Training Restricted Boltzmann Machines using Approximations to the Likelihood Gradient

Tijmen Tieleman
University of Toronto

A problem with MRFs

- Markov Random Fields for unsupervised learning (data density modeling).
- Intractable in general.
- Popular workarounds:
  - Very restricted connectivity.
  - Inaccurate gradient approximators.
  - Decide that MRFs are scary, and avoid them.
- This paper: there is a simple solution.

Details of the problem

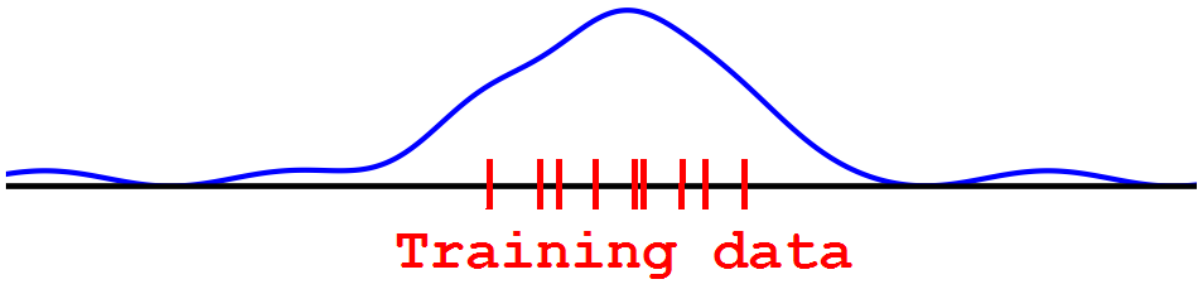
- MRFs are unnormalized.
- For model balancing, we need samples.
  - In places where the model assigns too much probability, compared to the data, we need to reduce probability.
  - The difficult thing is to find those places: exact sampling from MRFs is intractable.
- Exact sampling: MCMC with infinitely many Gibbs transitions.

Approximating algorithms

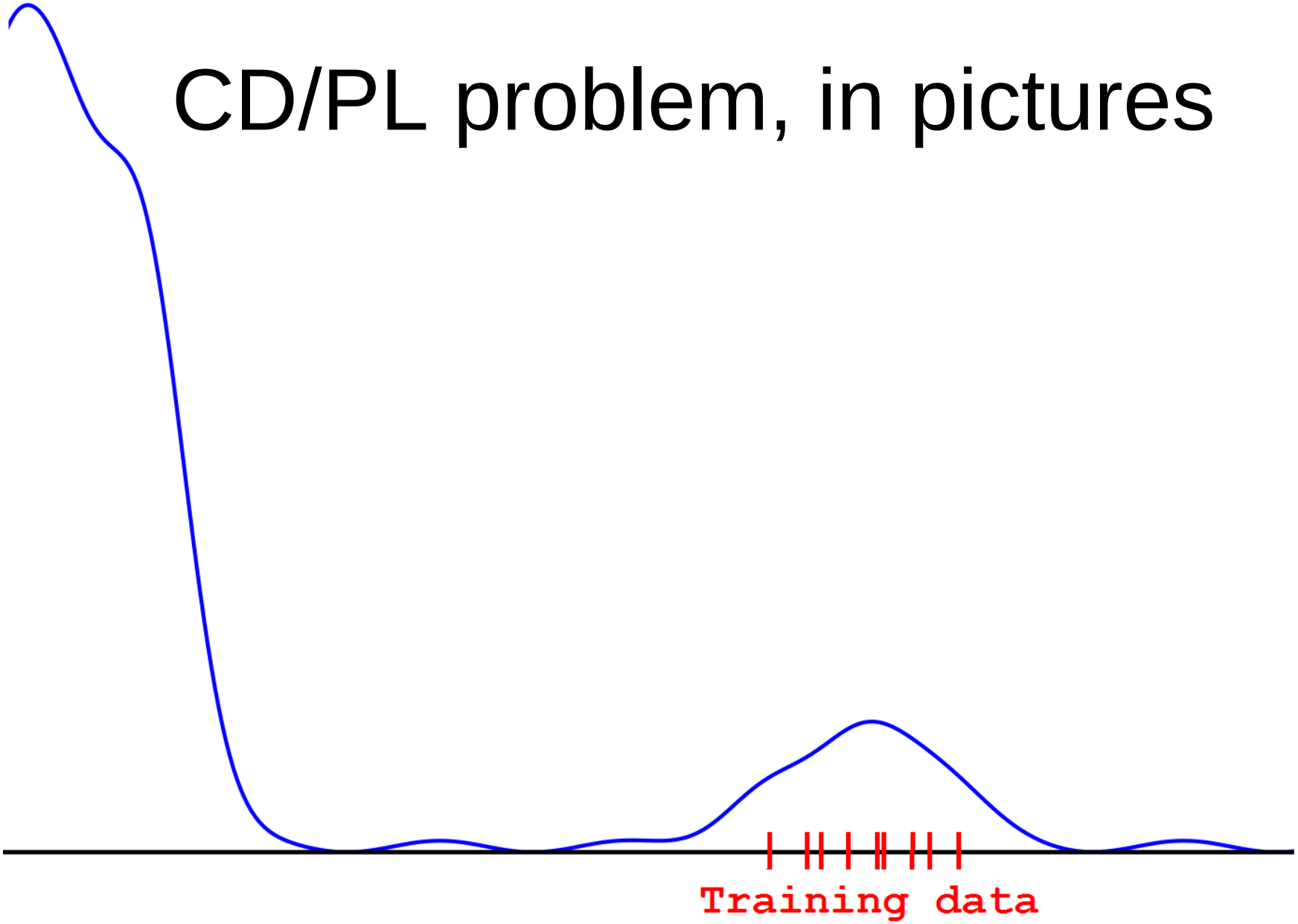
- Contrastive Divergence; Pseudo-Likelihood
- Use surrogate samples, close to the training data.
- Thus, balancing happens only locally.
- Far from the training data, anything can happen.
  - In particular, the model can put much of its probability mass far from the data.

