C Additional Examples

We used the same procedure as before for these examples. We first set a true weight matrix W_* and then generated M data samples $\{x_i\}_{i=1}^M$ from a density $p_{\sigma|W_*}$. Again, the chains were run for 1000 steps and the initial configurations were drawn uniformly at random from all the possible configurations.

C.1 CD for Boltzmann machine

Different CD algorithms were tested for the fully visible BM with 3 nodes specified in section 4.7.1. The number of observations was set to $M = 2000$. We examine how fast the time averages of the CD-1 algorithm, the persistent CD-1 algorithm and the CD-5 algorithm converge to the true parameter.

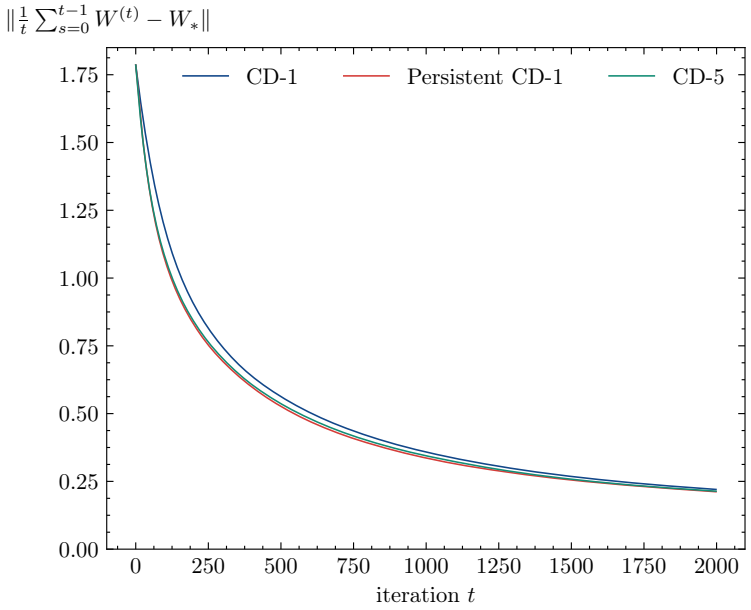


Figure 7: The distance between the time average $\frac{1}{t}\sum_{s=0}^{t-1} W^{(t)}$ and the true parameter W_* for $0 < t \leq 2000$.

For this example with only 3 units, the different CD algorithms produce similar estimates.

