1 Derivation of conditional distribution

The probability distribution over a vector $y \in \{-1, +1\}^d$ is given by,

$$p(y|b,w) = \frac{1}{Z} \prod_{i=1}^{d} \exp(b_i y_i) \prod_{(i,j)\in \text{pairs}} \exp(w_{ij} y_i y_j).$$

By conditional probability $p(a, b) = p(a|b)p(b)$

$$\begin{aligned}
p(y|b,w) &= p(y_i|y_{-i}, b, w)p(y_{-i}|b,w) \\
p(y_i|y_{-i}, b, w) &= \frac{p(y|b,w)}{p(y_{-i}|b,w)} \\
&= \frac{p(y|b,w)}{\sum\limits_{y_i} p(y|b,w)} \\
p(y_i = 1, y_{-i}|b,w) &= \frac{p(y|b,w)}{p(y_{-i}, y_i = -1|b,w) + p(y_{-i}|y_i = 1, b, w)} \\
&\quad \textit{Substituting the probability definition} \\
&= \frac{1}{Z(y_{-i})} \frac{\prod_{k=1}^{d} \exp(b_k y_k) \prod_{(k,j)\in \text{pairs}} \exp(w_{kj} y_k y_j)}{\sum_{y_k} \prod_{k=1}^{d} \exp(b_k y_k) \prod_{(k,j)\in \text{pairs}} \exp(w_{kj} y_k y_j)}
\end{aligned} \tag{1}$$

The common terms in the numerator and denominator cancel out, and only the terms that contain i survive. Further we substitute both values of y_i. Note, that the normalizing constant depend on y_{-i}

$$\begin{aligned}
p(y_i = 1|y_{-i}, b, w) &= \frac{\exp(b_i) \prod_{j\in nb(i)} \exp(w_{ij} y_j)}{\exp(-b_i) \prod_{j\in nb(i)} \exp(-w_{ij} y_j) + \exp(b_i) \prod_{j\in nb(i)} \exp(w_{ij} y_j)} \\
&\quad \textit{Since, } \exp(a)\exp(b) = \exp(a + b) \\
&= \frac{\exp(b_i + \sum_{j\in nb(i)} w_{ij} y_j)}{\exp(-b_i - \sum_{j\in nb(i)} w_{ij} y_j) + \exp(b_i + \sum_{j\in nb(i)} w_{ij} y_j)} \\
&\quad \textit{Dividing by the numerator} \\
&= \frac{1}{1 + \exp(-2(b_i + \sum_{j\in nb(i)} w_{ij} y_j)))} \\
&= \sigma(2(b_i + \sum_{j\in nb(i)} w_{ij} y_j)) \quad \text{Since, } \sigma(x) = 1/(1 + \exp(-x))
\end{aligned} \tag{2}$$

2 Pseudo-Code for Gibbs Sampling

Algorithm 1: Gibbs Sampling

/* Input model parameters and number of iterations */
Input: t_{max}, b, w
Init $y^0 \in \mathbb{R}^d$;
for $t \leftarrow 1$ *to* t_{max} **do**
    $y^t \leftarrow y^{t-1}$;
    /* Iterate over the dimensions */
    for $i \leftarrow 1$ *to* d **do**
        /* Sample $r \in \{-1, 1\}$ from the conditional distribution */
        $r \backsim P(Y_i | Y_{-i} = y^t_{-i}, b, w)$;
        $y^t_i \leftarrow r$
    end for
end for
return y^0, ..., $y^{t_{max}}$