CD/PL problem, in pictures

CD/PL problem, in pictures

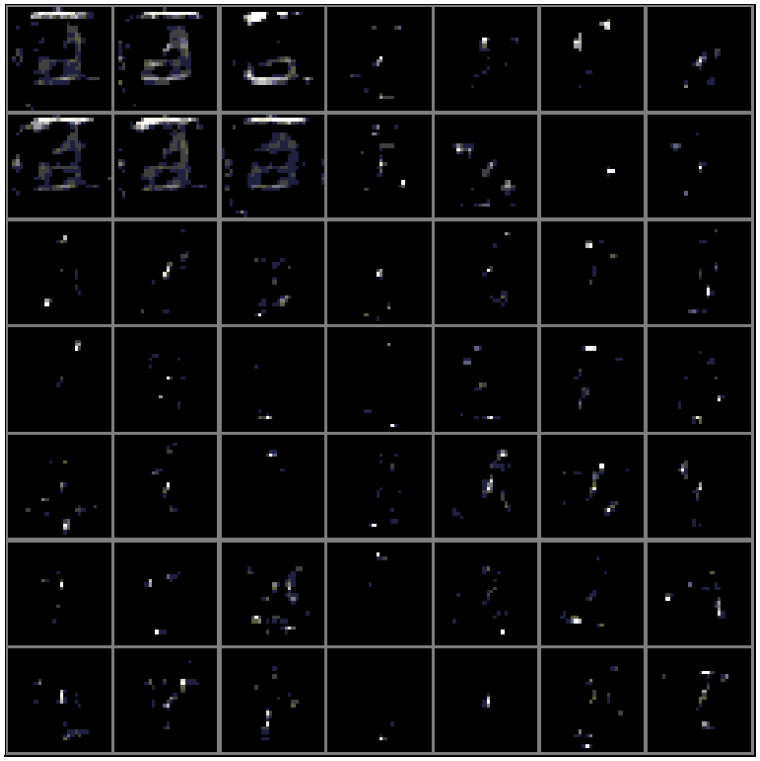


CD/PL problem, in pictures

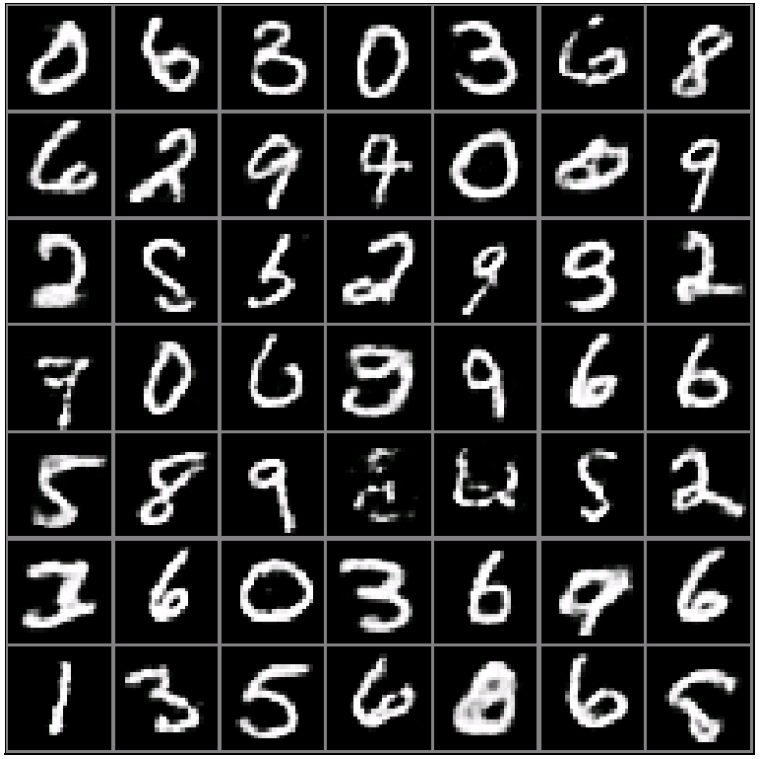


CD/PL problem, in pictures

Samples from an RBM that was trained with CD-1:



Better would be:



Solution

- Gradient descent is iterative.
  - We can reuse data from the previous estimate.
- Use a Markov Chain for getting samples.
- Plan: keep the Markov Chain close to equilibrium.
- Do a few transitions after each weight update.
  - Thus the Chain catches up after the model changes.
