Deep Learning III: Regularization

CSci 5525: Machine Learning

Instructor: Nicholas Johnson

October 22, 2020

Announcements

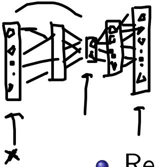
- HW3 will be posted on Tue, Oct 27 (due Nov 10)
- No QA session or office hours today (Oct 22)

Training Deep Networks

- Data augmentation
  - Create a larger dataset for training
  - Large patches, translation, rotations, noise
- Unsupervised pre-training
  - Pre-train parameters using unlabeled data
  - Initialize supervised training from pre-trained parameters
- Dropout
  - Structural changes in deep networks while training
  - Robust representation learning, better performance

Pre-training

- Unsupervised model for pre-training, without labels
- Noisy auto-encoders
  - Reconstruct input $\mathbf{x}$ from noisy version $C(\mathbf{x})$
  - For suitable activation (e.g., sigmoid) h, reconstruct



$$\hat{\mathbf{x}} = \text{sigmoid}(\mathbf{c} + W^\top h(C(\mathbf{x})))$$

$\ell(x, \hat{x}) = \|x - \hat{x}\|_2$

  - Find $W, \mathbf{c}$ to minimize reconstruction error
- Restricted Boltzman Machines (RBM) (000, not 101)
  - Observed and hidden units: $\mathbf{x}$ and $\mathbf{h}$
  - Probability distribution over the units $P(\mathbf{x}, \mathbf{h})$
  - Conditional distributions factorize

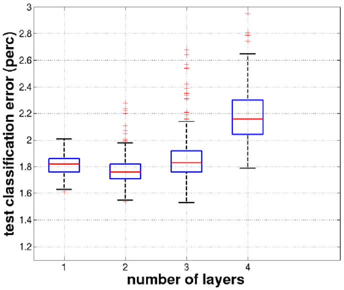
$$P(\mathbf{h}|\mathbf{x}) = \prod_i p(h_i|\mathbf{x}) \qquad p(\mathbf{x}|\mathbf{h}) = \prod_j p(x_j|\mathbf{h})$$

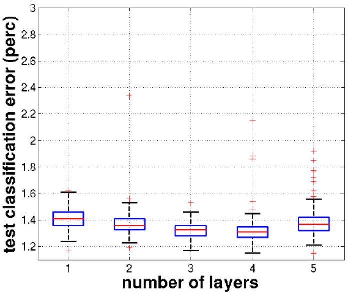
  - Sufficient statistics: $h_i, x_j, h_i x_j$

$$P(\mathbf{x}, \mathbf{h}) \propto \exp\left\{\mathbf{h}^\top W \mathbf{x} + \mathbf{b}^\top \mathbf{x} + \mathbf{c}^\top \mathbf{h}\right\}$$

