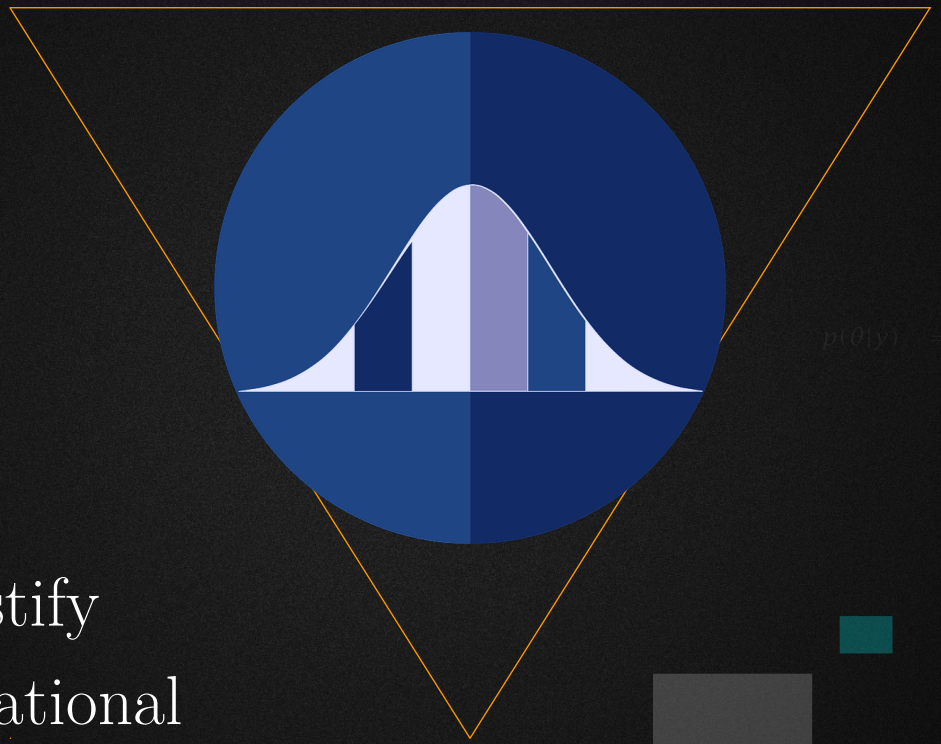


UNIGE R Lunch

Striving to Demystify Bayesian Computational Modelling

Speaker: Marco Wirthlin
@marcowirthlin

Abstract and Context of the Talk

ONLINE VERSION

Abstract

Bayesian approaches to computational modelling have experienced a slow, but steady gain in recognition and usage in academia and industry alike, accompanying the growing availability of evermore powerful computing platforms at shrinking costs. Why would one use such techniques? How are those models conceived and implemented? Which is the recommended workflow? Why make life hard when there are P-values?

In his talk, Marco Wirthlin will first attempt an introduction to statistical notions supporting Bayesian computation and explain the difference to the Frequentist framework. In the second half, an example of a recommended workflow is outlined on a simple toy model, with simulated data. Live coding will be used as much as possible to illustrate concepts on an implementational level in the R language. Ample literature and media references for self-learning will be provided during the talk.

Context and Licence

This talk was performed in the context of the “R Lunch” on the 29 of October 2019 at the University of Geneva and was organized by **Elise Tancoigne** (@tancoigne) & **Xavier Adam** (@xvrdm). Many thanks for inviting me! :D

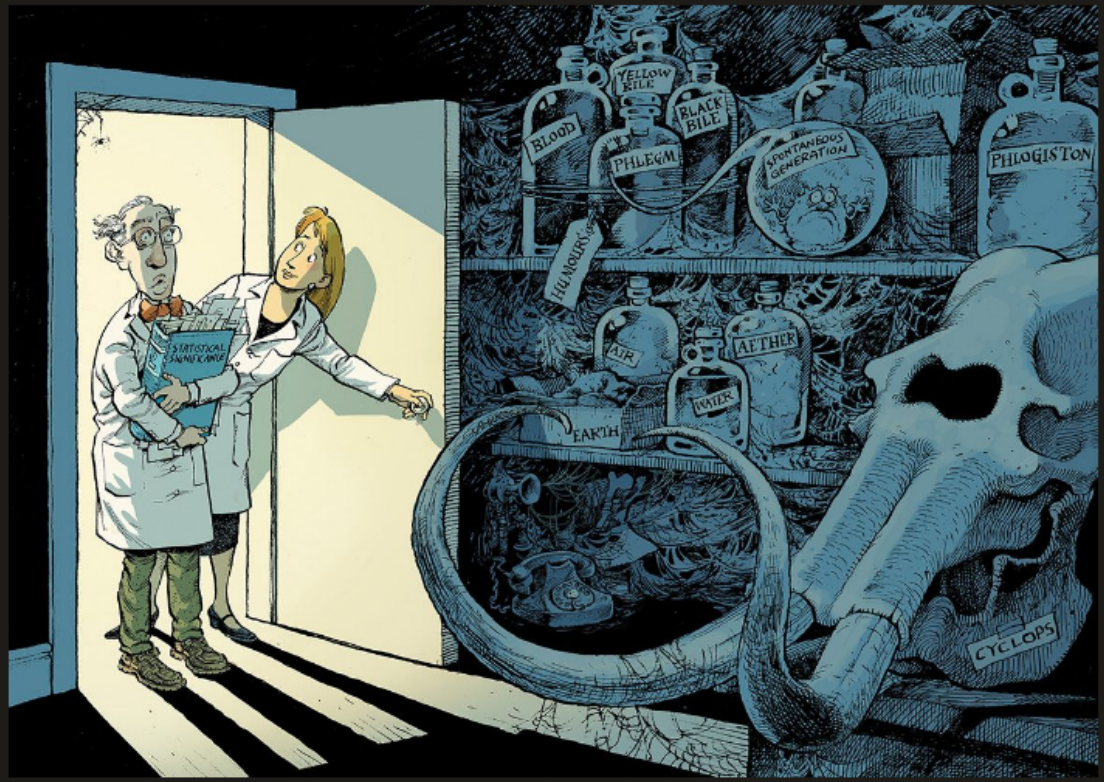
The difference between Bayesian and Frequentist statistics is how probability theory is applied to achieve their respective goals.

COMMENT · 20 MARCH 2019

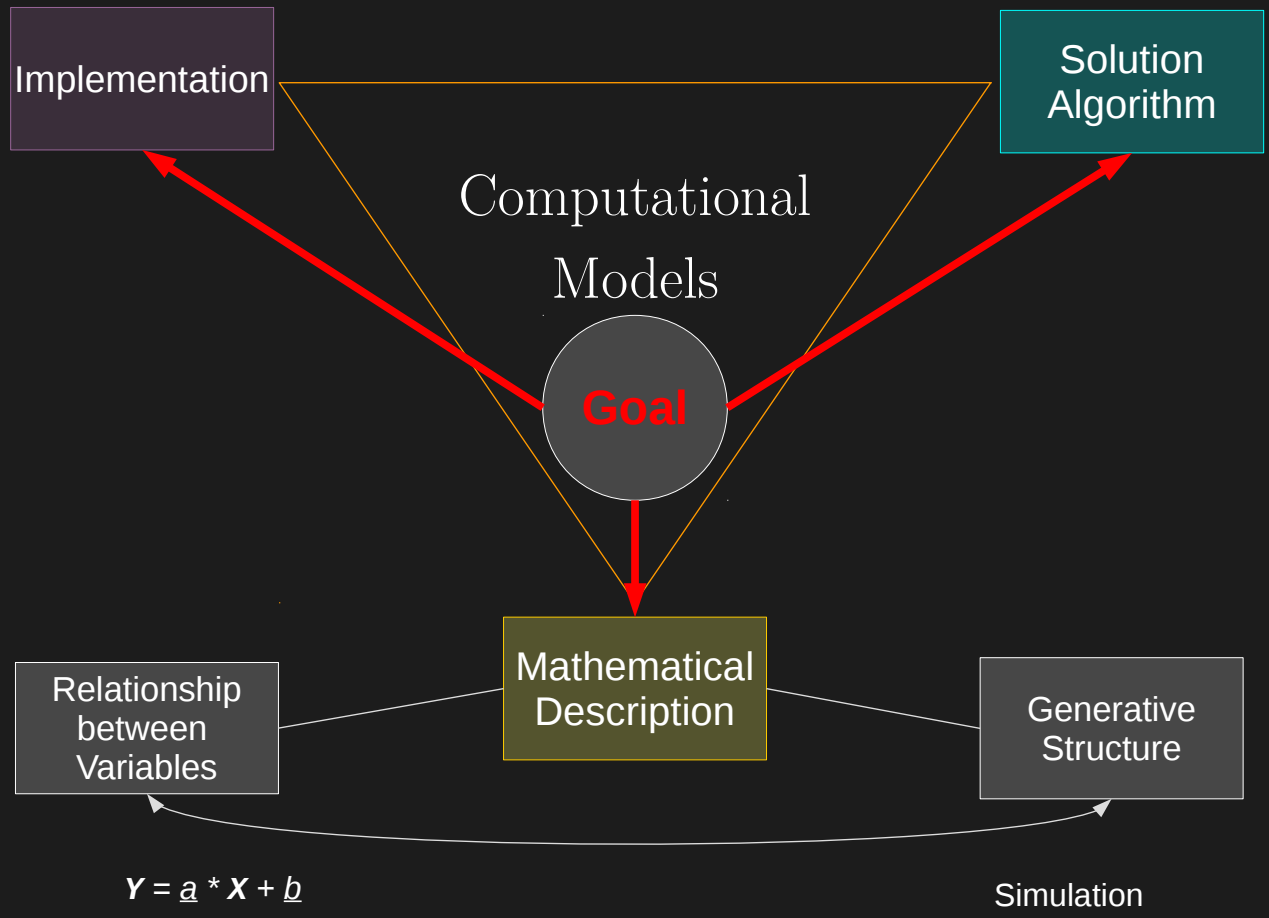
Scientists rise up against statistical significance

Valentin Amrhein, Sander Greenland, Blake McShane and more than 800 signatories call for an end to hyped claims and the dismissal of possibly crucial effects.

