PHYS465: Statistical Data Analysis in Physics

Week 3: Bayesian Statistics

Dr. Mathew Smith

mat.smith@lancaster.ac.uk

Physics Building; C46

Module Structure

'*model testing*': does our data agree with our model?

Week 11: Fitting a model to data; estimating parameters

Week 12: Hypothesis testing; the likelihood; estimating uncertainties

Week 13: Posterior sampling; Bayesian statistics

'*data driven*': what does our data tell us?

Week 14: Clustering and Classification algorithms

Week 15: Machine learning techniques

Week 13: Previous session

Last week we learnt:

- How to design our experiment through the Null hypothesis
  - This requires us to define a measure of confidence for acceptance
- We can measure the probability of our model through *the likelihood*
  - This is the probability of our model given the data we obtained
  - In the case of a Gaussian likelihood we can calculate this through the χ^2

$$-2\ln\mathscr{L}(\theta\,|\,x) = \chi^2 + 2\sum_{i=1}^{N}\ln\sigma_i + N\ln(2\pi)$$

  - Best-fit parameter values correspond to where the χ^2 is minimised
  - Uncertainties on our parameters can be calculated from this PDF

Week 13: Learning aims

Today we will introduce

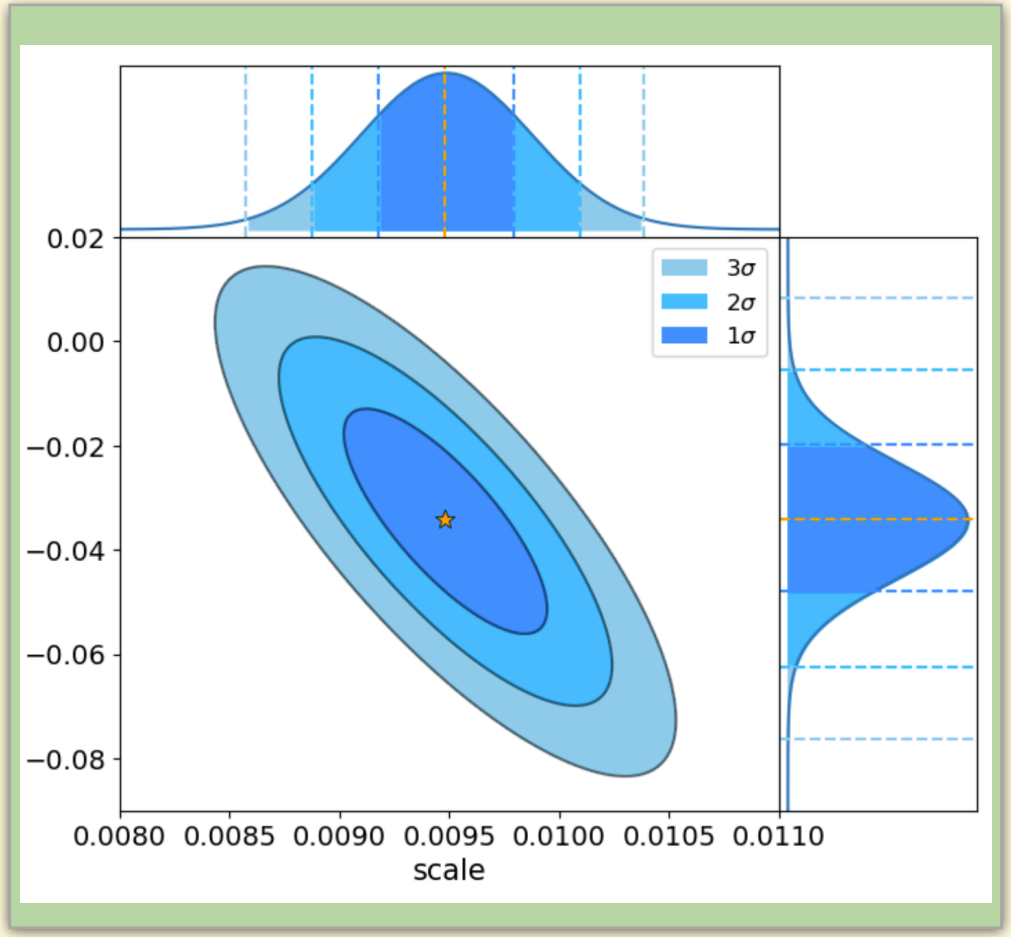
- Bayes theorem and how it relates to parameter estimation
  - The *posterior*, *likelihood*, *prior* and *evidence*
- The difference between Bayesian v Frequentist
- Sampling methods:
  - Markov Chain Monte Carlo (MCMC)
- Practical tips for smooth sampling
- Estimating distributions from MCMC

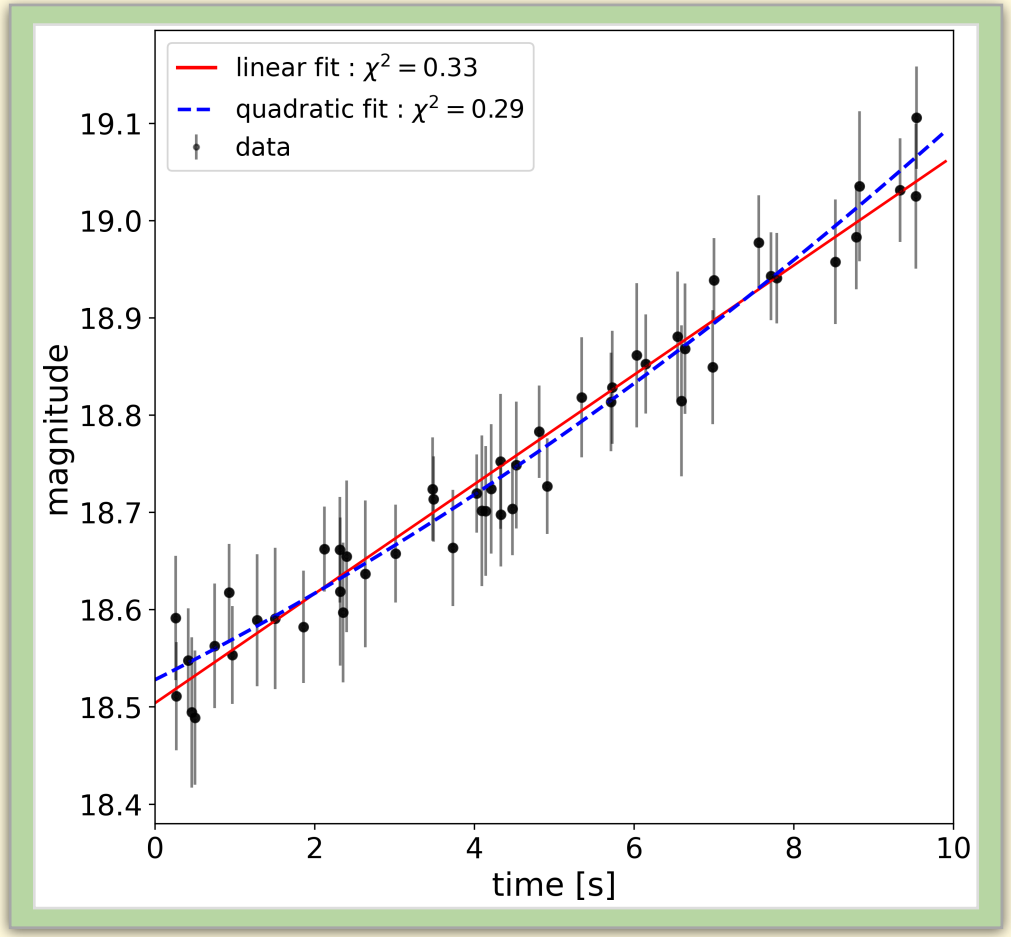
Practical examples on Friday!

Recall: statistical inference

1. Given two models M_1, M_2 how do we know which one is best?

2. Given a parameter P what is the probability that the true value lies in a given range?





Recall: Conditional Probabilities

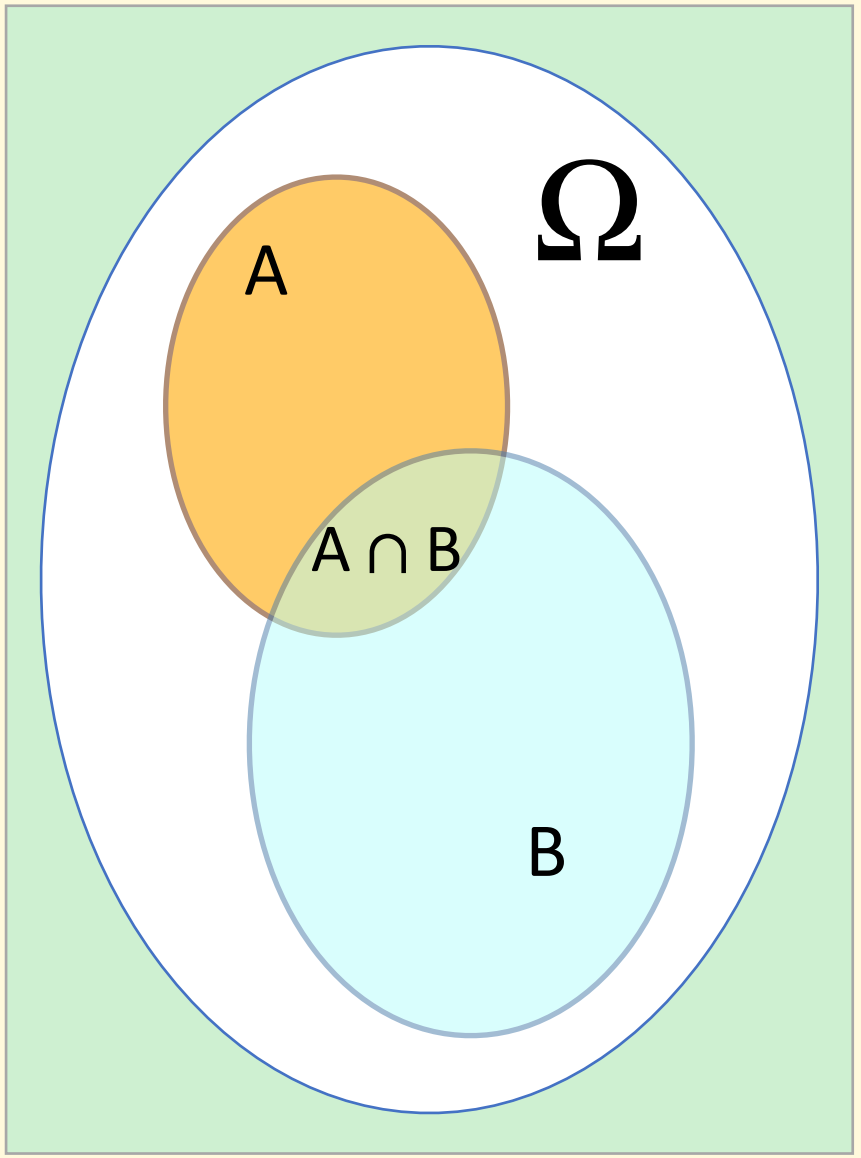
$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

