Charlie Tang
Ilya Sutskever
Department of Computer Science,
University of Toronto, Canada

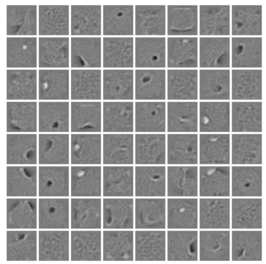
Data Normalization in the Learning of Restricted Boltzmann Machines

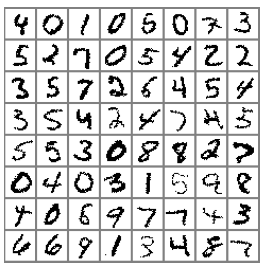
Introduction

- RBM training with CD works well when data is sparse (black background)
- Training is much worse with inverted images
- We present a simple and effective solution to this problem
- New algorithm requires the addition of 3 lines of code
- Data normalization improves RBM training and should *always* be used

Standard MNIST filters

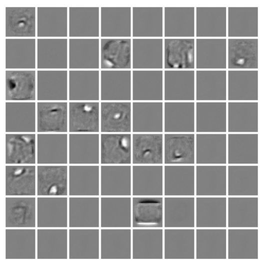
-96 nats





Many "dead" filters

-110 nats



"Zero-mean" solution:

$$z_i = v_i - \frac{1}{N}\sum_n^N v_i^n$$
$$= v_i - \mu_i$$

$$E_{RBM}(\mathbf{z},\mathbf{h};\theta) = -\sum_i^{N_v} b_i(v_i - \mu_i) - \sum_j^{N_h} c_j h_j - \sum_{i,j}^{N_v,N_h} W_{ij}(v_i - \mu_i)h_j$$
$$= -\sum_i b_i v_i - \sum_{i,j} W_{ij} v_i h_j - \sum_j (c_j - \sum_i W_{ij}\mu_i) h_j + const.$$

$$p(z_i = 1 - \mu_i|\mathbf{h}) = \frac{1}{1+\exp(-\phi_i)}$$

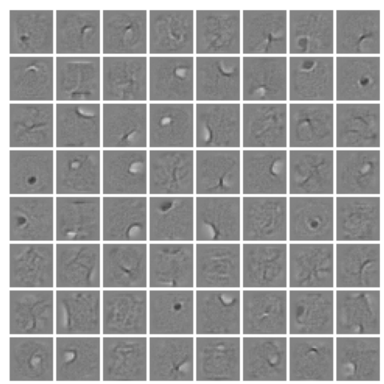
$$p(z_i = 1 - \mu_i|\mathbf{h}) \equiv p(v_i = 1|\mathbf{h})$$

Algorithm

Algorithm 1 Contrastive Divergence Training of RBM on zero-meaned data.

0: Randomly initialize the weights and biases to be small.
1: **Subtract data mean from all training data vectors: $\mathbf{z} = \mathbf{v} - \boldsymbol{\mu}$**
  for $t = 1 : NumberEpochs$ **do**
    for $n = 1 : NumberDataSamples$ **do**
      Positive Phase:
2:       Compute hidden activations and sample using Eq. 6.
3:       Calculate the MLE gradient of the positive phase.
      Negative Phase:
4:       Compute reconstructions $\hat{v}_i$ and sample using Eq. 8.
5:       **Compute reconstructions $\hat{\mathbf{z}}$: $\hat{\mathbf{z}} = \hat{\mathbf{v}} - \boldsymbol{\mu}$**
6:       Compute hidden activations and sample using Eq. 6.
7:       Calculate the MLE gradient of the negative phase.
8:       Approx. gradient = positive phase gradient − negative phase gradient.
9:       Update the parameters.
    end for
  end for
10: **Modify hidden biases: $\mathbf{c}_{new} \leftarrow \mathbf{c} - \boldsymbol{\mu}^T\mathbf{W}$**

Filters learned using zero-mean



AIS estimation of log-prob in nats: test (training)

Dataset	FPCD-30	ZM-30	FPCD-1000	ZM-1000
MNIST	-96 (-96)	-94 (-94)	-84(-81)	-84 (-81)
Neg. MNIST	-110(-110)	-96 (-96)	-87(-85)	-84 (-81)

Conclusions

- Extremely simple way to improve training
- Zero-mean works well with CD, PCD, and FPCD
- Leads to models with higher log-probs
- Learns sparser features, which are better for classification

Tech Report: www.cs.toronto.edu/~tang