Further simulations issues

- Bayesian inference is based on properties of posterior.
- We have seen that MCMC (the MH algorithm/Gibbs sampler) is a general tool which will (at least in theory) explore the posterior.
- In the remainder of this lecture we will briefly touch upon cases where the methods considered so far will work very ineffciently — or not at all.

Intractable normalising constants

Setup: Consider situation where the data model

$$\pi(x|\theta) = \frac{1}{c(\theta)} f(x|\theta),$$

where $c(\theta) = \int f(x|\theta) dx$ is the normalising constant which depends on the parameter θ.

Assume that is either impossible or infeasible to calculate $c(\theta)$.

Example: The **Ising model** is one such model.

Here $c(\theta)$ is obtained by a sum over all possible pixel images. For even moderately large pixel images, this becomes infeasible.

Posterior inference and intractable normalising constants

- Assume $\pi(x|\theta) = \frac{1}{c(\theta)} f(x|\theta)$ where $c(\theta)$ is intractable.
- We want to perform Bayesian inference, i.e. we want to sample the posterior $\pi(\theta|x) \propto \pi(x|\theta)\pi(\theta)$ using a MH algorithm.
- Sampling the posterior using the MH algorithm we have:

$$H(\theta, \theta') = \frac{\pi(x|\theta')\pi(\theta')q(\theta', \theta)}{\pi(x|\theta)\pi(\theta)q(\theta, \theta')} = \frac{\frac{1}{c(\theta')} f(x|\theta')\pi(\theta')q(\theta', \theta)}{\frac{1}{c(\theta)} f(x|\theta)\pi(\theta)q(\theta, \theta')}$$

Notice that a ratio $c(\theta)/c(\theta')$ appears.

Intractable normalising constants: One solution

- Assume that $\pi(x|\theta) > 0$ implies $\pi(x|\theta') > 0$ for all pairs θ, θ'.
- Then we have the following importance sampling identity:

  $$\frac{c(\theta)}{c(\theta')} = \mathbb{E}_{\theta'}\left[\frac{f(X|\theta)}{f(X|\theta')}\right],$$

  where $\mathbb{E}_{\theta'}$ is expectation wrt. $\pi(x|\theta')$.
- Hence the ratio $\frac{c(\theta)}{c(\theta')}$ can be estimated using a MH algorithm (say) with $\pi(x|\theta')$ as the invariant distribution.
- That is, for each update of the main MH algorithm we need to run an addition MH algorithm to estimating the Hastings ratio in the main algorithm.
- There are number of alternative solutions to one sketched above.

Likelikelihood free Bayesian inference

Setup: We do not have an expression for the likelihood $\pi(x|\theta)$ — not even $f(x|\theta)$.

BUT, given θ, we can generate $x|\theta \sim \pi(x|\theta)$.

Likelikelihood free Bayesian inference

Setup: We do not have an expression for the likelihood $\pi(x|\theta)$ — not even $f(x|\theta)$.

BUT, given θ, we can generate $x|\theta \sim \pi(x|\theta)$.

We can simulate from the posterior, $\theta|x \sim \pi(\theta|x)$, as follows:

```
* Repeat steps 1 and 2...
1       Generate $\theta \sim \pi(\theta)$
2       Generate $\tilde{x}|\theta \sim \pi(x|\theta)$
* Until $\tilde{x} = x$.
* Return $\theta$
```

The repeat loop generates prior predictions until $\tilde{x}$ matches the data x (exactly).

In most situations of interest the probability of $\tilde{x} = x$ is very small — maybe zero. In other words: the algorithms does not work in practise.

Approximate Bayesian Computations (ABC)

The idea is to make an approximation of the previous algorithm:

Assume we have *distance* measue $d(x, x')$ which measures the difference between two data sets.

We can simulate from the posterior, $\theta|x \sim \pi(\theta|x)$, as follows:

```
* Repeat steps 1 and 2...
1         Generate $\theta \sim \pi(\theta)$
2         Generate $\tilde{x}|\theta \sim \pi(x|\theta)$
* Until $d(\tilde{x}, x) < \epsilon$.
* Return $\theta$
```

The repeat loop generates prior predictions until $\tilde{x}$ *approximatly* matches the data x.