The probability of A given B (an event or condition)

when θ is a model, or parameter

$P(A \mid \theta)$ is called the ***likelihood:*** $\mathcal{L}(\theta \mid A)$

"the ***likelihood*** of obtaining the data 'A' given the model"



Thomas Bayes *1701-1761*

Bayes Theorem

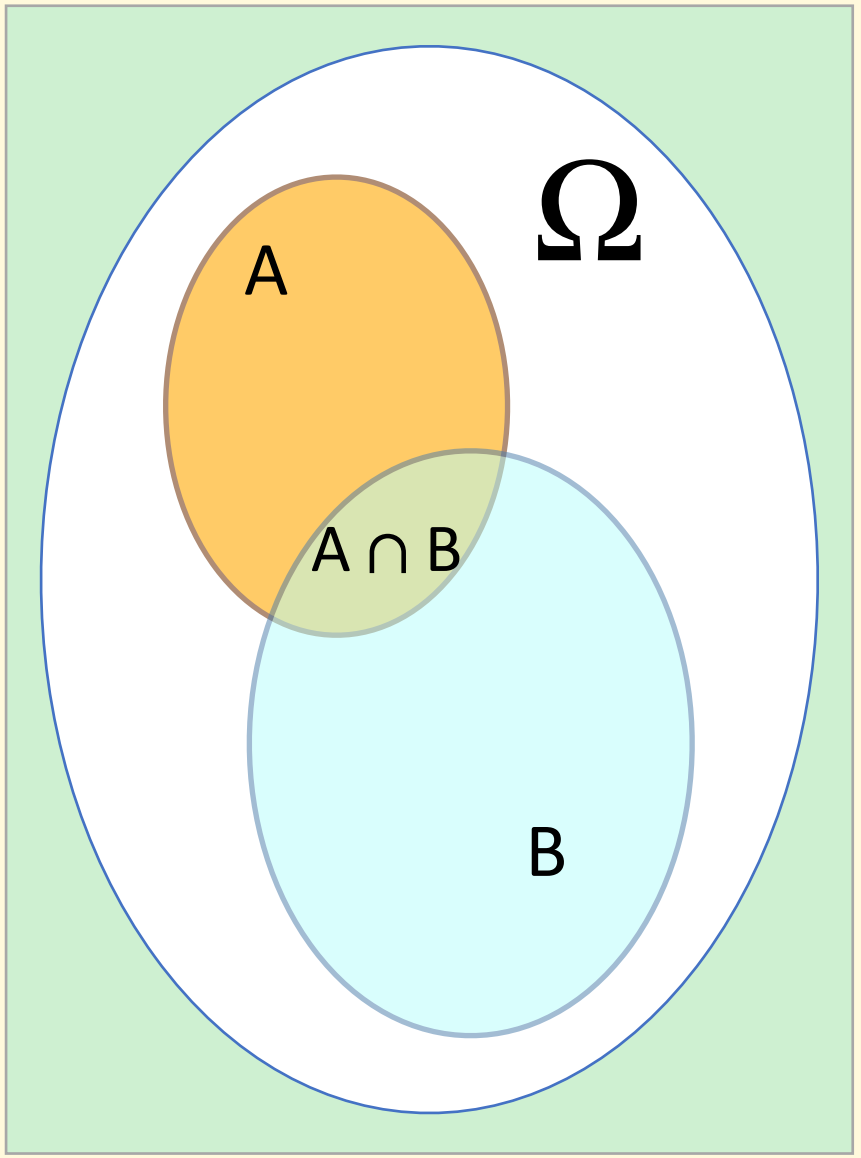
Under the assumption that:

$$P(A \cap B) = P(B \cap A)$$

$$P(A \mid B)\, P(B) = P(B \mid A)\, P(A)$$

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

The probability of A given B (an event or condition)



Thomas Bayes *1701-1761*

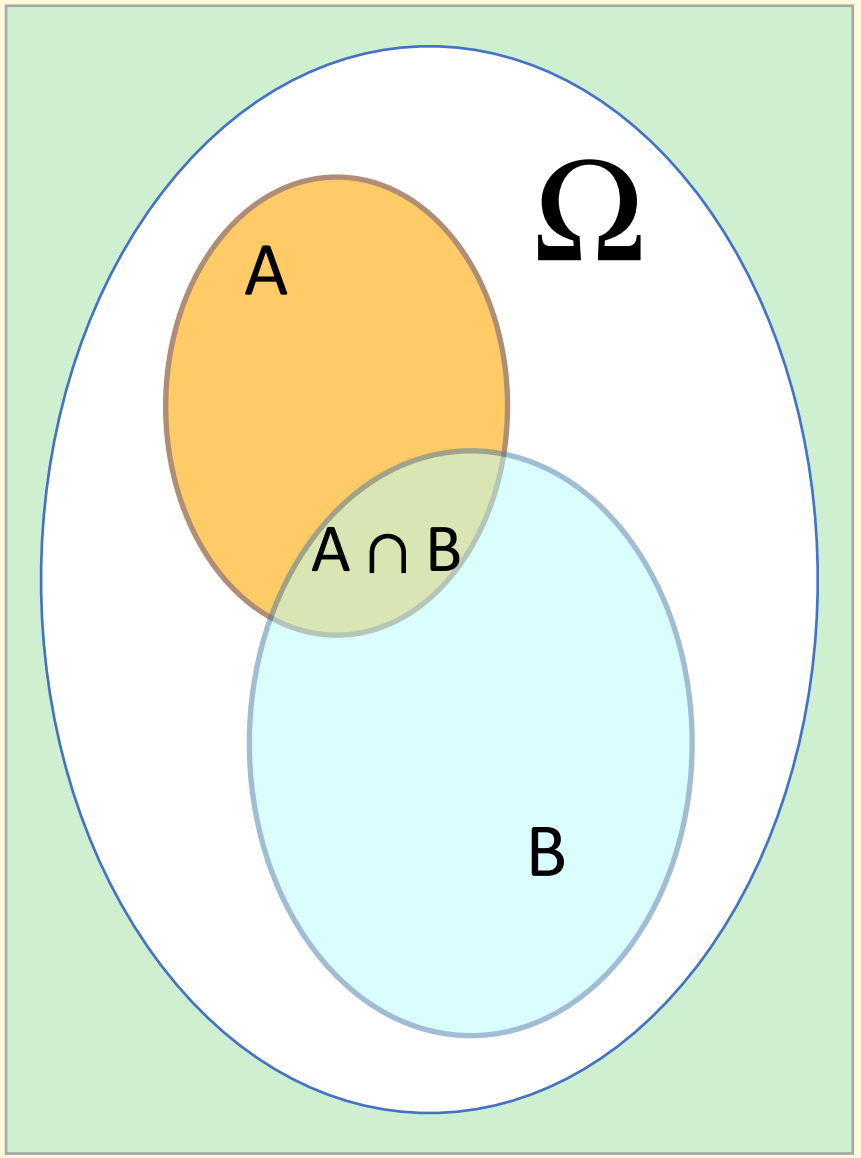
Bayes Theorem

for a specific model θ

And some data D

$$P(\theta \mid D) = \frac{P(D \mid \theta)\, P(\theta)}{P(D)}$$

*The probability of the **model** given our measurements*



Thomas Bayes *1701-1761*

Bayes Theorem

$$P(\theta|D) = \frac{P(D|\theta)\,P(\theta)}{P(D)}$$

*The probability of the **model** given our measurements*

$P(\theta\|D)$	*The posterior*	*The probability of the model (θ) being true given the data*
$P(D\|\theta)$	*The likelihood*	*Given the model (θ) what is the likelihood of getting the data*
$P(\theta)$	*The prior*	*The probability of the model (θ) being true*
$P(D)$	*The evidence*	*The probability of getting the data given all model possibilities*

Likelihoods

Given a set of measurements $x = (x_1, x_2, \dots)$

the ***likelihood:*** $$\mathscr{L}(\theta \,|\, x) = \prod_{i=0}^{N} P(x_i \,|\, \theta)$$

Best-fit value:	The value that maximises $\mathscr{L}$
Plausible values:	Determined by the width of $\mathscr{L}$
Uncertainty:	For a Gaussian distribution 1σ

