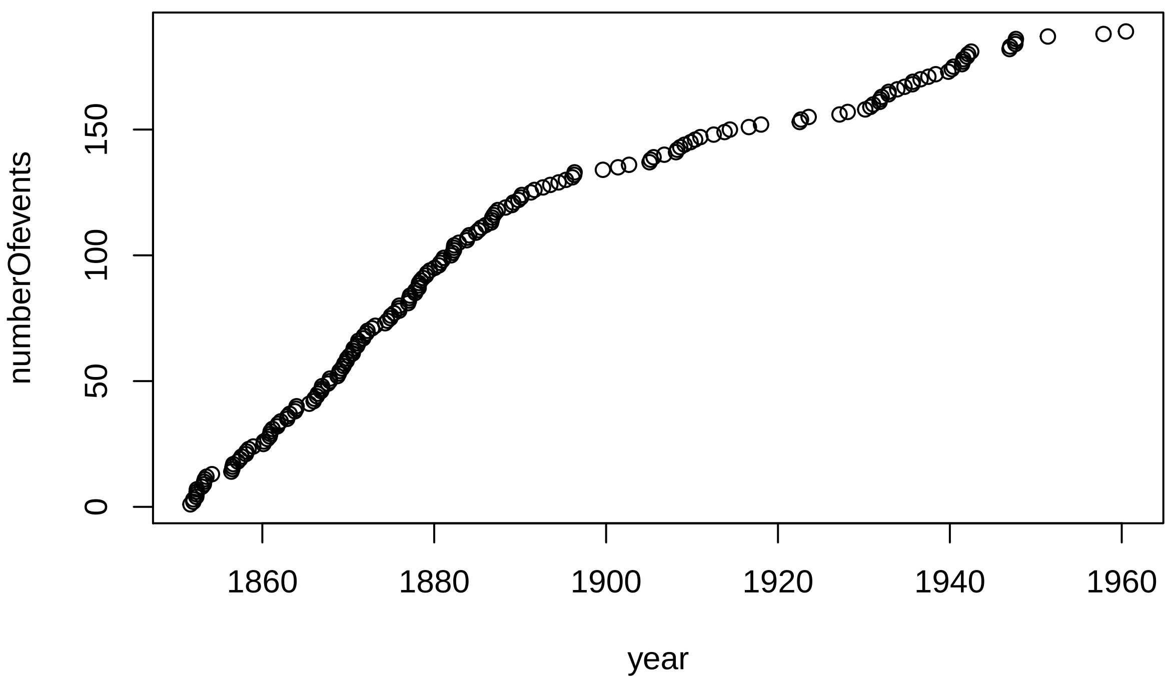
CompulsoryExercise2

Jens&Sindre

Problem A: The Coal-mining Disaster Data

We analyze a data set of time intervals between sucessive coal-mining disasters in the UK involving ten or more men killed (Jarrett, 1979). The data set is for the period March 15th 1851 to March 22nd 1962. In this period, there were 189 coal-mining disasters. In order to get a first impression of the data, we make a plot with year along the x-axis and cumulative number of disasters along the y-axis.

```
year = coal[2:190,]
numberOfevents = seq(1,189,1)
plot(year,numberOfevents)
```



From figure 1 we notice that the intensity of coal-mining disasters appear to be fairly constant in the time interval from 1860 to about 1890, as well as in the interval from 1900 to 1960. There were far more accidents in the first time interval than the last.

