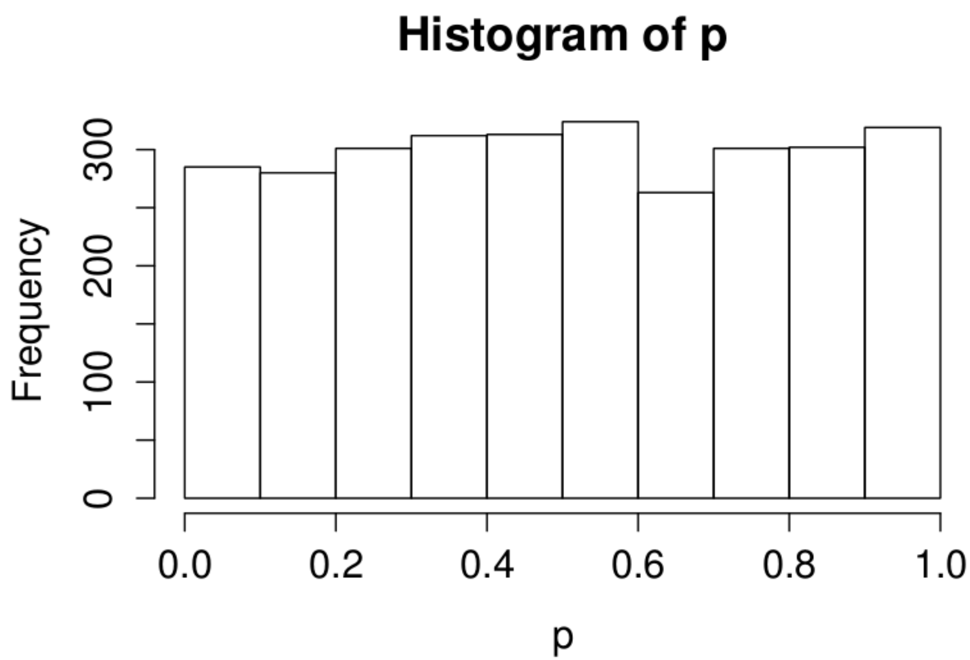
JAGS code

```
model {
   n <- 30
   p ~ dunif(0, 1)
   x ~ dbinom(p, n)
}
```

R code

```
n <- 30
p <- runif(1, 0, 1)
x <- rbinom(1, p, n)
```

```
jags.model(..., data = list())
```



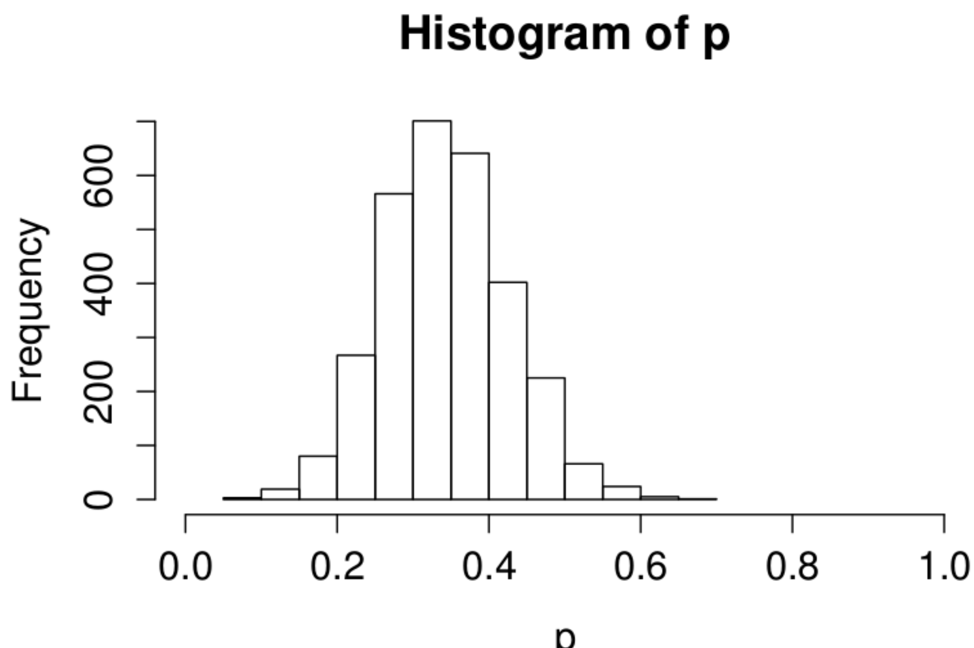
JAGS code

```
model {
   n <- 30
   p ~ dunif(0, 1)
   x ~ dbinom(p, n)
}
```

