Introduction to Bayesian Statistics

Arnaud Legrand

SMPE lecture
January 2023

- How to leverage your knowledge about the system?
- How to check whether it is reasonable or not?
- Given a some observations, how to generate "similar" values? ~~GANs ? No, not today~~
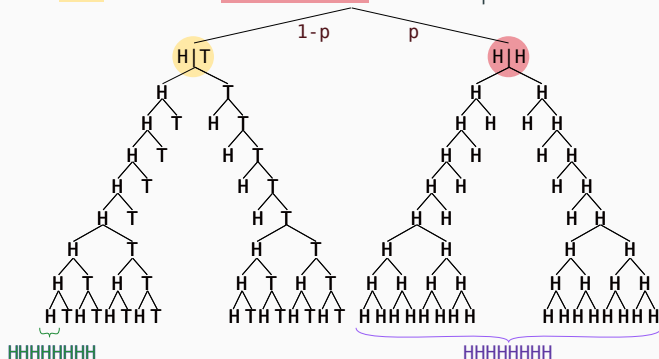
This talk:

1. Brief introduction to Bayesian statistics.
2. Brief introduction to Bayesian sampling with a brief presentation of STAN
3. A brief discussion on model selection

Bayesian Statistics
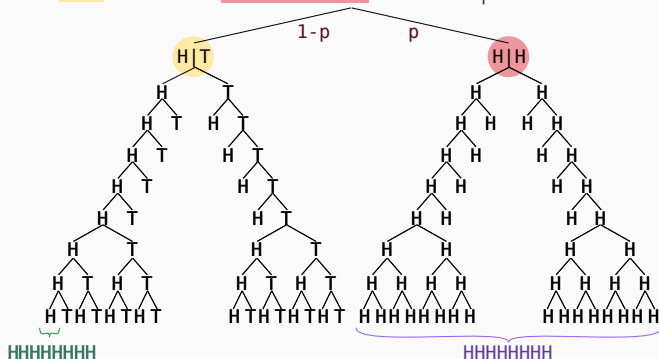
Consider a fair coin and two-headed one and pick one at random

The coin is tossed 8 times. The outcome is a head 8 times in a row.

· What is the probability that the coin is the two-headed one ?

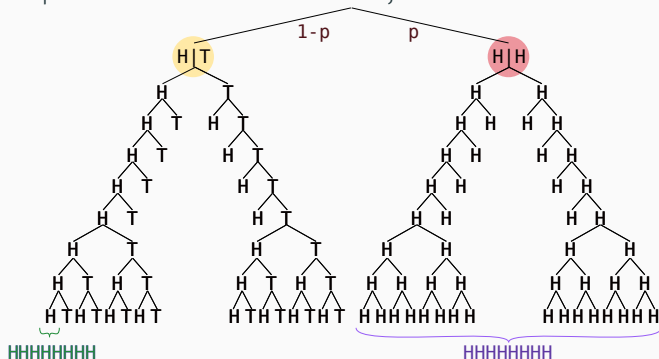Consider a **fair** coin and **two-headed** one and pick one at random

The coin is tossed 8 times. The outcome is a <u>head 8 times in a row</u>.

· What is the probability that the coin is the two-headed one ?

$$p[\textit{Two} - \textit{Headed}] = \frac{256}{256 + 1} \approx 0.996$$

Now let's put the two-headed one in a jar with 999 fair coins...

The coin is tossed 8 times. The outcome is a <u>head 8 times in a row</u>.

· What is the probability that the coin is the two-headed one ?

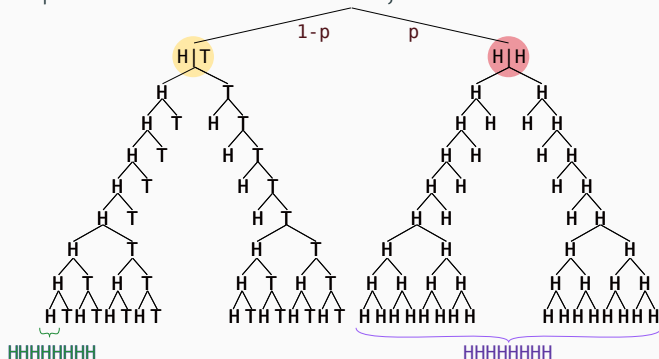Now let's put the two-headed one in a jar with 999 fair coins...

The coin is tossed 8 times. The outcome is a <u>head 8 times in a row</u>.

· What is the probability that the coin is the two-headed one ?

$$p[Two - Headed] = \frac{256 \times .001}{256 \times 0.001 + 1 \times 0.999} \approx 0.204$$

Notation

- $p(A)$ = probability that $\boxed{A}$ occurs
- $p(A, B)$ = probability that $\boxed{A \text{ and } B}$ occurs
- $p(A|B)$ = probability that $\boxed{A}$ occurs, given that $\boxed{B}$ occurs

Bayes Rule

Notation

- $p(A)$ = probability that $\boxed{A}$ occurs
- $p(A, B)$ = probability that $\boxed{A \text{ and } B}$ occurs
- $p(A|B)$ = probability that $\boxed{A}$ occurs, given that $\boxed{B}$ occurs

Conjunction rule

- $p(A, B) = p(A|B)p(B)$
- $p(B, A) = p(B|A)p(A)$

Notation

- $p(A)$ = probability that $\boxed{A}$ occurs
- $p(A, B)$ = probability that $\boxed{A \text{ and } B}$ occurs
- $p(A|B)$ = probability that $\boxed{A}$ occurs, given that $\boxed{B}$ occurs

Conjunction rule

- $p(A, B) = p(A|B)p(B)$
- $p(B, A) = p(B|A)p(A)$

Bayes rule Equate and divide by $p(B)$

$$\boxed{p(A|B) = \frac{p(B|A)p(A)}{p(B)}}$$

Model we assume $y \sim \mathcal{M}(\theta, x)$

- θ: Model parameters
- y: Dependent data (response)
- x: Independent data (covariates/predictors/constants)

