Bayesian statistics and sampling techniques

Florian Ruppin (Univ. Lyon I) & Cyrille Doux (LPSC Grenoble)

IN2P3 School of statistics - June 5th 2026

Learning objectives

We will learn:

Learning objectives

We will learn:

1. Basics of Bayesian statistics

Learning objectives

We will learn:

1. Basics of Bayesian statistics
2. Mathematical differences with the frequentist approach

Learning objectives

We will learn:

1. Basics of Bayesian statistics
2. Mathematical differences with the frequentist approach
3. Traditional posterior sampling techniques

Learning objectives

We will learn:

1. Basics of Bayesian statistics
2. Mathematical differences with the frequentist approach
3. Traditional posterior sampling techniques
4. How to generate, represent, and process MCMC samples

Learning objectives

We will learn:

1. Basics of Bayesian statistics
2. Mathematical differences with the frequentist approach
3. Traditional posterior sampling techniques
4. How to generate, represent, and process MCMC samples
5. Beyond Metropolis-Hastings

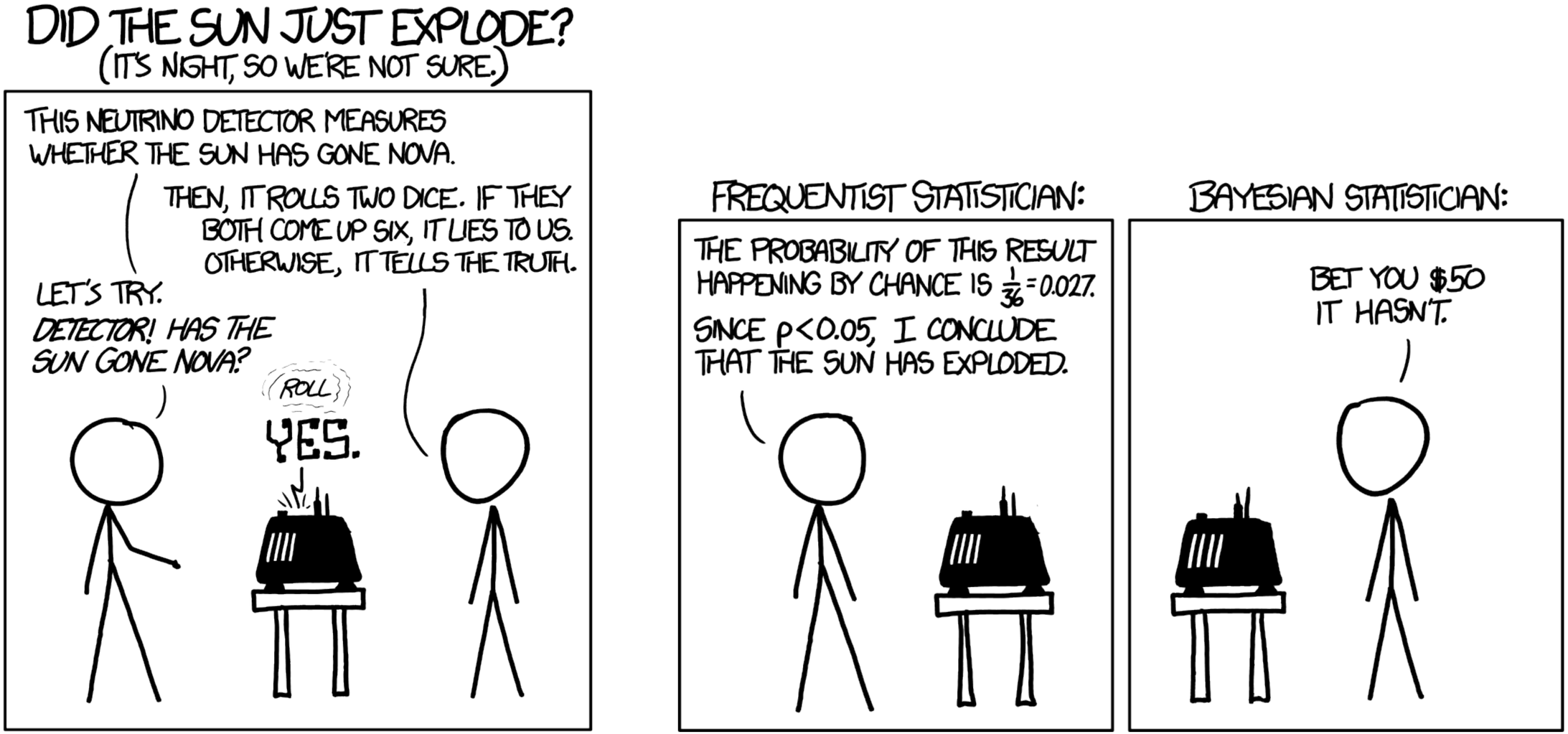
Deduction and induction

Frequentist and Bayesian

- **Deduction**: type of logical reasoning linking initial propositions (*premises*) to a final proposition (*conclusion*)
  - *Example*: I know that if A is true then B is true. I observe that A is true so I deduce that B is also true. I observe that B is false so I deduce that A is also false.
- Very often we cannot apply this type of reasoning because we do not have enough information.
- **Induction**: type of logical reasoning allowing general laws to be formed from particular facts on a probabilistic basis.
  - *Example*: I know that if A is true then B is true. I observe that B is true so I deduce that A is more plausible
- The deductive approach is the basis of the **frequentist** method.
  Probability = limit of a frequency
- The inductive approach is the basis of the **Bayesian** method.
  Probability = plausibility of a proposition within a given paradigm
- The frequentist approach assigns probabilities to data while the Bayesian approach assigns probabilities to hypotheses.

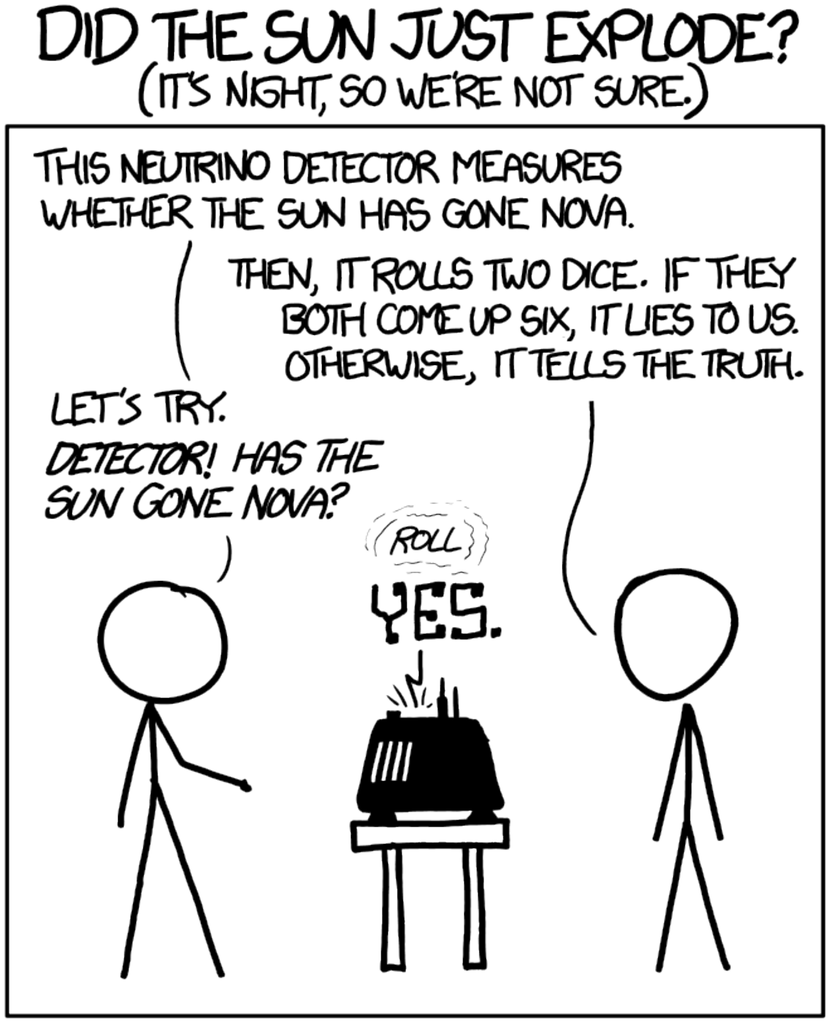
Deduction and induction

Frequentist and Bayesian

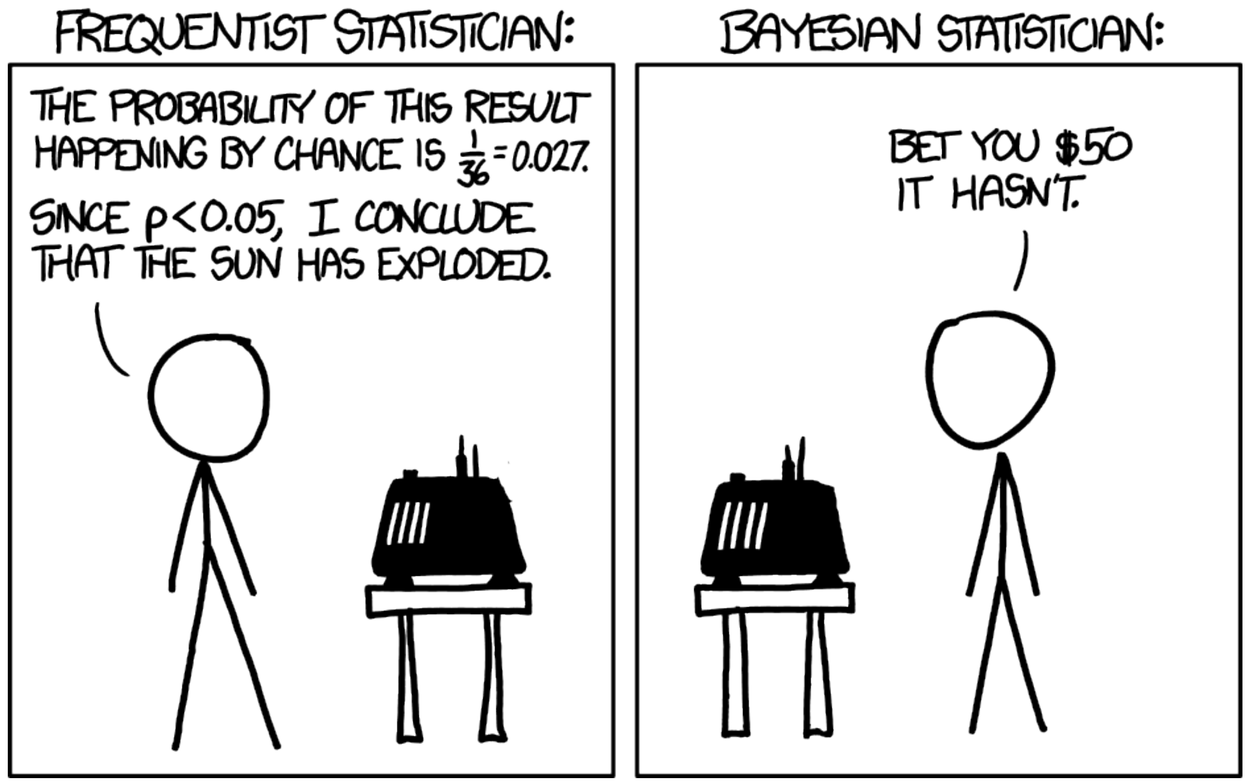


Deduction and induction

Frequentist and Bayesian



A true frequentist statistician would run the machine several more times…

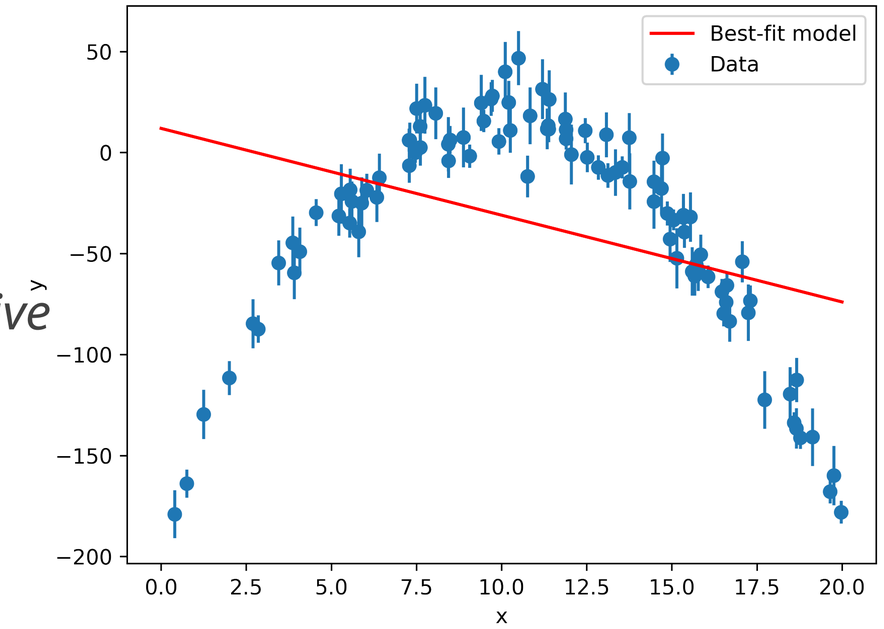


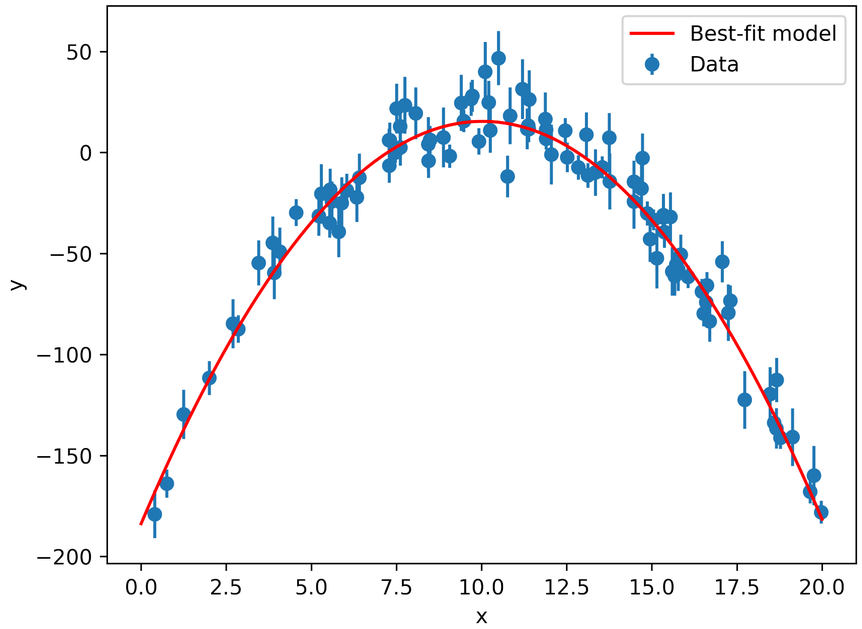
What is a good model?

- Two criteria define a good model:
  - **representative**
  - generalisable
- A model is representative of a dataset if the residuals obtained by subtracting this model from the data have the characteristics of the measurement noise (RMS, spectrum)

A model can perfectly well be representative of the data without corresponding to their underlying distribution (large measurement uncertainties)

Bad model: not representative





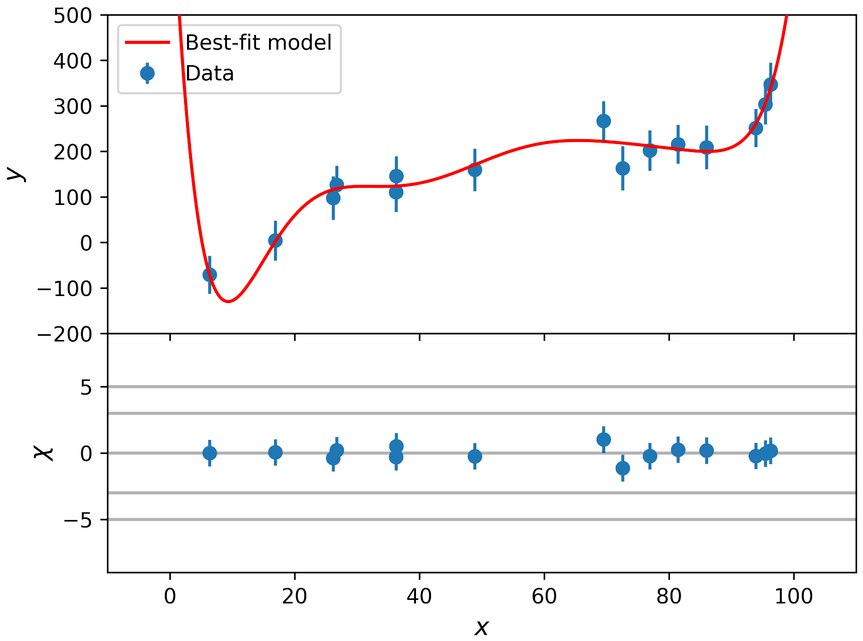
Good model: representative

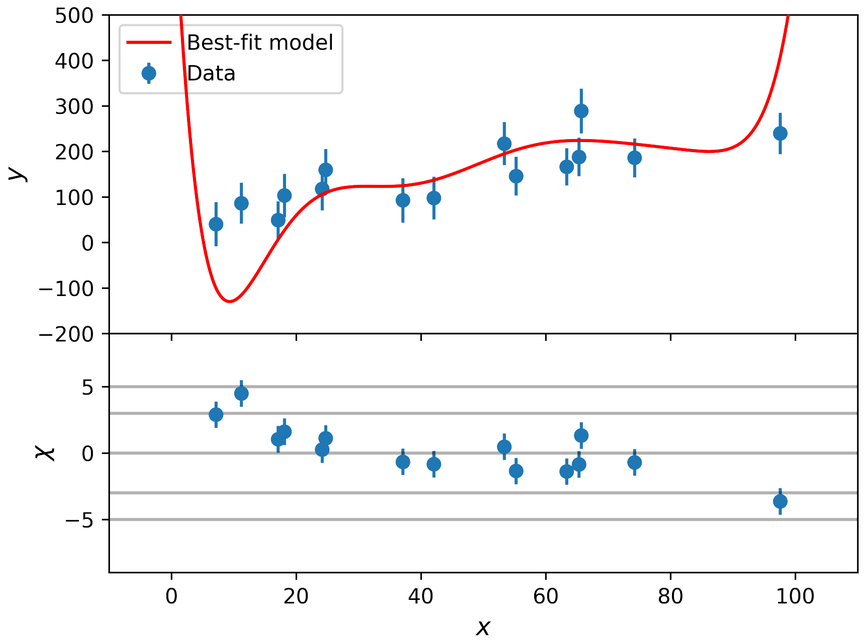
What is a good model?

- Two criteria define a good model:
  - representative
  - **generalisable**
- A model is generalizable if it can be applied to several realizations of the same underlying distribution

A generalizable model may turn out to be non-representative when measurement uncertainties decrease → always question the models!