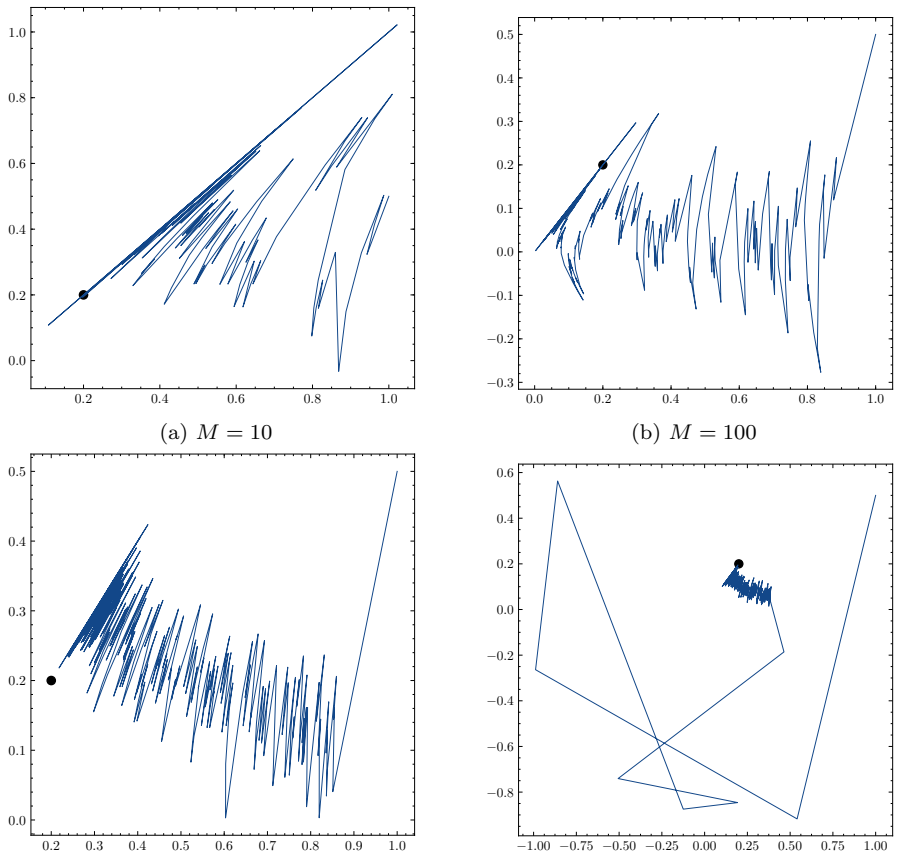
C.2 CD for Bernoulli-Bernoulli RBM

A second numerical simulation was run for same RBM as in section 4.7.3. All of the conditions are the same as in the first approach and the data was generated in the same way. Only in this case the CD algorithm was applied for the following representation of the gradient of the $\mathcal{LL}$:

$$\nabla\mathcal{LL}(W) = -\mathbb{E}_{\hat{\mathbb{P}}_M}[\nabla\mathcal{F}_W] + \mathbb{E}_{p_{\sigma|W}}[\nabla\mathcal{F}_W]\,.$$

For this model in particular, we have

$$\begin{aligned}&\frac{\partial}{\partial w_{kl}}\mathcal{LL}(W)\\ &= \mathbb{E}_{\hat{\mathbb{P}}_M}[\tanh(\beta(\sigma, w_{*,l}))\beta\sigma_k] - \mathbb{E}_{p_{\sigma|W}}[\tanh(\beta(\sigma, w_{*,l}))\beta\sigma_k]\,.\end{aligned}$$



(a) $M = 10$ (b) $M = 100$

(c) $M = 1000$ (d) $M = 10000$

Figure 8: The sequence $\{W^{(t)}\}_{t=1}^{3000}$ generated by the CD-2 algorithm for RBM. The chain was initialized at $W_0 = (1, 0.5)$. The black points are the true parameter W_*. In Figure (a) $M = 10^1$ data points were used, in (b) $M = 10^2$, (c) $M = 10^3$ and in (d) $M = 10^4$.

This method was computationally more expensive and needed more iterations until the chains reached the random walk neighborhood. Also, it is demanding to set a sensible learning rate. Here, we set $\eta = 0.3M^{\frac{1}{3}}$. Also in this case it is possible that the chains move to a neighborhood close to $-W_*$ if they are initialized at a different W_0.

The conditional density of the node σ_1 given σ_2, which is used for the Gibbs sampling, is of the form

$$p_{\sigma_1|\sigma_2,W}(\sigma_1 = 1; \sigma_2) = \frac{\cosh(\beta w_{11} + \beta\sigma_2 w_{21})}{\cosh(\beta w_{11} + \beta\sigma_2 w_{21}) + \cosh(-\beta w_{11} + \beta\sigma_2 w_{21})}.$$

The conditional density of σ_2 given σ_1 can be derived accordingly.

From lemma 6 we know that $\|\hat{W}_M - W_*\| < M^{\frac{-\gamma}{2}}$ with probability approaching 1 as $M \to \infty$. As $-\frac{\gamma}{2} < -\frac{\gamma}{3}$, it follows that

$$\begin{aligned}\limsup_{t\to\infty} \left\| \frac{1}{t}\sum_{s=0}^{t-1} W^{(s)} - W_* \right\| &\leq \limsup_{t\to\infty} \left\| \frac{1}{t}\sum_{s=0}^{t-1} W^{(s)} - \hat{W}_M \right\| + \|\hat{W}_M - W_*\| \\ &= \mathcal{O}\left(M^{-\frac{\gamma}{3}}\right), \qquad \mathbb{P}^{\mathbf{x}}\text{-a.s.}\end{aligned}$$