If $\epsilon = \infty$: We sample the prior.

If $\epsilon = 0$: We sample the posterior.

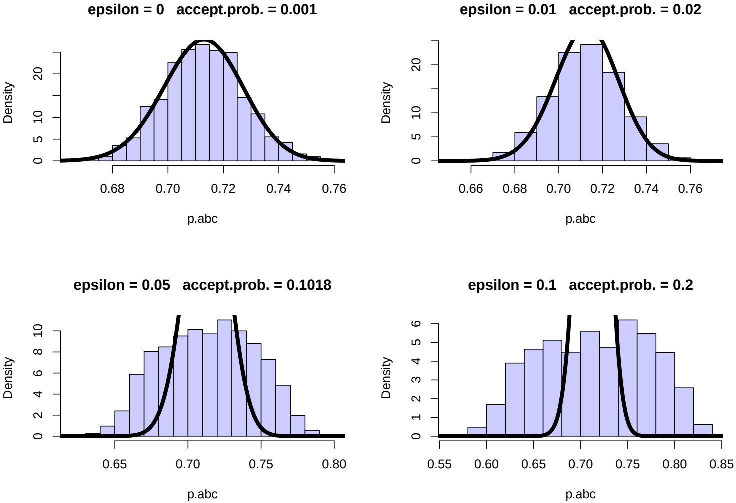
ABC: Binomial model

- **Data model**: $X \sim B(n, p)$, n known
- **Data**: x the number of successes
- **Prior**: $\pi(p) = Be(\alpha, \beta)$ — beta distribution
- **Posterior**: $\pi(p|x) = Be(\alpha + x, \beta + n - x)$

- Distance: $d(x, x') = |\frac{x}{n} - \frac{x'}{n}|$

ABC: Binomial model —*cont.*

In example: $n = 1000$ and $x = 713$ and 2500 samples



ABC MCMC

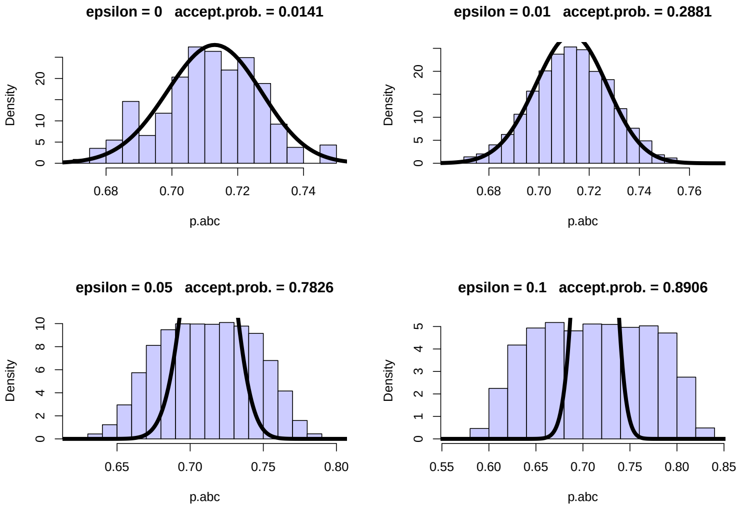
A Metropolis-Hastings style ABC algorithm

```
Choose inital value $\theta^{(0)}$
For $t = 1, \ldots, T$
    Generate $\theta' \sim q(\theta^{(t-1)}, \theta')$
    Generate $x'|\theta' \sim \pi(x'|\theta')$
    If $d(x, x') > \epsilon$ reject and set $\theta^{(t)} = \theta^{(t-1)}$.
    If $d(x, x') \leq \epsilon$
        Calculate $H(\theta^{(t-1)}, \theta') = \frac{\pi(\theta')}{\pi(\theta^{(t-1)})} \frac{q(\theta', \theta^{(t-1)})}{q(\theta^{(t-1)}, \theta')}$
        Generate $u \sim Unif([0, 1])$.
        If $u < H(\theta^{(t-1)}, \theta')$ set $\theta^{(t)} = \theta'$ else set $\theta^{(t)} = \theta^{(t-1)}$
```

Notice the Hastings ratio only involves the priors.

