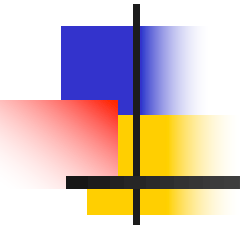
CMSC 510 – L19
Regularization Methods for Machine Learning

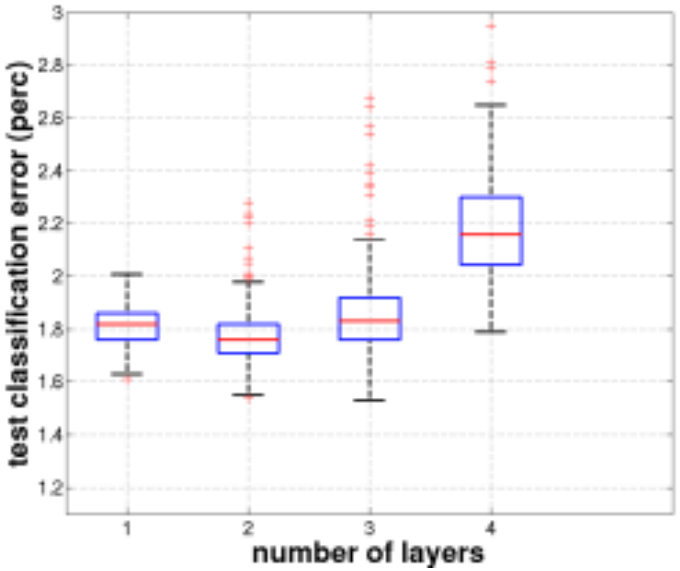


Part 19b: Greedy Pretaining

Instructor:
Dr. Tom Arodz

Training deep nets

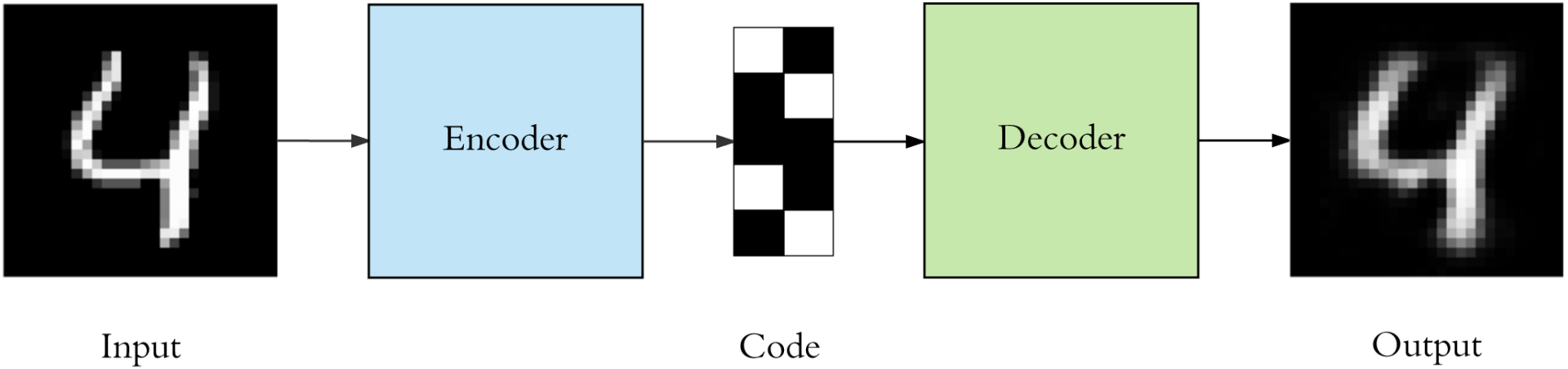
- Historically, deep networks (with many layers) were difficult to train



- A lot effort in recent years went into making training easier
  - ReLU instead of sigmoid
  - Unsupervised pre-training
  - Normalization techniques

Pre-training

- Pre-training for dealing with depth of the network
  - Greedy layer-wise pretraining
- First, training an autoencoder unsupervised network

