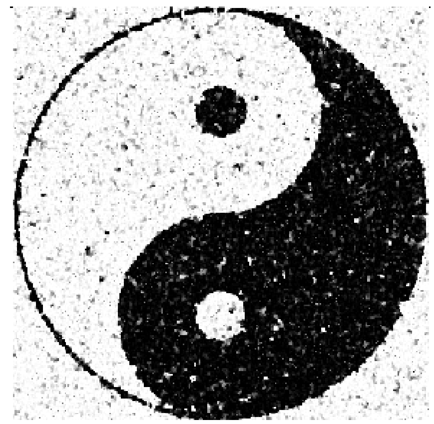
CS6957: Probabilistic Modeling **Prateep Mukherjee(u0876583)**

Homework 3

1

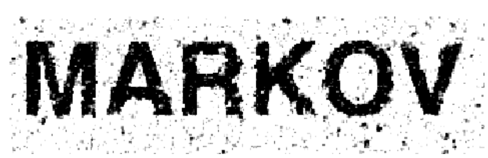
NOTE: Please see program p1a.r for part(a) and p1c.r for part(b) and part(c).



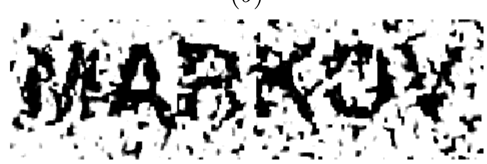
a *b*

Figure 1: Denoised versions of (a) **noisy-message.png** and (b) **noisy-yinyang.png**

Fig. 1 shows the result of Gibbs sampling. For both images, I use 20 *burnin* samples and 100 iterations. For **noisy-message.png**, I used $\alpha = 0.01$, $\beta = 0.85$, $\sigma = 1.5$. For **noisy-yinyang.png**, I used $\alpha = 0.01$, $\beta = 1.0$, $\sigma = 1.0$.



(a) (b)



(c) (d)

Figure 2: Results obtained by changing values of β. (a) 0.05, (b) 0.5, (c) 0.85 and (d) 2.85. All other parameters were fixed as mentioned above.

Next, I tested the effects of various values of β for **noisy-message.png**. The results are shown in Fig. 2. The interpretation of β is that it penalizes pixels which mismatch. Mismatching pixels form an **edge** and so we can infer that β controls the edges in the cleaned image. Higher values of β makes the term $\beta x_i x_j$ term in the energy objective positive, when $x_i \neq x_j$. This penalizes the energy U and makes this step unfavourable. Hence, as we see in Fig 2, increasing β values actually loses the edges in the original image. A very low value for β, on the other hand, hardly removes any noise.(Fig. 2(a)).

(a) (b) (c)

Figure 3: Results obtained by changing values of α (a) 0, (b) 0.05, (c) 2. All other parameters were fixed as mentioned above.

A similar experiment for α gives us the results shown in Fig. 3. As we increase α, more black(=-1) pixels are washed away.