ABC: Binomial model —*cont.*

In example: $n = 1000$ and $x = 713$ and 25,000 iterations.



Mixture models

The density of the data model may take the form

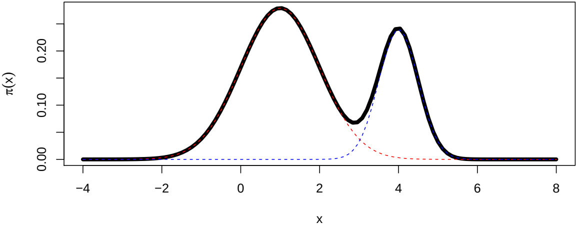
$$\pi(y|\theta) = \lambda_1 \pi_1(y|\theta_1) + \lambda_2 \pi_2(y|\theta_1) + \cdots + \lambda_k \pi_k(y|\theta_1)$$

where each $\pi_j(y|\theta_j)$ is a normalised density and $\lambda_1, \ldots, \lambda_k \geq 0$ are weight with $\sum_{j=1}^{k} \lambda_j = 1$.
This is refered to as a (k component) mixture distribution.

Mixture model: example

Mixture distribution consisting of two normal distributions, ie. $\pi_j(y;\theta_j) = \mathcal{N}(y;\mu_j,\sigma_j^2)$:

- Component 1: $\lambda_1 = 0.7$ $\mu_1 = 1$ $\sigma_1^2 = 1$
- Component 2: $\lambda_2 = 0.3$ $\mu_2 = 4$ $\sigma_2^2 = 0.25$



Notice that the likelihood is *exactly* the same if we swap $(\lambda_1, \mu_1, \sigma_1^2)$ and $(\lambda_2, \mu_2, \sigma_2^2)$. In other words: the model is symmetric in the mixture components.

Direchlet distribution

To perform a Bayeisan analysis we need a prior for the unknown weights.

A k dimensional random vector $\theta = (\theta_1, \ldots, \theta_k)$ is said to follow a Dirichlet distribution with parameters $\alpha = (\alpha_1, \ldots, \alpha_k)$, $\alpha_i > 0$, if is has density

$$\pi(\theta|\alpha) \propto \prod_{j=1}^{k} \theta_j^{\alpha_j - 1},$$

where $\theta_j \geq 0$ and $\sum_{j=1}^{k} \theta_j = 1$.

As a prior for λ we typicaly use a $Dirichlet(1, 1, \ldots, 1)$ prior, ie. a uniform distribution on the allowed set of weights.

The posterior

We assume a priori that the parameters for each component are independent and independent of the weight:

$$\pi(\theta, \lambda) = \prod_{j=1}^{k} \pi(\theta_j)\pi(\lambda)$$

The posterior is then

$$\pi(\theta, \lambda | y) \propto \pi(\theta, \lambda) \prod_{i=1}^{n} \sum_{j=1}^{k} \pi_j(y_i|\theta_j)\lambda_j$$

The full conditional for θ_j is

$$\pi(\theta_j | \theta_{-j}, \lambda, y) \propto \pi(\theta_j) \prod_{i=1}^{n} \sum_{j=1}^{k} \pi_j(y_i|\theta_j)\lambda_j$$

Problem: Even in a simple setup with normal mixtures this is a non standard distribution — which quickly becomes intractable.

Solution: Make the model more complicated!

Dummy variables

For each observation y_i, introduce dummy variable $z_i \in \{1, \ldots, k\}$ which indicates which component x_i belongs to.

$$\pi(y_i|z_i = j, \theta) = \pi_l(y_i|\theta_j)$$

Let λ_j be the a priori probability that y_i belongs to the $j$the component. Hence $\pi(z_i = j|\lambda) = \lambda_j$.

This can be combined to

$$\begin{aligned} \pi(y_i, z_i|\lambda, \theta) &= \pi(y_i|z_i, \theta, \lambda)\pi(z_i|\theta, \lambda) \\ &= \pi(y_i|z_i, \theta)\pi(z_i|\lambda) \end{aligned}$$

Next: Verify that the introduction of the z_is does not change the model.

