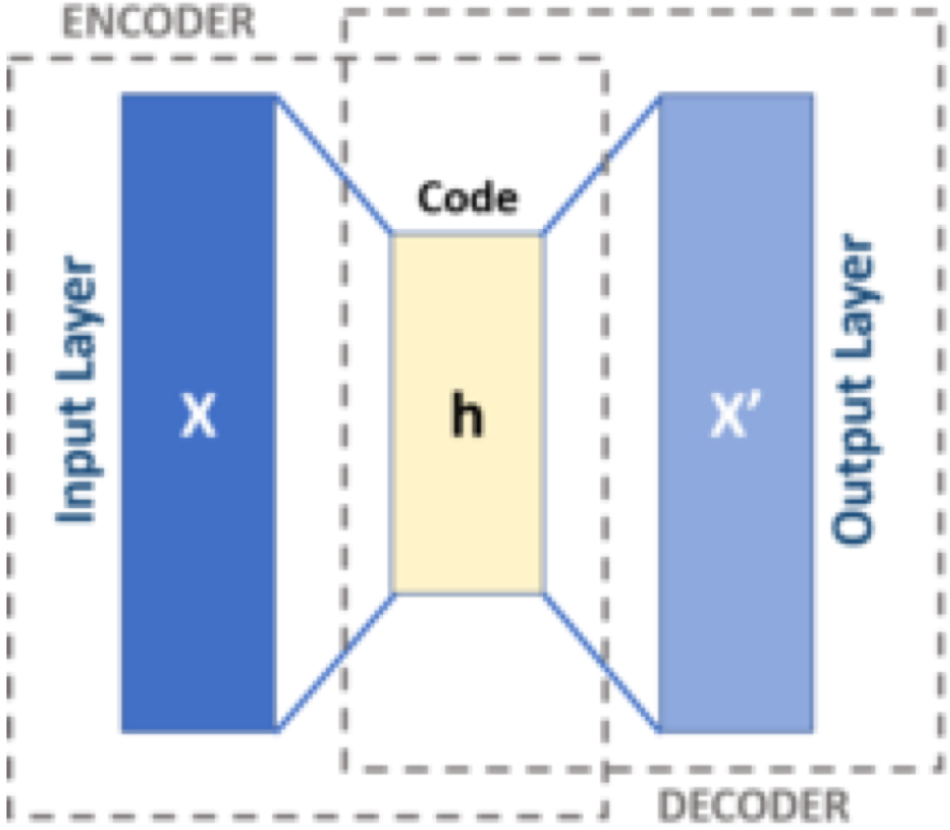
Autoencoders and Representations

Lecture 5

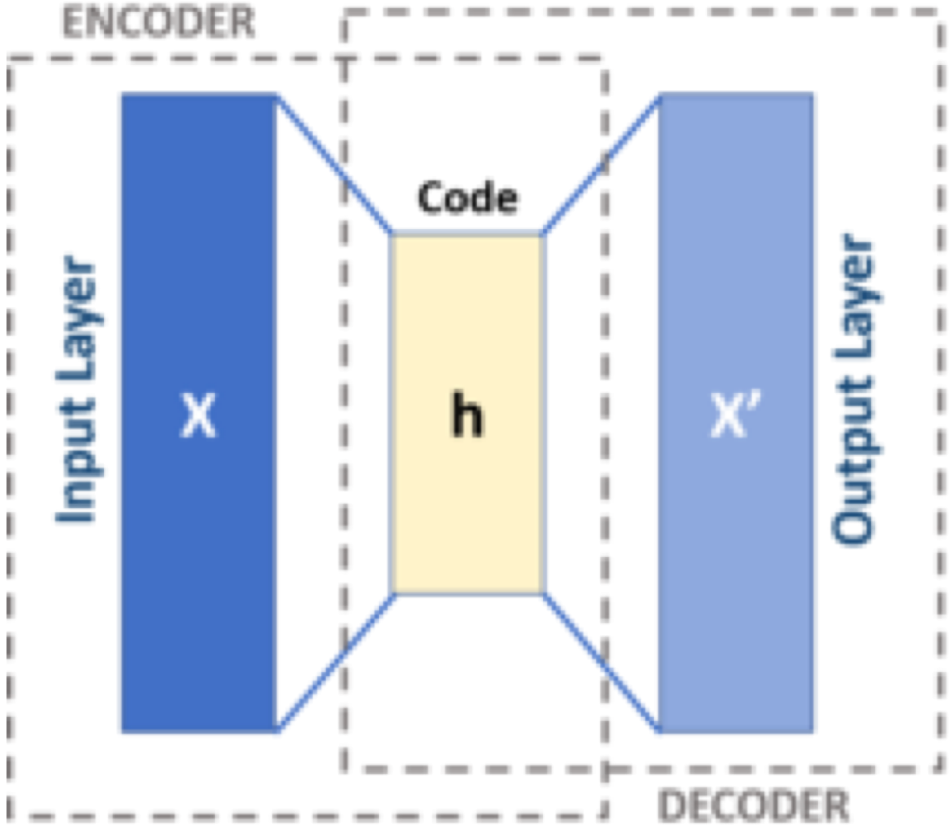
Autoencoder

- Neural network trained copy its input to its output
- Divided into two pieces
  - Encoder - $h = f(x)$
  - Decoder - $r = g(h)$
- Learns $g\big(f(x)\big) = x$
- Trained like a normal neural network
- Considered a form of unsupervised learning



Restricting Autoencoders

- If the dimension of h is equal to or greater than or equal to the size of the input, x, then it will learn the identity function
- In general we are not interested in the output of the model, but instead the output of the hidden layer h
- Several methods are used to create a more interesting output from h