Representative model…





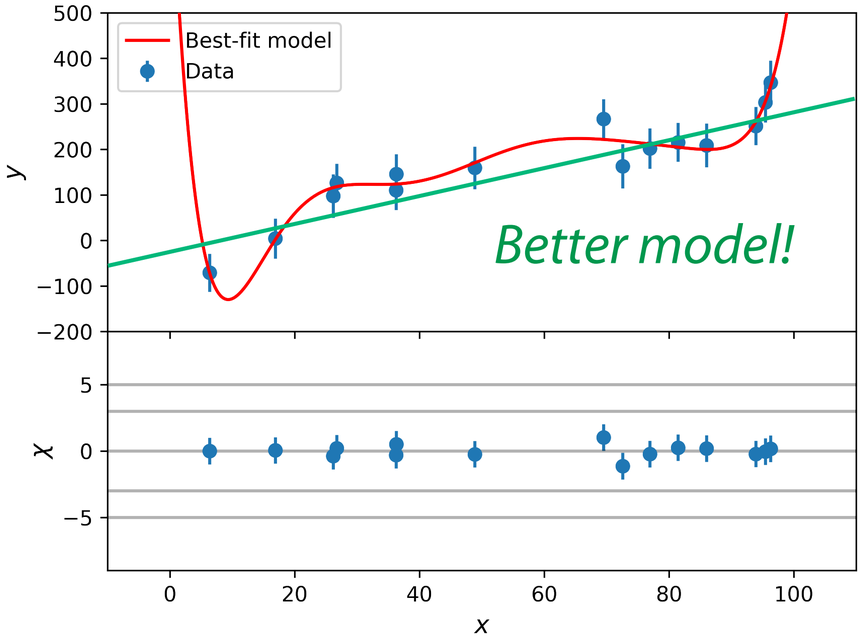
… but not generalizable!

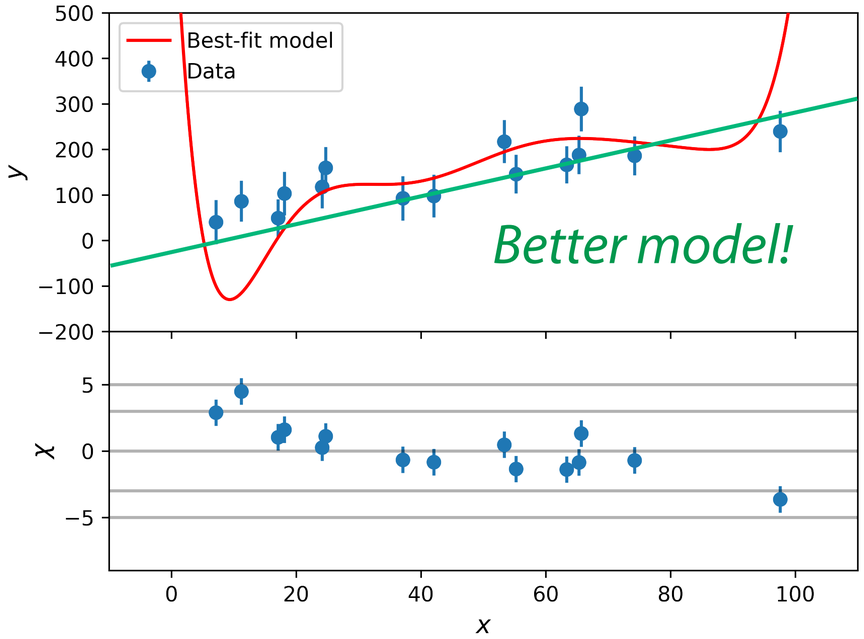
What is a good model?

- Two criteria define a good model:
  - representative
  - **generalisable**
- A model is generalizable if it can be applied to several realizations of the same underlying distribution

A generalizable model may turn out to be non-representative when measurement uncertainties decrease → always question the models!

Representative model…





… but not generalizable!

Maximum likelihood Best-fit

Maximum likelihood Best-fit

- The **data** vector $\vec{d}$ with N dimensions is considered a **random draw** from a joint probability **distribution** f that depends on unknown parameters $\vec{\theta} = [\theta_1, \ldots, \theta_k]^T$

Maximum likelihood Best-fit

- The **data** vector $\vec{d}$ with N dimensions is considered a **random draw** from a joint probability **distribution** f that depends on unknown parameters $\vec{\theta} = [\theta_1, \ldots, \theta_k]^T$
- Goal of maximum likelihood estimation: find $\hat{\vec{\theta}}$ such that $f(\vec{d}; \hat{\vec{\theta}})$ is maximum. Assume the **data** are not an improbable realization of the underlying distribution but the **most probable case**

Maximum likelihood Best-fit

- The **data** vector $\vec{d}$ with N dimensions is considered a **random draw** from a joint probability **distribution** f that depends on unknown parameters $\vec{\theta} = [\theta_1, \ldots, \theta_k]^T$
- Goal of maximum likelihood estimation: find $\hat{\vec{\theta}}$ such that $f(\vec{d}; \hat{\vec{\theta}})$ is maximum. Assume the **data** are not an improbable realization of the underlying distribution but the **most probable case**
- The form of the **underlying distribution** depends on the **physical process** considered

Maximum likelihood Best-fit

- The **data** vector $\vec{d}$ with N dimensions is considered a **random draw** from a joint probability **distribution** f that depends on unknown parameters $\vec{\theta} = [\theta_1, \ldots, \theta_k]^T$
- Goal of maximum likelihood estimation: find $\hat{\vec{\theta}}$ such that $f(\vec{d}; \hat{\vec{\theta}})$ is maximum. Assume the **data** are not an improbable realization of the underlying distribution but the **most probable case**
- The form of the **underlying distribution** depends on the **physical process** considered
- *Examples* :
  - flux measurement of an astrophysical object: flux has a true value and random part comes from noise
    - Normal distribution
  - measurement of the number of galaxy clusters as a function of redshift: counting process
    - Poisson distribution

Maximum likelihood Best-fit

- The **data** vector $\vec{d}$ with N dimensions is considered a **random draw** from a joint probability **distribution** f that depends on unknown parameters $\vec{\theta} = [\theta_1, \ldots, \theta_k]^T$
- Goal of maximum likelihood estimation: find $\hat{\vec{\theta}}$ such that $f(\vec{d}; \hat{\vec{\theta}})$ is maximum. Assume the **data** are not an improbable realization of the underlying distribution but the **most probable case**
- The form of the **underlying distribution** depends on the **physical process** considered
- *Examples* :
  - flux measurement of an astrophysical object: flux has a true value and random part comes from noise
    - Normal distribution
  - measurement of the number of galaxy clusters as a function of redshift: counting process
    - Poisson distribution
- If our measurements of the same physical process are **independent** then the underlying distribution satisfies: $f(\vec{d}; \vec{\theta}) = \prod_{k=1}^{N} f_k(d_k; \vec{\theta})$

Maximum likelihood Best-fit

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that

$f(\vec{d}; \hat{\mu}, \hat{\sigma})$ *is maximum*

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that

$f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d}; \mu, \sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k; \mu, \sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d}; \hat{\mu}, \hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2 \times \log(f)$

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

$$\text{And } \log[\mathcal{N}(d_k;\mu,\sigma)] = \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d_k-\mu)^2}{2\sigma^2}\right)\right] = -\frac{1}{2}\frac{(d_k-\mu)^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$$

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

And $\log[\mathcal{N}(d_k;\mu,\sigma)] = \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d_k-\mu)^2}{2\sigma^2}\right)\right] = -\frac{1}{2}\frac{(d_k-\mu)^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$

Therefore $-2\times\log[f(\vec{d};\mu,\sigma)] = \sum_{k=1}^{N}\left[\frac{(d_k-\mu)^2}{\sigma^2} + \log(2\pi\sigma^2)\right]$

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

And $\log[\mathcal{N}(d_k;\mu,\sigma)] = \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d_k-\mu)^2}{2\sigma^2}\right)\right] = -\frac{1}{2}\frac{(d_k-\mu)^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$

Therefore $-2\times\log[f(\vec{d};\mu,\sigma)] = \sum_{k=1}^{N}\left[\frac{(d_k-\mu)^2}{\sigma^2} + \log(2\pi\sigma^2)\right]$

The minimum of this function is reached when $\chi^2 = \sum_{k=1}^{N}\frac{(d_k-\mu)^2}{\sigma^2}$ is minimum

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

And $\log[\mathcal{N}(d_k;\mu,\sigma)] = \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d_k-\mu)^2}{2\sigma^2}\right)\right] = -\frac{1}{2}\frac{(d_k-\mu)^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$

Therefore $-2\times\log[f(\vec{d};\mu,\sigma)] = \sum_{k=1}^{N}\left[\frac{(d_k-\mu)^2}{\sigma^2} + \log(2\pi\sigma^2)\right]$

The minimum of this function is reached when $\chi^2 = \sum_{k=1}^{N}\frac{(d_k-\mu)^2}{\sigma^2}$ is minimum

Since χ^2 is generally not an analytical function, it is minimized numerically to find the best fit

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

  And $\log[\mathcal{N}(d_k;\mu,\sigma)] = \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d_k-\mu)^2}{2\sigma^2}\right)\right] = -\frac{1}{2}\frac{(d_k-\mu)^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$

  Therefore $-2\times\log[f(\vec{d};\mu,\sigma)] = \sum_{k=1}^{N}\left[\frac{(d_k-\mu)^2}{\sigma^2} + \log(2\pi\sigma^2)\right]$

  The minimum of this function is reached when $\chi^2 = \sum_{k=1}^{N}\frac{(d_k-\mu)^2}{\sigma^2}$ is minimum

  Since χ^2 is generally not an analytical function, it is minimized numerically to find the best fit

- For most Gaussian processes, σ is known and corresponds to the RMS of the measurement noise

Maximum likelihood Best-fit

- Application example in the Gaussian case: $f(\vec{d};\mu,\sigma) = \prod_{k=1}^{N} \mathcal{N}(d_k;\mu,\sigma)$

we look for $\hat{\mu}$ and $\hat{\sigma}$ such that $f(\vec{d};\hat{\mu},\hat{\sigma})$ is maximum

- Instead of looking for the maximum of f, we generally look for the minimum of $-2\times\log(f)$
- In the Gaussian case we have:

  And $\log[\mathcal{N}(d_k;\mu,\sigma)] = \log\left[\frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{(d_k-\mu)^2}{2\sigma^2}\right)\right] = -\frac{1}{2}\frac{(d_k-\mu)^2}{\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2)$

  Therefore $-2\times\log[f(\vec{d};\mu,\sigma)] = \sum_{k=1}^{N}\left[\frac{(d_k-\mu)^2}{\sigma^2} + \log(2\pi\sigma^2)\right]$

  The minimum of this function is reached when $\chi^2 = \sum_{k=1}^{N}\frac{(d_k-\mu)^2}{\sigma^2}$ is minimum

  Since χ^2 is generally not an analytical function, it is minimized numerically to find the best fit

- For most Gaussian processes, σ is known and corresponds to the RMS of the measurement noise
  - we just have to estimate μ (*may be a function*)

Maximum likelihood Parameter uncertainties

Maximum likelihood Parameter uncertainties

- Once the minimum is reached, it is possible to numerically compute the Hessian matrix $H_{ij}(m) = \frac{1}{\sigma} \frac{\partial^2 m}{\partial \theta_i \partial \theta_j} \Bigg|_{\hat{\vec{\theta}}}$

  associated with the model m considered to fit the data. This matrix estimates the relative variations of the model given the uncertainty on the data for a variation of the parameters $\vec{\theta}$ around the best-fit $\hat{\vec{\theta}}$.

Maximum likelihood Parameter uncertainties

- Once the minimum is reached, it is possible to numerically compute the Hessian matrix $H_{ij}(m) = \frac{1}{\sigma} \frac{\partial^2 m}{\partial \theta_i \partial \theta_j} \Bigg|_{\hat{\vec{\theta}}}$

  associated with the model m considered to fit the data. This matrix estimates the relative variations of the model given the uncertainty on the data for a variation of the parameters $\vec{\theta}$ around the best-fit $\hat{\vec{\theta}}$.
- Assuming that the data correspond to a Gaussian random draw around the model m with standard deviation σ, the inverse of the Hessian matrix gives the covariance matrix $\Sigma_{\vec{\theta}}$ associated with the parameters.

Maximum likelihood Parameter uncertainties

- Once the minimum is reached, it is possible to numerically compute the Hessian matrix $H_{ij}(m) = \frac{1}{\sigma} \frac{\partial^2 m}{\partial \theta_i \partial \theta_j} \Bigg|_{\hat{\vec{\theta}}}$

  associated with the model m considered to fit the data. This matrix estimates the relative variations of the model given the uncertainty on the data for a variation of the parameters $\vec{\theta}$ around the best-fit $\hat{\vec{\theta}}$.

- Assuming that the data correspond to a Gaussian random draw around the model m with standard deviation σ, the inverse of the Hessian matrix gives the covariance matrix $\Sigma_{\vec{\theta}}$ associated with the parameters.

- Knowing the covariance matrix of the parameters, we can define the confidence contours by diagonalizing it: eigenvalues = *axis lengths* | *eigenvectors = axis orientation*

Maximum likelihood Parameter uncertainties

- Once the minimum is reached, it is possible to numerically compute the Hessian matrix $H_{ij}(m) = \frac{1}{\sigma} \frac{\partial^2 m}{\partial \theta_i \partial \theta_j} \Big|_{\hat{\vec{\theta}}}$

  associated with the model m considered to fit the data. This matrix estimates the relative variations of the model given the uncertainty on the data for a variation of the parameters $\vec{\theta}$ around the best-fit $\hat{\vec{\theta}}$.

- Assuming that the data correspond to a Gaussian random draw around the model m with standard deviation σ, the inverse of the Hessian matrix gives the covariance matrix $\Sigma_{\vec{\theta}}$ associated with the parameters.

- Knowing the covariance matrix of the parameters, we can define the confidence contours by diagonalizing it:
  eigenvalues = *axis lengths* | *eigenvectors = axis orientation*

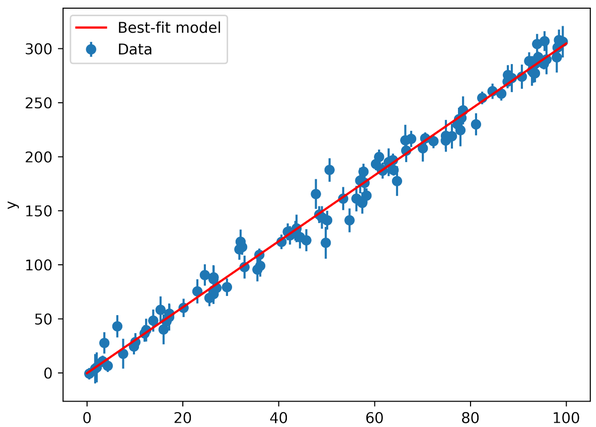
- In practice: perform a drawing from a multi-normal distribution with mean $\hat{\vec{\theta}}$ and covariance $\Sigma_{\vec{\theta}}$ and draw the ellipses using quantiles

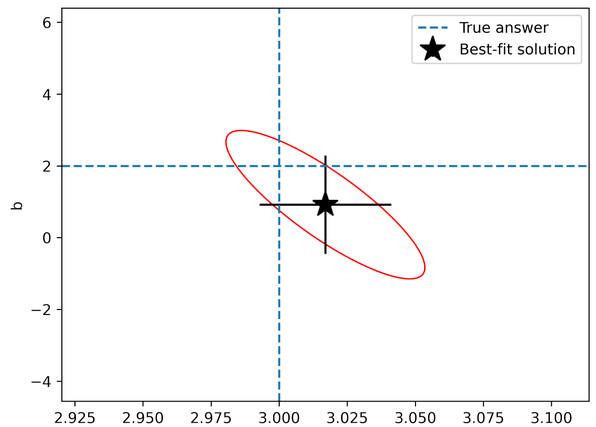
Maximum likelihood Parameter uncertainties

- Once the minimum is reached, it is possible to numerically compute the Hessian matrix $H_{ij}(m) = \frac{1}{\sigma} \frac{\partial^2 m}{\partial\theta_i \partial\theta_j}\Big|_{\hat{\vec{\theta}}}$ associated with the model m considered to fit the data. This matrix estimates the relative variations of the model given the uncertainty on the data for a variation of the parameters $\vec{\theta}$ around the best-fit $\hat{\vec{\theta}}$.
- Assuming that the data correspond to a Gaussian random draw around the model m with standard deviation σ, the inverse of the Hessian matrix gives the covariance matrix $\Sigma_{\vec{\theta}}$ associated with the parameters.
- Knowing the covariance matrix of the parameters, we can define the confidence contours by diagonalizing it: eigenvalues = *axis lengths* | *eigenvectors = axis orientation*
- In practice: perform a drawing from a multi-normal distribution with mean $\hat{\vec{\theta}}$ and covariance $\Sigma_{\vec{\theta}}$ and draw the ellipses using quantiles

Underlying distribution:

$$y = ax + b$$





Confidence contour at 1σ on the estimated parameters

Maximum likelihood

Maximum likelihood

- **Advantages**
  - Analytical solutions in several cases (exponential, Gaussian, Poisson likelihoods) if we have several measurements of the same random variable (*never the case in cosmology*)
  - Very fast compared to all other fitting methods
  - Preferred method in the case of easily reproducible Gaussian measurements

Maximum likelihood

- **Advantages**
  - Analytical solutions in several cases (exponential, Gaussian, Poisson likelihoods) if we have several measurements of the same random variable (*never the case in cosmology*)
  - Very fast compared to all other fitting methods
  - Preferred method in the case of easily reproducible Gaussian measurements
- **Limitations**
  - Numerical minimization required in most cases → strong dependence on the initial position (*start close to minimum*)
  - Problem of local minima: when the number of parameters is $\gtrsim 5$, must implement methods to escape local minima (*example: simulated annealing*) → increase in computation time
  - By assuming that the data are the most probable realization of a random process, we do not take into account prior knowledge of the underlying distribution
  - Difficult to *marginalize* over nuisance parameters

Bayesian inference

Motivation: from a point estimate to a distribution

Bayesian inference

Motivation: from a point estimate to a distribution

- Maximum likelihood gives $\hat{\theta}$ plus a Gaussian approximation from the Hessian.
  The underlying posterior is generically **non-Gaussian** *(skewed, multimodal, bounded by physical constraints, curved degeneracies)*. The Hessian misses all of this.

Bayesian inference

Motivation: from a point estimate to a distribution

- Maximum likelihood gives $\hat{\theta}$ plus a Gaussian approximation from the Hessian.
  The underlying posterior is generically **non-Gaussian** *(skewed, multimodal, bounded by physical constraints, curved degeneracies)*. The Hessian misses all of this.

- Physics rarely starts from scratch: positivity ($\Omega_m > 0$, masses, cross-sections), bounded ranges, calibration of nuisance parameters from auxiliary measurements, results from previous experiments.

Bayesian inference

Motivation: from a point estimate to a distribution

- Maximum likelihood gives $\hat{\theta}$ plus a Gaussian approximation from the Hessian.
  The underlying posterior is generically **non-Gaussian** *(skewed, multimodal, bounded by physical constraints, curved degeneracies)*. The Hessian misses all of this.
- Physics rarely starts from scratch: positivity ($\Omega_m > 0$, masses, cross-sections), bounded ranges, calibration of nuisance parameters from auxiliary measurements, results from previous experiments.
- Maximum likelihood discards this prior information by construction. We want a framework that **incorporates it explicitly and consistently**.

Bayesian inference

Motivation: from a point estimate to a distribution

- Maximum likelihood gives $\hat{\theta}$ plus a Gaussian approximation from the Hessian.
  The underlying posterior is generically **non-Gaussian** *(skewed, multimodal, bounded by physical constraints, curved degeneracies)*. The Hessian misses all of this.
- Physics rarely starts from scratch: positivity ($\Omega_m > 0$, masses, cross-sections), bounded ranges, calibration of nuisance parameters from auxiliary measurements, results from previous experiments.
- Maximum likelihood discards this prior information by construction. We want a framework that **incorporates it explicitly and consistently**.
- Detector systematics, foregrounds, calibration offsets… we don't care about their values, we care about $\theta_{physics}$ with their uncertainty properly propagated.