Examples

- $y \sim \mathcal{N}(\mu, \sigma)$
- $y \sim x^2 + \mathcal{U}(\alpha, \beta)$
- $y \sim \mathcal{N}(\alpha x + \beta, \sigma)$
- $y \sim \mathcal{N}(\alpha \log(x) + \beta, \alpha' x + \beta')$

<u>Everyone</u>: Model data as random

Background on Bayesian Statistics

<u>Bayesians:</u> Data is fixed (observed), model parameters as random

$$p(\theta, y, x) = p(y, \theta, x)$$

$$p(\theta|y, x)p(y, x) = p(y|\theta, x)p(\theta, x)$$

Hence $p(\theta|y, x) = \dfrac{p(y|\theta, x)p(\theta, x)}{p(y, x)} = \dfrac{p(y|\theta, x)p(\theta)p(x)}{p(y, x)}$

$\propto p(y|\theta, x)p(\theta)$ (y, and x are fixed for a given data set)

Bayesians: Data is fixed (observed), model parameters as random

$$p(\theta, y, x) = p(y, \theta, x)$$
$$p(\theta|y, x)p(y, x) = p(y|\theta, x)p(\theta, x)$$

Hence $p(\theta|y, x) = \dfrac{p(y|\theta, x)p(\theta, x)}{p(y, x)} = \dfrac{p(y|\theta, x)p(\theta)p(x)}{p(y, x)}$

$\propto p(y|\theta, x)p(\theta)$ (y, and x are fixed for a given data set)

Bayes rule $\boxed{\underbrace{p(\theta|y, x)}_{\text{Posterior}} \propto \underbrace{p(y|\theta, x)}_{\text{Likelihood}} \underbrace{p(\theta, x)}_{\text{Prior}}}$ assuming $y \sim \mathcal{M}(\theta, x)$

- Posterior: The answer, probability distributions of parameters
- Likelihood:
  A (model specific) computable function of the parameters
- Prior: "Initial guess", existing knowledge of the system

The key to building Bayesian models is specifying the likelihood function, same as frequentist.

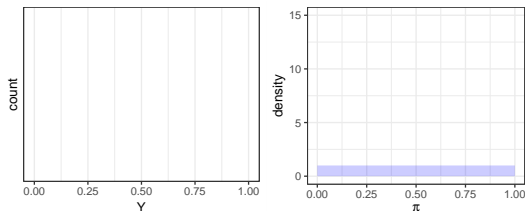> *When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.*
>
> *– From "The Gardian" quoted by MacKay*
> *in Information Theory, Inference and Learning Algorithms*

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- $p(y|\pi = 1/2)$
  $= \frac{(140+110)!}{110!140!} \cdot (\frac{1}{2})^{110} \cdot (\frac{1}{2})^{140}$
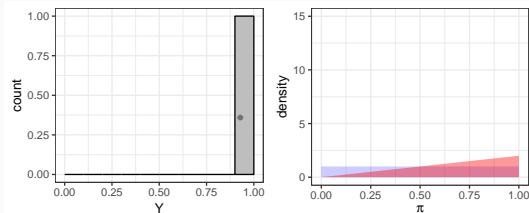 ≈ 0.00835

Check https://twitter.com/i/status/1447831352217415680

> *When spun on edge 250 times, a Belgian 1€ coin came up*
> *heads 140 times and tails 110. It looks very suspicious to me.*
> *If the coin were unbiased, the chance of getting a result as*
> *extreme as that would be less than 7%.*
>
> *– From "The Gardian" quoted by MacKay*
> *in Information Theory, Inference and Learning Algorithms*

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- $p(|y| \leq 110 | \pi = 1/2)$
  $= \sum_{k \leq 110} \frac{250!}{k!(250-k)!} \cdot \frac{1}{2^{250}}$
  ≈ 0.033

Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
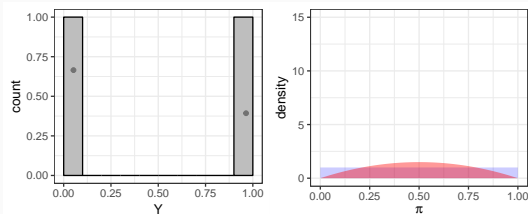
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

*– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms*

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

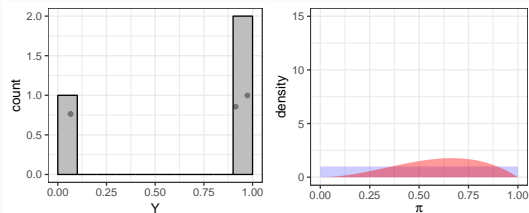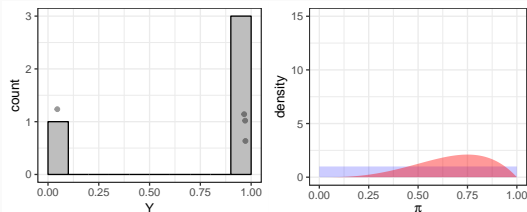$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0}\pi^{n_1} \cdot 1}{n_0!n_1!/(n_0 + n_1 + 1)!}$$

Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110 times. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
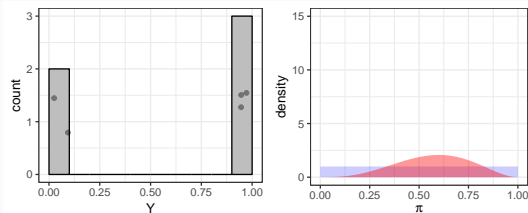
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110 times. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
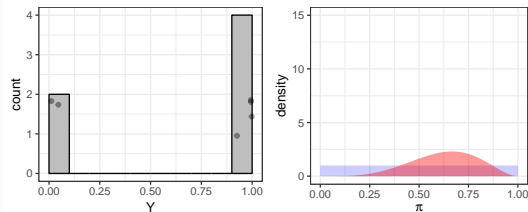
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
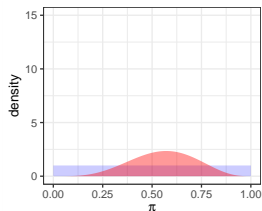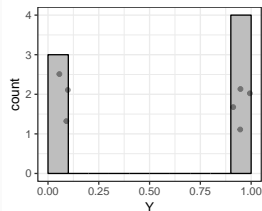
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
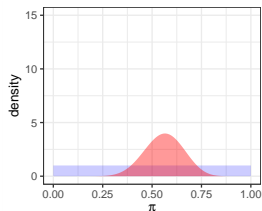
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
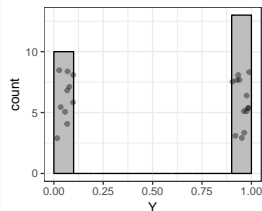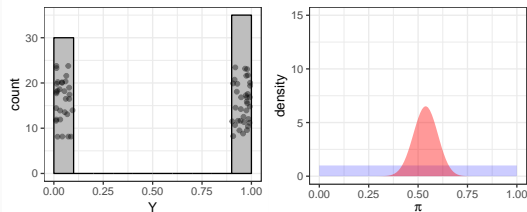
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \dots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
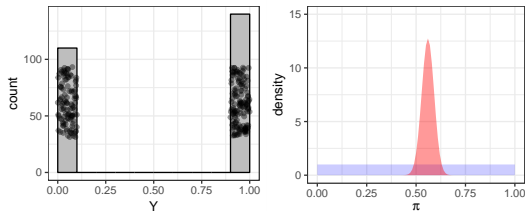
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
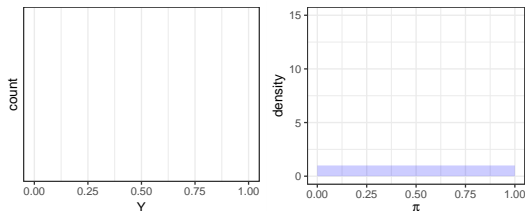
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$
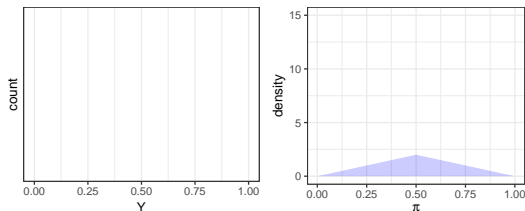
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1 - \pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1! / (n_0 + n_1 + 1)!}$$

Check https://twitter.com/i/status/1447831352217415680

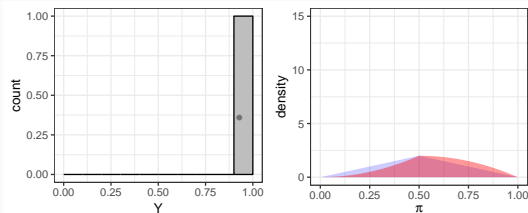*When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110 times. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.*

– From "The Gardian" quoted by MacKay in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{U}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot 1}{n_0! n_1!/(n_0 + n_1 + 1)!}$$

Check https://twitter.com/i/status/1447831352217415680

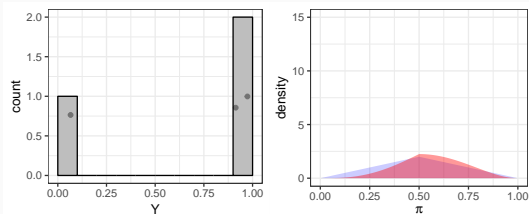*When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.*