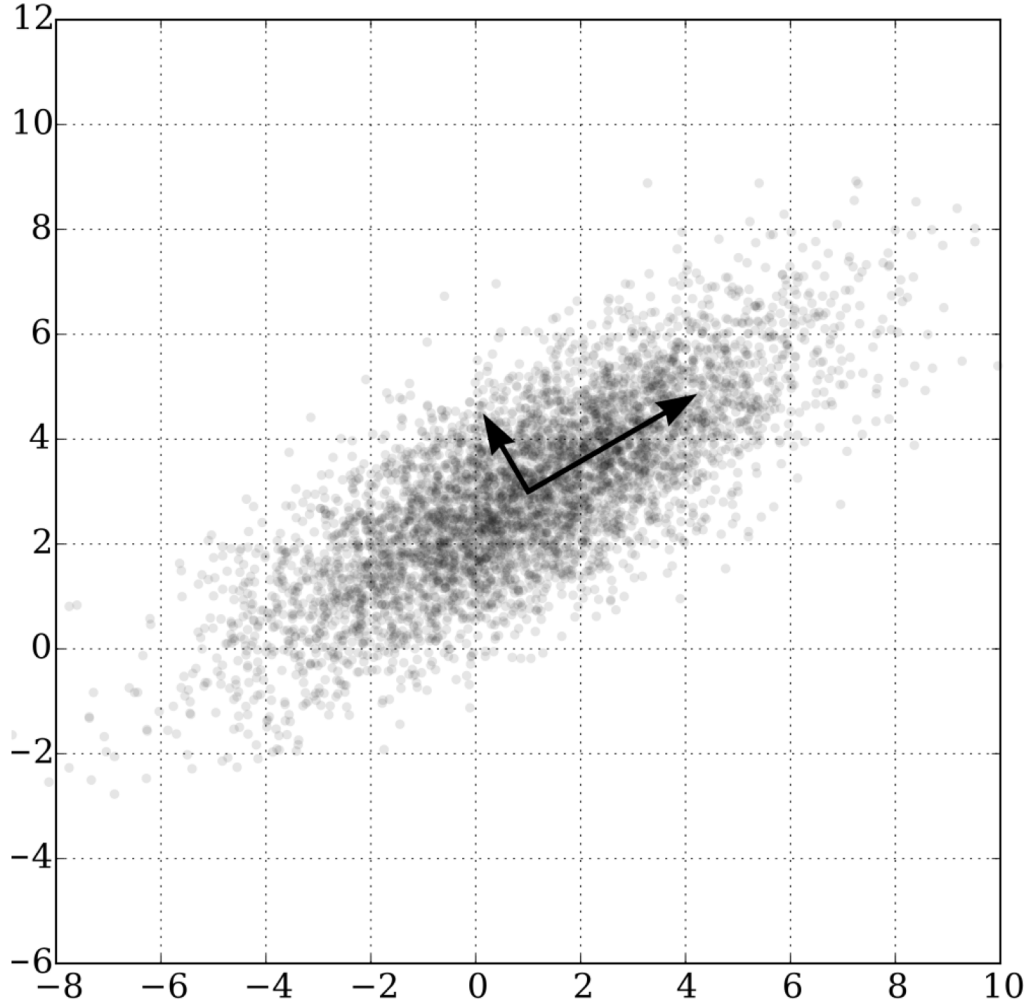


Reconstruction Error

- Error “reconstructing” the input
- $L\left(x, g\left(f\left(x\right)\right)\right)$
- Special name for the error in autoencoders

Principal Component Analysis (PCA)

- When the encoder and decoder a linear functions and the loss function is mean squared error
- Learns the principal subspace of the data
  - Direction of maximal covariance
- Useful method when relationships are linear



Undercomplete Autoencoders

- Constrain h to a smaller size than x
