CS5340

Uncertainty Modelling in AI

Project

Pranav Prathvikumar – A0186517W

1. Image Denoising

(i) Gibbs Sampling Algorithm

Gibbs sampling along with the Ising model was used for denoising a corrupted image. Ising model is a MRF of interacting dipoles. It consists of discrete variables consisting of dipoles of (+1 ,-1) which are arranged in a lattice. The state of each variable depends on the neighbouring variables through interaction potentials.

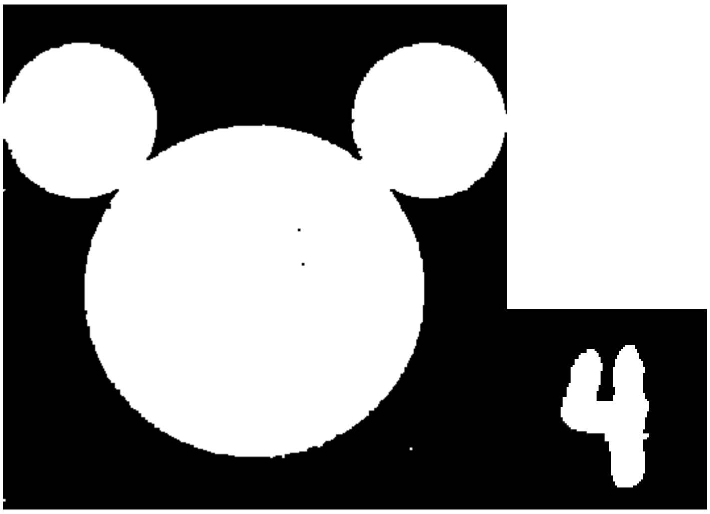
Gibbs sampling is an algorithm for approximating joint distributions by obtaining a sequence of observations from a probability distribution. Each point is sampled while keeping all other points constant while conditioning it on the points in its immediate vicinity (Markov Blanket). The sampled point is updated, and the sequential points sampled are conditioned upon this updated point.

$$p\,(x_t \mid x_{-t}, y, \boldsymbol{\theta}) = \frac{\psi_t(x_t)\prod_{s\in nbr(t)} \psi_{st}(x_t, x_s)}{\sum_{xt} \psi_t(x_t)\prod_{s\in nbr(t)} \psi_{st}(x_t, x_s)}$$

The algorithm is initialized with the image with the noise. The above equation is applied to each pixel. Each pixel's underlying value is assumed to be either (+1,-1) and its relationship with its four neighbours is calculated. Along with this a normal probability distribution is used to see the likelihood of the observed pixel having its current value. The probability of the underlying pixel is then calculated of it being either (+1,-1) and the sample is accepted after comparing it with a uniform distribution. This process is repeated for every pixel in the image and is done repeatedly until the image has been denoised.

Here the coupling constant was taken as 4 and the covariance as 1.

<u>Output:</u>



(ii) Variational Inference

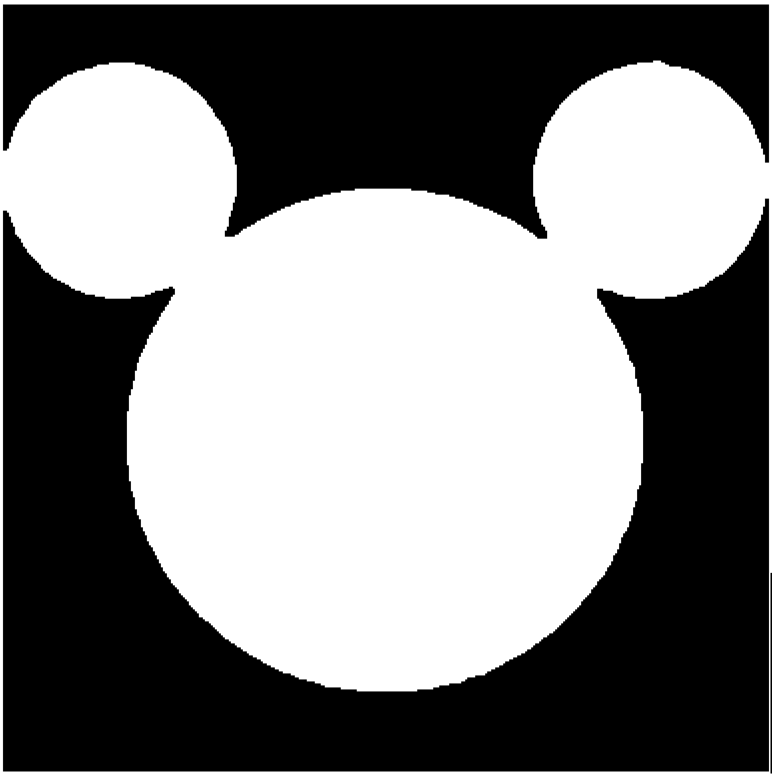
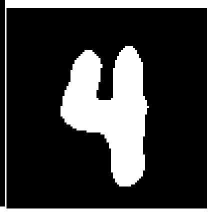
Variational inference along with the Ising model was used for denoising a corrupted image. The basic idea in Variational Inference is to choose and determine an approximating distribution q(x) which is close to the original distribution. The closeness is determined with the help of KL-Divergence and our goal is to minimize it. This is usually done by the Mean field approximation. In it the latent variables are partitioned into disjoint groups. Among these we seek the distribution which would minimize the KL divergence.

$$q(\boldsymbol{x}) = \prod_{i} q(x_{i}, \mu_{i})$$

The algorithm is initialized with the noisy image. In our case, we calculate the prior using the Ising model. We calculate the potentials between the observed pixel and its four neighbours by using a coupling constant. The likelihood is calculated as the log of the gaussian probability distribution of the observed variable with respect to the mean(+1,-1) and the strength constant. These two are then used to estimate the approximation function q(x) for both possible mean values of (+1,-1) and to narrow in on the actual mean. This process is iterated over the entire image repeatedly until it converges. This gives us the final mean value which is the true pixel value.

Here the coupling constant was taken as 1 and the covariance as 4.

Output:

2. Image Segmentation

Image segmentation was carried out with the help of Expectation-Maximisation or EM Algorithm. It is an iterative method to estimate the underlying parameters of a distribution in statistical models and segment. It does so by assuming that the data is a mixture of Gaussian models and has further unobserved data points which help explain the existing data. It iterates between assigning the data points to each Gaussian model and follow it by calculating what would be the best underlying parameters to explain the distribution. It goes on till there is no change in parameters or reaches a terminating condition.

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

In the above equation:

p(x) – Represents the data

π_k – Mixing factor. Represents the percentage of data present in each distribution

x – Represents the data point

μ_k – Mean of the distribution

$\sum_k$ – Covariance of the distribution

The algorithm is initialized with the random values for the mixing factor, mean and covariance. Over each iteration the algorithm allocates points to the distributions and then recalculates the above parameters.

In our case, as we have to segment the image into only the foreground and background, we take the number of distributions to be two. The data points in our case represent the colour scheme of each pixel in the CIE-Lab colour space. When EM algorithm is run on the colour space setting the number of distributions to two, we were able to segment the image into the foreground and background. Two distributions are built with the points corresponding to each segment being a part of a distribution.

Output: