MCMC algorithms: general principle

Approximate an integral (or another function) from a target distribution

MCMC algorithms: general principle

Approximate an integral (or another function) from a target distribution

⇒ sample a Markov chain whose stationary law is the target (such as the *posterior*) distribution, then apply the Monte Carlo method.

MCMC algorithms: general principle

Approximate an integral (or another function) from a target distribution

⇒ sample a Markov chain whose stationary law is the target (such as the *posterior*) distribution, then apply the Monte Carlo method.

Requires **two-fold convergence**:

1. the Markov chain must first converge to its stationary distribution:
   $\forall X_0 \, , \, X_n \xrightarrow[n \to +\infty]{\mathscr{L}} \tilde{p}$

MCMC algorithms: general principle

Approximate an integral (or another function) from a target distribution

⇒ sample a Markov chain whose stationary law is the target (such as the *posterior*) distribution, then apply the Monte Carlo method.

Requires **two-fold convergence**:

1. the Markov chain must first converge to its stationary distribution:

$$\forall X_0\,,\, X_n \xrightarrow[n\to+\infty]{\mathscr{L}} \tilde{p}$$

2. then Monte Carlo convergence must also happen:

$$\frac{1}{N}\sum_{i=1}^{N} f(X_{n+i}) \xrightarrow[N\to+\infty]{} \mathbb{E}[f(X)]$$

MCMC algorithms: general principle

Approximate an integral (or another function) from a target distribution

⇒ sample a Markov chain whose stationary law is the target (such as the *posterior*) distribution, then apply the Monte Carlo method.

Requires **two-fold convergence**:

1. the Markov chain must first converge to its stationary distribution:

$$\forall X_0, X_n \xrightarrow[n\to+\infty]{\mathscr{L}} \tilde{p}$$

2. then Monte Carlo convergence must also happen:

$$\frac{1}{N}\sum_{i=1}^{N} f(X_{n+i}) \xrightarrow[N\to+\infty]{} \mathbb{E}[f(X)]$$

$$\overbrace{X_0 \to X_1 \to X_2 \to \cdots \to X_n}^{\text{Markov chain convergence}} \to \overbrace{X_{n+1} \to X_{n+2} \to \cdots \to X_{n+N}}^{\text{Monte Carlo sample}}$$

General framework of MCMC algorithms

MCMC algorithms uses an acceptance-rejection framework

1. Initialise $x^{(0)}$
2. For $t = 1 \dots n+N$:
   a. Propose a new candidate $y^{(t)} \sim q(y^{(t)}|x^{(t-1)})$
   b. Accept $y^{(t)}$ with probability $\alpha(x^{(t-1)}, y^{(t)})$:
      $x^{(t)} := y^{(t)}$
      if $t > n$, "save" $x^{(t)}$ (as part of the final Monte Carlo sample)

where q is the instrumental distribution for proposing new samples and α is the acceptance probability.

Choosing the instrumental distribution

No absolute optimal choice for the instrumental distribution q proposing new samples

⇒ infinite possibilities: some better than others

Choosing the instrumental distribution

No absolute optimal choice for the instrumental distribution q proposing new samples

$\Rightarrow$ infinite possibilities: some better than others

To guaranty convergence towards the target $\tilde{p}$:

- the support of q has to cover the support of $\tilde{p}$
- q must not generate periodic values

Choosing the instrumental distribution

No absolute optimal choice for the instrumental distribution q proposing new samples

$\Rightarrow$ infinite possibilities: some better than others

To guaranty convergence towards the target $\tilde{p}$:

- the support of q has to cover the support of $\tilde{p}$
- q must not generate periodic values

NB: *ideally* q is **easy** and **fast** to **compute**