- Process is learning $L\left(x, g\left(f(x)\right)\right)$
- If $h < x$, then the autoencoder will learn a representation that contains as much information from x as possible

Regularized Autoencoders

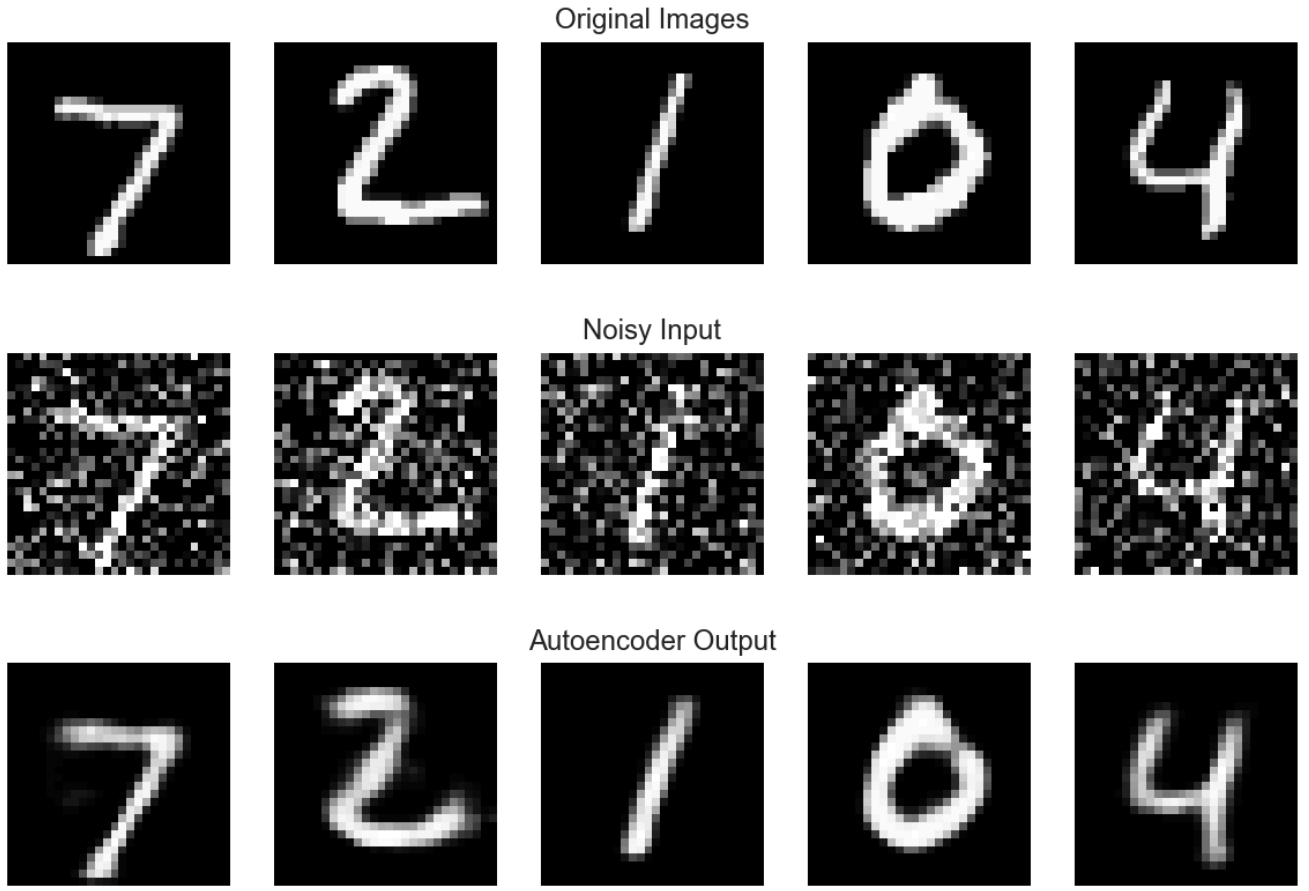
- Autoencoders with hidden size greater than or equal to the input are possible
  - The size of the hidden layer is the capacity of the model
- By adding regularization to the weights of the hidden layers, the hidden representations can learn interesting representations