Bayesian inference

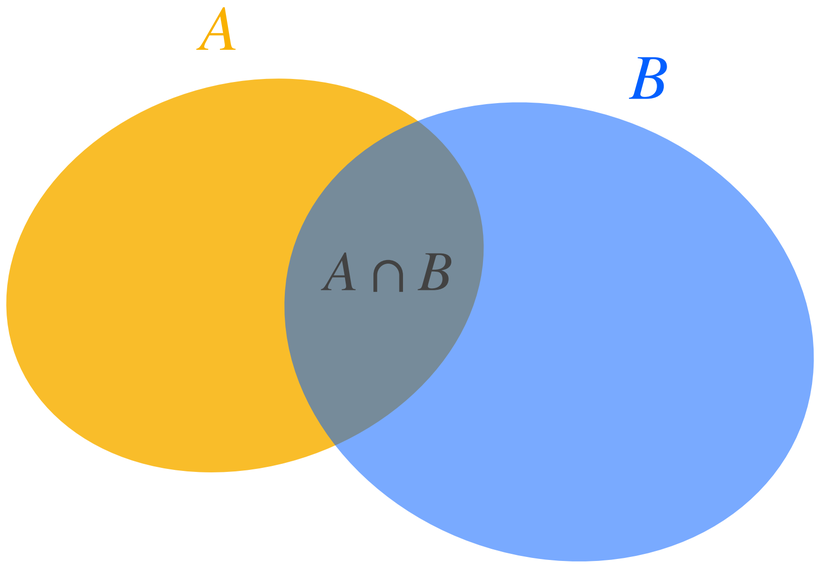
Motivation: from a point estimate to a distribution

- Maximum likelihood gives $\hat{\theta}$ plus a Gaussian approximation from the Hessian.
  The underlying posterior is generically **non-Gaussian** *(skewed, multimodal, bounded by physical constraints, curved degeneracies)*. The Hessian misses all of this.
- Physics rarely starts from scratch: positivity ($\Omega_m > 0$, masses, cross-sections), bounded ranges, calibration of nuisance parameters from auxiliary measurements, results from previous experiments.
- Maximum likelihood discards this prior information by construction. We want a framework that **incorporates it explicitly and consistently**.
- Detector systematics, foregrounds, calibration offsets… we don't care about their values, we care about $\theta_{physics}$ with their uncertainty properly propagated.
- With a posterior, this is **a single integral** — no profile-likelihood gymnastics.

Conditional probability distributions

- **Definition**: in a probability space Ω, for two (non-empty) events $A, B \subset \Omega$,

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



- For **continuous variables** with:
  - joint probability distribution $x, y \sim p_{X,Y}$,
  - marginal distributions $x \sim p_X$ and $y \sim p_Y$,
  - the conditional distribution of "X given $Y = y$" is

$$p_{X|Y}(x \mid y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}$$

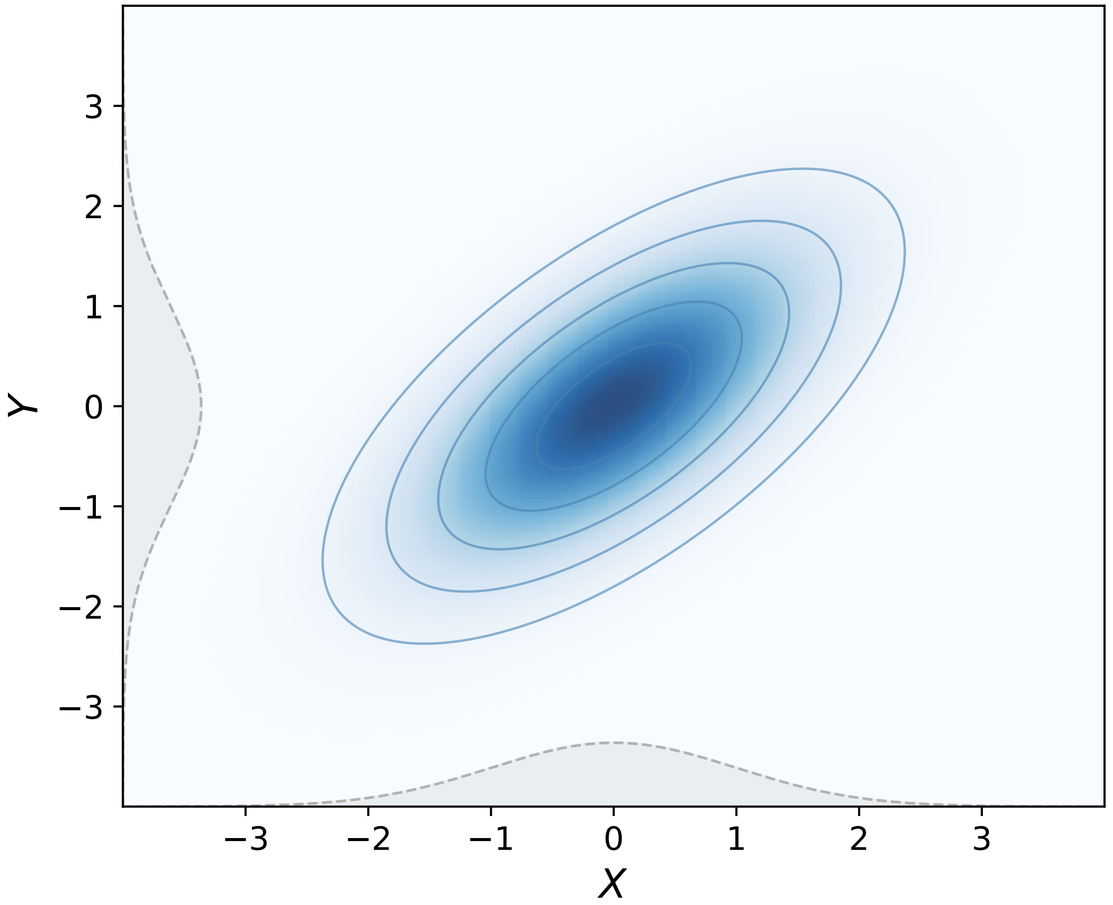
Conditional probability distributions

- **Definition**: in a probability space Ω, for two (non-empty) events $A, B \subset \Omega$,

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

- For **continuous variables** with:
  - joint probability distribution $x, y \sim p_{X,Y}$,
  - marginal distributions $x \sim p_X$ and $y \sim p_Y$,
  - the conditional distribution of "X given $Y = y$" is

$$p_{X|Y}(x \mid y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}$$



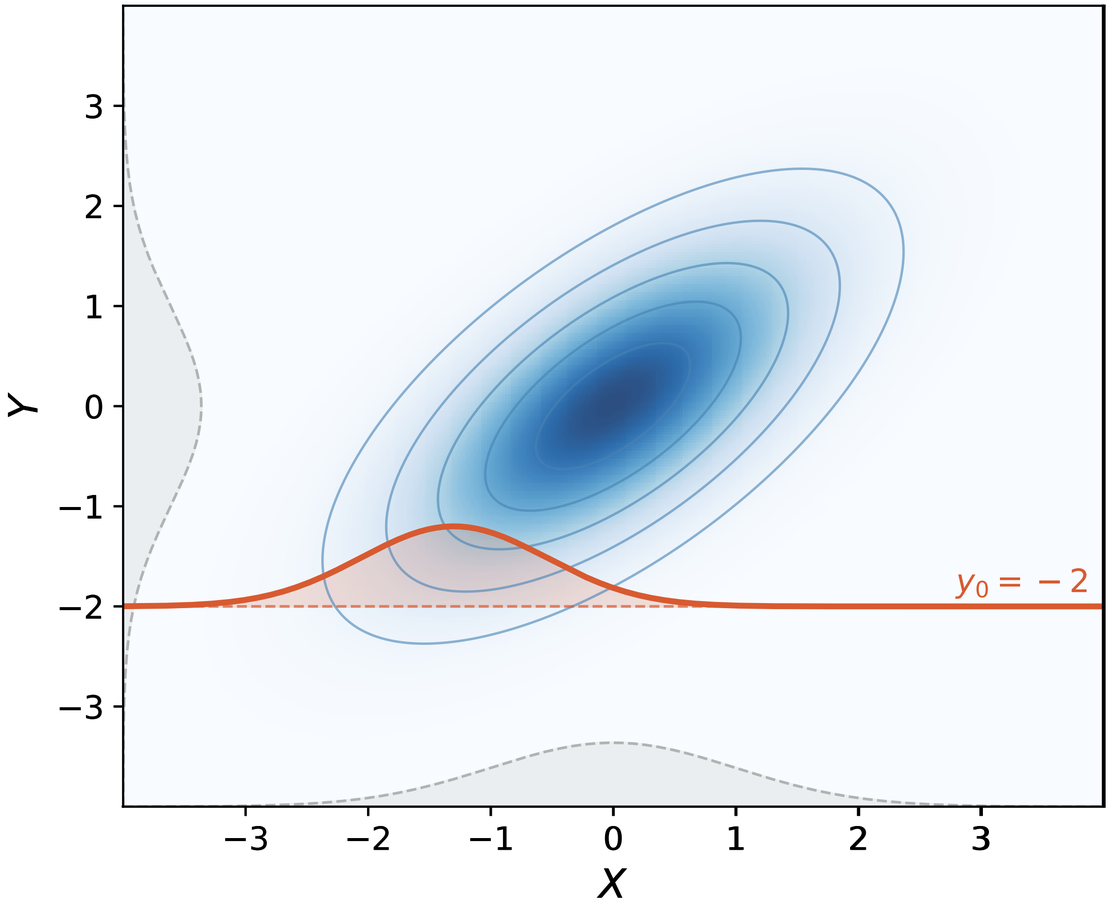
Conditional probability distributions

- **Definition**: in a probability space Ω, for two (non-empty) events $A, B \subset \Omega$,

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

- For **continuous variables** with:
  - joint probability distribution $x, y \sim p_{X,Y}$,
  - marginal distributions $x \sim p_X$ and $y \sim p_Y$,
  - the conditional distribution of "X given $Y = y$" is

$$p_{X|Y}(x \mid y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}$$



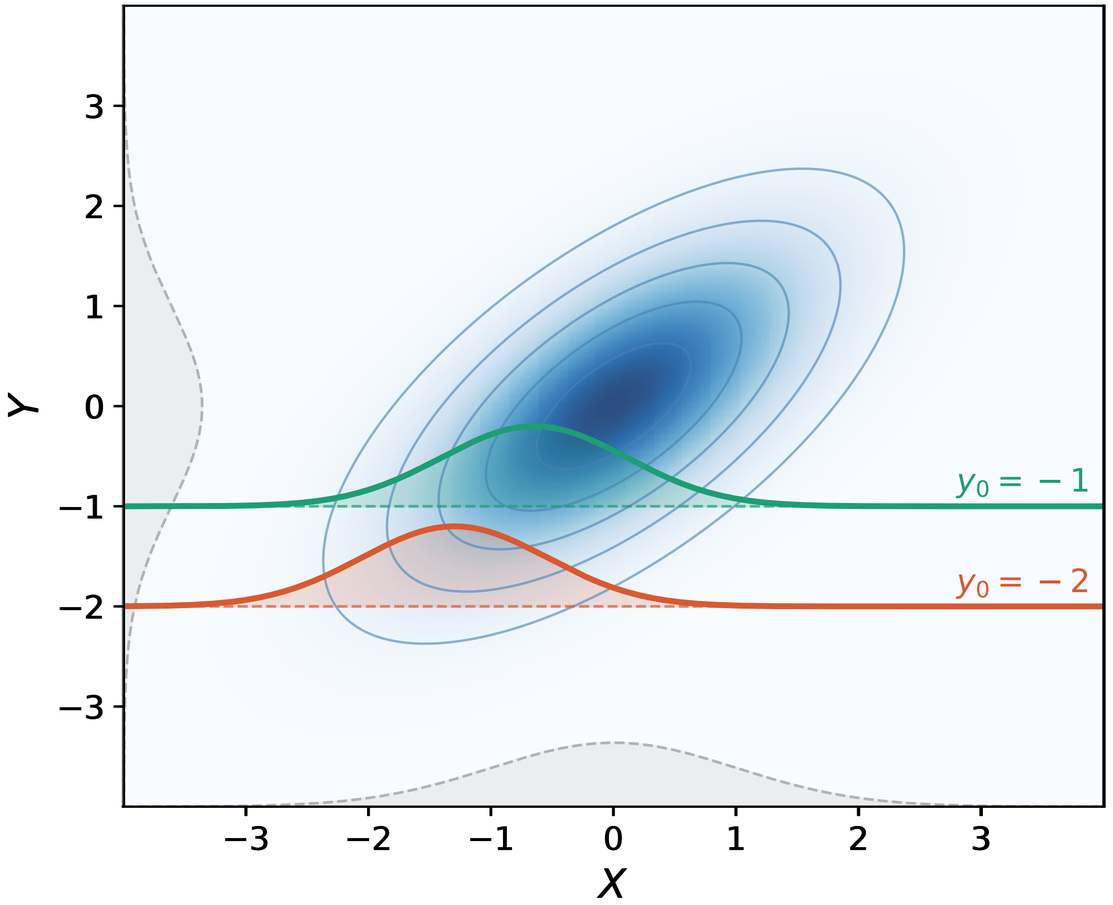
Conditional probability distributions

- **Definition**: in a probability space Ω, for two (non-empty) events $A, B \subset \Omega$,

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

- For **continuous variables** with:
  - joint probability distribution $x, y \sim p_{X,Y}$,
  - marginal distributions $x \sim p_X$ and $y \sim p_Y$,
  - the conditional distribution of "X given $Y = y$" is

$$p_{X|Y}(x \mid y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}$$



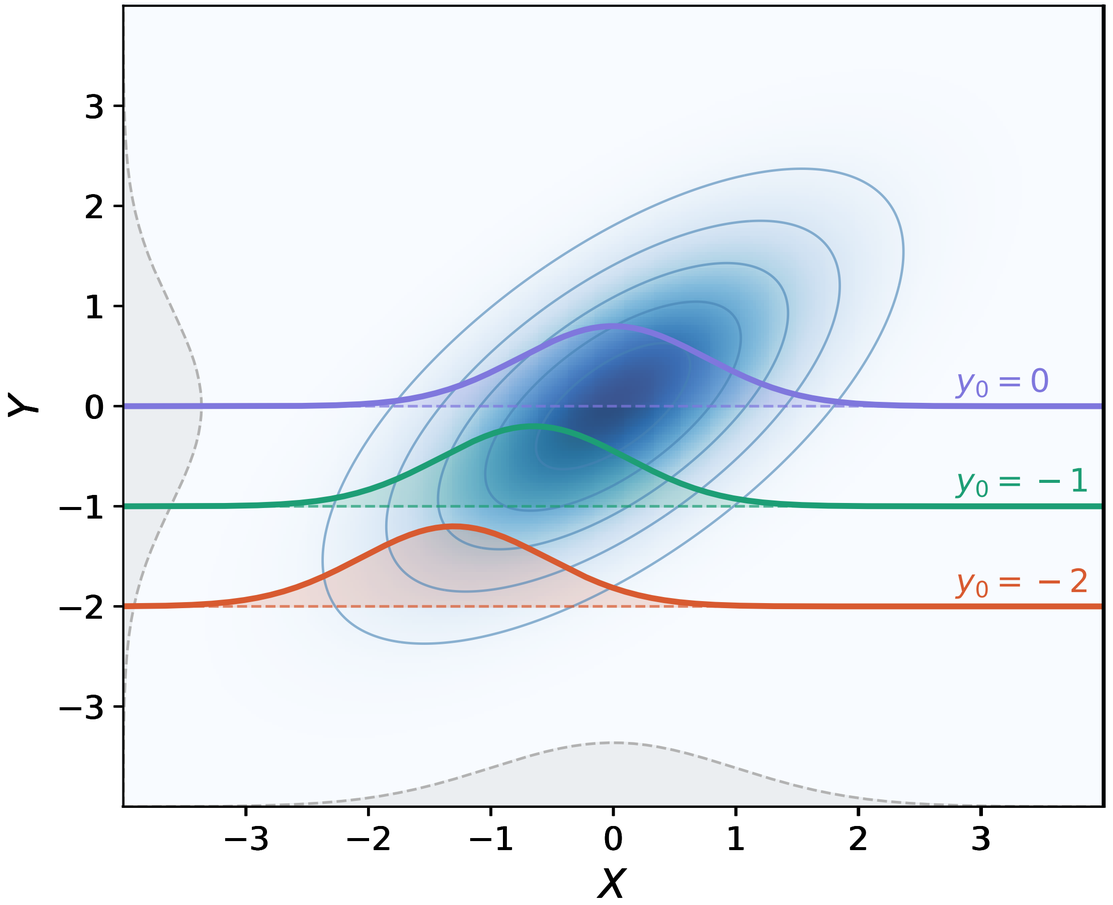
Conditional probability distributions

- **Definition**: in a probability space Ω, for two (non-empty) events $A, B \subset \Omega$,

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

- For **continuous variables** with:
  - joint probability distribution $x, y \sim p_{X,Y}$,
  - marginal distributions $x \sim p_X$ and $y \sim p_Y$,
  - the conditional distribution of "X given $Y = y$" is

$$p_{X|Y}(x \mid y) = \frac{p_{X,Y}(x, y)}{p_Y(y)}$$



Conditional probability distributions

Marginalisation

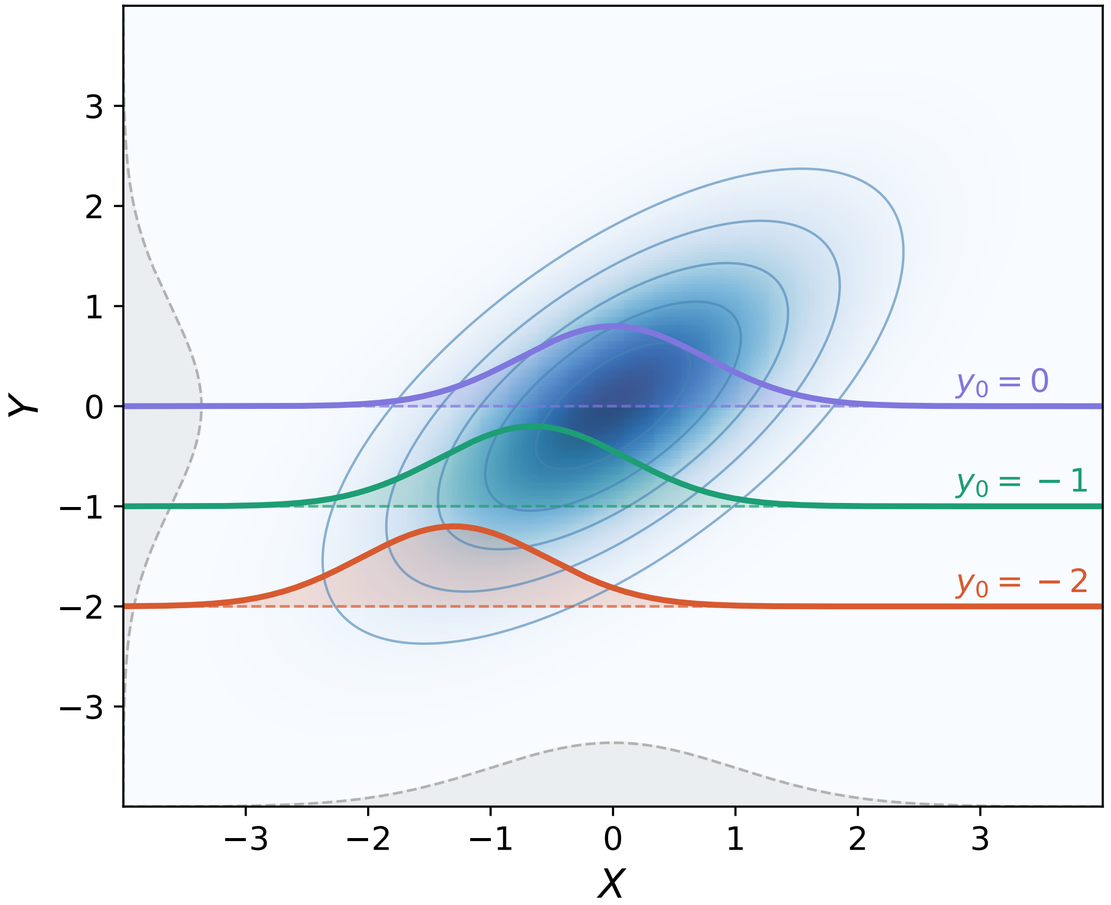
- For two continuous random variables x and y, we can go back and forth by integrating/marginalising:

$$p(x) = \int p(x, y)\mathrm{d}y = \int p(x \,|\, y)p(y)\mathrm{d}y$$

- But also with a third variable,

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

and $p(x \,|\, z) = \int p(x, y \,|\, z)p(y \,|\, z)\mathrm{d}y.$



Bayes' theorem

Thomas Bayes

- **Basic idea**: given a statistical model with likelihood function $P(d \mid \theta) = \mathscr{L}(d \mid \theta)$
  - Assume that parameters θ are themselves *random variables* 🤯
  - Their *marginal* distribution $P(\theta)$ represents our uncertainty *before* observing data
- **Bayes' theorem**
  - Let's apply conditional probability to the **joint probability space over parameters and data** (θ, d)

$$\begin{aligned} P(\theta, d) &= P(\theta \mid d) P(d) \\ &= P(d \mid \theta) P(\theta) \end{aligned}$$