- Do not reset the Markov Chain after a weight update (hence ‘Persistent’ CD).
- Thus we always have samples from very close to the model.

More about the Solution

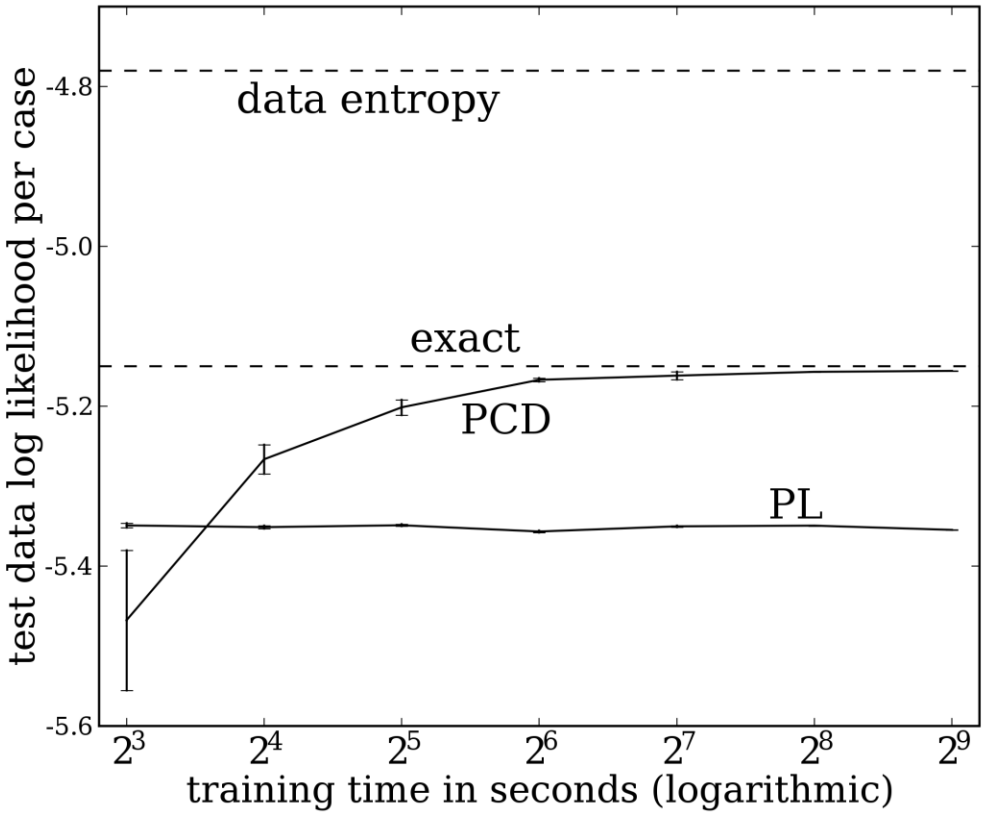
- If we would not change the model at all, we would have exact samples (after burn-in). It would be a regular Markov Chain.
- The model changes slightly,
  - So the Markov Chain is always a little behind.
- Known in statistics as ‘stochastic approximation’.
  - Conditions for convergence have been analyzed.