Valentin Amrhein, Sander Greenland & Blake McShane



Conceptual Hygiene



Fitting Linear Mixed-Effects Models Using lme4

Douglas Bates University of Wisconsin-Madison
Martin Mächler ETH Zurich
Benjamin M. Bolker McMaster University
Steven C. Walker McMaster University

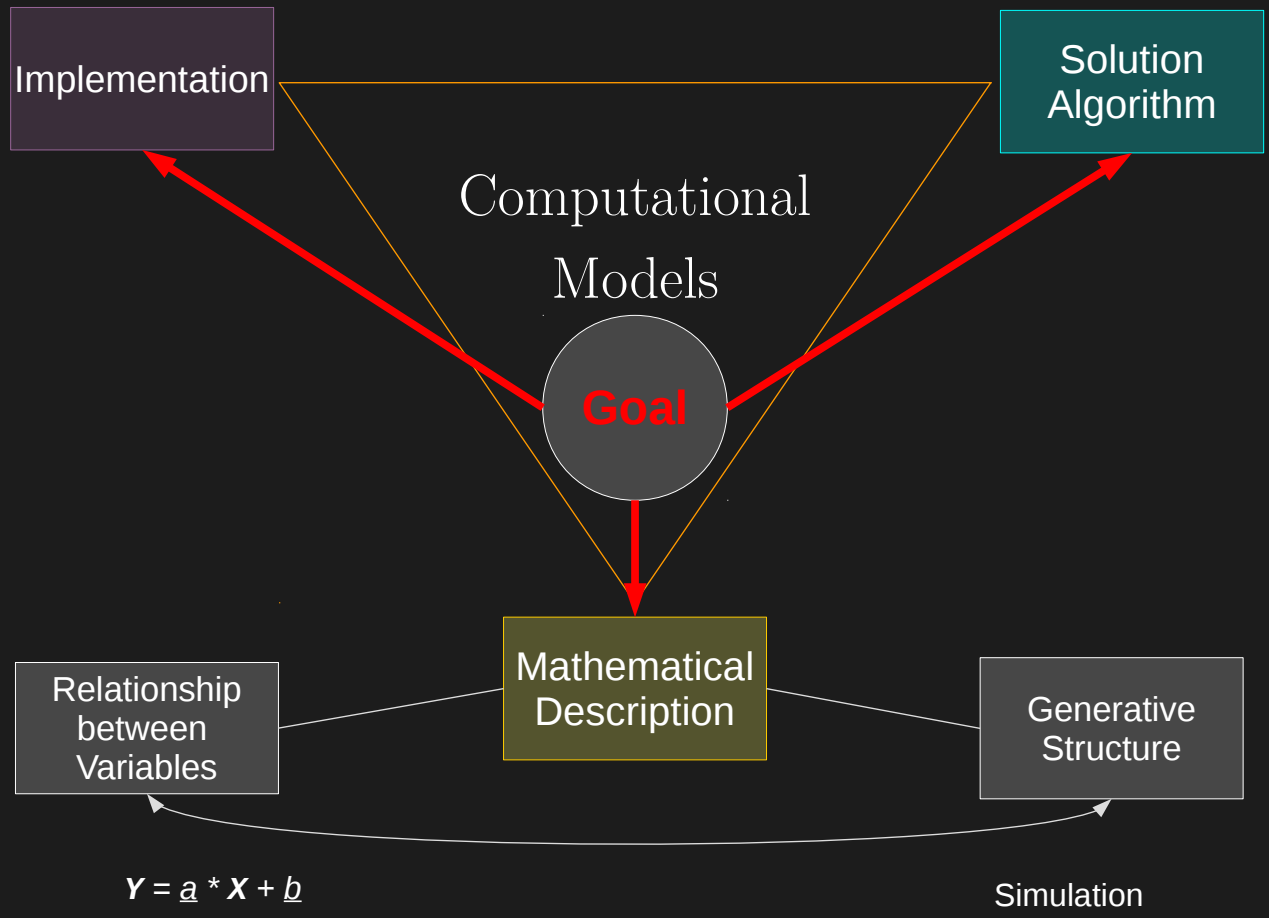
Abstract

Maximum likelihood or restricted maximum likelihood (REML) estimates of the parameters in linear mixed-effects models can be determined using the `lmer` function in the **lme4** package for R. As for most model-fitting functions in R, the model is described in

Module		R function	Description
Formula module	(Section 2)	`lFormula`	Accepts a mixed-model formula, data, and other user inputs, and returns a list of objects required to fit a linear mixed model.
Objective function module	(Section 3)	`mkLmerDevfun`	Accepts the results of `lFormula` and returns a function to calculate the deviance (or restricted deviance) as a function of the covariance parameters, $\boldsymbol{\theta}$.
Optimization module	(Section 4)	`optimizeLmer`	Accepts a deviance function returned by `mkLmerDevfun` and returns the results of the optimization of that deviance function.
Output module	(Section 5)	`mkMerMod`	Accepts an optimized deviance function and packages the results into a useful object.

Table 1: The high-level modular structure of `lmer`.

Conceptual Hygiene

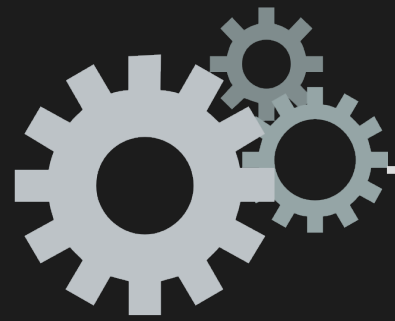


What is a generative model?

Parameters

$\theta, \mu, \xi, \ldots$

Hypothesis of underlying mechanisms
AKA: “Learning the class”
No shorcuts!



$\mathcal{D} =$
[2, 8, ..., 9]

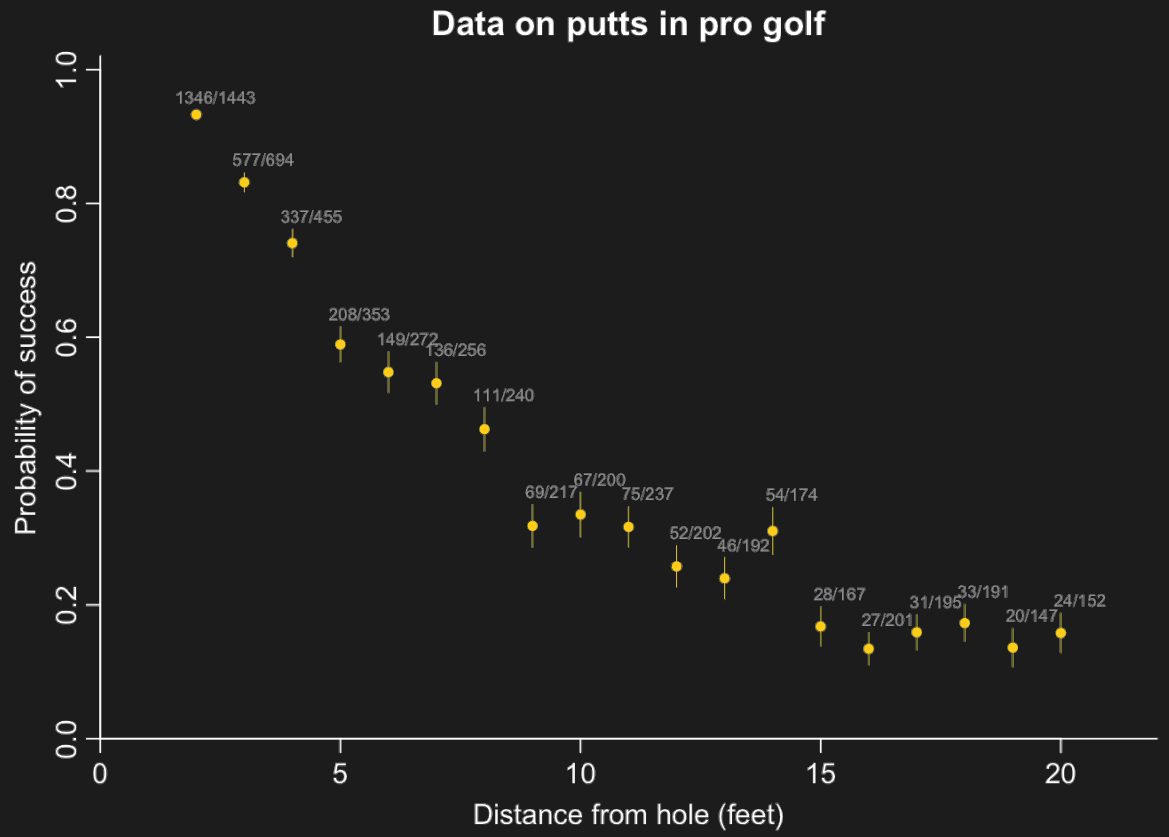
Model

Bayesian Inference

- *Ng, A. Y. and Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In Advances in neural information processing systems, pages 841–848.*
- *Rasmus Bååth, Video Introduction to Bayesian Data Analysis, Part 1: What is Bayes?: https://www.youtube.com/watch?time_continue=366&v=3OJEae7Qb_o*