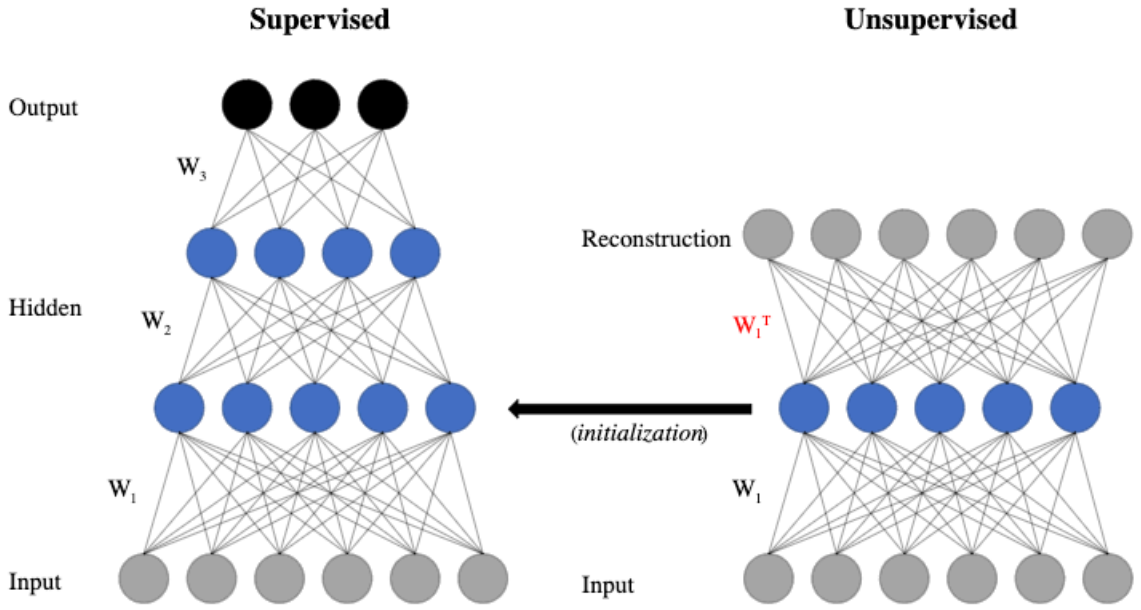


  - Two networks: an encoder + a decoder
  - Unsupervised training (no classes)
    - MSE: $(x - \text{decoder}(\text{encoder}(x))^2$