We adopt a hierarchical Bayesian model to analyze the data set and assume that the coal-mining disasters follow an inhomogeneous Poisson process with with intensity function $\lambda(t)$. We assume $\lambda(t)$ to be piecewise constant with n breakpoints. We let t_0 and t_{n+1} denote the start and end times for the data set and let $t_k; k = 1, \ldots, n$ denote the break points of the intensity function. Hence,

$$\lambda(t) = \begin{cases} \lambda_{k-1} & \text{for } t \in [t_{k-1}, t_k),\ k = 1, \ldots, n, \\ \lambda_n & \text{for } t\ \in [t_n, t_{n+1}]. \end{cases}$$

2. We now find an expression of the data likelihood for the observed data x, $f(x|t_1,\dots,t_n,\lambda_0,\dots,\lambda_n)$. We define z_i as the number of coal-mining accidents occuring in the between $[t_i,t_{i+1})$ for $i=0,\dots,n$. We use that for the Poisson the number of events in the disjoint intervals $[t_0,t_1),\dots,[t_{n-1},t_n),[t_n,t_{n+1}]$ are independent and Poisson distributed. Thus

$$f(x|t_1,\dots,t_n,\lambda_0,\dots,\lambda_n)=\prod_{i=0}^{n} f(x|t_i,t_{i+1},\lambda_i)=\prod_{i=0}^{n}\frac{(\lambda_i(t_{i+1}-t_i))^{z_i}}{z_i!}e^{-\lambda_i(t_{i+1}-t_i)}$$

We assume $t_1,\dots,t_n$ to be apriori uniformly distributed on the allowed values and $\lambda_0,\dots,\lambda_n$ to be apriori independent of $t_1,\dots,t_n$ and apriori independent each other. Apriori we assume all $\lambda_0,\dots,\lambda_n$ to be distributed from the same gamma distribution with shape parameter $\alpha=2$ and scale parameter β, i.e.

$$f(\lambda_i|\beta)=\frac{1}{\beta^2}\lambda_i e^{-\frac{\lambda_i}{\beta}}\text{ for }\lambda_i\geq 0,\ i=1,\dots,n.$$

Finally, for β we use the improper prior

$$f(\beta)\propto\frac{\exp\{-\frac{1}{\beta}\}}{\beta}\text{ for }\beta>0.$$

In the following we assume $n=1$ and define $\theta=(t_1,\lambda_0,\lambda_1,\beta)$.

3. We find an expression for the posterior distribution (up to a normalising constant) for θ given x, $f(\theta|x)$.

   $f(\theta|x)\propto f(\theta,x)\propto f(x|\theta)f(\theta)\propto f(x|t_1,\lambda_0,\lambda_1,\beta)f(t_1,\lambda_0,\lambda_1,\beta)\propto f(x|t_1,\lambda_0,\lambda_1)f(t_1|\lambda_0,\lambda_1,\beta)\propto f(x|t_1,\lambda_0,\lambda_1)f(t_1)f(\lambda$

   The first term is what we calculated earlier and we know the prior of the other distributions. Hence, the posterior distribution is given by

$$f(\theta|Z)=C\cdot\frac{(\lambda_0(t_1-t_0))^{z_0}}{z_0!}e^{-\lambda_0(t_1-t_0)}\cdot\frac{(\lambda_1(t_2-t_1))^{z_1}}{z_1!}e^{-\lambda_1(t_2-t_1)}\cdot\frac{1}{\beta^2}\lambda_0e^{-\lambda_0/\beta}\cdot\frac{1}{\beta^2}\lambda_1e^{-\lambda_1/\beta}\cdot\frac{e^{-\frac{1}{\beta}}}{\beta^2}$$

   where C is the normalising constant.

4. We now find the full conditionals for each of the elements in θ. We start by finding $\beta|t_1,\lambda_0,\lambda_1,x$. Hence,

$$f(\beta|t_1,\lambda_0,\lambda_1,x)\propto f(\beta,t_1,\lambda_0,\lambda_1,x)$$

   We exclude everything that is now constant to find

$$f(\beta|t_1,\lambda_0,\lambda_1,x)\propto\frac{1}{\beta^5}e^{-\frac{1}{\beta}(1+\lambda_0+\lambda_1)}$$

   which we recognice as the inverse gamma distribution with the following parameters

$$\beta|t_1,\lambda_0,\lambda_1,x\sim\text{InvGamma}(4,1/(1+\lambda_0+\lambda_1))$$

