Markov chain Monte Carlo

Conditioning via MCMC

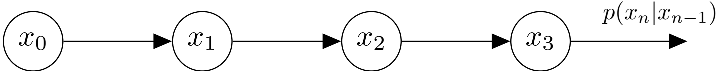
- **Problem**: Importance sampling degrades poorly as the dimension of the latent variables increases, unless we have a very good proposal $q(\mathbf{x})$.
  - Setting the proposal to the prior, as in likelihood weighting, has $q(\mathbf{x}) = p(\mathbf{x})$, which is rarely a very good choice.
  - A “good” proposal $q(\mathbf{x})$ will be very close to the posterior $p(\mathbf{x}|\mathbf{y})$, which might be quite different than the prior

Conditioning via MCMC

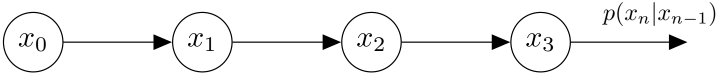
- **Problem**: Importance sampling degrades poorly as the dimension of the latent variables increases, unless we have a very good proposal $q(\mathbf{x})$.
  - Setting the proposal to the prior, as in likelihood weighting, has $q(\mathbf{x}) = p(\mathbf{x})$, which is rarely a very good choice.
  - A "good" proposal $q(\mathbf{x})$ will be very close to the posterior $p(\mathbf{x}|\mathbf{y})$, which might be quite different than the prior
- **Alternative**: Markov chain Monte Carlo (MCMC) methods draw samples from a target distribution by performing a biased random walk over the space of latent variables $\mathbf{x}$.

Conditioning via MCMC

- **Problem**: Importance sampling degrades poorly as the dimension of the latent variables increases, unless we have a very good proposal $q(\mathbf{x})$.
  - Setting the proposal to the prior, as in likelihood weighting, has $q(\mathbf{x}) = p(\mathbf{x})$, which is rarely a very good choice.
  - A "good" proposal $q(\mathbf{x})$ will be very close to the posterior $p(\mathbf{x}|\mathbf{y})$, which might be quite different than the prior
- **Alternative**: Markov chain Monte Carlo (MCMC) methods draw samples from a target distribution by performing a biased random walk over the space of latent variables $\mathbf{x}$.
- Idea: create a Markov chain such that the sequence of states $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \ldots$ are samples from $p(\mathbf{x}|\mathbf{y})$.



Markov chains (1/2)



- A **Markov chain** is a sequence with a "memoryless" property; that is, with the factorization

$$p(\mathbf{x}_n|\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}) = p(\mathbf{x}_n|\mathbf{x}_{n-1}).$$

(See Bishop PRML, Chapter 11.2.1 for more)

Markov chains (1/2)

$$x_0 \longrightarrow x_1 \longrightarrow x_2 \longrightarrow x_3 \xrightarrow{p(x_n|x_{n-1})}$$

- A **Markov chain** is a sequence with a "memoryless" property; that is, with the factorization

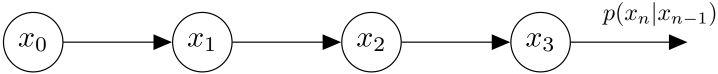
$$p(\mathbf{x}_n|\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}) = p(\mathbf{x}_n|\mathbf{x}_{n-1}).$$

- The marginal distribution of any n can be found by

$$p(\mathbf{x}_n) = \sum_{\mathbf{x}_{n-1}} p(\mathbf{x}_n|\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}).$$

(See Bishop PRML, Chapter 11.2.1 for more)

Markov chains (1/2)



- A **Markov chain** is a sequence with a "memoryless" property; that is, with the factorization
$$p(\mathbf{x}_n|\mathbf{x}_1,\ldots,\mathbf{x}_{n-1}) = p(\mathbf{x}_n|\mathbf{x}_{n-1}).$$
- The marginal distribution of any n can be found by
$$p(\mathbf{x}_n) = \sum_{\mathbf{x}_{n-1}} p(\mathbf{x}_n|\mathbf{x}_{n-1})p(\mathbf{x}_{n-1}).$$
- If this distribution is the same for all n, then it is said to be *homogeneous*, and we denote it as T, the transition probability $T(\mathbf{x}_n|\mathbf{x}_{n-1}) = p(\mathbf{x}_n|\mathbf{x}_{n-1})$.

