Final Presentation

Michael Downs

Dartmouth College

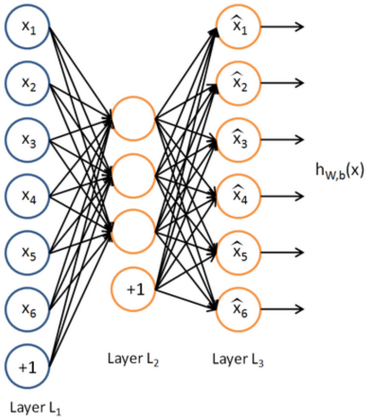
June 2nd, 2015

Project

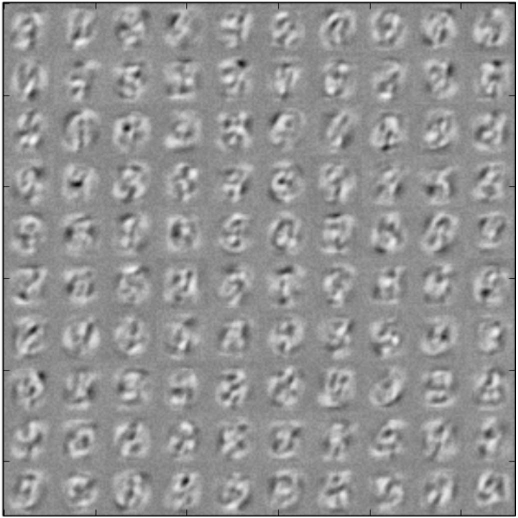
- Learn & use Theano
- Pre-training weights with autoencoders acts as a regularization mechanism – what effects do different autoencoders have, if any? Do any yield superior pretraining?
- Analyze novel restrictive autoencoder

Autoencoder

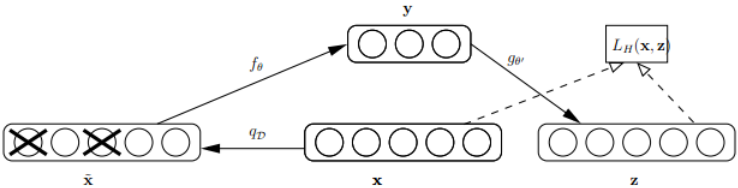
- 100 hidden units
- Binary cross entropy loss
- Tied weights
- 100 epochs