   Furthermore, we determine the distributions $\lambda_0|t_1,\lambda_1,\beta,x$ and $\lambda_1|t_1,\lambda_0,\beta,x$ in the same manner.

$$f(\lambda_0|t_1,\lambda_1,\beta,x)\propto f(\lambda_0,t_1,\lambda_1,\beta,x)$$

   Excluding everything that is now constant and combining the terms with λ_0, we get

$$f(\lambda_0|t_1,\lambda_1,\beta,x)\propto\lambda_0^{z_0+1}e^{-\lambda_0(t_1-t_0+1/\beta)}$$

   This is the well-known gamma distribution. Hence,

$$\lambda_0|t_1,\lambda_1,\beta,x\sim\text{Gamma}(z_0+2,1/(t_1-t_0+1/\beta))$$

   Similarly,

$$\lambda_1|t_1,\lambda_0,\beta,x\sim\text{Gamma}\sim(z_1+2,1/(t_2-t_1+1/\beta))$$

Finally, we determine $t_1|\lambda_0, \lambda_1, \beta, x$ using the same trick as earlier.

$$f(t_1|\lambda_0, \lambda_1, \beta, x) \propto f(t_1|\lambda_0, \lambda_1, \beta, x)$$

This becomes

$$f(t_1|\lambda_0, \lambda_1, \beta, x) \propto \frac{(\lambda_0(t_1 - t_0))^{z_0}}{z_0!} \frac{(\lambda_1(t_2 - t_1))^{z_1}}{z_1!} e^{-\lambda_0(t_1-t_0)} e^{-\lambda_1(t_2-t_1)}$$

which we do not recognize as a well-known named distribution.

5. We will now define and implement a single site MCMC algorithm for $f(\theta|x)$.

Include libraries

```
library(boot)
library(invgamma)
library(mcmc)
library(coda)
library(ggplot2)
```

Task 5 First, we create a help function that divides the number of coal-mining disasters into two, in the intervals $[t_0, t_1)$ and $[t_1, t_2]$.

```
divide_data = function(t) {
  coalEvents = coal[2:190,]
  indicator = coalEvents < t
  index = sum(indicator)
  return (index)
}
```

In order to sample from $t_1|\lambda_0, \lambda_1, \beta, x$, we will use a random walk proposal. We propose a $\tilde{t}|t_1$ from a normal distribution with mean t_1, i.e. the last step, and tuning parameter σ, $\tilde{t}|t_1 \sim N(t_1, \sigma^2)$. The Q's in the acceptance probability term cancels so that the acceptance probability becomes

$$\alpha = \min\left(1, \frac{f(\tilde{t})}{f(t)} \cdot \frac{Q(t)}{Q(\tilde{t})}\right) = \min\left(1, \frac{f(\tilde{t})}{f(t)}\right)$$

where

$$\frac{f(\tilde{t}|\tilde{z})}{f(t|z)} = \frac{\frac{(\lambda_0(\tilde{t}_1-t_0))^{\tilde{z}_0}}{\tilde{z}_0!} \cdot \frac{(\lambda_1(t_2-\tilde{t}_1))^{\tilde{z}_1}}{\tilde{z}_1!} \cdot e^{-\lambda_0(\tilde{t}_1-t_0)} \cdot e^{-\lambda_1(t_2-\tilde{t}_1)}}{\frac{(\lambda_0(t_1-t_0))^{z_0}}{z_0!} \cdot \frac{(\lambda_1(t_2-t_1))^{z_1}}{z_1!} \cdot e^{-\lambda_0(t_1-t_0)} \cdot e^{-\lambda_1(t_2-t_1))}}$$

The MCMC algorithm is shown below. Note that the acceptance probability is computed on log-scale.