(See Bishop PRML, Chapter 11.2.1 for more)

Markov chains (2/2)

- A distribution is **invariant** or **stationary** w.r.t. a Markov chain if each step leaves the distribution invariant. That is, a distribution $p^{\star}(\mathbf{x})$ is invariant for a homogeneous Markov Chain with transition $T(\mathbf{x}'|\mathbf{x})$ if,

$$p^{\star}(\mathbf{x}') = \sum_{\mathbf{x}} T(\mathbf{x}'|\mathbf{x}) p^{\star}(\mathbf{x}).$$

Markov chains (2/2)

- A distribution is **invariant** or **stationary** w.r.t. a Markov chain if each step leaves the distribution invariant. That is, a distribution $p^\star(\mathbf{x})$ is invariant for a homogeneous Markov Chain with transition $T(\mathbf{x}'|\mathbf{x})$ if,

$$p^\star(\mathbf{x}') = \sum_{\mathbf{x}} T(\mathbf{x}'|\mathbf{x}) p^\star(\mathbf{x}).$$

- A sufficient (but not necessary) condition for $p^\star$ to be invariant is if we choose the transition probabilities T to satisfy the **detailed balance** property,

$$p^\star(\mathbf{x}) T(\mathbf{x}'|\mathbf{x}) = p^\star(\mathbf{x}') T(\mathbf{x}|\mathbf{x}')$$

- A Markov chain that satisfies detailed balance is said to be **reversible**.

Markov chains (2/2)

- A distribution is **invariant** or **stationary** w.r.t. a Markov chain if each step leaves the distribution invariant. That is, a distribution $p^\star(\mathbf{x})$ is invariant for a homogeneous Markov Chain with transition $T(\mathbf{x}'|\mathbf{x})$ if,

$$p^\star(\mathbf{x}') = \sum_{\mathbf{x}} T(\mathbf{x}'|\mathbf{x})p^\star(\mathbf{x}).$$

- A sufficient (but not necessary) condition for $p^\star$ to be invariant is if we choose the transition probabilities T to satisfy the **detailed balance** property,

$$p^\star(\mathbf{x})T(\mathbf{x}'|\mathbf{x}) = p^\star(\mathbf{x}')T(\mathbf{x}|\mathbf{x}')$$

- A Markov chain that satisfies detailed balance is said to be **reversible**.
- The one other property we will want is **ergodicity**, which guarantees that the same unique invariant distribution will be reached regardless of initial starting point $\mathbf{x}_0$.

MCMC: Algorithm

- The MCMC proposal distribution makes **local** changes to a current value. We choose a $q(\mathbf{x}'|\mathbf{x})$ defines a distribution of candidate values $\mathbf{x}'$, given a current value $\mathbf{x}$
  - Default choice: add a small amount of Gaussian noise
- We use the proposal and the joint density to define an "acceptance ratio"

$$A(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')q(\mathbf{x}|\mathbf{x}')}{p(\mathbf{y}, \mathbf{x})q(\mathbf{x}'|\mathbf{x})}\right)$$

- With probability A we "move" to the new value $\mathbf{x}'$, otherwise we stay at $\mathbf{x}$

MCMC: Algorithm

- The MCMC proposal distribution makes **local** changes to a current value. We choose a $q(\mathbf{x}'|\mathbf{x})$ defines a distribution of candidate values $\mathbf{x}'$, given a current value $\mathbf{x}$
  - Default choice: add a small amount of Gaussian noise
- We use the proposal and the joint density to define an "acceptance ratio"

$$A(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')q(\mathbf{x}|\mathbf{x}')}{p(\mathbf{y}, \mathbf{x})q(\mathbf{x}'|\mathbf{x})}\right)$$

- With probability A we "move" to the new value $\mathbf{x}'$, otherwise we stay at $\mathbf{x}$
- Performing this update repeatedly defines a sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2 \ldots$ of **dependent** draws.

