Probabilistic Programming and Bayesian Computing

with PyMC

SDSS 2024

Probabilistic Programming

Stochastic language "primitives"

Distribution over values:

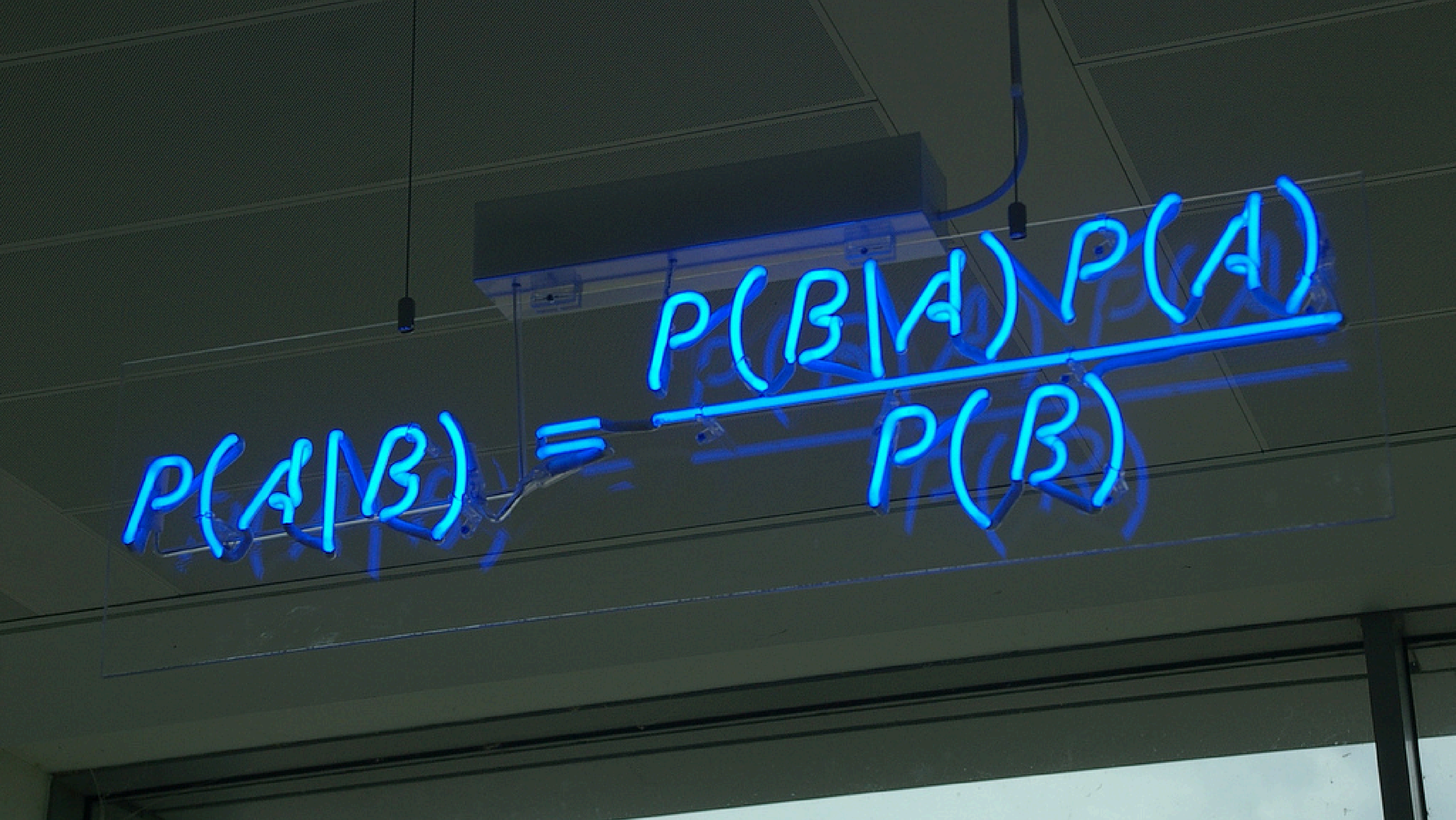
```
X ~ Normal(μ, σ)
x = X.random(n=100)
```

Distribution over functions:

```
Y ~ GaussianProcess(mean_func(x), cov_func(x))
y = Y.predict(x2)
```

Conditioning:

```
p ~ Beta(1, 1)
z ~ Bernoulli(p) # z|p
```



P(A|B) = P(B|A)P(A) / P(B)

What is Bayes?

