IN2P3 School of Statistics
15th of May, 2024

Leïla Haegel
IP2I Lyon / CNRS

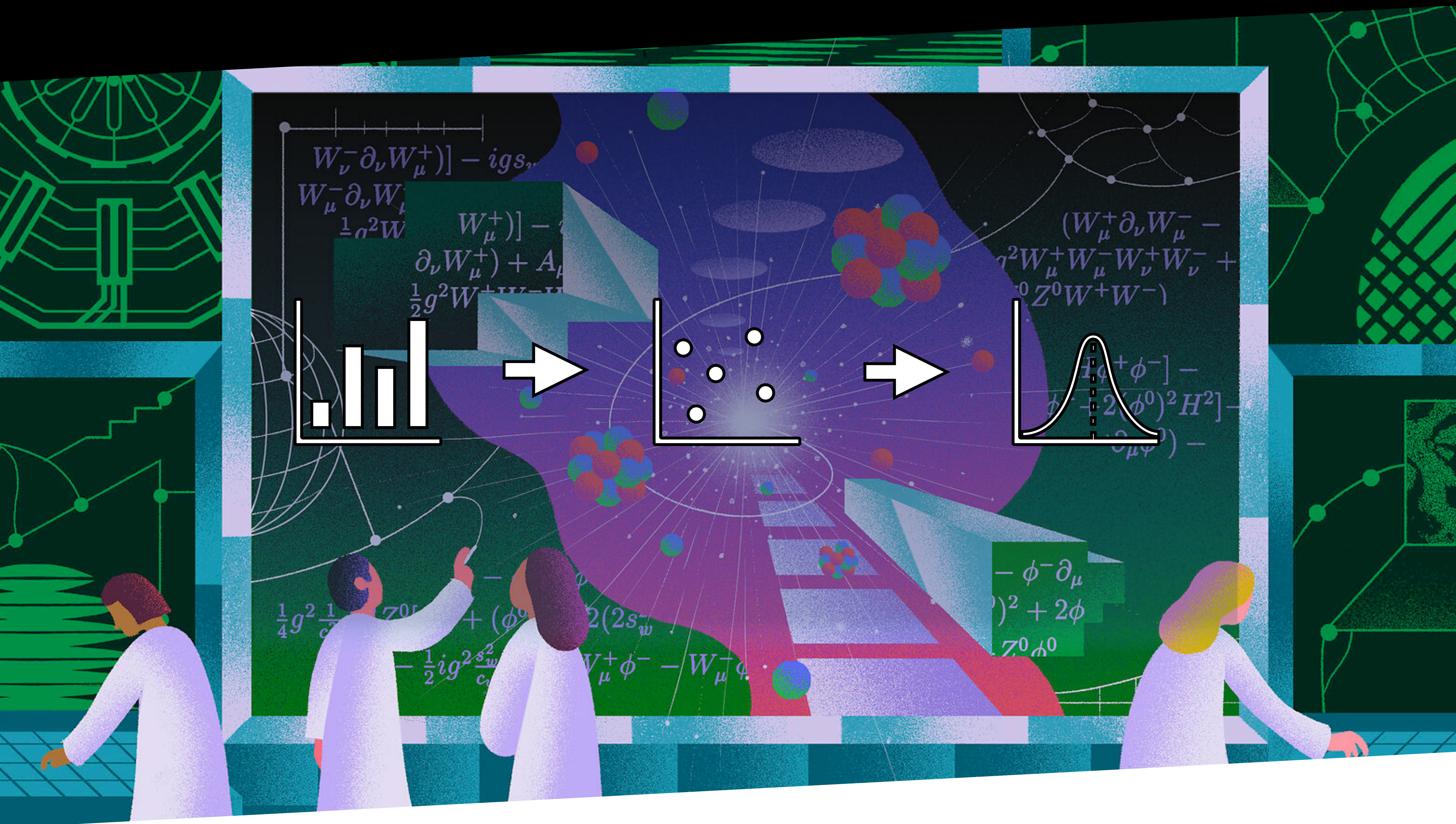


Illustration by Sandbox Studio, Chicago with Ariel Davis

Bayesian inference with Markov Chains for particle & astro physics

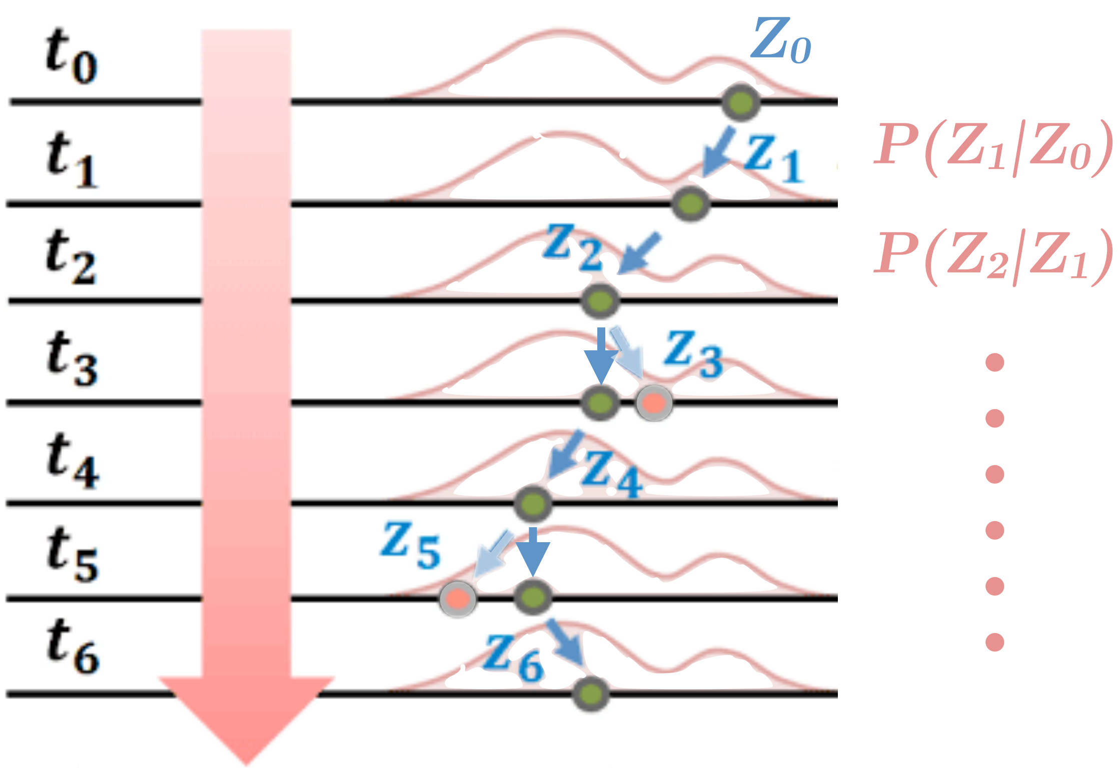
Outline

- **Markov chains**
  - Definition and properties
  - The Metropolis-Hastings algorithm
  - Other algorithms
- **Bayesian inference**
  - Reminder of Bayesian statistics
  - Application of Markov chains
  - Marginalisation, etc

Introduction to Markov chains

Markov chains: a primer

- **Markov Chain Monte-Carlo (MCMC)**
  - Markov chains are a *semi-random sequence* of events, or states $\vec{Z} = \{Z_i\}$
  - *Stochastic process*: each state Z_i is reached randomly
  - *Sequential process*: the probability of reaching a state Z_i only depends on the state Z_{i-1} reached before
  - *Memory-less process*: the chain does not remember states before Z_{i-1}



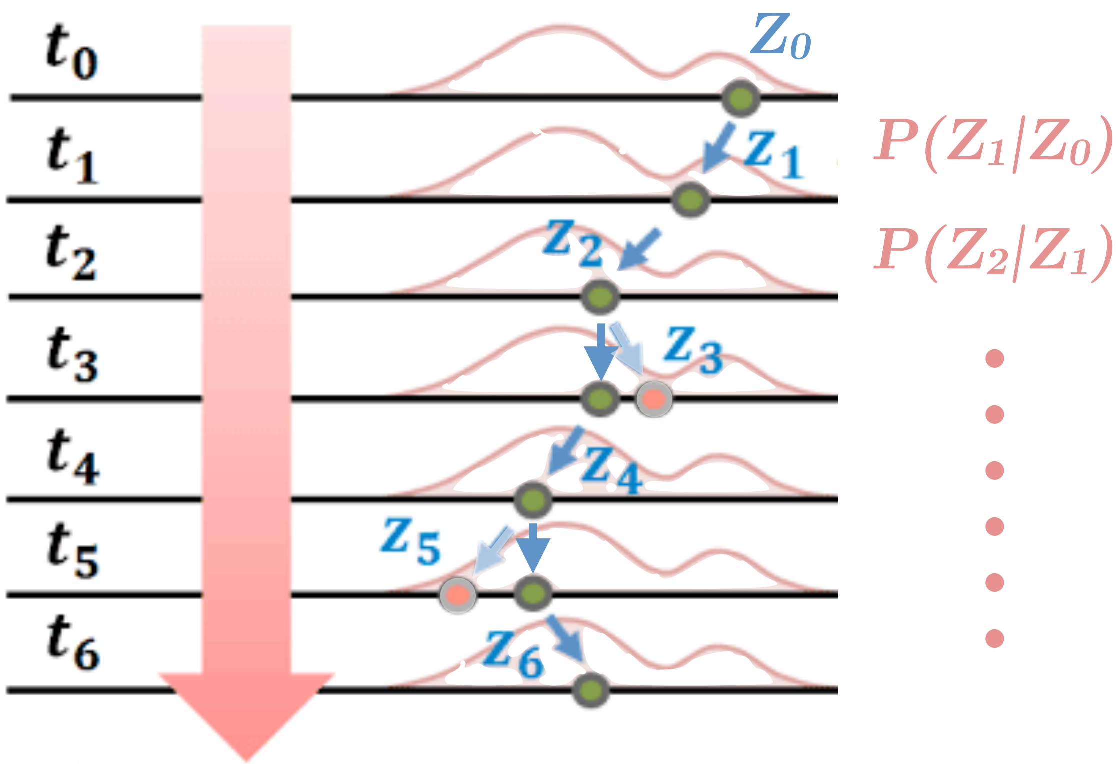
Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chains: a primer

- **Markov Chain Monte-Carlo (MCMC)**
  - Markov chains are a *semi-random sequence* of events, or states $\overrightarrow{Z} = \{Z_i\}$
  - *Stochastic process*: each state Z_i is reached randomly
  - *Sequential process*: the probability of reaching a state Z_i only depends on the state Z_{i-1} reached before
  - *Memory-less process*: the chain does not remember states before Z_{i-1}

- **Many applications**
  - Sampling of unknown distributions → ***What we will focus on***
  - Modelling stochastic processes
  - Random number generation

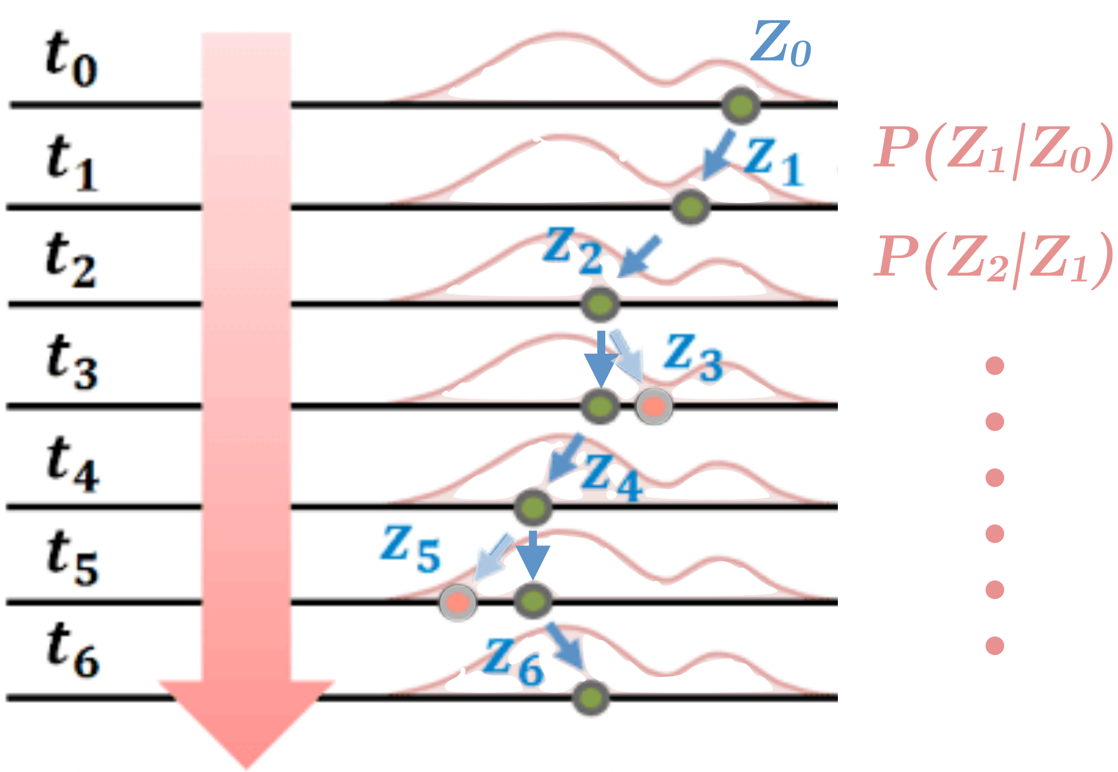
- **Irreducibility**
  - A Markov chain is *irreducible* if any state in $\vec{Z}$ can be reached in a finite number of steps: $P(X_{t+\tau} = Z_i \mid X_t = Z_j) \;>\; 0$



Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chain properties

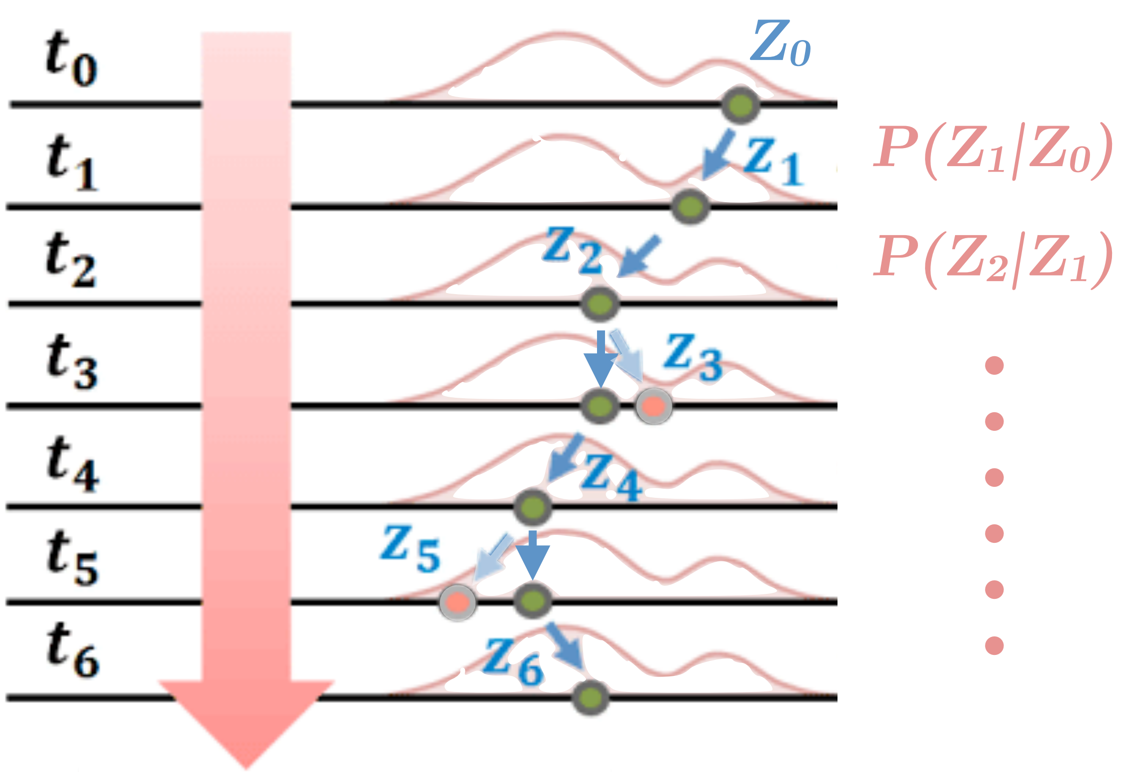
- **(A)periodicity**
  - A state Z_i is *periodic* if it is visited every fixed number of step Δ (or a multiple $N\Delta$)
  - The period d_i is given by the greatest commun denominator (gcd):
    $d_i = gcd\{t : P(X_t = Z_i \mid X_0 = Z_i) > 0\}$
  - If $d_i = 1$, the state is aperiodic, and so is the Markov chain



Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chain properties

- **Recurrence**
  - A state Z_i is *recurrent* if there is a non-0 probability that the Markov chain returns to Z_i, and *positive recurrent* if the number of steps to return to Z_i is finite
  - The number of steps to return to Z_i is:
    $\tau_{ii} = min\{t > 0 : P(X_t = Z_i \mid X_0 = Z_i) \ > \ 0\}$
  - *Recurrence* is defined that $P(\tau_{ii} < \infty) = 1$
    Positive recurrence is defined by the expectation is $E(\tau_{ii}) < \infty$