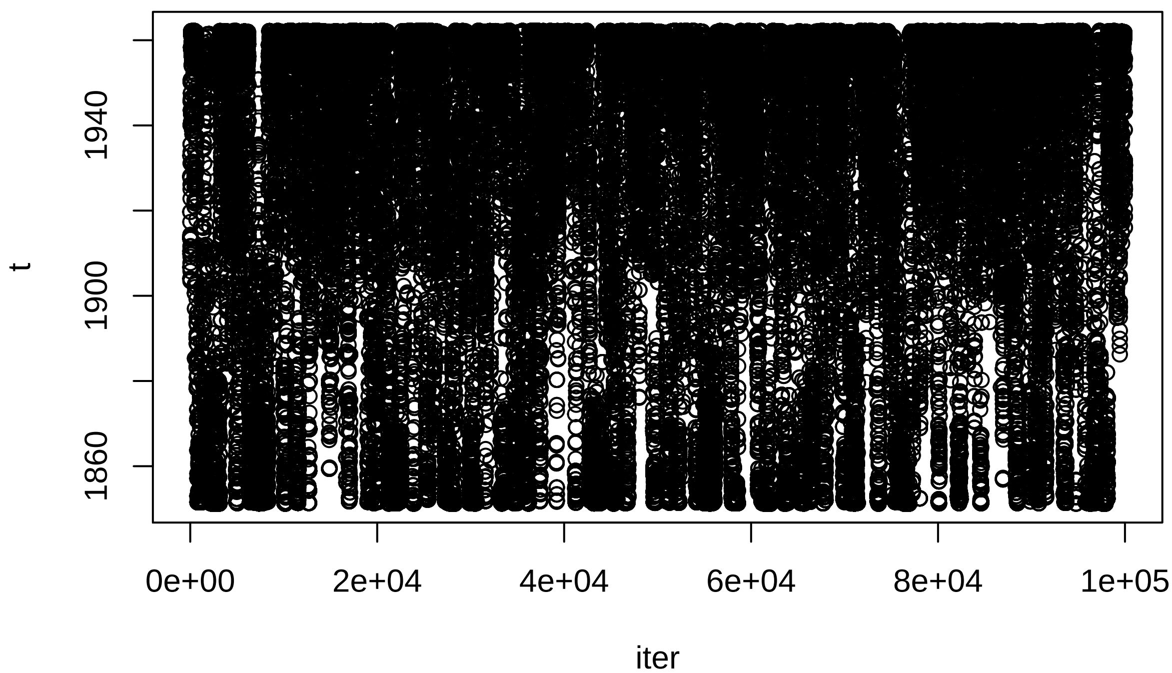
```
# intital cond
n = 100000

t_0 = coal[1,]
t_2 = coal[191,]
lambda0 = rep(0,n)
lambda1 = rep(0,n)
beta = rep(0,n)
beta[1] = 1
t= rep(0,n)
sigma = (t_2-t_0)/7
#sigma = 8
t[1] = (t_0+t_2)/2
lambda0[1] = 3
lambda1[1] =1
N0 = rep(0,n)
```

```r
N1 = rep(0,n)
index = divide_data(t[1])
N0[1] = index
N1[1] = 189-index
counterAccRate = 0
fraction_vec = rep(0,1)


#Metropolis Hastings, we know how to sample beta, and lambda, and always accept the proposals. For t, u

for (i in 1:(n-1)) {
  #i = 1
  t_prop = rnorm(1,t[i],sigma)
  #find split of N_prop
  index = divide_data(t_prop)
  N0_prop = index
  N1_prop = 189-(index)

  t_out_of_range = (t_prop > t_2) || (t_prop < t_0)
  #find acceptance probability
  if (!t_out_of_range) {
    logterm1 = N0_prop*(log(lambda0[i]) + log(t_prop-t_0)) + N1_prop*(log(lambda1[i]) + log(t_2-t_prop))
    logterm2 = N0[i]*(log(lambda0[i]) + log(t[i]-t_0)) + N1[i]*(log(lambda1[i]) + log(t_2-t[i]))
    logterm3 = (lambda1[i] -lambda0[i])*(t_prop-t[i])
    logterm4 = lfactorial(N0_prop) + lfactorial(N1_prop)
    logterm5 = lfactorial(N0[i]) + lfactorial(N1[i])
    fraction = (logterm1 - logterm2 + logterm3 - logterm4 + logterm5)
    fraction_vec[i] = fraction
  }
  else {
    fraction = -Inf
  }

  alpha = min(0,fraction)
  u = log(runif(1))
  if (u < alpha) { # if true accept new proposal
    N0[i+1] = N0_prop
    N1[i+1] = N1_prop
    t[i+1] = t_prop
    counterAccRate = counterAccRate + 1
  }
  else { #or if false, keep the old state
    N0[i+1] = N0[i]
    N1[i+1] = N1[i]
    t[i+1] = t[i]
  }

  #sample beta
  beta[i+1] = rinvgamma(1,shape = 4, scale = 1/(lambda0[i] + lambda1[i] + 1))
  #sample lambda
  lambda0[i+1] = rgamma(1,shape = N0[i+1] + 2,scale = 1/((t[i+1]-t_0)+1/beta[i+1]))
  lambda1[i+1] = rgamma(1,shape = N1[i+1] + 2,scale = 1/((t_2-t[i+1]) + 1/beta[i+1]))
```