Log Likelihoods: the Gaussian case

Given a set of measurements $x = (x_1, x_2, \dots)$

$$p(x_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x_i - \mu_i(\theta))^2}{2\sigma_i^2}\right)$$

$$\mathscr{L}(\theta|x) = p(x_1|\theta) * p(x_2|\theta) * \dots$$

$$\mathscr{L}(\theta|x) = e^{-\chi^2/2} * \prod_{i=1}^{N} \frac{1}{\sigma_i} \times (2\pi)^{(-N/2)}$$

$$-2 \ln \mathscr{L}(\theta|x) = \chi^2 + 2 \sum_{i=1}^{N} \ln \sigma_i + N \ln(2\pi)$$

Maximising $\mathscr{L}(\theta|x)$ is equivalent to minimising χ^2

Log Likelihoods: the Gaussian case

Given a set of measurements $x = (x_1, x_2, \dots)$

$$p(x_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x_i - \mu_i(\theta))^2}{2\sigma_i^2}\right)$$

$$\mathscr{L}(\theta \,|\, x) = p(x_1 \,|\, \theta) * p(x_2 \,|\, \theta) * \dots$$

$$-2 \ln \mathscr{L}(\theta \,|\, x) = \chi^2 + 2 \sum_{i=1}^{N} \ln \sigma_i + N \ln(2\pi)$$

Penalty terms: Only needed for normalisation

Maximising $\mathscr{L}(\theta \,|\, x)$ is equivalent to minimising χ^2

An aside: Maximum Likelihood Estimators

Used across Machine Learning

The value of the model parameters that maximises the probability

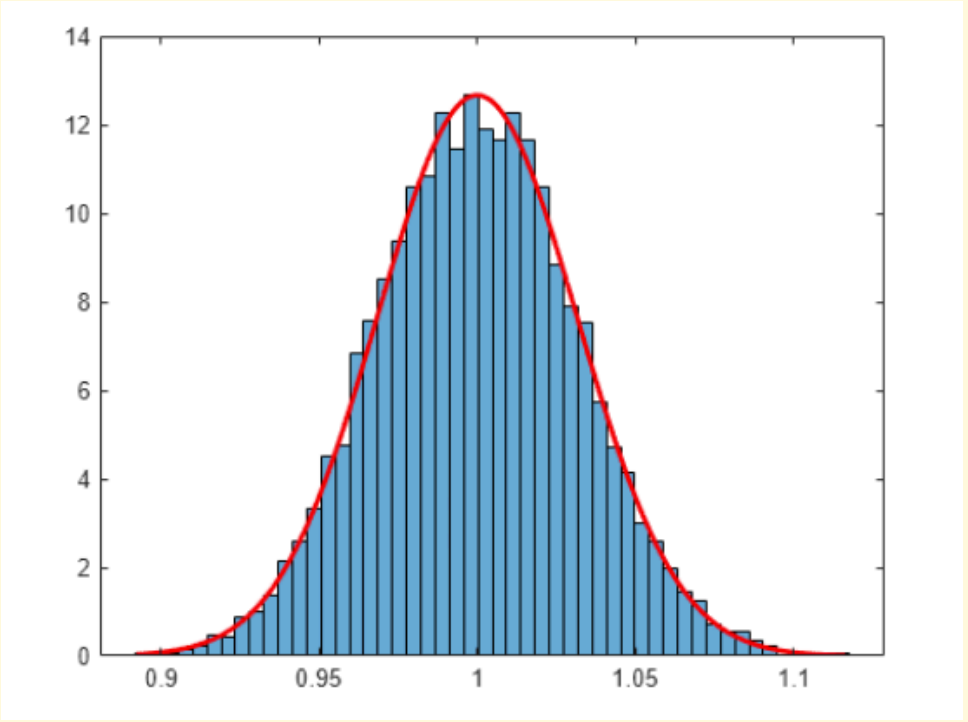
Probability of the data given the model

$$p_{\text{max}} = \max(p_{\text{model}}(x_1, x_2, \ldots, x_N))$$

Probability of the model given the data

$$\mathscr{L}_{\text{max}} = \max(\mathscr{L}(\theta \mid x))$$

Many available algorithms



An aside: Maximum Likelihood Estimators

***Many* available algorithms**

MINUIT ***Used a lot in particle physics*** ***Comparable to curve_fit but faster***

```
# everything in iminuit is done through the Minuit object, so we import it
from iminuit import Minuit

# we also need a cost function to fit and import the LeastSquares function
from iminuit.cost import LeastSquares
def line(x, α, β):
    return α + x * β

least_squares = LeastSquares(data_x, data_y, data_yerr, line)

m = Minuit(least_squares, α=0, β=0)  # starting values for α and β

m.migrad()  # finds minimum of least_squares function
m.hesse()  # accurately computes uncertainties
```

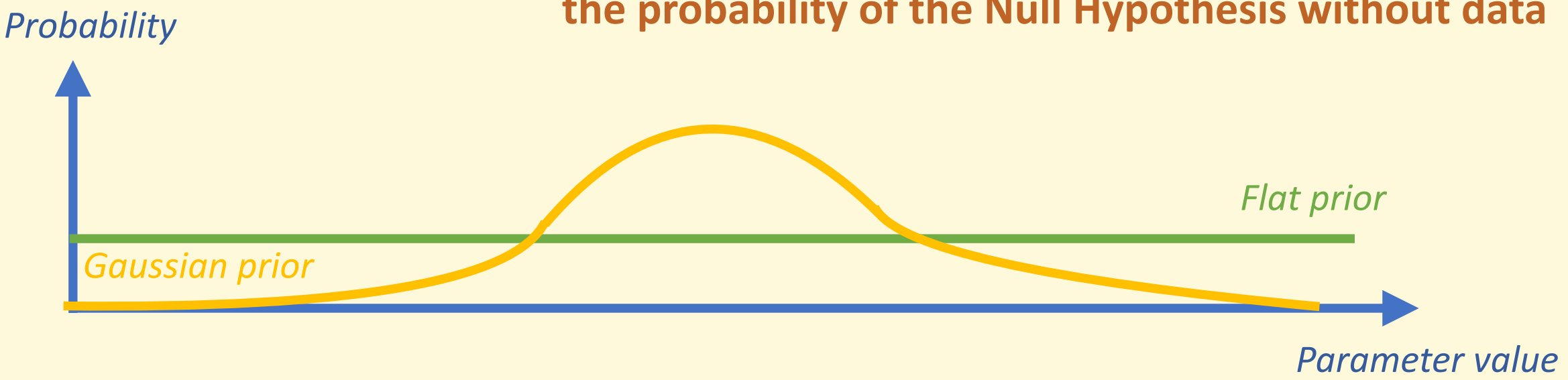
The Prior

Our decision adapts as new data comes in

Our previous experience

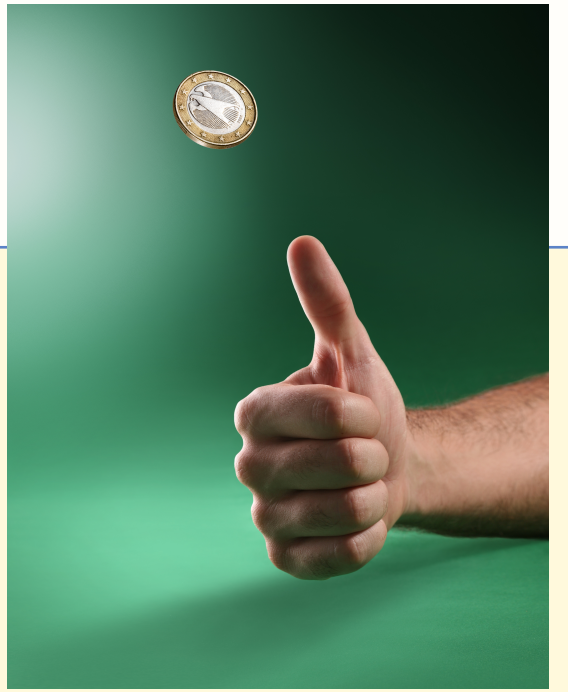
- *An opportunity to state previous information relevant to our experiment*
- **"no assumption" is still an assumption*
- *a feedback loop: past results inform new results*

the probability of the Null Hypothesis without data



Bayesian v Frequentist

- ***Bayesian:*** *Models are not ruled out: they are assigned a probability*
  - *Probability is a degree of belief: it can change with new information*
  - *The data is a fixed outcome*
- ***Frequentist:*** *Attempt to rule out models given the data*
  - *Probabilities are assigned to the data not the model*



Different approaches depending on the experiment

- ***Frequentist :*** *single correct answer P(heads)=0 or 1*
- ***Bayesian:*** *depends on beliefs (prior results)*