In practice…

- You use 1 transition per weight update.
- You use several chains (e.g. 100).
- You use smaller learning rate than for CD-1.
- Convert CD-1 program.

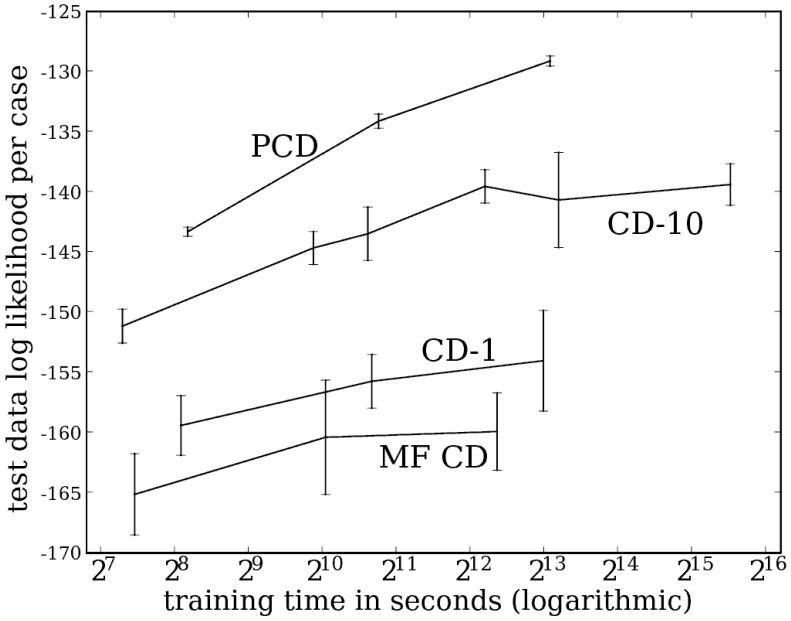
Results on fully visible MRFs

- Data: MNIST 5x5 patches.
- Fully connected.
- No hidden units, so training data is needed only once.