> Practical methods for making inferences from data using probability models for quantities we observe and about which we wish to learn. – *Gelman et al. 2013*

Why Bayes?

> The Bayesian approach is attractive because it is **useful**. Its usefulness derives in large measure from its **simplicity**. Its simplicity allows the investigation of **far more complex models** than can be handled by the tools in the classical toolbox. – *Link and Barker 2010*

Statistical Paradigms

Any statistical analysis includes:

1. Parameters
2. Data
3. Models

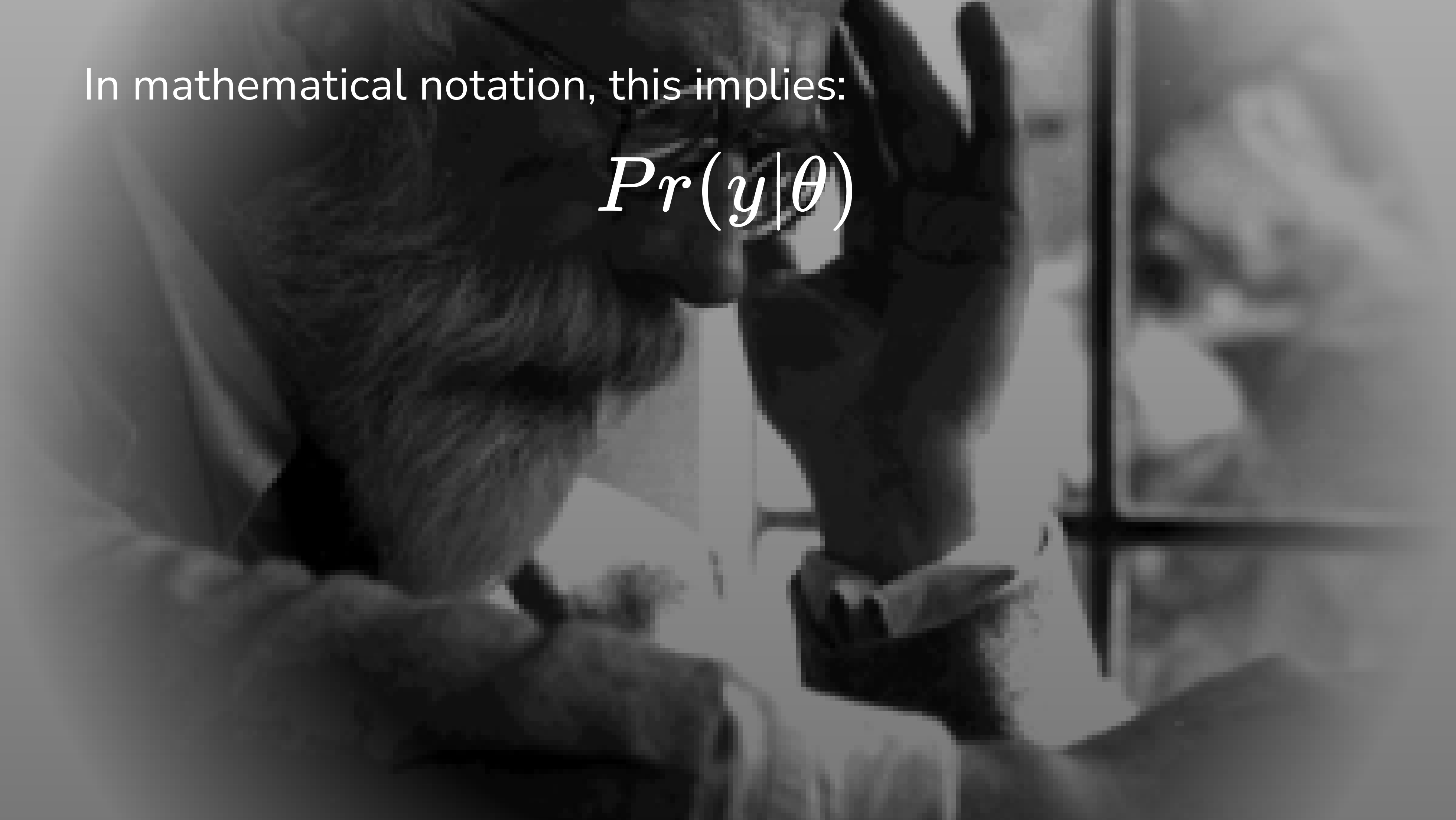
Frequentist Worldview

Data are

random.