https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798

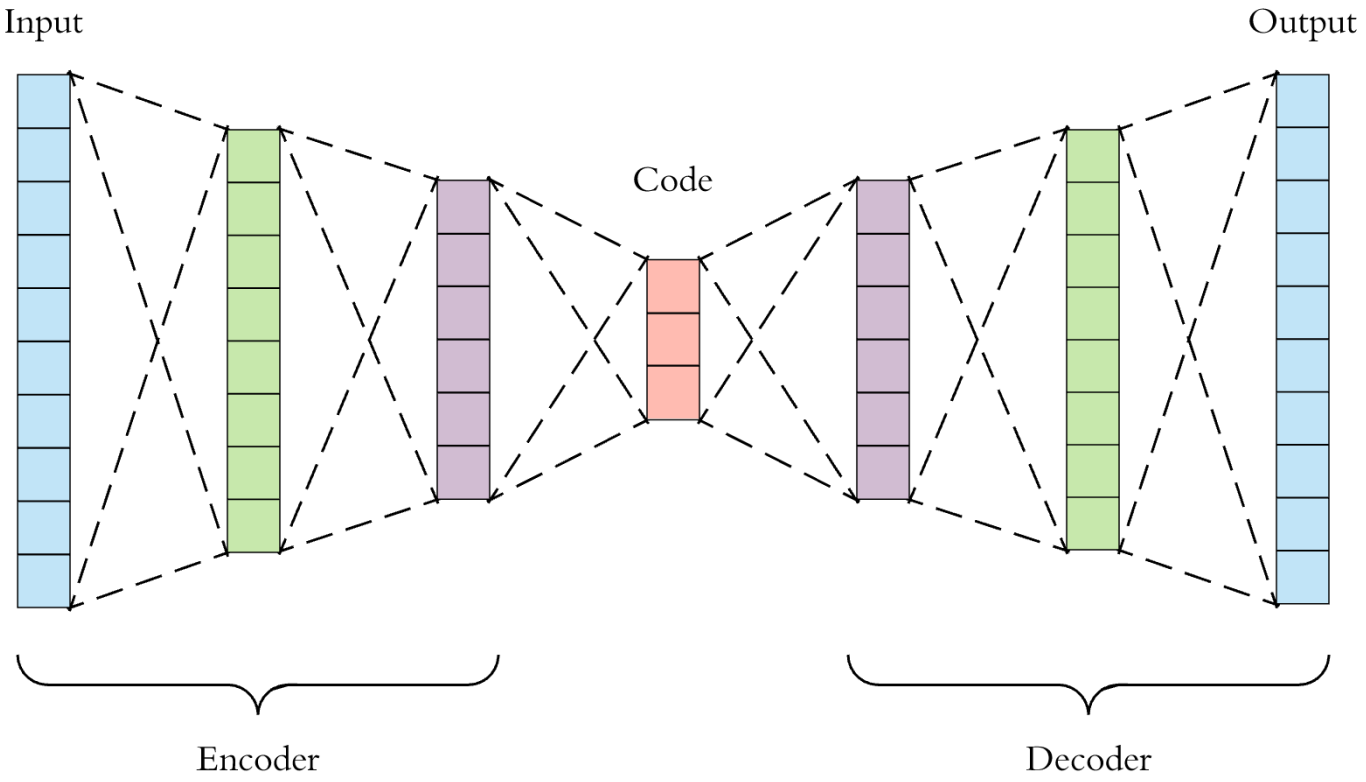
Pre-training

- Pre-training for dealing with depth of the network
  - Greedy layer-wise pretraining
- Second:
  - eliminate the decoder