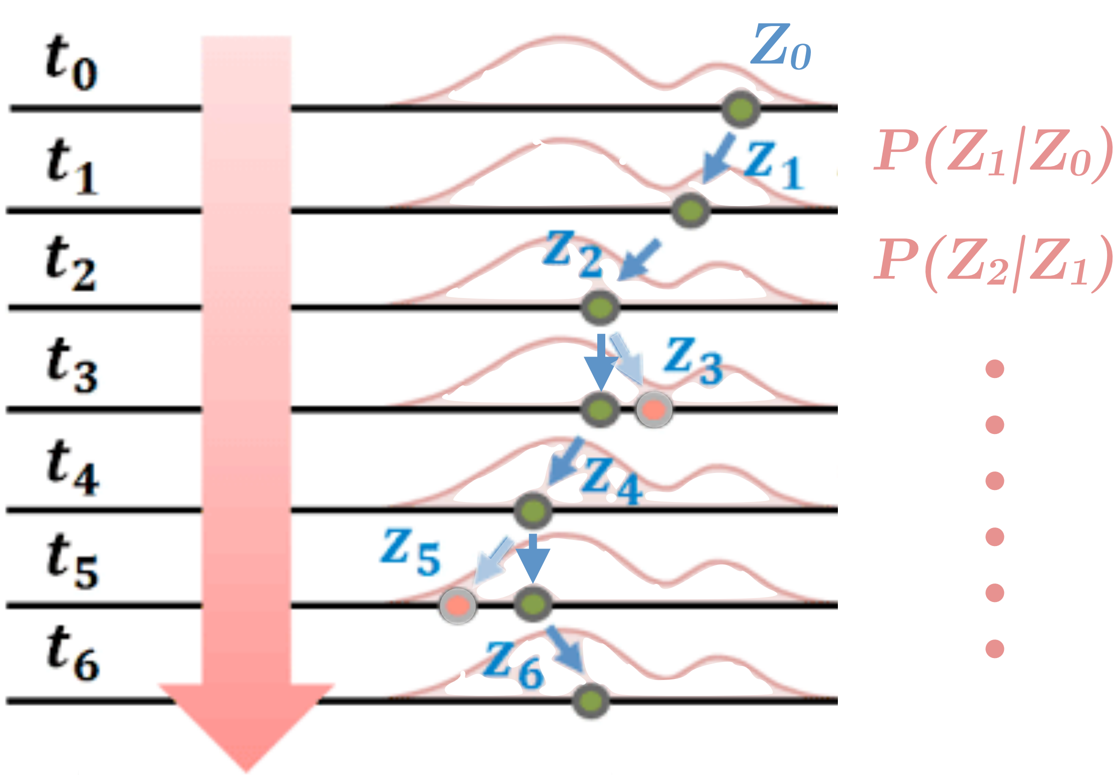


Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chain properties

- **Ergodicity**
  - A Markov chain is *ergodic* if it is possible to reach any state Z_i from any initial state Z_0
  - A chain is ergodic if it is *aperiodic*, *irreducible* and *positive recurrent*

A good reference on the topic: Gregory Gundersen article

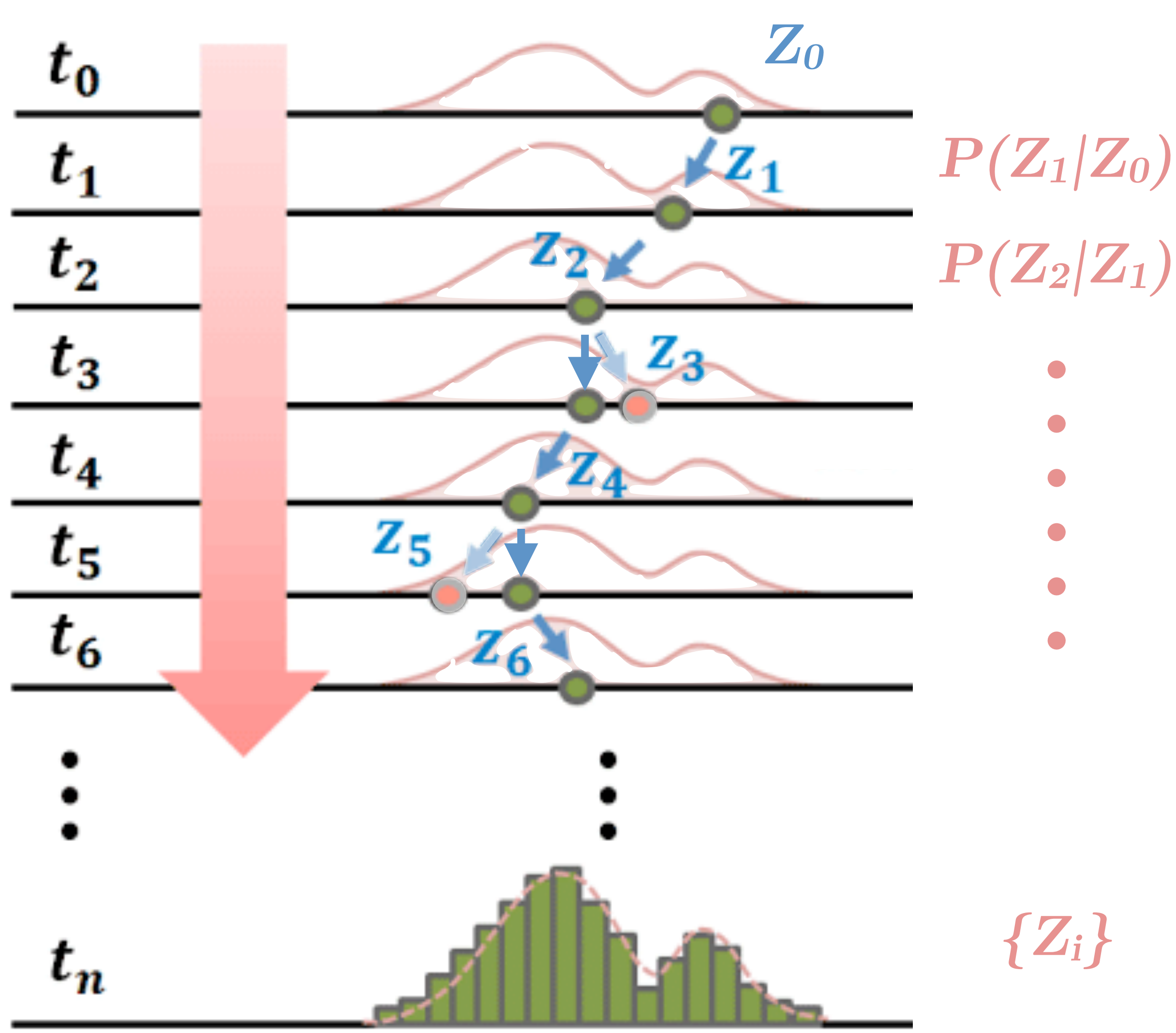


Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chain properties

- **Stationarity**
  - The probability to go from a state to another is: $P_{ij} = P(X_{t+1} = Z_i \mid X_t = Z_j)$
  - The matrix of transition probabilities P give the probability to reach any state when in another
  - A distribution Z is *stationary* if $Z = ZP$:
    the distribution of states is invariant under the transition probability and remains unchanged as the chain progress
  - The chain goes to each state Z_i proportionally to the distribution Z: $\sum_i Z_i \, P_{ij} = Z_j$
    for any $Z_i, Z_j \in S$ where S in the state space

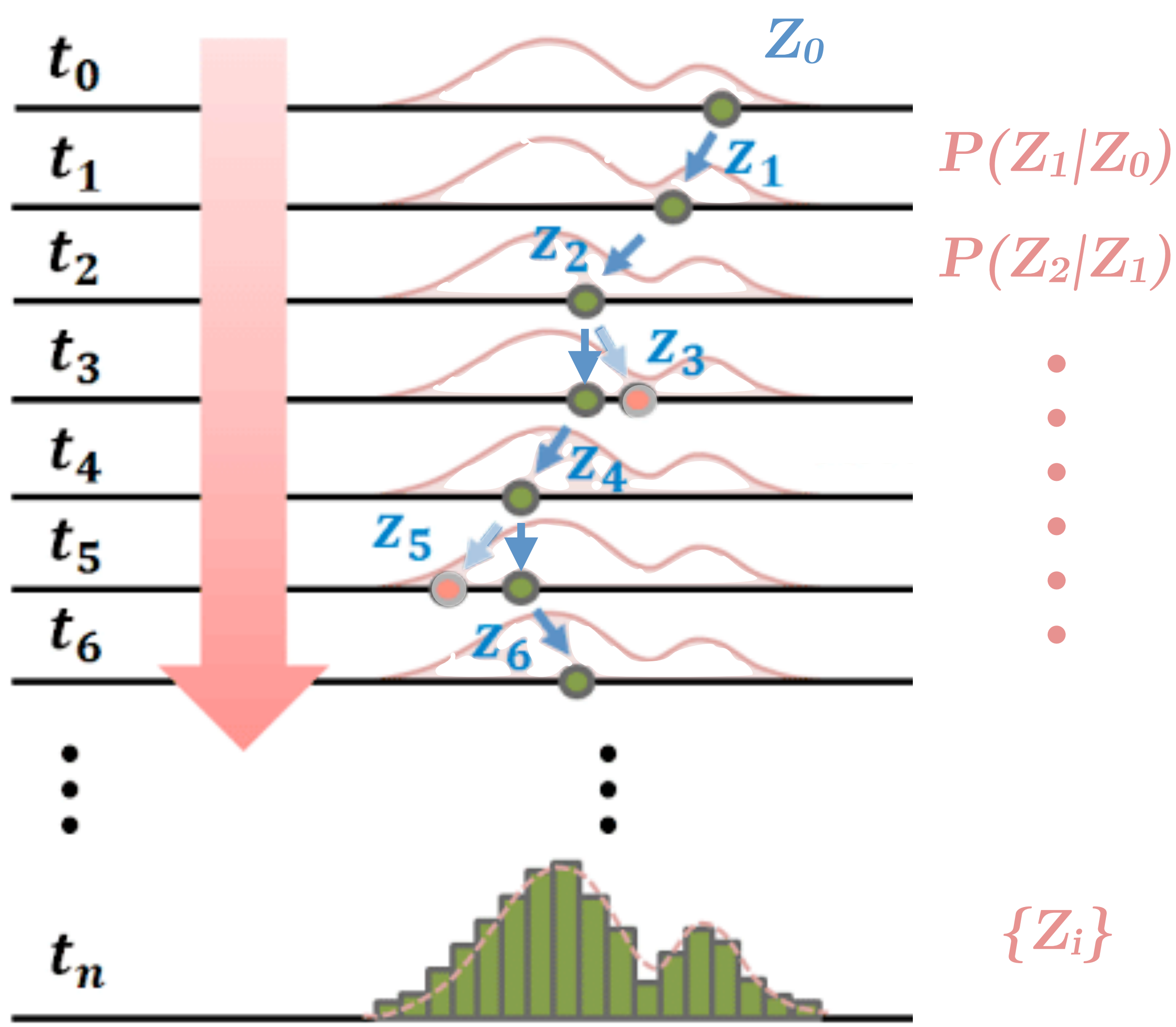
- **Uniqueness**
  - A chain that is irreducible and aperiodic has a *unique* stationary distribution Z



Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chain properties

- **Convergence**
  - A chain that is irreducible and aperiodic will always *converge* to the unique stationary distribution Z:
    $P(X_t = Z_i \mid X_0 = Z_0) \;\rightarrow\; Z(t)$ when $t \rightarrow \infty$



Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Markov chain properties

- **Sampling a distribution with a Markov chain**
  - If we create a Markov chain that is:
    - irreducible (can reach any state in the state space)
    - aperiodic (is not stuck between a subset of the space)
    - positive recurrent (can visit all the steps)
  - then the chain is:
    - defined by a unique stationary distribution (for the chain steps transition)
    - ergodic (it can reach any state wherever it starts)
    - therefore convergent to the stationary distribution

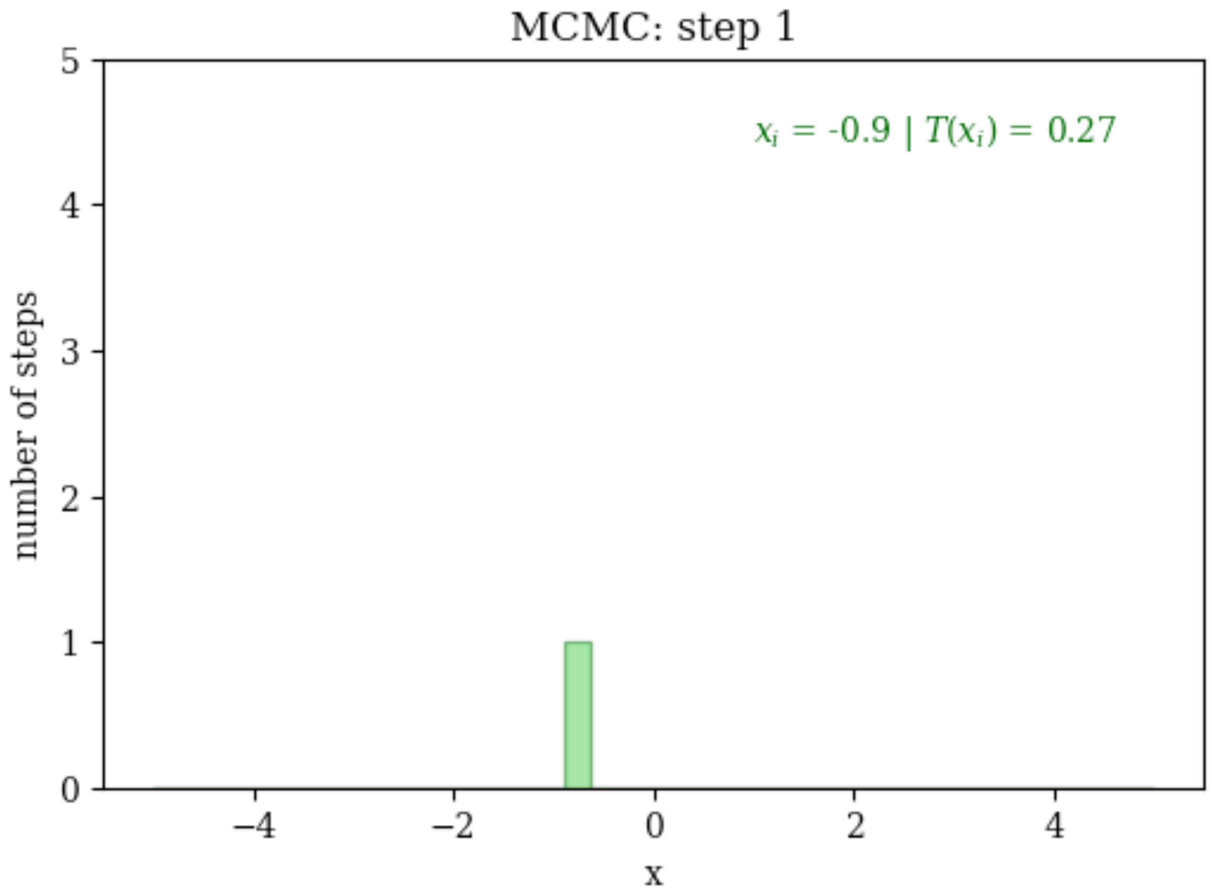
If we can create a Markov chain with those properties, the steps of the chain will be proportional to a distribution → the chain steps are samples from the distribution

Metropolis-Hastings algorithm

- **Metropolis-Hastings (MH) algorithm**
  - Algorithm defining a Markov chain with the properties mentioned beforehand
  - Can sample any probability distribution $Z(\vec{x})$ if we know a function $f(\vec{x}) \propto Z(\vec{x})$
  - The sampled probability distribution is often referred to as the *target distribution*
  - Notably, the MH algorithm can sample:
    - multidimensional distributions
    - distributions with local minima
    - non-continuous functions
    - non-differentiable functions

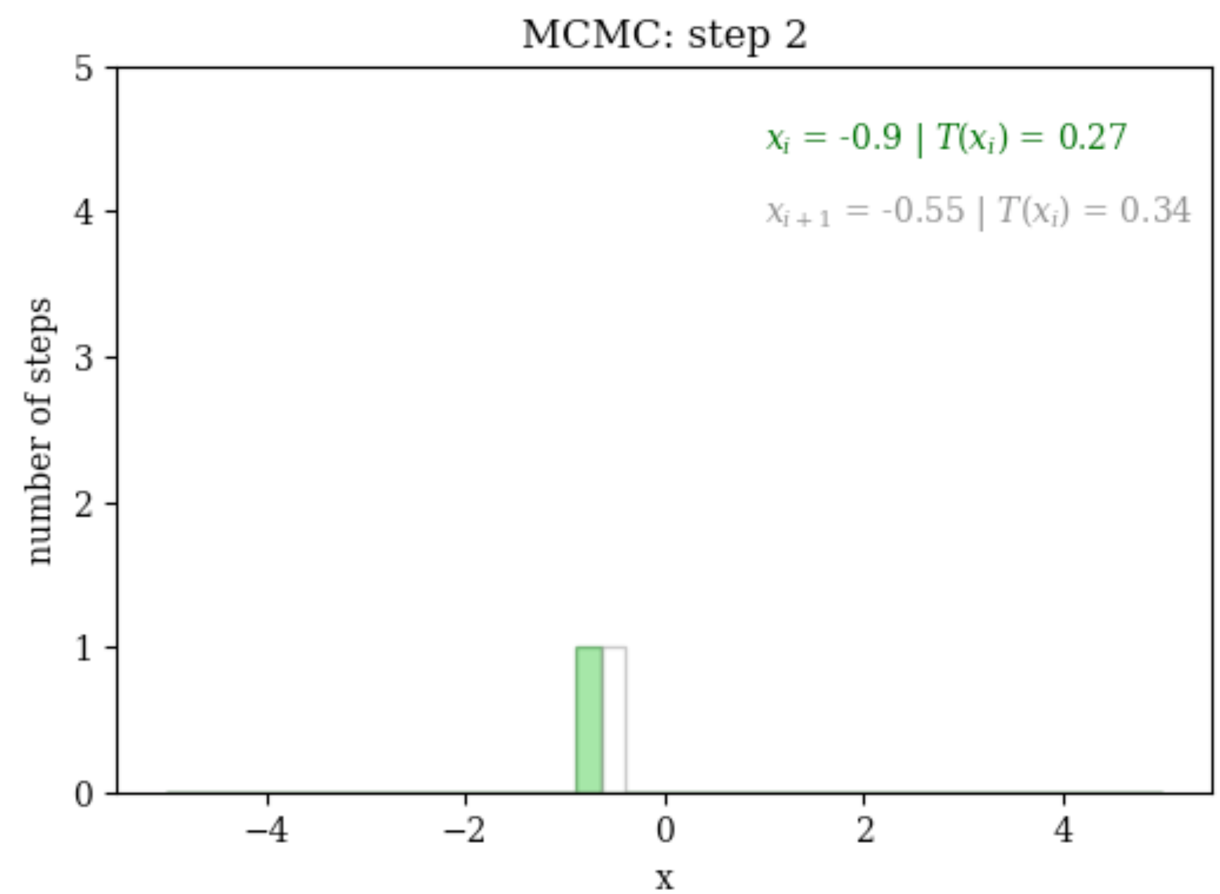
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution $G(x)$**
  - First step $i = 1$: start with a random choice of hypothetical value x_i



