STAT40380/STAT40390/STAT40850 Bayesian Analysis

Dr Niamh Russell

School of Mathematics and Statistics
University College Dublin

niamh.russell@ucd.ie

April 2016

Advanced topics in Bayesian Analysis

Today we will cover some advanced topics in Bayesian statistics that may be of interest to those who wish to take their study further

- Advances in Bayesian modelling:
    - Mixture models
    - Outliers and robust models
    - Decision analysis
- Advances in computation:
    - Parallelised MCMC
    - Reversible Jump MCMC
    - Approximate Bayesian Computation (ABC)

Mixture models

- Often a simple likelihood function, such as the normal, will not be appropriate for our data
- This may be because the data are especially complicated, or because there are known groups in our data, or because there are unknown sub-populations
- The usual practice is to write the likelihood as a *finite mixture* of proability distributions:

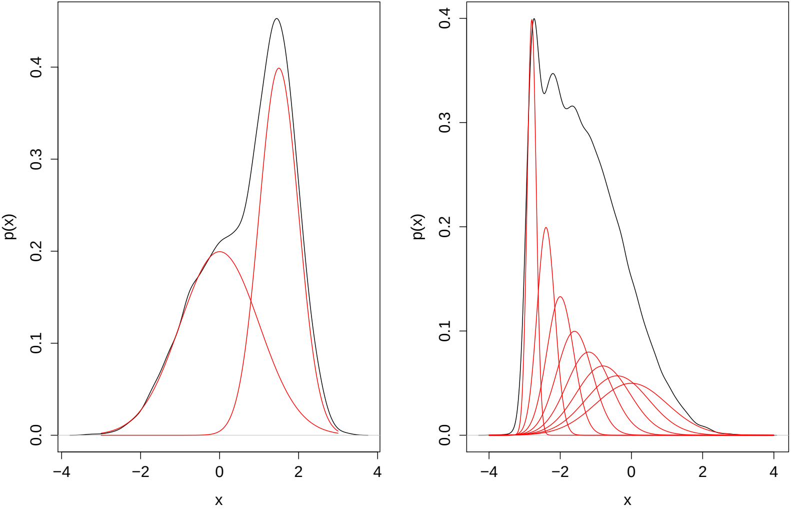
$$p(y_i|\boldsymbol{\theta}, \boldsymbol{\lambda}) = \lambda_1 p_1(y_i|\boldsymbol{\theta}_1) + \lambda_2 p_2(y_i|\boldsymbol{\theta}_2) + \cdots + \lambda_M p_M(y_i|\boldsymbol{\theta}_M)$$

- Here we have M different components in our finite mixture with proportions λ_m where $\sum_{m=1}^{M} \lambda_m = 1$
- Each component probability distribution $p_m()$ could be an individual normal distribution with unknown mean and variance

Mixture models 2

- We need to place prior distributions on the means and variances
- We also need a prior distribution for the mixture weights λ_m. A common choice is the Dirichlet distribution
- If the number of components M is unknown we might try to use a prior distribution here, but this causes extra problems!
- Another problem we may face is that the parameters may not be *identifiable*. If we switch the mean and variance of one component probability distribution with another we will have exactly the same model. This problem can be reduced by forcing the means to lie in a certain order, or by placing stronger prior distributions on the parameters

Mixture models: picture



Outliers and robust models

- Lots of standard frequentist theory looks at how to deal with *outliers* or other extreme observations
- If some observations are strange we may question the experimenters
- However, if the data are correct there is *no reason* to remove them from our modelling
- Instead we might like to try and use a heavy-tailed distribution as our likelihood (instead of the usual Gaussian, Poisson, binomial etc)
- A long-tailed distribution, such as the *t*-distribution, could be used in place of the Gaussian (this could be part of our model-building and selection stage)
- The longer-tailed distributions can often be seen as an infinite mixture of our simpler distribution

Outliers and robust models 2

- A robust model can be used if the data are *over-dispersed*, but never when they are *under-dispersed*
- There are several over-dispersed distributions which we may consider:
  - The t_d distribution rather than the normal. If d is small the distribution is long-tailed. This is a mixture of normal distributions:

$$x_i | v_i \sim N(\mu, v_i), v_i \sim \sigma^2 \chi_d^{-2}$$

  - The negative binomial rather than the Poisson:

$$x_i \sim Po(\lambda_i), \lambda_i \sim Gamma(\alpha, \beta)$$

  - The beta-binomial rather than the Binomial:

$$x_i \sim Bin(n, \pi_i), \pi_i \sim Be(\alpha, \beta)$$

- In each case we can either fit as a hierarchical model if we are interested in the top level parameters, ie v_i, λ_i, or π_i or simply using the new longer tailed distribution