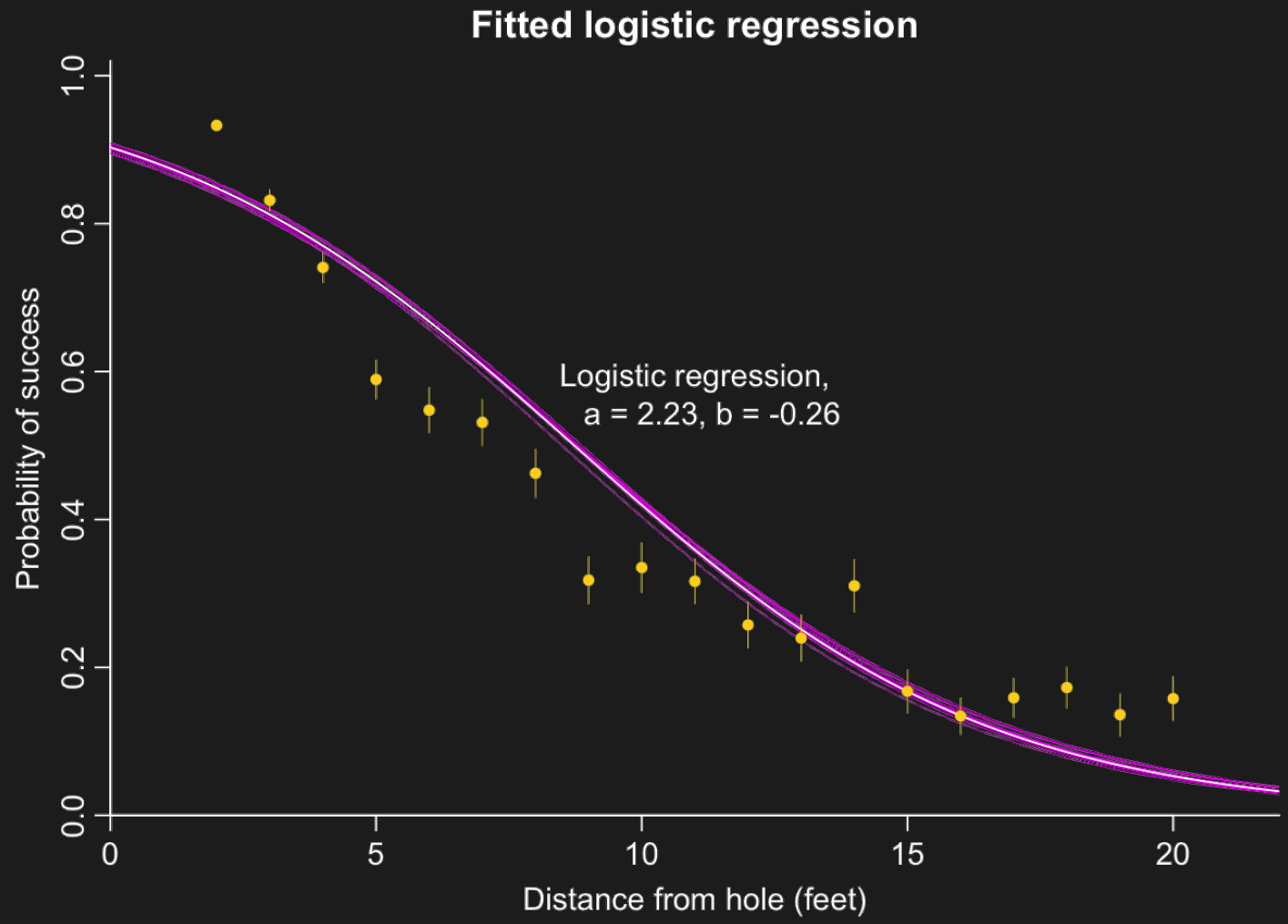
Golf Example (Gelman 2019)

Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html



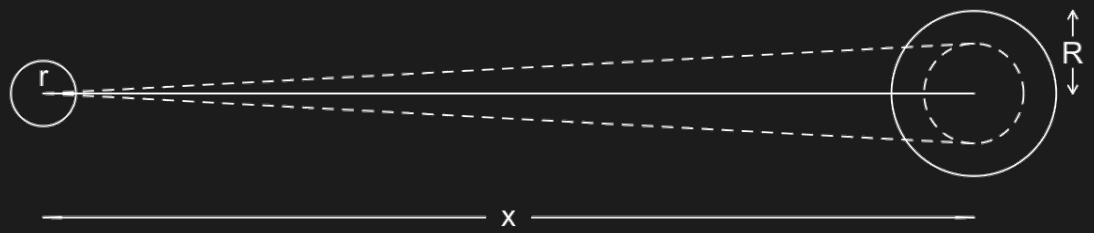
$$y_j \sim \text{binomial}(n_j, \text{logit}^{-1}(a + bx_j)), \text{ for } j = 1, \ldots, J.$$

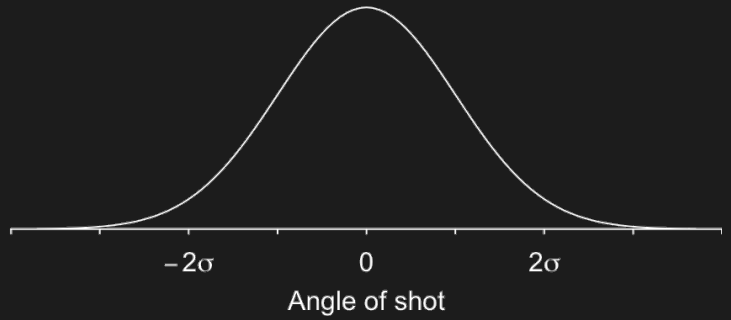
Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html



Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html

Lets think about how golf works!

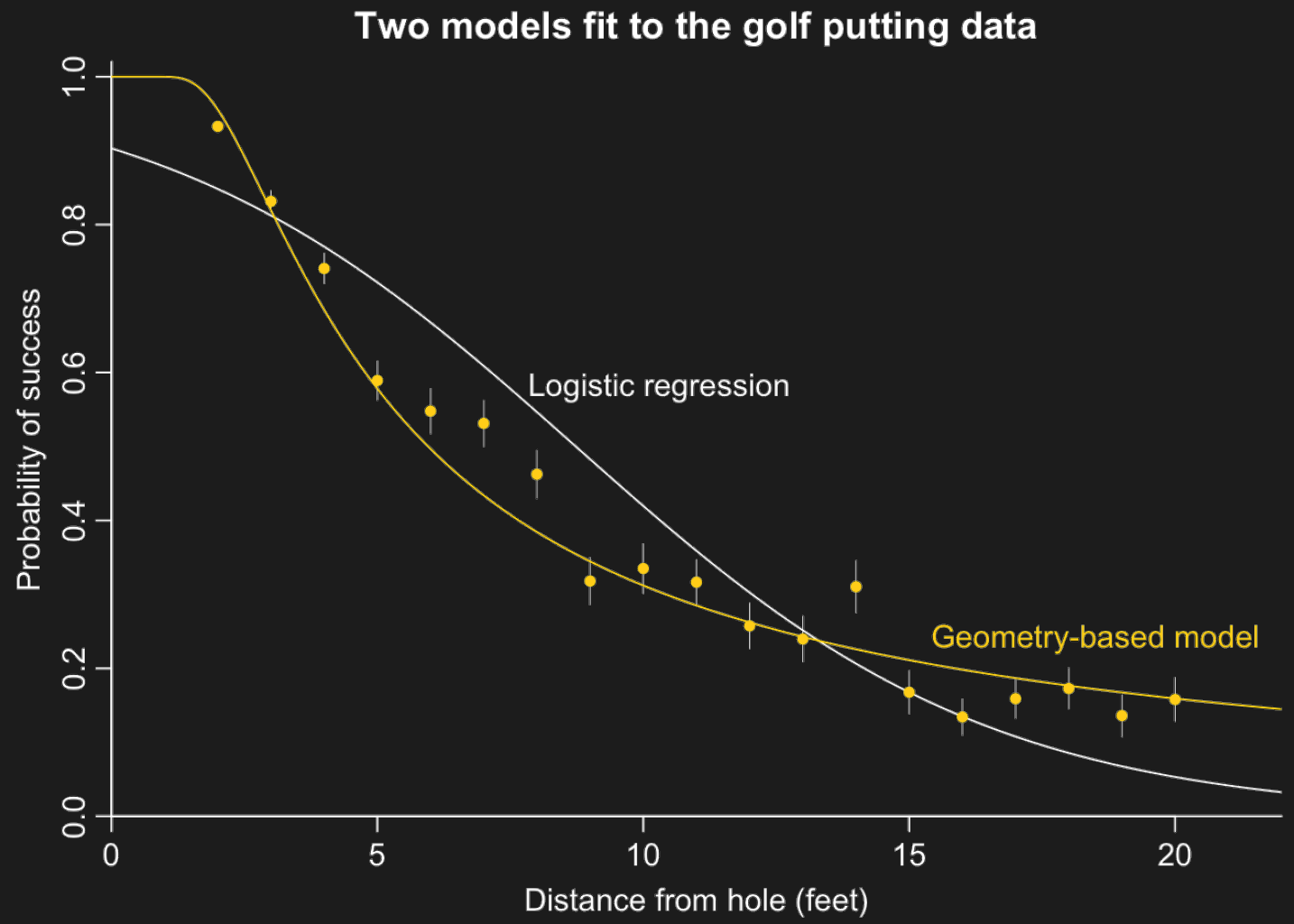




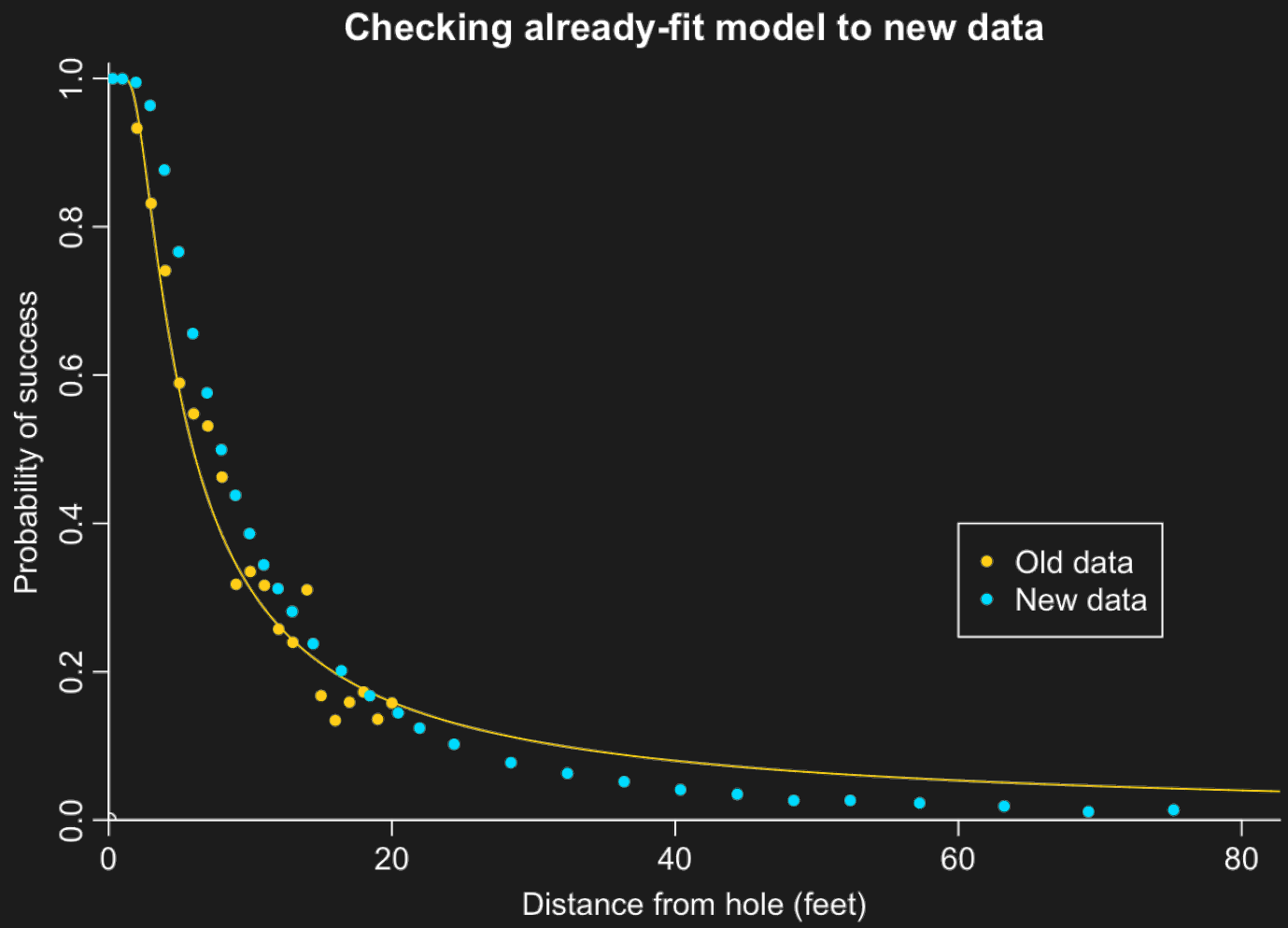
$$y_j \sim \text{binomial}(n_j, p_j)$$

$$p_j = 2\Phi\left(\frac{\sin^{-1}((R-r)/x_j)}{\sigma}\right) - 1, \text{ for } j = 1, \ldots, J.$$

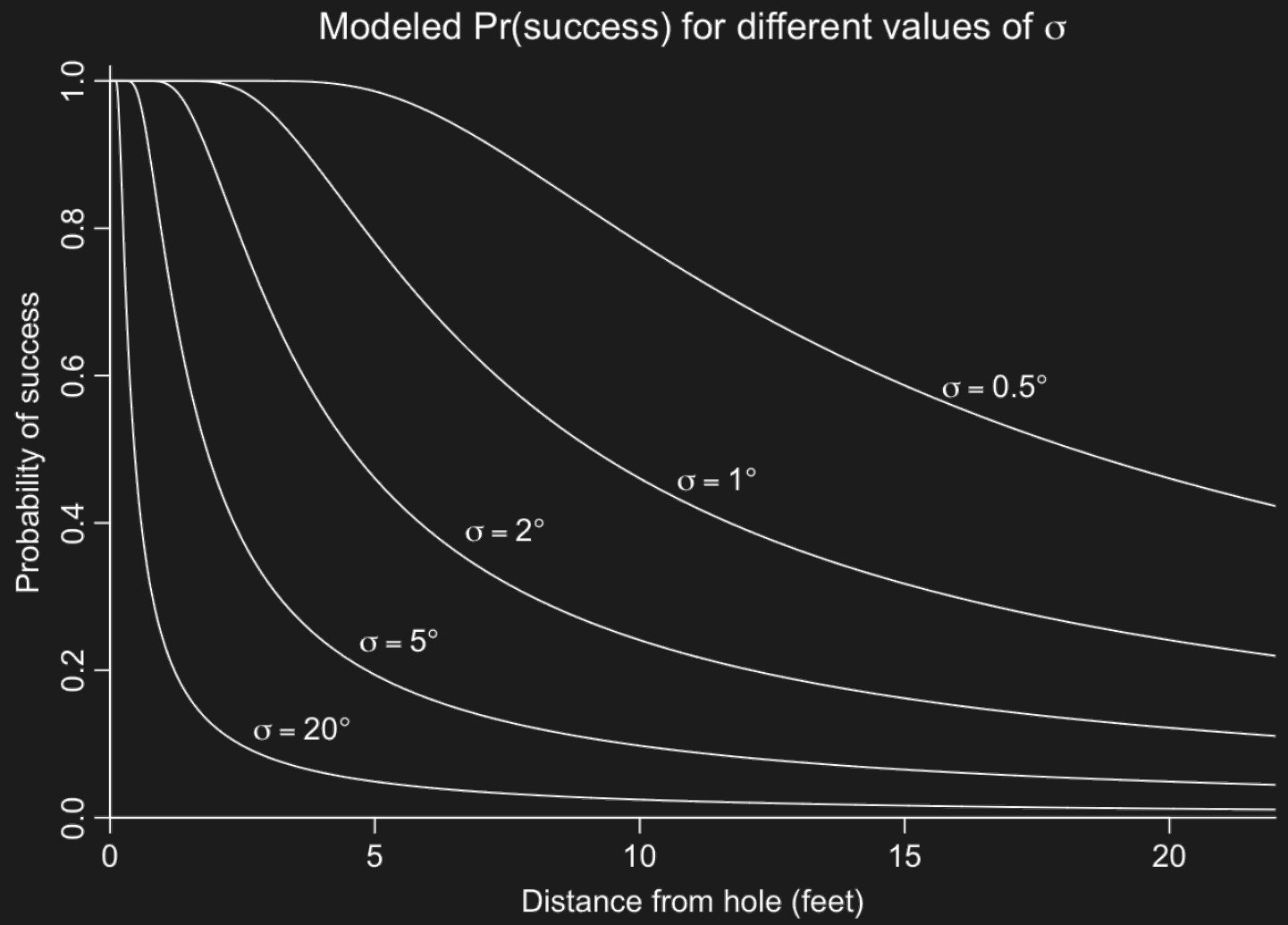
Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html



Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html



Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html



Andrew Gelman: https://mc-stan.org/users/documentation/case-studies/golf.html

Computational Models can have different Goals

Decision making

Fisherian (frequentist) statistics, hypothesis tests, p-values...

Quantifying how well we understand a phenomenon

Bayesian statistics, generative models, simulations...

Richard McElreath: Statistical Rethinking book and lectures (https://www.youtube.com/watch?v=4WVelCswXo4)

Rasmus Bååth, Video Introduction to Bayesian Data Analysis, Part 1: What is Bayes?: https://www.youtube.com/watch?time_continue=366&v=3OJEae7Qb_o

Frequentist statistics: parameter “fixed”
Bayesian statistics: parameter “variable”

¯\(°͡ ͜ʖ °͡)/¯

Frequentist statistics: maximum likelihood
Bayesian statistics: likelihood

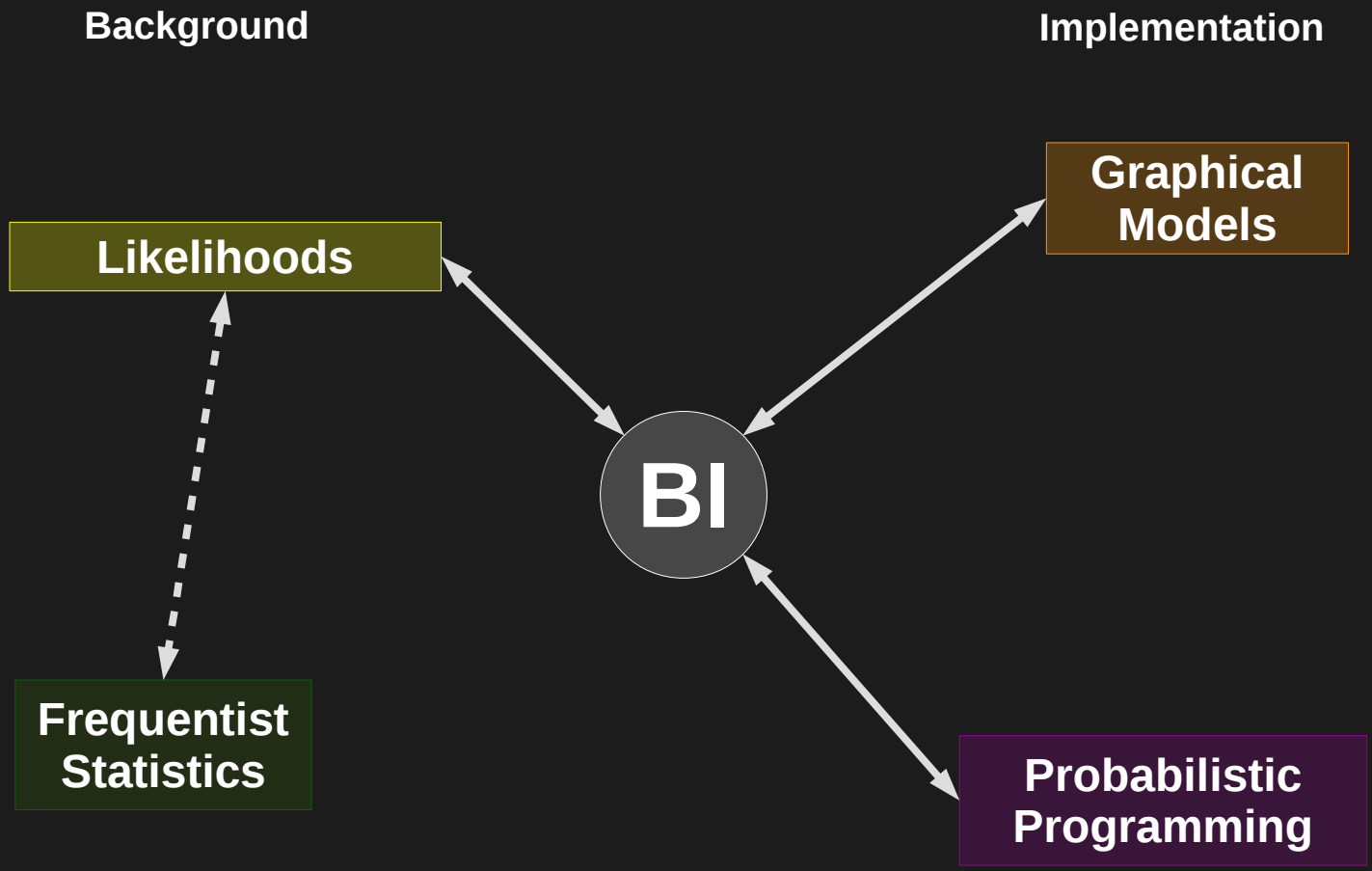
ヽ('Д')ノ

$$\frac{\overbrace{\sum_{1...N} \ln(p(\mathcal{D}|\sigma_k, d_k, a_k)}^{Likelihood} \overbrace{p(\sigma)p(d)p(a)}^{Priors}}{\underbrace{\iiint p(\mathcal{D}|\sigma_k, d_k, a_k)p(\sigma)p(d)p(a)\mathrm{d}\sigma_k\mathrm{d}d_k\mathrm{d}a_k}_{Evidence}}$$

¯_(⊙_ʖ⊙)_/¯

Rasmus Bååth, Video Introduction to Bayesian Data Analysis, Part 1: What is Bayes?: https://www.youtube.com/watch?time_continue=366&v=3OJEae7Qb_o

Bayesian Inference (BI)



Likelihoods

$$\mathcal{L} = p(\mathcal{D} \mid \theta)$$

Dashboard link: https://seneketh.shinyapps.io/Likelihood_Intuition

Normal Distribution

$x \sim \mathcal{N}(\mu, \sigma^2)$ "X is normally distributed"

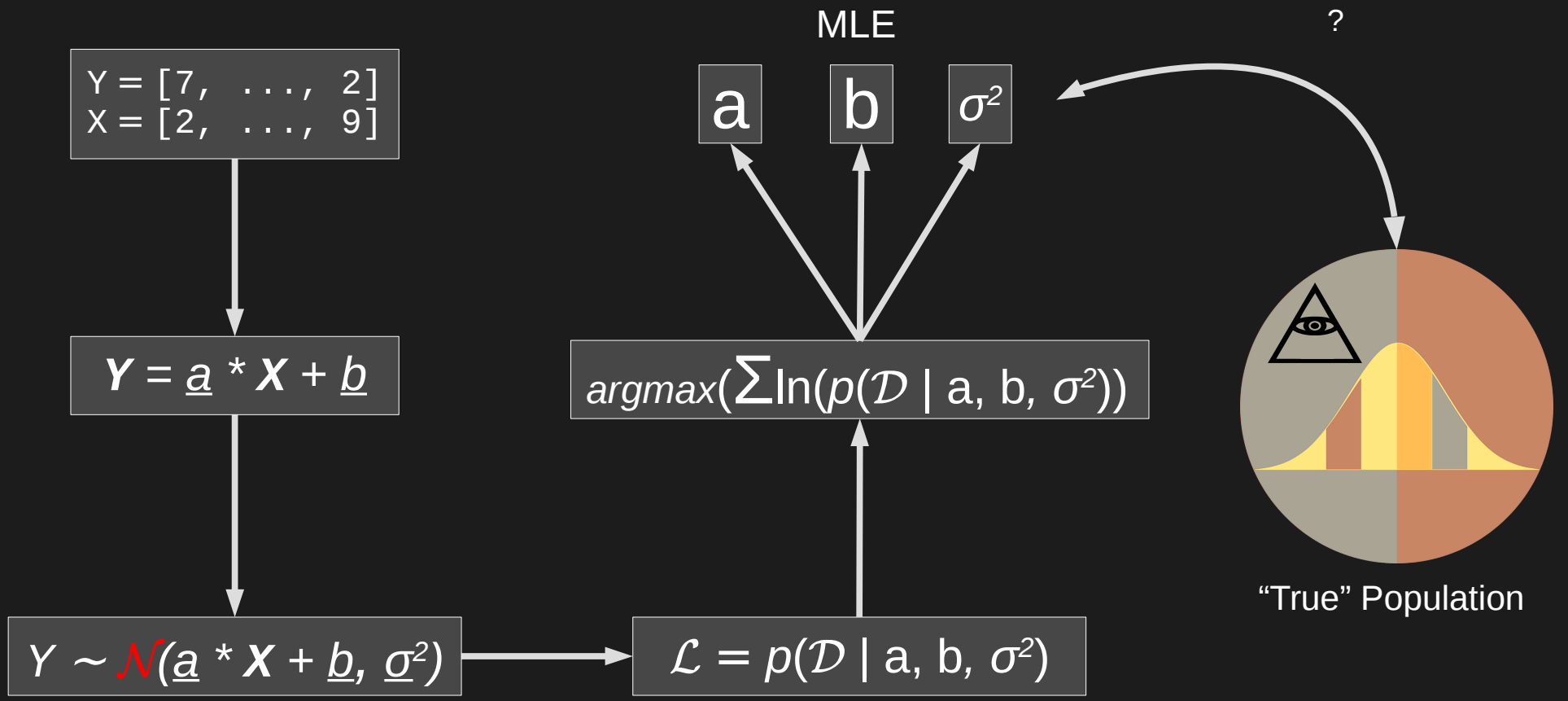
$\mathcal{L} = p(\mathcal{D} \mid \mu, \sigma^2)$ "The probability that $\mathcal{D}$ belongs to a distribution with mean μ and SD σ"

PDF: Fix parameters, vary data

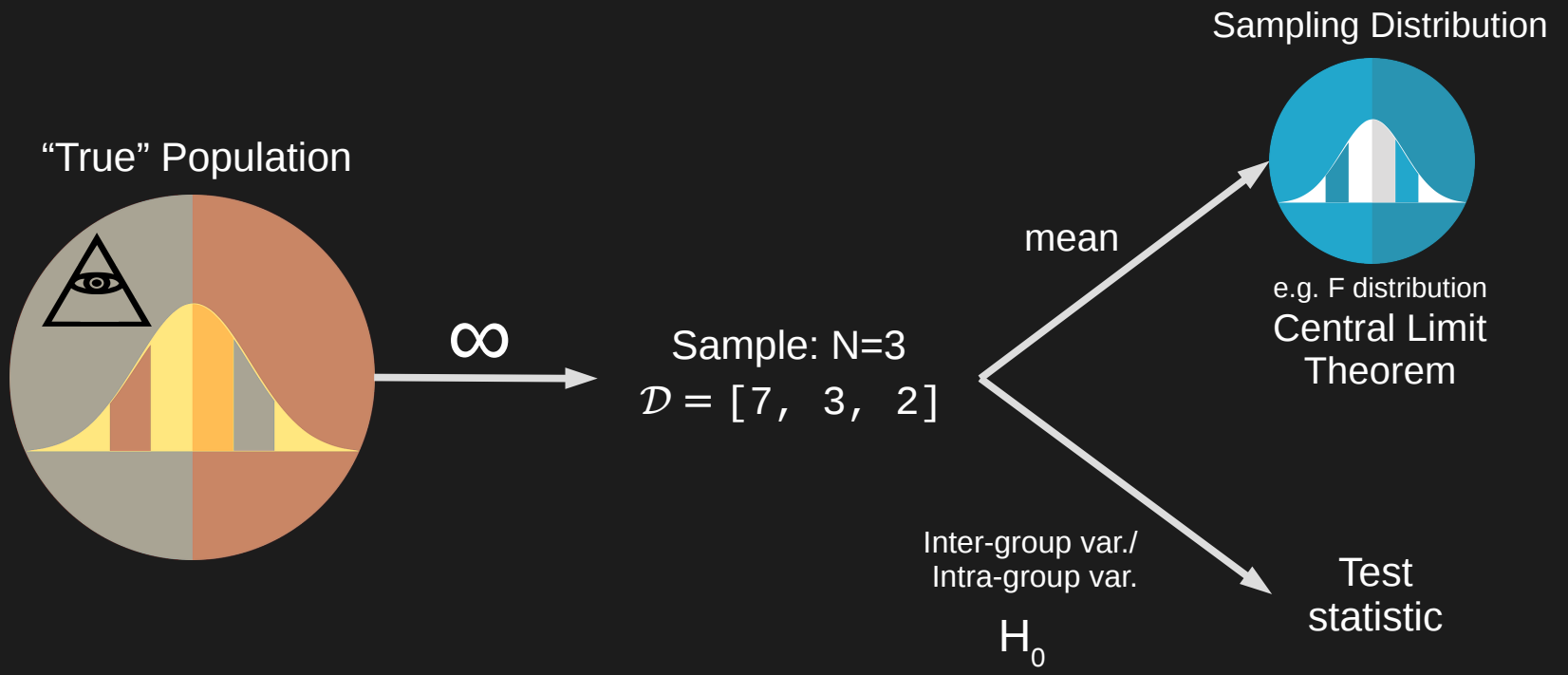
$\mathcal{L}$: Fix data, vary parameters

- *https://www.youtube.com/watch?v=ScduwntrMzc*

Frequentist Inference



Frequentist Inference



"Long range" probability

- *Sampling distribution applet: http://onlinestatbook.com/stat_sim/sampling_dist/index.html*

Frequentist Inference

“When a frequentist says that the probability for "heads" in a coin toss is 0.5 (50%) she means that in infinitively many such coin tosses, 50% of the coins will show "head"”.