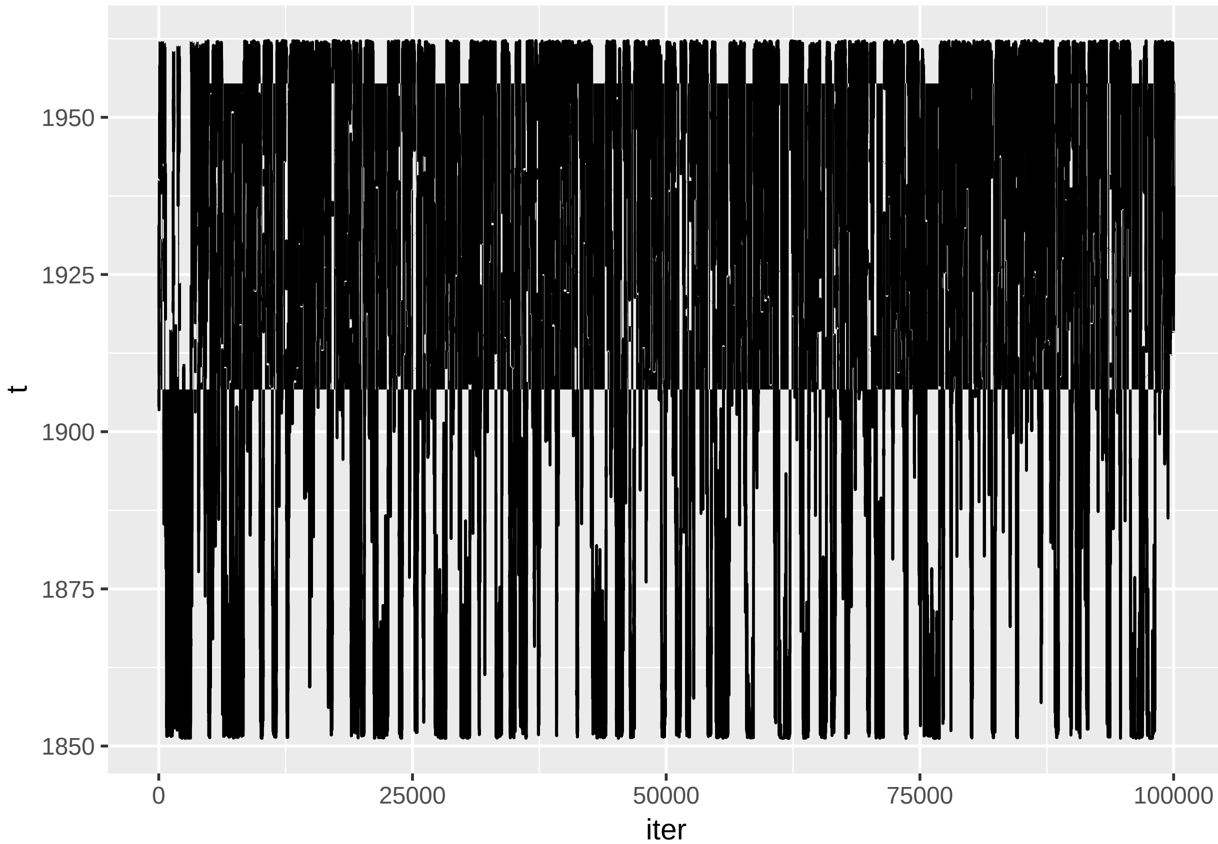
```r
}
counterAccRate/n
```

```
## [1] 0.24431
```

```r
iter = seq(1,n,1)
plot(iter,t)
```




```r
ggplot(data = as.data.frame(t), aes(x = iter, y = t) ) + geom_path()
```




```r
mean(lambda0)
```

```
## [1] 2.227385
```

```r
mean(lambda1)
```

```
## [1] 1.278374
```