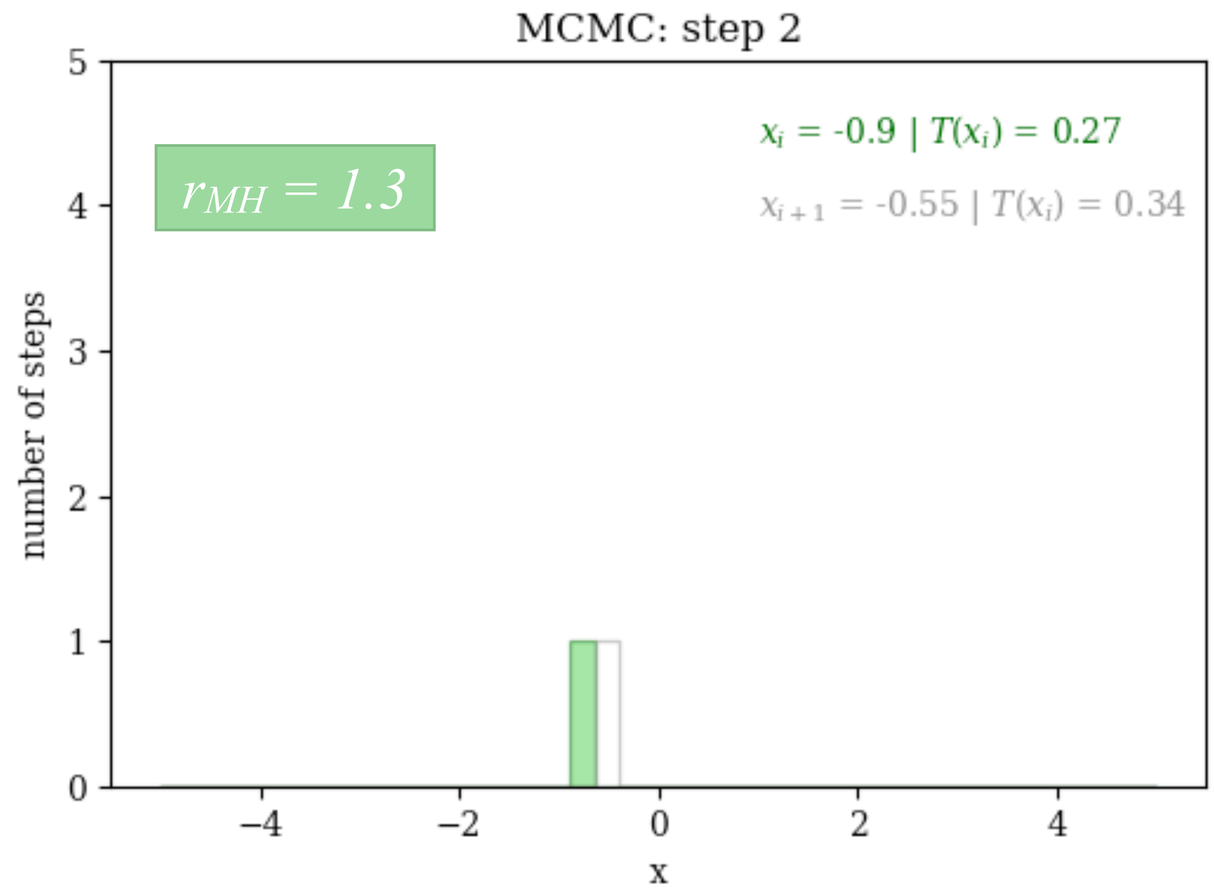
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution** $G(x)$
  - First step $i = 1$: start with a random choice of hypothetical value x_i
  - Propose a new step $i + 1$: using the *jump function* $J(x_i + 1 \,|\, x_i)$



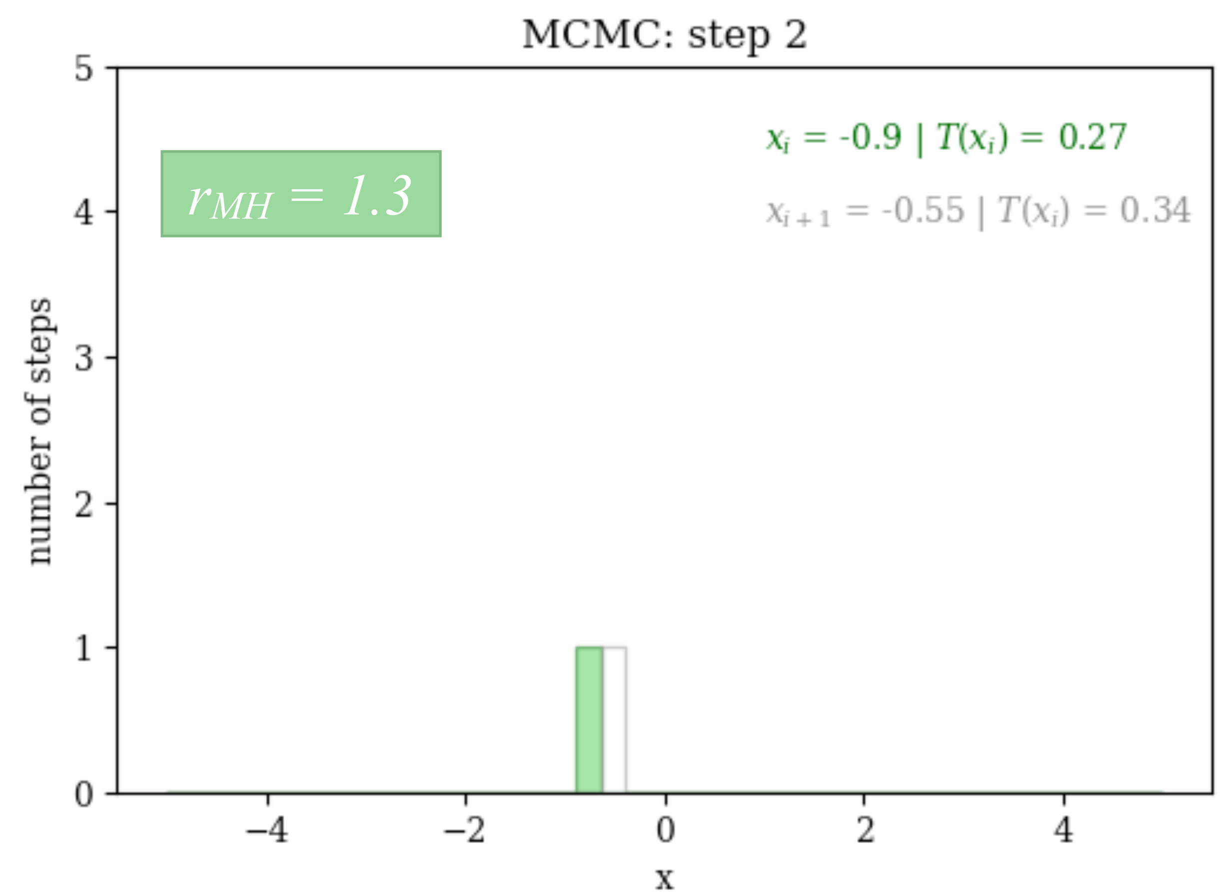
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution** $G(x)$
  - First step $i = 1$: start with a random choice of hypothetical value x_i
  - Propose a new step $i+1$: using the *jump function* $J(x_i + 1 \,|\, x_i)$
  - Compute the Metropolis-Hastings ratio r_{MH}: $r_{MH} = \dfrac{G(x_{i+1})\ J(x_i \,|\, x_{i+1})}{G(x_i)\ J(x_{i+1} \,|\, x_i)}$



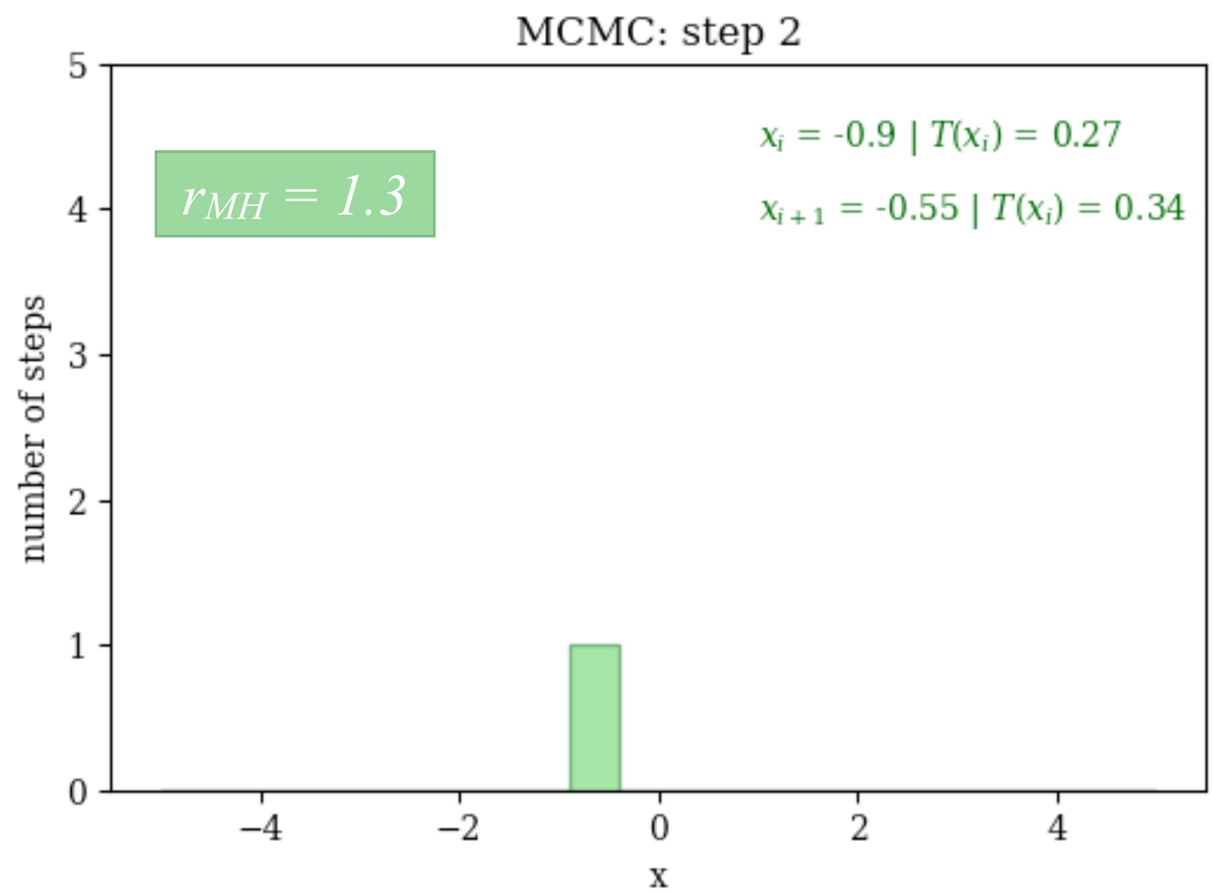
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution** $G(x)$
  - First step $i = 1$: start with a random choice of hypothetical value x_i
  - Propose a new step $i + 1$: using the *jump function* $J(x_i + 1 \,|\, x_i)$
  - Compute the Metropolis-Hastings ratio r_{MH}: $r_{MH} = \dfrac{G(x_{i+1})\ J(x_i \,|\, x_{i+1})}{G(x_i)\ J(x_{i+1} \,|\, x_i)}$
  - Apply the *acceptance function* $A(x_{i+1}, x_i) = min\{1,\ r_{MH}\}$



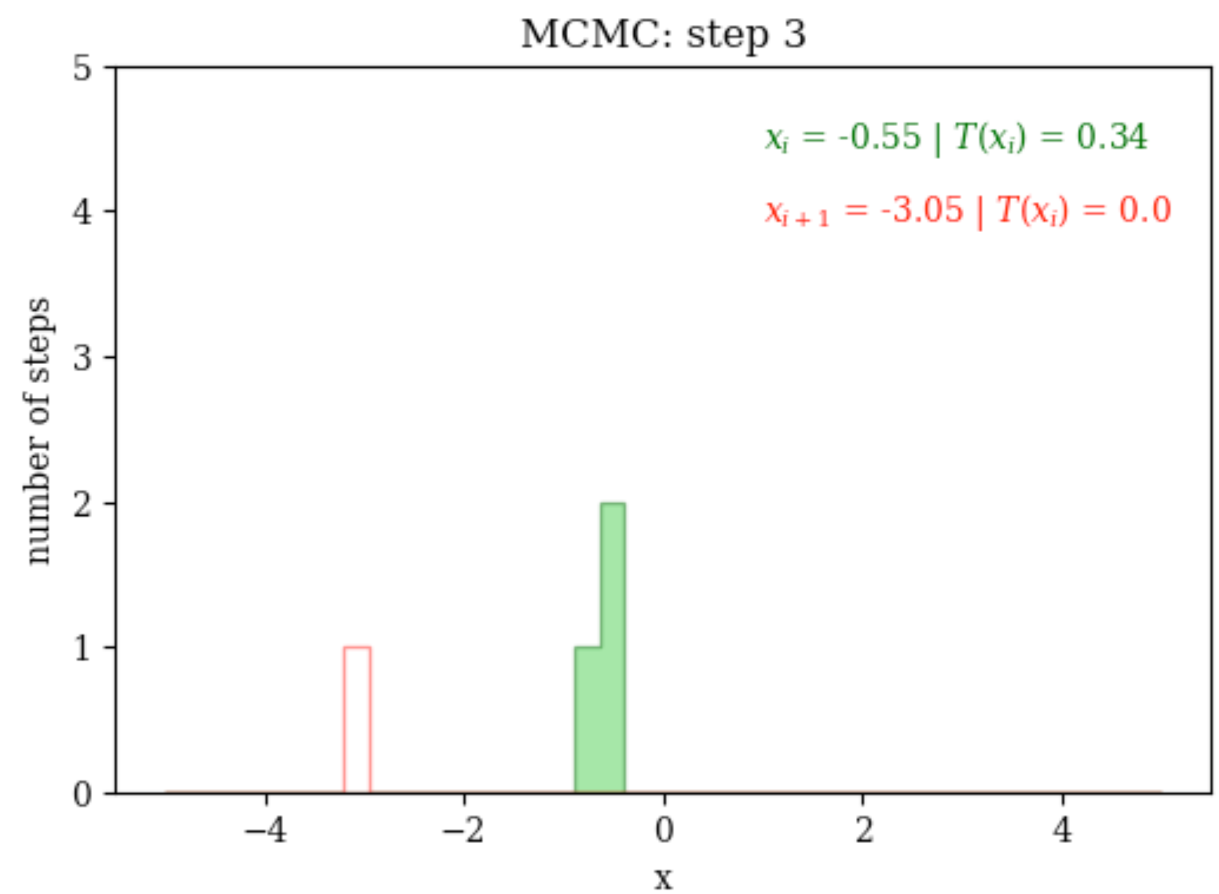
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution** $G(x)$
  - First step $i = 1$: start with a random choice of hypothetical value x_i
  - Propose a new step $i + 1$: using the *jump function* $J(x_i + 1 \,|\, x_i)$
  - Compute the Metropolis-Hastings ratio r_{MH}: $r_{MH} = \dfrac{G(x_{i+1})\ J(x_i \,|\, x_{i+1})}{G(x_i)\ J(x_{i+1} \,|\, x_i)}$
  - Apply the *acceptance function* $A(x_{i+1}, x_i) = min\{1,\ r_{MH}\}$
    Equivalent to:
    - $r_{MH} \geq 1 \rightarrow$ accept step $i + 1$