R code

```
n <- 30
p <- runif(1, 0, 1)
x <- rbinom(1, p, n)
+ ABC step
```

```
jags.model(..., data = list(x = 10))
```



JAGS: Just Another Gibbs Sampler

- JAGS is declarative…

```
model {
  n <- 30
  p ~ dunif(0, 1)
  x ~ dbinom(p, n)
}
```

=

```
model {
  x ~ dbinom(p, n)
  n <- 30
  p ~ dunif(0, 1)
}
```

JAGS: Just Another Gibbs Sampler

- JAGS is declarative…

```
model {
  n <- 30
  n <- n + 1
  p ~ dunif(0, 1)
  x ~ dbinom(p, n)
  x ~ dbinom(p, n)
}
```

JAGS: Just Another Gibbs Sampler

- JAGS is (unfortunately not) vectorized.

R code

```
> m <- 1:5
> x <- rpois(5, m)
> x
[1] 0 2 5 4 6
```

JAGS code

```
model {
  x ~ dpois(m)
}
```

JAGS: Just Another Gibbs Sampler

- JAGS is (unfortunately not) vectorized.

R code

```
> m <- 1:5
> x <- rpois(5, m)
> x
[1] 0 2 5 4 6
```

JAGS code

```
model {
  for(i in 1:length(m)) {
    x ~ dpois(m[i])
  }
}
```

JAGS: Just Another Gibbs Sampler

- JAGS is (unfortunately not) vectorized.

JAGS code

```
model {
  for(i in 1:length(m)) {
    x[i] ~ dpois(m[i])
  }
}
```

R code

```
> m <- 1:5
> x <- rpois(5, m)
> x
[1] 0 2 5 4 6
```

Demo

The script that was "live coded" can be found here:
http://rpubs.com/rasmusab/live_coding_user_2015_bayes_tutorial

Exercise 2

Bayesian computation with JAGS and farmer Jöns

http://bit.ly/1RGuK0X

Fitting Bayesian models in R

MCMCpack

Pre-specified models and a general metropolis algorithm.

JAGS

Gibbs sampling

Fitting Bayesian models in R

Hamiltonian
Monte Carlo

R-inla

Integrated nested Laplace
approximation

Extra everything!

Some things we have not covered

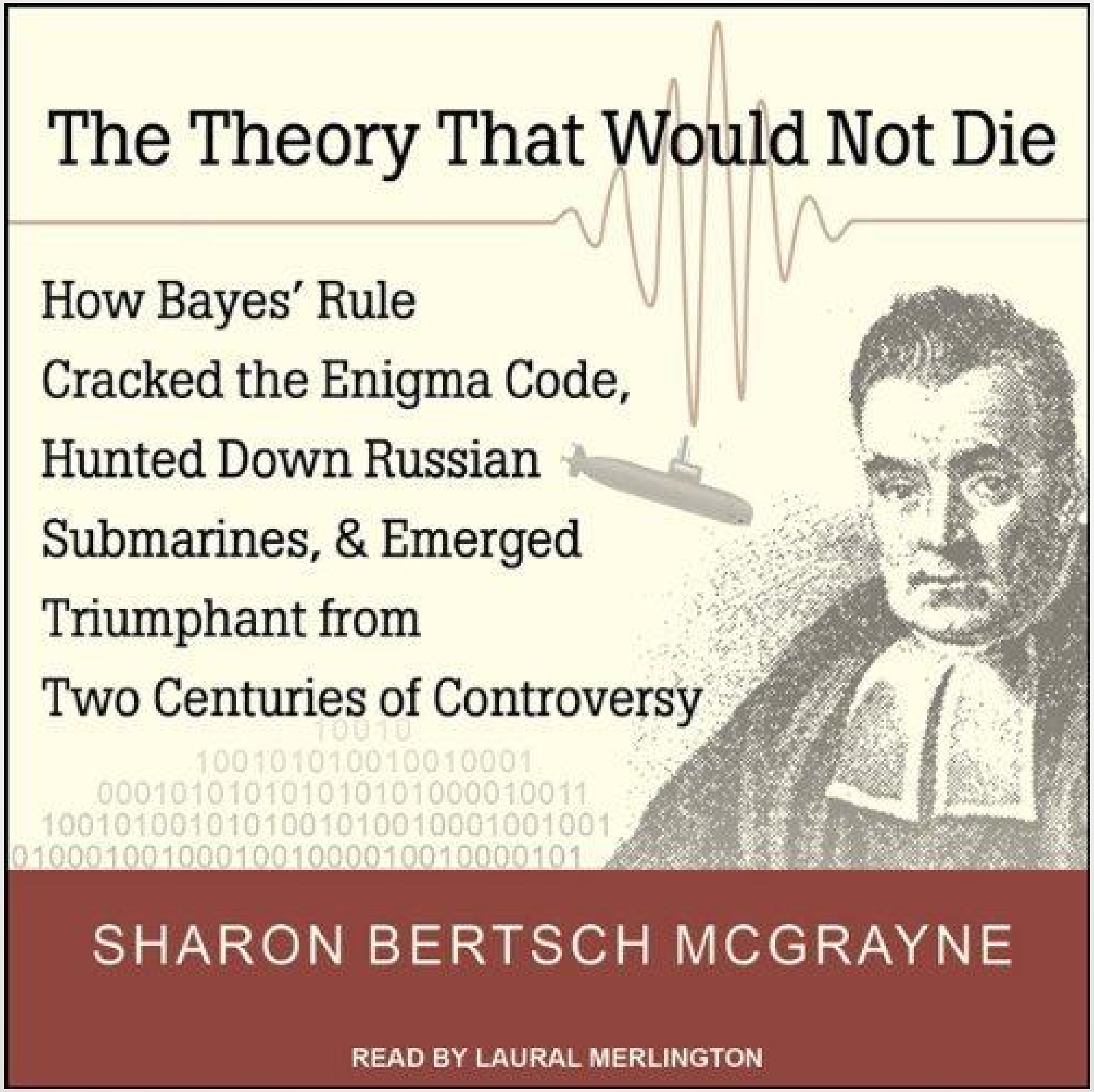
- Priors
- Distributions
- Decision analysis and “post-processing” of posteriors.
- Model selection and Bayes factors
- Philosophy
- Math