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
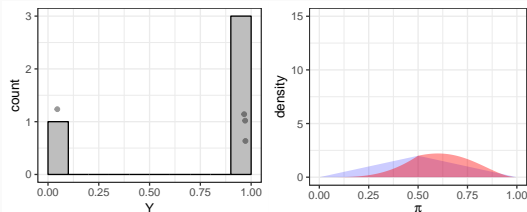
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
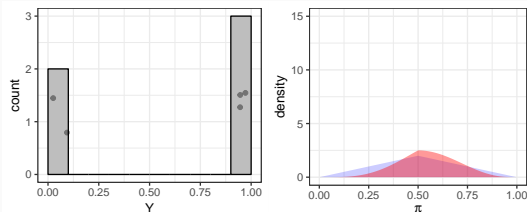
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
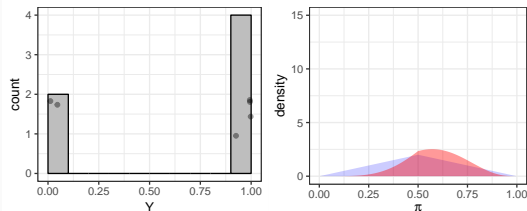
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0}\pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
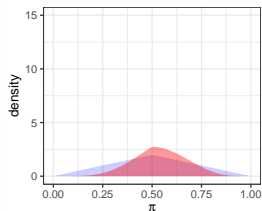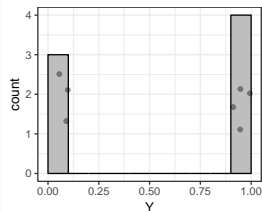
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
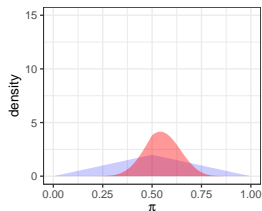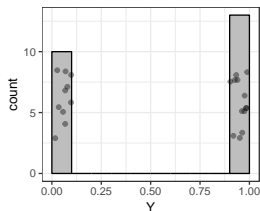
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1 - \pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
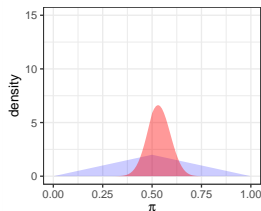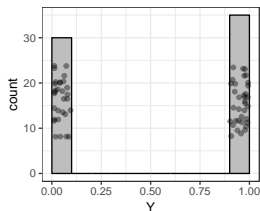
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1 - \pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
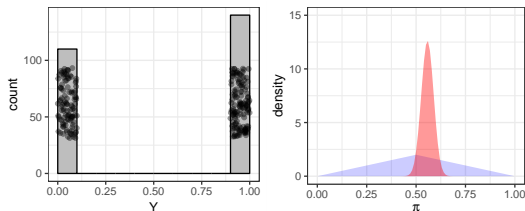
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0}\pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
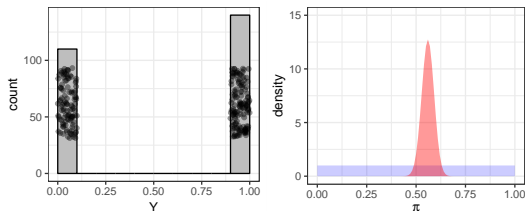
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
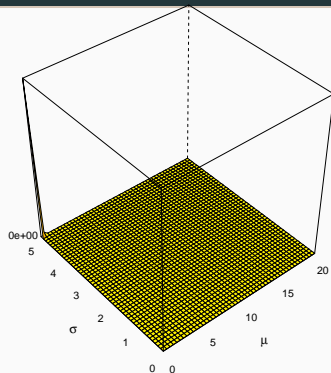- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$
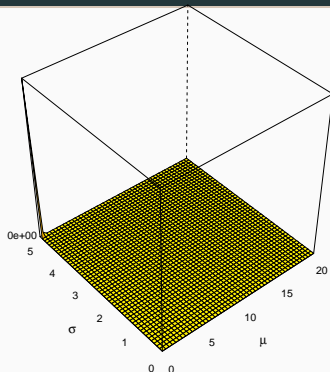
Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$

Check https://twitter.com/i/status/1447831352217415680

When spun on edge 250 times, a Belgian 1€ coin came up heads 140 times and tails 110. It looks very suspicious to me. If the coin were unbiased, the chance of getting a result as extreme as that would be less than 7%.

– From "The Gardian" quoted by MacKay
in Information Theory, Inference and Learning Algorithms

- Model: $Y \sim \mathcal{B}(\pi)$
- Data: $y = 1, 0, 1, 1, 0, 0, 1, 1, 1, \ldots$
- Prior: $\pi \sim \mathcal{T}(0, 1)$

$$p(\pi|y) = \frac{p(y|\pi) \cdot p(\pi)}{p(y)} = \frac{(1-\pi)^{n_0} \pi^{n_1} \cdot (2 - 4|\pi - 0.5|)}{\text{some normalization}}$$

Check https://twitter.com/i/status/1447831352217415680

A Simple Gaussian Model

- Model: $Y \sim \mathcal{N}(\mu, \sigma)$
- Prior: $\mu \sim \mathcal{U}(0, 20)$ and $\sigma \sim \mathcal{U}(0, 5)$

- Model: $Y \sim \mathcal{N}(\mu, \sigma)$
- Prior: $\mu \sim \mathcal{U}(0, 20)$ and $\sigma \sim \mathcal{U}(0, 5)$


```r
set.seed(162);
n=20; mu=12.5; sigma=1.6;
Y=rnorm(n, mean=mu, sd=sigma);
Y
```

```
 [1] 13.899247 12.951346 12.164091 10.869858 13.075777 12.552552 15.446823
 [8] 11.920264 12.849875  9.367122 12.083848 13.852930 12.740590  9.674321
[15] 11.489182 12.195024 13.946985  9.220992 11.821921  9.347013
```

Model: $Y \sim \mathcal{N}(\mu, \sigma)$, hence $p(y|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{y-\mu}{2}\right)^2\right)$