Therefore, there exists $A_k \in \mathbb{R}_+$ such that

$$\lim_{M\to\infty} \mathbb{P}\left(\limsup_{t\to\infty} \left\| \frac{1}{t}\sum_{s=0}^{t-1} W^{(s)} - W_* \right\| > A_k M^{-\frac{\gamma}{3}} \right) = 0.$$

□

Hence, we find a k such that any limit point of the time average $\frac{1}{t}\sum_{s=0}^{t-1} W_s$ is a consistent estimate for W_*.

4.7 Examples: CD algorithm

In all of the following examples we first chose a true weight matrix W_* and then generated M data samples $\{x_i\}_{i=1}^M$ from a density $p_{\sigma|W_*}$. For this M initial configurations were drawn uniformly at random from all the possible configurations and then M chains were run for 1000 steps from these initial values. The data $\{x_i\}_{i=1}^M$ was set to be the obtained samples after 1000 steps.

Then, we examined if the learning algorithms are able to recover the true weight matrix W_* from the data $\{x_i\}_{i=1}^M$.

4.7.1 Fully visible Boltzmann machine with 3 units

In this example CD-1 algorithm was implemented for the fully visible Boltzmann machine with $N = 3$ units to illustrate the theoretical findings from the previous chapter. The priors are chosen to be binary random variables. Therefore, the sample space is given by $\Sigma = \{-1, 1\}^3$.

For such a BM, the value of w_{ii} does not influence the probability distribution $p_{\sigma|W}$. This can be seen by

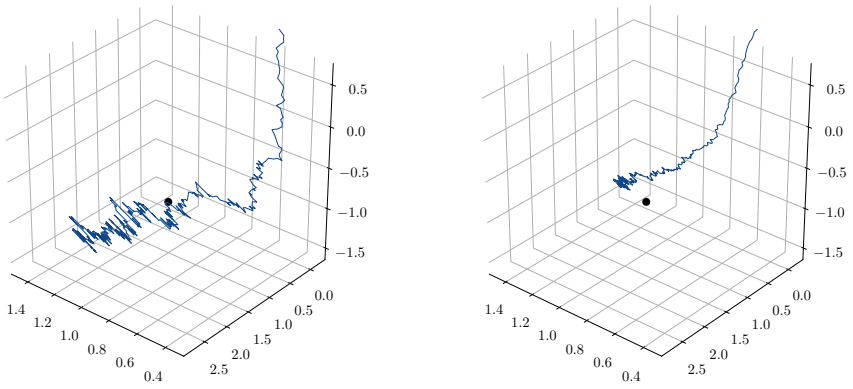
$$\begin{aligned}p_{\sigma|W}(\sigma; W) &= \frac{\exp(\beta(\sigma, W\sigma))}{\sum_{\sigma\in\{-1,1\}^3} \exp(\beta(\sigma, W\sigma))} \\ = \frac{\exp(\beta w_{ii}) \exp(\beta \sum_k \sum_{l\neq i} \sigma_k w_{kl} \sigma_l)}{\exp(\beta w_{ii}) \sum_{\sigma\in\{-1,1\}^3} \exp(\beta \sum_k \sum_{l\neq i} \sigma_k w_{kl} \sigma_l)} &= \frac{\exp(\beta \sum_k \sum_{l\neq i} \sigma_k w_{kl} \sigma_l)}{\sum_{\sigma\in\{-1,1\}^3} \exp(\beta \sum_k \sum_{l\neq i} \sigma_k w_{kl} \sigma_l)}\end{aligned}$$

Therefore, the diagonal entries of the weight matrix are determined to be 0. Let the true weight matrix W_* be given by

$$W_* = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & -1 \\ 1 & -1 & 0 \end{pmatrix}$$