  - Estimate parameters $(W, \mathbf{b}, \mathbf{c})$ by maximizing log-likelihood

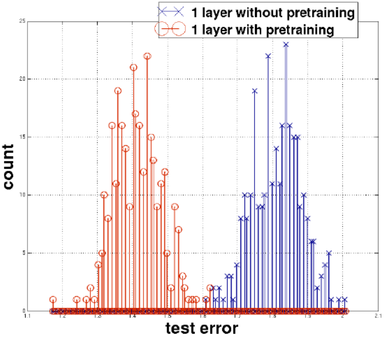
Pre-training: Effect of Depth

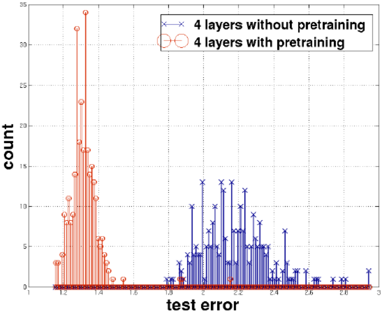




Left: without pre-training, right: with pre-training

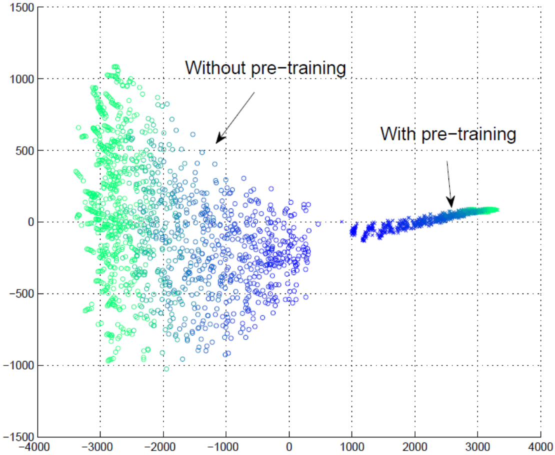
Pre-training: Histogram of Test errors





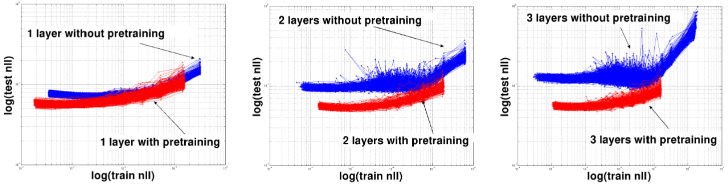
Results from 400 different initializations

Trajectory of learned parameters



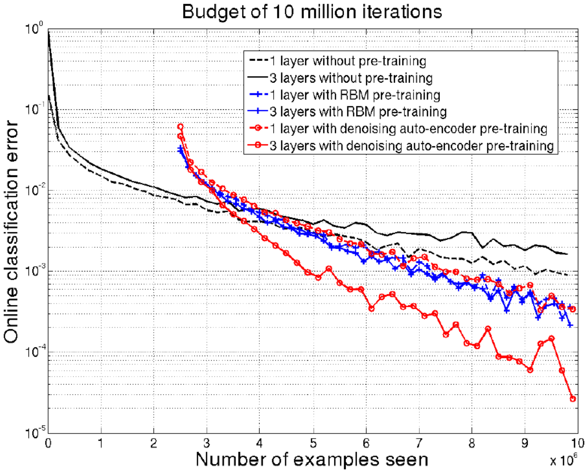
50 networks with and 50 networks without pre-training
Blue to green shows progress over training iterations

Trajectory of Negative Log-likelihood



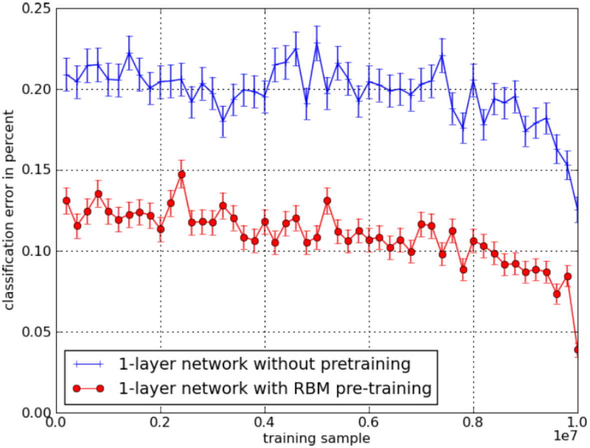
Training proceeds from right to left (NLL decreases in training)
Lower NLL is better
Upward movement towards the end (left) implies overfitting

Online Classification Error



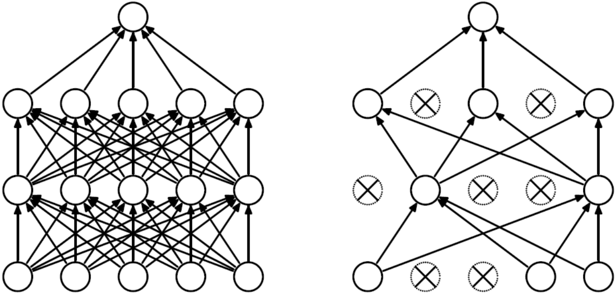
Online errors, over blocks of 100,000 exmples

Test Set Performance



Performance (learning curve) over 10 million examples

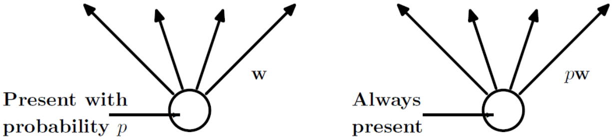
Dropout



(a) Standard Neural Net (b) After applying dropout.