  - take the encoder part
  - add a layer or two
  - softmax on top
  - train for classification (e.g. cross-entropy)



Pre-training

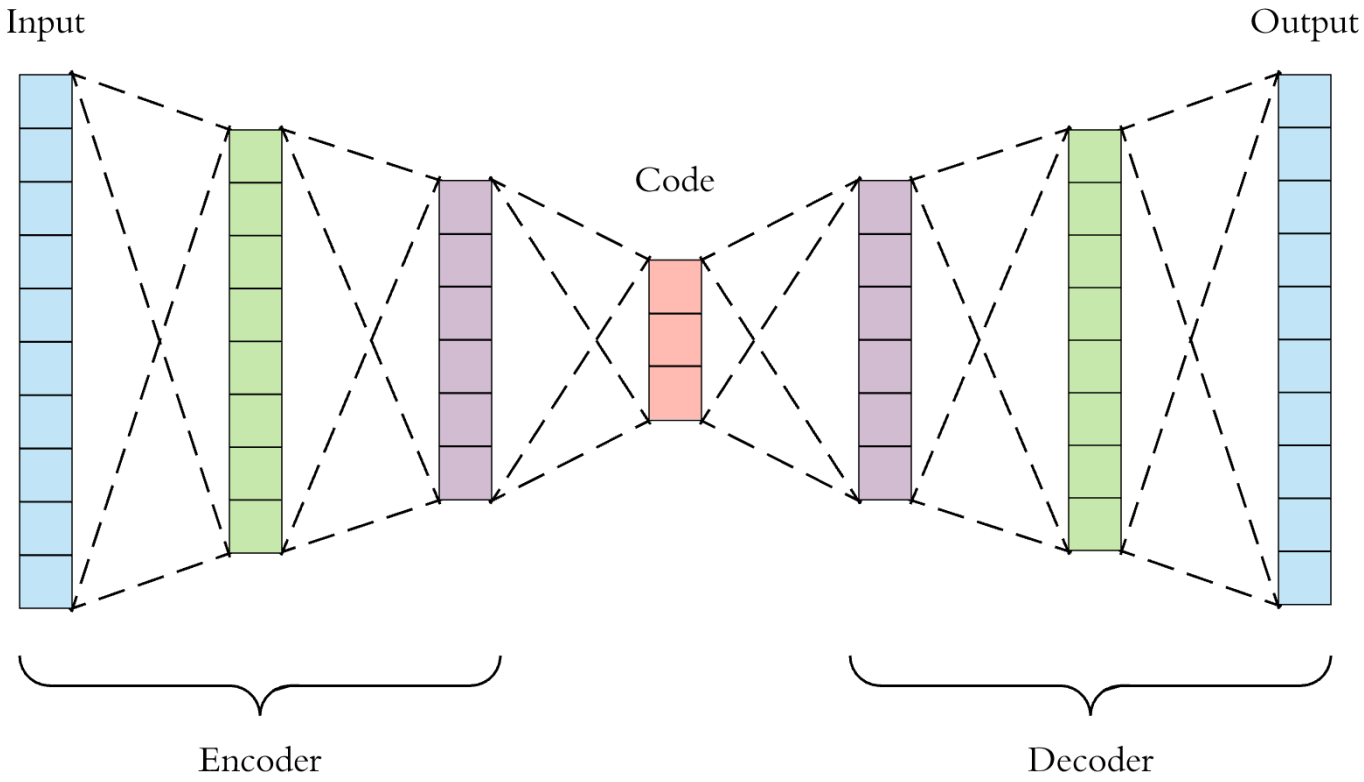
- training an autoencoder unsupervised network