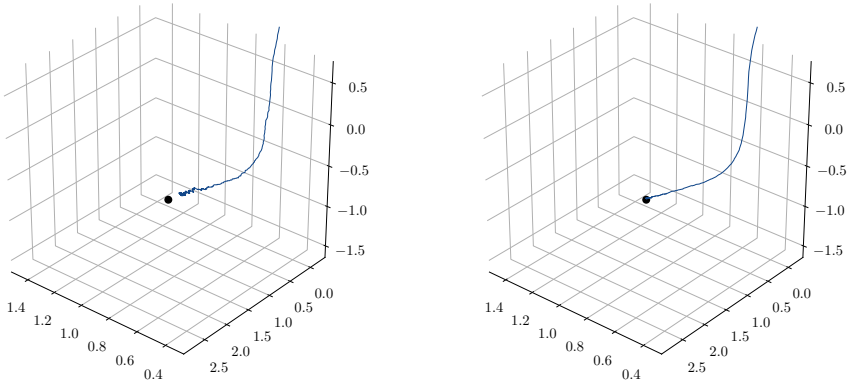
A random-scan Gibbs sampling method was used to obtain the MCMC samples. We obtain the following conditional densities for this model:

$$\begin{aligned} p(\sigma_i = 1 \mid \sigma_{-i}) &= \frac{\exp(2\beta \sum_j w_{ij}\sigma_j)}{\exp(2\beta \sum_j w_{ij}\sigma_j) + \exp(-2\beta \sum_j w_{ij}\sigma_j)} \\ &= \frac{1}{2}(1 - \tanh(-2\beta \sum_j w_{ij}\sigma_j)). \end{aligned}$$



(a) $M = 10$

(b) $M = 100$



(c) $M = 1000$

(d) $M = 10000$

Figure 4: The sequence $\{(w_{12}^{(t)}, w_{13}^{(t)}, w_{23}^{(t)})\}_{t=1}^{1000}$ generated by the CD-1 algorithm for BM with 3 nodes in blue. The black points are the true parameter W_*. In Figure (a) $M = 10^1$ data points were used, in (b) $M = 10^2$, (c) $M = 10^3$ and in (d) $M = 10^4$.

Each algorithm starts with the fixed initial weight matrix

$$W^{(0)} = \begin{pmatrix} 0 & 3/4 & -3/4 \\ 3/4 & 0 & 3/4 \\ -3/4 & 3/4 & 0 \end{pmatrix}$$

The learning rate is chosen to be $\eta = 0.2$ and β was set $\frac{1}{3}$.

We will focus on the CD-1 algorithm for this model but vary the number of observations. Since the matrix $W^{(t)}$ is symmetric and all the diagonal entries are 0, it is enough to look at $(w_{12}^{(t)}, w_{13}^{(t)}, w_{23}^{(t)})$. Figure 4 illustrates the behavior of $\{(w_{12}^{(t)}, w_{13}^{(t)}, w_{23}^{(t)})\}_{t=1}^{1000}$ for different M.

At the beginning all the chains move quickly to a neighborhood which is close to the true parameter and in which the chain "walks" around randomly. For larger M these neighborhoods are getting smaller and are closer to the true parameter.

In the appendix C there is a second example, where the CD-1, CD-5 and the persistent CD were tested for this model.

4.7.2 Fully visible Boltzmann machine with 10 units

This example illustrates the influence of different values for β on the convergence of CD-algorithms for a binary BM with 10 units and for $M = 5000$ data points. At each iteration a mini-batch sampling was applied. This means a random subset of 500 data points was selected at each iteration of the algorithm.

The true weight matrix W is given by the matrix that has 0 on the diagonal and 1 as off-diagonal entries. This BM corresponds to the CW model with $\beta_{BM} = \frac{1}{2N}\beta_{CW}$ (section 2.6).

The estimates $\{W^{(t)}\}_{t\geq 0}$ were generated with the CD-2 or the persistent CD-2 algorithm. These algorithms were tested for $\beta \in \{\frac{0.5}{2N}, \frac{1.1}{2N}, \frac{2.3}{2N}\}$. The initial estimate $W^{(0)}$ was given by the matrix that has 0 on the diagonal and alternating $-0.75, 0.75$ as off-diagonal entries.