To summarize Bayesian data analysis

- What?
  - Bayesian data analysis is a flexible method to fit any type of statistical model.
  - Maximum likelihood is a special case of Bayesian model fitting.
- Why?
  - Makes it possible to define highly custom models.
  - Makes it possible to include information from many sources, for example, data and expert knowledge.
  - Quantifies and retains the **uncertainty** in parameter estimates and predictions.
- How?
  - R! Using ABC, MCMCpack, JAGS, STAN, R-inla, etc.


Texts in Statistical Science
Bayesian Data Analysis
Third Edition
Texts in Statistical Science
The BUGS Book
A Practical Introduction to Bayesian Analysis
Machine Learning
A Probabilistic Perspective
Kevin P. Murphy
Second Edition
Doing Bayesian Data Analysis
A Tutorial with R, JAGS, and Stan
=
×
/
$p(\theta|D)$
$p(D|\theta)$
$p(\theta)$
$p(D)$
John K. Kruschke

Summer reading / listening

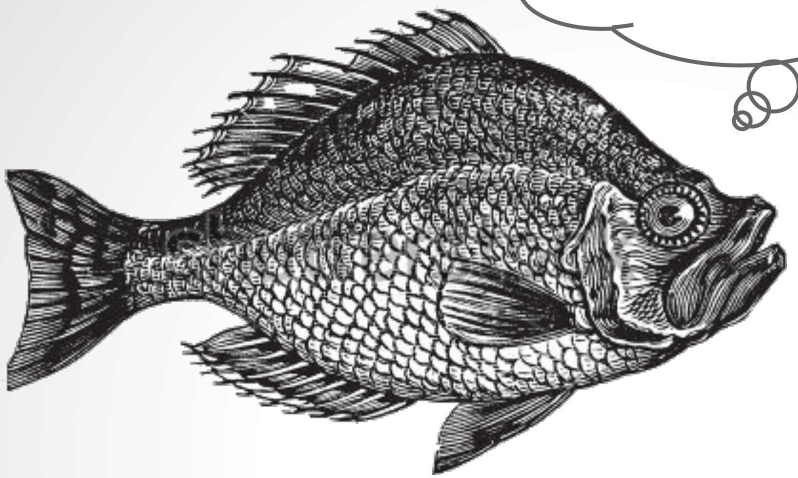


"The theory of probabilities is at bottom nothing but common sense reduced to calculus; it enables us to appreciate with exactness that which accurate minds feel with a sort of instinct for which ofttimes they are unable to account."

Pierre-Simon Laplace

Essai philosophique sur les Probabilités (1814)

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}\,!$$



Introduction to Bayesian Data Analysis with R.

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