MPF Objective Function for a Potts Model

Pooneh Mohammadiara

April 26, 2016

This is a document to show the MPF objective function for a Potts model. The MPF objective function is:

$$K(\mathbf{J}) = \sum_{\mathbf{x}\notin\mathcal{D}} \sum_{\mathbf{x}'\in\mathcal{D}} g(\mathbf{x},\mathbf{x}')\exp(\frac{1}{2}[E(\mathbf{x};\mathbf{J}) - E(\mathbf{x}';\mathbf{J})]), \tag{1}$$

Where $g(\mathbf{x},\mathbf{x}') = g(\mathbf{x}',\mathbf{x}) \in \{0,1\}$ **not being considered in the code similar to data and not data** is the connectivity function, $E(\mathbf{x};\mathbf{J})$ is an energy function parameterized by $\mathbf{J}$, and $\mathcal{D}$ is the list of data states. For the Potts model, the energy function is:

$$E(\mathbf{x};\mathbf{J}) = \sum_{ij} \delta_{x_i x_j} J_{ij} \tag{2}$$

where $\mathbf{x} \in \{0, 1, ..., (q-1)\}^N$, $\mathbf{J} \in \mathcal{R}^{N\times N}$ and $\mathbf{J}$ is symmetric ($\mathbf{J} = \mathbf{J}^T$).

We consider the case where the connectivity function $g(\mathbf{x},\mathbf{x}')$ is set to connect all states which differ in a single bit by one unit,

$$g(\mathbf{x},\mathbf{x}') = \begin{cases} 1 & \mathbf{x} \text{ and } \mathbf{x}\text{' differ one in a bit} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The MPF objective function in this case is:

$$K(\mathbf{J}) = \sum_{\mathbf{x}\in\mathcal{D}} \sum_{n=1}^{N} \exp\left(\frac{1}{2}[E(\mathbf{x};\mathbf{J}) - E(\mathbf{x}+\mathbf{d}(\mathbf{x},n);\mathbf{J})]\right), \tag{4}$$

where the sum over n is a sum over all data dimensions, and the bit upgrade function $\mathbf{d}(\mathbf{x},n) \in \{0,1\}^N$ is:

$$\mathbf{d}(\mathbf{x},n)_i = \begin{cases} 0 & \text{i} \neq \text{n} \\ 1 & \text{i} = \text{n} \end{cases} \tag{5}$$

There is one exception to the equation above and that is when x_i equals $q-1$. If we add one to it we get q which is not allowed. Therefore we change that one to zero. This correction is applied the code.

For the Potts model, this MPF objective function becomes:

$$K(\mathbf{J}) = \sum_{\mathbf{x}\in\mathcal{D}} \sum_{n} \exp\left(\frac{1}{2}\left[\sum_{i}\sum_{j}(\delta_{x_i x_j} - \delta_{(x_i+d_i)(x_j+d_j)})J_{ij}\right]\right) \tag{6}$$

To simplify the objective function we break the summations in the exponential argument based on i and j being or not being equal to n. We add them up later to get the total function again.

$$A(\mathbf{J}) = \frac{1}{2}\left[\sum_{i}\sum_{j}(\delta_{x_i x_j} - \delta_{(x_i+d_i)(x_j+d_j)})J_{ij}\right] \tag{7}$$

1. $i \neq n$ and $j \neq n$:

$$A_1(\mathbf{J}) = \frac{1}{2}\left[\sum_{i\neq n}\sum_{j\neq n}(\delta_{x_i x_j} - \delta_{(x_i+d_i)(x_j+d_j)})J_{ij}\right]$$

In this case $d_i = d_j = 0$ based on Eq. (5). Therefore:

$$A_1(\mathbf{J}) = \frac{1}{2}\left[\sum_{i\neq n}\sum_{j\neq n}(\delta_{x_i x_j} - \delta_{x_i x_j})J_{ij}\right] = 0 \tag{8}$$

2. $i = j = n$:

$$A_2(\mathbf{J}) = \frac{1}{2}\left[\sum_{i=n}\sum_{j=n}(\delta_{x_i x_j} - \delta_{(x_i+d_i)(x_j+d_j)})J_{ij}\right]$$

In this case $d_i = d_j = 1$ based on Eq. (5). Therefore:

$$A_2(\mathbf{J}) = \frac{1}{2}\left[(\delta_{x_n x_n} - \delta_{(x_n+1)(x_n+1)})J_{nn}\right] = 0 \tag{9}$$

3. $i = n$, $j \neq n$:

$$A_3(\mathbf{J}) = \frac{1}{2}\left[\sum_{i=n}\sum_{j\neq n}(\delta_{x_i x_j} - \delta_{(x_i+d_i)(x_j+d_j)})J_{ij}\right]$$

In this case $d_i = 1$ and $d_j = 0$ based on Eq. (5). Therefore:

$$A_3(\mathbf{J}) = \frac{1}{2}\left[\sum_{j\neq n}(\delta_{x_n x_j} - \delta_{(x_n+1)x_j})J_{nj}\right] \tag{10}$$

4. $i \neq n$, $j = n$:

$$A_4(\mathbf{J}) = \frac{1}{2}\left[\sum_{i\neq n}\sum_{j=n}(\delta_{x_i x_j} - \delta_{(x_i+d_i)(x_j+d_j)})J_{ij}\right]$$

In this case $d_i = 1$ and $d_j = 0$ based on Eq. (5). Therefore:

$$A_4(\mathbf{J}) = \frac{1}{2}\left[\sum_{i\neq n}(\delta_{x_i x_n} - \delta_{x_i(x_n+1)})J_{in}\right] \tag{11}$$

Adding the 4 parts above we will get:

$$\begin{aligned} A(\mathbf{J}) &= A_1(\mathbf{J}) + A_2(\mathbf{J}) + A_3(\mathbf{J}) + A_4(\mathbf{J}) \qquad (12)\\ &= \frac{1}{2}\left[\sum_{j\neq n}(\delta_{x_n x_j} - \delta_{(x_n+1)x_j})J_{nj}\right] + \frac{1}{2}\left[\sum_{i\neq n}(\delta_{x_i x_n} - \delta_{x_i(x_n+1)})J_{in}\right] \end{aligned}$$

The two terms in Eq. (12) are the same summation if we rename the j to i in the first one:

$$A(\mathbf{J}) = \sum_{i\neq n}(\delta_{x_i x_n} - \delta_{x_i(x_n+1)})J_{in} \tag{13}$$

To include n in the summation, we add and subtract the following term to the Eq. (13):

$$\mathbf{J}_{nn} = (\delta_{x_n x_n} - \delta_{x_n(x_n+1)})J_{nn} \tag{14}$$

$$
\begin{aligned}
A(\mathbf{J}) &= \sum_{i \neq n} (\delta_{x_i x_n} - \delta_{x_i (x_n+1)}) J_{in} + (\delta_{x_n x_n} - \delta_{x_n (x_n+1)}) J_{nn} - \mathbf{J}_{nn} \qquad (15) \\
&= \sum_{i} (\delta_{x_i x_n} - \delta_{x_i (x_n+1)}) J_{in} - \mathbf{J}_{nn}
\end{aligned}
$$

We finally substitute the above into the objective function Eq. (6):

$$
K(\mathbf{J}) = \sum_{\mathbf{x} \in \mathcal{D}} \sum_{n} \exp \left(\sum_{i} (\delta_{x_i x_n} - \delta_{x_i (x_n+1)}) J_{in} - \mathbf{J}_{nn} \right) \qquad (16)
$$

That final one is the one used in the code.