Marginal distributiuon of y_i

The marginal disitribution of y_i is

$$\begin{aligned}\pi(y_i|\theta) &= \sum_{j=1}^{k} \pi(y_i, z_i = j|\lambda, \theta) \\ &= \sum_{j=1}^{k} \pi(y_i|z_i = j, \theta)\pi(z_i = j|\lambda) \\ &= \sum_{j=1}^{k} \pi_j(y_i|\theta_j)\lambda_j\end{aligned}$$

Hence, the marginal distribution af y_i is unaffected by the introduction of the indicator variables.

Rewriting the likelihood

Notice that

$$\pi(y_i|z_i = l, \theta) = \pi_l(y_i|\theta_l) = \prod_{j=1}^{k} \pi_j(y_i|\theta_j)^{1[z_i=l]}$$

Similarly

$$\pi(z_i = l|\lambda) = \lambda_l = \prod_{j=1}^{k} \lambda_j^{1[z_i=l]}$$

Likelihood

$$\pi(y, z|\lambda, \theta) = \prod_{i=1}^{n} \prod_{j=1}^{k} \left(\pi_j(y_i|\theta_j)\lambda_j\right)^{1[z_i=j]}$$

Full conditionals: z_i

The full conditional for the dummy variables:

$$\begin{aligned}\pi(z_s = l|y,\theta,\lambda,z_{-s}) &\propto \pi(\theta,\lambda)\prod_{i=1}^{n}\prod_{j=1}^{k}\left(\pi_j(y_i|\theta_j)\lambda_j\right)^{1[z_i=j]} \\ &\propto \pi(\theta,\lambda)\prod_{j=1}^{k}\left(\pi_j(y_s|\theta_j)\lambda_j\right)^{1[z_s=j]} \\ &\propto \pi_l(y_s|\theta_l)\lambda_l\end{aligned}$$

Normalising the probabilty we obtain

$$\pi(z_s = l|y,\theta,\lambda,z_{-s}) = \frac{\pi_l(y_s|\theta_j)\lambda_l}{\sum_{j=1}^{k}\pi_j(y_s|\theta_j)\lambda_j}$$

This is a simple distribution to sample from.

Full conditionals: θ_l

The full conditional for θ_j is

$$\begin{aligned}\pi(\theta_l|\theta, y, z) &\propto \pi(\theta, \lambda) \prod_{i=1}^{n} \prod_{j=1}^{k} \left(\pi_j(y_i|\theta_j)\lambda_j\right)^{1[z_i=j]} \\ &\propto \pi(\theta_l) \prod_{i=1}^{n} \pi_l(y_i|\theta_l)^{1[z_i=l]} \\ &\propto \pi(\theta_l) \prod_{i:z_i=l} \pi_l(y_i|\theta_l)\end{aligned}$$

This is equavalent to the posterior in the case of independent observations from π_l (restricted to observation for the lth component).

If the mixture components are normal and we choose priors as in ealier lectures, we know how to sample this full conditional.

Full conditionals: λ

The (joint) full conditional distribution of λ is

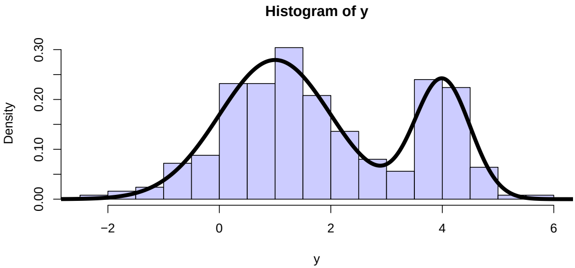
$$\begin{aligned}
\pi(\lambda|\theta, y, z) &\propto \prod_{i=1}^{n}\prod_{j=1}^{k} \big(\pi_j(y_i|\theta_j)\lambda_j\big)^{1[z_i=j]}\pi(\theta,\lambda) \\
&\propto \prod_{i=1}^{n}\prod_{j=1}^{k} \lambda_j^{1[z_i=j]}\pi(\lambda) \\
&\propto \prod_{j=1}^{k} \lambda_j^{n_j(z)} \prod_{j=1}^{k} \lambda_j^{\alpha_j-1} \\
&\propto Direchlet(n_1(z)+\alpha_1, \ldots, n_k(z)+\alpha_k),
\end{aligned}$$

where $n_j(z)$ is the number of dummy variables equal to j.