Parameters are

fixed.



In mathematical notation, this implies:

$$Pr(y|\theta)$$


Estimators
$\hat{\theta}$

Example

Data have been collected on the prevalence of **autism spectrum disorder (ASD)** in some defined population. Our sample includes n sampled children, y of them having been diagnosed with autism. A frequentist estimator of the prevalence p is:

$$\hat{p} = \frac{y}{n}$$

Why this particular function?

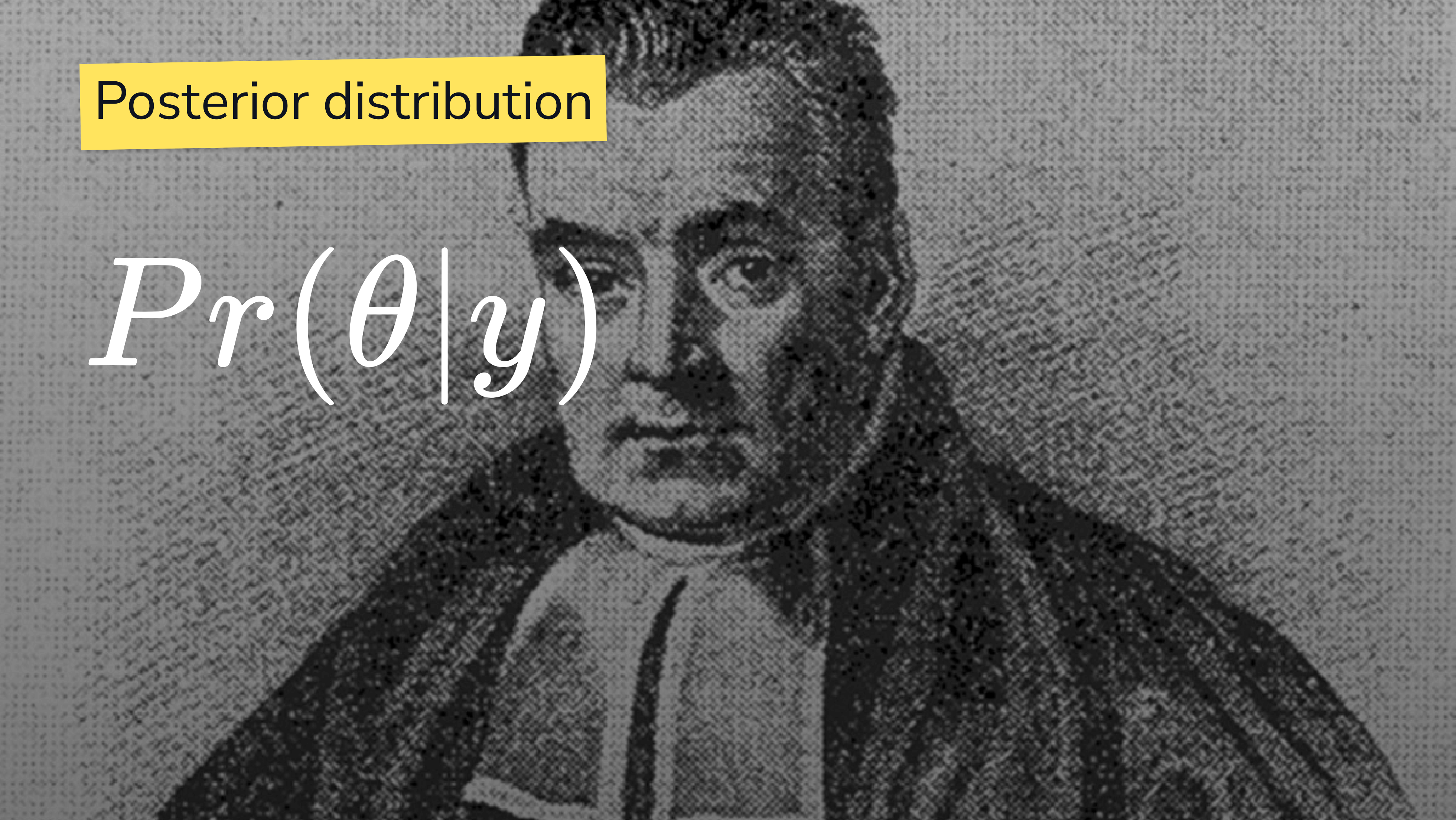
Bayesian Worldview

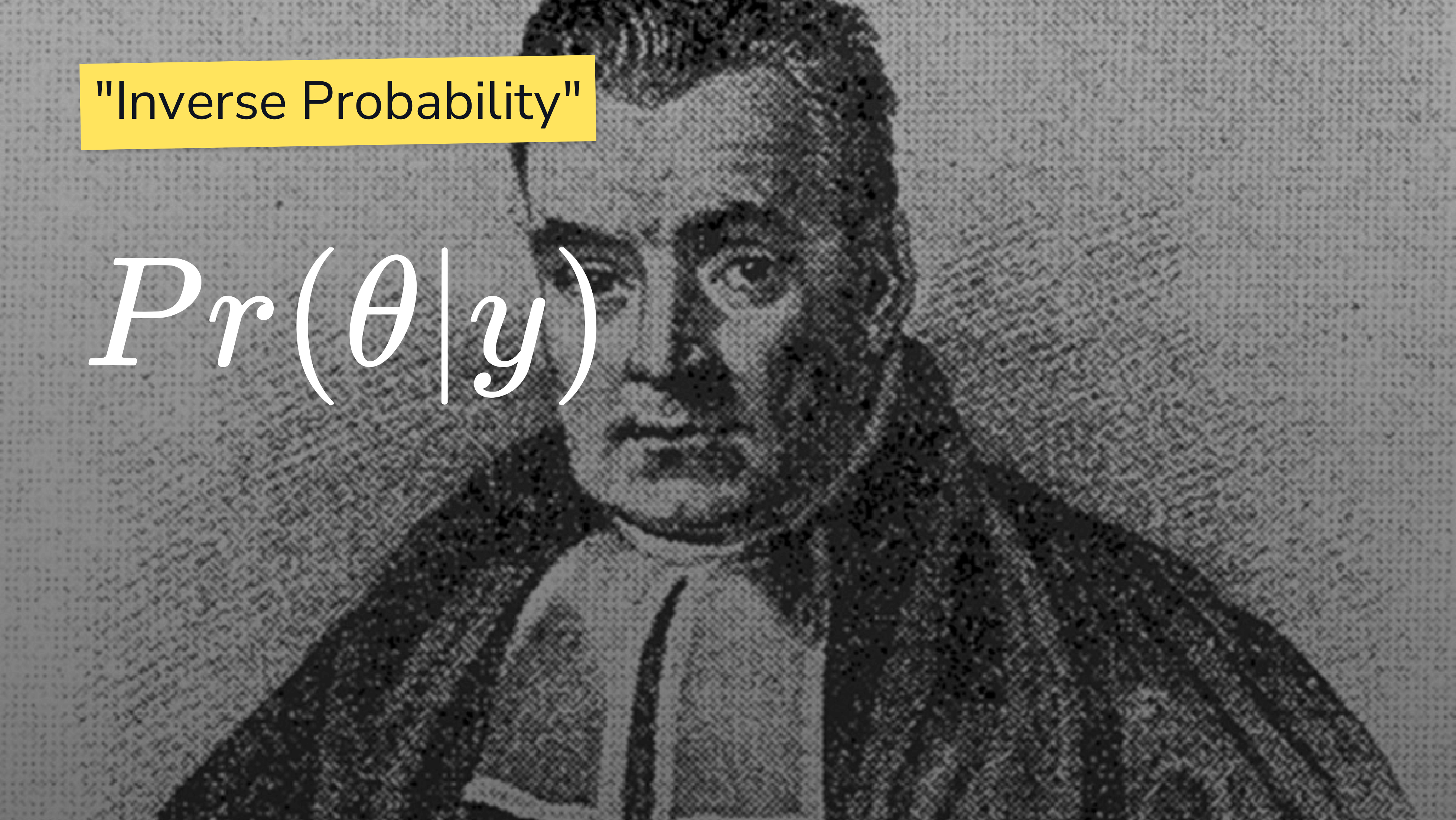
Data are **fixed.**

Parameters are **random.**

Posterior distribution

$$Pr(\theta|y)$$




"Inverse Probability"
$Pr(\theta|y)$

Bayesian Inference

How do we make a probability statement about θ *given y?*

Consider the **joint probability**

$$Pr(\theta, y)$$

Conditional Probability

Let’s break this joint probability into two components:

$$Pr(\theta, y) = Pr(y|\theta)Pr(\theta)$$

Conditional Probability

Similarly, we can use the definition of conditional probability on $Pr(\theta, y)$:

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

Bayes Formula

Substituting our expressions for $Pr(\theta, y)$, we get Bayes' Formula:

$$Pr(\theta|y) = \frac{Pr(y|\theta)Pr(\theta)}{Pr(y)}$$

Posterior Probability

Likelihood of Observations

Prior Probability

$$\Pr(\theta|y) = \frac{\Pr(y|\theta)\Pr(\theta)}{\Pr(y)}$$

Normalizing Constant

Normalization constant

Denominator $Pr(y)$ is the probability of observing the data, **integrated** over all possible values of the parameters.

$$Pr(\theta|y) = \frac{Pr(y|\theta)Pr(\theta)}{\int_\theta Pr(y|\theta)Pr(\theta)d\theta}$$

Infer Values for Latent Variables

Calculate the posterior distribution

$$Pr(\theta|y) \propto Pr(y|\theta)Pr(\theta)$$

Stochastic program

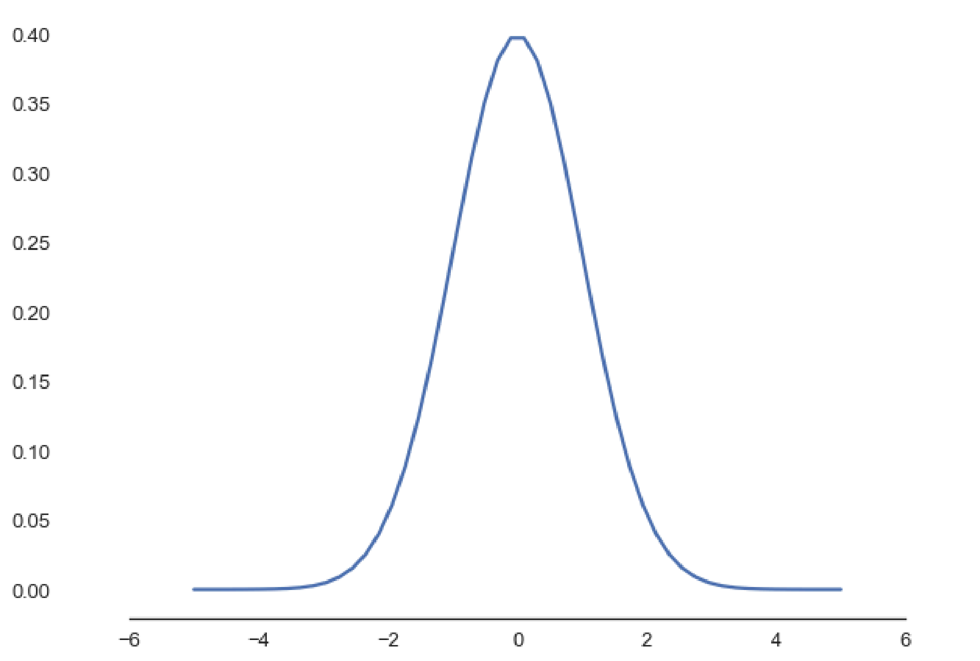
Joint distribution of latent variables and data