- If the final supervised network needs to be large (many layers) the autoencoder also needs to be large (many layers)

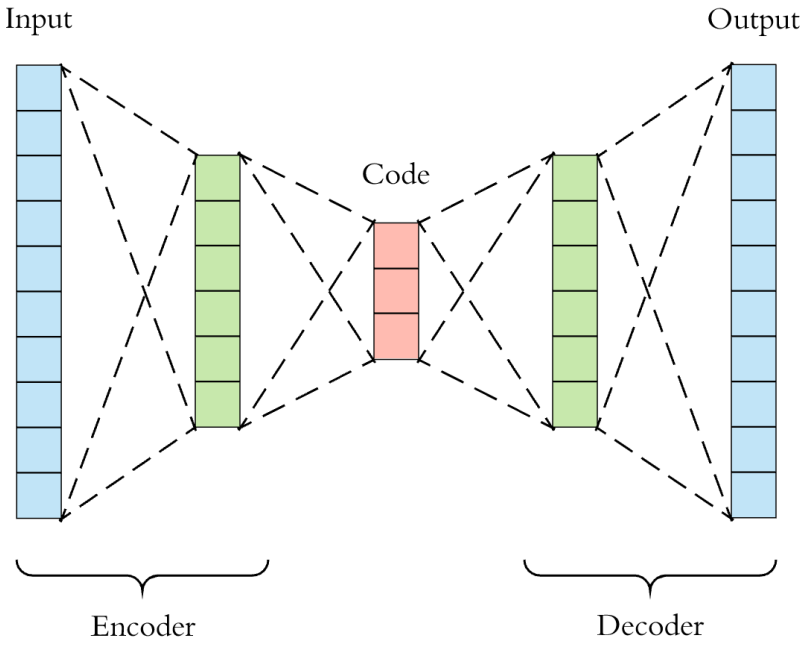
Pre-training

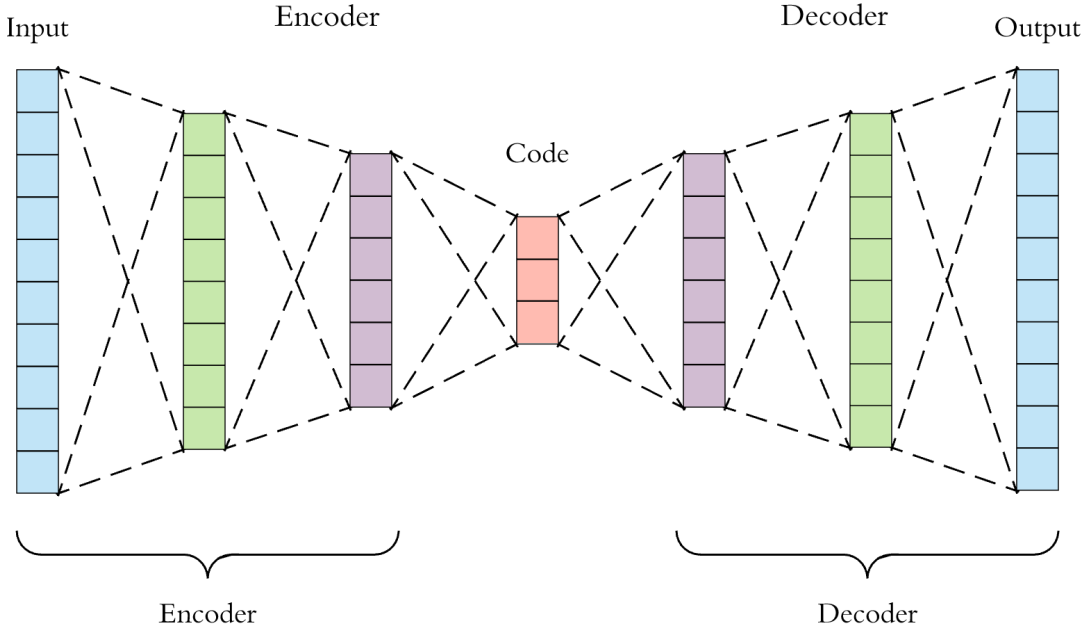
- Autoencoder with many layers
  - *n* layers - may be as difficult to train as a deep supervised network with *n* layers



Pre-training

- Greedy layer-wise pretraining
  - First train a shallow autoencoder (one hidden layer – green)
  - Next, add another hidden layer (violet)
    - Keep the previous weights as initialization
  - Repeat, adding one layer at a time
    - Actually, two: one in encoder one in decoder