If the H_0 is very unlikely “in the long run”, we accept H_1.

- *Sampling distribution applet: http://onlinestatbook.com/stat_sim/sampling_dist/index.html*

Bayesian Inference

1) Build a Generative model, imitate the data structure. (model)

2) Assign principled probabilities to your parameters. (priors)

3) Update those probabilities by incorporating data. (posteriors)

Probabilities: Epistemic/ontological uncertainty.

How well do we understand our phenomenon?

https://betanalpha.github.io/assets/case_studies/principled_bayesian_workflow.html

Assign probabilities to the parameters

$$\boldsymbol{Y} = \underline{a} * \boldsymbol{X} + \underline{b}$$

$$Y \sim \mathcal{N}(\underline{a} * \boldsymbol{X} + \underline{b}, \underline{\sigma}^2)$$

$\underline{a} \sim \mathcal{N}(1, 0.1)$ → $p(\mathrm{a})$

$\underline{b} \sim \mathcal{N}(4, 0.5)$ → $p(\mathrm{b})$

$\underline{\sigma}^2 \sim \mathcal{G}(1, 0.1)$ → $p(\sigma^2)$

$p(\mathcal{D} \mid \mathrm{a}, \mathrm{b}, \sigma^2)$ → $p(\mathcal{D} \mid \theta)$

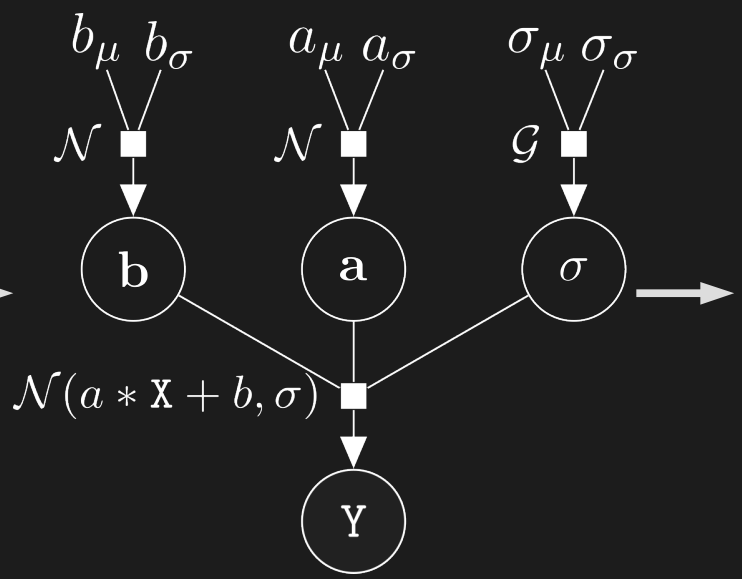
$$p(\mathcal{D}, \mathrm{a}, \mathrm{b}, \sigma^2) = p(\mathcal{D} \mid \mathrm{a}, \mathrm{b}, \sigma^2) * p(\mathrm{a}) * p(\mathrm{b}) * p(\sigma^2)$$

$$p\left(Data|\theta\right) \times p\left(\theta\right)$$

From model to code

$Y = \underline{a} * X + \underline{b}$

$\underline{a} \sim \mathcal{N}(1, 0.1)$
$\underline{b} \sim \mathcal{N}(4, 0.5)$
$\underline{\sigma^2} \sim \mathcal{G}(1, 0.1)$
$Y \sim \mathcal{N}(\underline{a} * X + \underline{b}, \underline{\sigma^2})$




```stan
//completely pooled model
data {
    int<lower=0> N; //amount data

    vector[N] X;
    vector[N] Y;
}
parameters {
    real a;
    real b;
    real<lower=0,upper=100> sigma;
}
transformed parameters {
     real y_hat;
     y_hat = a * X + b;
}
model {
    Y ~ normal(y_hat, sigma);
}
```

- *More examples: https://mc-stan.org/users/documentation/case-studies*

Implementation in R

```
//completely pooled model
data {
    int<lower=0> N; //amount data

    vector[N] X;
    vector[N] Y;
}
parameters {
    real a;
    real b;
    real<lower=0,upper=100> sigma;
}
transformed parameters {
     real y_hat;
     y_hat = a * X + b;
}
model {
    Y ~ normal(y_hat, sigma);
}
```

```
library(rstan)
library(rstantools)

pooled_data = list(
    'N'= length(data$Y),
    'X'= data$X,
    'Y'= data$Y)

 fit ← stan(model_code = pooled,
            data = unpooled_data,
            iter = 1000,
            warmup = 500
            chains = 4)

estimates ← rstan::extract(fit, permuted = TRUE)
```

Stan

https://mc-stan.org/

Stan Blocks

```
data {
//Input the data from R
}

parameters {
//Declare the parameters of the model
}

transformed parameters {
//Perform calculations on or between parameters
}

model {
//Configure probabilistic quantities and link with data
}

generated quantities {
//Perform data simulations
}
```

https://dougj892.github.io/bayes4ie2/texts/3_intro_to_stan/

```
//completely pooled model
data {
    int<lower=0> N; //amount data

    vector[N] X;
    vector[N] Y;
}
parameters {
    real a;
    real b;
    real<lower=0,upper=100> sigma;
}
transformed parameters {
     real y_hat;
     y_hat = a * X + b;
}
model {
    Y ~ normal(y_hat, sigma);
}
```

Golf Again!

$$y_j \sim \text{binomial}(n_j, \text{logit}^{-1}(a + bx_j)), \text{ for } j = 1, \ldots, J.$$

Golf Data

```
//golf putting data from berry (1996)
//distance(feet), #tries, #successes
x n y
2 1443 1346
3 694 577
4 455 337
5 353 208
6 272 149
7 256 136
8 240 111
9 217 69
10 200 67
11 237 75
12 202 52
13 192 46
14 174 54
15 167 28
16 201 27
17 195 31
18 191 33
19 147 20
20 152 24
```

Logistic Golf Model

```
data {
  int J;
  int n[J];
  vector[J] x;
  int y[J];
}
parameters {
  real a;
  real b;
}
model {
  y ~ binomial_logit(n, a + b*x);
}
```

Corresponding R Code

```
golf_data <- list(x=x, y=y, n=n, J=J)

fit_logistic <- stan("golf_logistic.stan",
                     data = golf_data)

a_sim <- extract(fit_logistic)$a

b_sim <- extract(fit_logistic)$b
```