Sparse Autoencoders

- Add a sparsity penalty to the reconstruction error to form the regularized loss function
- $J(x) = L\left(x, g\left(f(x)\right)\right) + \alpha\Omega(W_f)$
- Sparsity penalty typically only applied to weights of the encoder function
  - Weights of the decoder function are generally less interesting
  - Weights of decoder are also a function of the weights of the encoder

Denoising Autoencoder

- Add noise to the input, train the autoencoder to predict the “clean” input
- $J(x) = L\left(x, g\left(f\left(x + \epsilon\right)\right)\right)$

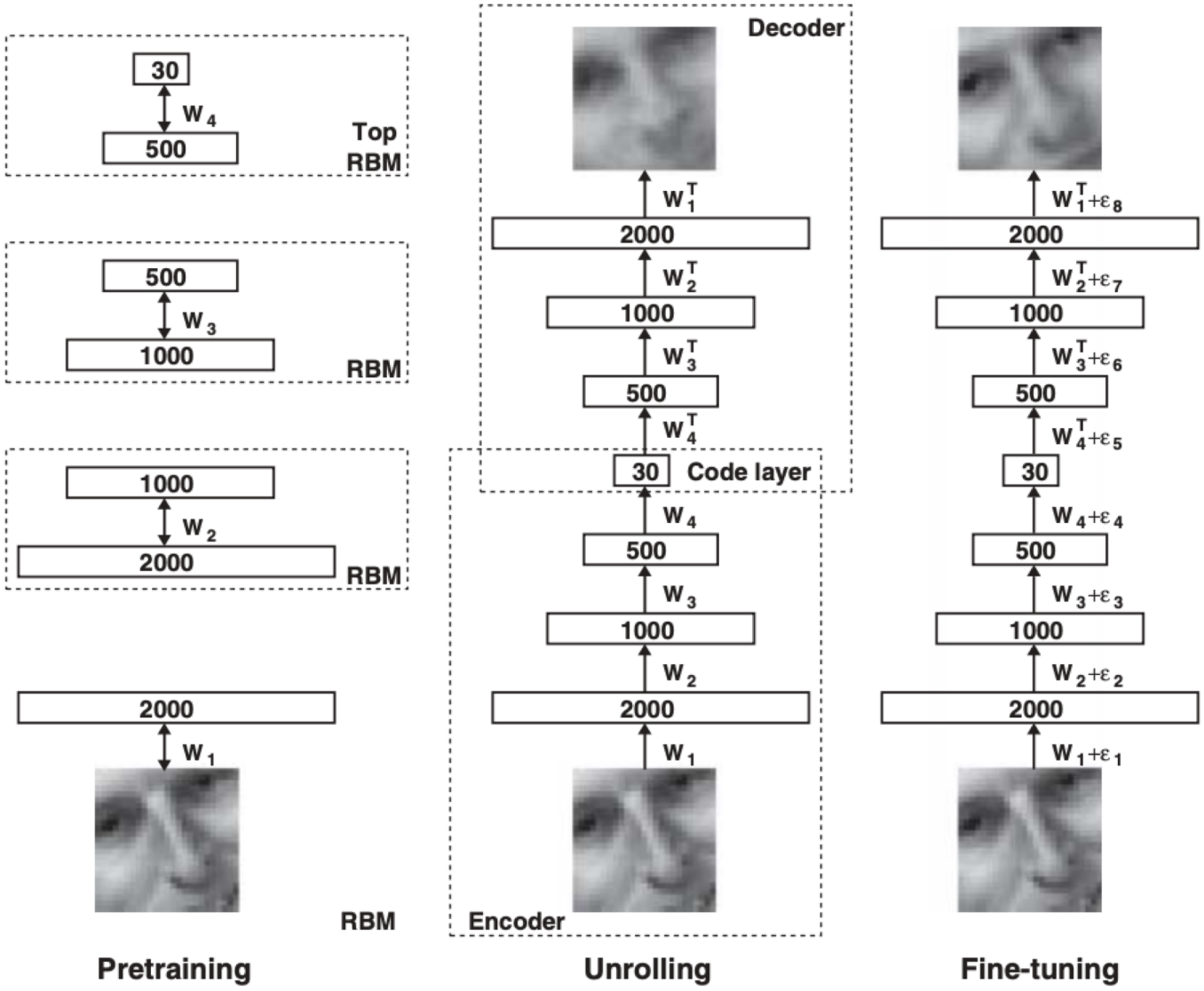


Brief discussion about noise