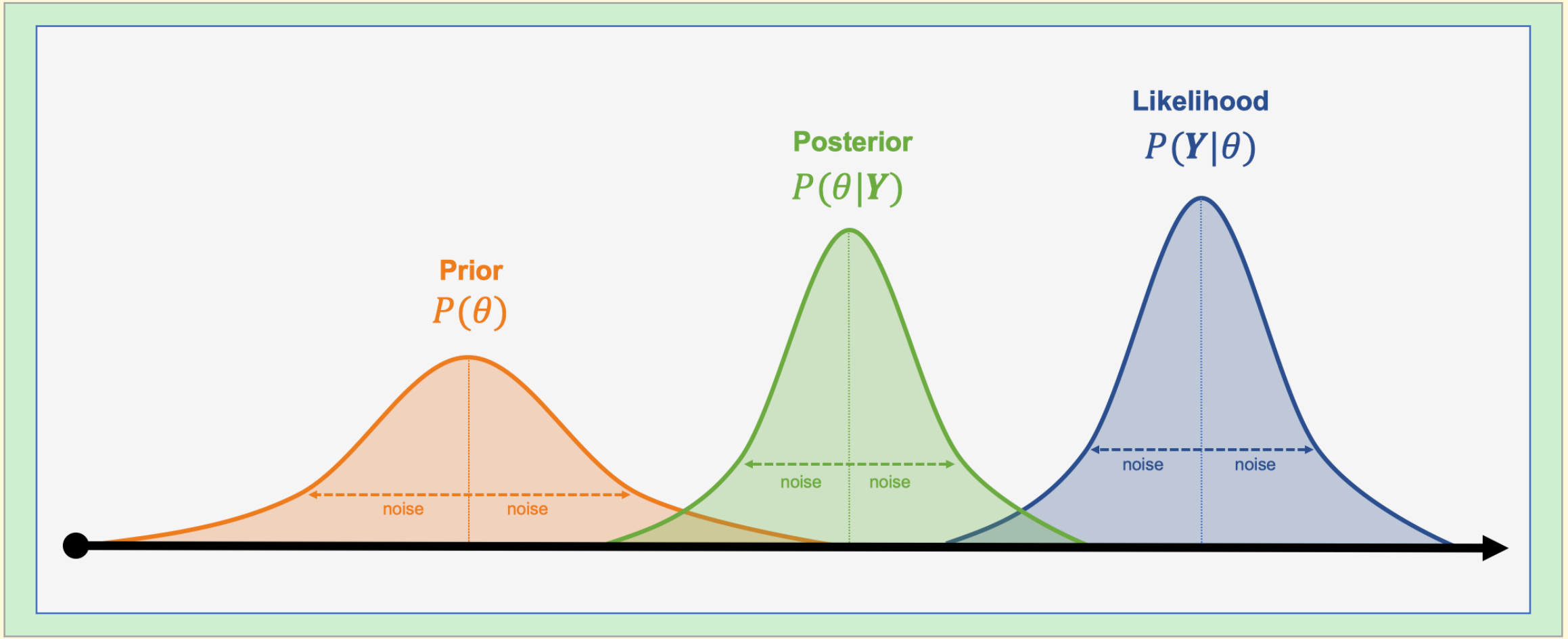
What will the weather be like tomorrow?

** considered difficult to calculate given multiple parameters*

The Posterior

Combines previous results with new data

$$posterior = likelihood \times prior$$



The Posterior

Combines previous results with new data

$$posterior = likelihood \times prior$$

A worked example:

A person's DNA matches that of a sample found at a crime scene.

The chances of a DNA match are 1 in 2 million

What is the probability that the DNA comes from someone else?

Frequentist: 1 in 2 million

The Posterior

Combines previous results with new data

$$posterior = likelihood \times prior$$

A worked example:

What is the probability that the DNA comes from someone else?

Bayesian: Question the model not the data

testing positive is a given measurement ;

>how much more likely does this make her guilty than before?

1 in 2 million is “the likelihood that a random person’s DNA will match”

The Posterior

Combines previous results with new data

$$posterior = likelihood \times prior$$

A worked example:

What is the probability that the DNA comes from someone else?

$$likelihood = \frac{probability\ of\ observation\ if\ guilty}{probability\ of\ observation\ if\ innocent} = 2\ million$$

$$prior = the\ woman\ is\ equally\ likely\ to\ be\ guilty\ as\ anyone\ else$$

=> ***Bayesian:*** How big is the wider population?

The Posterior

Combines previous results with new data

$$posterior = likelihood \times prior$$

A worked example:

What is the probability that the DNA comes from someone else?

if population == 300,000 $\quad prior = \frac{1}{300{,}000}$

$$posterior = likelihood \times prior \quad = 1 \text{ in } 7$$

$$p(guilty|DNA) = 87\,\%$$

Analytic Solution

"conjugate prior"

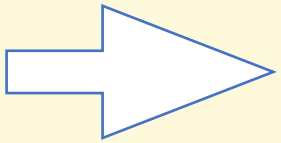
the likelihood and prior have the same functional form

if

$$likelihood \propto gaussian$$

$$prior \propto gaussian$$

The prior is just extra data



$$posterior \propto gaussian$$

Rule of thumb:

n_{data} *large | small* σ *: posterior dominated by likelihood*
n_{data} *small | large* σ *: posterior dominated by prior*

The Evidence

The probability of getting the data given all model possibilities

A single value : no impact on the shape of posterior

=> not required to compare parameter values

but also:

"the probability that the model is a correct given the data"

=> model selection in Bayesian statistics

** difficult to calculate in the presence of noise*

Credible Intervals

Bayesian: *Given the observed data,*

the interval within which a parameter value falls with given probability:

$$\hat{p} - \delta p \leq P_0 \leq \hat{p} + \delta p$$

XX% probability that the true value of P lies between $\hat{p} - \delta p$ ***and*** $\hat{p} + \delta p$

Frequentist: We measure confidence intervals *from* the data:

there's a XX% chance that the true mean falls within this range

Recall: Multiple parameters: *Marginalisation*

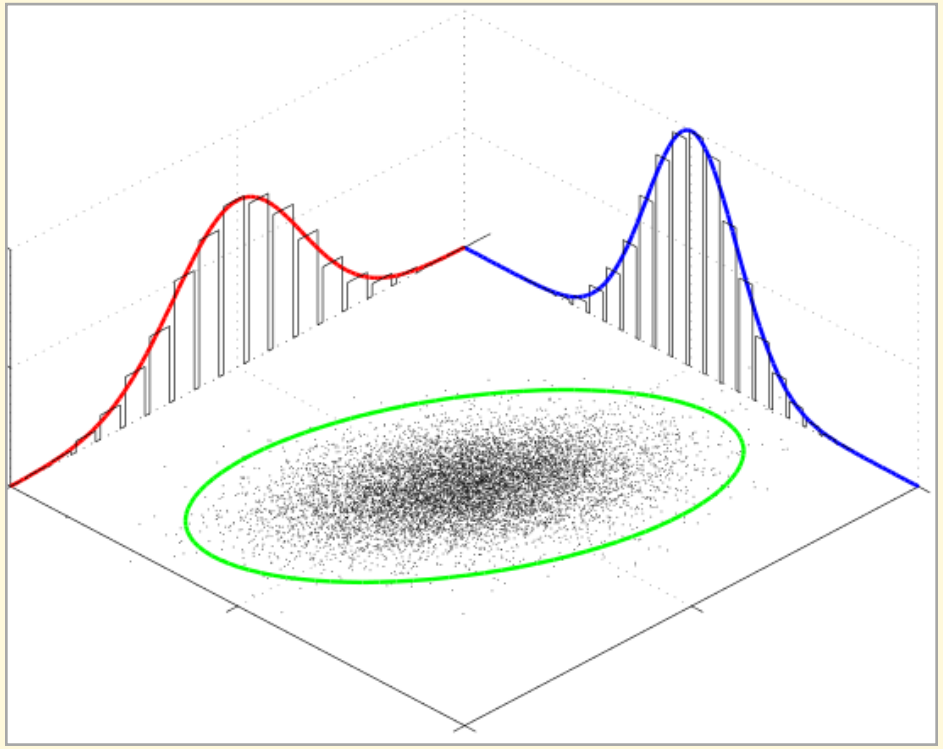
Distributions are often NOT the function of 1 parameter

$$p(x, y)$$

NB: Coursework 1: is 'happiness' a function solely of GDP?