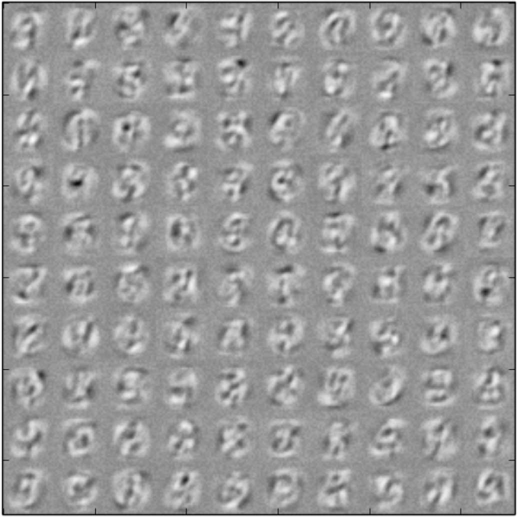
Autoencoder Filters



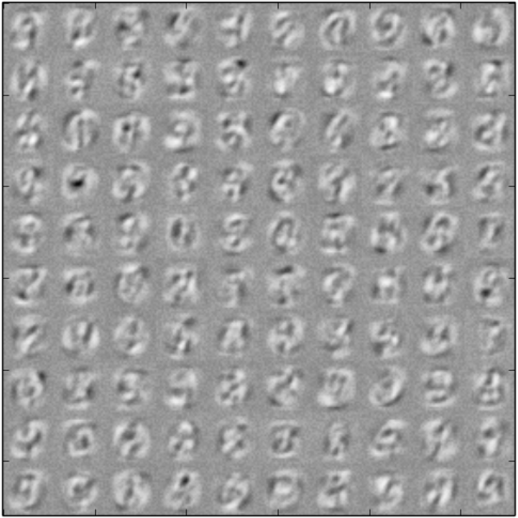
Denoising Autoencoder



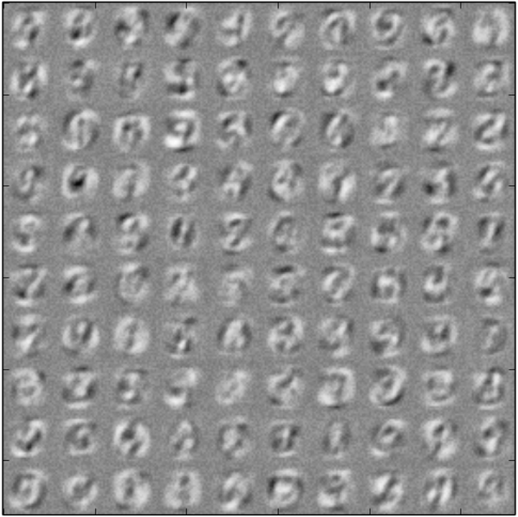
Denoising Autoencoder Filters - 20% Corruption



Denoising Autoencoder Filters - 50% Corruption

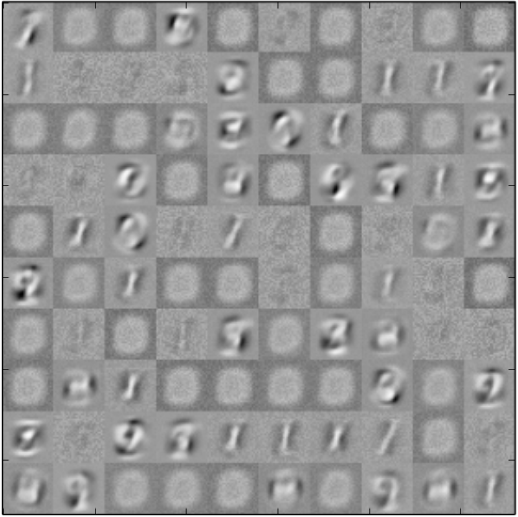


Denoising Autoencoder Filters - 80% Corruption



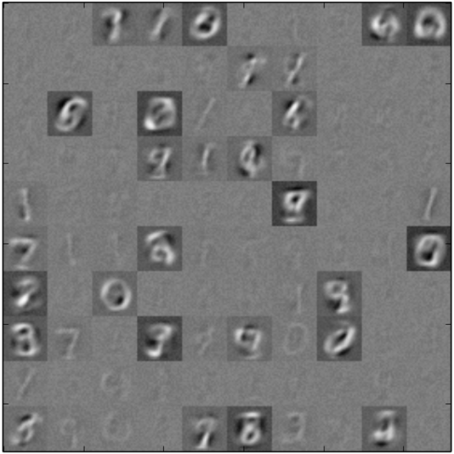
Contractive Autoencoder

- $C_{Contractive}(W, b) = C(W, b) + \lambda \sum_{i=1}^{D} ||\nabla_{x^{(i)}} h(x^{(i)})||_F^2$



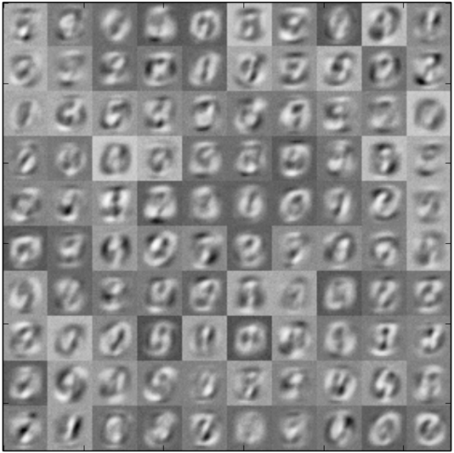
Sparse Autoencoder

- $\hat{\rho}_j = \frac{1}{m}\sum_{i=1}^{m}(a_j^{(2)}(x^{(i)}))$
- $KL(\rho||\hat{\rho}_J) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho)\log \frac{1-\rho}{1-\hat{\rho}_j}$
- $C_{Sparse} = C(W,b) + \lambda \sum_{i=1}^{n_2} KL(\rho||\hat{\rho})_j$