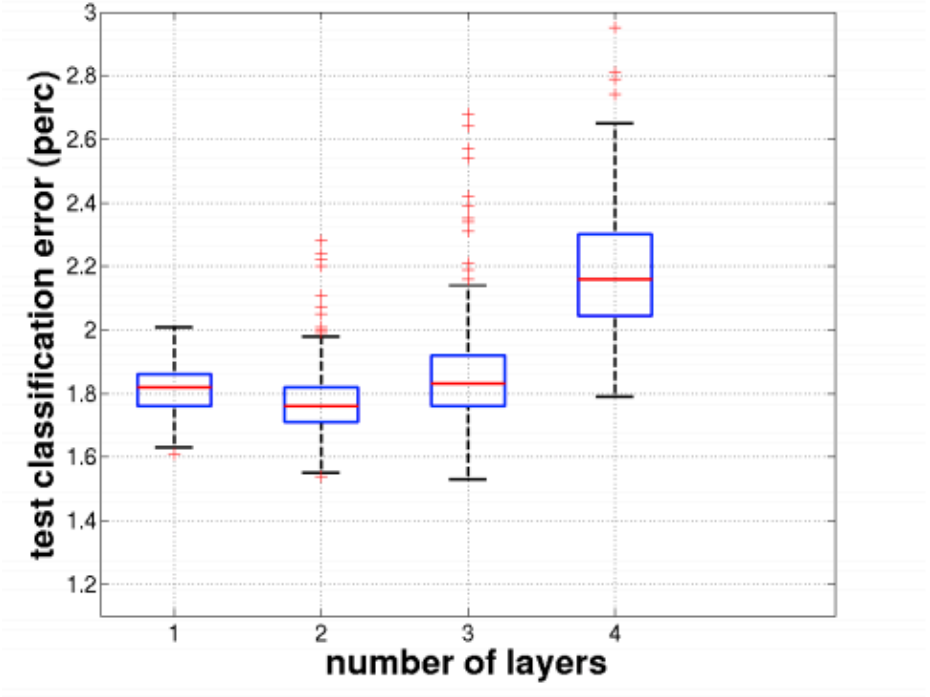


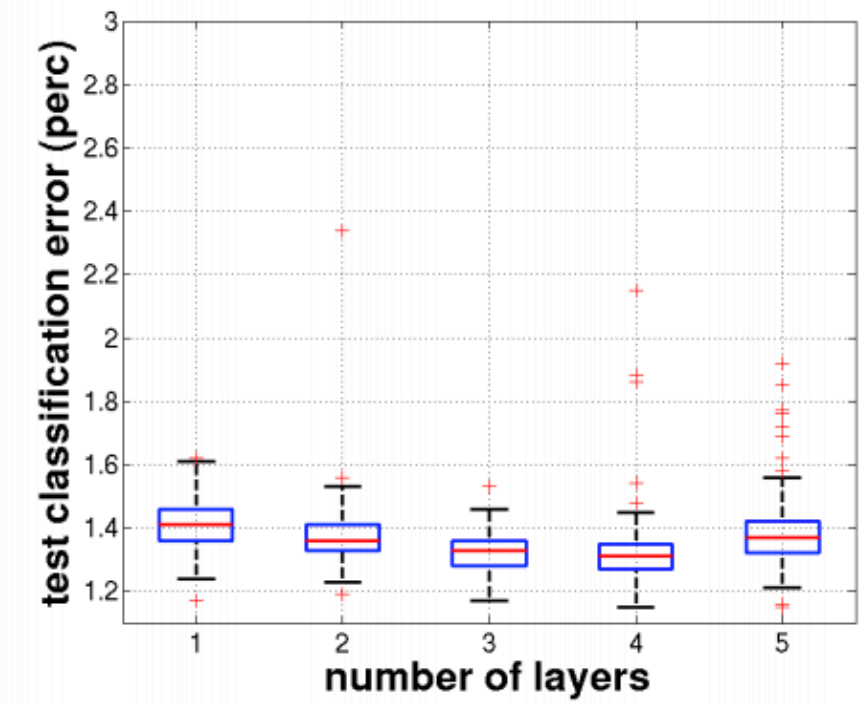


Training deep nets

- Pre-training can help in training deep nets

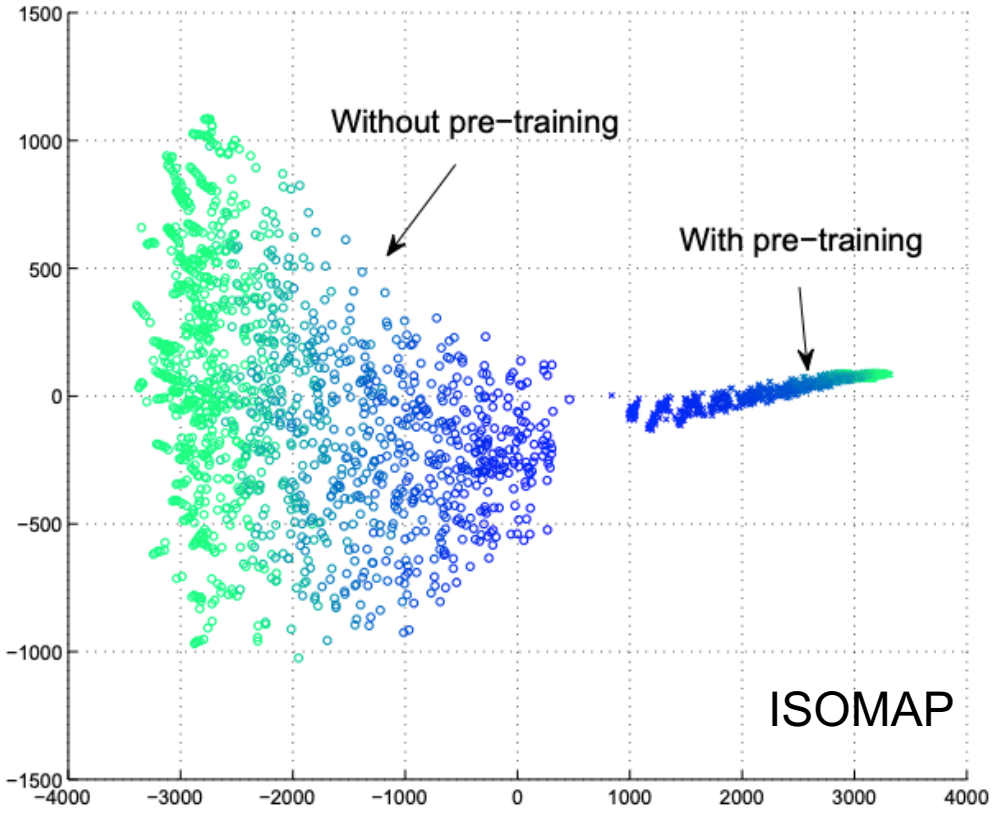
(no pre-training) (with pre-training)