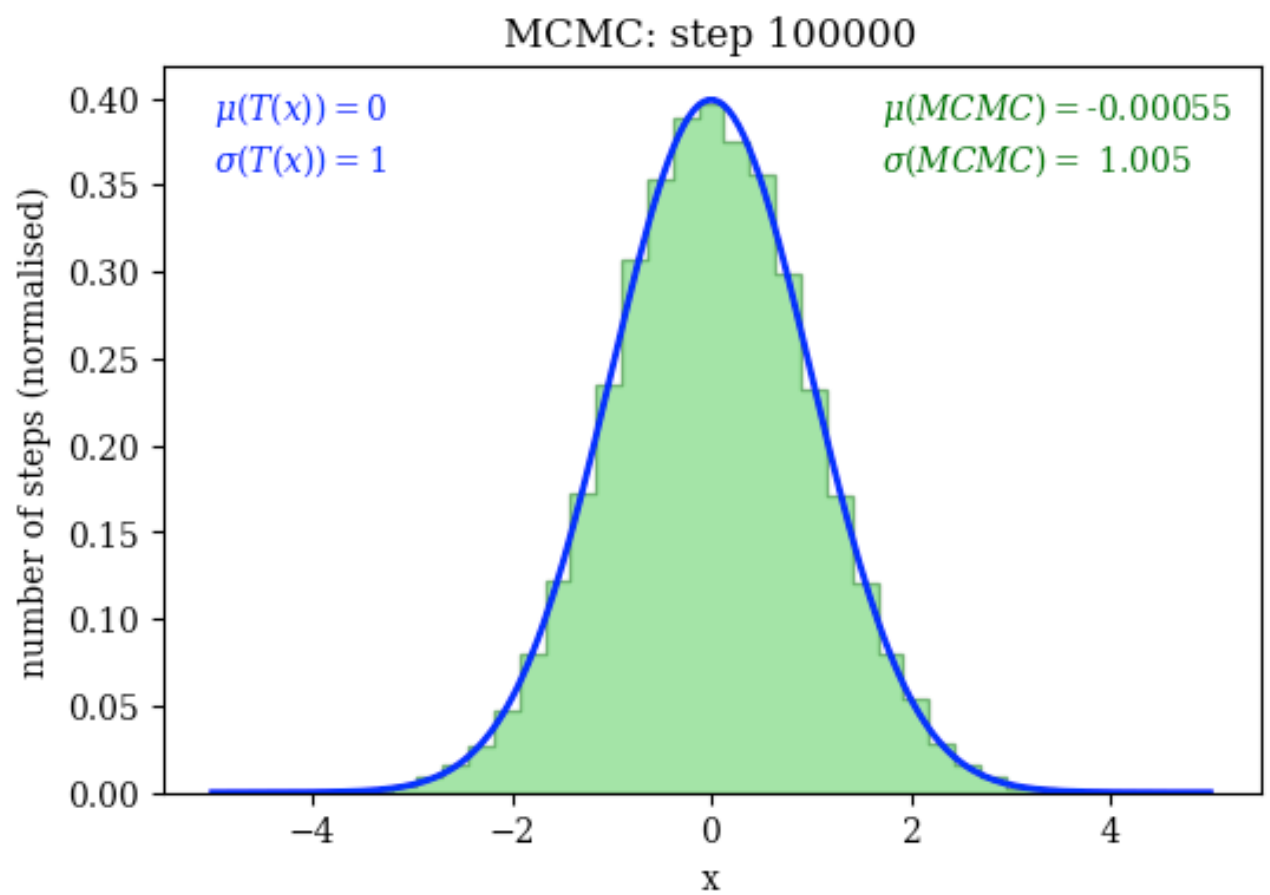
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution** $G(x)$
  - First step $i = 1$: start with a random choice of hypothetical value x_i
  - Propose a new step $i + 1$: using the *jump function* $J(x_i + 1 \,|\, x_i)$
  - Compute the Metropolis-Hastings ratio r_{MH}: $r_{MH} = \dfrac{G(x_{i+1})\ J(x_i \,|\, x_{i+1})}{G(x_i)\ J(x_{i+1} \,|\, x_i)}$
  - Apply the *acceptance function* $A(x_{i+1}, x_i) = min\{1,\ r_{MH}\}$
    Equivalent to:
    - $r_{MH} \geq 1 \rightarrow$ accept step $i + 1$
    - $r_{MH} < 1 \rightarrow$ throw a number $u \in \sqcap\,(0,1)$
      $r_{MH} \geq u \rightarrow$ accept step $i + 1$
      $r_{MH} < u \rightarrow$ reject step $i + 1$
      count again step i



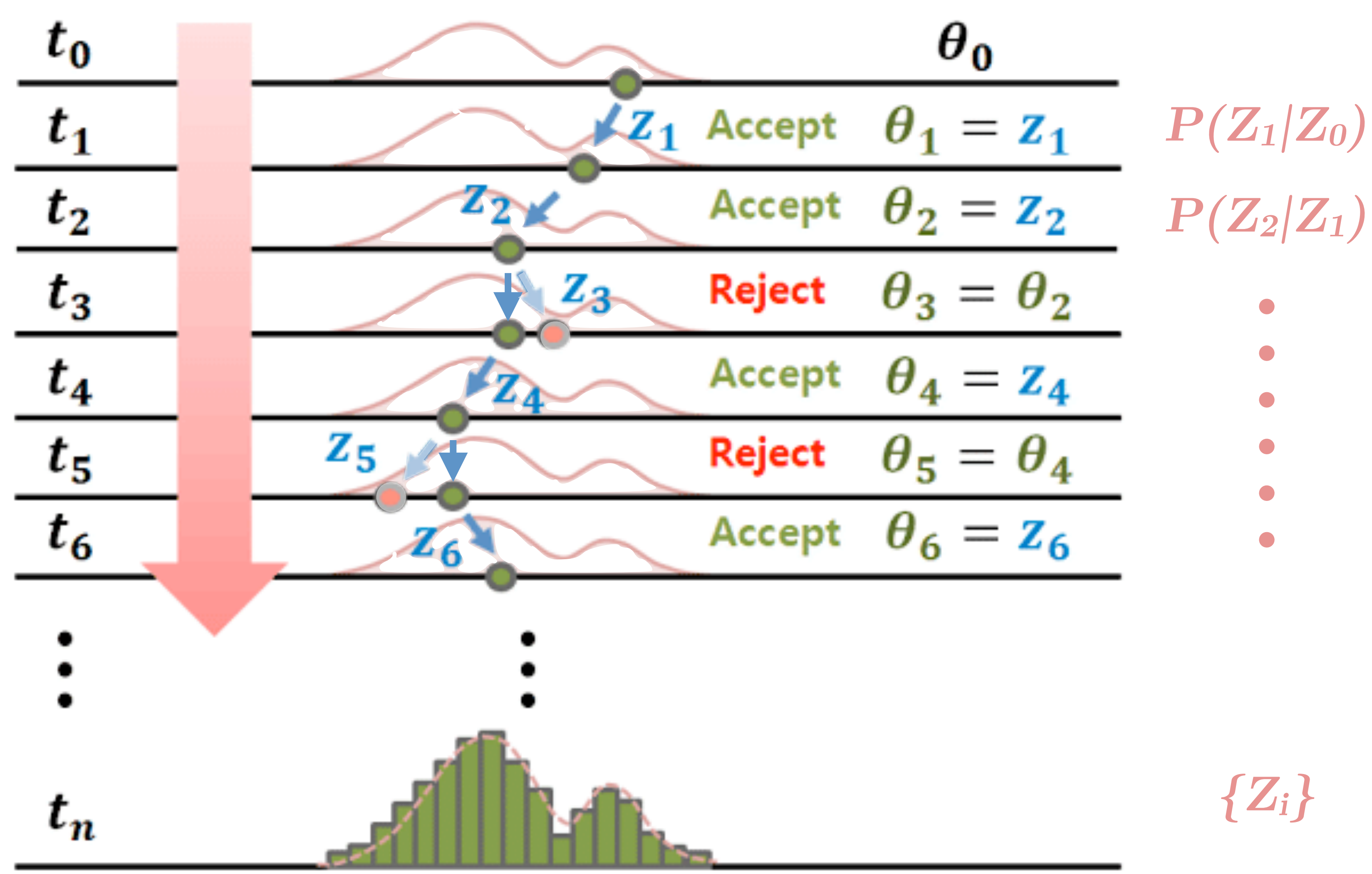
MH algorithm: an exemple

- **Demonstration for target distribution = Gaussian distribution** $G(x)$
  - First step $i = 1$: start with a random choice of hypothetical value x_i
  - Propose a new step $i + 1$: using the *jump function* $J(x_i + 1 \,|\, x_i)$
  - Compute the Metropolis-Hastings ratio r_{MH}: $r_{MH} = \dfrac{G(x_{i+1})\ J(x_i \,|\, x_{i+1})}{G(x_i)\ J(x_{i+1} \,|\, x_i)}$
  - Apply the *acceptance function* $A(x_{i+1}, x_i) = min\{1,\ r_{MH}\}$
    Equivalent to:
    - $r_{MH} \geq 1 \rightarrow$ accept step $i + 1$
    - $r_{MH} < 1 \rightarrow$ throw a number $u \in \sqcap\,(0{,}1)$
      $r_{MH} \geq u \rightarrow$ accept step $i + 1$
      $r_{MH} < u \rightarrow$ reject step $i + 1$
      count again step i
  - Iterate process until obtaining enough step to analyse the distribution



MH algorithm: an exemple

- **Case with local minima**
  - Acceptance function : $A(x_{i+1}, x_i) = min\{1,\ r_{MH}\}$
  - Can accept steps where $r_{MH} < 1$: can sample minima



Jin, Seung-Seop & Ju, Heekun & Jung, Hyung-Jo. (2019). Adaptive Markov chain Monte Carlo algorithms for Bayesian inference: recent advances and comparative study. Structure and Infrastructure Engineering. 10.1080/15732479.2019.1628077.

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \;=\; min(1,\ r)$

$$r \geq 1 \rightarrow A(x_{i+1}, x_i) \;=\; 1$$

$$r < 1 \rightarrow A(x_{i+1}, x_i) \;=\; r$$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \ = \ min(1, \ r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1} \,|\, x_i) = J(x_{i+1} \,|\, x_i) \, A(x_{i+1}, x_i)$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \ = \ min(1,\ r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1} \,|\, x_i) = J(x_{i+1} \,|\, x_i)\, A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

    $G(x_i)\ \ T(x_{i+1} \,|\, x_i) \ = G(x_i)\ \ J(x_{i+1} \,|\, x_i)\ \ A(x_{i+1}, x_i)$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \; = \; min(1, \; r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1} \,|\, x_i) = J(x_{i+1} \,|\, x_i) \, A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

$$
\begin{aligned}
G(x_i) \;\; T(x_{i+1} \,|\, x_i) \; &= G(x_i) \;\; J(x_{i+1} \,|\, x_i) \;\; A(x_{i+1}, x_i) \\
&= G(x_i) \;\; J(x_{i+1} \,|\, x_i) \;\; min(1, \; r)
\end{aligned}
$$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \;=\; min(1,\ r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1}\,|\,x_i) = J(x_{i+1}\,|\,x_i)\,A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

$$
\begin{aligned}
G(x_i)\ \ T(x_{i+1}\,|\,x_i) &= G(x_i)\ \ J(x_{i+1}\,|\,x_i)\ \ A(x_{i+1}, x_i) \\
&= G(x_i)\ \ J(x_{i+1}\,|\,x_i)\ \ min(1,\ r) \\
&= G(x_i)\ \ J(x_{i+1}\,|\,x_i)\ \ min(1,\ \frac{G(x_{i+1})\ \ J(x_i\,|\,x_{i+1})}{G(x_i)\ \ J(x_{i+1}\,|\,x_i)}\ \ )
\end{aligned}
$$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \;=\; min(1,\; r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1} \,|\, x_i) = J(x_{i+1} \,|\, x_i)\; A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

$$
\begin{aligned}
G(x_i)\;\; T(x_{i+1} \,|\, x_i) &= G(x_i)\;\; J(x_{i+1} \,|\, x_i)\;\; A(x_{i+1}, x_i) \\
&= G(x_i)\;\; J(x_{i+1} \,|\, x_i)\;\; min(1,\; r) \\
&= G(x_i)\;\; J(x_{i+1} \,|\, x_i)\;\; min(1,\; \frac{G(x_{i+1})\;\; J(x_i \,|\, x_{i+1})}{G(x_i)\;\; J(x_{i+1} \,|\, x_i)}\;) \\
&= \; min(G(x_i)\; J(x_{i+1} \,|\, x_i),\; G(x_{i+1})\; J(x_i \,|\, x_{i+1})\;)
\end{aligned}
$$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \; = \; min(1, \; r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1} \,|\, x_i) = J(x_{i+1} \,|\, x_i) \, A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

$$
\begin{aligned}
G(x_i) \;\; T(x_{i+1} \,|\, x_i) &= G(x_i) \;\; J(x_{i+1} \,|\, x_i) \;\; A(x_{i+1}, x_i) \\
&= G(x_i) \;\; J(x_{i+1} \,|\, x_i) \;\; min(1, \; r) \\
&= G(x_i) \;\; J(x_{i+1} \,|\, x_i) \;\; min(1, \; \frac{G(x_{i+1}) \;\; J(x_i \,|\, x_{i+1})}{G(x_i) \;\; J(x_{i+1} \,|\, x_i)} \;) \\
&= \; min(G(x_i) \;\; J(x_{i+1} \,|\, x_i), \; G(x_{i+1}) \;\; J(x_i \,|\, x_{i+1}) \;) \\
&= \; G(x_{i+1}) \;\; J(x_i \,|\, x_{i+1}) \, A(x_i, x_{i+1})
\end{aligned}
$$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \;=\; min(1,\ r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1}\,|\,x_i) = J(x_{i+1}\,|\,x_i)\,A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

$$\begin{aligned}
G(x_i)\;\; T(x_{i+1}\,|\,x_i) &= G(x_i)\;\; J(x_{i+1}\,|\,x_i)\;\; A(x_{i+1}, x_i) \\
&= G(x_i)\;\; J(x_{i+1}\,|\,x_i)\;\; min(1,\ r) \\
&= G(x_i)\;\; J(x_{i+1}\,|\,x_i)\;\; min\left(1,\ \frac{G(x_{i+1})\;\; J(x_i\,|\,x_{i+1})}{G(x_i)\;\; J(x_{i+1}\,|\,x_i)}\right) \\
&= min\big(G(x_i)\;\; J(x_{i+1}\,|\,x_i),\ G(x_{i+1})\;\; J(x_i\,|\,x_{i+1})\big) \\
&= G(x_{i+1})\;\; J(x_i\,|\,x_{i+1})\,A(x_i, x_{i+1}) \\
&= G(x_{i+1})\;\; T(x_i\,|\,x_{i+1})
\end{aligned}$$

Detailed balance equation

- **The *detailed balance equation* ensure that the steps follow the target distribution**
  - The *acceptance function* is: $A(x_{i+1}, x_i) \;=\; min(1,\ r)$
  - Defining the probability to transition to the step x_{i+1}, i.e. the *transition probability*:

    $T(x_{i+1}\,|\,x_i) = J(x_{i+1}\,|\,x_i)\; A(x_{i+1}, x_i)$
  - We can derive the detailed balance equation:

    $G(x_i)\;\; T(x_{i+1}\,|\,x_i) \;=\; G(x_{i+1})\; T(x_i\,|\,x_{i+1})$
  - Interpretation: if we propose a step with $G(x_{i+1}) \;> G(x_i)$

    The acceptance function is: $A(x_{i+1}, x_i) \;=\; 1$

    The transition probability is: $T(x_i\,|\,x_{i+1}) = \dfrac{G(x_{i+1})}{G(x_i)}$

    → The probability to jump back on the previous step is proportional to the ratio of $G(x)$ value

Other algorithms

- **There exist many algorithms fulfilling the sampling conditions**
  - *Hamiltonian MCMC*: introduce gradient of sampled probability to propose more accepted steps. Can make the chain converge faster, at the expense of the time to compute the derivative of the target distribution.
  - *Gibbs sampling*: for multidimensional distributions hard to sample, sample 1-dimension conditional posterior probability

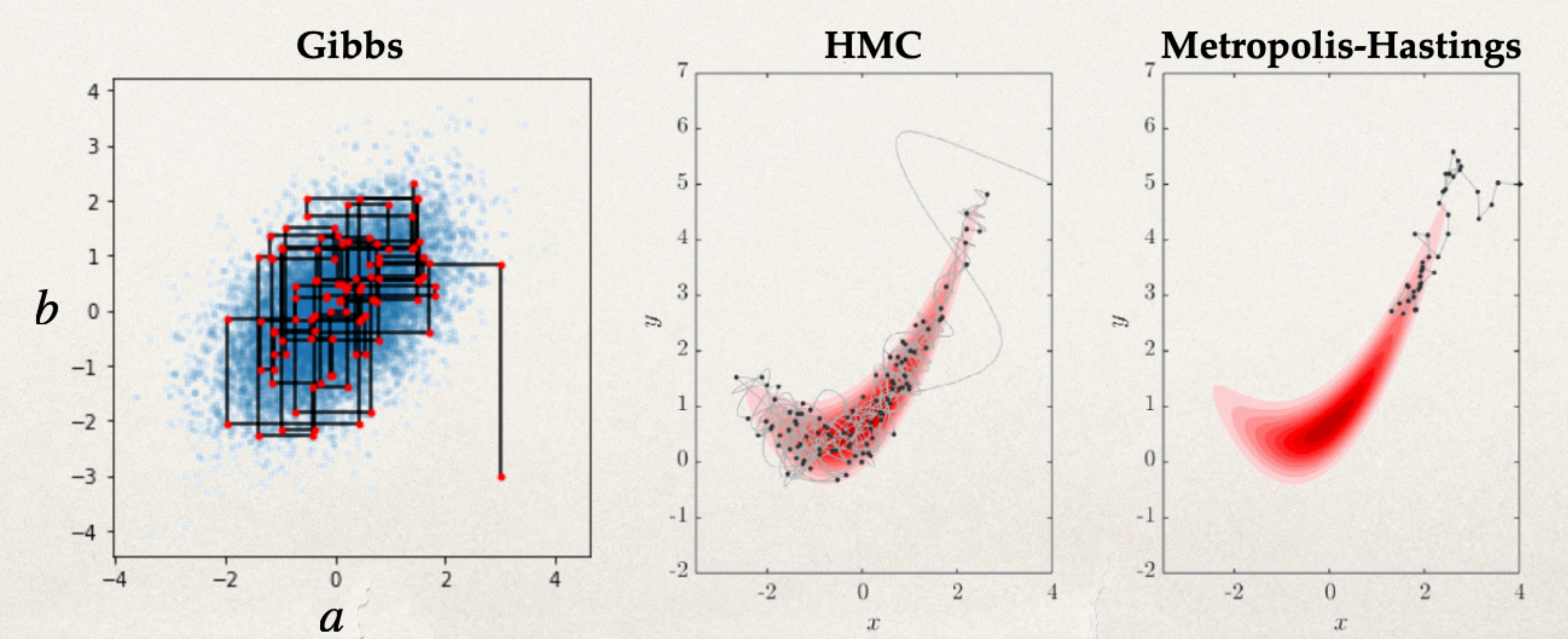


Diagram from Florian Ruppin

Other algorithms

- **There exist many algorithms fulfilling the sampling conditions**
  - *Hamiltonian MCMC*: introduce gradient of sampled probability to propose more accepted steps. Can make the chain converge faster, at the expense of the time to compute the derivative of the target distribution.
  - *Gibbs sampling*: for multidimensional distributions hard to sample, sample 1-dimension conditional posterior probability
  - *Nested sampling*: map the multidimensional distribution into a 1-dimensional case with a set of live points scanning the distribution to sample