- Previously, there was a question about cases where different noise distributions were more appropriate than Gaussian noise
- Modeling number of days a student missed school
  - Many students will have 0 days off, some will have multiple
  - Noise distribution is different for those two different groups
- Modeling the mass of a molecule
  - Depending on the isotopes of the atoms, different patterns of molecule weight will exist
- Voxel intensity in MRI
  - Noise is Rician parameterized by the true intensity of that location

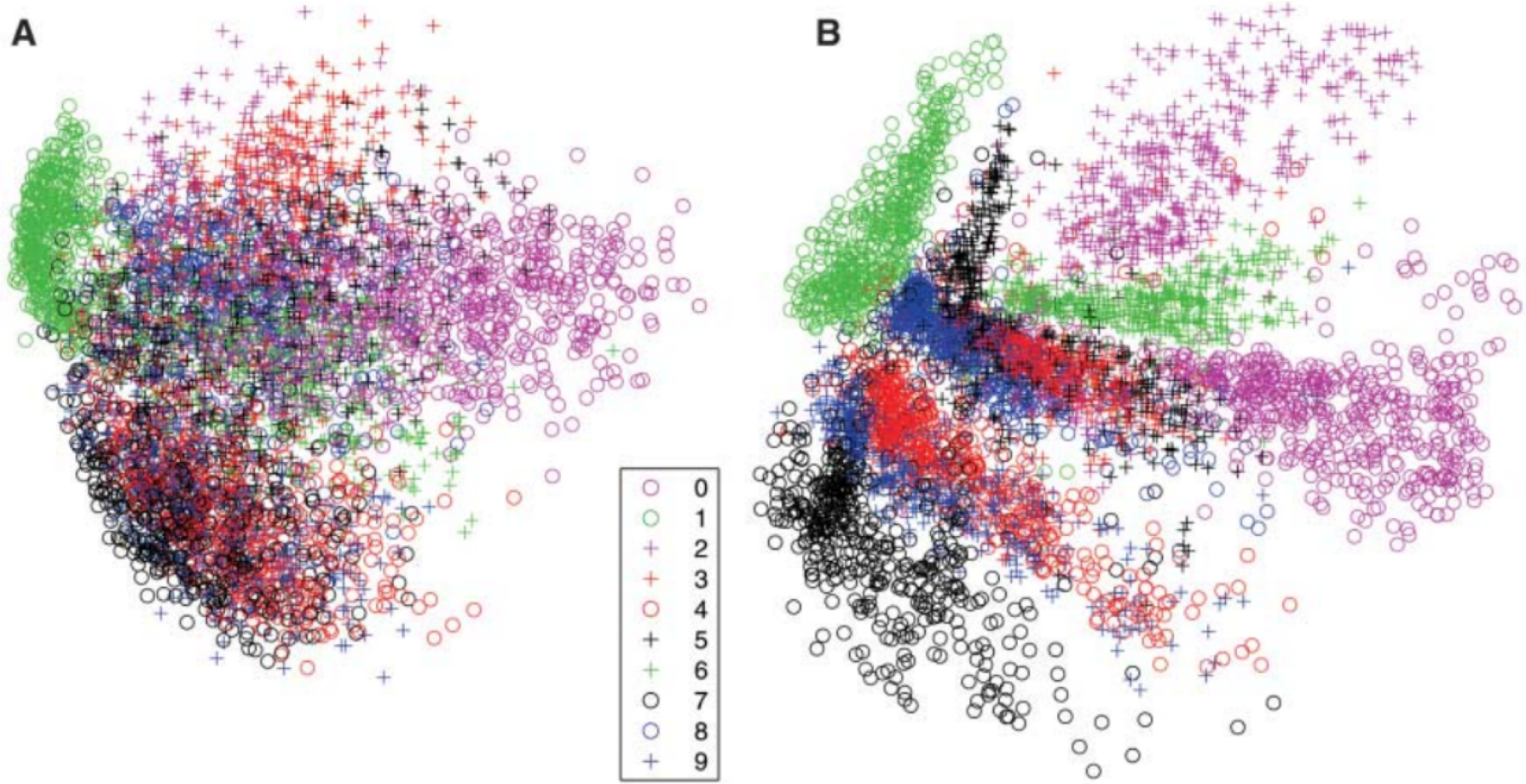
Greedy Layer-Wise Training and Stacked Autoencoders