Crossed out units have been dropped probabilistically during parameter updates

Dropout



(a) At training time (b) At test time

At test time, the weights are multiplied by (dropout) probability p
The expected output stays the same

Feed-forward with Dropout

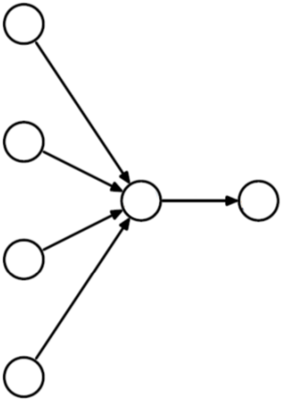
- Feed-forward neural network for node i:

$$\begin{aligned} a_i^{(\ell+1)} &= \mathbf{w}_i^{(\ell+1)}\mathbf{z}^{(\ell)} + b_i^{(\ell+1)} \\ z_i^{(\ell+1)} &= h(a_i^{(\ell+1)}) \end{aligned}$$

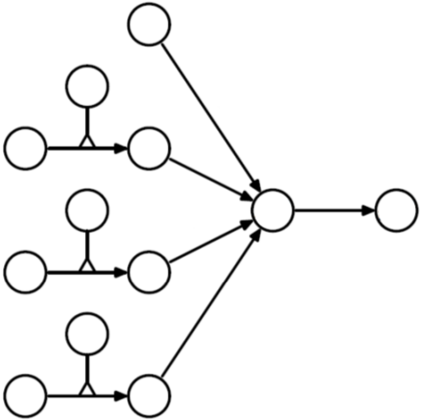
- Feed-forward with dropout neural network for node i:

$$\begin{aligned} r_j^{(\ell)} &\sim \text{Bernoulli}(p) \\ \tilde{\mathbf{z}}^{(\ell)} &= \mathbf{r}^{(\ell)} \odot \mathbf{z}^{(\ell)} \\ a_i^{(\ell+1)} &= \mathbf{w}_i^{(\ell+1)}\tilde{\mathbf{z}}^{(\ell)} + b_i^{(\ell+1)} \\ z_i^{(\ell+1)} &= h(a_i^{(\ell+1)}) \end{aligned}$$

Feed-forward with Dropout



(a) Standard network



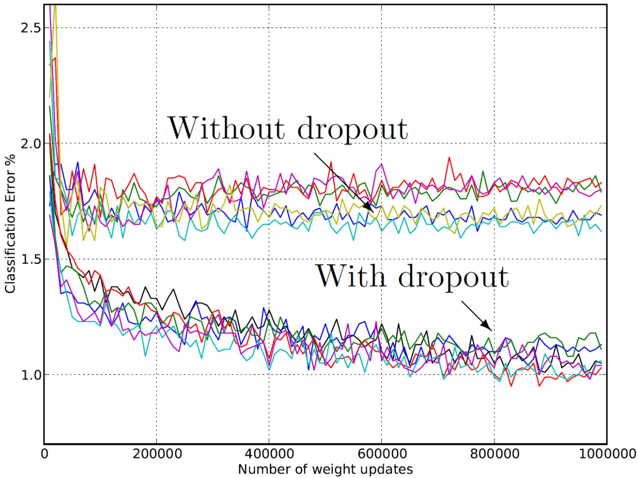
(b) Dropout network

Results on MNIST: 10 classes

Method	Unit Type	Architecture	Error %
Standard Neural Net (Simard et al., 2003)	Logistic	2 layers, 800 units	1.60
SVM Gaussian kernel	NA	NA	1.40
Dropout NN	Logistic	3 layers, 1024 units	1.35
Dropout NN	ReLU	3 layers, 1024 units	1.25
Dropout NN + max-norm constraint	ReLU	3 layers, 1024 units	1.06
Dropout NN + max-norm constraint	ReLU	3 layers, 2048 units	1.04
Dropout NN + max-norm constraint	ReLU	2 layers, 4096 units	1.01
Dropout NN + max-norm constraint	ReLU	2 layers, 8192 units	0.95
Dropout NN + max-norm constraint (Goodfellow et al., 2013)	Maxout	2 layers, (5×240) units	0.94
DBN + finetuning (Hinton and Salakhutdinov, 2006)	Logistic	500-500-2000	1.18
DBM + finetuning (Salakhutdinov and Hinton, 2009)	Logistic	500-500-2000	0.96
DBN + dropout finetuning	Logistic	500-500-2000	0.92
DBM + dropout finetuning	Logistic	500-500-2000	**0.79**

Table 2: Comparison of different models on MNIST.

