Assessing Convergence

The previous exercises raised an interesting question, which is how long should the Markov, the MCMC chains run for? Or in other words, how long should we run them, a model until it is ready to, to use? mainly to draw valuable inferences about the parameters of interest.

This takes us to the issue of convergence and how do we assess convergence when estimating Bayesian models or when estimating models in a Bayesian fashion using modern Bayesian computational approaches. One first way of assessing convergence is through the visual inspection of the traceplots, by traceplots I mean, the successive parameter draws. So, I mentioned when we talked about the Gibbs Sampler and we talked about the Metropolis Hastings algorithm, the idea was to draw samples of the parameters until we reach the steady state, stationary state of the posterior distribution of the parameters.

So, how do we know that we have actually reached the steady state, the steady state distribution? So, one way well, informal way of checking with convergence or then the parameters have are reached the parameter being drawn from the stationary part of the chain is through the visual inspection of this the succession of parameter draws.

So, here, you have a series of plots, each of which shows, draws, parameter draws. Some of these showed that the parameters are being drawn from a stationary distribution namely the parameter that the values of the parameters in successive in subsequent draws are essentially similar. Take a look for instance at the upper left panel.

However, if you look say, at the upper right panel or any other panel, except for the upper left one, we see that the each iteration yields values of the parameter that differed considerably from the previous one. So, what makes a convergent chain should look like this . There are obviously some variations in the values of the parameters we draw at each iteration of the algorithm, but not too much, right. So essentially, the values looked or the change looked almost vertical and dense, this is a plot of chain that has already converged.

In contrast, this plot shows that these chain has not compressed, so there's still a lot of variation and each iteration of the of the algorithm beat the Gibbs sampler beat, Metropolis Hastings yields very different values of the parameters right. Now, this is us taking a look at the trace plots. So, looking at the values of the parameters is one informal way of assessing conversion. There are however, more formal approaches to assessing conversion, which depends essentially, on which approach we use to estimating and essentially in a nutshell there are two main approaches to fitting a more than a Bayesian fashion. One is running one long chain, one Gibbs sampler one Metropolis Hastings algorithm, or combination of both for a very long time. Another approach is running a series of shorter chains. So we can run three say chains, and then assess the conversions of the three chains. And the criteria we use to assess conversions depends on which approach we use, right.

Typically, the second approach of running a series of different chains, each of them starting from different initial values, is the pro valid approach is the one I use in my own research.

But no matter which approach we use the first part of any chain, we're writing a single chain MCMC, if we're running multiple chains, the first part of the chain is used when we call a burn in period, essentially a period that we use as our preparations or as a warm up, then we are not going to use to draw inferences for the parameters when just we allow the chain enough time or enough durations so as to reach the stationary state. This figure here which essentially is a plot taken from the exercise we already conducted on using deep sampling to draw inferences about the mean and the variance of a normal distribution, this is a plot of the values, the samples of mean drawn using Gibbs sampling, so, that first before, at some point, the value starts being stable around a certain value, two, in this case, but before that, in this case, this steady state is reached quite fast.

Before that there are a series of values that are quite off mark. Now, this first part is typically used as a burn in. Now, the question is obviously, how do we know when we are feeding them all? how long the burn in period is? Well, here's the approach. The purchase varies some people some approach, some researchers use the first half of a chain, so the bottom running Gibbs sampling algorithm for 1000 iterations, well, we can use the first half of those 1000 iterations. So the first 500 iterations as burn in, some other researchers use shorter burn in periods ,a quarter of the change, the important is to not to use the first 25-50% of the chain to draw inferences with rather using a sub burn in period.

Now going back to the issue of convergence criteria, when we run a single chain for a long period, there are two main conversions criteria Geweke's us criteria and Heidel's criteria criterion. I'm not going to go into too much detail about it, but the idea is to compare the first part of that single chain after the burn in period to the last part of the chain. And essentially the idea is well, if the values sample in the first few iterations of the algorithm, post burn in, and the value sample towards the end of the chain do not differ very much, while we can be reasonably sure the key here is reasonably sure that the parameters are being drawn from the stationary part of the chain.

So, suppose we are running chain for 1000 iterations and we use the first 200 iterations as burn in. Well, if the first 100 iterations post burn in, so, the body is sampled from the 201st to 300 iteration are reasonably similar to the value sample at the durations 900 to 1000. Well, we can be reasonably confident that the chain has or the algorithm has converged.

A little bit more formally, for instance, making squinted Geweke's criteria what it does is take the first 10% of the chains or of the values sample post burn in against and compare it against the last 50% of the sample value and performance sort of T type of test. And if the test statistics are outside the -1.96 to 1.96 range, well, that indicates a lack of convergence. Otherwise, if the statistics are within that, those values we can be reasonably confident that the chain has converged.

All these diagnostics you make a Heidel are available readily available in our through the [inaudible] package and you have some exercises here and R script in which I perform Heidel use the Heidel criterion to assess convergence. When instead of running one long chain chain, we run a series of shorter chains. The most widely used criterion to assess conversion is Gelman and Rubin R hat. Again, the slides go a little bit more in detail about what exactly these are, but essentially, what we what it does is compares the playability of the sample bodies within each chain and between chains in order to compute a certain measure R hat and if the value of this R hat is below 1.2 1.1 we can be reasonably certain that the different chains have converged. Again, Gelman is R hat diagnostic is readily available in our and you have a script uses these R hat criterion to assess conversions in the case of Gibbs sampler with multi chain.

Let me also say that this whole notion of conversions is a lot of the way down when can we absolutely sure that chain has convert or a series of chains has converted? The short answer is probably never, but in practice, people, researchers applying Bayesian methods take this criterion where they R hat is below 1.2 or when the Heidel criteria is between the test statistic is between -1.96 to 1.6. We can reasonably assume that we have reached convergence