**Metropolis-Hasting Algorithm**

* The requirements on the target f are quite minimal, which allows for settings where very little is known about f.
* The Markov perspective leads to efficient decompositions of high-dimensional problems in a sequence of smaller problems that are much easier to solve.
* Monte Carlo is to Bayesian inference as optimization as to maximum likely-hood
* We use M-H algorithm because:

1. We don’t know our target distribution(posterior) or we don’t have the pdf of our target distribution.
2. Using our proposal, we might generate lot of samples or RV but those RV might not represent our target distribution. Infact, if we accept all the values generated by proposed distribution using MCMC then the resultant posterior distribution keeps changing based on number of samples
3. We use M-H algo to find our posterior(target distribution) so that it can accept only some values generated by random walk.

**“***The obvious difficulty with accepting and rejecting proposed values of θ based on the posterior distribution is that we usually don’t know the functional form of the posterior distribution. If we knew the functional form, we could calculate probabilities directly without generating a random sample. So how can we accept or reject proposed values of θ without knowing the functional form? One answer is the M–H algorithm!*”

1. The M–H algorithm can be used to decide which proposed values of θ to accept or reject even when we don’t know the functional form of the posterior distribution.

* The Metropolis–Hastings algorithm works by generating a sequence of sample values in such a way that, as more and more sample values are produced, the distribution of values more closely approximates the desired distribution {\displaystyle P(x)}----**Achieving stationary distribution of some kind.**
* The fundamental problem is that we are often unable to evaluate this integral in the denominator of Bayesian probability formula analytically and so we must turn to a numerical approximation method instead. An additional problem is that our models might require a large number of parameters. This means that our prior distributions could potentially have a large number of dimensions. This in turn means that our posterior distributions will also be high dimensional. Hence, we are in a situation where we have to numerically evaluate an integral in a potentially very large dimensional space.

**Questions?**

* **What is the meaning of posterior\_proposed\_value/posterior\_current\_value?**

1. By dividing the posterior at one position by the posterior at another, we're sampling regions of higher posterior probability more often than not, in a manner which fully reflects the probability of the data.
2. Basically in MH, we try to remain in area with high probability, i.e at the peak of the hill. By remaining in the dense probable area we can represent more accurate data. The posterior value of the desired distribution tells about the height in target distribution. Basically, if the proposed location is higher than the previous distribution then moves to the new distribution (the probability will be greater than 1). If the proposed location is smaller than the previous, then the probability will be less than 1. But still this can be accepted if the uniform distribution value is lesser than this value. Why we compare with uniform...i don’t know

* **Why and when we use metropolis hasting algorithm?**