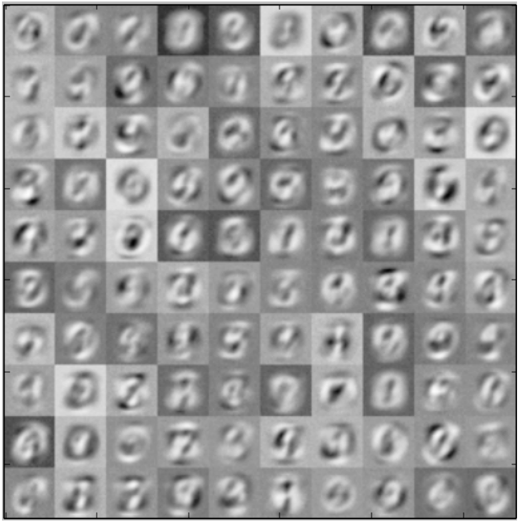


Restrictive Autoencoder - $\alpha = 50$

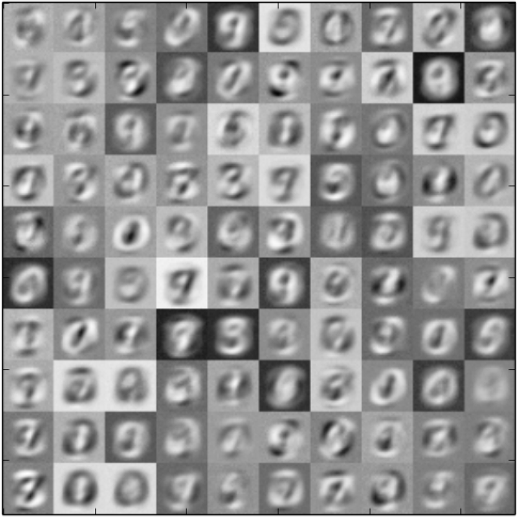
- $W \in \mathbb{R}^{n_2 x n_1}$
- $W = UV$
- $U \in \mathbb{R}^{n_2 x \alpha}$
- $V \in \mathbb{R}^{\alpha x n_1}$



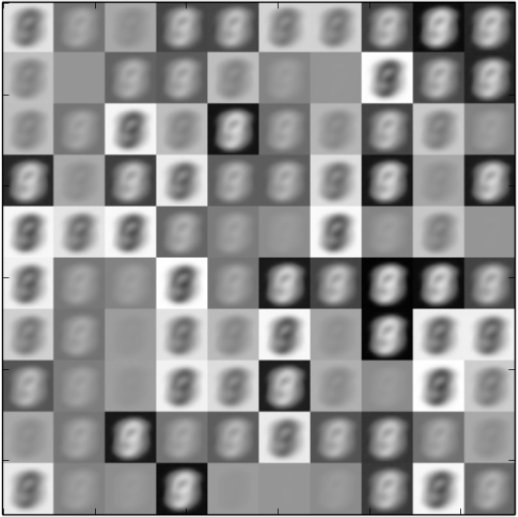
Restrictive Autoencoder - $\alpha = 25$



Restrictive Autoencoder - $\alpha = 10$

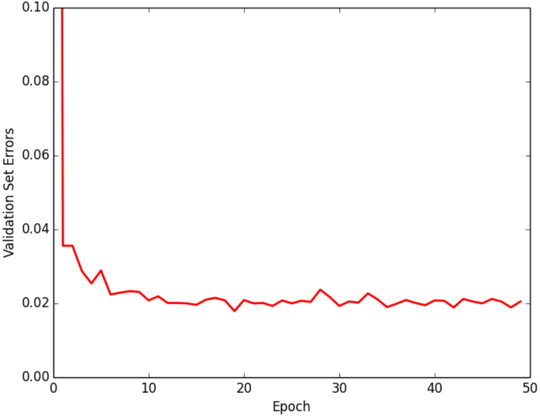


Restrictive Autoencoder - $\alpha = 1$



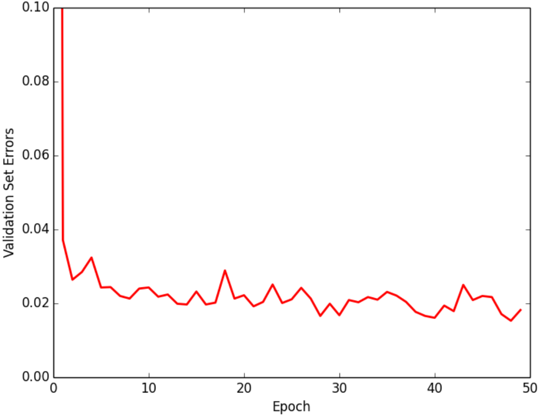
Shallow Network

- 784, 100, 10
- ReLu activations
- Weight decay, $\lambda = 0.0001$
- 50 epochs, 0.2 learning rate, 2.02% test error



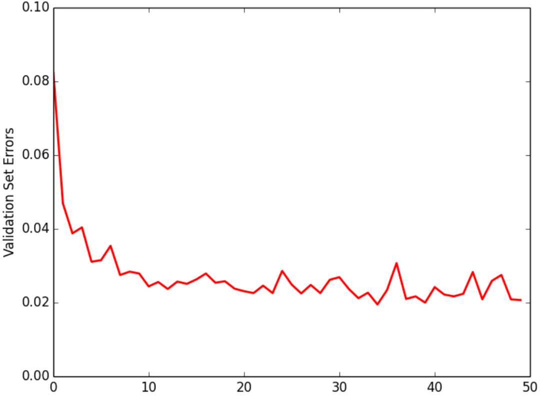
Deep Network - ReLu

- 784, 500, 250, 100, 10
- ReLu activations
- 1.57 % test error



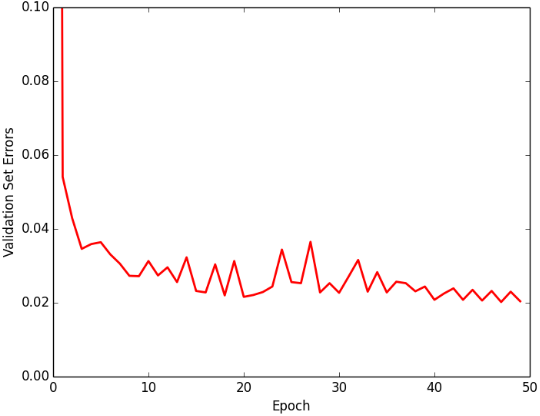
Deep Network - ReLu, Normal Autoencoder

- Interesting observation – error blows up when using ReLu activations after pretraining with sigmoid activation autoencoder
- Use relu activations in autoencoder instead, starts fine-tuning at 8.23% error, achieve 2.9% test error



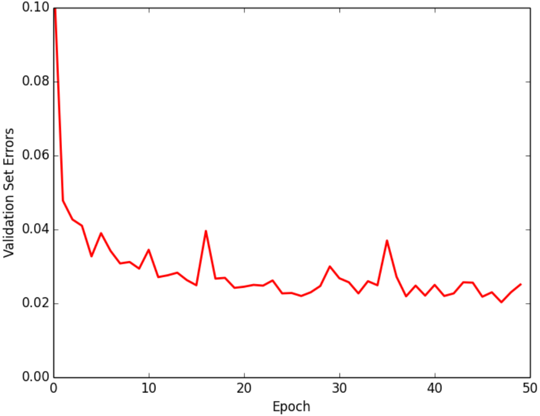
Deep Network - Sigmoid

- 784, 500, 250, 100, 10
- Sigmoid activations
- 1.98 % test error



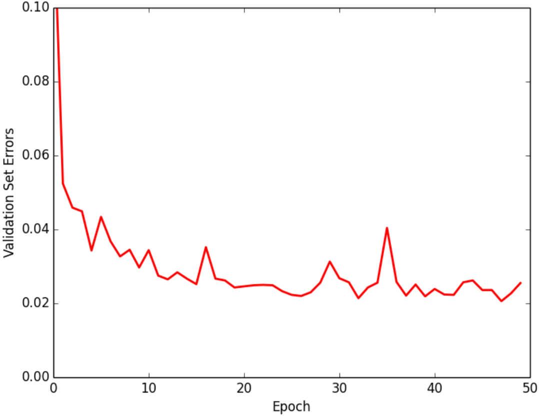
Deep Network - Sigmoid, Normal Autoencoder

- Sigmoid activations in autoencoder and network
- Starts fine-tuning at 11.4% error, achieve 2.24% test error



Deep Network - Denoising Autoencoder 20% Corruption

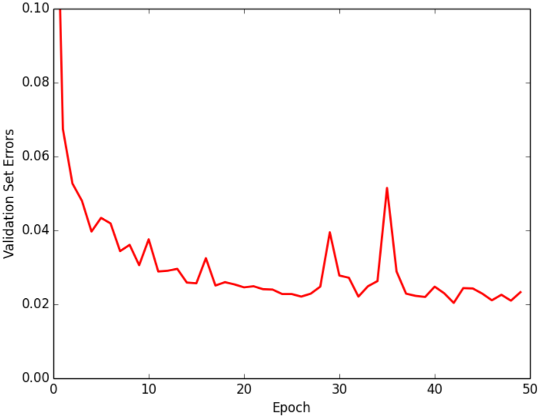
- Sigmoid activation
- Starts at 12.96% error, achieves 2.12% test error



.

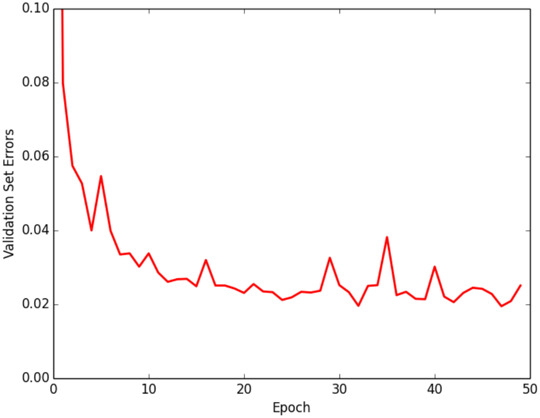
Deep Network - Denoising Autoencoder 50% Corruption

- Sigmoid activation
- Starts at 17.81% error, achieves 2.31% test error



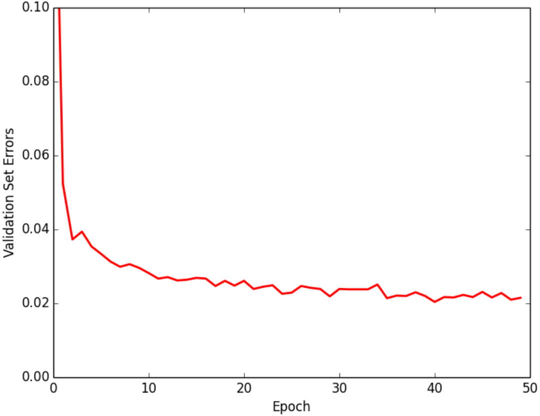
Deep Network - Denoising Autoencoder 80% Corruption

- Sigmoid activation
- Starts at 41.81% error, achieves 2.26% test error



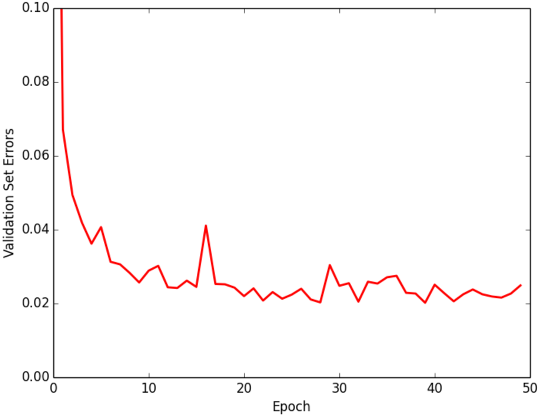
Deep Network - Contractive Autoencoder

- Sigmoid activation, .01 contraction level
- Takes significantly longer to train – pretrained on shallow network
- Starts at 17.59% error, achieves 2.08% test error



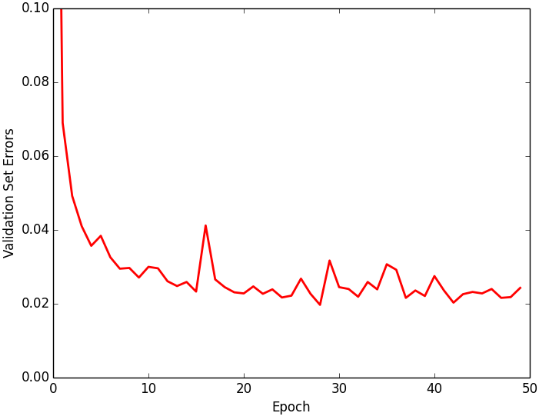
Deep Network - Sparse Autoencoder, $\rho = 0.01$

- Sigmoid activation, 0.5 sparsity
- Starts at 28.87% error, achieves 2.2% test error