3 Samples

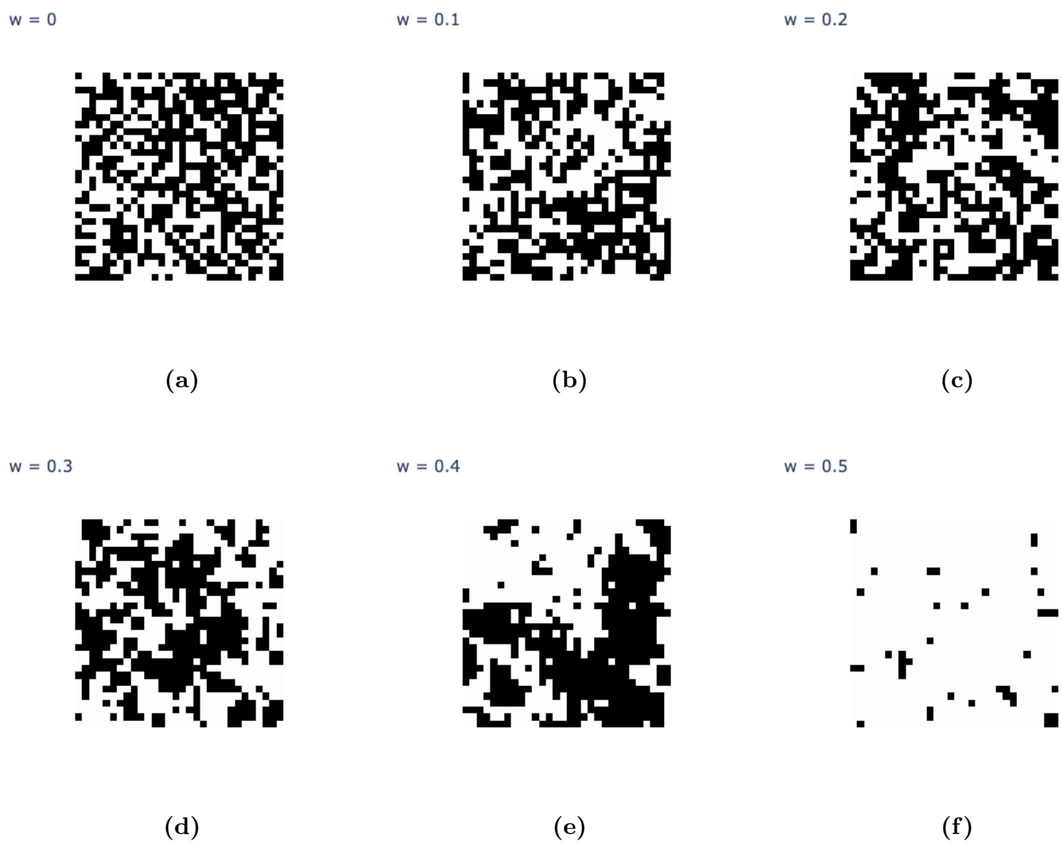


Figure 1 – Samples obtained from Gibbs algorithm over 100 iterations for $\tilde{w} \in \{0..1, .2, .3, .4, .5\}$. y initialzed as $\{+1\} \in \mathbb{R}^d$

4 Discussion

Given the probability distribution $(y|b, w) \propto \exp(w_{i,j} y_i, y_j)$, we see that for higher values of w_{ij}, the neighbouring pixels are inclined to have the same value. Initially, the images are $\{+1\}^d$ and progressively they form larger regions of pixels with the same values. For large values of w ≥ 0.5, the images are mostly white, due to the initialization.

5 Mixing Times

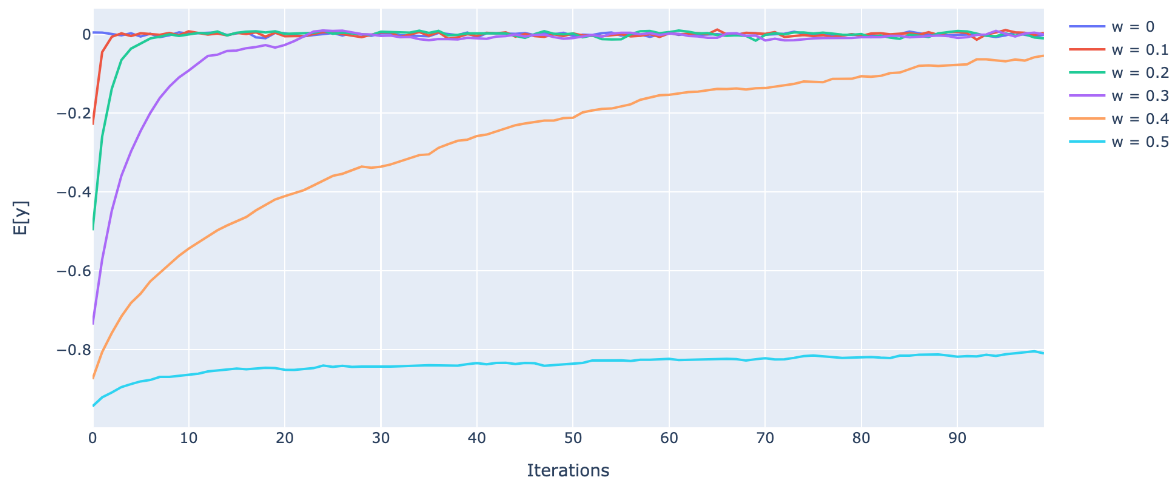


Figure 2 – Plot of the E[y] over 100 independent Markov chains for $\tilde{w} \in \{0..1, .2, .3, .4, .5\}$, when initialized with $\{-1\}^d$