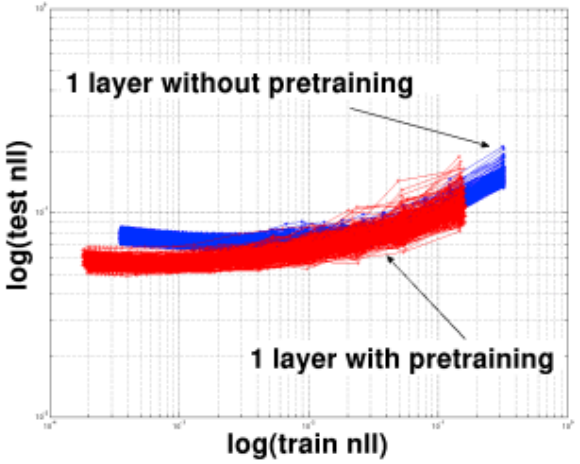
Training deep nets

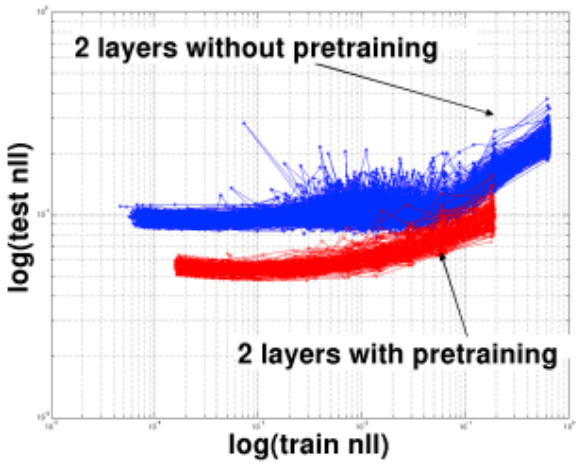
- Pre-training serves as a regularizer
  - Different runs of pre-trained nets are similar, and converge during training
  - Different runs of non-pre-trained nets progress in different directions during training (blue to green)

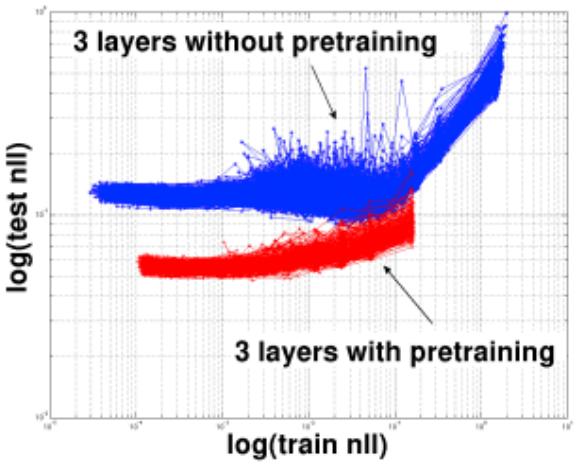


Training deep nets

- Pre-trained networks overtrain less
  - For the same training loss test loss is lower for pre-trained net