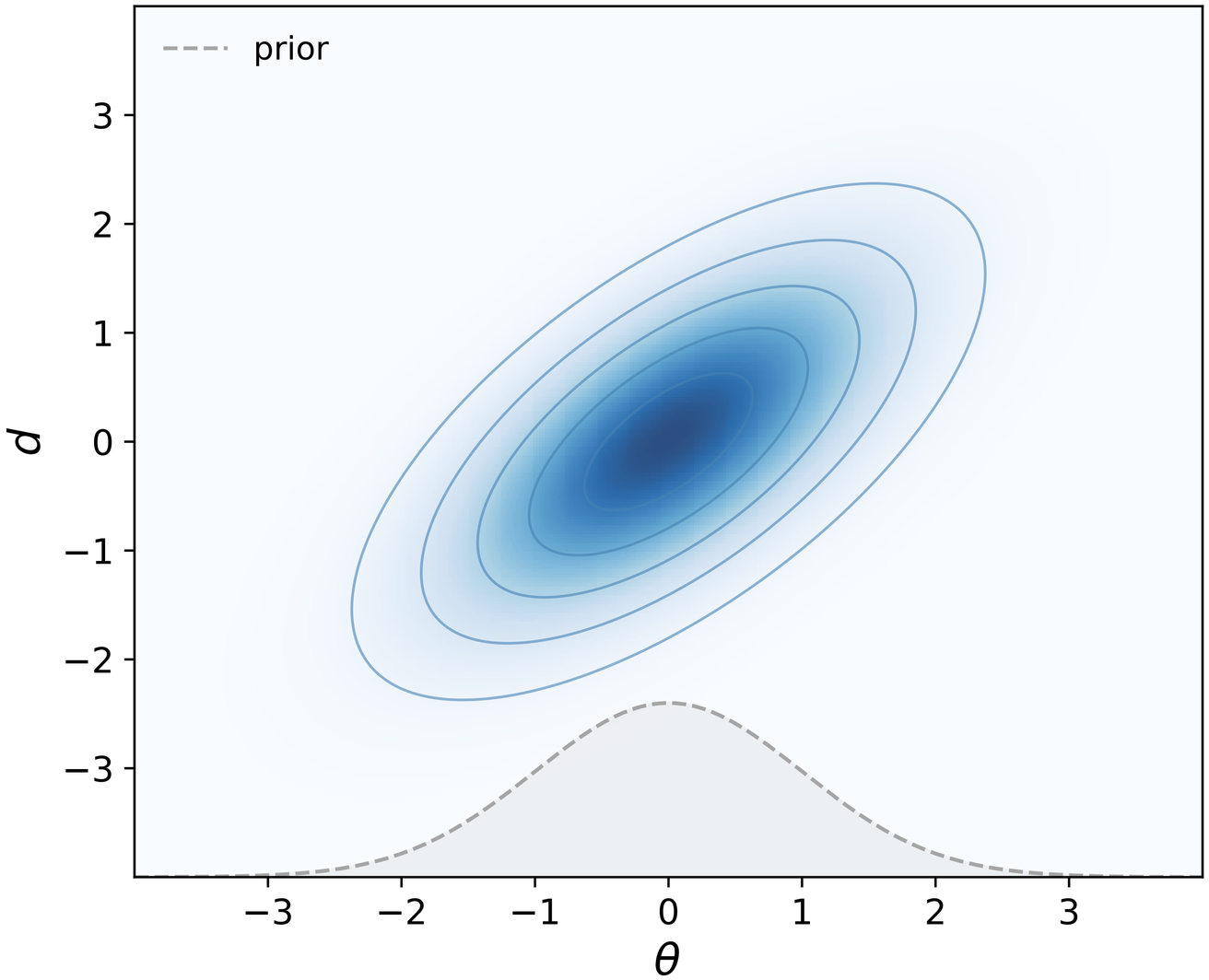
  - If follows that:

$$P(\theta \mid d) = \frac{P(d \mid \theta) P(\theta)}{P(d)}$$

posterior ← $P(\theta \mid d)$; likelihood ← $P(d \mid \theta)$; prior ← $P(\theta)$; evidence ← $P(d)$

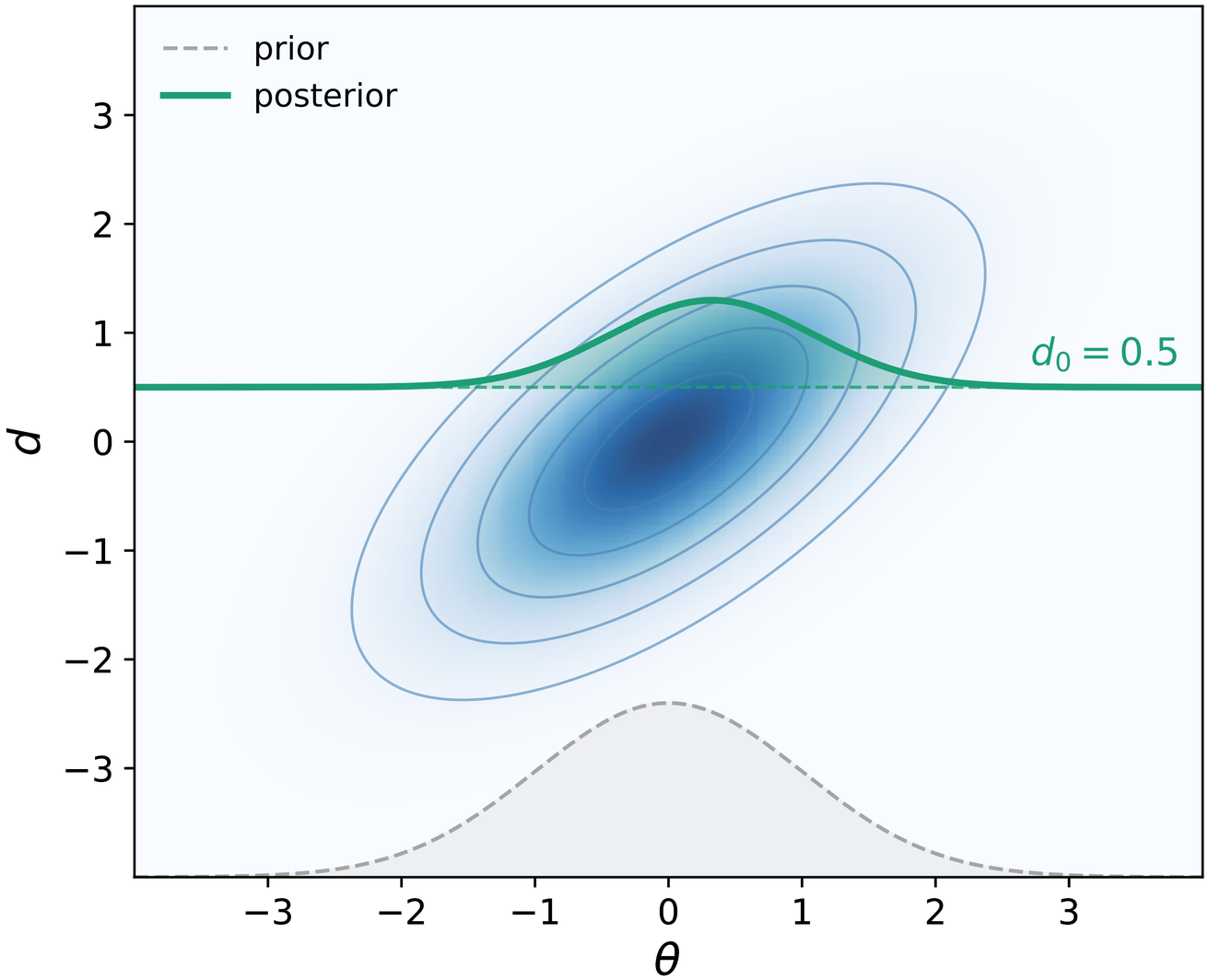
Bayes' theorem

Illustrated



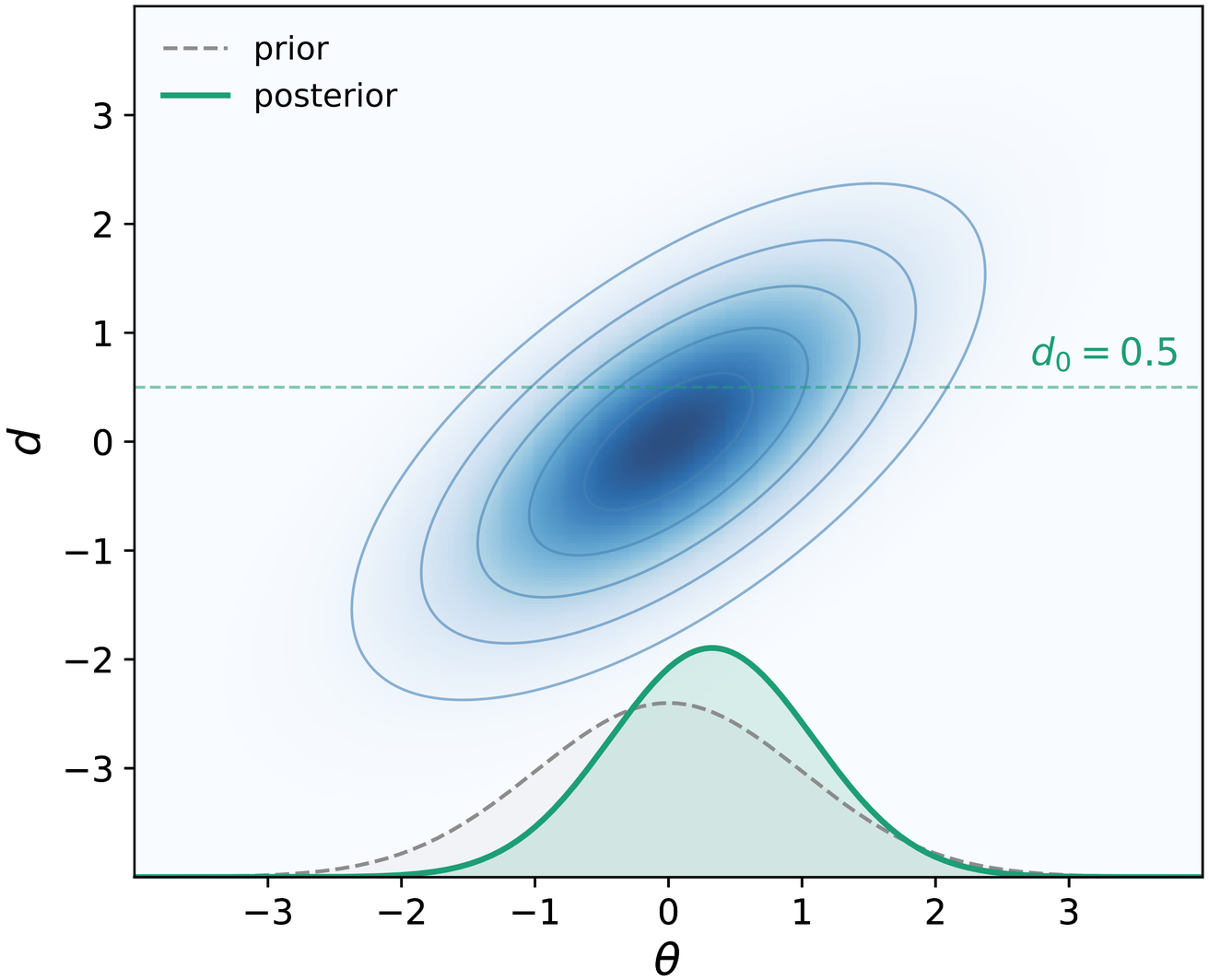
Bayes' theorem

Illustrated



Bayes' theorem

Illustrated



Bayes' theorem

Prior

- **The prior $P(\theta)$ represents our uncertainty on parameters, *before* observing data**
  - Flat/uniform prior : θ is somewhere in $[\theta_{\min}, \theta_{\max}]$, no preferred value *
  - Gaussian prior : $\theta \sim \mathcal{N}(\mu, \sigma)$ is around $\mu \pm \sigma$ — which you may know from some calibration experiment for a nuisance parameter
- **Priors are *parametrisation-dependent***
  - A uniform prior on θ^2 is *not* uniform on θ, even though the physical model itself is agnostic
  - Jeffreys prior $P(\theta) \propto |I(\theta)|^{1/2}$ are independent of parametrization, but hard to compute in general

* sometimes wrongly called "uninformative" prior

$I(\theta) = E\left(\frac{\mathrm{d}}{\mathrm{d}\theta} P(d\,|\,\theta)\right)$ is the Fisher information

Bayes' theorem

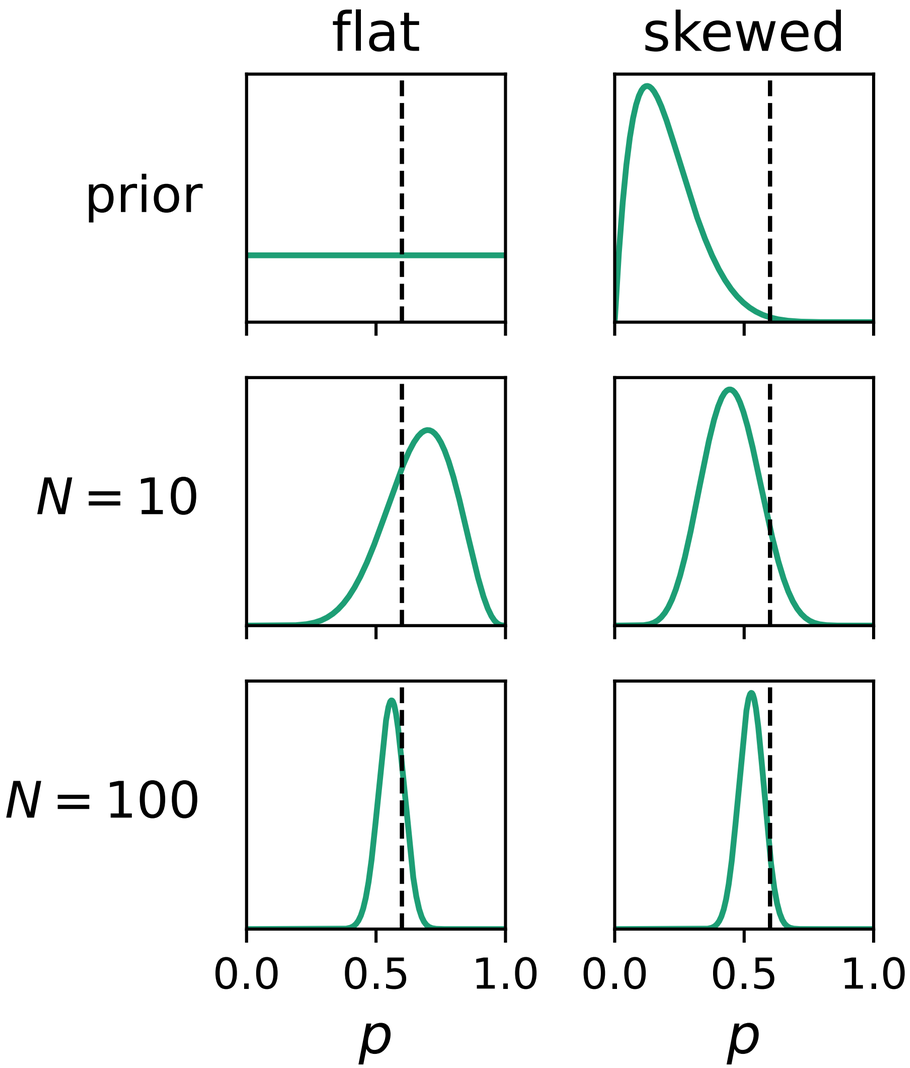
Posterior

- **Data d is used to *update* the parameter distribution from the prior $P(\theta)$ to the posterior $P(\theta \mid d)$**
  - If d_1 and d_2 are independent (two different experiments), then $P(\theta \mid d_1, d_2) \propto P(d_1, d_2 \mid \theta) P(\theta) \propto P(d_1 \mid \theta) P(d_2 \mid \theta) P(\theta)$

Bayes' theorem

Posterior

- **Data d is used to *update* the parameter distribution from the prior $P(\theta)$ to the posterior $P(\theta \mid d)$**
  - If d_1 and d_2 are independent (two different experiments), then $P(\theta \mid d_1, d_2) \propto P(d_1, d_2 \mid \theta) P(\theta) \propto P(d_1 \mid \theta) P(d_2 \mid \theta) P(\theta)$
- **Prior dependence**
  - The posterior *always* depends on the choice of prior…
    - However, if data is sufficiently constraining, the posterior will be much narrower than the prior, making the dependence weaker ("non-informative" prior).
    - If the data do not constrain a parameter, it may hit prior boundaries.