The probability of x is calculated by integrating over y

$$p(x) = \int p(x, y)\, dy$$

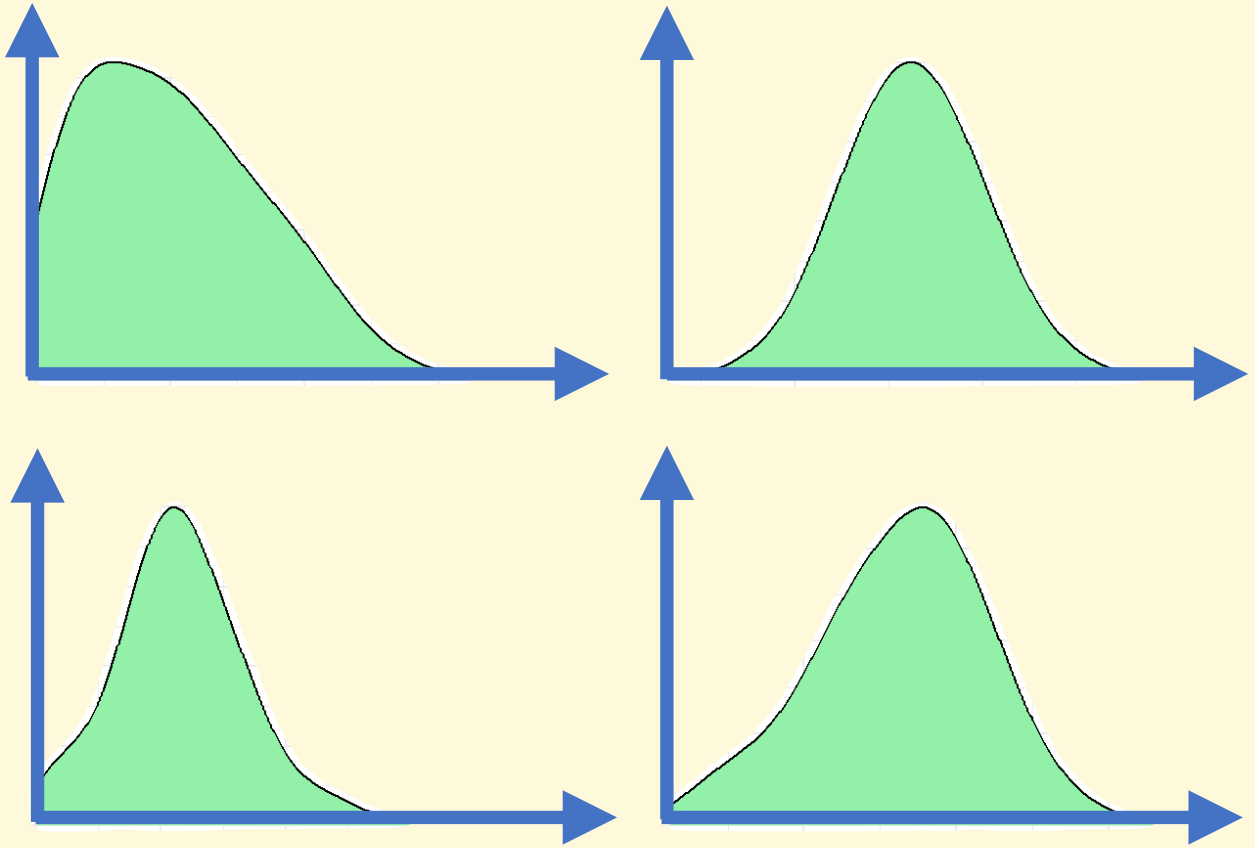


NB: We wanted to measure $p(\textbf{happiness} \mid \textbf{GDP})$, but we measured $p(\textbf{happiness}, \textbf{country} \mid \textbf{GDP})$

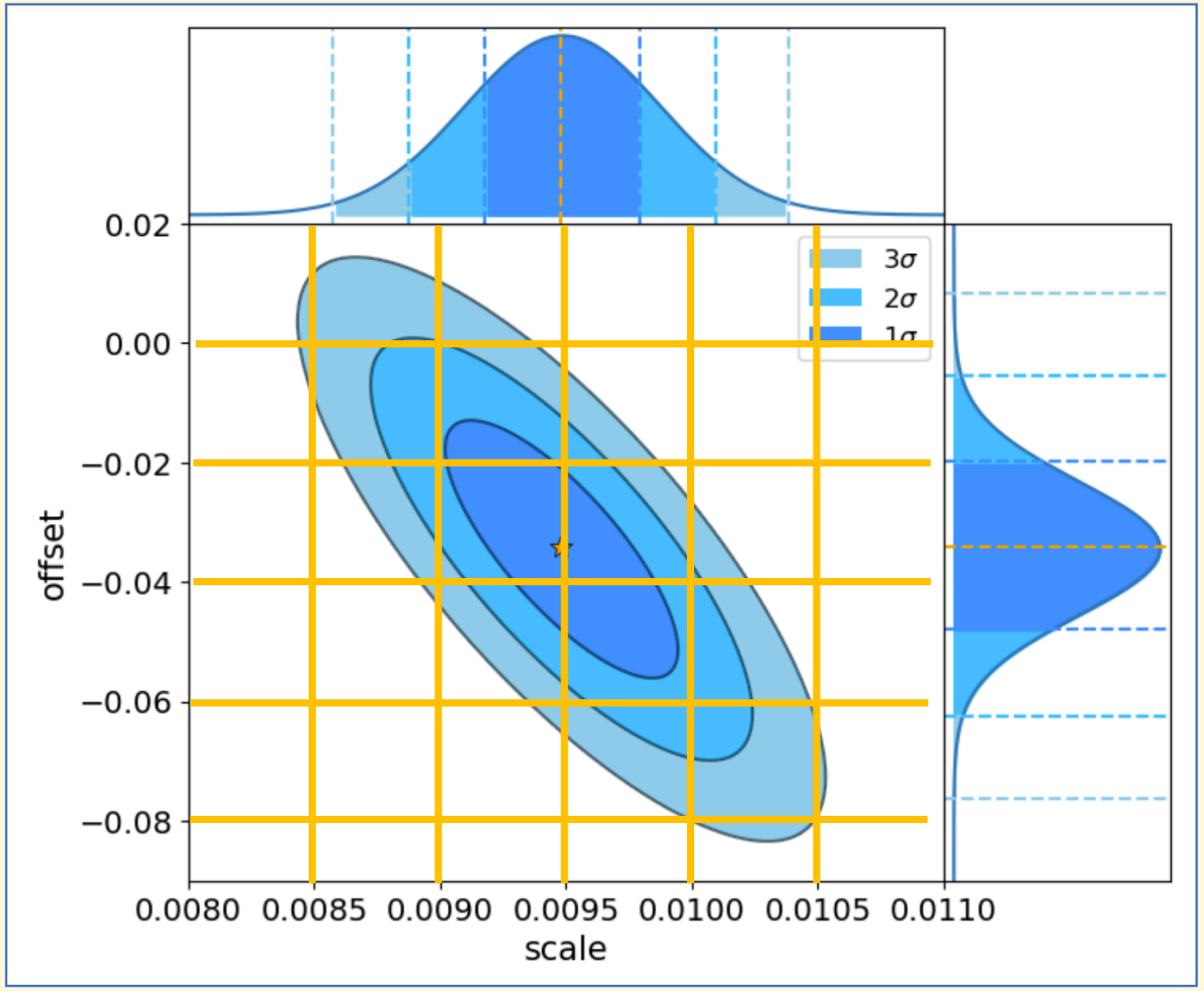
$$p(\textbf{happiness} \mid \textbf{GDP}) = p(\textbf{happiness}, \textbf{UK} \mid \textbf{GDP}) + p(\textbf{happiness}, \textbf{France} \mid \textbf{GDP}) + p(\textbf{happiness}, \textbf{Spain} \mid \textbf{GDP}) + \ldots$$

Bayes Theorem : In practice

- Physics is complex: priors and likelihoods are rarely analytic and routinely complicated
- Multiple peaks is often common



Bayes Theorem : Sampling




```
grid_like = []
for s in scale:
    for o in offset:
        like_val = chi2(data, model(*(x,y))
        grid_like.append(like_val)
...
plt.contour(scale, offset, grid_like)
```

- Very slow and very inefficient
- Requires starting point

Markov Chain Monte Carlo Sampling

Monte Carlo:

- Computer simulation where we generate many realisations of the data
- Either from the data (e.g. uncertainties) or the model

```
> from numpy import random
```

Markov chain:

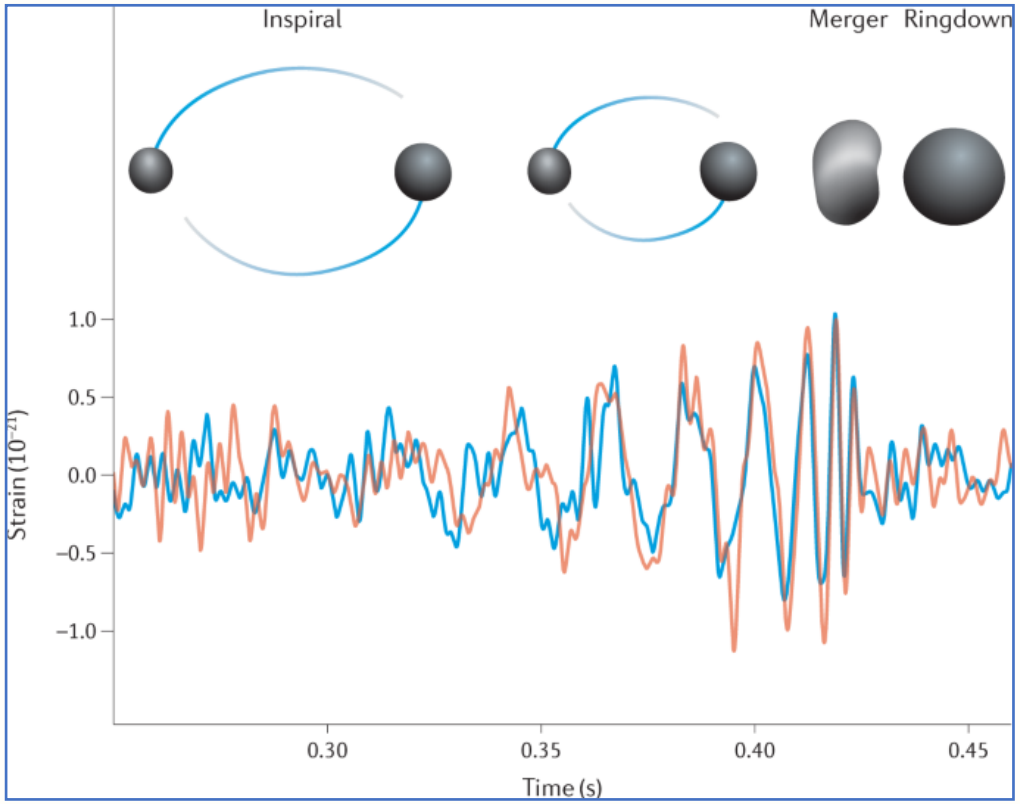
- Sequence of events : each outcome determines what happens next
- No prior memory; only the current step is relevant