- Train one autoencoder, then train a second on the representation learned by the first, etc
- ”Original” deep learning paper used this technique



https://www.cs.toronto.edu/~hinton/science.pdf

Decreasing the Dimensionality of Data with Neural Networks



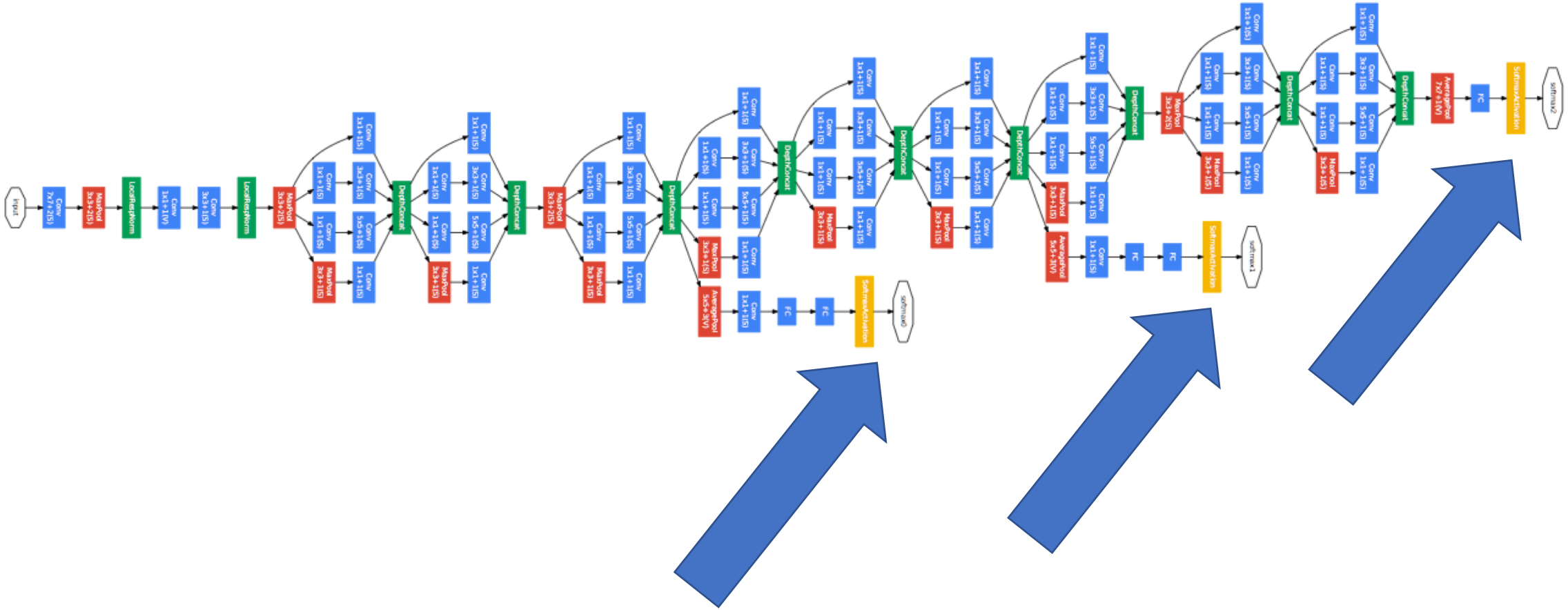
Why Autoencoders?