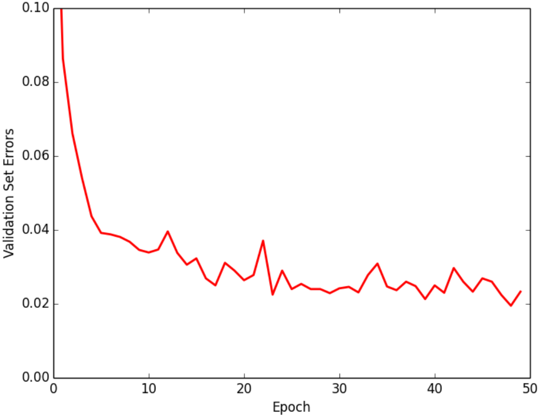
Deep Network - Sparse Autoencoder, $\rho = 0.001$

- Sigmoid activation, 0.5 sparsity
- Starts at 28.43% error, achieves 2.08% test error



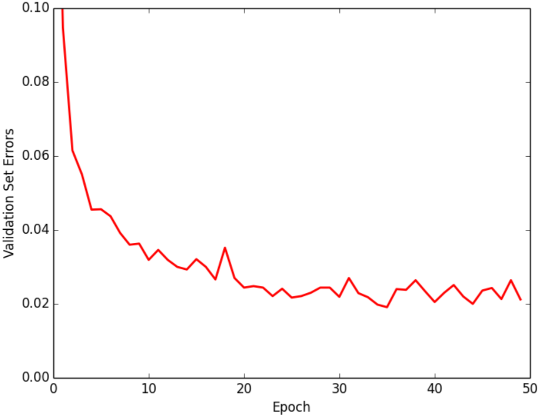
Deep Network - Restrictive Autoencoder, $\alpha = 50$

- Sigmoid activation
- Starts at 16.7% error, achieves 2.59% test error



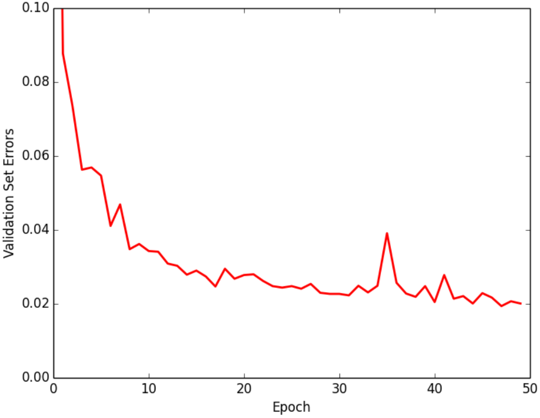
Deep Network - Restrictive Autoencoder, $\alpha = 25$

- Sigmoid activation
- Starts at 20.62% error, achieves 2.12% test error



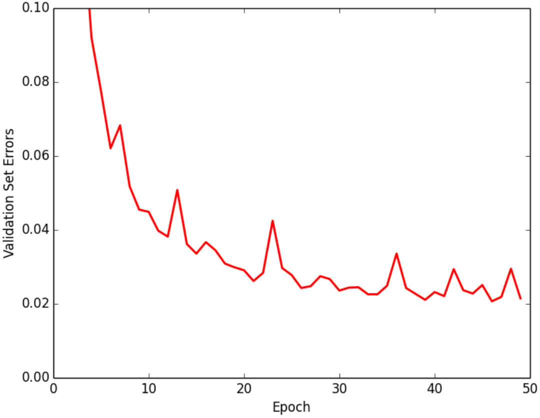
Deep Network - Restrictive Autoencoder, $\alpha = 10$

- Sigmoid activation
- Starts at 30.89% error, achieves 2.31% test error



Deep Network - Restrictive Autoencoder, $\alpha = 1$

- Sigmoid activation
- Starts at 82.42% error, achieves 2.35% test error



Conclusions

- Restrictive autoencoder learns nontrivial structure in data, reduces parameters to $\mathit{inputdim} * \alpha$
- Unfortunately, pretraining not able to outperform random initialization in experiments
- Best result on sparse autoencoder
- Seems that stochasticity of training overshadows effect of pretraining

Future Work

- Determine why learning only V (in W $=$ UV factorization) does not yield meaningful features
- Analyze effect of restricting parameters - regularization?
- α can also be made larger than the outer dimensions – effect?
- Use different hyperparameters, train longer