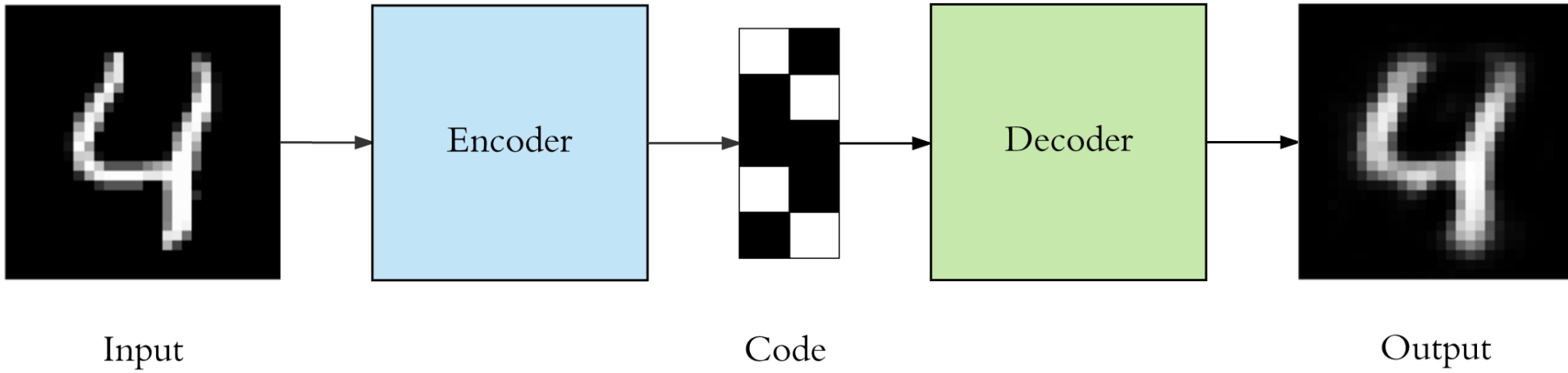




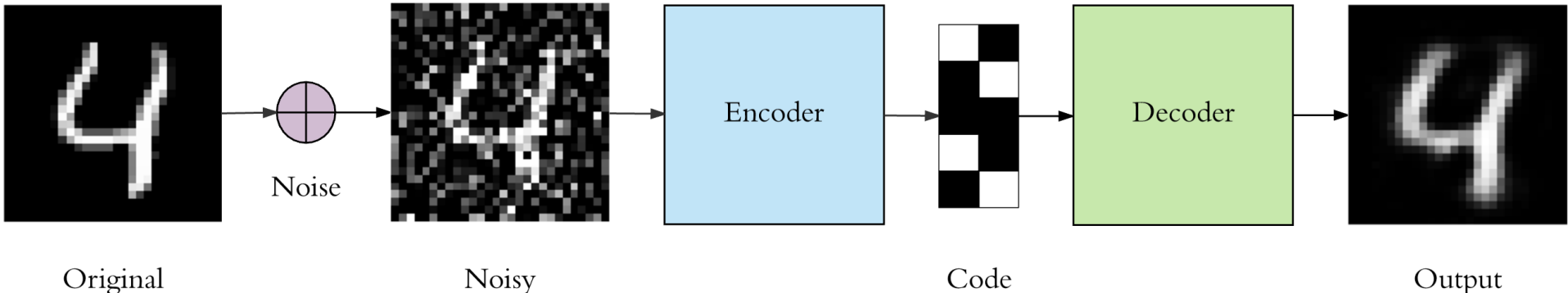


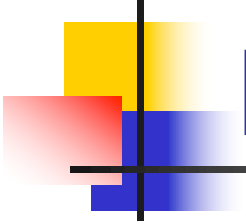
Pre-training

- De-noising autoencoder:



  - Can make the network more resistant to minor variations in input





Pre-training

- Pre-training for dealing with depth of the network
  - Greedy layer-wise pretraining
  - Is not as much needed these days – there are other techniques for helping with optimization of deep nets
- Pre-training (not greedy, not layer-wise) is still very useful for dealing with limited training data
  - Transfer learning
  - Teacher-student learning