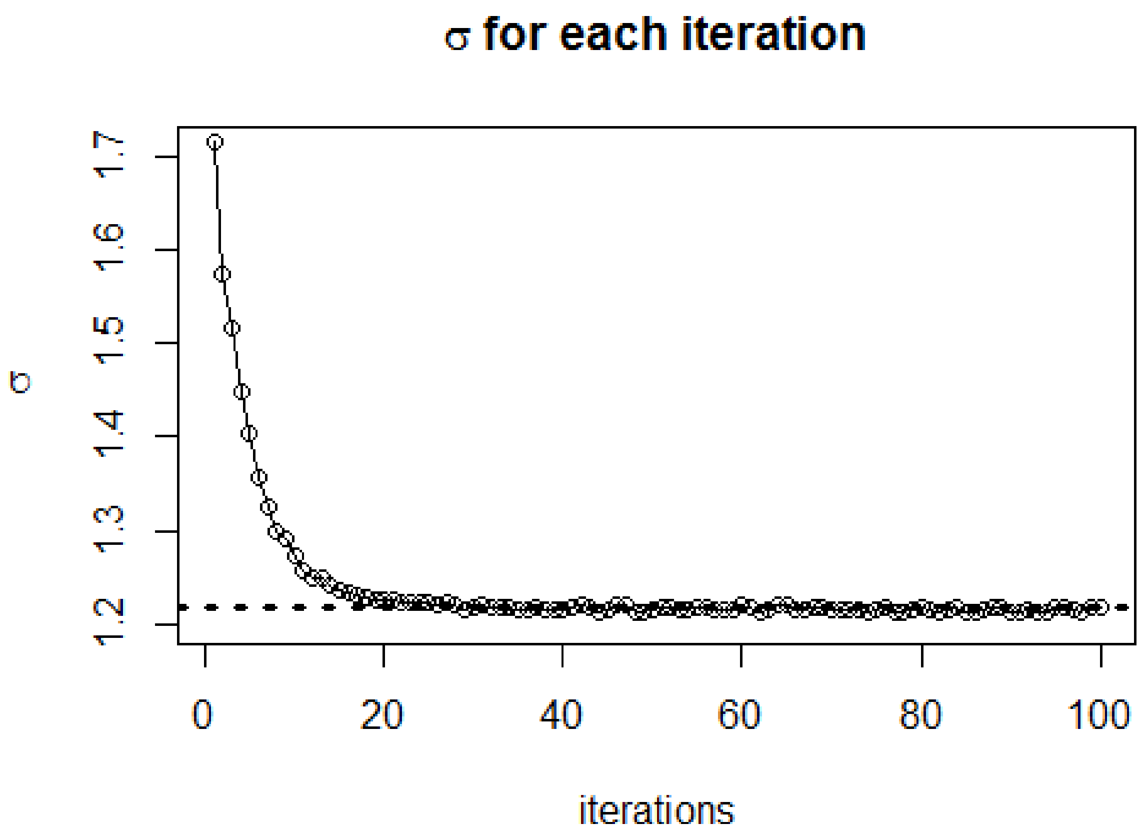


Figure 4: σ values plotted for each iteration for **noisy-message.png**. Black dotted line shows convergence value.

For question 1(c), I changed the value of σ at each iteration using $\sigma_t = \sqrt{\frac{|X_t - Y|^2}{w \times h}}$, where X_t

is the image obtained at the t^{th} iteration and Y is the true image. Fig. 4 shows the plot of σ for the iterations. It converges to a value of ~ 1.2 for **noisy-message.png** and ~ 1.3 for **noisy-yinyang.png**.

2

NOTE: Please see program p2.r for this question.

(a) The formula for un-normalized posterior of $\beta|\mathbf{Y}$ is :

$$p(\beta|\mathbf{Y};\mathbf{X},\sigma) \propto \prod_{i=1}^{n}[(\frac{1}{1+\exp^{-x_i^T\beta}})^{y_i} \cdot (\frac{\exp^{-x_i^T\beta}}{1+\exp^{-x_i^T\beta}})^{(1-y_i)}] \cdot \exp(-\frac{\|\beta\|^2}{2\sigma^2}) \tag{1}$$

(b) Taking logarithm of both sides of Eq. 1, we get

$$\begin{aligned}
log\ \ p(\beta|\mathbf{Y};\mathbf{X},\sigma) &\propto \sum_{i=1}^{n}[-y_i log\ \ (1+\exp^{-x_i^T\beta}) \\
&+ (1-y_i)log(\exp^{-x_i^T\beta}) - (1-y_i)log(1+\exp^{-x_i^T\beta})] - \frac{\|\beta\|^2}{2\sigma^2} \\
&\propto -\sum_{i=1}^{n}[(1-y_i)log(\exp^{-x_i^T\beta}) + log(1+\exp^{-x_i^T\beta})] + \frac{\|\beta\|^2}{2\sigma^2} \\
&\propto -\sum_{i=1}^{n}(1-y_i)x_i^T\beta + log(1+\exp^{-x_i^T\beta}) + \frac{\|\beta\|^2}{2\sigma^2} \\
&= U(\beta)
\end{aligned}$$

Therefore, $p(\beta|\mathbf{Y};\mathbf{X},\sigma) \propto \exp(-U(\beta))$.

(e) Trace plots for β sequence is shown in Fig. 5.