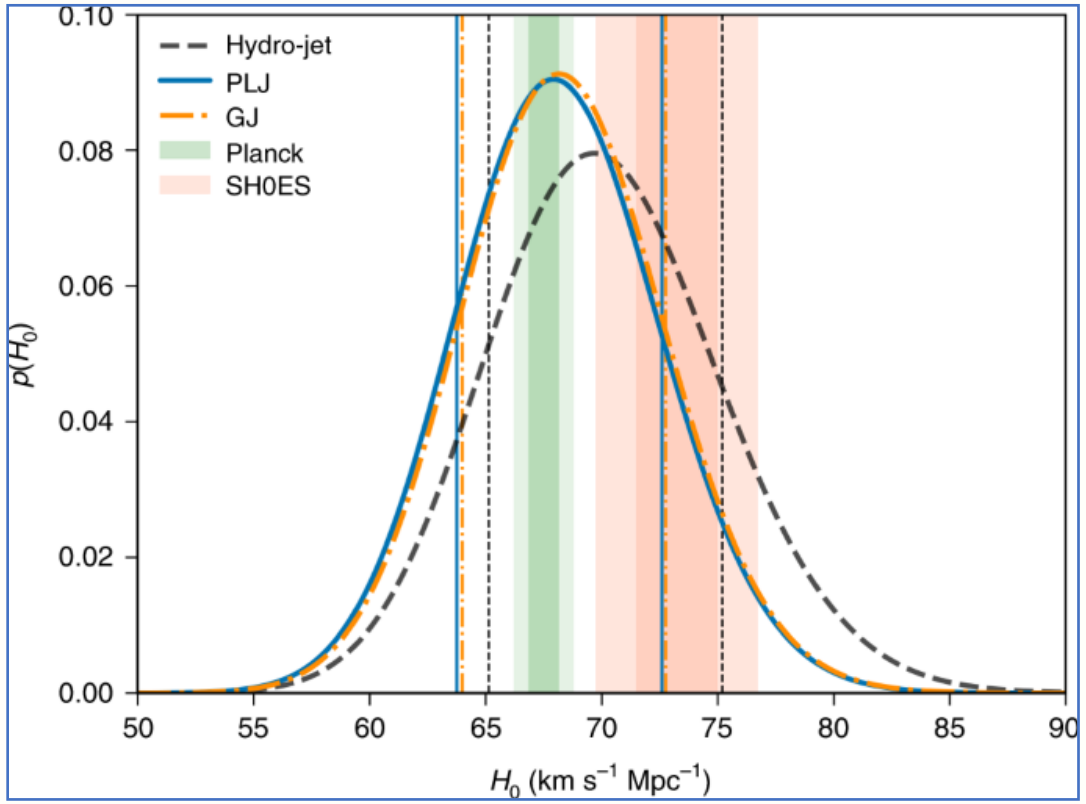
MCMC:

- Stochastic process to explore a parameter space
- Sampler of the posterior: $p(\theta \mid D) \propto p(D \mid \theta)\, p(\theta)$
- Many different mathematically motivated examples

Markov Chain Monte Carlo Sampling

Very common across physics





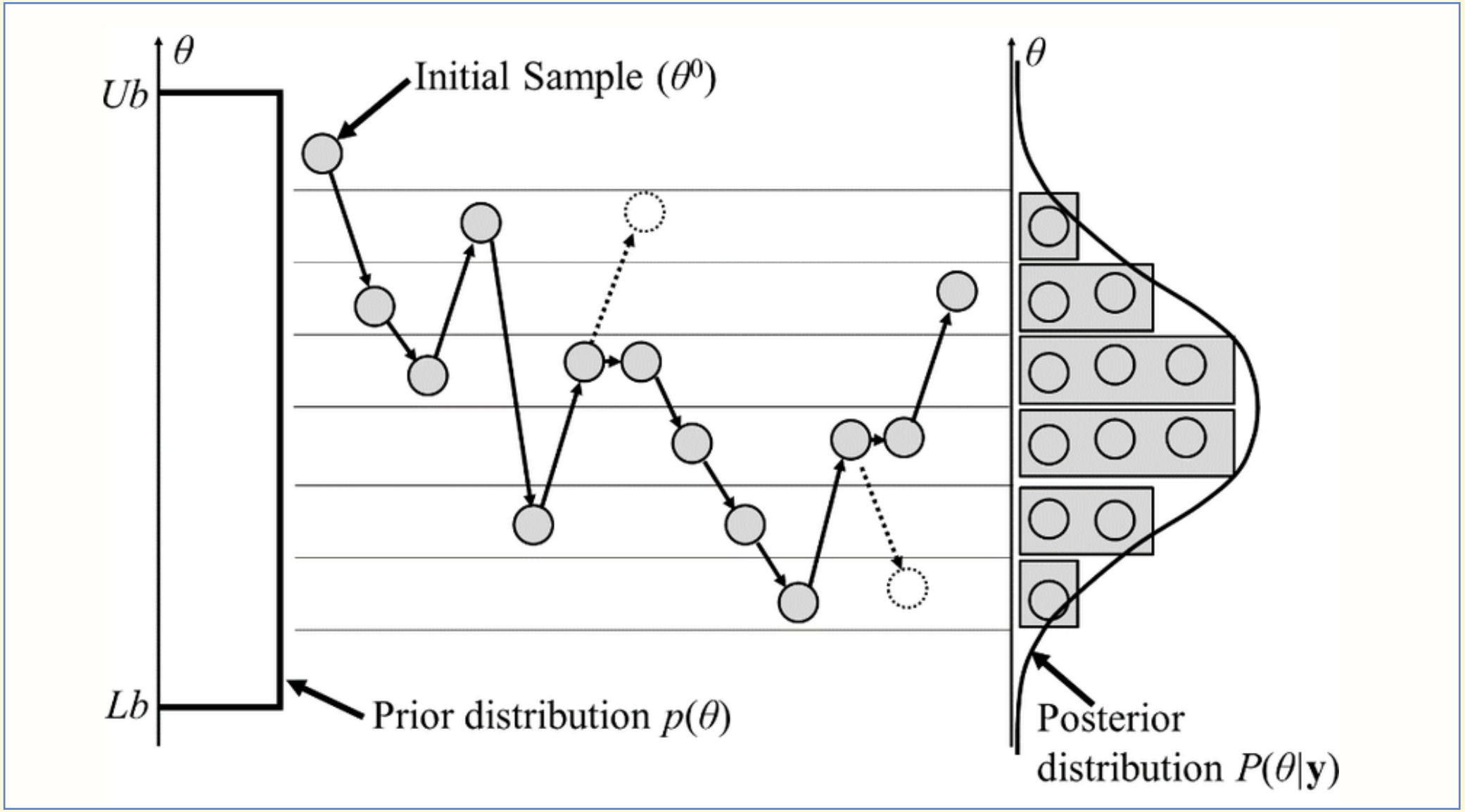
Metropolis-Hastings Algorithm

Consider a distribution $P(\theta)$

1. Make an initial guess for the variables (θ_{current})
2. Propose a random step to a new location ($\theta_{\text{proposed}} = \theta_{\text{current}} + \Delta\theta$):
3. Calculate the probability of both points ($p(\theta_{\text{current}})$, $p(\theta_{\text{proposed}})$)
4. If $p(\theta_{\text{proposed}}) > p(\theta_{\text{current}})$ then accept θ_{proposed}
5. If $p(\theta_{\text{proposed}}) < p(\theta_{\text{current}})$ then:
    1. Draw a *uniform random number,* u, between [0,1]
    2. If $u \leq \frac{p(\theta_{\text{proposed}})}{p(\theta_{\text{current}})}$ then accept θ_{proposed}
6. else: then stay at θ_{current}

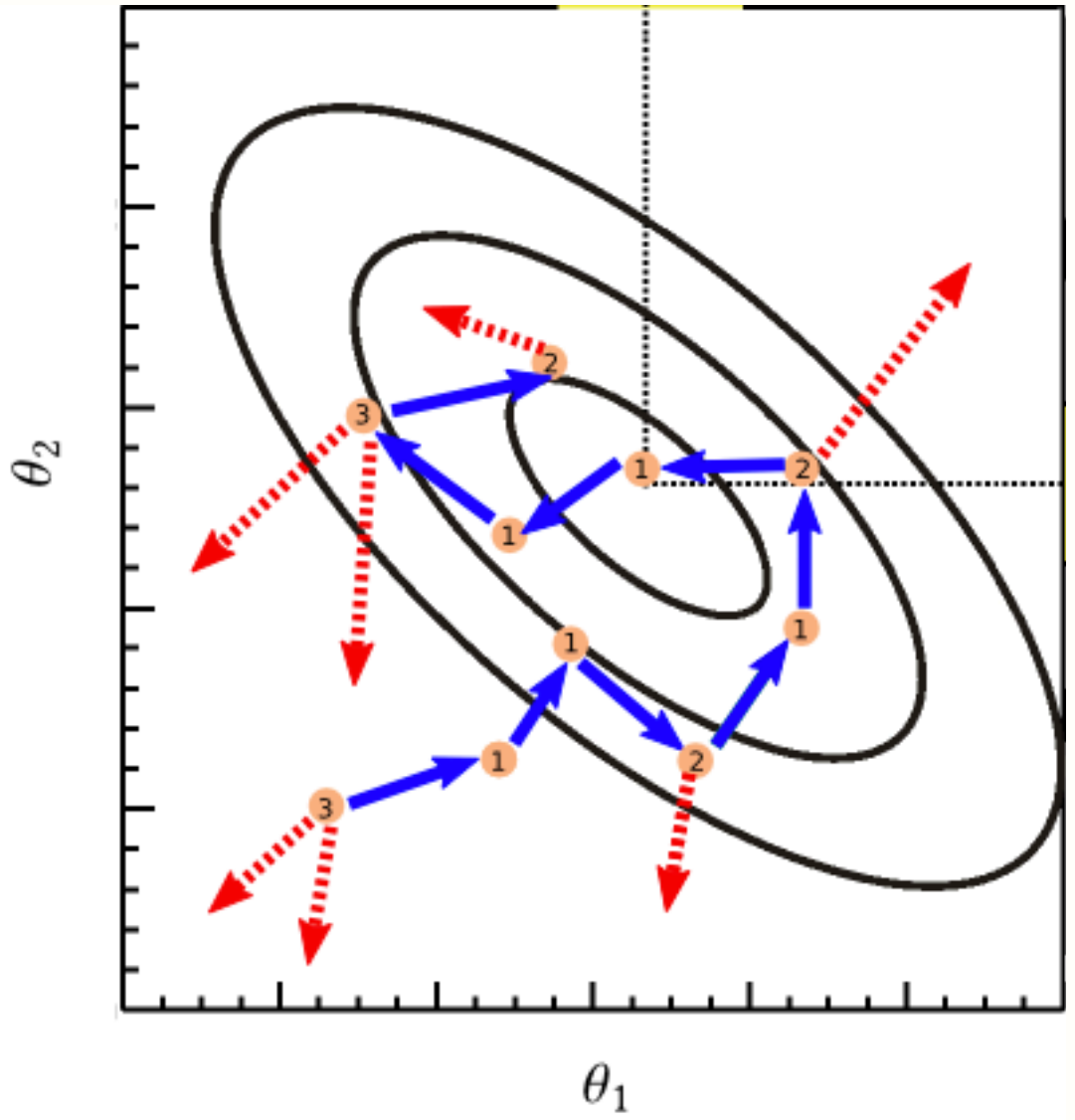
Metropolis-Hastings Algorithm

Consider a distribution $P(\theta)$



Metropolis-Hastings Algorithm

Consider a 2D distribution $P(\theta)$



Metropolis-Hastings Algorithm

Key features:

- no need to know the true normalised probabilities

$$p_{\text{move}} = \frac{p(\theta_{\text{proposed}})}{p(\theta_{\text{current}})}$$

- only requirement is that $p > 0$
- no memory of previous steps

Simple to implement:

1. Generate a random value
2. Evaluate $p(\theta_{\text{proposed}})$
3. Draw a random *uniform* value
4. Repeat N times

Metropolis-Hastings Algorithm

Common problems:

- starting value far from $\mathscr{L}_{\max}$
  - Reject the first N steps : the **burn in**
- suitable values for $\Delta\theta$
  - Small value == slow exploration
  - Large value == low acceptance rate
- local maxima
  - multiple peaks in the distribution

Trial & error

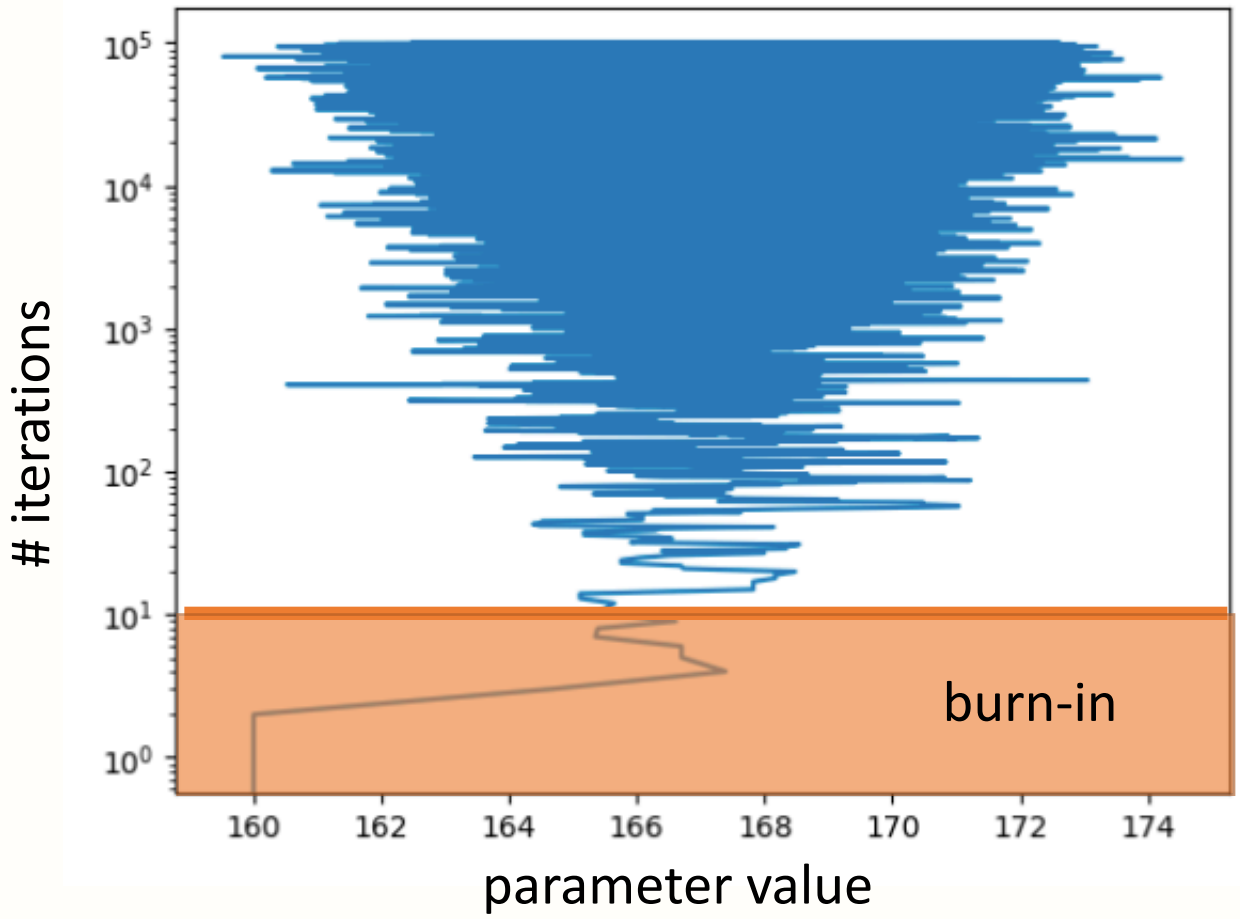
Metropolis-Hastings Algorithm

Advantages:

- easy to code
  - plenty of pre-made packages available
- easy to test
  - fine-tuning is possible with modern computers
- mathematically rigorous
  - always works as $n_{\text{samples}} \to \infty$