We observe that if $\beta = \frac{1.1}{2N}$, learning is possible. For $\beta = \frac{0.5}{2N}$ the chain only moves slowly to the true parameter. Here, the issue is that the resulting Gibbs distribution is close to the uniform distribution and therefore W does not have a strong influence on the distribution. Thus, more data would be necessary for a better parameter learning.

If $\beta = \frac{2.3}{2N}$, then it would be reasonable to chose a different k for the CD-k algorithm. We saw that for the Curie-Weiss model the mixing of the Gibbs chain occurs fast if $\beta_{CW} < 1$. For $\beta_{CW} > 1$, we know that the mixing takes longer than some upper bound that grows exponentially in N. Hence, in this case assumption (A2) is not satisfied.

This seems to be a general issue as we always need β to be small for the mixing and β large for the convergence to the MLE.

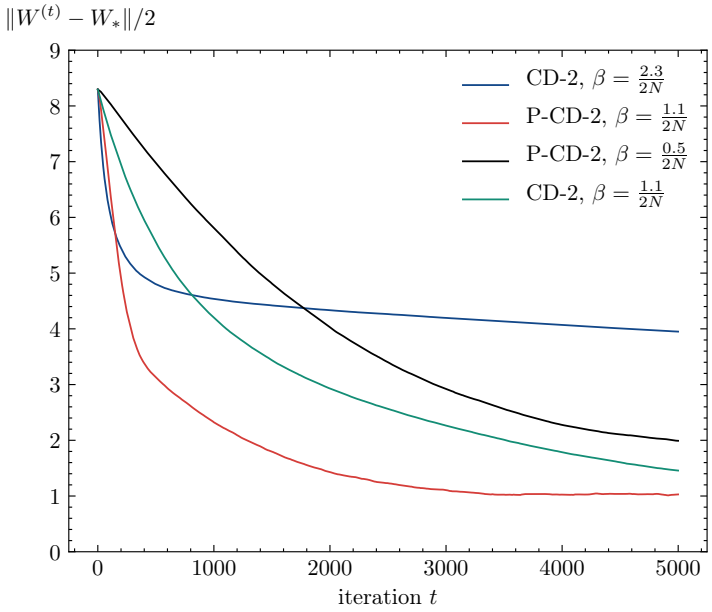


Figure 5: The distance between the estimate $W^{(t)}$ for different CD algorithms and the true parameter W_* for $0 < t \leq 5000$.

4.7.3 Bernoulli-Bernoulli RBM

In comparison, a similar numerical experiment as in section 4.7.1 for BM was performed for a Bernoulli-Bernoulli RBM with 2 visible units and 1 hidden unit. The true weight matrix was set to be

$$W_* = \begin{pmatrix} 0.2 \\ 0.2 \end{pmatrix}$$

For the first approach, the following representation of the gradient of the $\mathcal{LL}$ was used

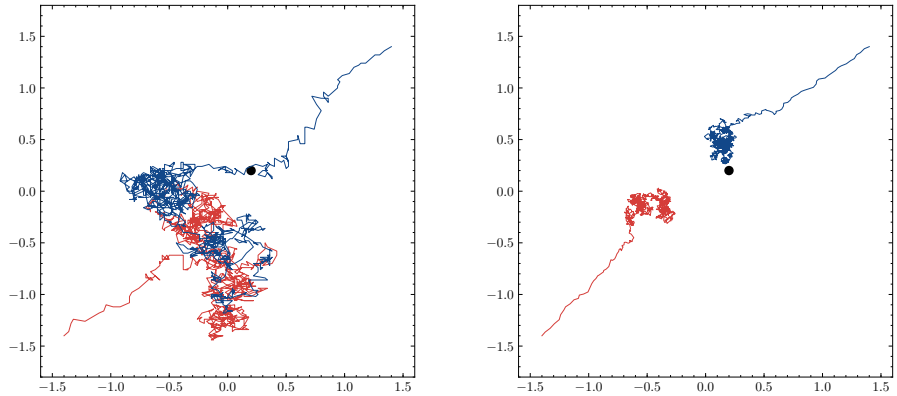
$$\frac{\partial}{\partial w_{kl}} \mathcal{LL}(W) = \beta \mathbb{E}_{\hat{\mathbb{P}}_M}[\mathbb{E}_{p_{s|\sigma,W}}[\sigma_k s_l]] - \beta \mathbb{E}_{p_{\sigma,s|W}}[\sigma_k s_l]\,.$$