Posteriors for a coin toss (p=0.6) with different priors

Parameter constraints

Marginal parameter distributions

- **Definition**
  - For parameters $\theta = (\theta_1, \ldots, \theta_n)$, the *marginal* distribution of θ_i is obtained by integrating out (=marginalizing) all the *other* parameters:

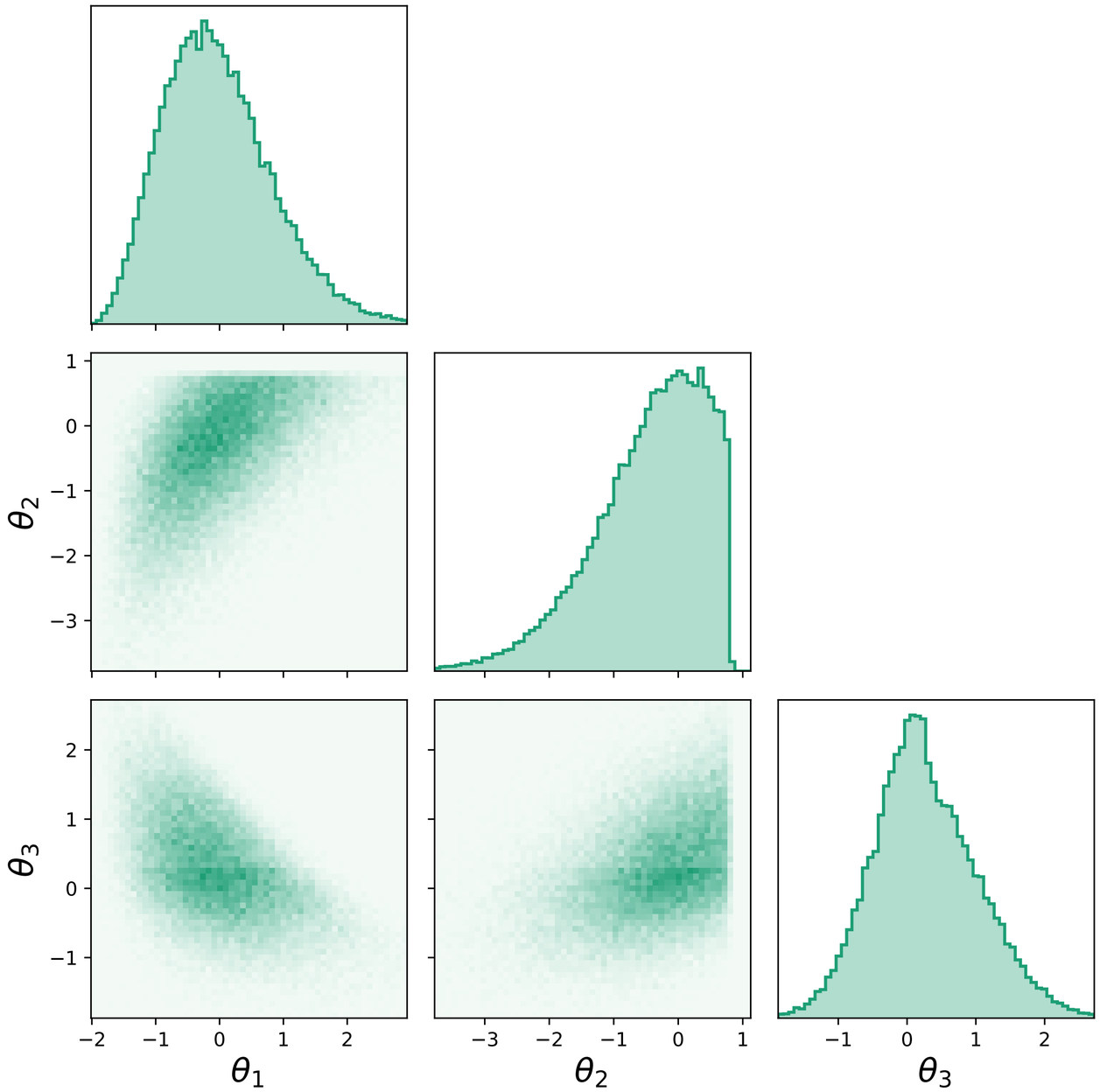
$$P(\theta_i \,|\, d) = \int P(\theta \,|\, d)\, \mathrm{d}\theta_1 \ldots \mathrm{d}\theta_{\mathrm{i}-1} \mathrm{d}\theta_{\mathrm{i}+1} \mathrm{d}\theta_{\mathrm{n}}$$

  - $P(\theta_i \,|\, d)$ includes *uncertainty on all other* parameters, e.g. nuisance parameters describing systematics! 💪
- **Estimation**
  - Given samples $\theta \sim P(\theta \,|\, d)$, $P(\theta_i \,|\, d)$ is approximated by the **histogram of θ_i from the samples** to arbitrary precision.

Parameter constraints

Corner plots

- Diagonal:**1D marginals**/histograms
- Lower corner: **2D marginals**/histograms
- Visualisation of *correlations* and *degeneracies* between parameter pairs
  (higher-dimensional information still hidden)

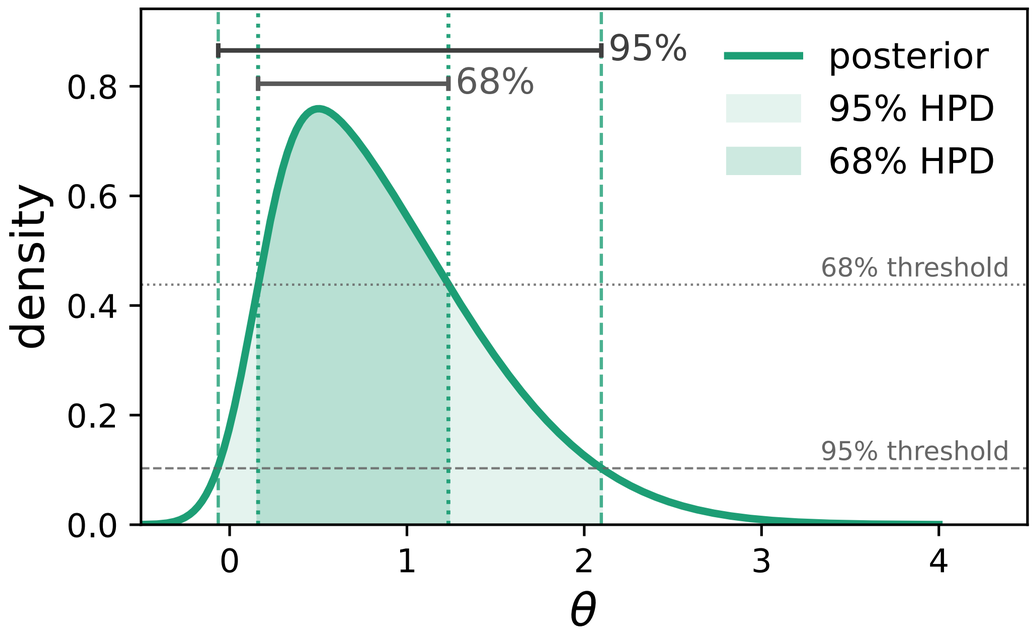


Parameter constraints

Credible intervals

- **Definition (not unique)**
  - Highest posterior density (HPD) interval/region is the smallest one having a given mass $1-\alpha$.
  - Formally, given by $C_\alpha(d) = \{\theta : P(\theta\,|\,d) > k_\alpha\}$ with threshold k_α such that

$$\int_{C_\alpha(d)} P(\theta\,|\,d)\mathrm{d}\theta = 1-\alpha.$$

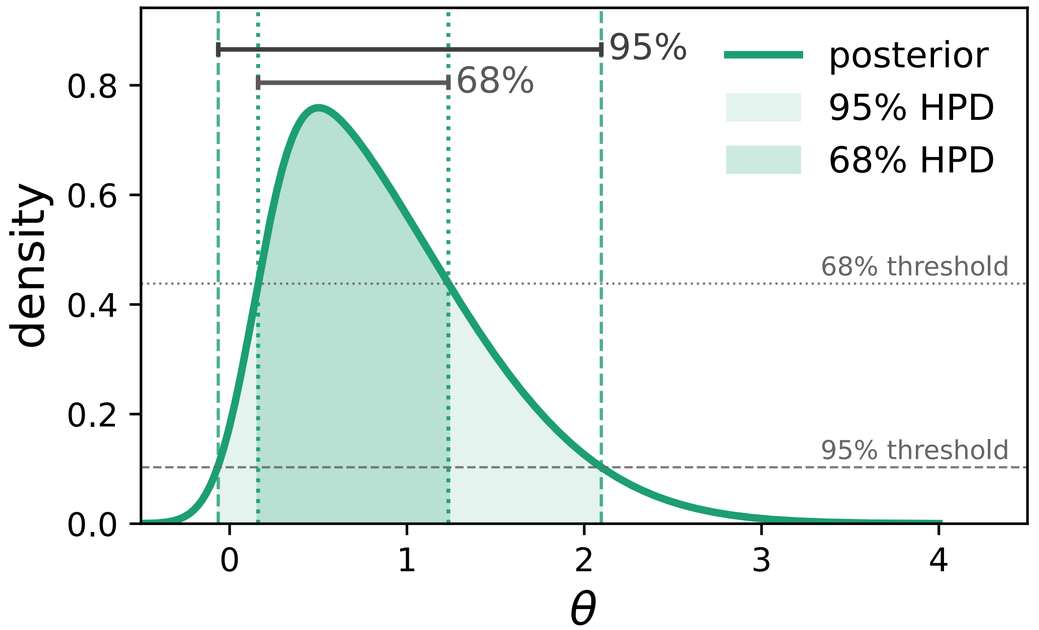


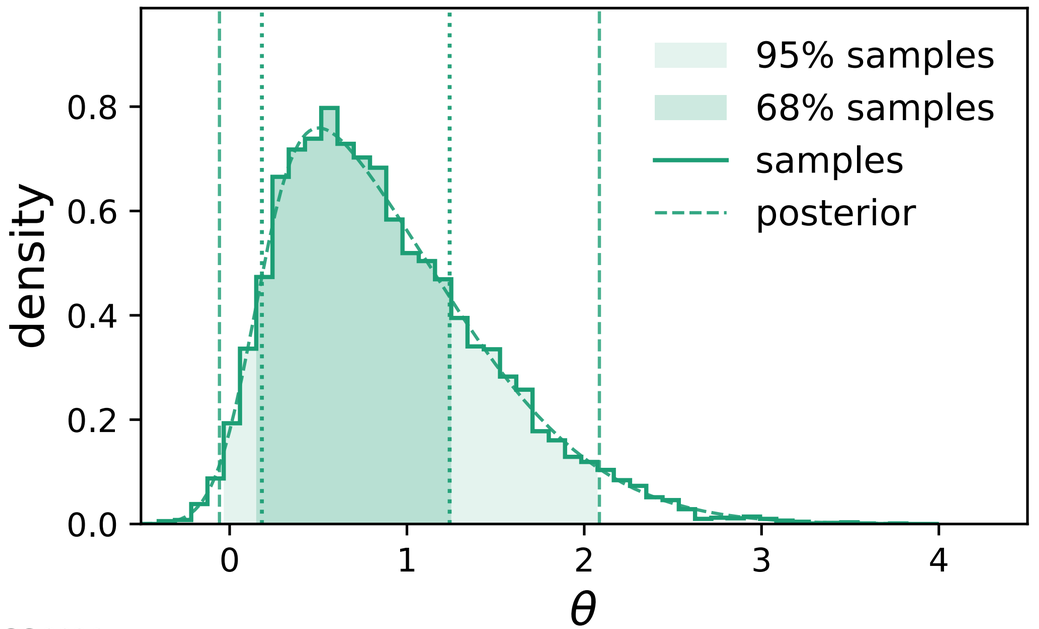
- **Constraints**
  - For a Gaussian, $C_\alpha = [\mu-\sigma, \mu+\sigma]$ for $\alpha = 68\,\%$ and we cite $\theta = \mu \pm \sigma$.
  - For an asymmetric distribution, $C_\alpha = [\theta_{\mathrm{low},\alpha}, \theta_{\mathrm{high},\alpha}]$ and we typically cite $\theta = \theta_{\mathrm{med}}{}^{+(\theta_{\mathrm{high}}-\theta_{\mathrm{med}})}_{-(\theta_{\mathrm{med}}-\theta_{\mathrm{low}})}$.

Parameter constraints

Credible intervals

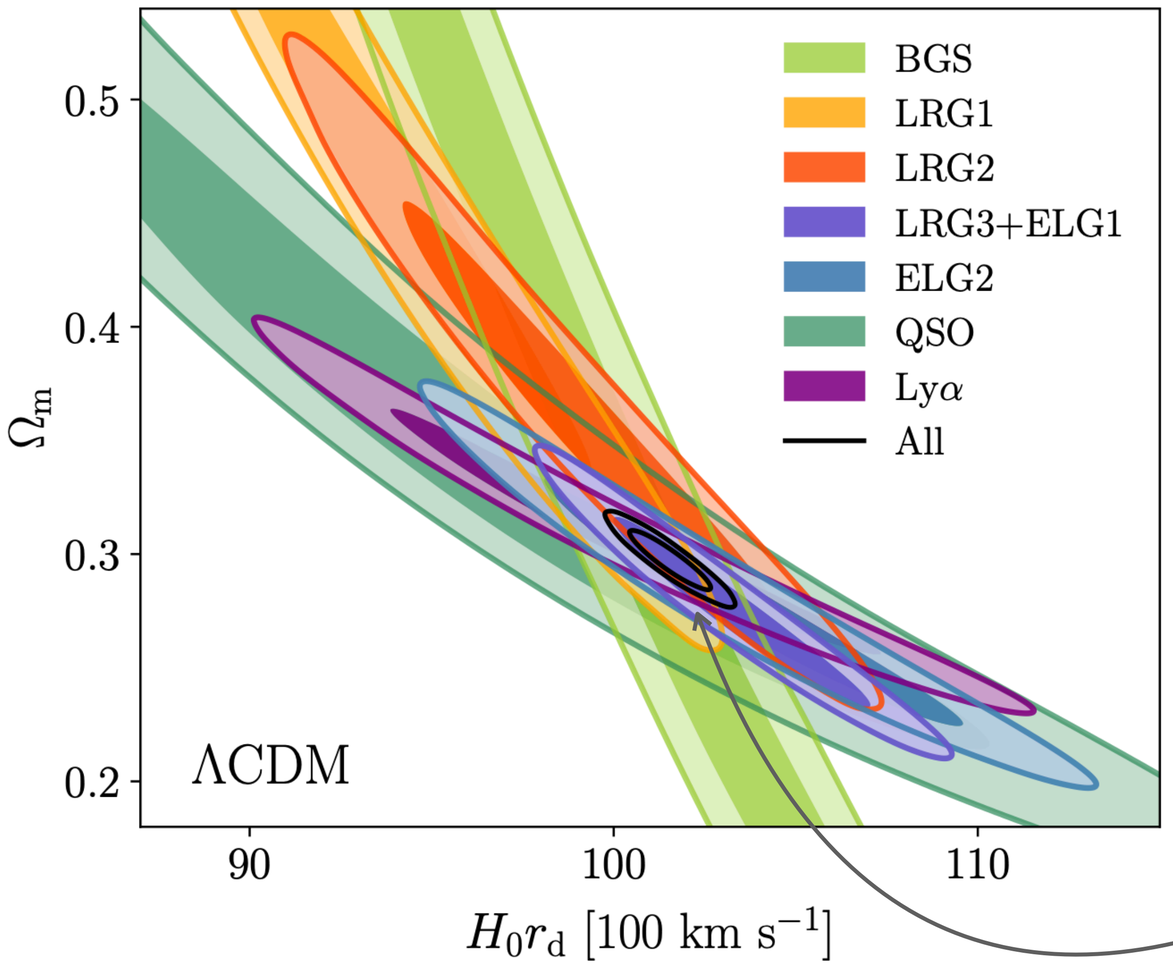
- **Computing from samples**
  - Given samples, the 68% *credible* interval is the region containing the 68% highest-probability samples.
  - Algorithm in 1D for n samples, $m = \lfloor \alpha n \rfloor$
    1. sort samples $\theta^{(0)} < \ldots < \theta^{(n)}$,
    2. compute widths of all windows $[\theta^{(j)}, \theta^{(j+m)}]$ with 68% of samples,
    3. keep shortest one. 😎



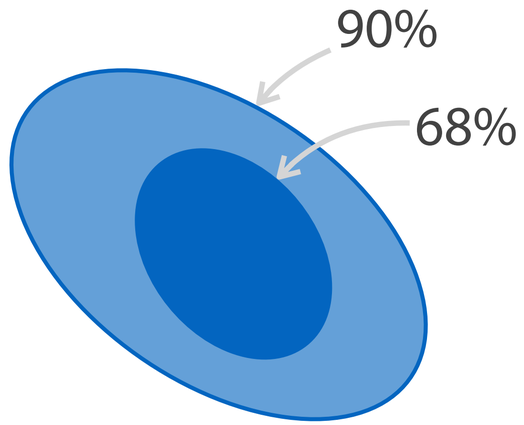


Parameter constraints

Credible regions



1. How to read each 2D marginal posterior:



2. Posteriors show different degeneracies between the two parameters.

3. Combined posterior from independent measurements: broken degeneracy!

DESI DR2 BAO results : arXiv:2503.14738

Parameter constraints

Credible intervals: coverage

What if I repeat an experiment at fixed parameter values θ_0? Do credible intervals (Bayesian) for each realisations have guarantees to contain that value, like confidence intervals (frequentist) do?

- Averaged over the prior, yes: $\int \underbrace{P_{d\sim P(\cdot|\theta)}(\theta \in C_\alpha(d))}_{\text{freq.coverage}} P(\theta)\mathrm{d}\theta = 1-\alpha$. But that's not very useful.
- Special case of Gaussian model + flat prior on μ: HPD coincide with frequentist confidence intervals, so yes again.
- Asymptotically: for data $d_1, \ldots, d_n$, posteriors converge to $P(\theta \,|\, d_1, \ldots, d_n) \approx \mathcal{N}\left(\hat{\theta}_n, I(\theta_0)^{-1}/n\right)$, so credible intervals acquire frequentist coverage.
- Generally (low n, non-Gaussian, non-linear): no guarantee. You're on your own.

Model comparison

Evidence

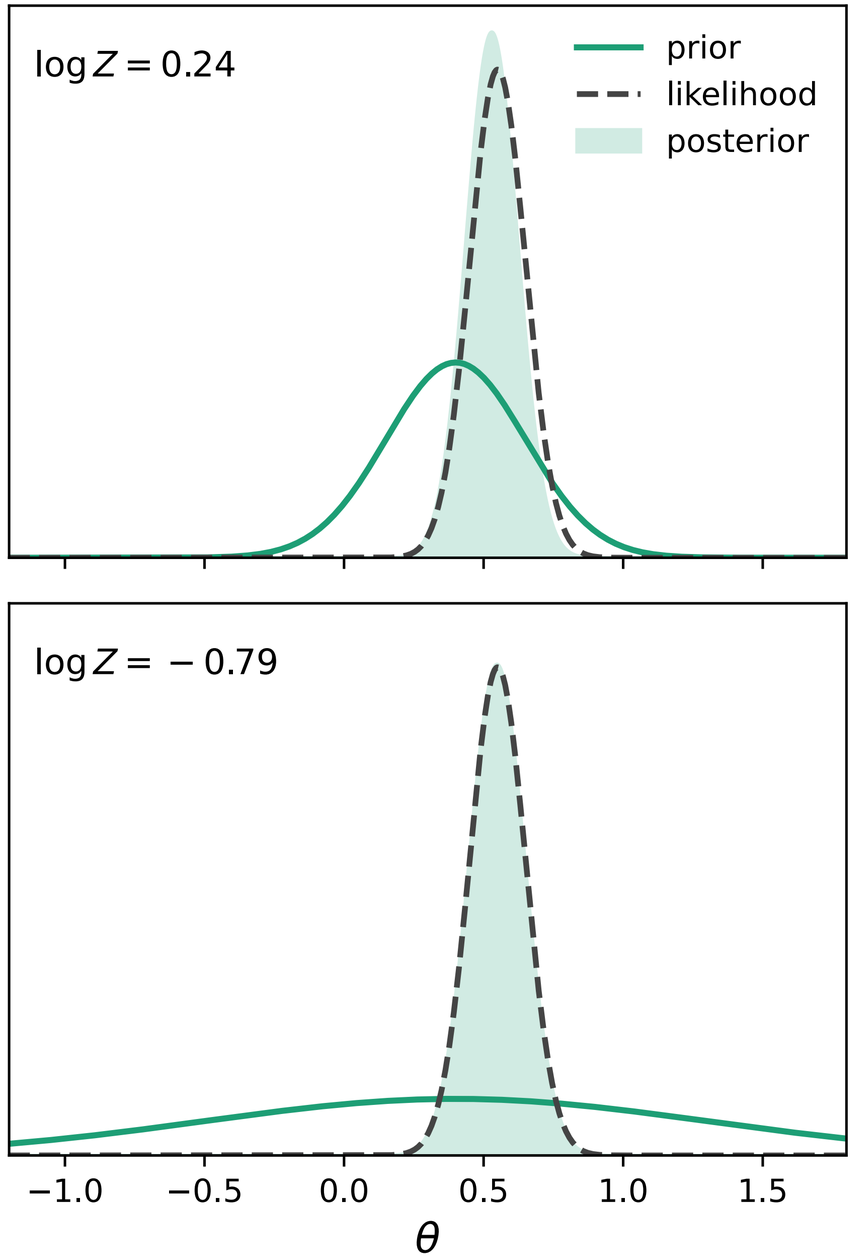
- **Posterior normalisation**

$$P(\theta \mid d) = \frac{P(d \mid \theta) P(\theta)}{P(d)}$$

  - Given data, $P(d)$ (often denoted Z) is fixed and is a normalisation for the posterior
  - All we need for inference is $P(\theta \mid d) \propto P(d \mid \theta) P(\theta)$!