Task 8 a) A block proposal for $(t_1, \lambda_0, \lambda_1$ keeping β unchanged. Generate the potential new values $(\tilde{t}_1, \tilde{\lambda}_0, \tilde{\lambda}_1$ by first generating $\tilde{t}_1$ from a normal distributi1on centered at the current value of t_1 and thereafter generate $\tilde{\lambda}_0, \tilde{\lambda}_1$ from their joint full conditionals inserted the potential new value t_1, i.e. $f(\lambda_1, \lambda_2|x, t_1, \beta)$. b) Now the acceptance probability becomes where

$$\frac{f(\tilde{t}|\tilde{z})}{f(t|z)} = \frac{(\tilde{\lambda}_0(\tilde{t}_1 - t_0))^{\tilde{z}_0} \cdot (\tilde{\lambda}_1(t_2 - \tilde{t}_1))^{\tilde{z}_1} \cdot exp(\tilde{\lambda}_0(\tilde{t}_1 - t_0)) \cdot exp(\tilde{\lambda}_1(t_2 - \tilde{t}_1))}{(\lambda_0(t_1 - t_0))^{z_0} \cdot (\lambda_1(t_2 - t_1))^{z_1} \cdot exp(\lambda_0(t_1 - t_0)) \cdot exp(\lambda_1(t_2 - t_1))}$$

And

$$\frac{Q(\lambda_0, \lambda_1, t_1|\tilde{\lambda}_0, \tilde{\lambda}_1, \tilde{t}_1)}{Q(\tilde{\lambda}_0, \tilde{\lambda}_1, \tilde{t}_1|\lambda_0, \lambda_1, t_1)} = \frac{Q(\lambda_0) \cdot Q(\lambda_1) \cdot Q(t_1)}{Q(\tilde{\lambda}_0) \cdot Q(\tilde{\lambda}_1) \cdot Q(\tilde{t}_1)} = \frac{Q(\lambda_0) \cdot Q(\lambda_1)}{Q(\tilde{\lambda}_0) \cdot Q(\tilde{\lambda}_1)} = \frac{\lambda_0 exp(\frac{\lambda_0}{\beta}) \cdot \lambda_1 exp(\frac{\lambda_1}{\beta})}{\tilde{\lambda}_0 exp(\frac{\tilde{\lambda}_0}{\beta}) \cdot \tilde{\lambda}_1 exp(\frac{\tilde{\lambda}_1}{\beta})}$$