Decision analysis

- What do we do when we have fitted the model and obtained estimates of our parameters?
- Sometimes (but not always) we want to make *decisions* based on our posterior distributions, eg if a parameter representing the effect of a drug is less than a certain value, we might be exposed to a cost; if it is above a certain value we might make some money. Should we produce the drug or not?
- The parameter values are uncertain, so we should *use the posterior distribution to make the best guess*
- The usual steps in a decision analysis are:
  1. Write down the list of all possible decisions **d** and outcomes **y** (in years, or euro...)
  2. Determine the probability distribution of **y** for each decision option
  3. Define a utility function $U(\mathbf{y})$ mapping outcomes onto real numbers (eg years of life or profit)
  4. Compute $\mathbb{E}(U(\mathbf{y})|\mathbf{d})$, and choose the decison with the highest expected utility

Decision analysis example

- A 95-year old man with a tumour in the lung must decide between radiotherapy, surgery or no treatment (d_1, d_2, d_3)
- It is know that:
    - There is a 90% chance that the tumour is malignant
    - If the man does not have lung cancer, his life expectancy is 34.8 months
    - If the man does have lung cancer:
        - with radiotherapy his life expectancy is 16.7 months
        - with surgery, there is a 35% chance he will die immediately but if he survives his life expentancy is 20.3 months
        - with no treatment, his life expectancy is 5.6 months
- If the patient goes through a treatment, his life expentancy is reduced by one month to compensate for discomfort
- We could now work out the life expectancy for each treatment

Speeding up Bayesian computation

- Fitting a large Bayesian model with many parameters can be very slow
- Much research in recent years has focussed on how to speed up the fitting of Bayesian models, and especially making use of computers with multiple processors or fast graphical processing units (GPUs)
- There is aned to balance the extra effort in designing algorithms with the speed at which computer hardware and software improves
- Gibbs sampling and MCMC in particular are *hard to parallelise* because each iteration requirs the new parameter values to be generated from the previous values
- We can, however, return to *grid-based methods* to produce paralllelised estimates of the posterior density

Reversible Jump MCMC

- *Reversible Jump MCMC* (RJ-MCMC) is used when our model contains a variable number of parameters
- In such situations the size of the model (eg the number of mixture components) is another parameter to which we give a prior distribution
- The clever step is thst we can *propose a move between models of different dimension* by accepting a new model based on the Metropolis-Hastings acceptance ratio corrected by the first derivatives of the parameter transformations
- All the within-model updates are handled by standard Gibbs sampling Green (1995) showed that this satidfied all the requirements to converge to the correct posterior distribution

Approximate Bayesian Computation (ABC)

- What if we could not specify a probability distribution for the likelihood?
- Such a situation often occurs when the data are generated via a 'black box', for example when looking at the results of a weather forecasting model or that of a nuclear reactor
- Often these models will have inputs (parameters) and outputs(simulations) which we can compare with real data. Our task is to determine which values of the parameters best match the outputs
- *Approximate Bayesian Computation* works by comparing *summary statistics* based on the simulations with summary statistics computed on the real data

Approximate Bayesian Computation 2

- *Approximate Bayesian Computation* works when we can only simulate data from our model, but cannot necessarily write down a complete likelihood. We may still have prior distributions on parameters θ
- Let **x** be the data, and $\boldsymbol{\theta}$ be parameters. Define the summary statistic to be $S(\mathbf{x})$. This may be the mean of all the **x** values, or something more complicated
- An approximate algorithm given starting values for $\boldsymbol{\theta}$ is:
  1. Propose some new parameter values $\boldsymbol{\theta}^*$
  2. Run simulations for get simulated data $\mathbf{X}^*$
  3. Calculate $\delta = g(S(\mathbf{x}), S(\mathbf{x}^*))$ where $g()$ is a measure of distance
  4. If $\delta < \epsilon$ where ϵ is a pre-defined value, accept $\boldsymbol{\theta}^*$
- If the summary statistics are chosen to match the sufficient statistics of the 'true' likelihood, we may have a very efficient MCMC-like algorithm