Therefore $p(\mu, \sigma|y) \propto \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{y_i - \mu}{2}\right)^2\right) \cdot \frac{1}{100}$
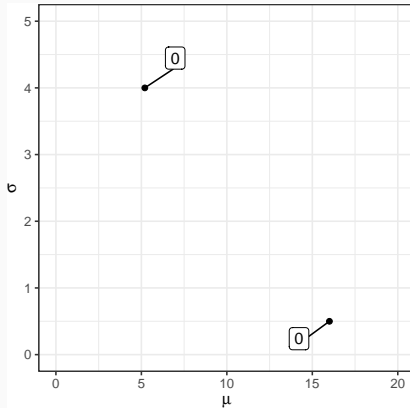
```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```
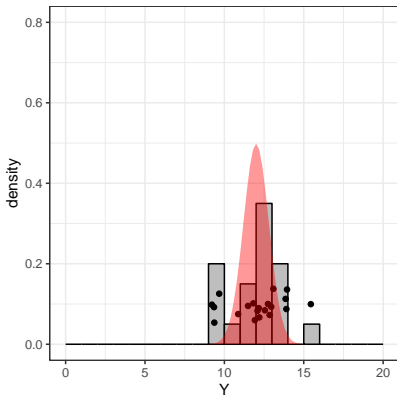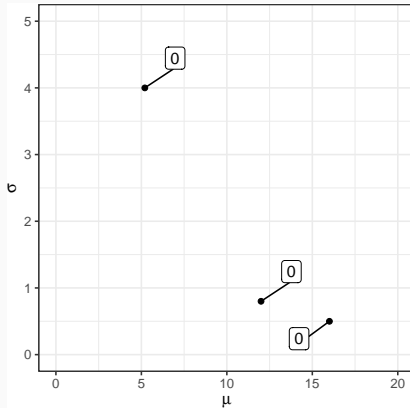
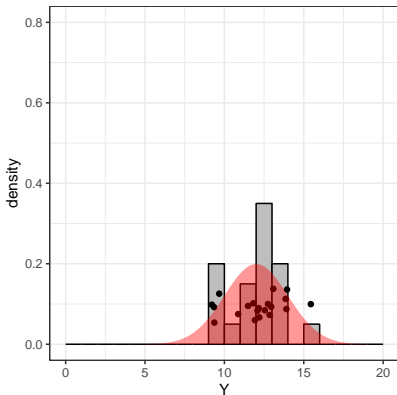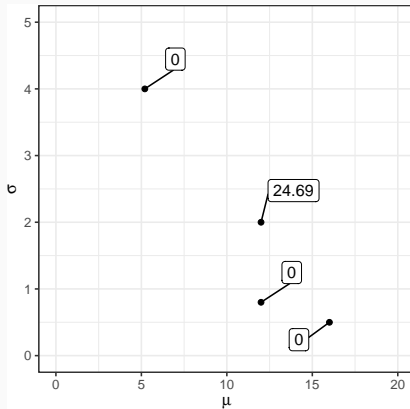Distribution of observations Y Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```


Distribution of observations Y

Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```


Distribution of observations Y

Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```


Distribution of observations *Y*

Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```
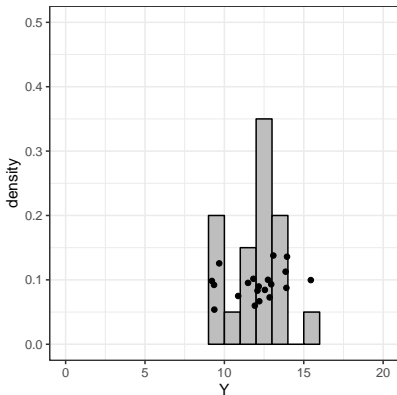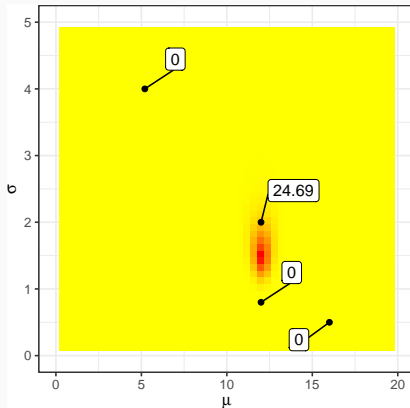

Distribution of observations Y

Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```


Distribution of observations Y

Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```


Distribution of observations Y

Posterior distribution

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```
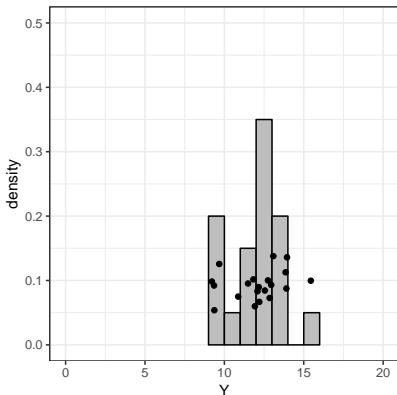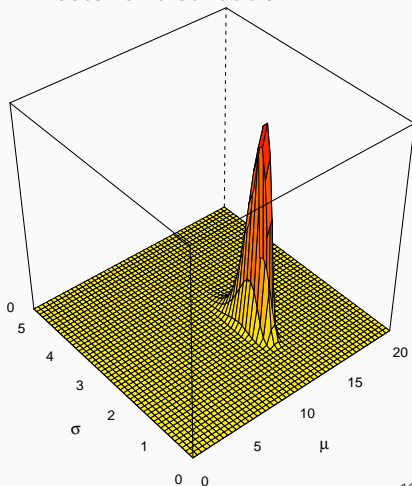

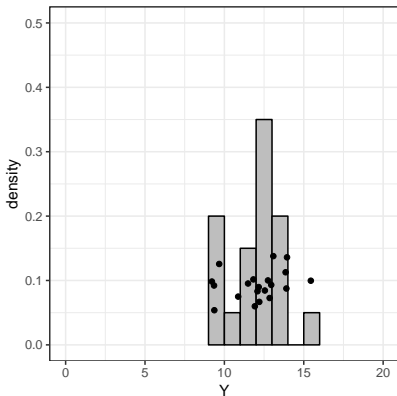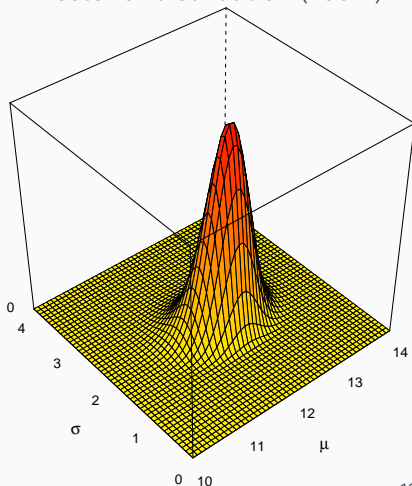Distribution of observations Y

Posterior distribution (Zoom)

Single point estimate (Normal model)