Sample model

Generate a sample of size 250 from the mixture distribution:

```
N = 250
z = sample(size = N, x = 1:2, prob = lambda, replace = TRUE)
y = rnorm(N, mean = mu[z], sd = sd[z])
```



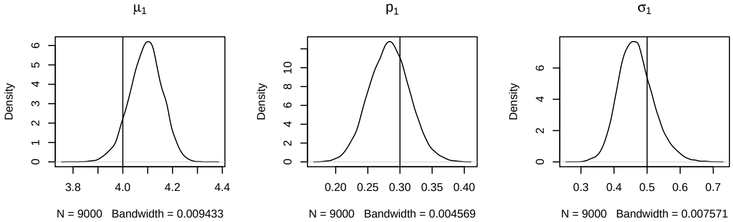
JAGS

```
model{
  # Likelihood:
  for(i in 1 : N){
    y[i] ~ dnorm( mu[z[i]] , tau[z[i]] )
    z[i] ~ dcat( p[1:2])
  }
  # Prior:
  for ( j in 1:2 ) {
    tau[j] ~ dgamma( 0.001 , 0.001 )
    mu[j]  ~ dnorm(0,0.001)
  }
  p ~ ddirch(alpha)
  alpha[1] <- 1
  alpha[2] <- 1
}
```

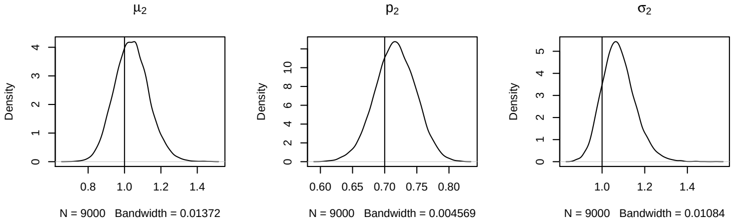
```r
library(rjags)
m1 <- jags.model("mixturemodel.jag", data = list(N = length(y), y = y))
res <- coda.samples(m1, var = c("mu", "tau", "p", "z"), n.iter = 10000)
```

Posterior distributions

Summary for component 1:

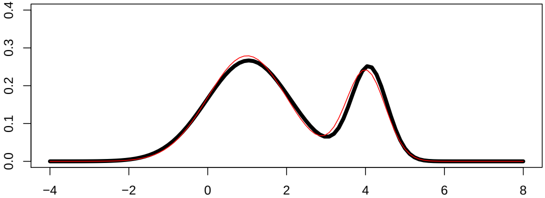


Summary for component 2:



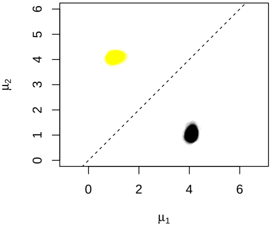
An estimate of the mixture distribution

Comparison of true mixture model (red line) and fitted mixture model (using posterior mean values).



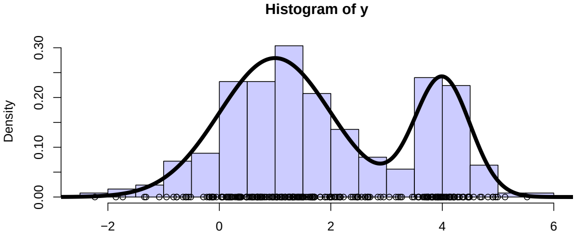
Joint posterior of mean parameters

joint posterior distribution of μ_1 and μ_2 (black) and the expected (due to symmetry) but missing part of the posterior (yellow).

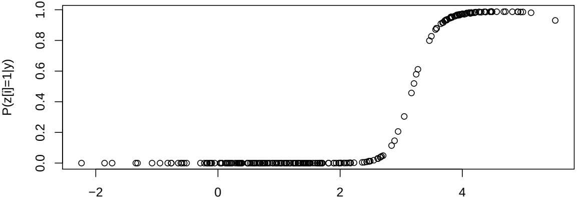


As expected the posterior simulations do not cover both modes.

Posterior probability of $z_i = 1$



Plot of posterior probability that $z_i = 1$ for each y_i:



Notice that this is not simply (a function of) the likelihood ratio.