- **What is encodes**

  - $P(d) = \int P(d \mid \theta) P(\theta) \mathrm{d}\theta$ is average of the likelihood over the full prior.



Model comparison

Bayes factors

think of them as

- Given two models M_1, M_2 and corresponding evidences $Z_i \equiv P_{M_i}(d) \sim P(d \mid M_i)$, write

$$\frac{P(M_1 \mid d)}{P(M_2 \mid d)} = \frac{Z_1}{Z_2}\frac{P(M_1)}{P(M_2)} \text{ and define the } \textbf{Bayes factor } B_{12} \equiv \frac{Z_1}{Z_2}$$

- Evidences average the likelihood over the prior: a model is penalised if it wastes prior volume on low-likelihood regions — automatic **Occam's razor**.
- Assuming models have equal probability $P(M_1) = P(M_2)$ (why not), evidences can be used to compare the likeliness of two models with the **Jeffrey scale**:

$\lvert \ln B_{12} \rvert$	Odds	Strength
0 – 1	< 3 : 1	Inconclusive
1 – 2.5	3 – 12 : 1	Weak / moderate
2.5 – 5	12 – 150 : 1	Strong
> 5	> 150 : 1	Decisive

Evidence is prior-dependent though… 🤔

Frequentist *vs* Bayesian approaches: wrapping up

- **Same key ingredient**: the likelihood $\mathscr{L}(d\,|\,\theta) \equiv P(d\,|\,\theta)$ is the distribution of data given parameter values.

Frequentist *vs* Bayesian approaches: wrapping up

- **Same key ingredient**: the likelihood $\mathscr{L}(d\,|\,\theta) \equiv P(d\,|\,\theta)$ is the distribution of data given parameter values.

Frequentist

- Assume there exists a true θ_0 value at which data were realised $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$
- Use likelihood (ratios) to compute confidence intervals $I_\alpha(d)$, such that $P_d\big(\theta_0 \in I_\alpha(d)\big) \geq 1-\alpha$
  - For many data realizations d generated at θ_0, $I_{68\%}(d)$ contains θ_0 for 68% of realizations
  - $I_{68\%}(d)$ is difficult to derive in general (requires approx like Wilk's at finite n), even more for ≥2D

Bayesian

- Assume θ are themselves random variables
- Use likelihood to update the parameter distribution from $P(\theta)$ to $P(\theta\,|\,d)$ (by sampling)
- Compute credible intervals, such that $P_{\theta|d}\big(\theta \in I_\alpha(d)\,|\,d\big) \geq 1-\alpha$
  - Very easy to compute given posterior samples (just histograms)…
  - … but no *strict* guarantee to contain "true" θ in general (only asymptotically or special cases)

Frequentist *vs* Bayesian approaches: wrapping up

- **Same key ingredient**: the likelihood $\mathscr{L}(d\,|\,\theta) \equiv P(d\,|\,\theta)$ is the distribution of data given parameter values.

Frequentist

- Assume there exists a true θ_0 value at which data were realised $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$
- Use likelihood (ratios) to compute confidence intervals $I_\alpha(d)$, such that $P_d\big(\theta_0 \in I_\alpha(d)\big) \geq 1-\alpha$
  - For many data realizations d generated at θ_0, $I_{68\%}(d)$ contains θ_0 for 68% of realizations
  - $I_{68\%}(d)$ is difficult to derive in general (requires approx like Wilk's at finite n), even more for ≥2D

Bayesian

- Assume θ are themselves random variables
- Use likelihood to update the parameter distribution from $P(\theta)$ to $P(\theta\,|\,d)$ (by sampling)
- Compute credible intervals, such that $P_{\theta|d}\big(\theta \in I_\alpha(d)\,|\,d\big) \geq 1-\alpha$
  - Very easy to compute given posterior samples (just histograms)…
  - … but no *strict* guarantee to contain "true" θ in general (only asymptotically or special cases)

*The honest reasons to pick one over the other are **practicability** and **habits** within a research field.*

Cool. Btw, how do we *sample* the posterior?

Sampling the posterior

Naive methods

- **Grid evaluation (no sampling)**
  - Compute $P(\theta \mid d) \propto P(d \mid \theta) P(\theta)$ on a grid of θ values: N^{dim} evaluations
  - Intractable for dimension > 2: too many evaluations

Sampling the posterior

Naive methods

- **Grid evaluation (no sampling)**
  - Compute $P(\theta \mid d) \propto P(d \mid \theta)P(\theta)$ on a grid of θ values: N^{dim} evaluations
  - Intractable for dimension > 2: too many evaluations
- **Rejection sampling**
  - Generate prior samples $\theta \sim P(\theta)$, compute posterior $P(\theta \mid d) \propto P(d \mid \theta)P(\theta)$
  - Keep samples with probability $P(\theta \mid d) / \max(P(\theta \mid d))$
  - Very inefficient for dimension > few because that ratio decreases by orders of magnitude away from the max

Sampling the posterior

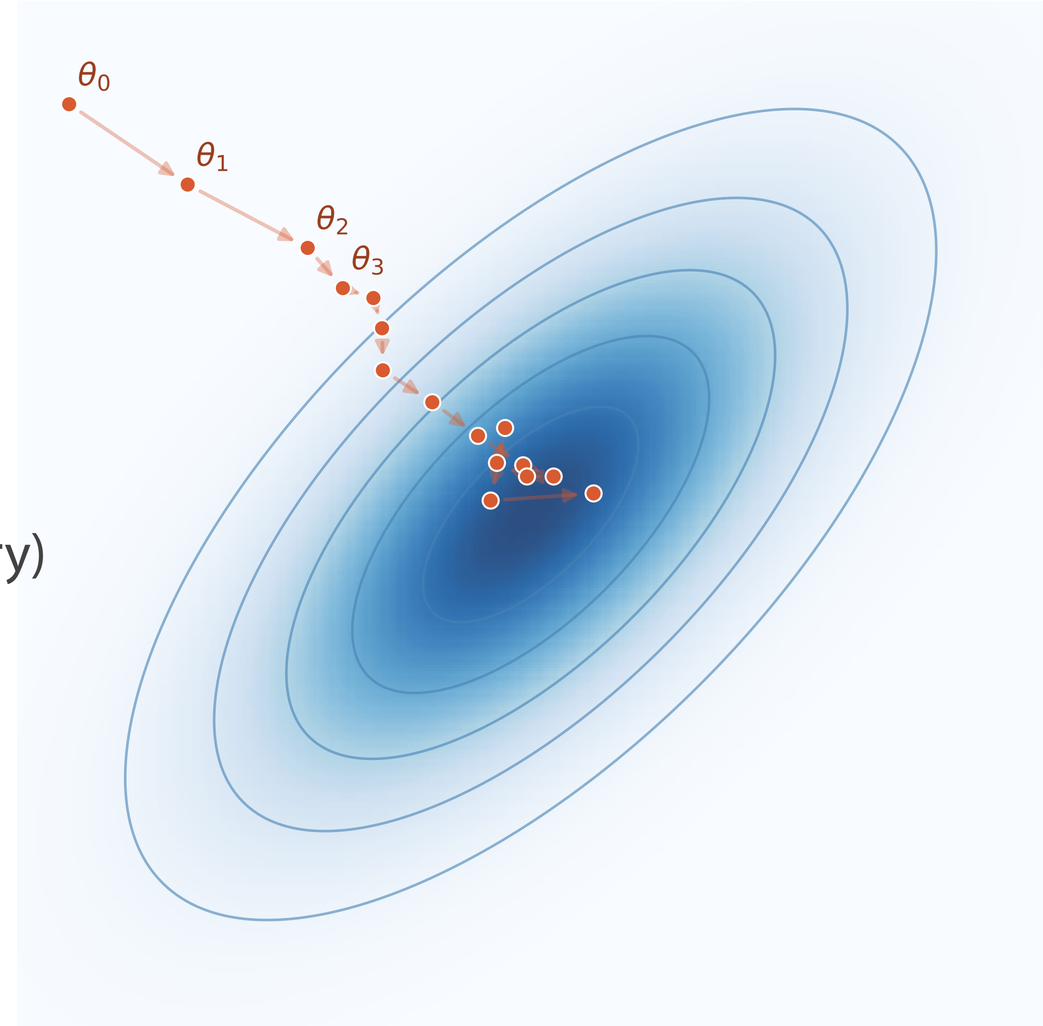
Naive methods

- **Grid evaluation (no sampling)**
  - Compute $P(\theta \mid d) \propto P(d \mid \theta)P(\theta)$ on a grid of θ values: $N^{\dim}$ evaluations
  - Intractable for dimension > 2: too many evaluations
- **Rejection sampling**
  - Generate prior samples $\theta \sim P(\theta)$, compute posterior $P(\theta \mid d) \propto P(d \mid \theta)P(\theta)$
  - Keep samples with probability $P(\theta \mid d) / \max(P(\theta \mid d))$
  - Very inefficient for dimension > few because that ratio decreases by orders of magnitude away from the max
- **Importance sampling**
  - Same, but instead weigh all samples by $P(\theta \mid d)/P(\theta)$
  - Same limitations

MCMC sampling

Markov chains

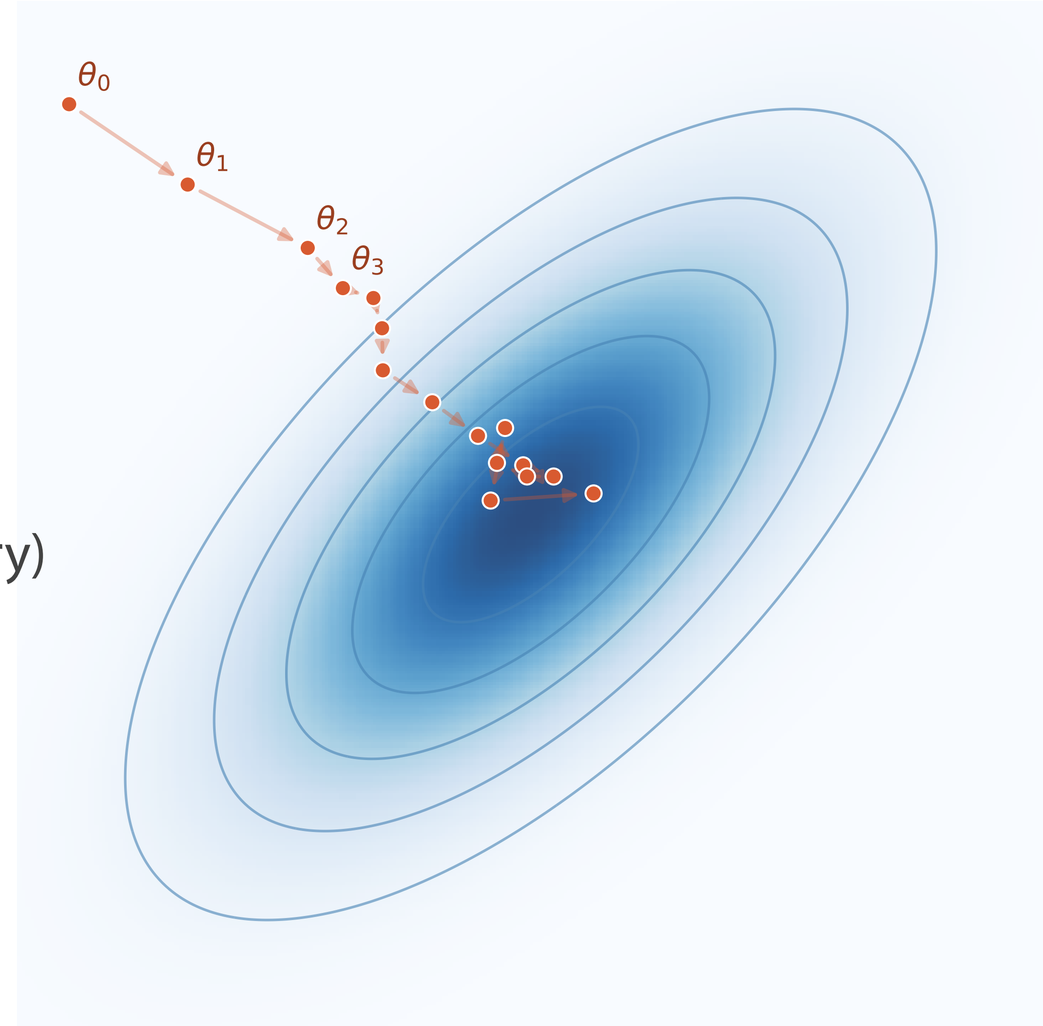
- **Definition**
  - Discrete stochastic process $\{\theta_i\}_{i\in\mathbb{N}}$ such that
    $P(\theta_{i+1}\,|\,\theta_0,\ldots,\theta_i) = P(\theta_{i+1}\,|\,\theta_i) \equiv T(\theta_{i+1}\,|\,\theta_i)$
  - Think: random walk in parameter space where each step $\theta_i \to \theta_{i+1}$ only depends on current position θ_i (no memory)



MCMC sampling

Markov chains

- **Definition**
  - Discrete stochastic process $\{\theta_i\}_{i\in\mathbb{N}}$ such that
    $P(\theta_{i+1}\,|\,\theta_0, \ldots, \theta_i) = P(\theta_{i+1}\,|\,\theta_i) \equiv T(\theta_{i+1}\,|\,\theta_i)$
  - Think: random walk in parameter space where each step $\theta_i \to \theta_{i+1}$ only depends on current position θ_i (no memory)



- **Theorem**

  A Markov chain that is well-behaved, i.e.
  - irreducible (can reach any value in the space),
  - aperiodic (cannot be stuck in a space subset),
  - positive recurrent (can revisit all the steps),

  converges to a *unique* stationary distribution π such that $T\pi = \pi$ (transition doesn't change π).

MCMC sampling

Sampling with Markov chains: Markov chain Monte-Carlo

MCMC sampling

Sampling with Markov chains: Markov chain Monte-Carlo

- **Markov chains**
  - Markov chain defined by its transition probability $T(\theta_{i+1} \,|\, \theta_i) \equiv P(\theta_{i+1} \,|\, \theta_i)$
  - Can we chose $T(\theta_{i+1} \,|\, \theta_i)$ such that stationary distribution = posterior $P(\theta \,|\, d)$ (*target* distribution)?

MCMC sampling

Sampling with Markov chains: Markov chain Monte-Carlo

- **Markov chains**
  - Markov chain defined by its transition probability $T(\theta_{i+1} \,|\, \theta_i) \equiv P(\theta_{i+1} \,|\, \theta_i)$
  - Can we chose $T(\theta_{i+1} \,|\, \theta_i)$ such that stationary distribution = posterior $P(\theta \,|\, d)$ (*target* distribution)?
- **MCMC**
  - Yes! This is ensured if the *detailed balance equation* is verified

$$T(\theta_{i+1} \,|\, \theta_i) P(\theta_i \,|\, d) = T(\theta_i \,|\, \theta_{i+1}) P(\theta_{i+1} \,|\, d)$$

  - Starting anywhere in space, the random walk will provide samples of the stationary distribution, i.e. the posterior

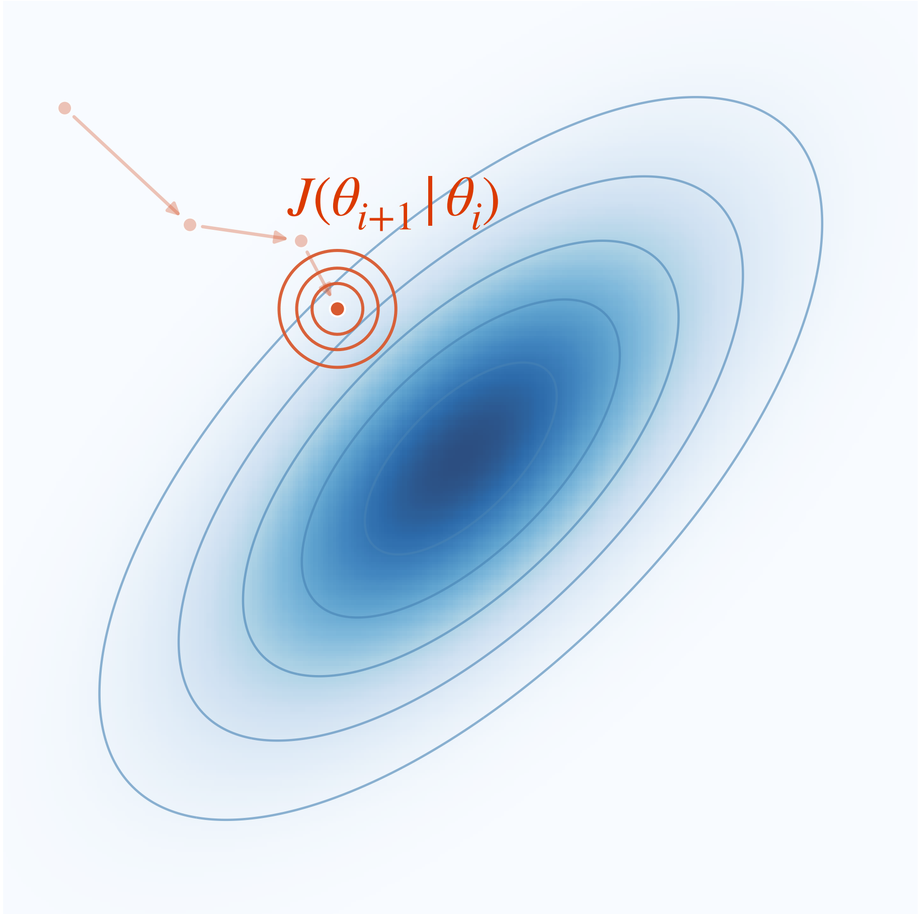
MCMC sampling

Metropolis-Hastings algorithm

- Chose a proposal distribution (jump) $J(\theta_{i+1} \mid \theta_i)$,
  for instance $J(\theta_{i+1} \mid \theta_i) = \mathcal{N}(\mu = \theta_i, \Sigma)$
- At step i,
  - Sample $\theta_{i+1} \sim J(\theta_{i+1} \mid \theta_i)$
  - Compute ratio

$$r = \frac{P(\theta_{i+1} \mid d) J(\theta_i \mid \theta_{i+1})}{P(\theta_i \mid d) J(\theta_{i+1} \mid \theta_i)}$$