```
[1] "Mean: 12.07348806679"
[1] "Standard Deviation: 1.70127707382769"
```
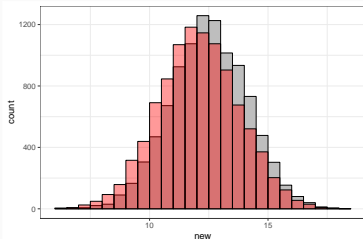
$$p(\mu, \sigma | y) \propto \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{y_i - \mu}{2}\right)^2\right) \cdot \frac{1}{100}$$

- *Machine Learning*: Maximum Likelihood |y
  - $\mu_{MLE} = \frac{1}{n}\sum_{i=1}^{n} y_i$
  - $\sigma_{MLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \mu_{MLE})^2}$
- *Frequentist*: ensure $\mathbb{E}[\mu_F] = \mu$ and $\mathbb{E}[\sigma_F^2] = \sigma^2$
  - $\mu_F = \frac{1}{n}\sum_{i=1}^{n} y_i$
  - $\sigma_F = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \mu_F)^2}$
- *Bayesian*: sample the posterior

- θ: unknown parameter ($\mu = 12.5$, $\sigma = 1.6$)
- y: observation
- $\hat{\theta}$: single point estimate of θ ($\mu \approx 12.07$, $\sigma \approx 1.7$)
- $\tilde{y}$: future observations

Generating $\tilde{y}$ from $\hat{\theta}$

(does not account for the
uncertainty on $\hat{\theta}$)

- θ: unknown parameter ($\mu = $ `12.5`, $\sigma = $ `1.6`)
- y: observation
- $\hat{\theta}$: single point estimate of θ ($\mu \approx$ `12.07`, $\sigma \approx$ `1.7`)
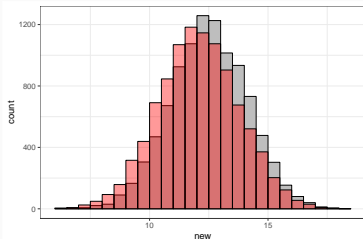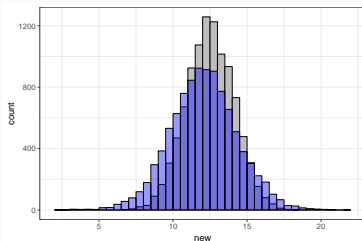- $\tilde{y}$: future observations

Generating $\tilde{y}$ from $\hat{\theta}$

Generating $\tilde{y}$ from many $\tilde{\theta}|y$

(does not account for the
uncertainty on $\hat{\theta}$)

Noise on y + uncertainty on θ

Take away messages:

1. With enough data, reasonable people converge.
2. If any $p(\theta) = 0$, no data will change that
   - Sometimes imposing $p(\theta) = 0$ is nice (e.g., $\theta > 0$)
3. An uninformative prior is better than a wrong highly (supposedly) informative prior.
4. With conjugate priors, calculus of the likelihood is possible
   Otherwise, the normalization is a huge pain

Computing confidence intervals, high density regions, expectation of complex functions... Samples are easier to use than distributions.

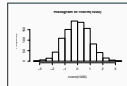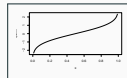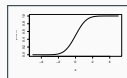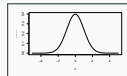BUGS: Bayesian inference Using Gibbs Sampling

$$\underbrace{p(\theta|y,x)}_{\text{Posterior}} \propto \underbrace{p(y|\theta,x)}_{\text{Likelihood}} \underbrace{p(\theta,x)}_{\text{Prior}}$$

Bayesian Sampling

Generating random number: direct method

- Input:
  - $\mathcal{U}(0, 1)$

  - A target density f_Y

- 3 *Easy* steps:

  1. Compute $F_Y(t) = \int_{-\infty}^{t} f_Y(y).dy$

  2. Compute the inverse F_Y^{-1}

  3. Apply F_Y^{-1} to your uniform numbers
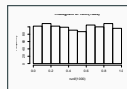
Step 1 is generally quite complicated. The *prior* makes it *even worse*.

Multi-dimensional densities: just as complicated unless the law has a very particular structure

Assume we have M and g, s.t. $p(\theta|y) \leq Mg(\theta)$

- Draw $\theta \sim g$ and accept with probability $\dfrac{p(\theta|y)}{Mg(\theta)}$
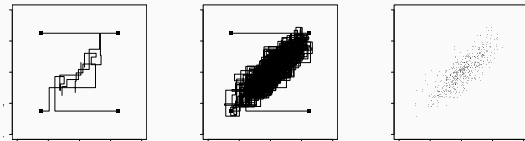
Works well if Mg is a good approximation of $p(.|y)$

Issues:

- p is multiplied by the prior. Where is the max? Which g, which M?
- Is the landscape flat, hilly, spiky?
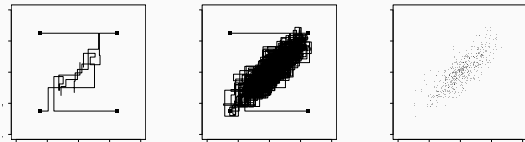- Rejection can be quite inefficient ($\leadsto$ Importance sampling)

Dimension by dimension (Gibbs sampler): $\theta_j^t \sim p(.|\theta_{-j}^{t-1}, y)$

Dimension by dimension (Gibbs sampler): $\theta_j^t \sim p(.|\theta_{-j}^{t-1}, y)$

Metropolis-Hasting: Jumping distribution J

- $\theta^* \sim J(\theta^{t-1})$ $r = \dfrac{p(\theta^*|y)}{p(\theta^{t-1}|y)}$ $\theta^t = \begin{cases} \theta^* & \text{with proba. } \min(r, 1) \\ \theta^{t-1} & \text{otherwise} \end{cases}$

Look for high density areas

- Highly skewed (short/long-tail) or multi-modal are problematic
- Transformation, reparameterization, auxiliary variables, simulated tempering, ...
- Trans-dimensional Markov chains: the dimension of the parameter space can change from one iteration to the next
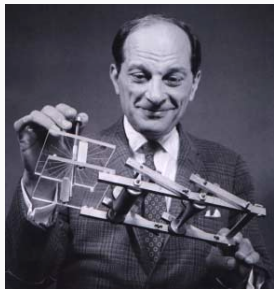
Try to eliminate the random walk inefficiency

- Add a momentum variable ϕ_j for each component θ_j and move to the right direction

Hamiltonian Monte-Carlo combines MCMC with deterministic optimization methods

- Leapfrog: L steps of $\varepsilon/2$ ($L\varepsilon \approx 1$)
- No U-turn Sampler (NUTS): adapt step sizes locally, the trajectory continues until it turns around

A probabilistic programming language implementing full Bayesian statistical inference with MCMC sampling (NUTS, HMC) and penalized maximum likelihood estimation with optimization (L-BFGS)

Stanislaw Ulam, namesake of Stan and co-inventor of Monte Carlo methods shown here holding the Fermiac, Enrico Fermi's physical Monte Carlo simulator for neutron diffusion
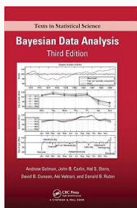
Bayesian Data Analysis, Gelman et al., 2013
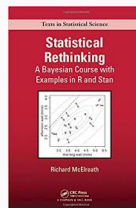
Bayesian Course with examples in R and Stan, Richard McElreath, 2015

```
ggplot(df, aes(x, y))+geom_point(alpha=0.3) + theme_bw()
```
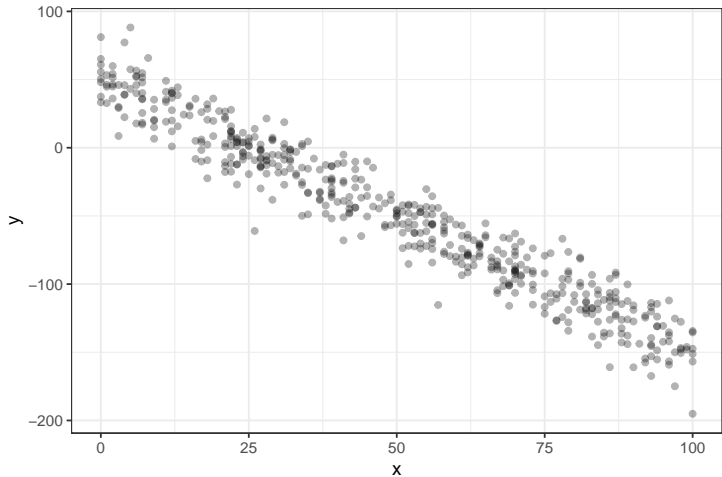
Model $y \sim \mathcal{N}(\alpha x + \beta, \sigma^2)$

Prior

- $\alpha \sim \mathcal{N}(0, 10)$
- $\beta \sim \mathcal{N}(0, 10)$
- $\sigma \sim \mathcal{N}(0, 10)^+$

```
library(rstan)

modelString = "data { // the observations
    int<lower=1> N; // number of points
    vector[N] x;
    vector[N] y;
}
parameters { // what we want to find
    real intercept;
    real coefficient;
    real<lower=0> sigma; // indication: sigma cannot be negative
}
model {
    // We define our priors
    intercept   ~ normal(0, 10); // We know that all the parameters follow a nor
    coefficient ~ normal(0, 10);
    sigma       ~ normal(0, 10);

    // Then, our likelihood function
    y ~ normal(coefficient*x + intercept, sigma);
}
"
sm = stan_model(model_code = modelString)
```

```
1  data = list(N=nrow(df),x=df$x,y=df$y)
2  fit = sampling(sm,data=data, iter=500, chains=8)
```

```
SAMPLING FOR MODEL 'ea4b5a288cf5f1d87215860103a9026e' NOW (CHAIN 1).
Chain 1: Gradient evaluation took 7.6e-05 seconds
Chain 1: 1000 transitions using 10 leapfrog steps per transition would take 0.76
Chain 1: Iteration:   1 / 500 [  0%]  (Warmup)
Chain 1: Iteration:  50 / 500 [ 10%]  (Warmup)
Chain 1: Iteration: 100 / 500 [ 20%]  (Warmup)
Chain 1: Iteration: 150 / 500 [ 30%]  (Warmup)
Chain 1: Iteration: 200 / 500 [ 40%]  (Warmup)
Chain 1: Iteration: 250 / 500 [ 50%]  (Warmup)
Chain 1: Iteration: 251 / 500 [ 50%]  (Sampling)
Chain 1: Iteration: 300 / 500 [ 60%]  (Sampling)
Chain 1: Iteration: 350 / 500 [ 70%]  (Sampling)
Chain 1: Iteration: 400 / 500 [ 80%]  (Sampling)
Chain 1: Iteration: 450 / 500 [ 90%]  (Sampling)
Chain 1: Iteration: 500 / 500 [100%]  (Sampling)
Chain 1:  Elapsed Time: 0.101632 seconds (Warm-up)
Chain 1:                0.044023 seconds (Sampling)
Chain 1:                0.145655 seconds (Total)

SAMPLING FOR MODEL 'ea4b5a288cf5f1d87215860103a9026e' NOW (CHAIN 2).
Chain 2: Gradient evaluation took 2e-05 seconds
```

```
1  print(fit)
```

```
              mean se_mean   sd     2.5%      25%      50%      75%    97.5%
intercept    49.12    0.04 1.31    46.53    48.24    49.13    50.00    51.68
coefficient  -1.96    0.00 0.02    -2.01    -1.98    -1.96    -1.95    -
1.92
sigma        15.48    0.01 0.48    14.56    15.15    15.47    15.79    16.44
lp__      -1630.71    0.04 1.14 -1633.61 -1631.32 -1630.42 -1629.85 -1629.36
           n_eff Rhat
intercept    997 1.00
coefficient  979 1.00
sigma       1057 1.00
lp__         840 1.01