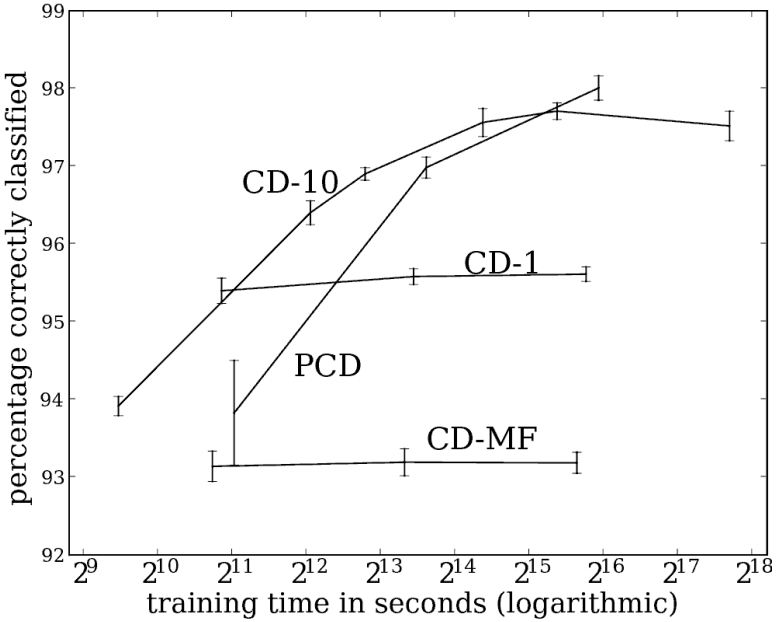


Results on RBMs

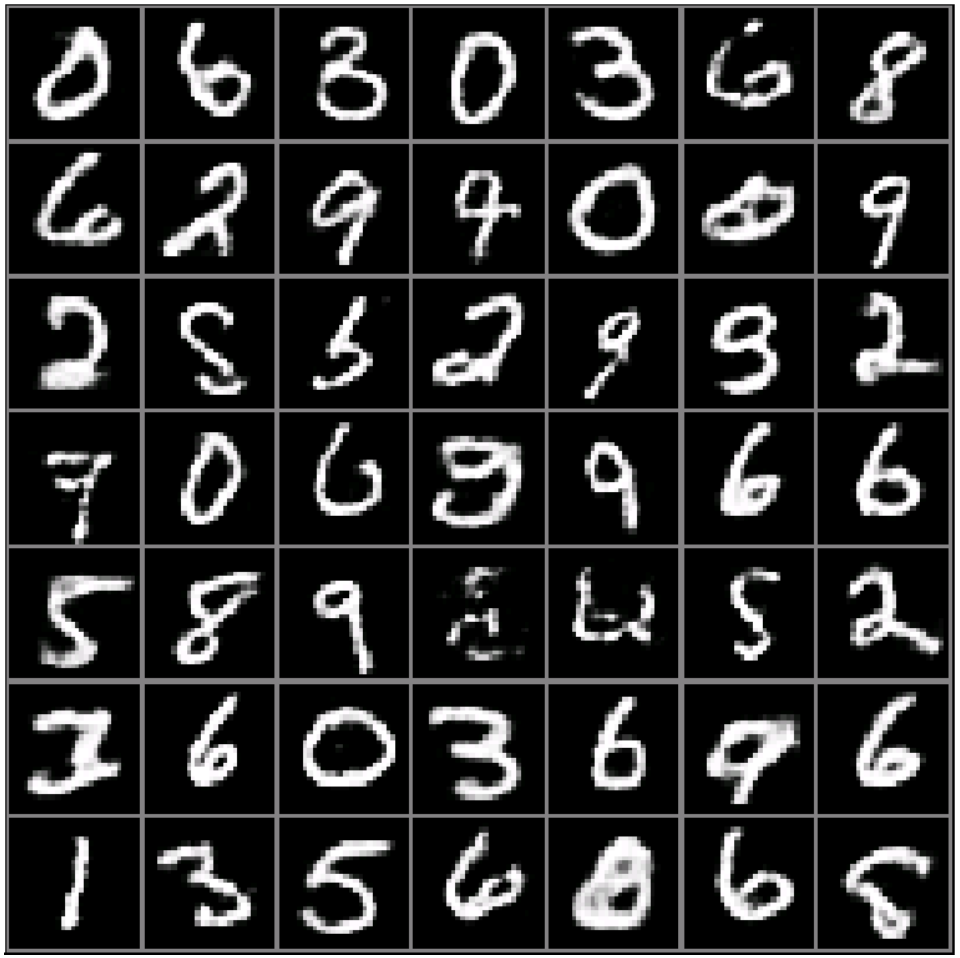
- Data density modeling:



- Classification:



Balancing now works



Conclusion

- Simple algorithm.
- Much closer to likelihood gradient.

The end (question time)

Notes: learning rate

- PCD not always best
  - Little training time
  - (i.e. big data set)
- Variance
- CD-10 occasionally better

Notes: weight decay

- WD helps all CD algorithms, including PCD.
- PCD needs less.
- In fact, zero works fine.