    - If $r \geq 1$, accept proposal θ_{i+1}
    - If $r < 1$, accept proposal θ_{i+1} with probability r, otherwise set $\theta_{i+1} = \theta_i$

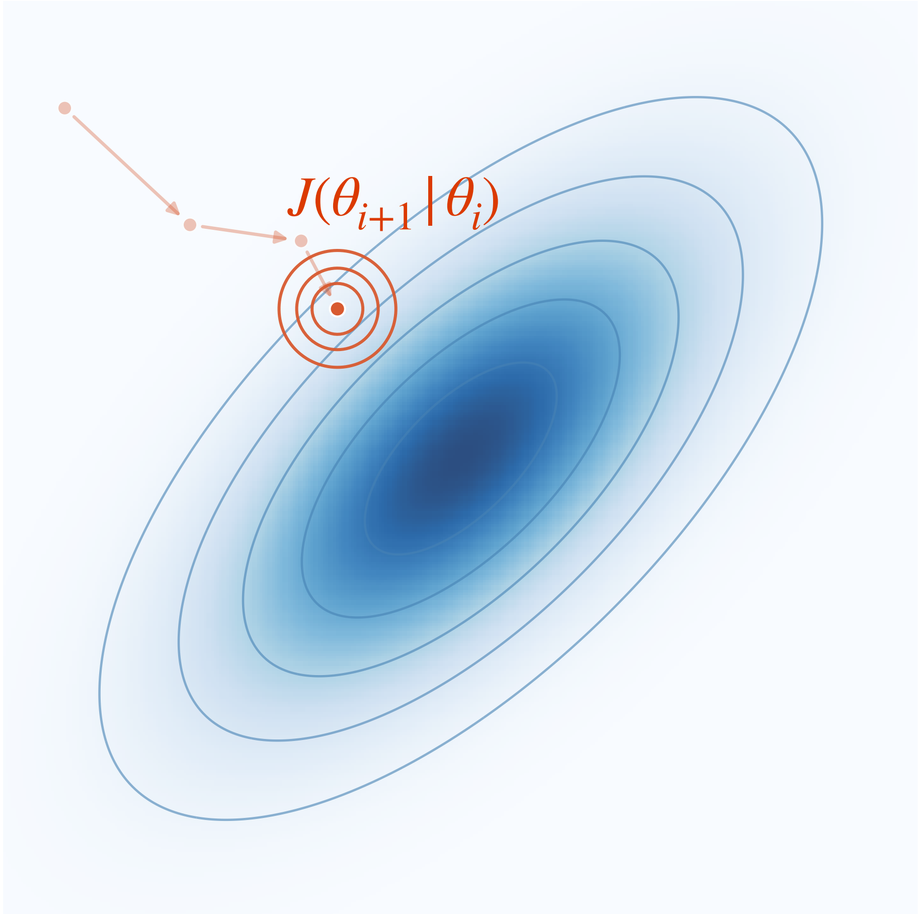


See https://chi-feng.github.io/mcmc-demo/app.html

MCMC sampling

Metropolis-Hastings algorithm

- Chose a proposal distribution (jump) $J(\theta_{i+1} \mid \theta_i)$,
  for instance $J(\theta_{i+1} \mid \theta_i) = \mathcal{N}(\mu = \theta_i, \Sigma)$
- At step i,
  - Sample $\theta_{i+1} \sim J(\theta_{i+1} \mid \theta_i)$
  - Compute ratio
    $$r = \frac{P(\theta_{i+1} \mid d) J(\theta_i \mid \theta_{i+1})}{P(\theta_i \mid d) J(\theta_{i+1} \mid \theta_i)}$$
  - Transition probability $T(\theta_{i+1} \mid \theta_i) = J(\theta_{i+1} \mid \theta_i) \min(1, r)$
- The set of *accepted* samples will converge to $P(\theta \mid d)$



See https://chi-feng.github.io/mcmc-demo/app.html

MCMC sampling

Final samples

- The initialization of the Markov chains is done by random draws within the priors

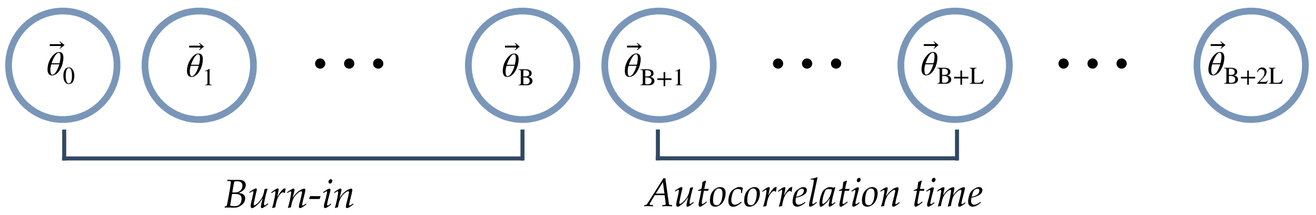
  → The first parameter values may be very far from the stationary distribution

  Burn-in: remove first samples from the Markov chains to "forget" the influence of the initial position

- By definition of a Markov process, two successive samples are highly correlated

  → We cannot keep all samples after the burn-in if we only want independent samples

  Autocorrelation time: separation to consider between 2 samples for them to be nearly independent
  (study of the autocorrelation function of the chains)



MCMC sampling

Convergence / Gelman-Rubin test

- In theory, the stationary distribution is only reached asymptotically by a Markov chain (*infinite time*)

  → Use of convergence tests to stop sampling of the parameter space

- Consider a set of M Markov chains from which the burn-in has been removed, then containing N samples

- **Gelman-Rubin test**

  1. Compute the mean of each chain j: $\bar{y}^j = \frac{1}{N}\sum_i y_i^j$
  2. Compute the mean of all samples: $\bar{y} = \frac{1}{NM}\sum_{ij} y_i^j$
  3. Define the variance between chains by: $B = \frac{1}{M-1}\sum_j (\bar{y}^j - \bar{y})^2$
  4. Compute the mean of the variances of each chain: $W = \frac{1}{M}\sum_j \left[\frac{1}{N-1}\sum_i (y_i^j - \bar{y}^j)^2\right]$

     (estimator that tends to 0 when the chains have converged)
  5. Define the quantity: $R = \frac{\frac{N-1}{N}W + \frac{M+1}{M}B}{W}$

Since $R \underset{t\to\infty}{\longrightarrow} 1$, we define a threshold close to 1 below which the chains are considered to have converged

Hands-on 1

Implement Metropolis-Hastings and diagnostics

```
    chi2 = (y - y_model)**2 / (yerr**2)
    return np.sum(-chi2 / 2)

def log_prior(theta):
    if all(theta > theta_low) and all(theta < theta_high):
        return 0
    return -np.inf

def log_posterior(theta, x, y, yerr):
    lp = log_prior(theta)
    if np.isfinite(lp):
        lp += log_likelihood(theta, x, y, yerr)
    return lp


# create a small ball around the MLE the initialize each walker
nwalkers, ndim = 30, 5
theta_guess = [0.5, 0.6, 0.2, -0.2, 0.1]
pos = theta_guess + 1e-4 * np.random.randn(nwalkers, ndim)

# run emcee
sampler = emcee.EnsembleSampler(nwalkers, ndim, log_posterior, args=(x, y, y_err))
sampler.run_mcmc(pos, 10000, progress=True);
```

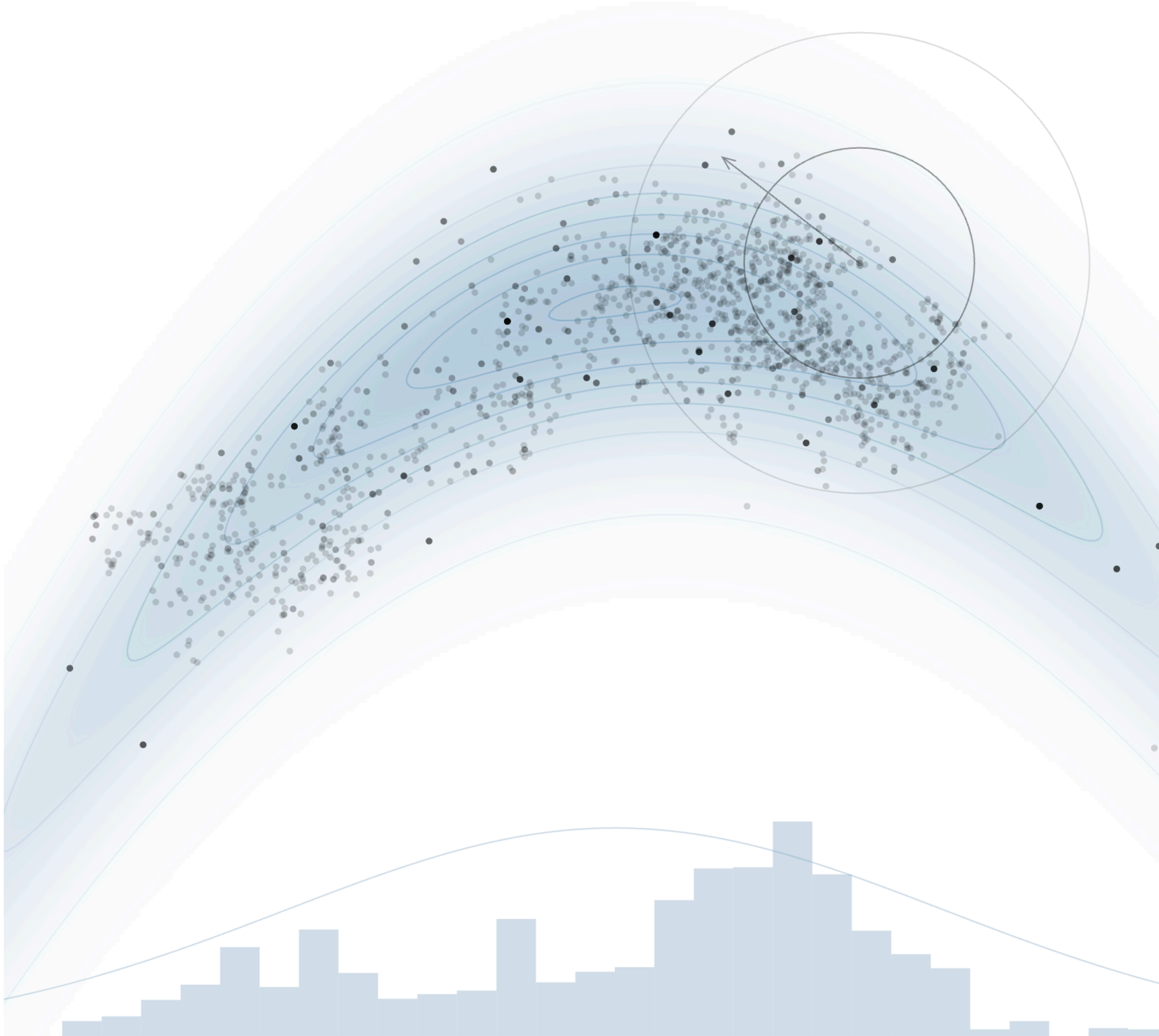
```
100%|██████████| 10000/10000 [00:45<00:00, 220.62it/s]
```

```
fig, axes = plt.subplots(ndim, sharex=True)
mcmc_samples = sampler.get_chain()
labels = ["l", "m", "s", "a", "b"]
for i in range(ndim):
    ax = axes[i]
    ax.plot(mcmc_samples[:, :, i], "k", alpha=0.3, rasterized=True)
    ax.set_xlim(0, 1000)
    ax.set_ylabel(labels[i])

axes[-1].set_xlabel("step number");
```

Beyond MCMC

Modern sampling techniques

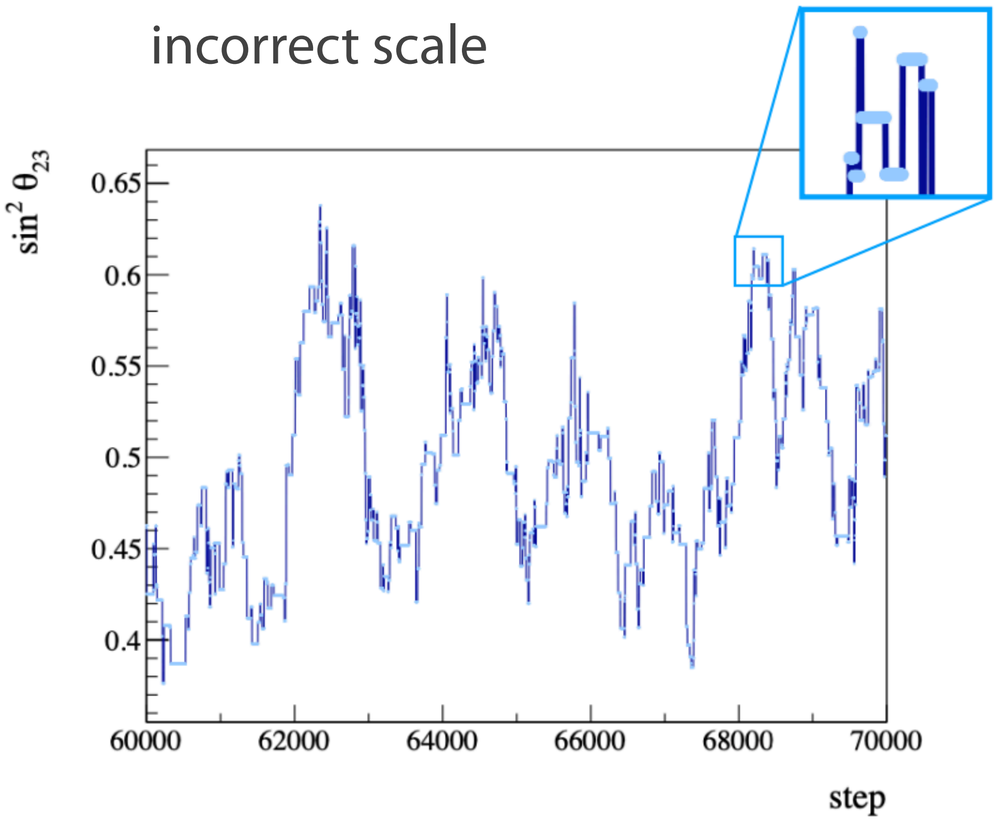


Ensemble MCMC

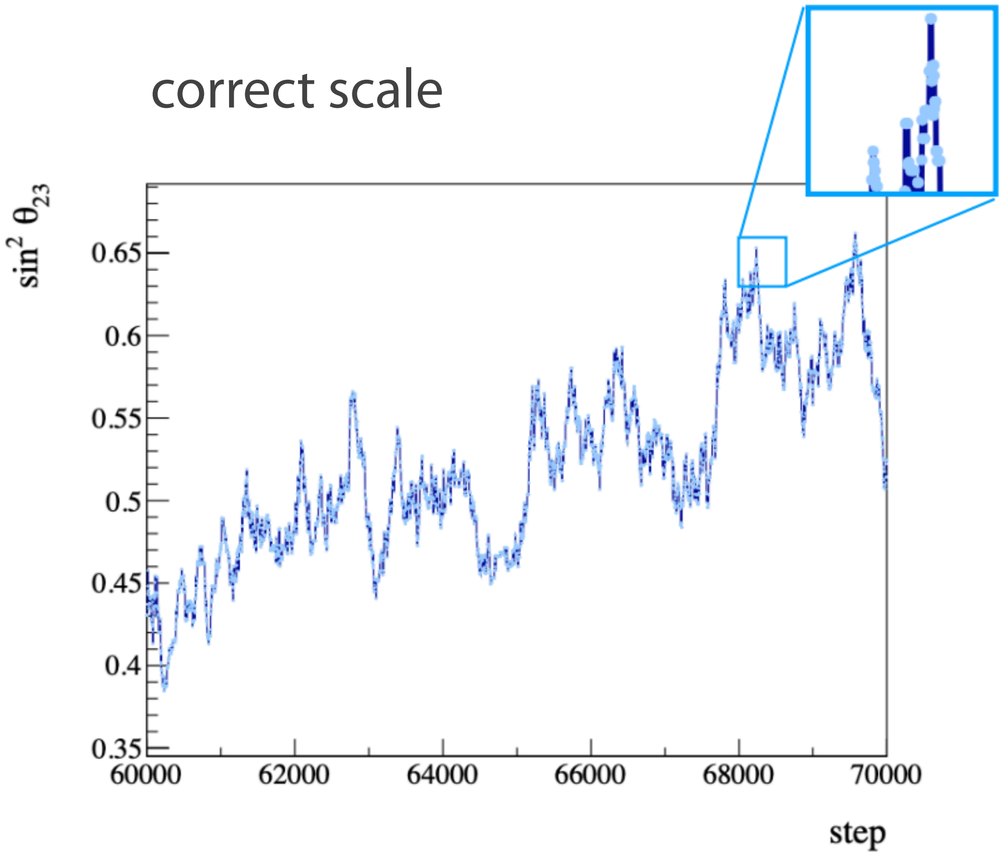
Motivation: limitations with MH

- Need to specify proposal with adequate scale/covariance — completely heuristic.
- Requires tuning to the posterior scale, unknown *a priori*! See https://chi-feng.github.io/mcmc-demo/app.html

incorrect scale

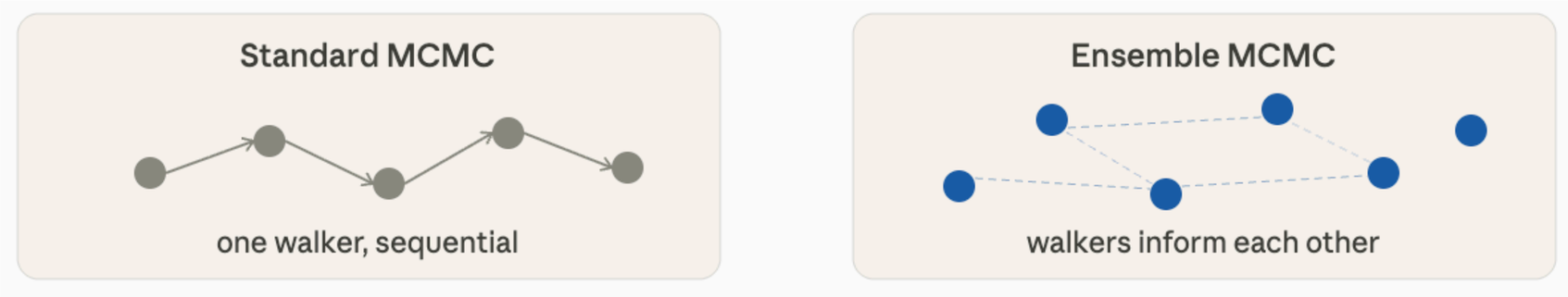


correct scale



Ensemble MCMC

- **General idea**
  - Instead of a single chain, run *N* walkers in parallel. Each walker's proposal is built from the current positions of the others — giving automatic adaptation to the posterior shape, with no scale to tune.



- **Markov property preserved**
  - Proposals depend only on walkers' current positions, not their history — so the ensemble is still Markov in the sense of $P\left((\theta_1^{i+1}, \ldots, \theta_{2n}^{i+1}) \,|\, (\theta_1^{i}, \ldots, \theta_{2n}^{i})\right)$.
- **Main reference**: *Ensemble samplers with affine invariance* (Goodman and Weare, 2010) with *stretch* move. Implemented as emcee (Foreman-Mackey et al. 2013) — standard in astrophysics.