(f) The average error rate computed for 40 test samples is on average 0.025. The values of parameters are $\sigma = 1.0$, $\epsilon = 0.05$, $L = 10$.

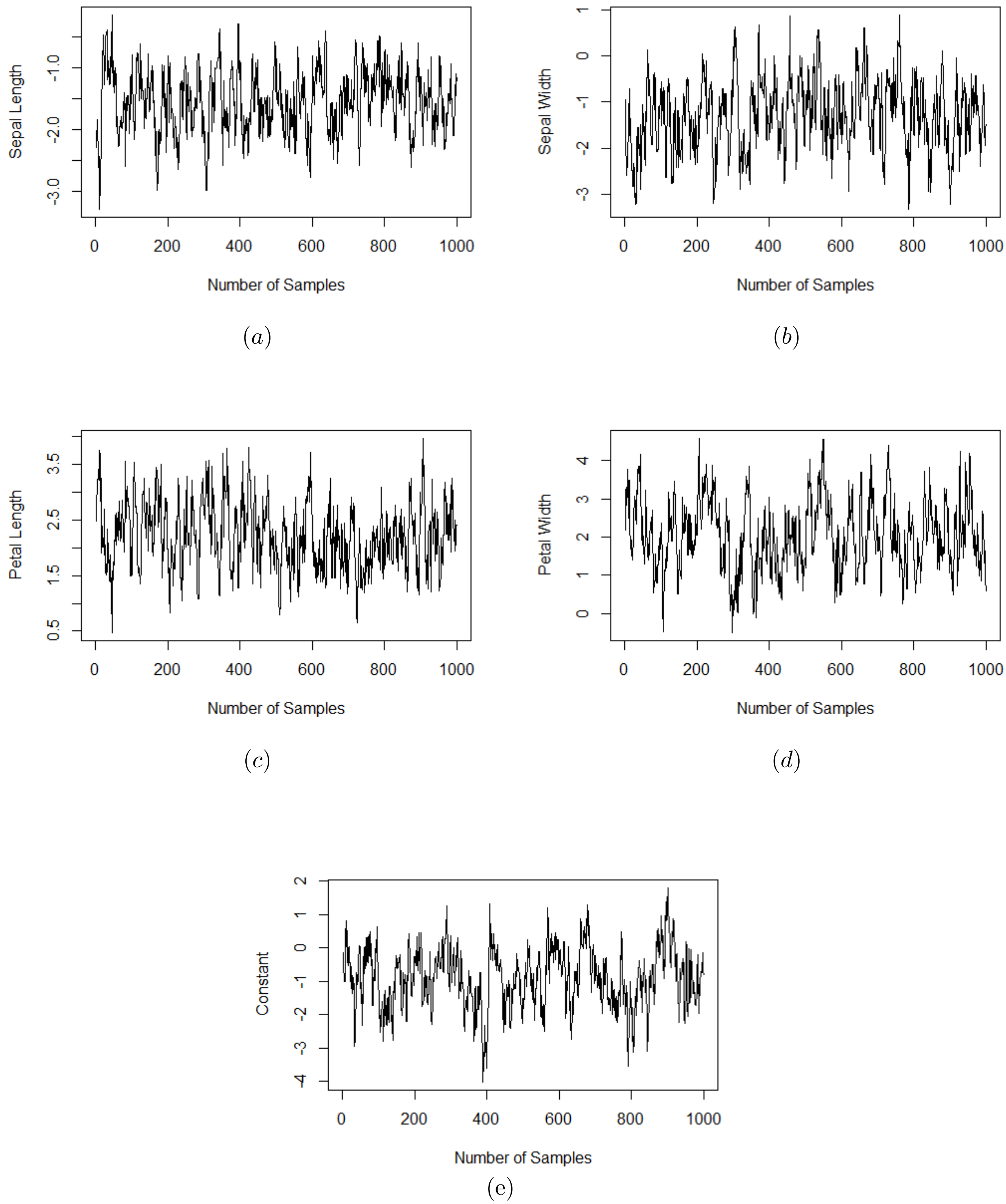


Figure 5: Trace plots for (a) Sepal length, (b) Sepal width, (c) Petal length, (d) Petal width and (e) Constant term.