$$Pr(\theta, y) = Pr(y|\theta)Pr(\theta)$$

Prior distribution

Quantifies the **uncertainty** in latent variables

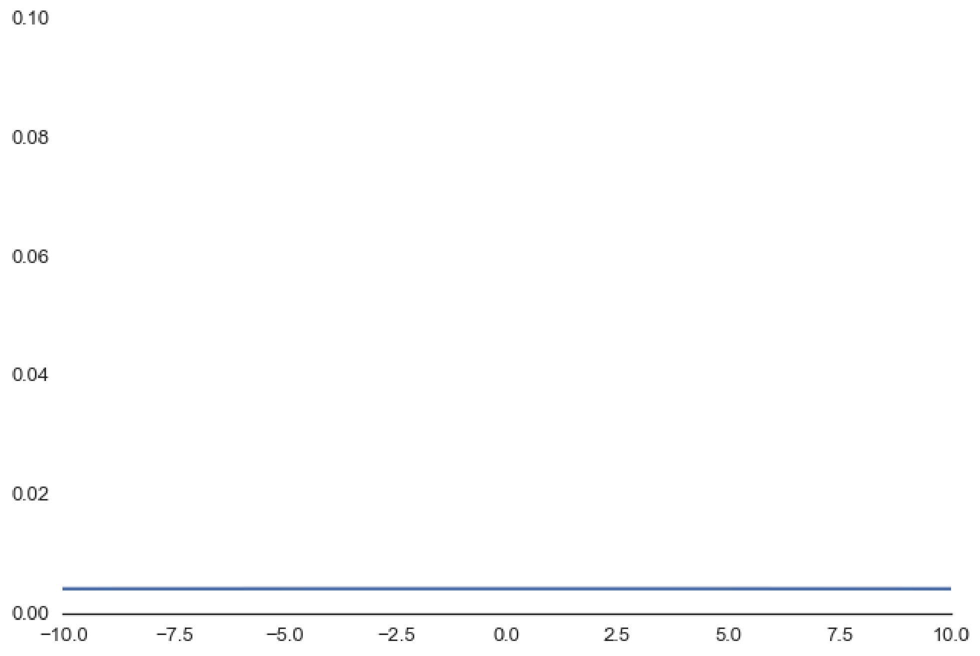
$$\theta \sim \mathrm{Normal}(0, 1)$$



Prior distribution

Quantifies the **uncertainty** in latent variables

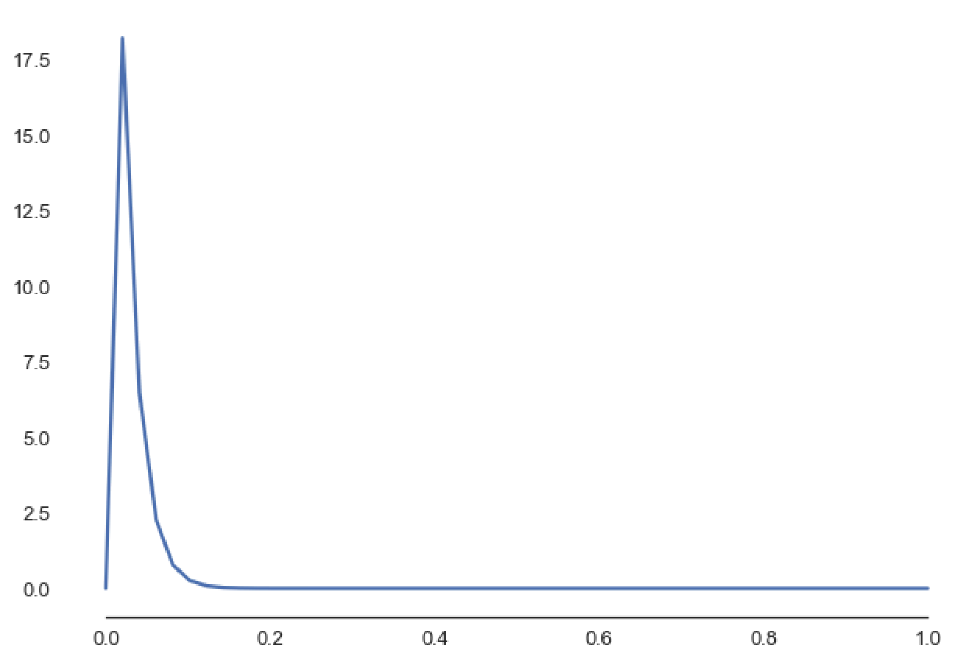
$$\theta \sim \mathrm{Normal}(0, 100)$$



Prior distribution

Quantifies the **uncertainty** in latent variables

$$\theta \sim \mathrm{Beta}(1, 50)$$



Likelihood function

Conditions our model on the observed data

$$Pr(y|\theta)$$

The Likelihood Principle

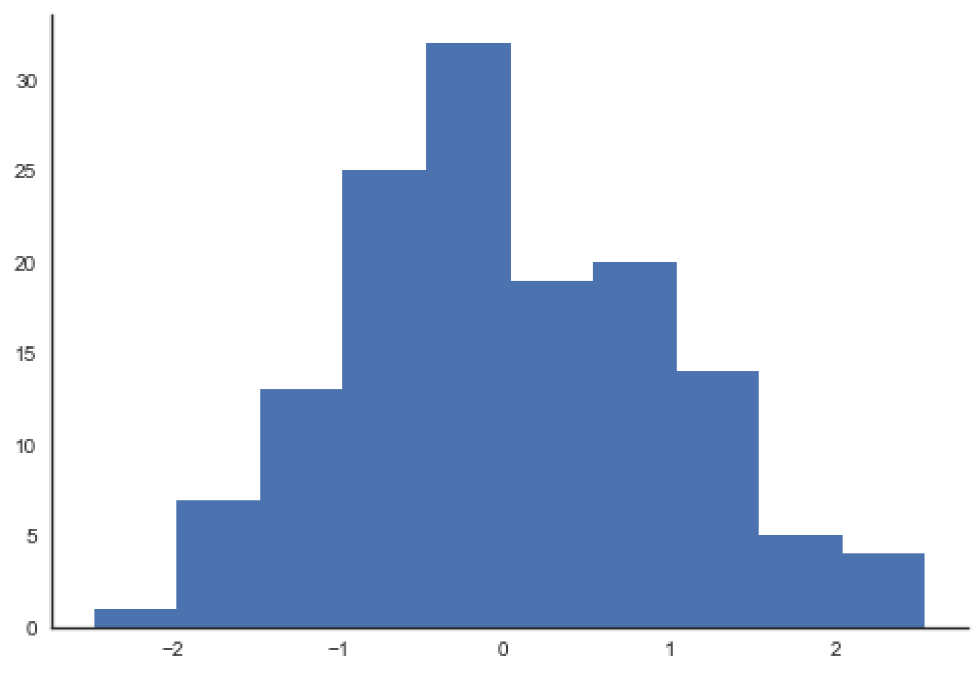
