Bayesian Reasoning:

Markov Chain Monte Carlo

A/Prof Geraint F. Lewis

Rm 560: gfl@physics.usyd.edu.au

Sampling the Posterior

So far we have investigated how to use Bayesian techniques to determine posterior probability distribution for a set of parameters in light of some data.

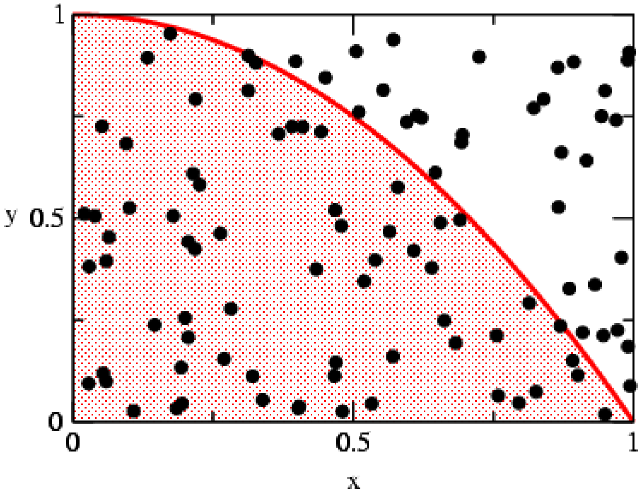
However, our parameter set may be highly-dimensional, and we may only be interested in a sub-set of (marginalized) parameters.

Markov Chain Monte Carlo (MCMC) is an efficient approach to this problem.

Why Monte Carlo?

Monte Carlo is a district of Monaco which is famous for casinos and gambling. Monte Carlo techniques rely on a sequence of random numbers (like the roll of dice) to make calculations.

By throwing random numbers over the x-y plane it's easy to see that an approximation of the integral is the fraction of points "under the curve" times the area of interest.



Integration of Functions

Instead of sampling the area we can sample a function directly. For a posterior distribution *p(X|D)* we can calculate an expectation value through

$$\langle f(X) \rangle = \int f(X)\; p(X|D,I)\; dX = \int g(X)\; dX$$

This can be calculated through a Monte Carlo sample by multiplying the volume by the mean value of the sampled function through

$$\langle f(X) \rangle = \int_V g(X)\; dX \approx V \times \langle g(X) \rangle \pm \sqrt{\frac{\langle g^2(X) \rangle - \langle g(X) \rangle^2}{n}}$$

Markov Chains

Straight-forward Monte Carlo integration suffers from some problems, especially if your posterior probability is peaked in a small volume of your parameter space.

What we would like is a method to throw down more points into the volume in regions of interest, and not waste points where the integrand is negligible.

We can use a Markov Chain to “walk” through the parameter space, loitering in regions of high significance, and avoiding everywhere else.

Metropolis-Hastings

Metropolis-Hastings (1953) presented a (very simple) approach to extract a sample from the *target distribution, $p(X|D,I)$,* via a weighted random walk.

To do this, we need a *transition probability, $p(X_{t+1}|X_t)$,* which takes us from one point in the parameter space to the next.

Suppose we are at a point X_t and we choose another point Y (drawn from the *proposal distribution, $p(Y|X_t)$*), how do we choose to accept the step $X_t \rightarrow Y$?

Metropolis-Hastings

We can then calculate the *Metropolis ratio* where

$$r = \frac{p(Y|D,I)}{p(X_t|D,I)}\frac{q(X_t|Y)}{q(Y|X_t)}$$

If the proposal distribution is symmetric, the right-most factor is unity.

If *r>1* we take the step. If not, we accept it with a probability of *r* (i.e. compare to a uniformly drawn random deviate).

Metropolis-Hastings

Hence, the probability that we take the step can be summarized in the *acceptance probability* given by

$$\alpha(X_t, Y) = min(1, r) = min\left(1, \frac{p(Y|D,I)}{p(X_t|D,I)}\frac{q(X_t|Y)}{q(Y|X_t)}\right)$$

and the Metropolis-Hastings algorithm is simply;

```
1) Set $t=0$ and initialize $X_0$
2) Repeat { Chose $Y$ from $q(Y|X_t)$,
            Obtain a random deviate $U=U(0,1)$,
            If $U \leq r$ then $X_{t+1}=Y$, else $X_{t+1}=X_t$; increment t }
```

Example 1

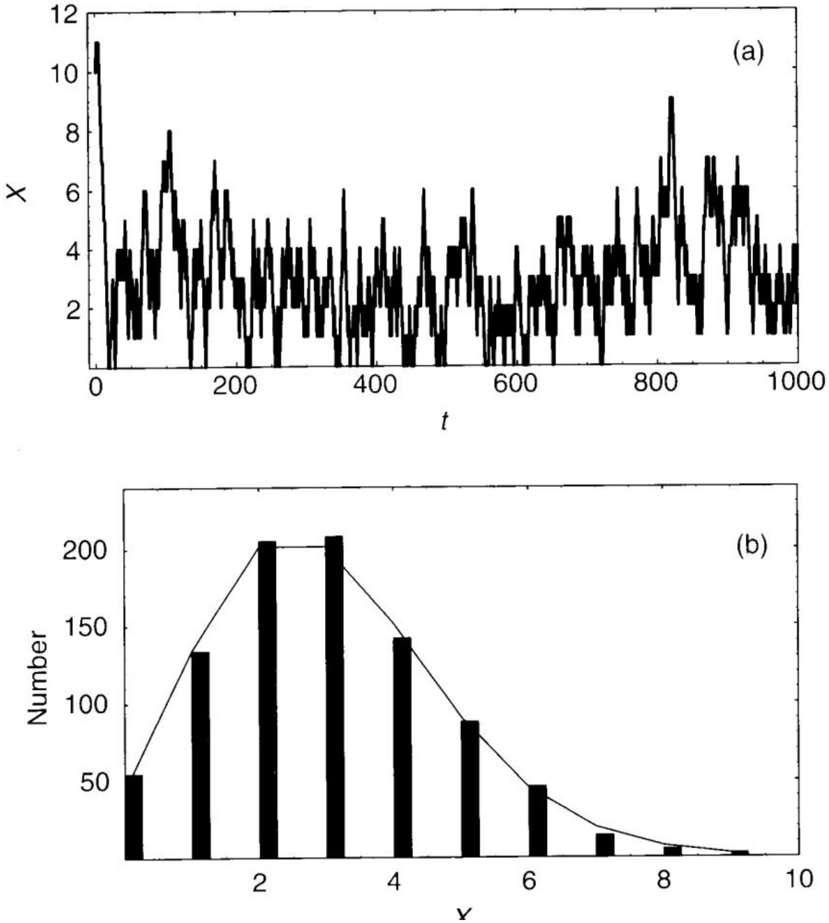
Suppose that the posterior distribution is given by

$$p(X|D,I) = \lambda^X \ e^{-\lambda}/X!$$

This is the Poisson distribution and *X* can only take non-negative integer values. Our MCMC algorithm is;

```
1) Set $t=0$ and initialize $X_0$
2) Repeat {   Given $X_t$, choose $U_1=U(0,1)$;
              If $U_1>0.5$, propose $Y=X_t+1$ else $Y=X_t-1$;
              Calculate $r = p(Y|D,I)/p(X_t|D,1) = \lambda^{Y-Xt} X_t! / Y!$ ;
              If $U_2=U(0,1)<r$, then $X_{t+1}=Y$, else $X_{t+1}=X_t$ }
```