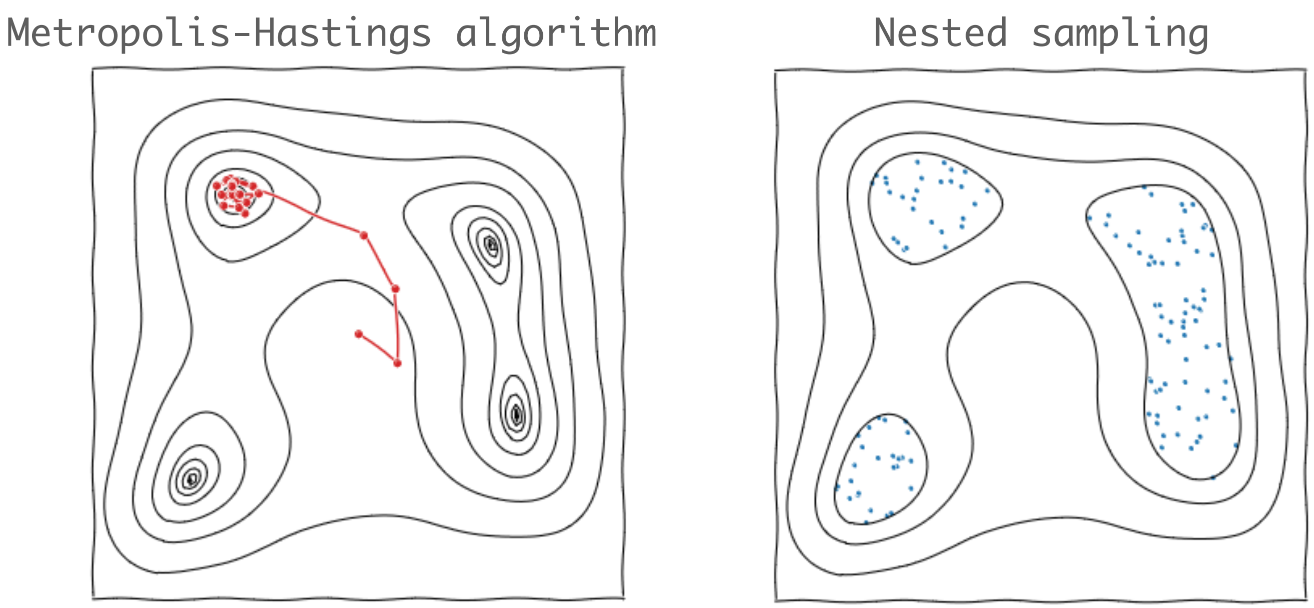


Diagram by Will Handley

Application for Bayesian inference

A reminder of Bayes theorem

- **Derivation from conditional probabilities**
  - Probability to observe A and B:

$$P(A \cap B) \;=\; P(A)\; P(B\,|\,A) \;=\; P(B)\; P(A\,|\,B)$$

$$\Rightarrow\; P(A\,|\,B) \;=\; \frac{P(B\,|\,A)\; P(A)}{P(B)}$$

- **Physical interpretation**
  - To perform Bayesian inference, we interpret *A* as the hypothesis *H,* and *B* as the data *D:*

Probability of observing the data D according to hypothesis H = "likelihood"

Probability of the hypothesis H = "prior probability"

$$\boxed{P(H\,|\,D)} = \frac{\boxed{P(D\,|\,H)}\;\boxed{P(H)}}{\boxed{P(D)}}$$

Probability of the hypothesis H given the data D = "posterior probability on H"

Probability of the data D independently of the hypothesis H = " evidence"

A reminder of Bayes theorem

- **Bayesian inference is the process of updating the probability on a statement**
  - Evaluation of the posterior probability on H according the data D
  - Bayes theorem reweighs the prior probability according to the likelihood
  - Also referred to as "updating belief on H "

- **Physical interpretation**
  - To perform Bayesian inference, we interpret A as the hypothesis H, and B as the data D:

Probability of observing the data D according to hypothesis H = "likelihood"

Probability of the hypothesis H = "prior probability"

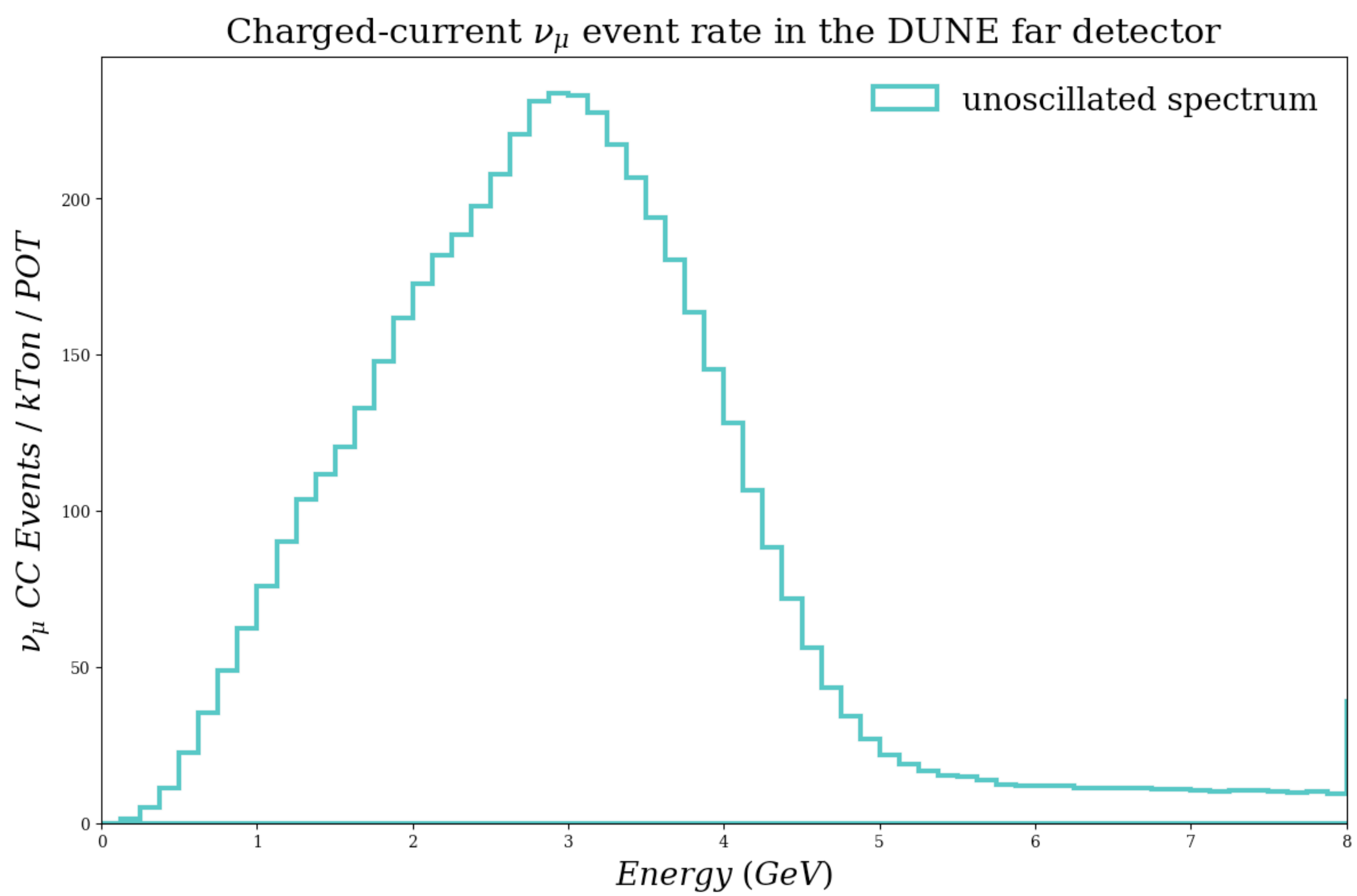
$$\boxed{P(H|D)} = \frac{\boxed{P(D|H)}\,\boxed{P(H)}}{\boxed{P(D)}}$$

Probability of the hypothesis H given the data D = "posterior probability on H"

Probability of the data D independently of the hypothesis H = " evidence"

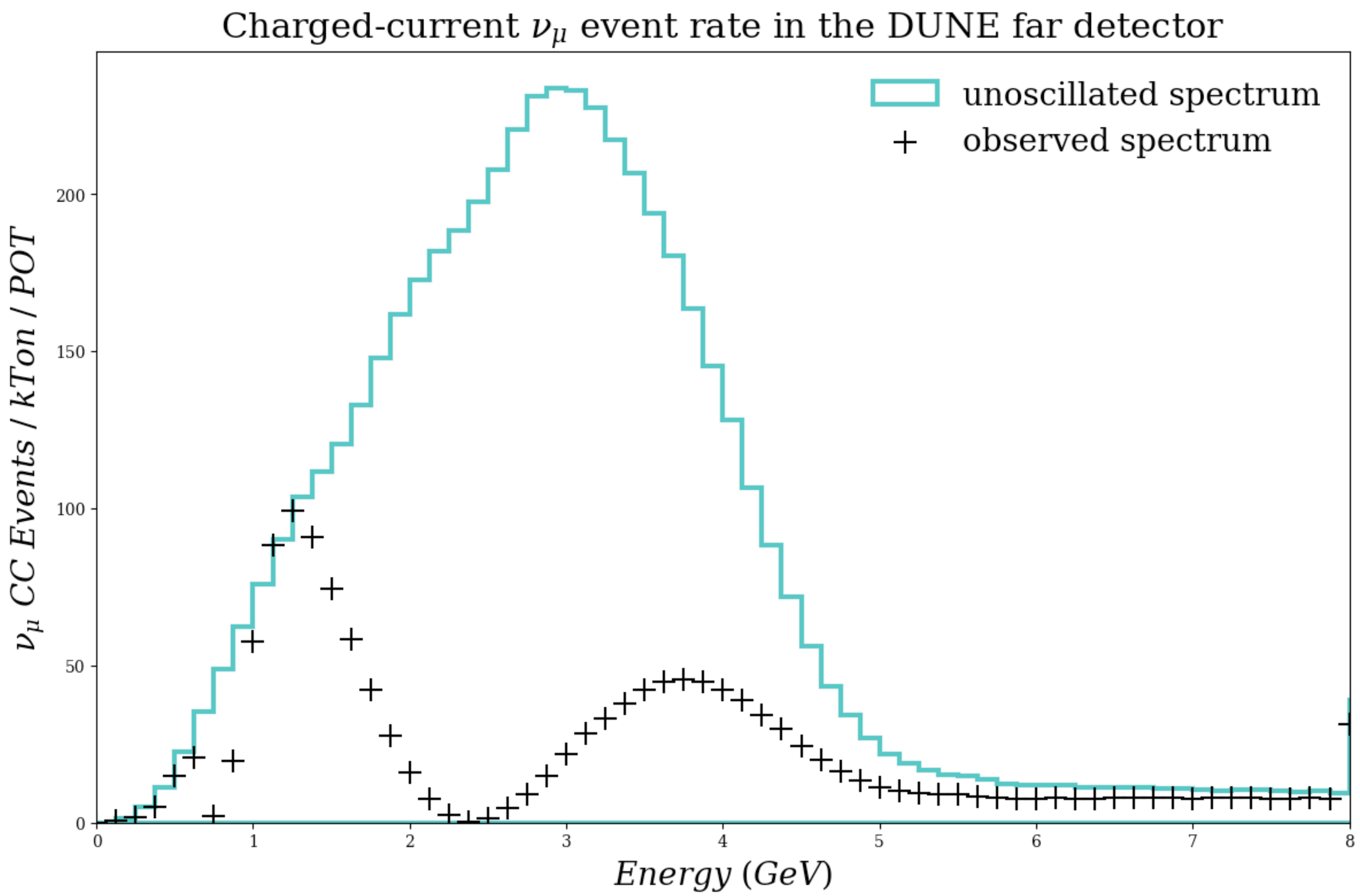
Bayesian inference: example

- **Example from neutrino physics**
  - We create a beam of ν_μ with a known spectrum shape



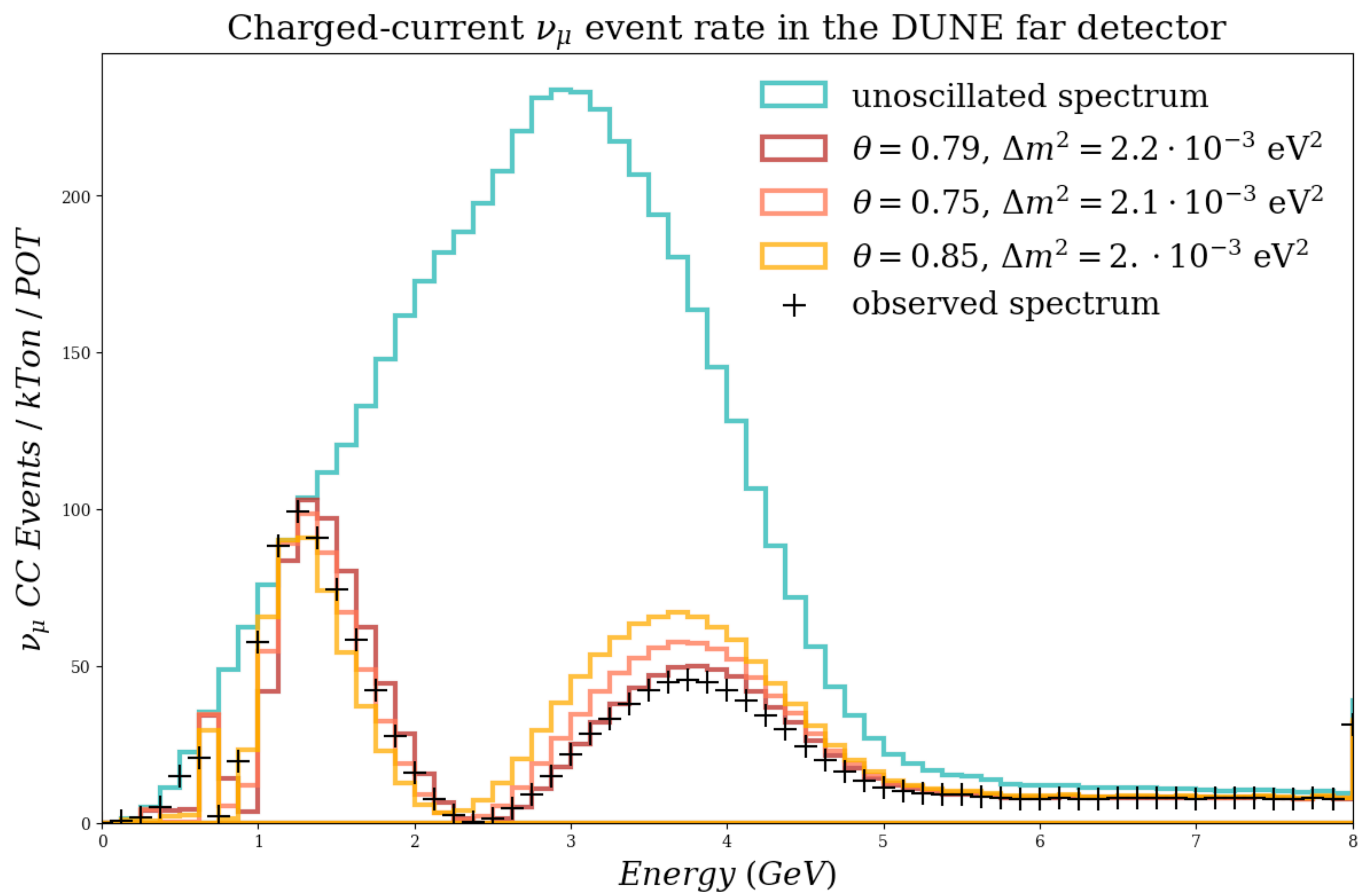
Bayesian inference: example

- **Example from neutrino physics**
  - We create a beam of ν_μ with a known spectrum shape
  - We observe the ν_μ spectrum after a certain time and realise that some ν_μ are missing
    $\rightarrow$ ν_μ have oscillated into ν_e or ν_τ



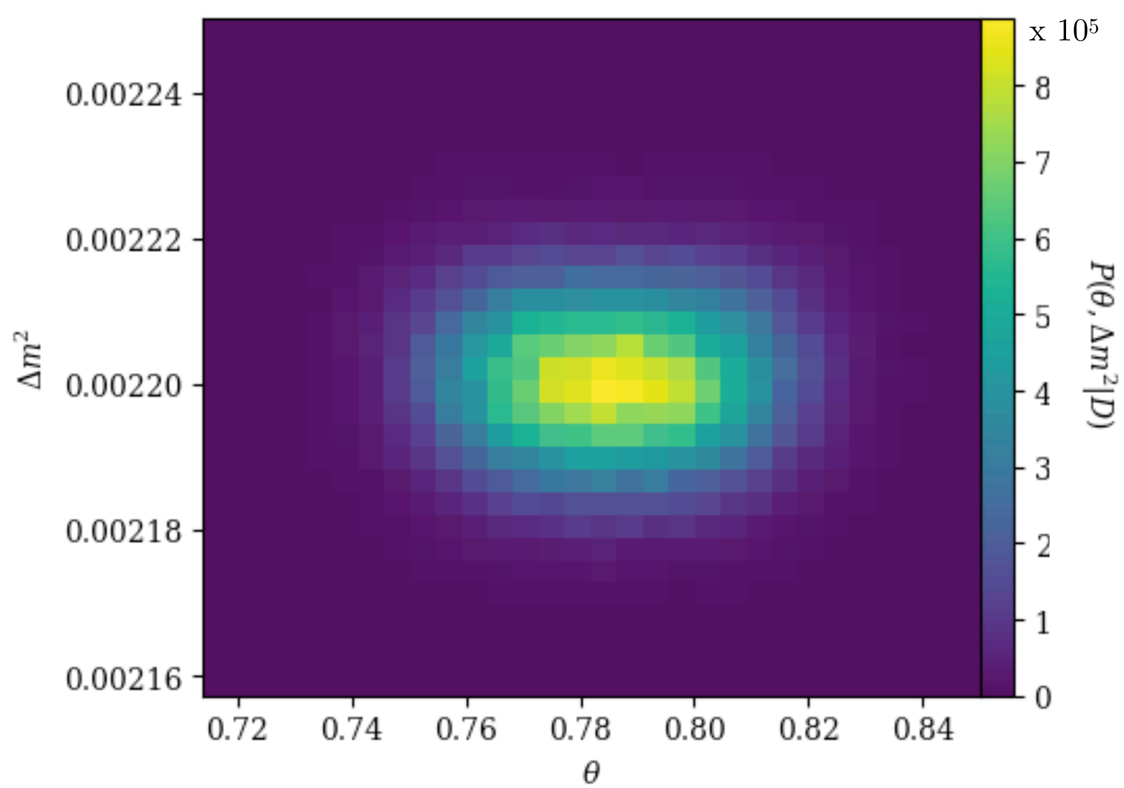
Bayesian inference: example

- **Example from neutrino physics**
  - We create a beam of ν_μ with a known spectrum shape
  - We observe the ν_μ spectrum after a certain time and realise that some ν_μ are missing → ν_μ have oscillated into ν_e or ν_τ
  - The oscillation probability depends on the parameters θ and Δm^2
  - We simulate the expected spectrum with several values of $\{\theta_i\}$ and $\{\Delta m_i^2\}$