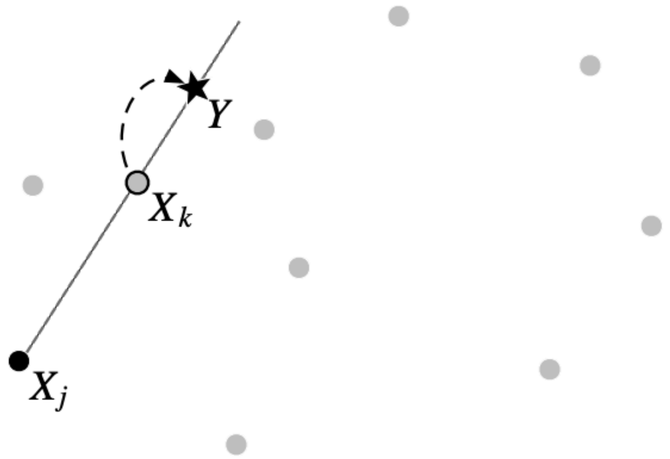
Ensemble MCMC

Algorithm

Split N walkers into even subsets A and B. For each $\theta_j^i \in A$, in parallel

1. Pick a random $\theta_k \in B$.
2. Draw $z \sim g(z) \propto 1/\sqrt{z}$ on $[1/a, a]$ and propose:
   $$\theta_j^{i+1} = \theta_j^i + Z(\theta_k^i - \theta_j^i)$$
3. Accept with MH ratio weighted by Z^{d-1}

Then repeat for B using updated A.



Avantages

- No scale to tune
- Invariant under affine transforms
- Good mixing and parallelisation

Limitations

- Burn-in needed before proposals are useful
- Poor on strongly multimodal posteriors

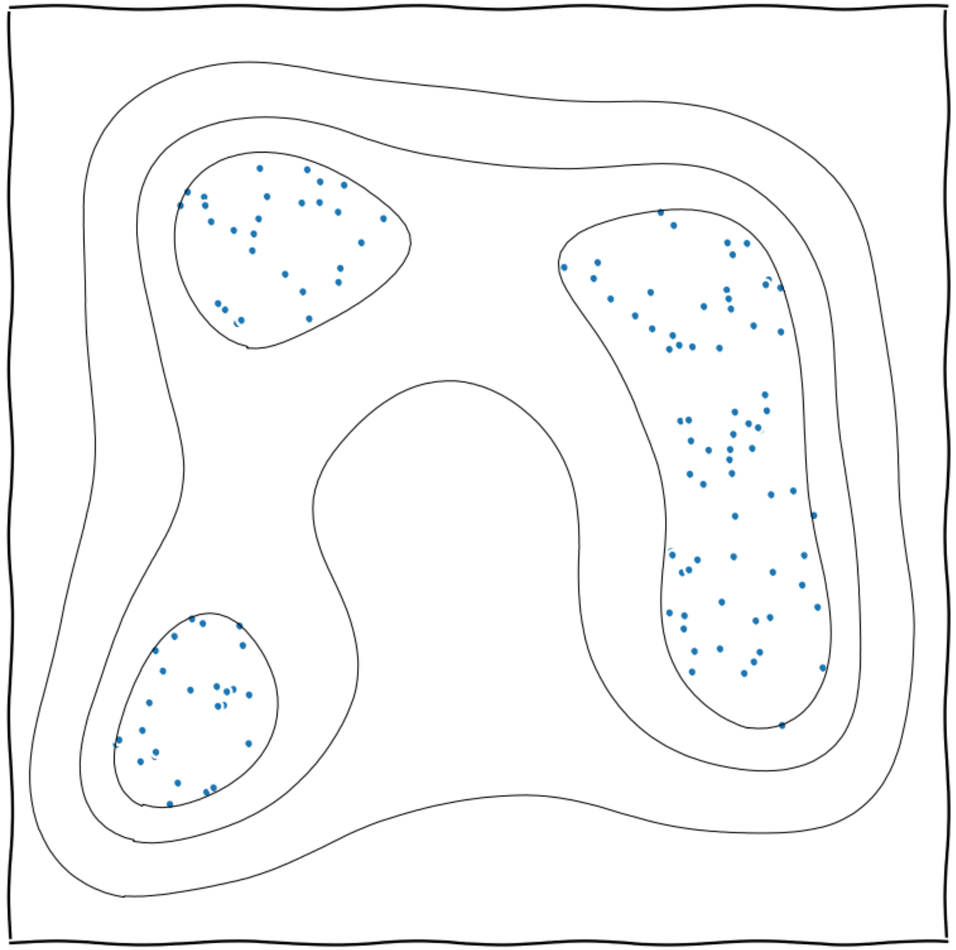
Nested sampling

- **What is it?**
  - The evidence $P(d) = \int P(d \mid \theta) P(\theta) \mathrm{d}\theta$ is a high-dimensional integral, exponentially difficult to compute.
  - Nested sampling is a Monte Carlo algorithm for computing the Bayesian evidence (model comparison) and posterior samples (parameter estimation) as a byproduct.
- **Key insight**
  - Define $X(\lambda)$ = prior volume where $P(d \mid \theta) > \lambda = \int_{P(d|\theta)>\lambda} P(\theta) \mathrm{d}\theta \in [0,1]$.
  - $X(\lambda)$ can be approximated by the ratio of prior samples within/without $\{\theta : P(d \mid \theta) > \lambda\}$.
  - The evidence becomes a 1d integral : $P(d) = \int_0^1 \lambda(X) \mathrm{d}X \approx \sum_i \lambda_i \Delta X_i$

Nested sampling

Algorithm

1. Draw N live points from the prior $P(\theta)$.
2. Find the point with the lowest likelihood L* — call it the "dead point".
3. Record it: it **contributes ΔX · L* to the evidence sum**, where ΔX is the decrease in prior volume.
4. Replace it with a **new sample drawn from the prior subject to L > L***.
5. Repeat — the likelihood threshold rises, the prior volume shrinks by ~1/N each step.



At each step the prior volume shrinks by a factor $\sim 1/N$. After k steps, $X_k \approx \exp(-k/N)$.

Nested sampling

Posterior samples

Strengths

- Computes the evidence
- Handles multimodel posteriors
- Independent posterior samples for free from dead points with weights $\Delta X_i \cdot L_i$

Limitations

- Constrained sampling is hard in high-D
- Z has statistical uncertainty
- Tuning N live points matters

- **Implementations** differ by how they sample from the constrained prior
  - Multinest fits ellipsoids
  - Polychords does slice sampling with internal MCMC
  - Dynesty adapts the number of live points

Hands-on 2

Run ensemble MCMC and nested sampling

```
    chi2 = (y - y_model)**2 / (yerr**2)
    return np.sum(-chi2 / 2)

def log_prior(theta):
    if all(theta > theta_low) and all(theta < theta_high):
        return 0
    return -np.inf

def log_posterior(theta, x, y, yerr):
    lp = log_prior(theta)
    if np.isfinite(lp):
        lp += log_likelihood(theta, x, y, yerr)
    return lp


# create a small ball around the MLE the initialize each walker
nwalkers, ndim = 30, 5
theta_guess = [0.5, 0.6, 0.2, -0.2, 0.1]
pos = theta_guess + 1e-4 * np.random.randn(nwalkers, ndim)

# run emcee
sampler = emcee.EnsembleSampler(nwalkers, ndim, log_posterior, args=(x, y, y_err))
sampler.run_mcmc(pos, 10000, progress=True);
```

```
100%|██████████| 10000/10000 [00:45<00:00, 220.62it/s]
```

```
fig, axes = plt.subplots(ndim, sharex=True)
mcmc_samples = sampler.get_chain()
labels = ["l", "m", "s", "a", "b"]
for i in range(ndim):
    ax = axes[i]
    ax.plot(mcmc_samples[:, :, i], "k", alpha=0.3, rasterized=True)
    ax.set_xlim(0, 1000)
    ax.set_ylabel(labels[i])

axes[-1].set_xlabel("step number");
```

Other sampling techniques

- **Hamiltonian Monte Carlo (NUTS)**: use posterior gradients to make long, high-acceptance moves; excellent in high dimensions; needs differentiable models.
- **Gibbs/slice sampling**: cycle through full conditionals exploiting hierarchical structure of model; mixes slowly under strong correlation.
- **Multimodal / tempering** — Annealed importance sampling or Sequential Monte Carlo: explore a ladder or sequence of flattened posteriors to cross modes (i.e. initially add temperature and slowly "cool down"); many chains to manage.
- **Fast approximations** — Variational inference and the Laplace approximation: replace sampling by optimization or a Gaussian fit at the mode; very fast and scalable; approximate, but can be combined with importance sampling to become exact.
- **Likelihood-free** — Simulation-based inference: target the posterior from simulations alone when the likelihood is intractable; needs many simulations and careful validation. See tomorrow!

Summary

1. Frequentist inference aims at recovering true parameter value
2. Bayesian inference encodes parameter uncertainty by turning them into random variables
3. Bayes' theorem: data updates parameter distribution from $P(\theta)$ to $P(\theta \mid d)$
4. Deriving credible intervals requires sampling the posterior
5. MCMC performs a random walk in parameter space to produce posterior samples
6. Metropolis-Hastings is the traditional implementation of MCMC, but requires fine-tuning
7. Modern methods (ensemble MCMC, nested sampling, Hamiltonian MC) improve sampling efficiency dramatically: find which one suits your problem!

Parameter constraints

Frequentist confidence intervals

- **Setup**: Assume there exists a true θ_0 value at which data were realised $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ and recover it.

Parameter constraints

Frequentist confidence intervals

- **Setup**: Assume there exists a true θ_0 value at which data were realised $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ and recover it.

- **Confidence interval**: a data-dependent set $I_\alpha(d)$ such that $P_{d\sim\mathscr{L}(\cdot|\theta)}\big(\theta \in I_\alpha(d)\big) \geq 1-\alpha$ for all $\theta \in \Omega$. This means that if we drew many data realisations from the likelihood $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ — by repeating the experiment — a fraction $1-\alpha$ of the corresponding intervals would contain θ_0.

Parameter constraints

Frequentist confidence intervals

- **Setup**: Assume there exists a true θ_0 value at which data were realised $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ and recover it.
- **Confidence interval**: a data-dependent set $I_\alpha(d)$ such that $P_{d\sim\mathscr{L}(\cdot|\theta)}\big(\theta \in I_\alpha(d)\big) \geq 1-\alpha$ for all $\theta \in \Omega$. This means that if we drew many data realisations from the likelihood $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ — by repeating the experiment — a fraction $1-\alpha$ of the corresponding intervals would contain θ_0.
- **Pivot**: a scalar function $T(d,\theta)$ such that the distribution of $T(d,\theta)$, for $d \sim \mathscr{L}(\,\cdot\,|\,\theta)$, is the same for all $\theta \in \Omega$. If we know the quantiles $T(d,\theta)$, ie a, b such that $P_{d\sim\mathscr{L}(\cdot|\theta)}(a < T(d,\theta) < b) = 1-\alpha$, then $I_\alpha(d) = \{\theta : a < T(d,\theta) < b\}$. The trick is to find that function *in general*!

Parameter constraints

Frequentist confidence intervals

- **Setup**: Assume there exists a true θ_0 value at which data were realised $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ and recover it.
- **Confidence interval**: a data-dependent set $I_\alpha(d)$ such that $P_{d\sim\mathscr{L}(\cdot|\theta)}\big(\theta \in I_\alpha(d)\big) \geq 1-\alpha$ for all $\theta \in \Omega$. This means that if we drew many data realisations from the likelihood $d \sim \mathscr{L}(\,\cdot\,|\,\theta_0)$ — by repeating the experiment — a fraction $1-\alpha$ of the corresponding intervals would contain θ_0.
- **Pivot**: a scalar function $T(d,\theta)$ such that the distribution of $T(d,\theta)$, for $d \sim \mathscr{L}(\,\cdot\,|\,\theta)$, is the same for all $\theta \in \Omega$. If we know the quantiles $T(d,\theta)$, ie a, b such that $P_{d\sim\mathscr{L}(\cdot|\theta)}(a < T(d,\theta) < b) = 1-\alpha$, then $I_\alpha(d) = \{\theta : a < T(d,\theta) < b\}$. The trick is to find that function *in general*!
- **Special linear Gaussian case**: if $d \sim \mathcal{N}(\theta_0 x, \Sigma)$, then the MLE is $\hat{\theta}(d) \sim \mathcal{N}(\theta_0, \Sigma(x^\intercal x)^{-1})$. The pivot is simply $T(d,\theta) = \hat{\theta}(d) - \theta_0 \sim \mathcal{N}(0, \Sigma(x^\intercal x)^{-1})$ and $I_\alpha(d) = \hat{\theta}(d) \pm z_{\alpha/2}\sigma_{\hat{\theta}}$.

Parameter constraints Frequentist confidence intervals

- **Asymptotic laws**: MLE goes $\sqrt{n}\left(\hat{\theta}(d) - \theta_0\right) \xrightarrow{n\to\infty} \mathcal{N}\left(0, I(\theta_0)^{-1}\right)$ when data $n \to \infty$, where $I(\theta)$=Fisher.

Parameter constraints Frequentist confidence intervals

- **Asymptotic laws**: MLE goes $\sqrt{n}\left(\hat{\theta}(d) - \theta_0\right) \xrightarrow{n\to\infty} \mathcal{N}\left(0, I(\theta_0)^{-1}\right)$ when data $n \to \infty$, where $I(\theta)$=Fisher.
  - **Wald**: $T(d,\theta) = \left(\hat{\theta}(d) - \theta_0\right)/\sigma_{\hat{\theta}} \to \mathcal{N}(0,1)$. Symmetric interval $I_\alpha(d) = \hat{\theta}(d) \pm z_{\alpha/2}\sigma_{\hat{\theta}}$, but requires the likelihood Hessian to compute the MLE error $\sigma_{\hat{\theta}}$.

Parameter constraints Frequentist confidence intervals

- **Asymptotic laws**: MLE goes $\sqrt{n}\left(\hat{\theta}(d) - \theta_0\right) \xrightarrow{n\to\infty} \mathcal{N}\left(0, I(\theta_0)^{-1}\right)$ when data $n \to \infty$, where $I(\theta)$=Fisher.
  - **Wald**: $T(d,\theta) = \left(\hat{\theta}(d) - \theta_0\right)/\sigma_{\hat{\theta}} \to \mathcal{N}(0,1)$. Symmetric interval $I_\alpha(d) = \hat{\theta}(d) \pm z_{\alpha/2}\sigma_{\hat{\theta}}$, but requires the likelihood Hessian to compute the MLE error $\sigma_{\hat{\theta}}$.
  - **Wilk**: likelihood ratio $T(d,\theta) = 2\log\dfrac{\mathscr{L}(d\,|\,\hat{\theta}(d))}{\mathscr{L}(d\,|\,\theta)} \to \chi^2_p$, so $I_\alpha(d) = \{\theta : T(d,\theta) < \chi^2_{p,\alpha}\}$ for p params.

    For a given parameter of interest, use profile likelihood (max over all other params) and set $p = 1$.

Parameter constraints Frequentist confidence intervals

- **Asymptotic laws**: MLE goes $\sqrt{n}\left(\hat{\theta}(d) - \theta_0\right) \xrightarrow{n\to\infty} \mathcal{N}\left(0, I(\theta_0)^{-1}\right)$ when data $n \to \infty$, where $I(\theta)$=Fisher.
  - **Wald**: $T(d,\theta) = \left(\hat{\theta}(d) - \theta_0\right)/\sigma_{\hat{\theta}} \to \mathcal{N}(0,1)$. Symmetric interval $I_\alpha(d) = \hat{\theta}(d) \pm z_{\alpha/2}\sigma_{\hat{\theta}}$, but requires the likelihood Hessian to compute the MLE error $\sigma_{\hat{\theta}}$.
  - **Wilk**: likelihood ratio $T(d,\theta) = 2\log\dfrac{\mathscr{L}(d\,|\,\hat{\theta}(d))}{\mathscr{L}(d\,|\,\theta)} \to \chi^2_p$, so $I_\alpha(d) = \{\theta : T(d,\theta) < \chi^2_{p,\alpha}\}$ for p params.

    For a given parameter of interest, use profile likelihood (max over all other params) and set $p = 1$.
    - Invariant to reparameterization; respects boundaries; naturally asymmetric; better coverage than Wald; but requires inner optimisations. Bartlet-like corrections rescale T to improve coverage.

Parameter constraints Frequentist confidence intervals

- **Asymptotic laws**: MLE goes $\sqrt{n}\left(\hat{\theta}(d) - \theta_0\right) \xrightarrow{n\to\infty} \mathcal{N}\left(0, I(\theta_0)^{-1}\right)$ when data $n \to \infty$, where $I(\theta)$=Fisher.
  - **Wald**: $T(d,\theta) = \left(\hat{\theta}(d) - \theta_0\right)/\sigma_{\hat{\theta}} \to \mathcal{N}(0,1)$. Symmetric interval $I_\alpha(d) = \hat{\theta}(d) \pm z_{\alpha/2}\sigma_{\hat{\theta}}$, but requires the likelihood Hessian to compute the MLE error $\sigma_{\hat{\theta}}$.
  - **Wilk**: likelihood ratio $T(d,\theta) = 2\log\dfrac{\mathscr{L}(d\,|\,\hat{\theta}(d))}{\mathscr{L}(d\,|\,\theta)} \to \chi^2_p$, so $I_\alpha(d) = \{\theta : T(d,\theta) < \chi^2_{p,\alpha}\}$ for p params.

    For a given parameter of interest, use profile likelihood (max over all other params) and set $p = 1$.
    - Invariant to reparameterization; respects boundaries; naturally asymmetric; better coverage than Wald; but requires inner optimisations. Bartlet-like corrections rescale T to improve coverage.
    - Neyman-Pearson lemma ensures likelihood ratio is optimal, but still only asymptotically χ^2.

Parameter constraints Frequentist confidence intervals

- **Asymptotic laws**: MLE goes $\sqrt{n}\left(\hat{\theta}(d) - \theta_0\right) \xrightarrow{n\to\infty} \mathcal{N}\left(0, I(\theta_0)^{-1}\right)$ when data $n \to \infty$, where $I(\theta)$=Fisher.
  - **Wald**: $T(d,\theta) = \left(\hat{\theta}(d) - \theta_0\right)/\sigma_{\hat{\theta}} \to \mathcal{N}(0,1)$. Symmetric interval $I_\alpha(d) = \hat{\theta}(d) \pm z_{\alpha/2}\sigma_{\hat{\theta}}$, but requires the likelihood Hessian to compute the MLE error $\sigma_{\hat{\theta}}$.
  - **Wilk**: likelihood ratio $T(d,\theta) = 2\log\dfrac{\mathscr{L}(d\,|\,\hat{\theta}(d))}{\mathscr{L}(d\,|\,\theta)} \to \chi^2_p$, so $I_\alpha(d) = \{\theta : T(d,\theta) < \chi^2_{p,\alpha}\}$ for p params.

    For a given parameter of interest, use profile likelihood (max over all other params) and set $p = 1$.
    - Invariant to reparameterization; respects boundaries; naturally asymmetric; better coverage than Wald; but requires inner optimisations. Bartlet-like corrections rescale T to improve coverage.
    - Neyman-Pearson lemma ensures likelihood ratio is optimal, but still only asymptotically χ^2.
- **Bootstrapping**: for iid observations $d = (d_1, \ldots, d_n)$, resample (with replacement) the data set many times, compute the MLE $\hat{\theta}(d_1^*, \ldots, d_n^*)$ every time, use quantiles to obtain $I_\alpha(d)$.