1 Data Denoising

1.1 Gibbs sampling algorithm

We set $\sigma = 1$ and coupling strength $J = 2$.
Our algorithm works as follows: Looping through pixels one by one, calculate $p(x_t|x_{-t}, y, \theta)$ and update x_t right away by get a random number $r_i$between 0 and 1, if $r_i > p(x_t|x_{-t}, y, \theta)$, $x_i = 0$, else $x_i = 1$.

In total we loop through the image 4 times, we treated the first three iteration as burn-in, and take the output from 4th iteration as the output.

1.2 Variational inference algorithm

We are using $q(x)$ to approximate posterior distribution $p(x|y)$ where $q(x) = \prod_i q(x_i, u_i)$ and $p(y|x) = \prod_i p(y_i|x_i)$. We define $x_i \in \{-1, +1\}$ as the state of node x_i, and $y_i \in \{-1, +1\}$ as the observed state of node x_i. Before we derieve $ELBO$, we need to show that $p(x, y) = \prod_i p(x_i, y_i)$.

We prove it as follows:
since $p(x, y) = p(x)p(y|x)$ and we know $p(y|x) = \prod_i p(y_i|x_i)$
We are also given that

$$\begin{aligned} p(x) &= \frac{1}{Z_0} exp(\sum_i \sum_j \in nbr(i) W_{ij} x_i x_j) \\ &= \prod_i \frac{1}{Z_0} exp(\sum_j \in nbr(i) W_{ij} x_i x_j) \\ &= \prod_i p(x_i) \end{aligned}$$

Hence we conclude that $p(x, y) = \prod_i p(x_i, y_i)$.

We defined the $ELBO$ under for this problem as:

$$\begin{aligned} ELBO &= q(x) log \frac{p(x, y)}{q(x)} \\ &= q(x) log(p(x, y) - q(x) log(q(x)) \\ &= \prod_i q(x_i) log(\prod_i p(x_i, y_i)) - \prod_i q(x_i) log(\prod_i q(x_i)) \end{aligned}$$

It is obvious that to minimze ELBO, we just need to make

$$\begin{aligned} log(\prod_i p(x_i, y_i)) &= log(\prod_i q(x_i)) \\ \sum_i log(p(x_i, y_i)) &= \sum_i log(q(x_i)) \end{aligned}$$

As $x_i \in \{-1, +1\}$, it is obvious that $q(x_i, u_i)$ is of Bernoulli distribution.
To simplify this problem even more, we just need to calculate u_i for all i such that $p(x_i, y_i) == q(x_i, u_i)$. More explicitly, to calculate:

$$\frac{1 - u_i}{u_i} = \frac{p(y_i, x_i = 0)}{p(y_i, x_i = 1)}$$

For each iteration, we loop through all pixels one by one, calculate u_i. At the end of each iteration, we calculate x_i for all i using corresponding Bernoulli distribution: get a random number $r_i$between 0 and 1, if $r_i > u_i$, $x_i = 0$, else $x_i = 1$.
We set coupling strength $W_{i,j} = 2$ and $c = 1$ to be consistent with the parameters used in Gibbs sampling algorithm.
In total we loop through the image 4 times, and take the output from 4th iteration as the output.