https://github.com/stan-dev/example-models/tree/master/knitr/golf

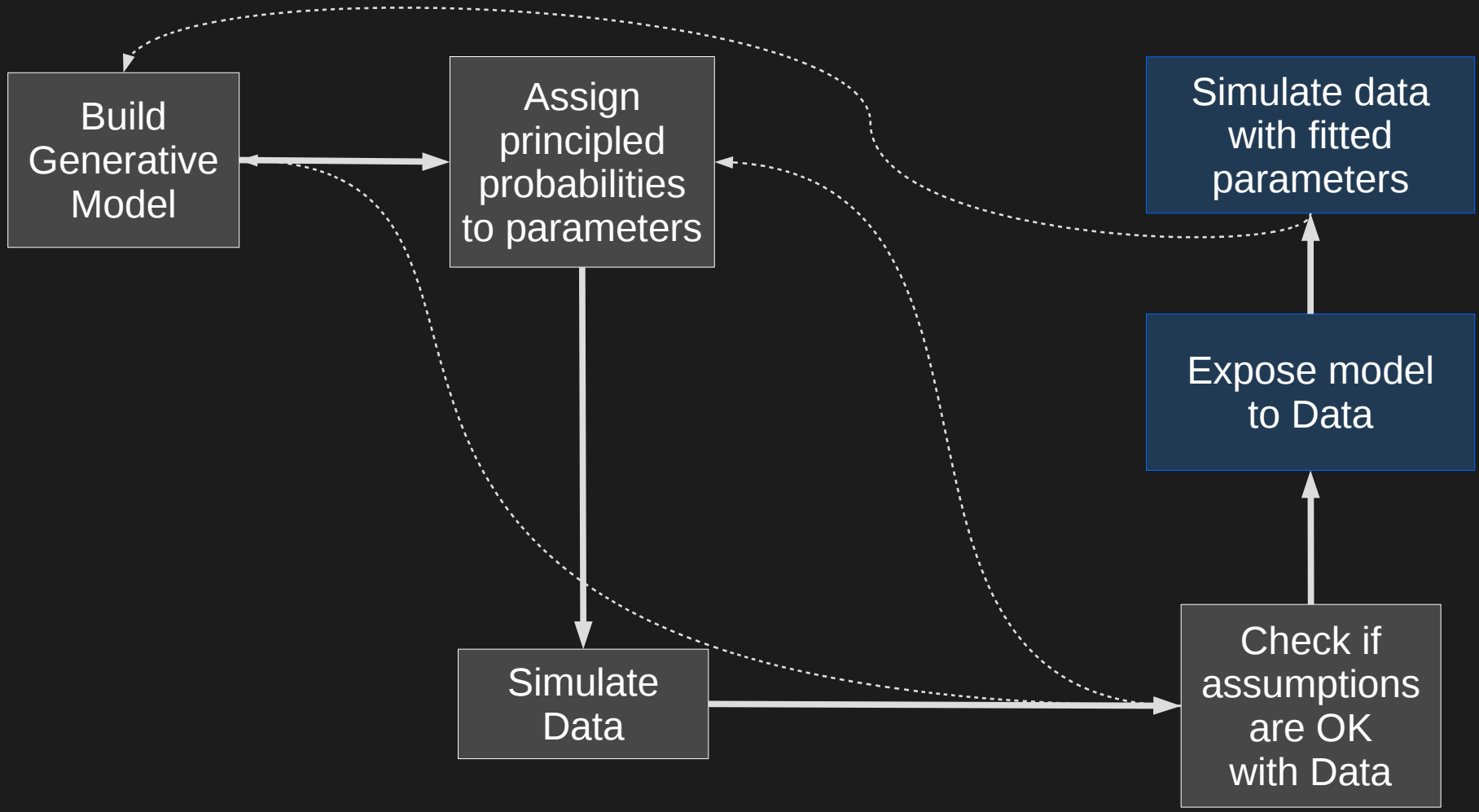
Model Output

```
Inference for Stan model: golf_logistic.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

   mean se_mean   sd   25%   50%   75% n_eff Rhat
a  2.23       0 0.06  2.19  2.23  2.27  1157    1
b -0.26       0 0.01 -0.26 -0.26 -0.25  1170    1

Samples were drawn using NUTS(diag_e) at Tue Oct  1 15:56:33 2019.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

Bayesian Workflow



- https://betanalpha.github.io/assets/case_studies/principled_bayesian_workflow.html
- Toward a principled Bayesian workflow in cognitive science: https://arxiv.org/abs/1904.12765

Short demo.

Thank you for your attention... and endurance!

Additional Slides

Sources, links and more!

Who to follow on Twitter?

- **Chris Fonnesbeck** @fonnesbeck (pyMC3)
- **Thomas Wiecki** @twiecki (pyMC3)
  Blog: https://twiecki.io/ (nice intros)
- **Bayes Dose** @BayesDose (general info and papers)
- **Richard McElreath** @rlmcelreath (ecology, Bayesian statistics expert)
  All his lectures: https://www.youtube.com/channel/UCNJK6_DZvcMqNSzQdEkzvzA
- **Michael Betancourt** @betanalpha (Stan)
  Blog: https://betanalpha.github.io/writing/
  Specifically: https://betanalpha.github.io/assets/case_studies/principled_bayesian_workflow.html
- **Rasmus Bååth** @rabaath
  Great video series: http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-one/
- **Frank Harrell** @f2harrell (statistics sage)
  Great Blog: http://www.fharrell.com/
- **Andrew Gelman** @StatModeling (statistics sage)
  https://statmodeling.stat.columbia.edu/
- **Judea Pearl** @yudapearl
  Book of Why: http://bayes.cs.ucla.edu/WHY/ (more about causality, BN and DAG)
- AND MANY MORE!

All sources in one place!

About Generative vs. Discriminative models:

Ng, A. Y. and Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In Advances in neural information processing systems, pages 841–848.

Rasmus Bååth:
Video Introduction to Bayesian Data Analysis, Part 1: What is Bayes?:
https://www.youtube.com/watch?time_continue=366&v=3OJEae7Qb_o

When to use ML vs. Statistical Modelling:

Frank Harrell's Blog:
http://www.fharrell.com/post/stat-ml/
http://www.fharrell.com/post/stat-ml2/

Frequentist approach: How do sampling distributions work (applet):

http://onlinestatbook.com/stat_sim/sampling_dist/index.html

Bayesian inference and computation:

John Kruschke:
Doing Bayesian Data Analysis:
A Tutorial with R, JAGS, and Stan Chapter 5

Rasmus Bååth:
http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-two/

Richard McElreath:
Statistical Rethinking book and lectures
(https://www.youtube.com/watch?v=4WVelCswXo4)

Many model examples in Stan:

https://mc-stan.org/users/documentation/case-studies

About Bayesian Neural Networks:

https://alexgkendall.com/computer_vision/bayesian_deep_learning_for_safe_ai/
https://twiecki.io/blog/2018/08/13/hierarchical_bayesian_neural_network/

Volatility Examples:

Hidden Markov Models:
https://github.com/luisdamiano/rfinance17

Volatility Garch Model and Bayesian Workflow:
https://luisdamiano.github.io/personal/volatility_stan2018.pdf

Dictionary: Stats ↔ ML

https://ubc-mds.github.io/resources_pages/terminology/

The Bayesian Workflow:

https://betanalpha.github.io/assets/case_studies/principled_bayesian_workflow.html

Algorithm explanation applet for MCMC exploration of the parameter space:

http://elevanth.org/blog/2017/11/28/build-a-better-markov-chain/

Probabilistic Programming Conference Talks:

https://www.youtube.com/watch?v=crvNIGyqGSU

Dictionary: Stats ↔ ML

Statistics		Machine learning / AI
Estimation/Fitting	~	Learning
Hypothesis	~	Classification rule
Data Point	~	Example/ Instance
Regression	~	Supervised Learning
Classification	~	Supervised Learning
Covariates	~	Features
Parameters	~	Features
Response	~	Label
Factor	~	Factor (categorical variables)
Likelihood	~	Cost Function (sometimes)

Check: https://ubc-mds.github.io/resources_pages/terminology/ for more terminology

Slides about Bayesian Computation

Bayesian Inference

Eye color	Hair color				Marginal (Eye color)
	Black	Brunette	Red	Blond	
Brown	0.11	0.20	0.04	0.01	0.37
Blue	0.03	0.14	0.03	0.16	0.36
Hazel	0.03	0.09	0.02	0.02	0.16
Green	0.01	0.05	0.02	0.03	0.11
Marginal (hair color)	0.18	0.48	0.12	0.21	1.0

Eye color	Hair color				Marginal (Eye color)
	Black	Brunette	Red	Blond	
Blue	0.03/0.36 = 0.08	0.14/0.36 = 0.39	0.03/0.36 = 0.08	0.16/0.36 = 0.45	0.36/0.36 = 1.0

- *John Kruschke: Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan Chapter 5*
- *Rasmus Bååth: http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-two/*
- *Richard McElreath: Statistical Rethinking book and lectures (https://www.youtube.com/watch?v=4WVelCswXo4)*

Bayesian Inference

	Model parameter			
Data	...	**θ value**	...	**Marginal**
⋮		⋮		⋮
***D* value**	...	$p(D, \theta) = p(D\|\theta)\, p(\theta)$	...	$p(D) = \sum_{\theta^*} p(D\|\theta^*)\, p(\theta^*)$
⋮		⋮		⋮
Marginal	...	$p(\theta)$	...	

$$\overbrace{p\left(\theta|Data\right)}^{Posterior} \propto \overbrace{p\left(Data|\theta\right)}^{Likelihood} \times \overbrace{p\left(\theta\right)}^{Prior}$$

- *John Kruschke: Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan Chapter 5*
- *Rasmus Bååth: http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-two/*
- *Richard McElreath: Statistical Rethinking book and lectures (https://www.youtube.com/watch?v=4WVelCswXo4)*

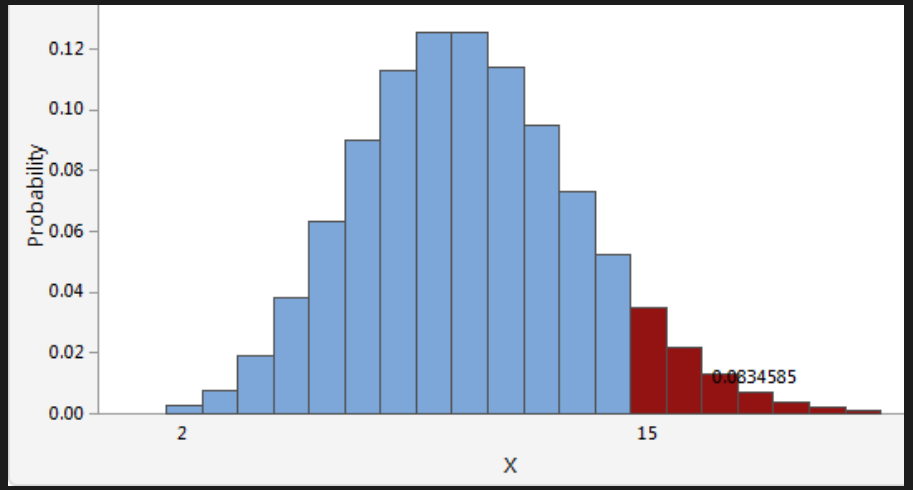
Bayesian Inference

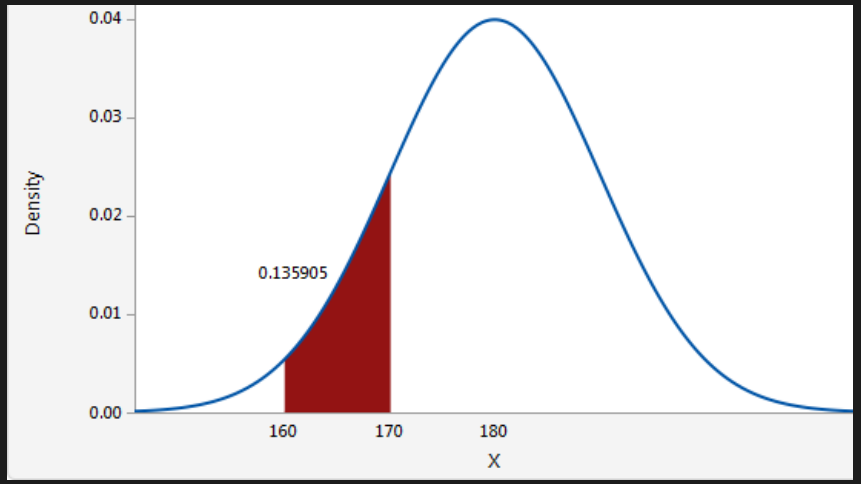
Discrete Values: Just sum it up!

:)

Cont. Values: Integration over complete parameter space...

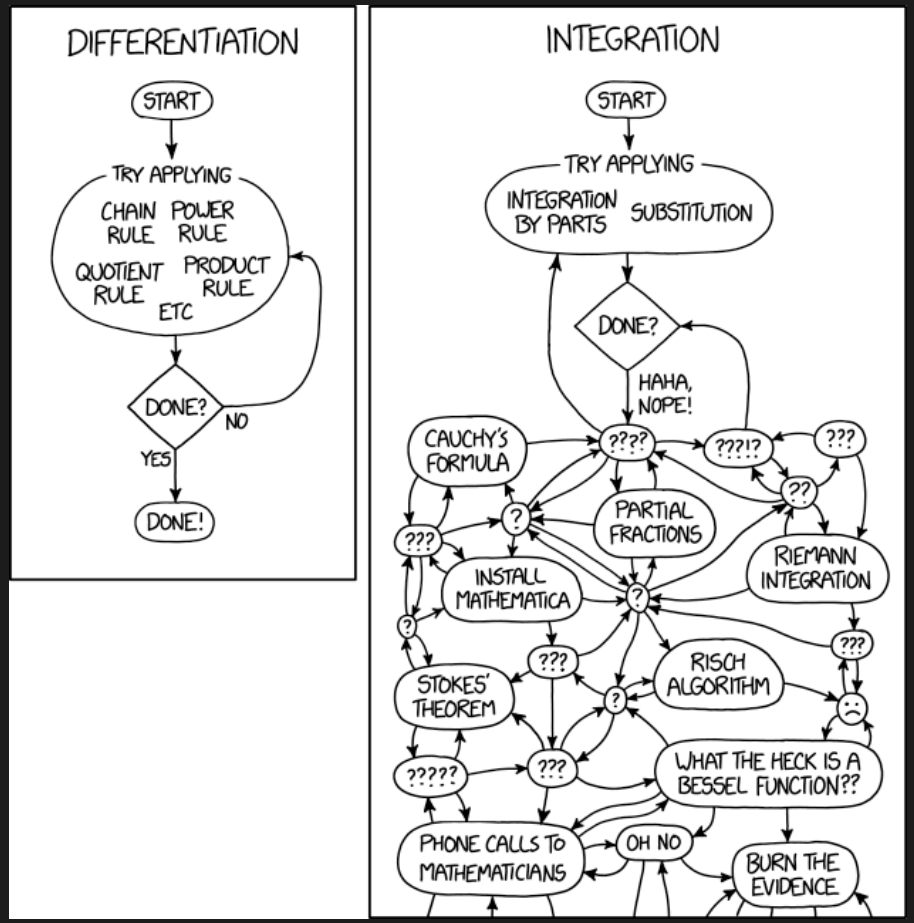
:(





- *John Kruschke: Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan Chapter 5*
- *Rasmus Bååth: http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-two/*
- *Richard McElreath: Statistical Rethinking book and lectures (https://www.youtube.com/watch?v=4WVelCswXo4)*

Bayesian Inference:



Averaging over the complete parameter space via integration is impractical!

Solution: We sample from the conjugate probability distribution with smart MCMC algorithms!

(Subject of another talk)

- *John Kruschke: Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan Chapter 5*
- *Rasmus Bååth: http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-two/*
- *Richard McElreath: Statistical Rethinking book and lectures (https://www.youtube.com/watch?v=4WVelCswXo4)*

Bayesian Inference

	Model parameter			
Data	...	**θ value**	...	**Marginal**
D value	...	$p(D, \theta) = p(D\|\theta)\, p(\theta)$	...	$p(D) = \sum_{\theta^*} p(D\|\theta^*)\, p(\theta^*)$
Marginal	...	$p(\theta)$	...	

Lets compute this and sample from it!

- *http://elevanth.org/blog/2017/11/28/build-a-better-markov-chain/*
- *John Kruschke: Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan Chapter 5*
- *Rasmus Bååth: http://www.sumsar.net/blog/2017/02/introduction-to-bayesian-data-analysis-part-two/*
- *Richard McElreath: Statistical Rethinking book and lectures (https://www.youtube.com/watch?v=4WVelCswXo4)*

Principles of Bayesian Modeling

$$\overbrace{p\left(\theta|Data\right)}^{Posterior} \propto \overbrace{p\left(Data|\theta\right)}^{Likelihood} \times \overbrace{p\left(\theta\right)}^{Prior}$$

Knowledge gained after measurement.

Knowledge gained by the measurement.

What we know about a cognitive phenomenon before the measurement.

Updated believes

Model/ Hypothesis Data

To-date scientific knowledge.

Bayesian Formulations

$$p(\theta|y) = \frac{p(y|\theta)\,p(\theta)}{p(y)}$$

$$p(\theta|y) \propto p(y|\theta)\,p(\theta)$$

$$p(\theta|y) \propto p(y,\theta)$$

MCMC Requirements

Requirements:

- If:
  The prior distribution is specified by a function that is easily evaluated. This simply means that if you specify a value for θ, then the value of $p(\theta)$ is easily determined,
- And If:
  The value of the likelihood function, $p(D|\theta)$, can be computed for any specified values of D and θ.
- Then:
  The method produces an approximation of the posterior distribution, $p(\theta|D)$, in the form of a large number of θ values sampled from that distribution (same as we would sample people from a determined population).

Metropolis Algorithm Steps

The Metropolis algorithm proceeds as follows. Start at an arbitrary initial value of θ (in the valid range). This is the current value, denoted θ_{cur}. Then:

1. Randomly generate a proposed jump, $\Delta\theta \sim \text{normal}(\mu = 0, \sigma)$ and denote the proposed value of the parameter as $\theta_{\text{pro}} = \theta_{\text{cur}} + \Delta\theta$.
2. Compute the probability of moving to the proposed value

$$p_{\text{move}} = \min\left(1, \frac{P(\theta_{\text{pro}})}{P(\theta_{\text{cur}})}\right) \qquad \text{generic Metropolis form}$$

$$= \min\left(1, \frac{p(D|\theta_{\text{pro}})p(\theta_{\text{pro}})}{p(D|\theta_{\text{cur}})p(\theta_{\text{cur}})}\right) \qquad P \text{ is likelihood times prior}$$

   If the proposed value θ_{pro} happens to land outside the permissible bounds of θ, the prior and/or likelihood is set to zero, hence p_{move} is zero.
3. Accept the proposed parameter value if a random value sampled from a $[0, 1]$ uniform distribution is less than p_{move}, otherwise reject the proposed parameter value and tally the current value again.

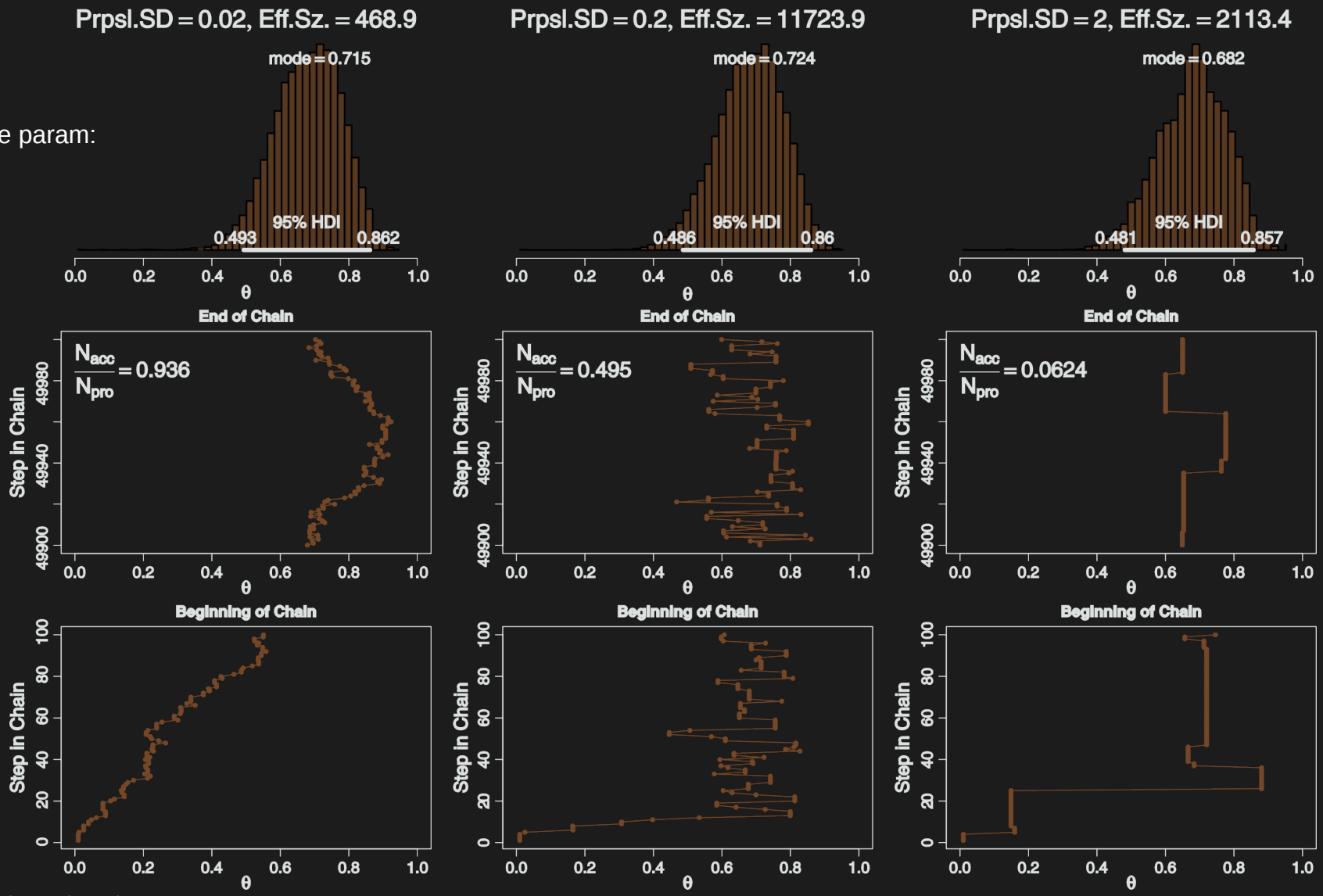
Source: Kruschke, J. (2014).

Metropolis Algorithm Steps

Source: Kruschke, J. (2014).

HMC Step-Sizes

True param:
0.7



Source: Kruschke, J. (2014).