The likelihood function contains all of the information about the data relevant to inference.

$$Pr(\theta|y) \propto Pr(y|\theta)Pr(\theta)$$

Likelihood function

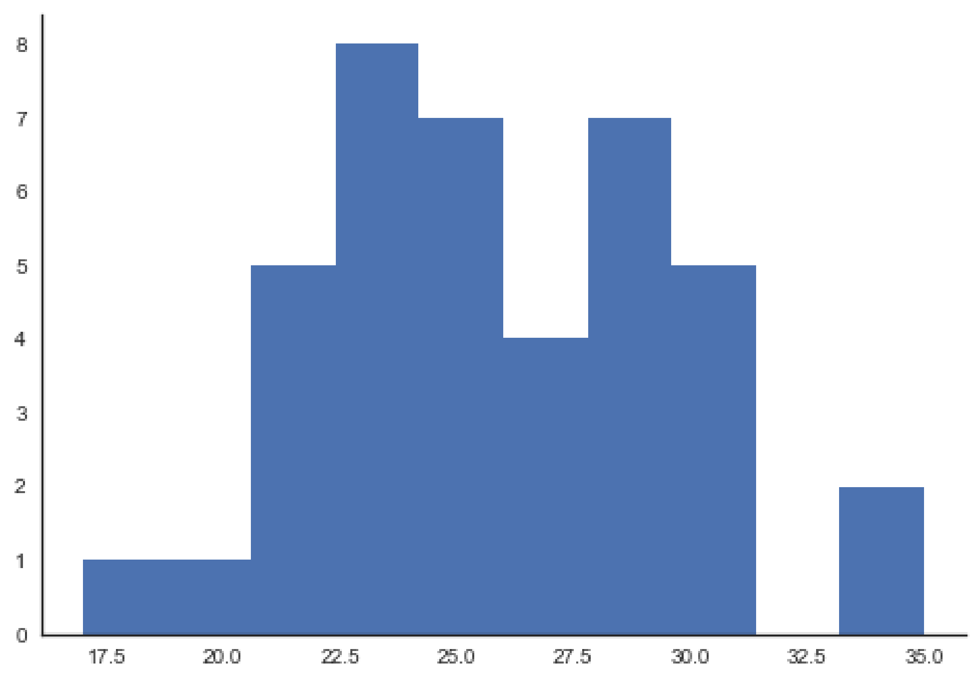
Conditions our model on the observed data

$$x \sim \text{Normal}(\mu, \sigma^2)$$



$$x_K \sim \text{Binomial}(n_{PA}, p_K)$$

Models the distribution of x strikeouts observed from n plate appearances.



Probabilistic programming *abstracts* the inference procedure

Probabilistic Programming in Python

- PyStan
- PyMC
- Pyro/NumPyro
- Tensorflow Probability
- emcee