Impact of Dropout on Test error



CIFAR Datasets

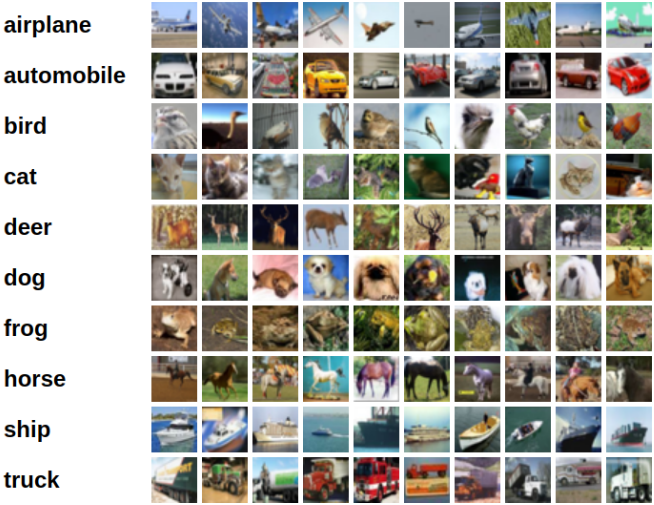
CIFAR-10 Dataset:

- 60000 32x32 color images
- 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck
- Classes are mutually exclusive

CIFAR-100 Dataset:

- Same as CIFAR-10 but has 100 classes
- Superclasses: aquatic mammals, fish, flowers, food containers, fruit and vegetables, etc.
- Classes: beaver, dolphin, shark, trout, orchids, roses, bottles, bowls, apples, peppers, etc.

CIFAR Datasets



Results on CIFAR: 10 and 100 classes

Method	**CIFAR-10**	**CIFAR-100**
Conv Net + max pooling (hand tuned)	15.60	43.48
Conv Net + stochastic pooling (Zeiler and Fergus, 2013)	15.13	42.51
Conv Net + max pooling (Snoek et al., 2012)	14.98	-
Conv Net + max pooling + dropout fully connected layers	14.32	41.26
Conv Net + max pooling + dropout in all layers	12.61	**37.20**
Conv Net + maxout (Goodfellow et al., 2013)	**11.68**	38.57

ImageNet Dataset

- 14 million images
- Annotated to indicate what objects are in each image
- 1+ million include bounding boxes
- 20000+ classes

Results on ImageNet: 1000 classes

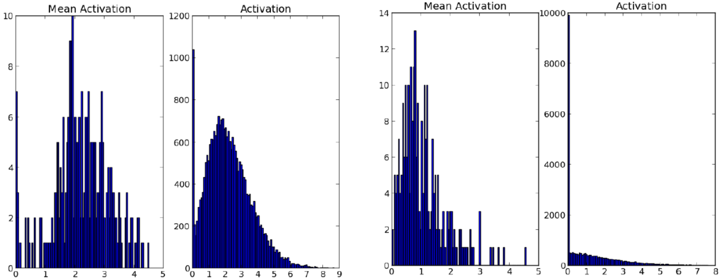
Model	Top-1	Top-5
Sparse Coding (Lin et al., 2010)	47.1	28.2
SIFT + Fisher Vectors (Sanchez and Perronnin, 2011)	45.7	25.7
Conv Net + dropout (Krizhevsky et al., 2012)	37.5	17.0

Table 5: Results on the ILSVRC-2010 test set.

Model	Top-1 (val)	Top-5 (val)	Top-5 (test)
SVM on Fisher Vectors of Dense SIFT and Color Statistics	-	-	27.3
Avg of classifiers over FVs of SIFT, LBP, GIST and CSIFT	-	-	26.2
Conv Net + dropout (Krizhevsky et al., 2012)	40.7	18.2	-
Avg of 5 Conv Nets + dropout (Krizhevsky et al., 2012)	38.1	16.4	16.4

Table 6: Results on the ILSVRC-2012 validation/test set.

Effect of Dropout on Sparsity



(a) Without dropout (b) Dropout with $p = 0.5$.

With dropout, mean activation is lower, around 0.7

With dropout, activation peaks sharply at zero