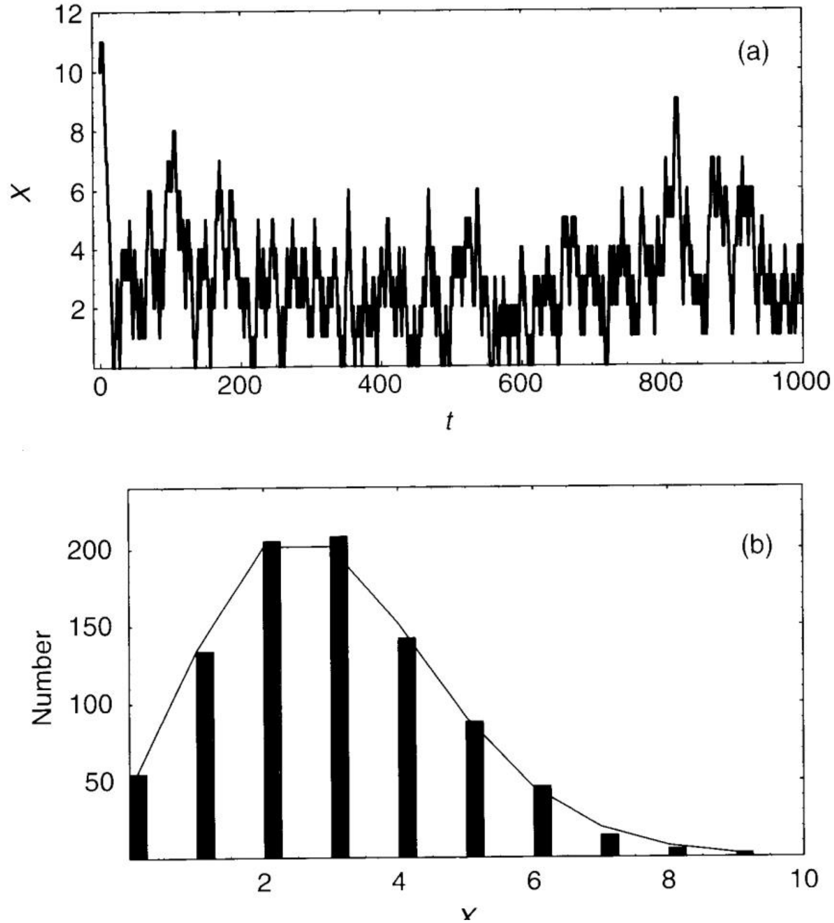
Example 1



The result is a chain of integer values. How are we to interpret the chain.

Neglecting the *burn-in* the distribution of integer values is the same as the posterior distribution we are sampling, as seen in the histogram of their values (solid line is the Poisson distribution).

Example 1



So the values in the chain are a representation of the posterior probability distribution.

Hence, the mean value of out parameter is simply given by

$$\langle X \rangle = \frac{\sum i \; n_i}{\sum n_i}$$

Example 2

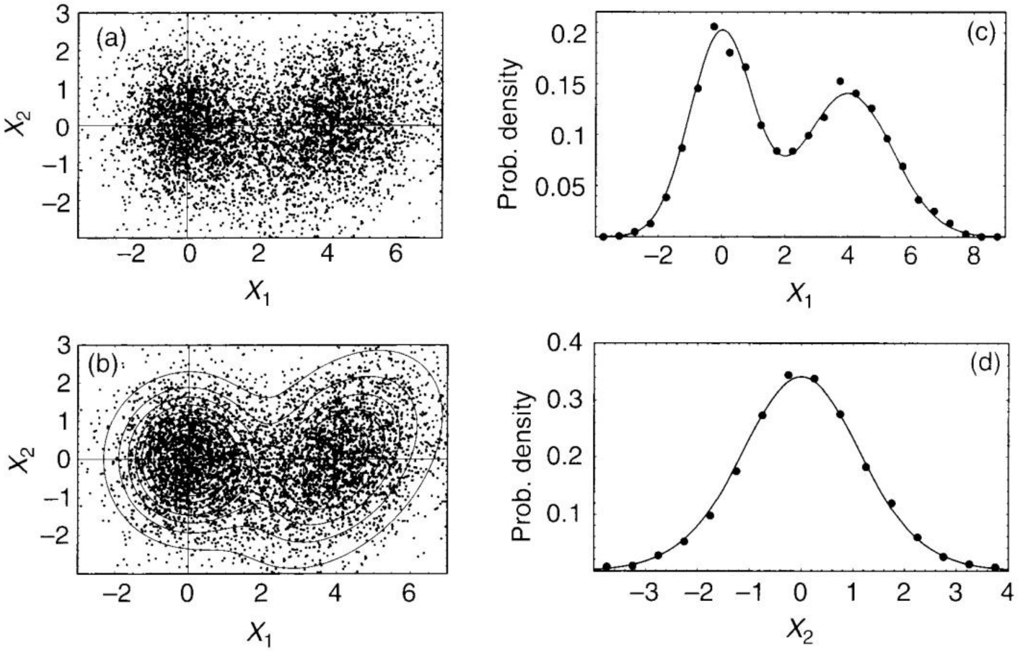
Clearly, the nature of the previous problem required us to take integer steps through the posterior distribution, but in general, our parameters will be continuous; what is our proposal distribution, *p(Y*|*X$_t$)*, in this case?

Typically, this is taken to be a multi-variate Gaussian distribution, centered upon *X$_t$,* with a width σ. If we have *n* parameters (and hence dimensions), the proposal is;

$$X_t \rightarrow Y = X_t + \sigma \times N(0,1)$$

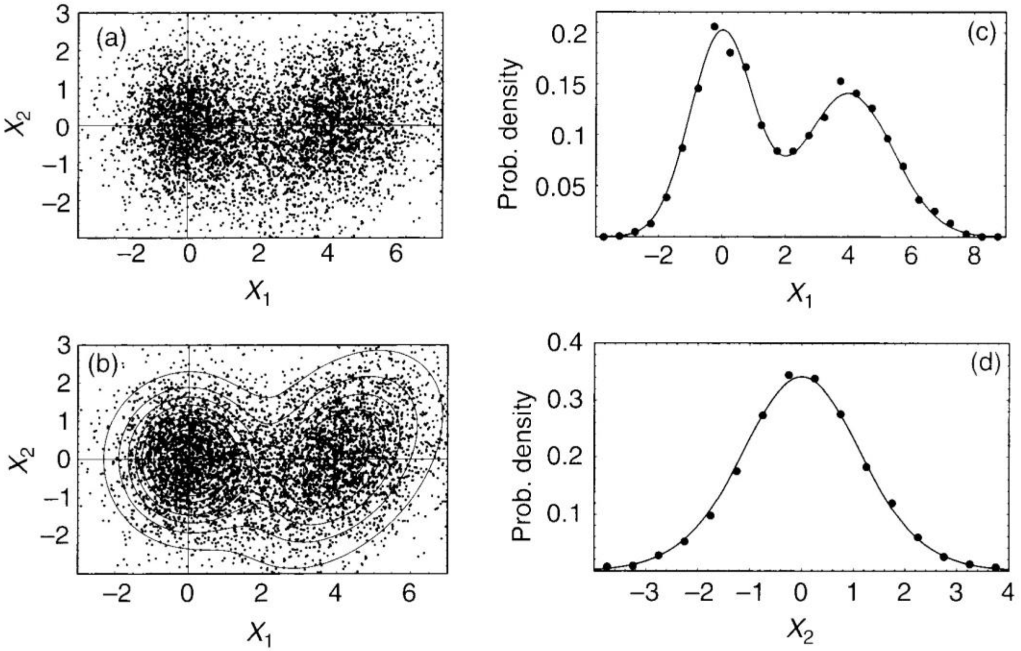
(Note, each dimension may have a different σ_i).

Example 2



Here is a case where we have a 2-dimensional, bimodal posterior distribution. Clearly the points are clustered in the parameter space in two “lumps”.

Example 2



The contours show that the density of points mirrors the underlying posterior distribution. Furthermore, the projected (integrated) density of points corresponds to the marginalized distribution.