Bayesian inference: example

- **Example from neutrino physics**
  - We create a beam of ν_μ with a known spectrum shape
  - We observe the ν_μ spectrum after a certain time and realise that some ν_μ are missing → ν_μ have oscillated into ν_e or ν_τ
  - The oscillation probability depends on the parameters θ and Δm^2
  - We simulate the expected spectrum with several values of $\{\theta_i\}$ and $\{\Delta m_i^2\}$
  - We compute the posterior probability for all the parameter values
  - The measured value correspond to the highest posterior probability



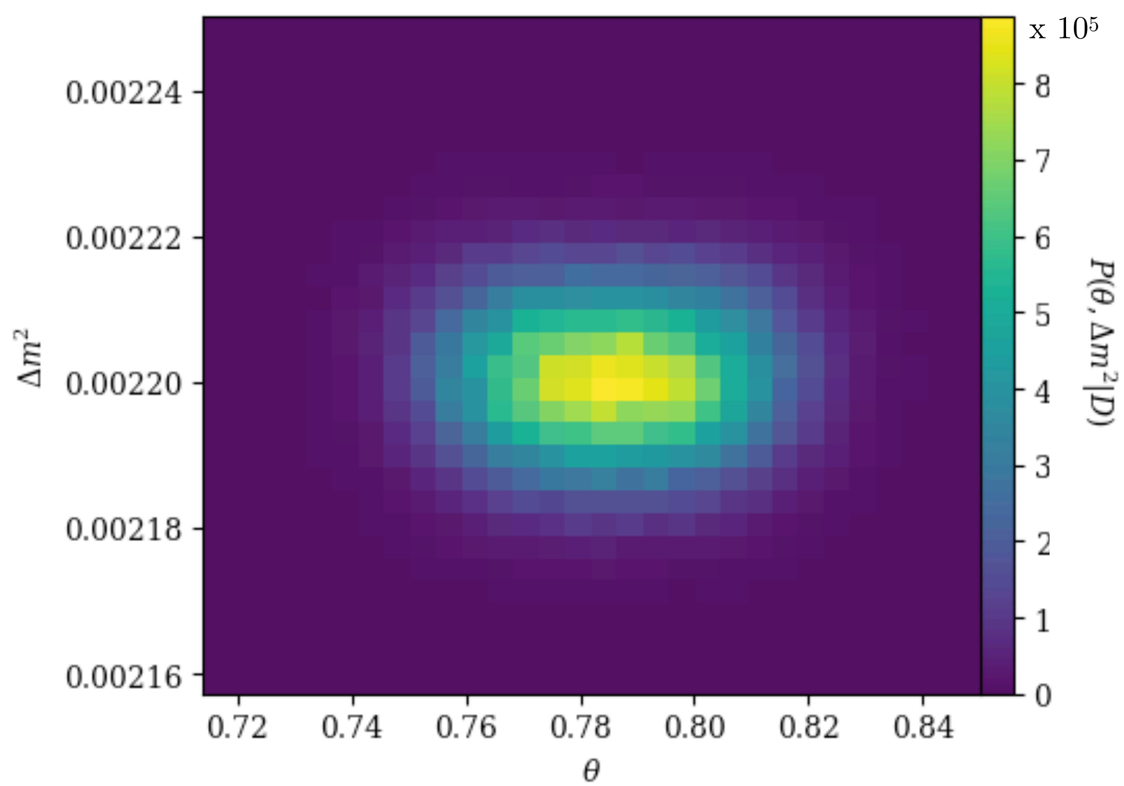
Bayesian inference: example

- **Example from neutrino physics**
  - We create a beam of ν_μ with a known spectrum shape
  - We observe the ν_μ spectrum after a certain time and realise that some ν_μ are missing → ν_μ have oscillated into ν_e or ν_τ
  - The oscillation probability depends on the parameters θ and Δm^2
  - We simulate the expected spectrum with several values of $\{\theta_i\}$ and $\{\Delta m_i^2\}$
  - We compute the posterior probability for all the parameter values
  - The measured value correspond to the highest posterior probability

Alternative option: a gradient descent towards the negative likelihood between the simulated spectrum and the data, and choosing the measured value as the minimal value

How to sample the space?

- **The grid option**
  - 2 parameters: we define a grid along the possible value and estimate $P(\theta, \Delta m^2 | D)$
  - Issue: incorporating the systematical uncertainties $\vec{\xi}$ (due to our limited knowledge on the flux, the interaction process, the detector response…)
    → need to be evaluated for each possible value of $\{\theta_i\}$ and $\{\Delta m_i^2\}$
    → the posterior we need is actually $P(\theta, \Delta m^2, \vec{\xi} | D)$
  - Grid searches become computationally expensive



How to sample the space?

- **The grid option**
  - 2 parameters: we define a grid along the possible value and estimate $P(\theta, \Delta m^2 | D)$
  - Issue: incorporating the systematical uncertainties $\vec{\varsigma}$ (due to our limited knowledge on the flux, the interaction process, the detector response…)
    → need to be evaluated for each possible value of $\{\theta_i\}$ and $\{\Delta m_i^2\}$
    → the posterior we need is actually $P(\theta, \Delta m^2, \vec{\varsigma} | D)$
  - Grid searches become computationally expensive

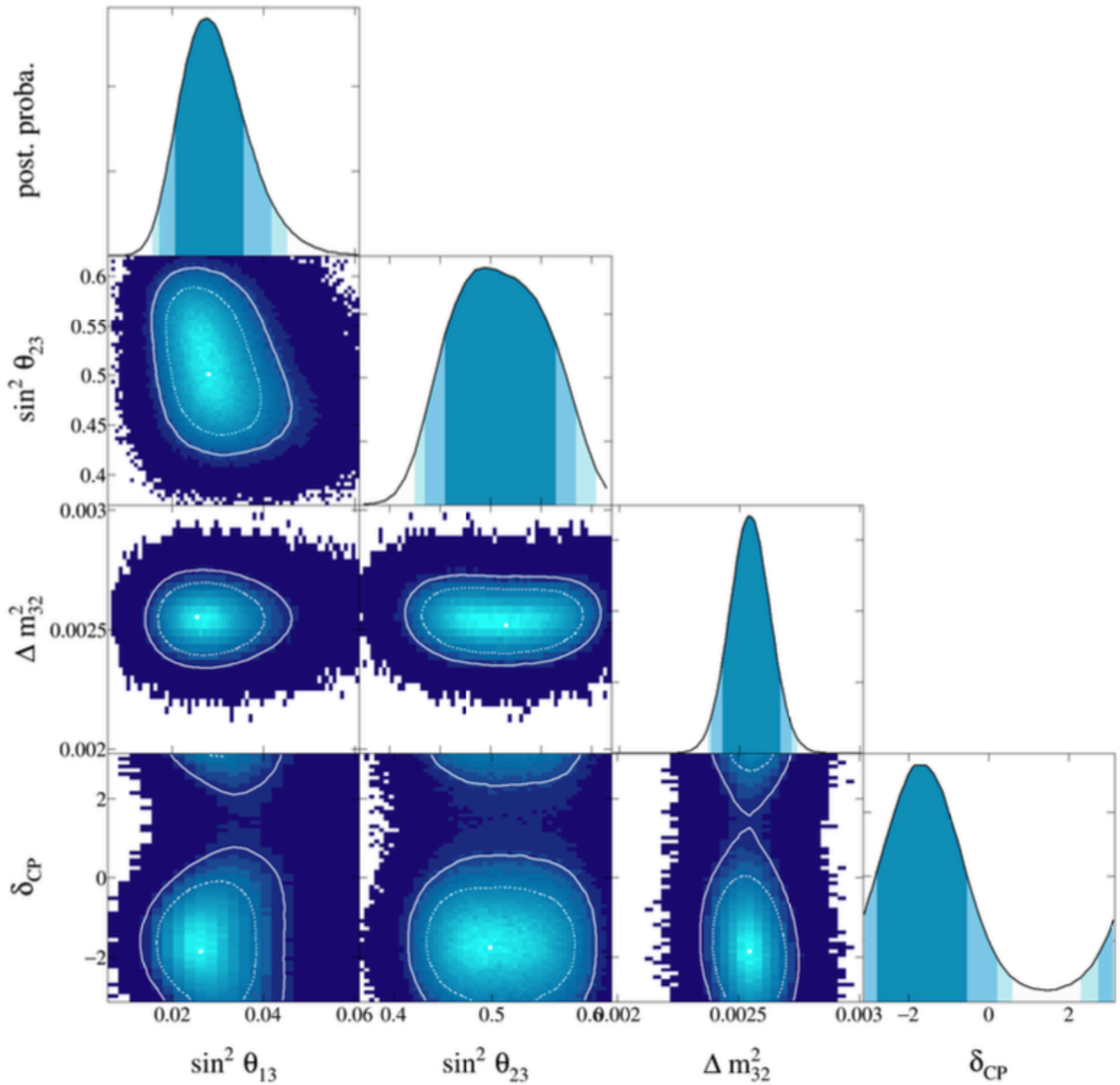
- **Markov Chain Monte Carlo (MCMC) option**
  - Grid searches spend the same time on all points of the posterior distribution
  - If we define the posterior distribution as the target function for a Markov chain, *the chain will visit each point of the distribution with a frequency proportional to its probability*
  - More suitable for high-dimensional distributions
  - Many packages exist in python (emcee, pymc)

MCMC applied to particle physics

- **An exemple of MCMC to sample neutrino oscillation parameters posterior probabilities**
  - The target distribution is the posterior probability on the oscillation parameters $\overrightarrow{\vartheta}$ and systematics parameters $\vec{\zeta}$: $P(\vartheta, \vec{\zeta} \mid D)$
  - All parameters are treated the same by the Markov chain:
    a state i is defined by a value of $\overrightarrow{\vartheta(i)}$ and $\overrightarrow{\varsigma(i)}$
  - The parameters can have different prior probabilities:
    → uniform is often chosen if no a priori knowledge
    → Gaussian if the parameter has been previously estimated
    → other option exist (Jeffrey priors, etc)

MCMC applied to particle physics

- **3-ν oscillation case:**
  - 4 oscillation parameters to estimate
  - $\mathcal{O}(100)$ systematics parameters
  - Results using T2K data

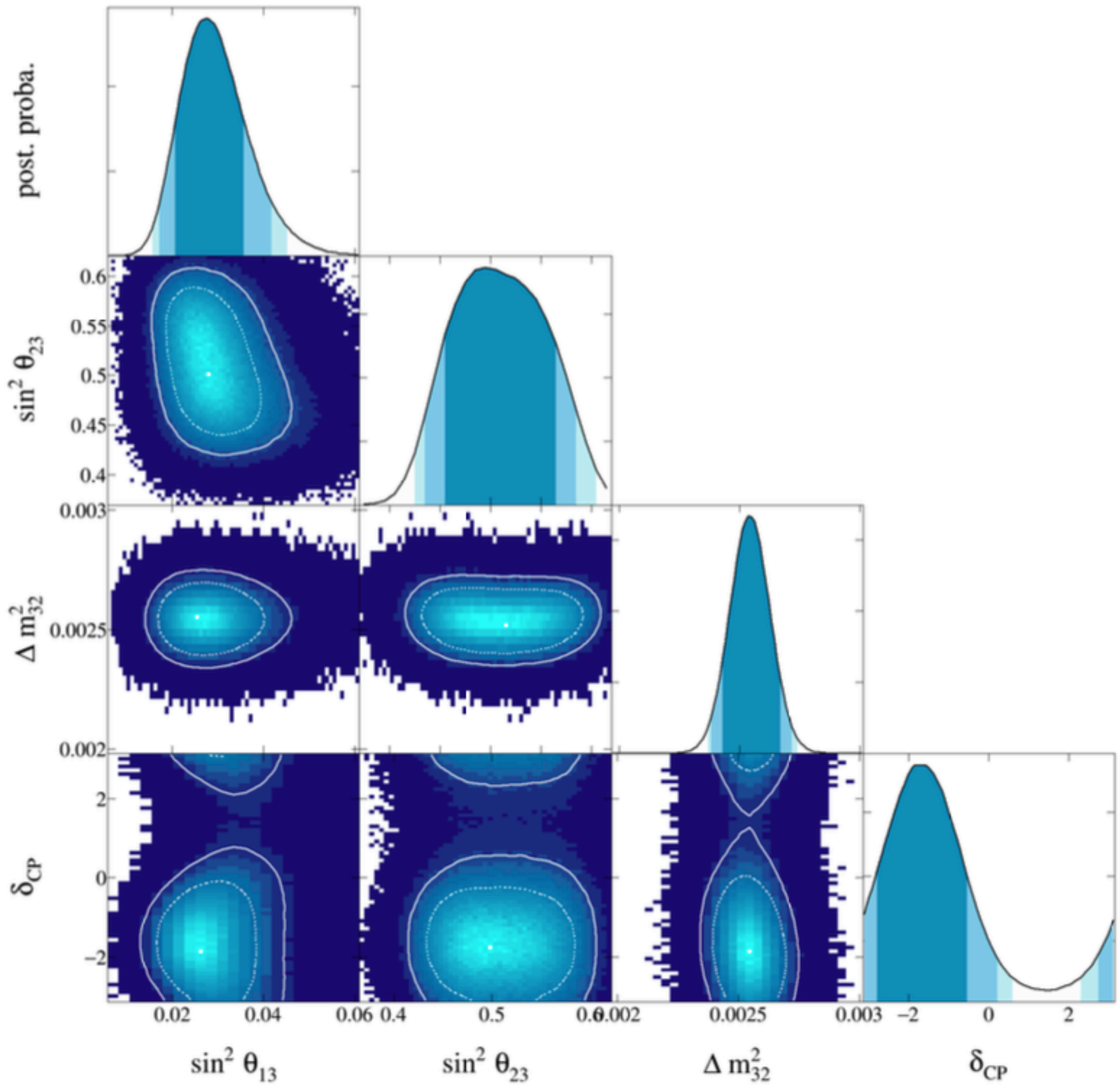


L.Haegel. Measurement of neutrino oscillation parameters using neutrino and antineutrino data of the T2K experiment. PhD thesis.

MCMC applied to particle physics

- **3-ν oscillation case:**
  - 4 oscillation parameters to estimate
  - $\mathcal{O}(100)$ systematics parameters
  - Results using T2K data

- **Where are the systematics?**
  - We marginalise over them!

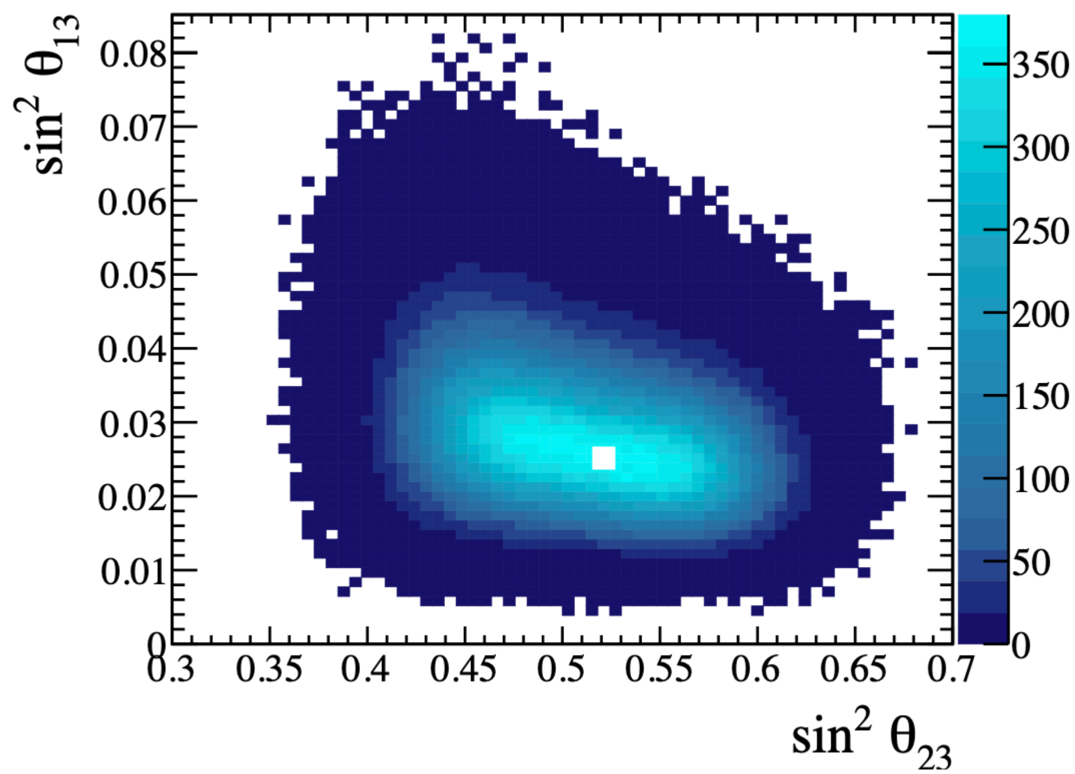


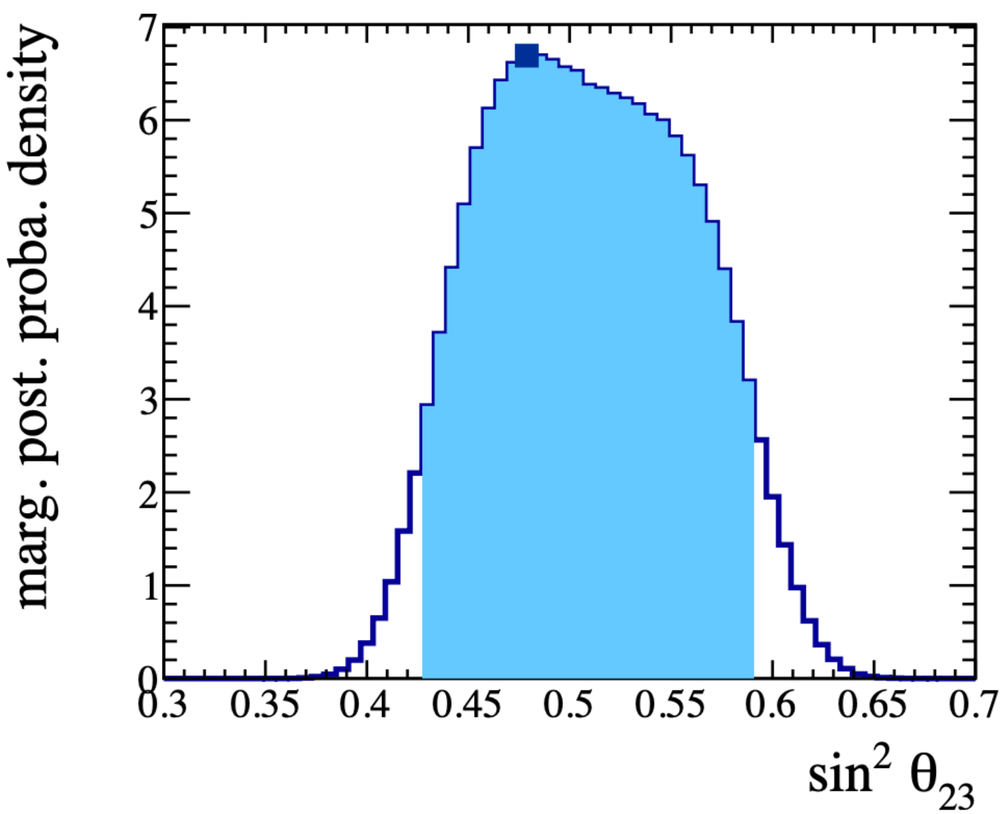
L.Haegel. Measurement of neutrino oscillation parameters using neutrino and antineutrino data of the T2K experiment. PhD thesis.