Samples were drawn using NUTS(diag_e) at Wed May 22 22:30:52 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).
```
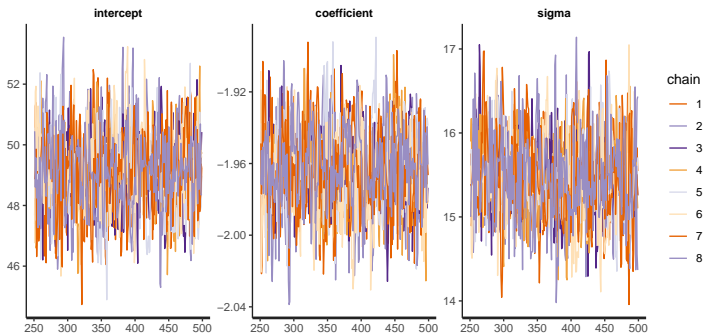
```
1  stan_trace(fit)
```
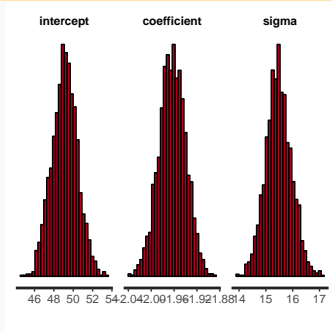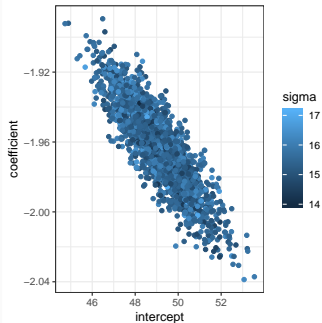
This allows to define credibility regions (or intervals).

A catch on model selection

What's a good model ? A model with a small prediction error...

- Adding parameters in a linear regression always improve the Residual Standard Error, hence the R^2.
- Yet we would like to have few parameters (parsimony, Occam's razor)

How do we distinguish "true" parameters from "false" ones ?

Intuitively:

- Non-significant β should go to 0
- The RSE should be penalized by the number of parameters

Option 1:

Let's consider several alternative models $M_1, M_2, \ldots$

BIC $(M) = k \ln(n) - 2 \ln(\widehat{L(M)})$, where

- $\hat{L}$ is the maximized value of the likelihood function
- n is the number of observations
- k is the number of parameters

Let's consider several alternative models $M_1, M_2, \ldots$

BIC $(M) = k \ln(n) - 2 \ln(\widehat{L(M)})$, where

- $\hat{L}$ is the maximized value of the likelihood function
- n is the number of observations
- k is the number of parameters

Bayesian argument:

- Uniform prior over alternative models
- When n is large the BIC is proportional to $-\log(p(M_i|Y))$
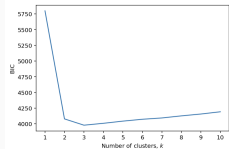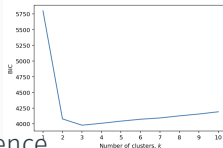 - Choose the model with the smaller BIC!!

Let's consider several alternative models $M_1, M_2, \ldots$

BIC $(M) = k \ln(n) - 2 \ln(\widehat{L(M)})$, where

- $\hat{L}$ is the maximized value of the likelihood function
- n is the number of observations
- k is the number of parameters

Bayesian argument:

- Uniform prior over alternative models
- When n is large the BIC is proportional to $- \log(p(M_i|Y))$
  - Choose the model with the smaller BIC!!

AIC $(M) = 2k - 2 \ln(\hat{L})$

- Based on information theory and KL divergence
- Asymptotic too

Option 2:

Wait! If I have $X_1, ..., X_k$ parameters, there are 2^k models.

- Heuristic 1: add parameters one after the other
- Heuristic 2: peel the model

OPTION 2:

Wait! If I have $X_1, ..., X_k$ parameters, there are 2^k models.

- Heuristic 1: add parameters one after the other
- Heuristic 2: peel the model

When we just don't know which parameters should be kept, an other option would be to penalize large coefficients

Lasso Min. $\sum_i (\beta.x_i - y_i)^2 + \lambda \sum_k |\beta_k|$ **Ridge** Min. $\sum_i (\beta.x_i - y_i)^2 + \lambda \sum_k \beta_k^2$

Wait! If I have $X_1, ..., X_k$ parameters, there are 2^k models.

- Heuristic 1: add parameters one after the other
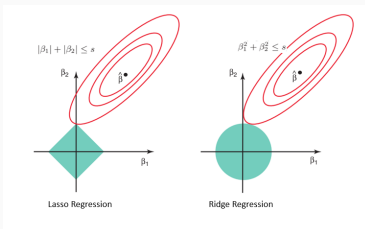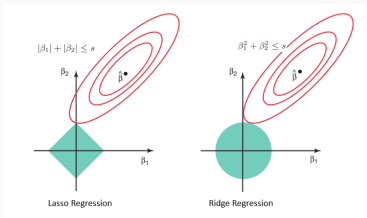- Heuristic 2: peel the model

When we just don't know which parameters should be kept, an other option would be to penalize large coefficients

Lasso Min. $\sum_i (\beta.x_i - y_i)^2 + \lambda \sum_k |\beta_k|$
  Exponential prior with parameter λ for β

Ridge Min. $\sum_i (\beta.x_i - y_i)^2 + \lambda \sum_k \beta_k^2$
  Gaussian prior with variance $1/\lambda$ for β

Lasso Regression   Ridge Regression

Standard linear regression can be seen as a uniform (improper) prior

Wrap-up

Where Bayesian sampling fails:

- ~~Cover the space~~ (e.g., high dimensions)
- ~~Uninformed far away density spikes~~ (mixtures requires informative models and priors)
- ~~High quantiles/rare events~~

Informative priors and starting points are difficult to come up with.

- Much more expensive than "simple" Likelihood optimization, which is also why machine learning techniques are so popular

Where it helps:

- Captures "correlations"
- Robust expectation estimation (1 simulation = very biased)
- Exploit all your knowledge about the system
- Uncertainty quantification with Monte Carlo