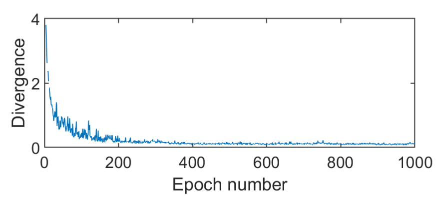
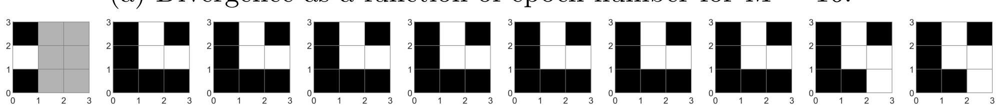
Restricted Boltzmann machine

David Tonderski (davton)

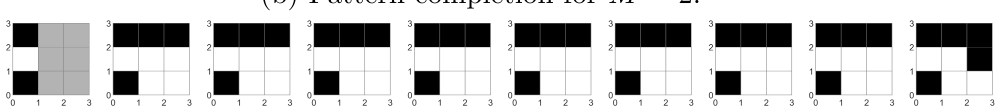
The divergence was calculated as follows: in each epoch, each *possible* pattern was fed to the Boltzmann machine. The dynamics were then run for 100 iterations, and the frequencies at which the different *possible* patterns occured were counted. This was then used as an approximation for the model distribution P_B for each epoch, and the Kullback-Leibler divergence was calculated using $d = \sum_\mu P_D(\mu) \log(P_D(\mu)/P_B(i_\mu))$, where $P_D(\mu) = \frac{1}{14}$ for all data set patterns μ, and i_μ is the index of the data set pattern μ in the set of all possible patterns. If $P_B(\mu) = 0$, I set $d = \infty$. For $M = 2, 4, 8$, the divergence was infinity for (almost) all epochs, so the plots are not shown. For $M = 16$, the divergence is shown in figure 1a. The first 10 produced patterns after feeding the first column are shown in figures 1b, 1c, 1d, 1e.



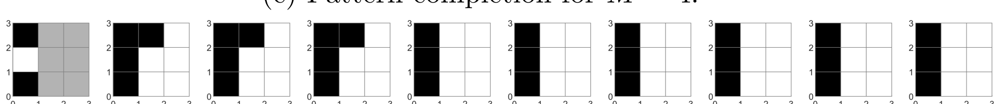
(a) Divergence as a function of epoch number for $M = 16$.



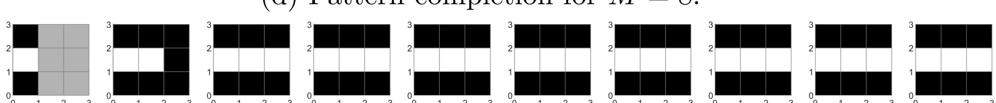
(b) Pattern completion for $M = 2$.



(c) Pattern completion for $M = 4$.



(d) Pattern completion for $M = 8$.



(e) Pattern completion for $M = 16$.

Figure 1: Training information for the two networks.

As expected, the model works well for $M = 16$, and not at all for $M = 2, 4$. Interestingly, the model converges to a data set pattern for $M = 8$, despite the divergence being ∞, but it is not the correct one. This probably means that the model has learned a few of the data set patterns, but not all of them.