Marginalisation

- **Marginal posterior probability:**
  - When lowering the dimension of the sampled posterior, integrate the probability of the marginalised parameters

$$P(\vartheta\,|\,D) \;=\; \int P(\vartheta\,|\,\varsigma)\; P(\varsigma\,|\,D)\; d\varsigma$$

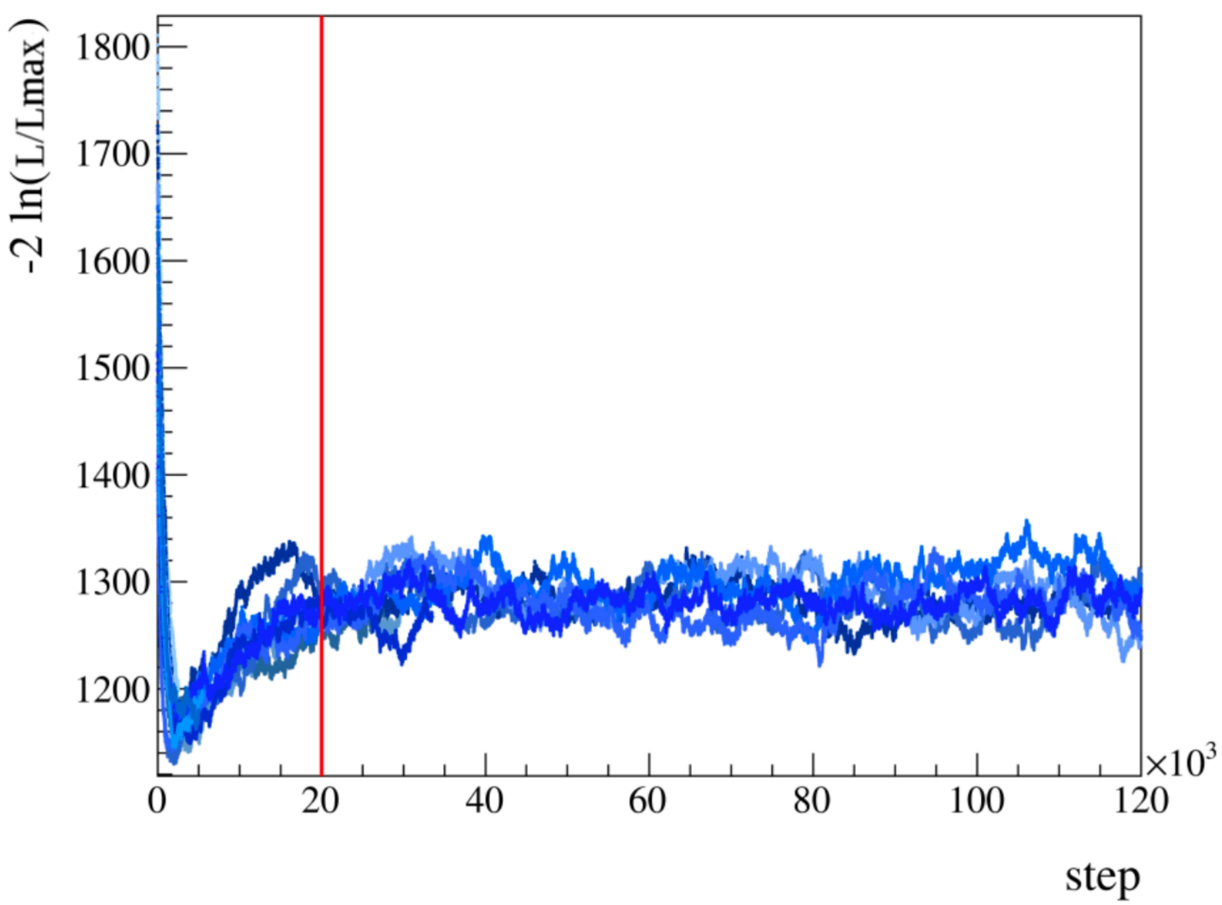




Convergence & burn-in

- **The crucial point: did the chain converge to the stationary distribution before being stopped?**
  - The chain can start far from the target distribution
  - A ergodic chain will reach the target distribution… eventually
  - How to ensure that you are in the stationary stage?

- **Look at the Markov chain trace**
  - Trace = value of the target distribution as a function of the step iteration
  - Sample around similar values at convergence
  - Steps before convergence must be discarded: called *burn-in*



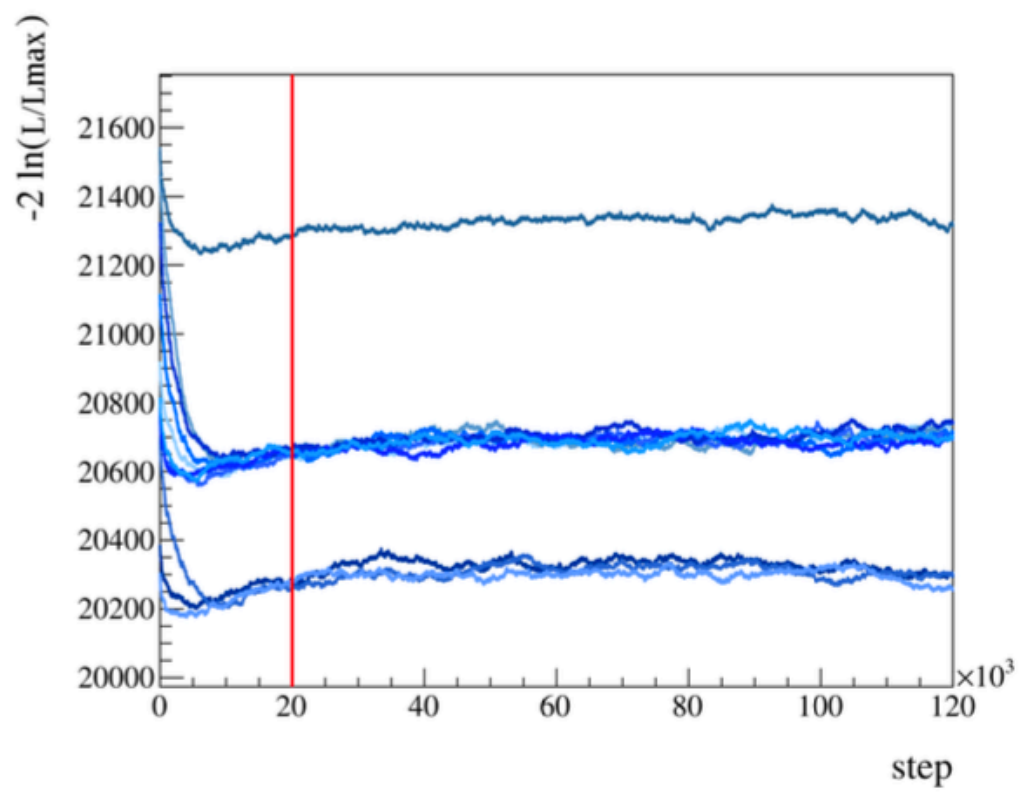
Convergence tests

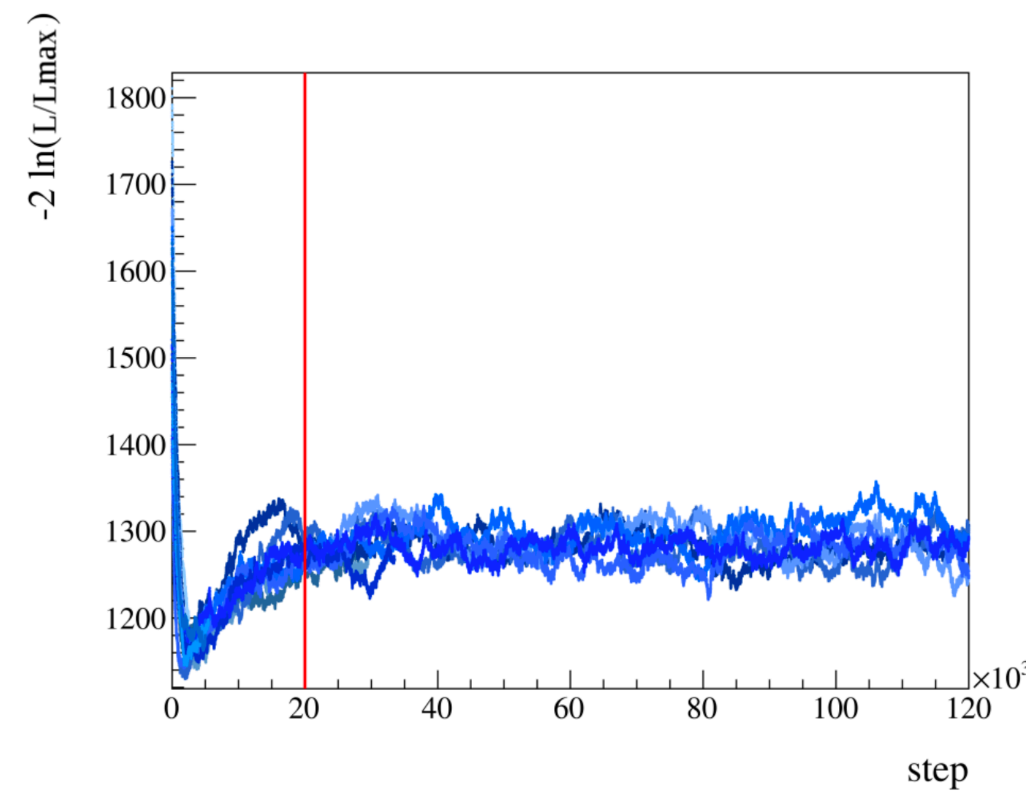
- **Ergodicity**
  - Are the chains spanning the entire value of parameter space?
  - Test: comparison of independent chains

Chains not properly tuned
Not ergodic

increase step size →

Chains properly tuned
Ergodic





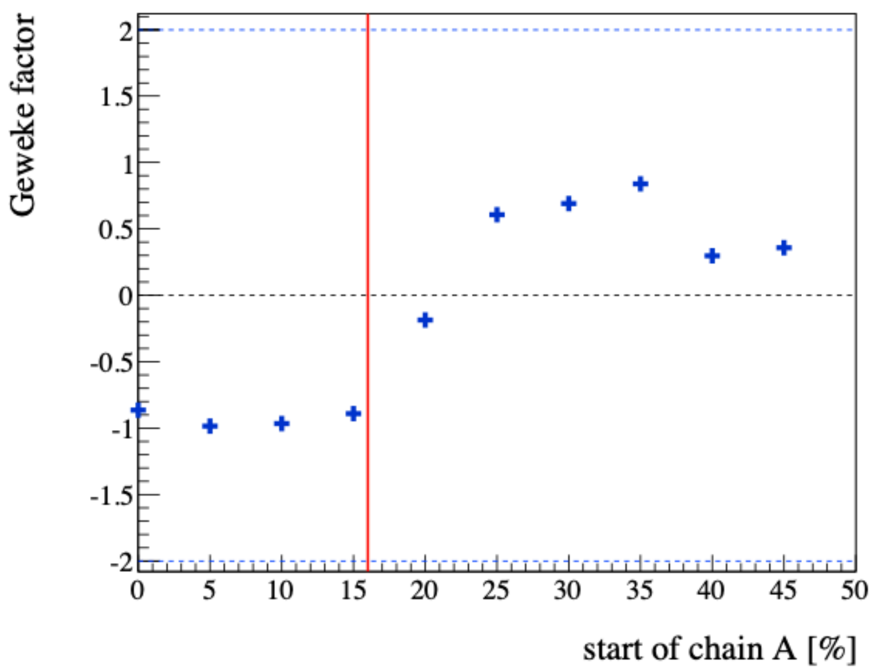
Convergence tests

- **Ergodicity**
  - Are the chains spanning the entire value of parameter space?
  - Test: comparison of independent chains

- **Geweke diagnostic**
  - Compare the beginning and the end of a Markov chain
  - Select 5% of the chain from its beginning and increment of 5%
    e.g. [0-5%], [5-10%], …, [45-50%]
    and compare with remaining 50% of the chain: [50-100%]
  - Useful to determine burn-in value and spot issues

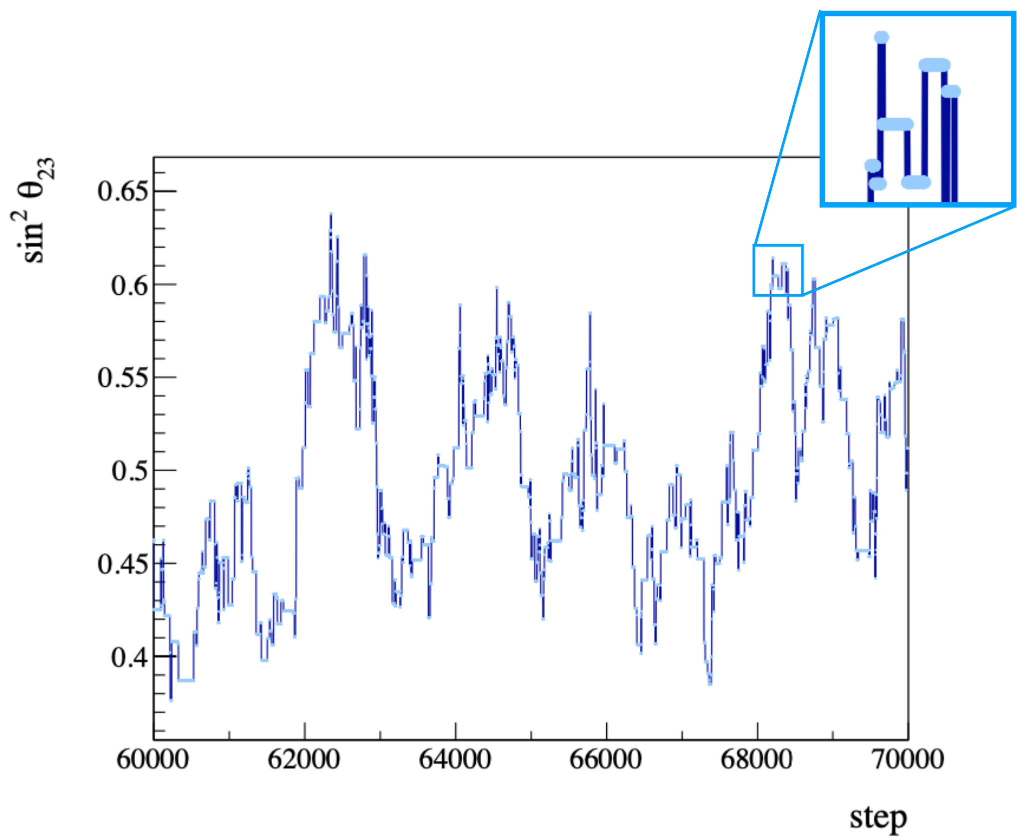
$$G = \frac{\bar{x}_{ini} - \bar{x}_{fin}}{\sqrt{\sigma(x)^2_{ini} + \sigma(x)^2_{fin}}}$$

Note: 5% is not a hard rule, other binning can be chosen

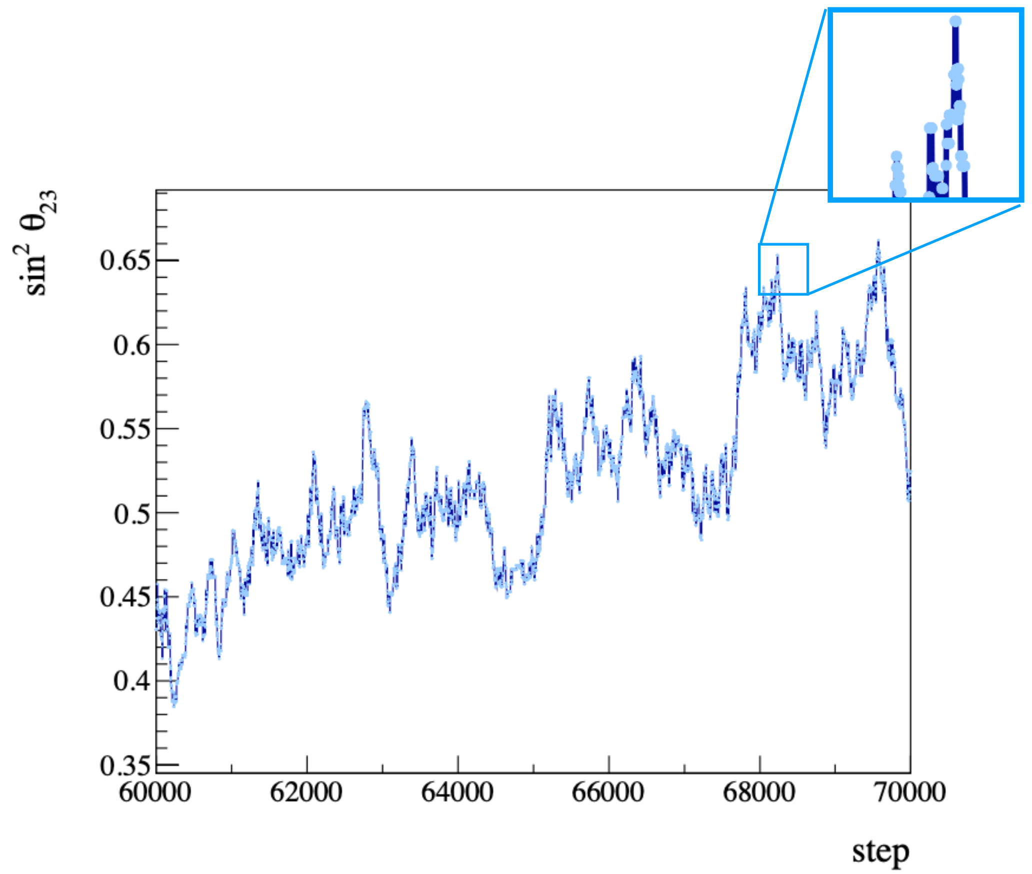


Step size

- **Jump function parameter**
  - The jump function can be symmetrical → Metropolis algorithm
    or asymmetrical → Hastings addition
  - The jump function has a width parameter:
    → this is referred to as the *step size*
    → its value is heuristic, although literature exist about its optimisation
    → strongly impacts the convergence rate of the chain



(c) $\sin^2\theta_{23}$, large step scale



(a) $\sin^2\theta_{23}$, correct step scale

Autocorrelation

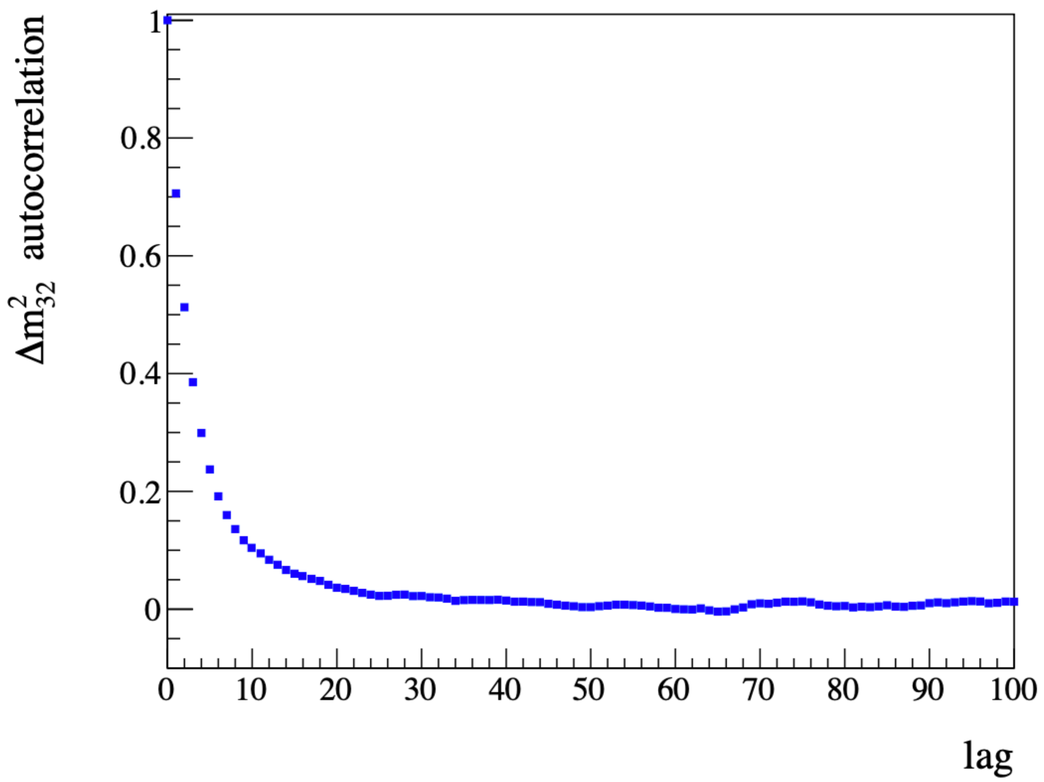
- **The steps are correlated between them**
  - Independent samples can be selected by subsampling the chain
  - Value of subsampling order can be determined from the autocorrelation function

$$\mathscr{A}(k) = \frac{\varrho(k)}{\varrho(0)}$$

where:

$$\varrho(k) = \mathbb{E}(x_i - \bar{x})\ \mathbb{E}(x_{i+k} - \bar{x})$$
$$= \frac{1}{N-k}\sum_{i}^{N-k}(x_i - \bar{x})(x_{i+k} - \bar{x})$$

$\mathbb{E}$ = expectation value



(b) Δm^2_{32}

Changing the prior

- **The posterior probability can be evaluated for a different definition of the prior**
  - Equivalent to a variable change of the distribution:
    prior in x → prior in $y = f(x)$
  - Need to evaluate the Jacobian of the transformation:

$$\begin{aligned} P(H(x)) \;\rightarrow\; P(H(y)) &= P(H(x))\,|J(y)| \\ &= P(H(x))\,\left|\frac{\partial x}{\partial y}\right| \end{aligned}$$

  - Can be extended to multi-variable cases

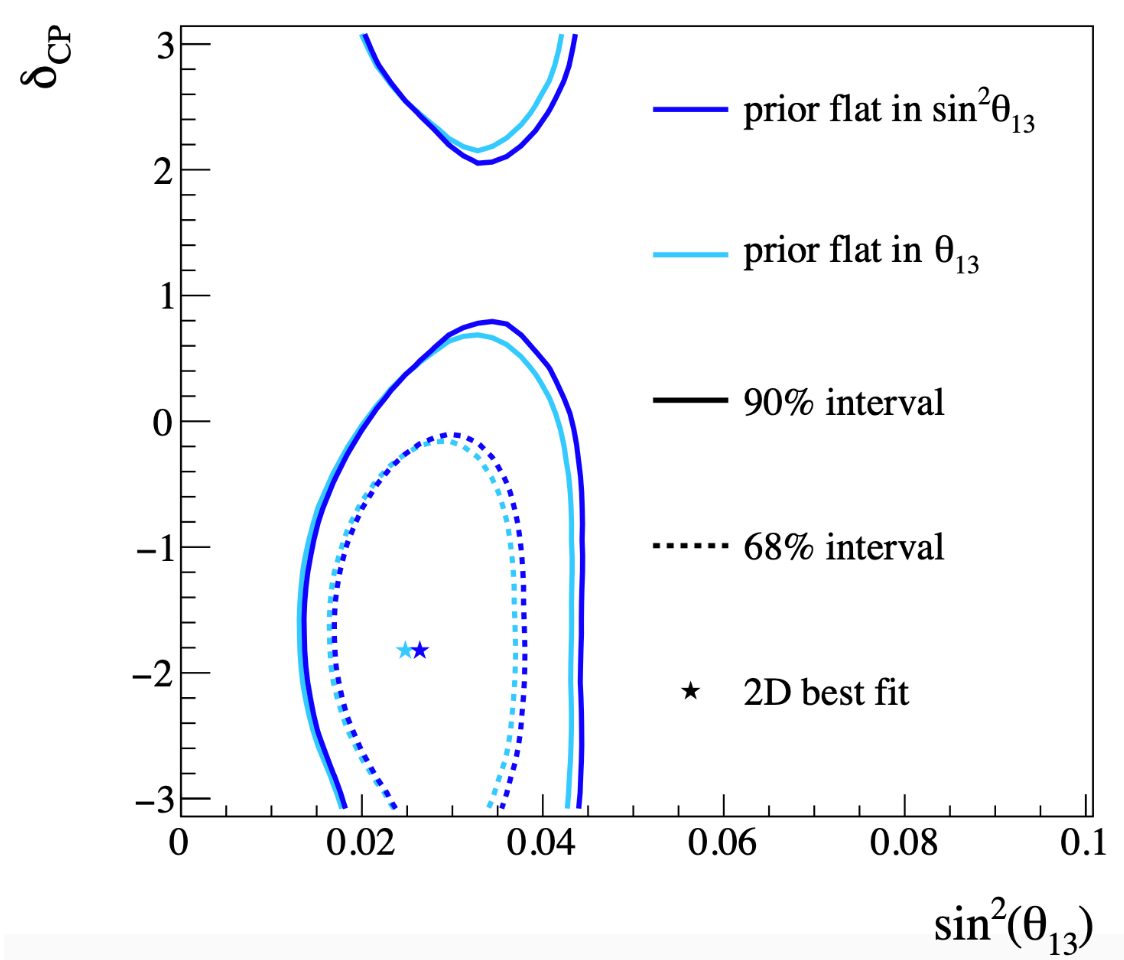
Changing the prior

- **The posterior probability can be evaluated for a different definition of the prior**
  - Equivalent to a variable change of the distribution: prior in x → prior in $y = f(x)$
  - Need to evaluate the Jacobian of the transformation:

$$P(H(x)) \rightarrow P(H(y)) = P(H(x))\,|J(y)|$$
$$= P(H(x))\,|\frac{\partial x}{\partial y}|$$

  - Can be extended to multi-variable cases

- **A useful way to:**
  - Check the robustness of the prior



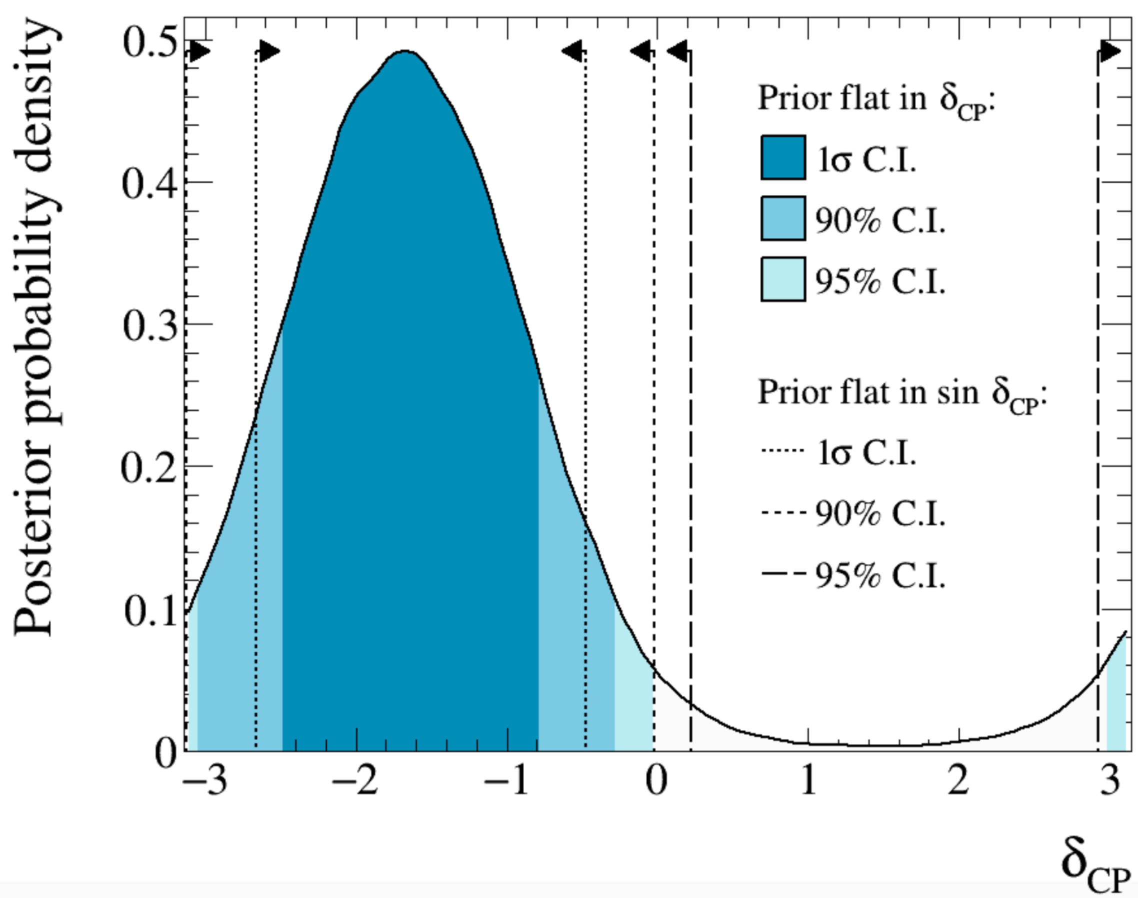
Changing the prior

- **The posterior probability can be evaluated for a different definition of the prior**
  - Equivalent to a variable change of the distribution:
    prior in x → prior in $y = f(x)$
  - Need to evaluate the Jacobian of the transformation:

$$\begin{aligned} P(H(x)) \;\rightarrow\; P(H(y)) &= P(H(x))\,|J(y)| \\ &= P(H(x))\,\left|\frac{\partial x}{\partial y}\right| \end{aligned}$$

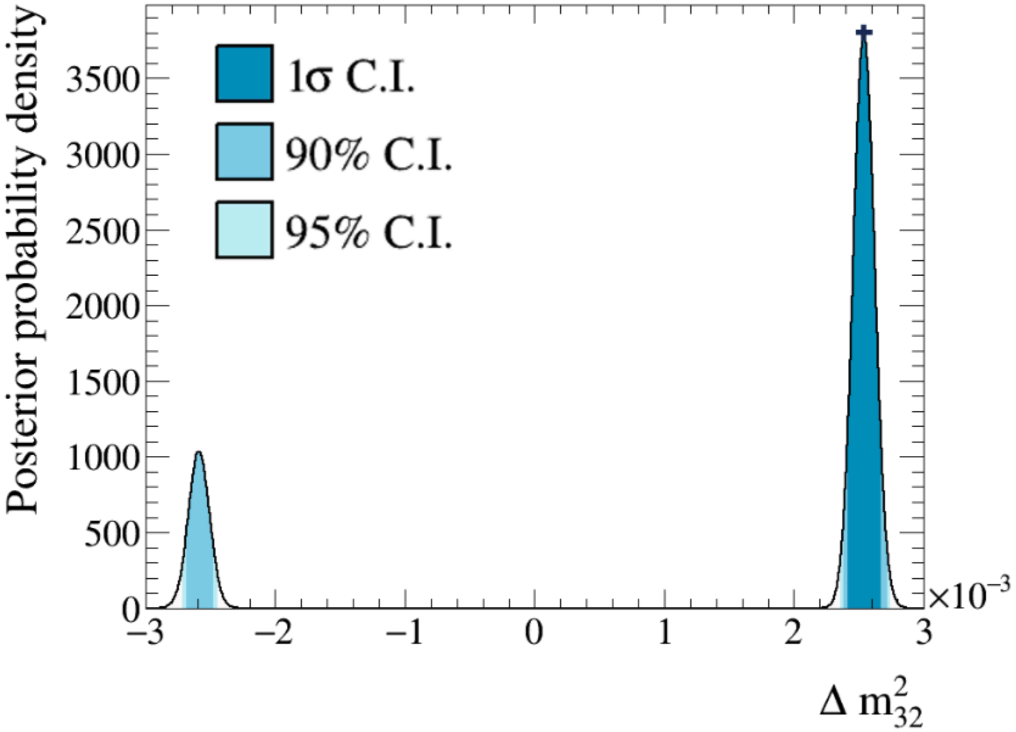
  - Can be extended to multi-variable cases

- **A useful way to:**
  - Check the robustness of the prior
  - Answer a different question
    e.g. what is the probability of CP-violation (instead of what is the δ_{CP} value)



Bayes factor

- **Comparison of 2 hypotheses**
  - If we have 2 hypotheses H_1 and H_2, we can compare them with the *Bayes factor*, i.e. the ratio of marginalised likelihood
  - Bayes factor: $B_F = \dfrac{P(D|H_1)}{P(D|H_2)}$
  - If the prior probabilities are the same, this is equivalent to the ratio of posterior probabilities
  - Example: the Bayes factor for normal ordering is $B_F = 3.72$ on this plot



Conclusion

- **Bayesian inference consist in computing a posterior probability density**
  - Update the probability of a hypothesis according to the information on the data
  - Markov Chain Monte-Carlo is a useful tool to sample high dimensional cases
  - Can infer any shape of posterior probabilities

- **The process requires careful tuning**
  - Asymptotically, MCMC properties ensure that it will converge to the target distribution
  - We do not have infinite time, neither an infinite number of processors
  - Ensuring convergence is key to the process
    → convincing ourselves that the output is the needed one is not easy!
  - Extensive literature about it, but no « one-solution-fit-all »
  - Does not mean it should no be used! But not blindly

Hands-on

- **Exercise 1:**
  - Simple Markov chain sampling example
  - Exercise, solution on Google Colab

- **Exercise 2:**
  - Bayesian inference with Markov chain example
  - Exercise, solution on Google Colab
  - Going further: reproduce with emcee or pyMC packages