MCMC: Algorithm

- The MCMC proposal distribution makes **local** changes to a current value. We choose a $q(\mathbf{x}'|\mathbf{x})$ defines a distribution of candidate values $\mathbf{x}'$, given a current value $\mathbf{x}$
  - Default choice: add a small amount of Gaussian noise
- We use the proposal and the joint density to define an "acceptance ratio"

$$A(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')q(\mathbf{x}|\mathbf{x}')}{p(\mathbf{y}, \mathbf{x})q(\mathbf{x}'|\mathbf{x})}\right)$$

- With probability A we "move" to the new value $\mathbf{x}'$, otherwise we stay at $\mathbf{x}$
- Performing this update repeatedly defines a sequence $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2 \ldots$ of **dependent** draws.
- Note this doesn't require the normalizing constant, just $p(\mathbf{x}, \mathbf{y})$!

Symmetric proposal distributions

Note that in the "default choice" of Gaussian noise,

$$q(\mathbf{x}'|\mathbf{x}) = \mathcal{N}(\mathbf{x}'|\mathbf{x}, \sigma^2\mathbf{I})$$

then the proposal is "symmetric", in that $q(\mathbf{x}'|\mathbf{x}) = q(\mathbf{x}|\mathbf{x}')$. In this setting, we have a simplified expression

$$A(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')q(\mathbf{x}|\mathbf{x}')}{p(\mathbf{y}, \mathbf{x})q(\mathbf{x}'|\mathbf{x})}\right) = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')}{p(\mathbf{y}, \mathbf{x})}\right).$$

Symmetric proposal distributions

Note that in the “default choice” of Gaussian noise,

$$q(\mathbf{x}'|\mathbf{x}) = \mathcal{N}(\mathbf{x}'|\mathbf{x}, \sigma^2\mathbf{I})$$

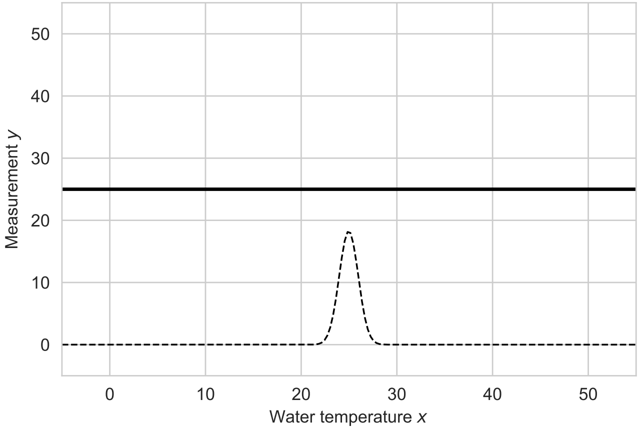
then the proposal is “symmetric”, in that $q(\mathbf{x}'|\mathbf{x}) = q(\mathbf{x}|\mathbf{x}')$. In this setting, we have a simplified expression

$$A(\mathbf{x} \to \mathbf{x}') = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')q(\mathbf{x}|\mathbf{x}')}{p(\mathbf{y}, \mathbf{x})q(\mathbf{x}'|\mathbf{x})}\right) = \min\left(1, \frac{p(\mathbf{y}, \mathbf{x}')}{p(\mathbf{y}, \mathbf{x})}\right).$$

Intuitively this looks like a noisy sort of hill climbing:

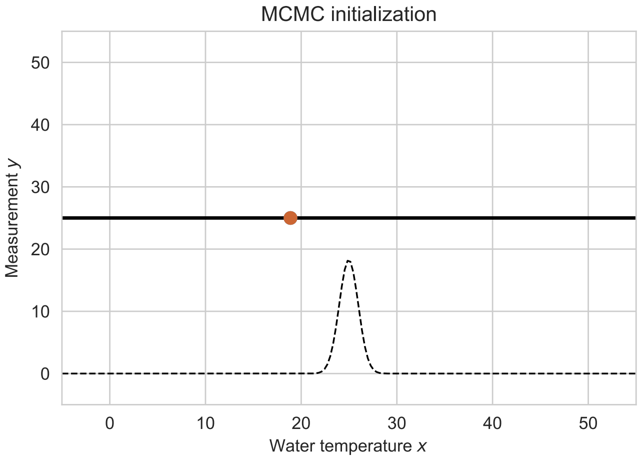
- sample a value $\mathbf{x}' \sim q(\mathbf{x}'|\mathbf{x})$
- if $p(\mathbf{y}, \mathbf{x}') > p(\mathbf{y}, \mathbf{x})$, then move to $\mathbf{x}'$
- otherwise, maybe move to $\mathbf{x}'$

MCMC schematic



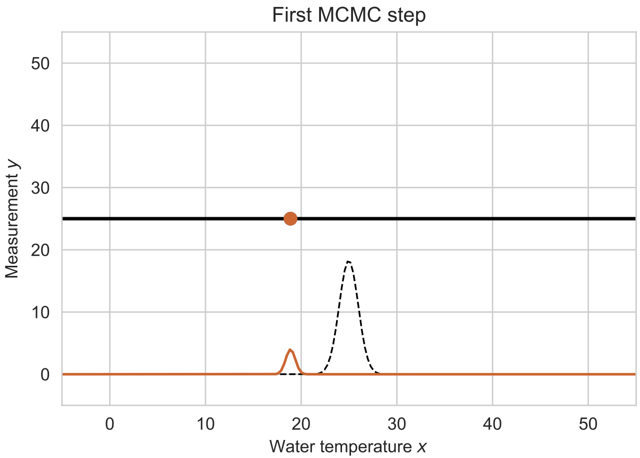
The (unnormalized) joint distribution $p(x, y)$ is shown as a dashed line

MCMC schematic



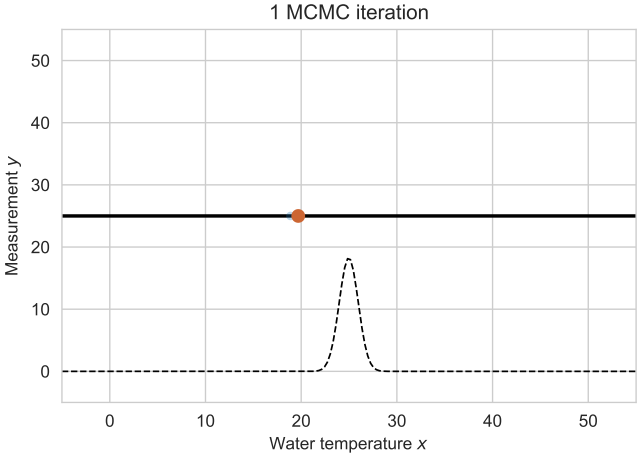
Initialize arbitrarily (e.g. with a sample from the prior)

MCMC schematic



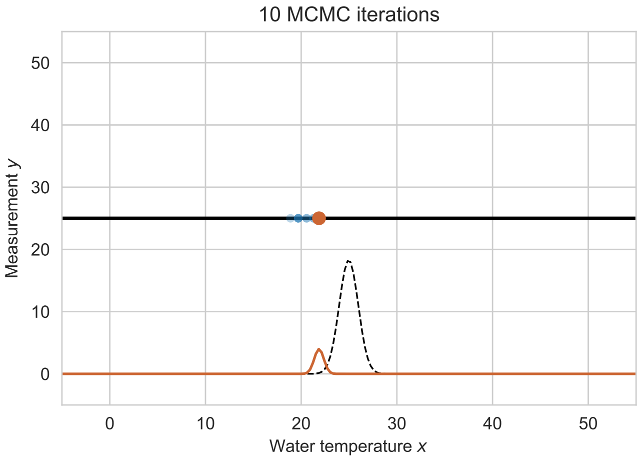
Propose a local move on x from a transition distribution

