Part 2 - Mountain Finding

1. **Using Bayes Net in Figure 1b) –**

Inference from the bayes net as shown in figure 1b), can be drawn for detecting a mountain in the image by making a probability query on the bayes net.

For this the construct\_ridge() method was written which makes a ridge by predicting rows with the mountain edge, by taking a max of the product of emission and transition probabilities for every column.

1. **MCMC on a Bayes Net as in figure 1a) –**

Gibbs Sampling or MCMC was used on a bayes net of observed - edge-intensities and unobserved - row values for every column . This was implemented as the function construct\_ridge2() and runs by taking a sampled particle and then updates the unobserved sampled particle attributes by calculating a probability distribution based on baye’s law. The sampled particle was chosen by taking a best approximate from the edge\_strength by distribution by taking a sample which represents a max of intensities for every column in edge\_strength.

The function probability\_distribution() then calculates the probability distribution of the unobserved attributes of the sample by by first narrowing down the search to 20 rows above and below the intended row and then calculates emission and transition probabilities from those rows into the intended unobserved attribute row of the sampled particle. This is done for every unobserved attribute of the sampled particle previously generated and updated in a dictionary data-structure which keeps track of the most recent sample at time t.

Based on an interval of time during which this function is called, the sampled particle in the dictionary data-structure smoothens given the observed attributes (which is basically the column values and edge-intensities).

The algorithm performs pretty well for a smaller interval of time. It can be perfected by doing the following –

1. Narrowing down the row range in which to look for the transition probabilities for a transition into the unobserved row.
2. Running the sample for a longer amount of time.
3. Or taking human feedback as done in sub-part3.

**Taking Human Feedback**