6 Discussion

We see that the rate of convergence of $E[y]$ is inversely proportional to the value of w; smaller $w \in \{0, .1, .2, .3, .4\}$ converge to 0 within 100 iterations, while $w \in \{0.4, 0.5\}$ have not yet converged. The quality of samples improves with the number of iterations. Since Gibbs sampling is initialized with $\{-1\}^d$, higher values of w, would want neighbouring pixles to have same values (-1)

7 Fixed Parameter Denoising

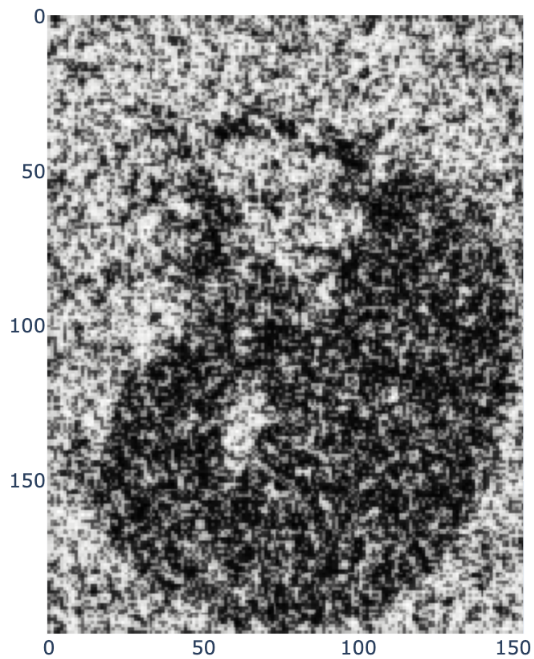
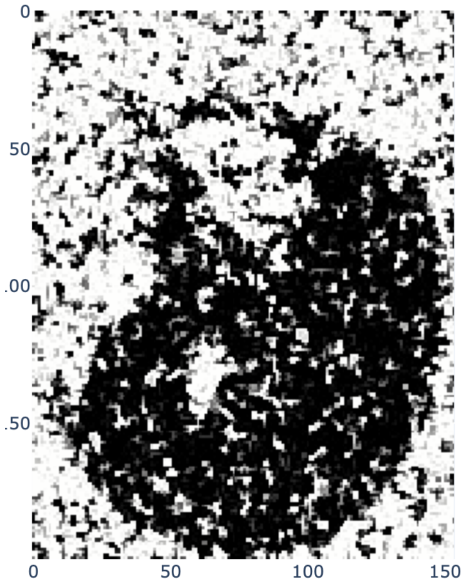


Figure 3 – Mean image obtained from Gibbs with $w = 0.3$ and $b = 0.5$

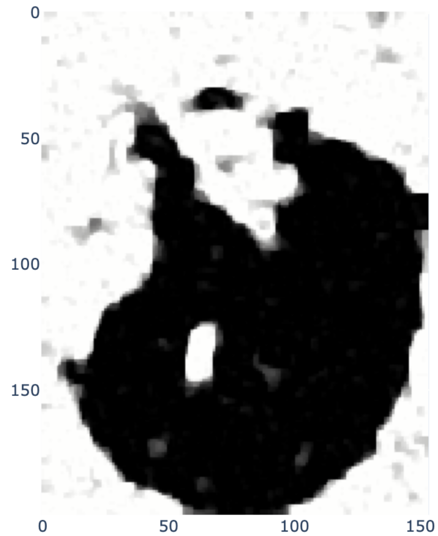
Per-pixel Averaged Mean Absolute Error = `0.65397`

8 Varying Paramters

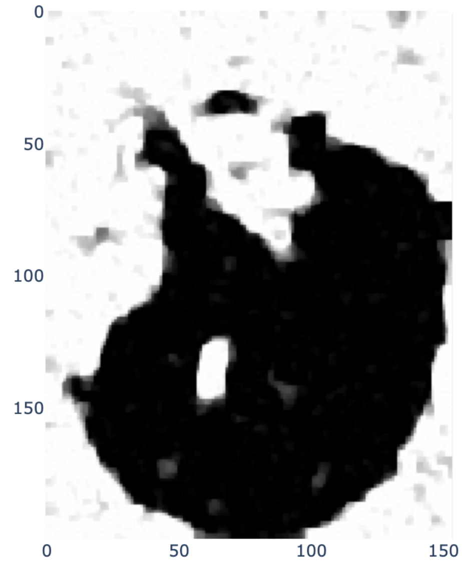
The default values of $w_{ij} = 0.3$ and $b_i = 0.5$ give a $MAE = 0.65397$. From inital analysis, we see that for $\theta_2 > 10, \theta_1 > 2$, there is only a marginal improvement or degradation in MAE, as see in Figure. We limit the ranges of $\theta_1 \in [0, .5, 1, 2]$ and $\theta_2 \in [0, .3, 1, 5, 10]$. We use a *Grid Search* over this parameter space to find the best combination. We find that for $\theta_1 = 0.5$ and $\theta_2 = 5$, we get the lowest error.
Lowest Mean Absolute Error: `0.1366`



(a) High θ_1 value $\theta_1 = 2$, $MAE = 0.25$



(b) High θ_2 value $\theta_2 = 10$, $MAE = 0.15$



(c) Best image obtained from Gibbs with $\theta_2 = 5$ and $\theta_1 = 0.5$, $MAE = 0.136$

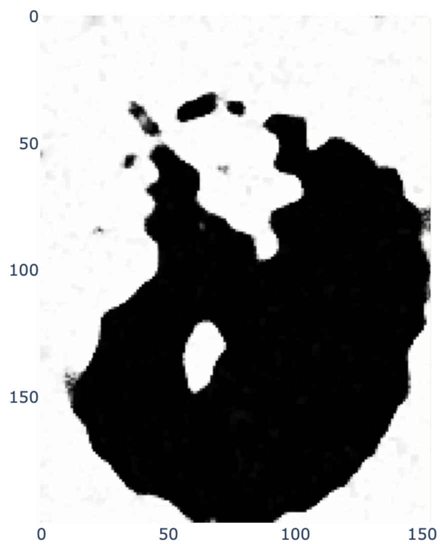
9 Varying w as a function of x

We modify w_{ij} such that it is dependent on x. Thus the new probability distribution $p(y|b,w) \propto \prod_{(i,j)\in pairs} \exp(\theta_2 f(x) y_i y_j)$. Having $f(x)$ term ensures that the samples obtained from the distribution "mimic" the noisy image. This could help preserving the *details* from the noisy input.

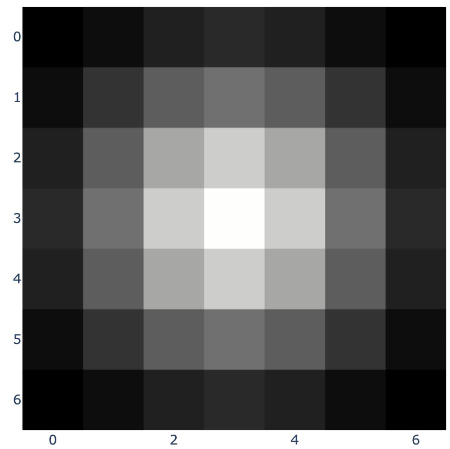
For this experiment, we choose $f(x)$ as a convolution operation. $w_{i,j} = \sum_{k\in nb(x_{i,j})} f_k x_k$. The motivation behind this, is that using the neighbourhood of x_{ij} would give a better estimate of the value of the $x_{i,j}$.

We define a *Gaussian kernel* with filter sizes $f \in \{5, 7, 10, 20\}$ and standard deviation $\sigma \in \{1, 2, 3\}$. We perform a convolution operation on x using this filter with appropriate padding. We take element-wise absolute value and the values bewtween $[3, 12]$ to obtain the final w. We see that a larger kernel size leads to more smoothing with a loss of detail.

Mean Absolute Error = `0.11` (+2%) improvement



(d) Best image, $MAE = 0.11$



(e) Gaussian Kernel