MCMC schematic



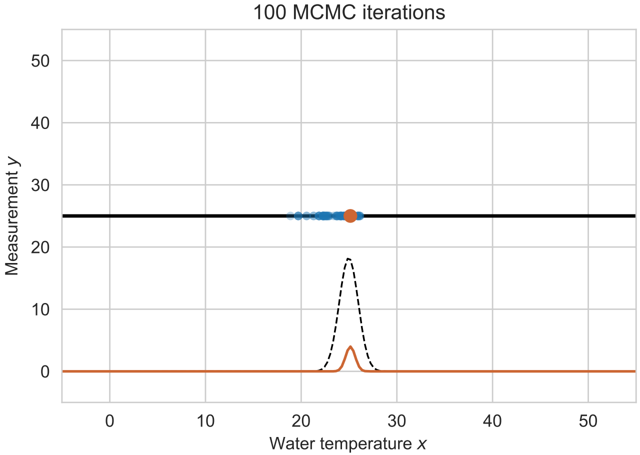
Here, we proposed in a region of higher probability density, and accepted

MCMC schematic



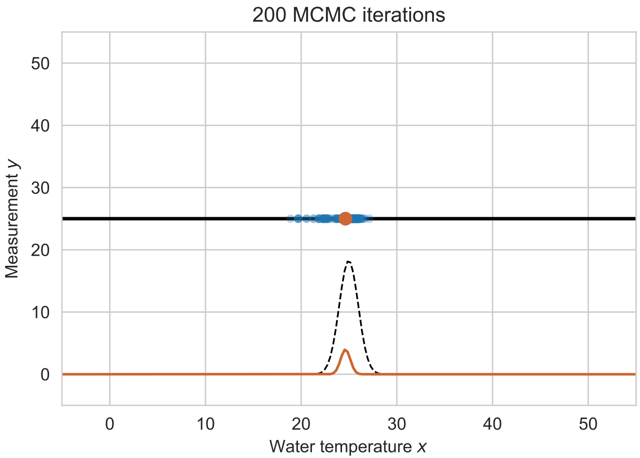
Continue: propose a local move, and accept or reject.
At first this will look like a stochastic search algorithm!

MCMC schematic



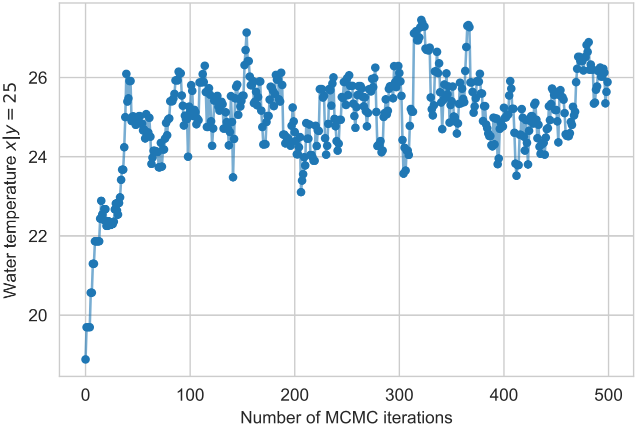
Once in a high-density region, it will explore the space

MCMC schematic



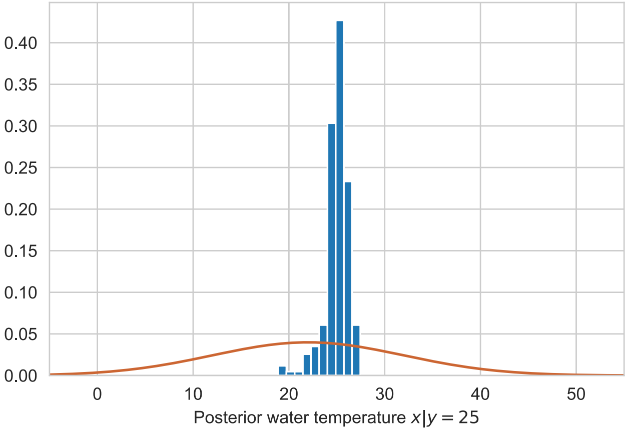
Once in a high-density region, it will explore the space