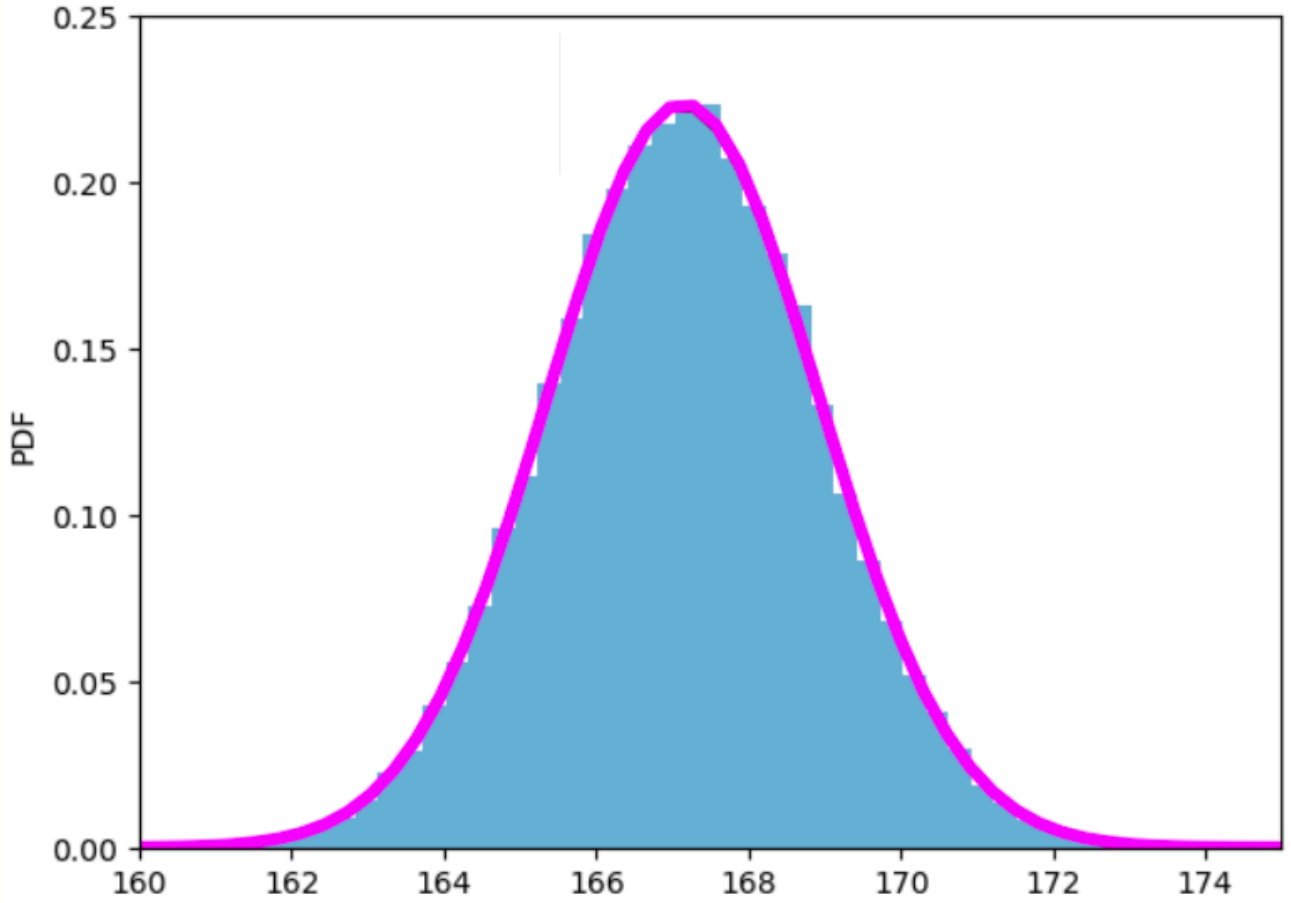
Convergence: look at the data

many mathematical approaches available

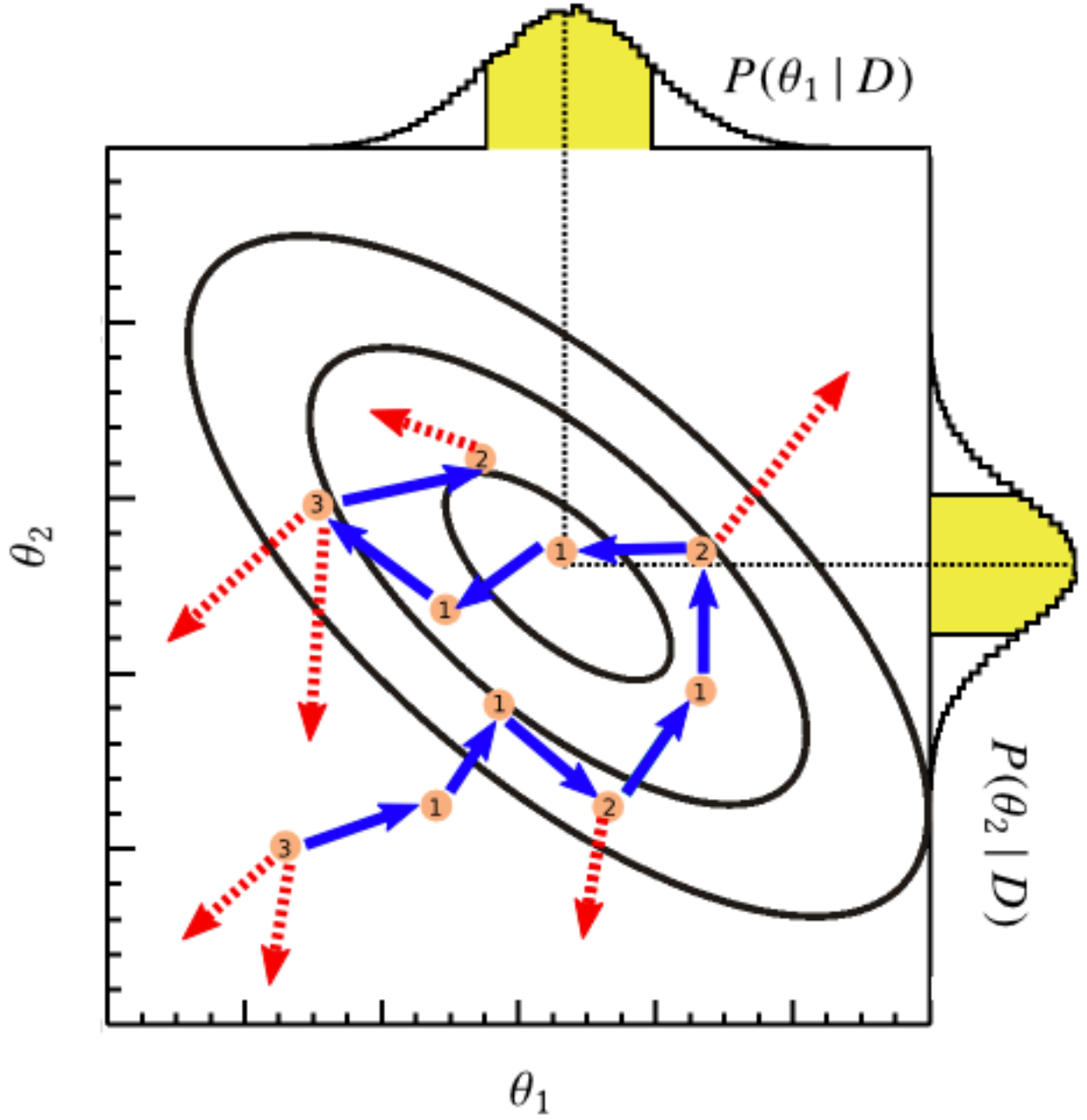


- Calculate mean, σ at regular intervals ; calculate acceptance rate (~0.4)
- Perform and combine multiple chains from different initial conditions

Finding the posterior: marginalisation



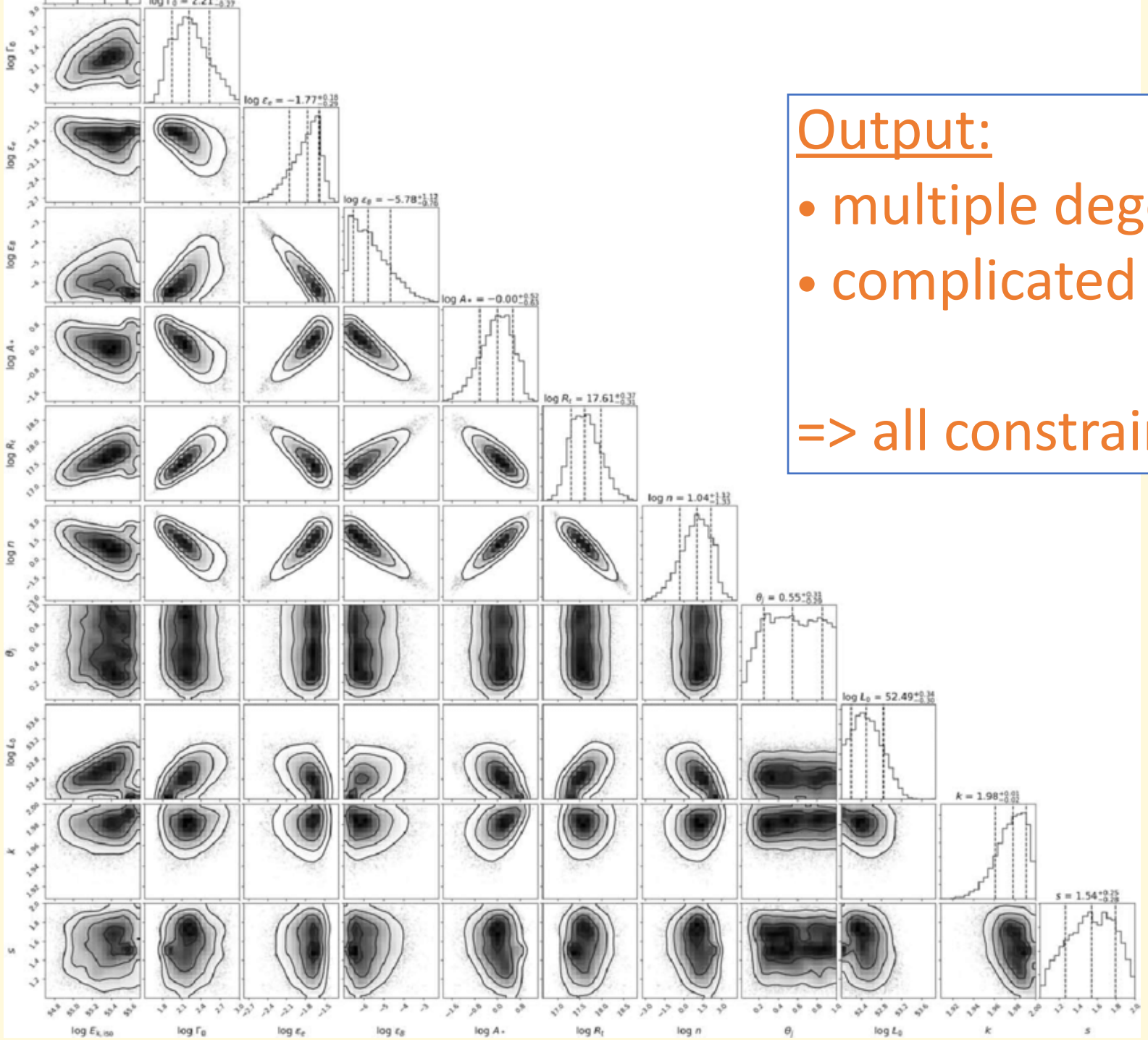
The cool part: A histogram of the chain is the marginalisation!


$P(\theta_1 \mid D)$
θ_2
θ_1
$P(\theta_2 \mid D)$

Markov-Chain Monte-Carlo

Alternatives:

- Gibbs Sampling :
  - https://en.wikipedia.org/wiki/Gibbs_sampling
- Nested Sampling :
  - https://en.wikipedia.org/wiki/Nested_sampling_algorithm
- Hierarchical Regression :
  - https://en.wikipedia.org/wiki/Multilevel_model
- Hamiltonian Monte Carlo (inc NUTS):
  - https://en.wikipedia.org/wiki/Hamiltonian_Monte_Carlo



Output:

- multiple degenerate parameters
- complicated distributions

=> all constrained simultaneously

Week 13: Learning outcomes

Today we have learnt

- The mathematical formalism for Bayes theorem
  - The difference between the *posterior*, *likelihood*, *prior* and *evidence*
- The fundamental difference between Bayesian v Frequentist
- How to sample a/multiple parameters efficiently
  - Markov Chain Monte Carlo (MCMC)
  - The Metropolis-Hastings algorithm
- How to implement this in python

Practical examples on Friday!