- In many cases the amount of true variation in some data is less than the number of input features
  - In images if one pixel were removed, it would be easy to guess what the value should be
- Often times the amount of unlabeled data available significantly outpaces the amount of labeled data
  - Autoencoders do not rely on labels and can learn from unlabeled data
  - Training off of a smaller representation on the labeled examples may then outperform training a larger network off of just the labeled cases

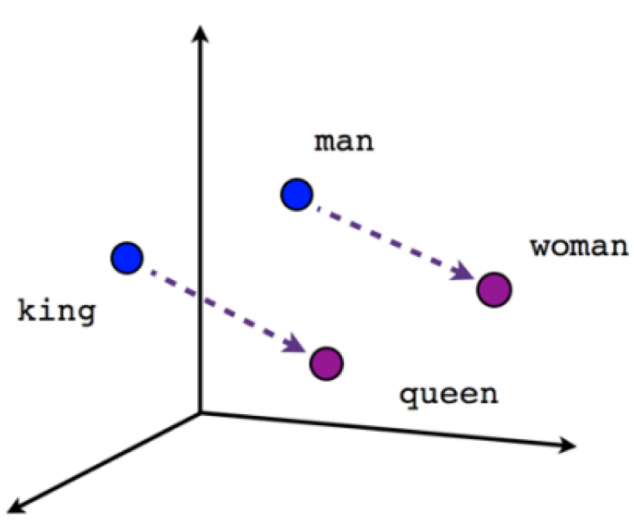
Neural Networks as Representation Learners

- The output of any layer of a neural network can be thought of as a new representation of the data
- Each layer of a network transforms the data to a new representation leading to the final task
- Depending on the task and constraints, the nature of each representation can change
  - Different loss functions product different results – cross entropy vs accuracy
  - Sparsity constraints
  - Independent features
  - Dropout – coexisting features

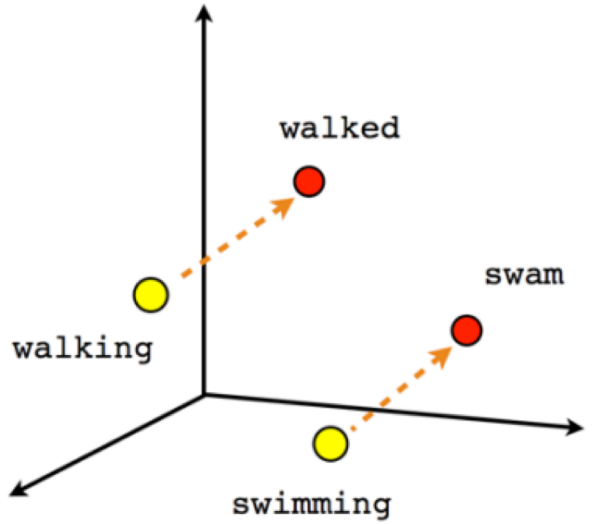
Examples of Representation Learning - Inception



Examples of Representation Learning – Word Embeddings



Male-Female



Verb tense

Failures of Representation Learning