Two chains $\{W_i^{(t)}\}_{t=1}^{1000}$ with $i = 1, 2$ were initialized at two different weight matrices $W_i^{(0)}$ with $i = 1, 2$. More explicitly, we set $W_1^{(0)} = (1.4, 1.4)$ and $W_2^{(0)} = (-1.4, -1.4)$. The chain, which was initialized at $(-1.4, -1.4)$, moves a neighborhoods that is close to $-W_*$.

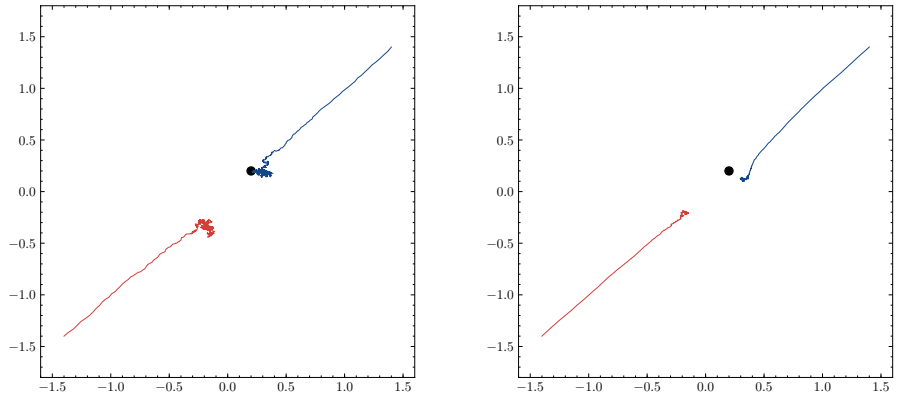
For this model the marginal probability distribution over the visible layer is given by

$$p_{\sigma|W}(\sigma; W) = \frac{\cosh(\beta(\sigma, W))}{Z(W)} p_\sigma(\sigma).$$

Because cosh is symmetric, $p_{\sigma|W}$ represents the same probability distribution as $p_{\sigma|-W}$. For this reason, the chain started at $W_0 = (-1.4, -1.4)$ approach neighborhoods close to $-W_*$ instead of W_*. In contrast to the previous examples, the MLE is not unique anymore.



(a) $M = 10$ (b) $M = 100$



(c) $M = 1000$ (d) $M = 10000$

Figure 6: Two sequences $\{W_i^{(t)}\}_{t=1}^{1000}$ with $i = 1, 2$ generated by the CD-5 algorithm for RBM with $\beta = 1$. The red chains are initialized at $W_2^{(0)} = (-1.4, -1.4)$ and the blue at $W_2^{(0)} = (1.4, 1.4)$. The black points are the true parameter W_*. In Figure (a) $M = 10^1$ data points were used, in (b) $M = 10^2$, (c) $M = 10^3$ and in (d) $M = 10^4$.

It seems that in order to obtain a reasonable estimate by the time average of the parameters, a RMB requires more data points than a BM. The advantage of RBM is the block Gibbs sampling. But with so few units this gets irrelevant.

There is a second example for this model in the appendix C, where the CD algorithm was implemented for the other representation of the gradient of $\mathcal{LL}$:

$$\nabla \mathcal{LL}(W) = -\mathbb{E}_{\hat{\mathbb{P}}_M}[\nabla \mathcal{F}_W] + \mathbb{E}_{p_{\sigma|W}}[\nabla \mathcal{F}_W].$$