Part A Simulation and Statistical Programming HT15

6th Practical: MCMC, convergence; Bayesian inference.

1. (a) Write an MCMC algorithm targeting Exp(μ) with mean $\mu = 1$. Check your MCMC.

   (b) Suppose we have an observation $X \sim Exp(\mu)$ and wish to estimate μ. If $X = x$ with $x = 1.7$ (say) and we have a Gamma($a = 3, b = 1$) prior for μ the posterior is

$$\begin{aligned}\pi(\mu|x) &= \mu^{-1}\exp(x/\mu)\mu^{a-1}\exp(-b\mu)\\ &= \mu^{a-2}\exp(-b\mu - x/\mu)\end{aligned}$$

   Write an MCMC algorithm targeting $\pi(\mu|x)$ and use it to give an estimate the posterior mean for $\mu|x$.

2. In lecture 12 we gave an MCMC algorithm to sample a mixture of bivariate normals

$$p(x) \propto 0.5N(x;\mu_1,\Sigma_1) + 0.5N(x;\mu_2,\Sigma_2)$$

with $x = (x_1, x_2)$ etc.

```
a=3; n=2000
mu1=c(1,1); mu2=c(5,5); S=diag(2); S1i=S2i=solve(S);
X=matrix(NA,2,n); X[,1]=x=mu1
for (t in 1:(n-1)) {
   y<-x+(2*runif(2)-1)*a
   MHR<-f(y,mu1,mu2,S1i,S2i)/f(x,mu1,mu2,S1i,S2i)
   if (runif(1)<MHR) x<-y
   X[,t+1]<-x
}

#MCMC simulate X_t according to a mixture of normals
f<-function(x,mu1,mu2,S1i,S2i,p1=0.5) {
  #mixture of normals, density up to constant factor
  c1<-exp(-t(x-mu1)%*%S1i%*%(x-mu1))
  c2<-exp(-t(x-mu2)%*%S2i%*%(x-mu2))
  return(p1*c1+(1-p1)*c2)
}
```

   (a) Modify the MCMC so that the proposal is $y_i \sim N(x_i, a^2), i = 1, 2$.

   (b) Modify the mixture distribution so that it targets

$$p(x) \propto (1/3)N(x;\mu_1,\Sigma_1) + (1/3)N(x;\mu_2,\Sigma_2) + (1/3)N(x;\mu_3,\Sigma_3)$$

   with $\mu_3 = (9, 9)^T$ and $\Sigma_3 = I_2$ (same as Σ_2 etc). Run your MCMC and make a scatter plot of the run in $\Re^2$.

(c) Run your MCMC for varying values of a such as $a = 0.1, 1, 10, 100$ plotting the MCMC trace of `X[1,]`, the first component.

(d) How would you define the "best" value of a?

3. (optional) The binary image below has been corrupted by "salt and pepper" noise.

If the true image was an $n \times n$ matrix X of 0's and 1's, then we observe Y where, for $i = 1, 2, ..., n^2$,

$$Y_i = \begin{cases} X_i & \text{with probability } p \\ 1 - X_i & \text{otherwise} \end{cases}$$

Use the Ising model prior for X with smoothing parameter $\theta = 0.8$. Let $|X - Y|$ give the number of pixels disagreeing between X and Y. The likelihood for X is

$$L(X; Y) = p^{n^2 - |X-Y|}(1-p)^{|X-Y|}.$$

Copy the true `Amat.txt` and noisy images `Ymat.txt` from

`www.stats.ox.ac.uk/~nicholls/PartASSP/data/`

and load them using

```
Xtrue<-as.matrix(read.table("Amat.txt"))
Ynoisy<-as.matrix(read.table("Ymat.txt"))
```

(it is convenient to convert the dataframes to matrices for the `image()` command to make nice pictures). Modify the MCMC code (which targets the posterior for the normal error model with Ising prior) from the lecture to target the posterior distribution

$$\pi(x|y) \propto p^{n^2 - |X-Y|}(1-p)^{|X-Y|} \exp(-\theta \# x).$$

Estimate the posterior mean for $X|Y$ using MCMC.