Helpful diagnostic



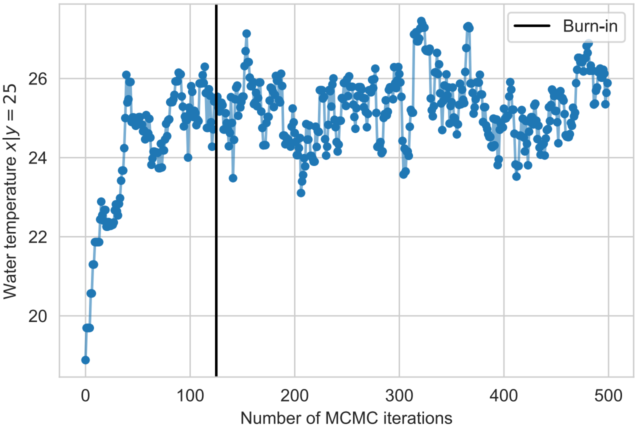
Helpful diagnostic: a **trace plot** shows the coordinate x on the y-axis, against iterations on the x-axis, showing the progression of the Markov chain.

MCMC schematic



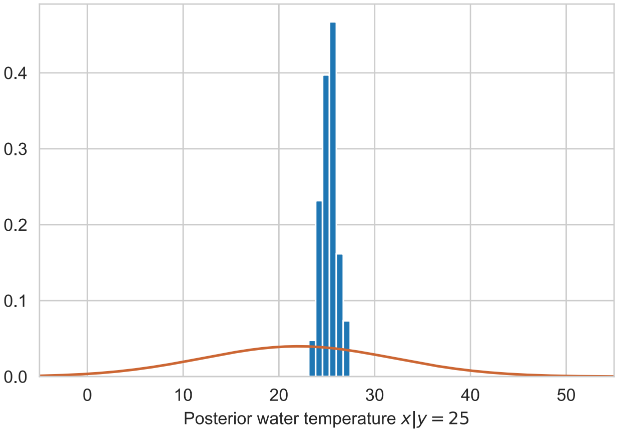
Histogram of trace plot, overlaid on prior probability density.
Notice the "tail" off to the left...

Solution: discard “burnin”



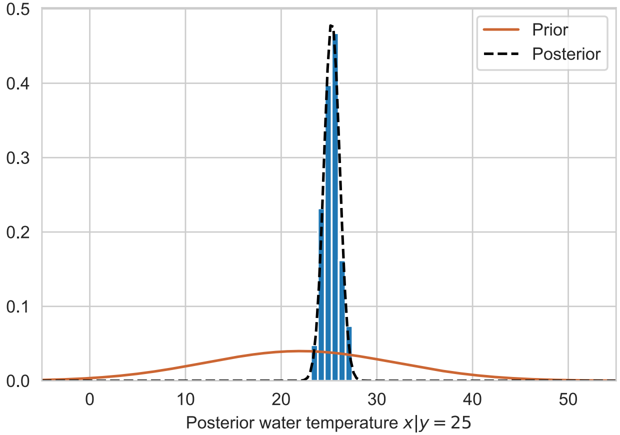
It can take a while before the MCMC chain can reach its equilibrium distribution. It’s customary to discard early samples until the chain has “burned in”...

Solution: discard “burnin”



It can take a while before the MCMC chain can reach its equilibrium distribution. It’s customary to discard early samples until the chain has “burned in”...

Estimating a parametric approximation



Gaussian approximation: $\hat{\mu} = \frac{1}{S - s_0} \sum_{s=s_0}^{S} x^{(s)}$; $\hat{\sigma}^2 = \frac{1}{S - s_0} \sum_{s=s_0}^{S} (x^{(s)} - \hat{\mu})^2$

Pitfalls

- How do we choose the proposal?
- Bad proposals can lead to **low acceptance rates**, or **very small steps** — both are problematic
- Diagnosing convergence can be tricky; when has "burn-in" ended? What happens if we have disconnected modes?
- In large data settings, evaluating the acceptance ratio can be expensive

We'll talk about some of these things later on, when we re-visit MCMC.