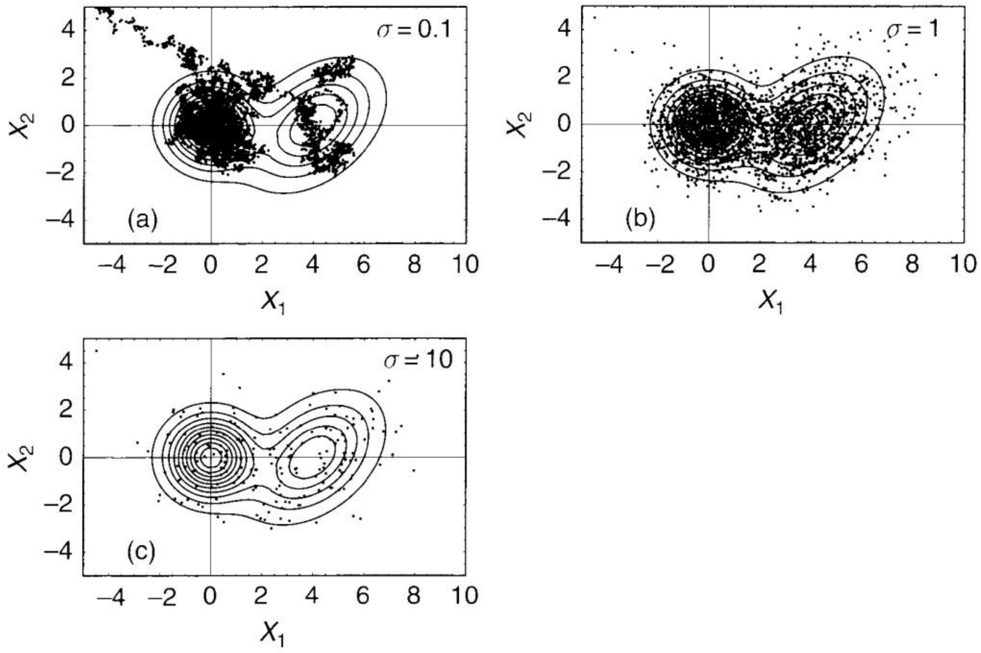
Proposal Choice

If we are going to step through the posterior probability distribution, the step-size of proposal distribution is going to be related to how efficiently we can explore the volume.

Clearly, if your step-size is too small, it will take many steps (a lot of time) to cover the volume.

If the step size is too large, we can skip over the volume without spending time in regions of interest.

Proposal Choice



Changing σ by a factor of 10 illustrate this. If we want to wait long enough, any of these proposal distributions will cover, but how do we know what to choose?

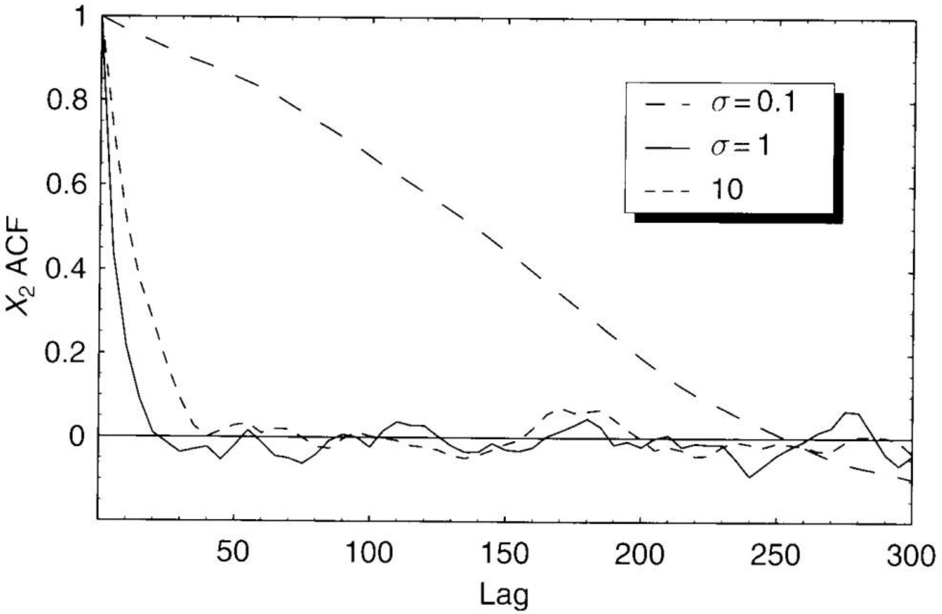
Autocorrelation Function

So, how do you choose the *ideal* proposal distribution? Unfortunately, this is an *art* which comes with experience, although the *auto-correlation* function gives us a good clue through;

$$\rho(h) = \frac{\sum[(X_t - \bar{X})(X_{t+h} - \bar{X}])}{\sqrt{\sum(X_t - \bar{X})^2 \times \sum(X_{t+h} - \bar{X})^2}}$$

The MCMC samples are not independent and so the auto-correlation reveals the convergence (i.e. we want solutions which the ACF goes to zero as quickly as possible).

Autocorrelation Function



The ACF is represented as an exponential function of the form;

$$\rho(h) \sim exp(-\frac{h}{\tau_{exp}})$$

The smaller the *h* the faster the convergence. But. Of course, we need to try several values to find the sweet spot. However, we have a rule of thumb to guide this.

Autocorrelation Function

Work by Roberts, Gelman & Gilks (1997) recommend calibrating the acceptance rate to 25% for high dimensional problems, and 50% for low (1-2) dimensional problems.

But there are questions remaining;

1) How long is the burn-in?

2) When do we stop the chain?

3) What’s the proposal?

Answer: Experience :)