Monte Carlo Markov Chain Sampling

Introduction

- Family of algorithms for sampling - most commonly used in Bayesian inference, also the building block of more complex sampling algorithms such as Hamiltonian Monte Carlo (HMC) and No U-Turn Samplers (NUTS)
- Monte Carlo methods refer to generating random samples to approximate properties of a distribution – for example, if we have a function *f(x)* and *x* is distributed under *p(x)* then the expectation of the function is approximated as $E[f(x)] \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$ where x_i is sampled from *p(x)*, but this is often difficult to do directly
- This is where MCMC comes in – we wish to find a Markov Chain whose stationary distribution matches the target distribution (e.g. out posterior)
- Stationary distribution – one that satisfies, for continuous variables, $\pi(x') = \int \pi(x) P(x'|x) dx$, where $P(x'|x)$ is the transition probability
- We choose a time reversible MC which satisfies the detailed balance eqn: $\pi(x)P(x'|x) = \pi(x')P(x|x')$ – this way we can ensure that a stationary distribution exists, and can choose our target as it
- Then we assert that our MC is aperiodic, irreducible (all states can be accessed from each other in a finite number of steps with non-zero prob) and positive recurrent (every state will be revisited with probability 1 and finite average number of steps) – all these conditions together imply the chain is ergodic and will converge to its stationary distribution

Metropolitan-Hastings

- Standard algorithm to create the MC with the properties above – we then run it for several steps until it converges, and then we can sample from the posterior
- We start with a proposal distribution $q(x'|x)$, and propose a new state $x' \sim q(x'|x)$
- The acceptance probability is given by $\alpha = \min(1, \frac{p(x')q(x|x')}{p(x)q(x'|x)})$
- Accept the proposal with probability α – if rejected, stay at *x*
- If $q(x'|x)$ is symmetric, like a Gaussian, the ratio simplifies to $\alpha = \min(1, \frac{p(x')}{p(x)})$ and $p(x)$ doesn't need to be normalised – sometimes can be quite useful
- There is a 'burn-in' period, where we ignore the first section of the iterations as it converges to the distribution
- There is high autocorrelation – next sample depends on the current one – so number of independent samples can be reduced – can thin the chain by taking every k^{th} sample, but controversial because it discards information. Could also run longer chain, but then this can become inefficient

Alternatives

- There are algorithms other than Metropolis-Hastings – Gibbs sampling, which uses conditional probabilities and distributions to samples, and adaptive MCMC where the proposal distribution is adjusted as the process goes on
- There are also alternatives to MCMC in general – some are extensions, such as HMC where we look at the gradients of the log probability to guide the sampling process, or Sequential MC, where we have a population of particles used to approximate the target distribution
- Some are quite different, e.g. Variational Inference which approximates the target with a simpler parametric distribution that is easy to sample from, and importance sampling where we weight samples from a proposal relative to their accuracy