Task 8 a) A block proposal for $(t_1, \lambda_0, \lambda_1$ keeping β unchanged. Generate the potential new values $(\tilde{t}_1, \tilde{\lambda}_0, \tilde{\lambda}_1$ by first generating $\tilde{t}_1$ from a normal distributi1on centered at the current value of t_1 and thereafter generate $\tilde{\lambda}_0, \tilde{\lambda}_1$ from their joint full conditionals inserted the potential new value t_1, i.e. $f(\lambda_1, \lambda_2|x, t_1, \beta)$. b) Now the acceptance probability becomes

$$\alpha = min(1, \frac{f(\tilde{t}, \tilde{\lambda}_0, \tilde{\lambda}_1|\tilde{z})}{f(t, \lambda_0, \lambda_1|z)} \cdot \frac{Q(\lambda_0, \lambda_1, t_1|\tilde{\lambda}_0, \tilde{\lambda}_1, \tilde{t}_1)}{Q(\tilde{\lambda}_0, \tilde{\lambda}_1, \tilde{t}_1|\lambda_0, \lambda_1, t_1)})$$

where

$$\frac{f(\tilde{t},\tilde{\lambda}_0,\tilde{\lambda}_1|\tilde{z})}{f(t,\lambda_0,\lambda_1|z)} = \frac{(\tilde{\lambda}_0(\tilde{t}_1-t_0))^{\tilde{z}_0}\cdot(\tilde{\lambda}_1(t_2-\tilde{t}_1))^{\tilde{z}_1}\cdot exp(\tilde{\lambda}_0(\tilde{t}_1-t_0))\cdot exp(\tilde{\lambda}_1(t_2-\tilde{t}_1))}{(\lambda_0(t_1-t_0))^{z_0}\cdot(\lambda_1(t_2-t_1))^{z_1}\cdot exp(\lambda_0(t_1-t_0))\cdot exp(\lambda_1(t_2-t_1))}\cdot\frac{z_0!z_1!}{\tilde{z}_0!\tilde{z}_1!}$$

And

$$\frac{Q(\lambda_0,\lambda_1,t_1|\tilde{\lambda}_0,\tilde{\lambda}_1,\tilde{t}_1)}{Q(\tilde{\lambda}_0,\tilde{\lambda}_1,\tilde{t}_1|\lambda_0,\lambda_1,t_1)} = \frac{f(\lambda_0,\lambda_1|z,\beta,t)\cdot Q(t_1|\tilde{t}_1)}{f(\tilde{\lambda}_0,\tilde{\lambda}_1|\tilde{z},\beta,\tilde{t})\cdot Q(\tilde{t}_1|t_1)} = \frac{f(\lambda_0|z,\beta)\cdot f(\lambda_1|z,\beta)}{f(\tilde{\lambda}_0|z,\beta)\cdot f(\tilde{\lambda}_1|z,\beta)} = \frac{\lambda_0^{z_0+1}\lambda^{z_1+1}exp(\lambda_0((t_1-t_0)+1/\beta)exp(\lambda_0((t_2 -}{\tilde{\lambda}_0^{\tilde{z}_0+1}\tilde{\lambda}^{\tilde{z}_1+1}exp(\tilde{\lambda}_0((\tilde{t}_1-t_0)+1/\beta)exp(\tilde{\lambda}_0((t_2 -}$$

- Problem B: Bayesian Image Reconstruction

We study how the ising model can be used as a prior distribution in an image reconstruction setting. We let $y = (y_{ij}, i = 1,\ldots,89, j = 1,\ldots,85)$ be the observed “image.txt”. We assume this to be a noisy version of an unobserved image $x = (x_{ij}, i = 1,\ldots,89, j = 1,\ldots,85)$ with $x_{ij}\in\{0,1\}$. Our goal in this problem is to use the observed y to estimate x. We assume the elements in y to be conditionally independent given x and

$$y_{ij}|x \sim N(\mu_{x_{ij}},\sigma^2_{x_{ij}})$$