Day 3 Lecture 1

Unsupervised Learning

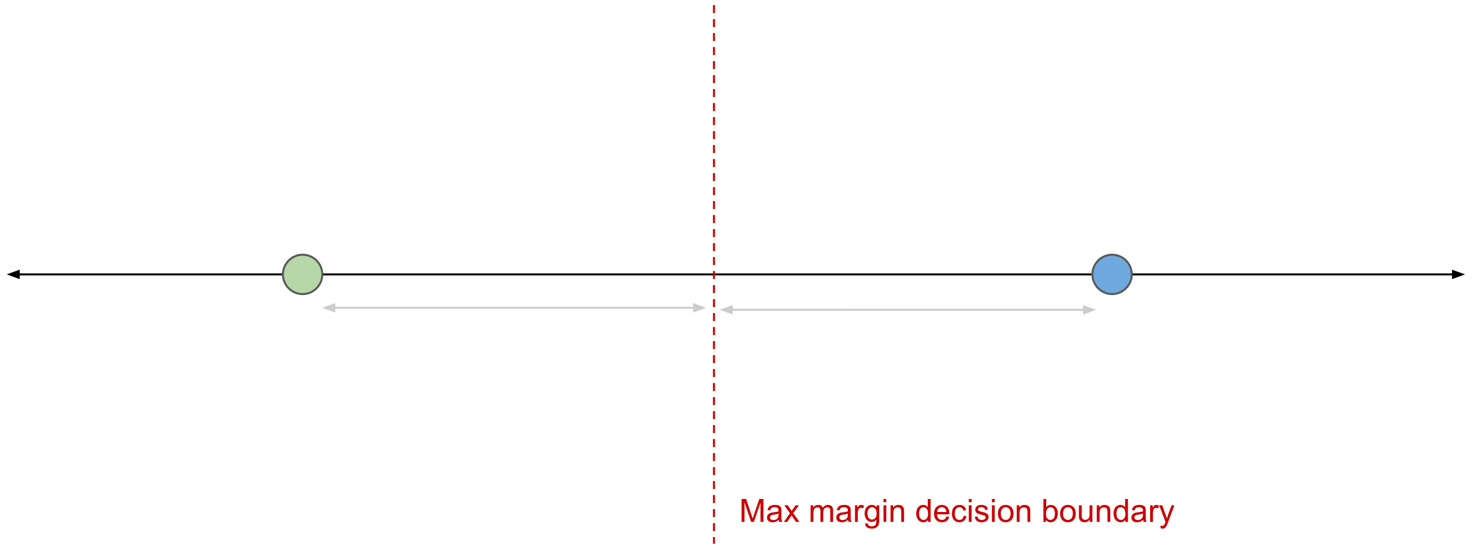
Semi-supervised and transfer learning

Myth: you can't do deep learning unless you have a million labelled examples for your problem.

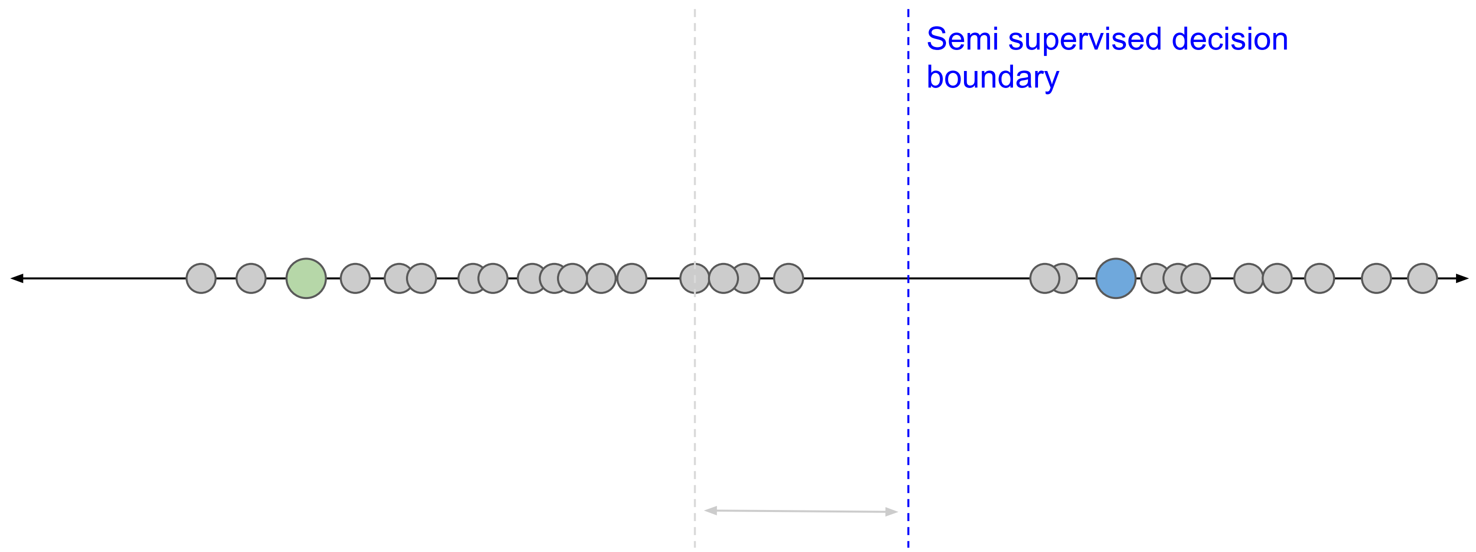
Reality

- You can learn useful representations from **unlabelled data**
- You can **transfer** learned representations from a related task
- You can train on a nearby **surrogate objective** for which it is easy to generate labels

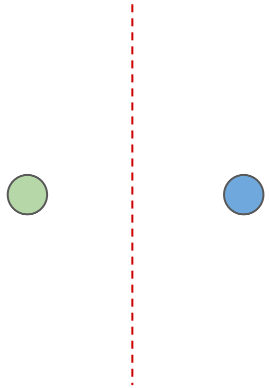
Using unlabelled examples: 1D example



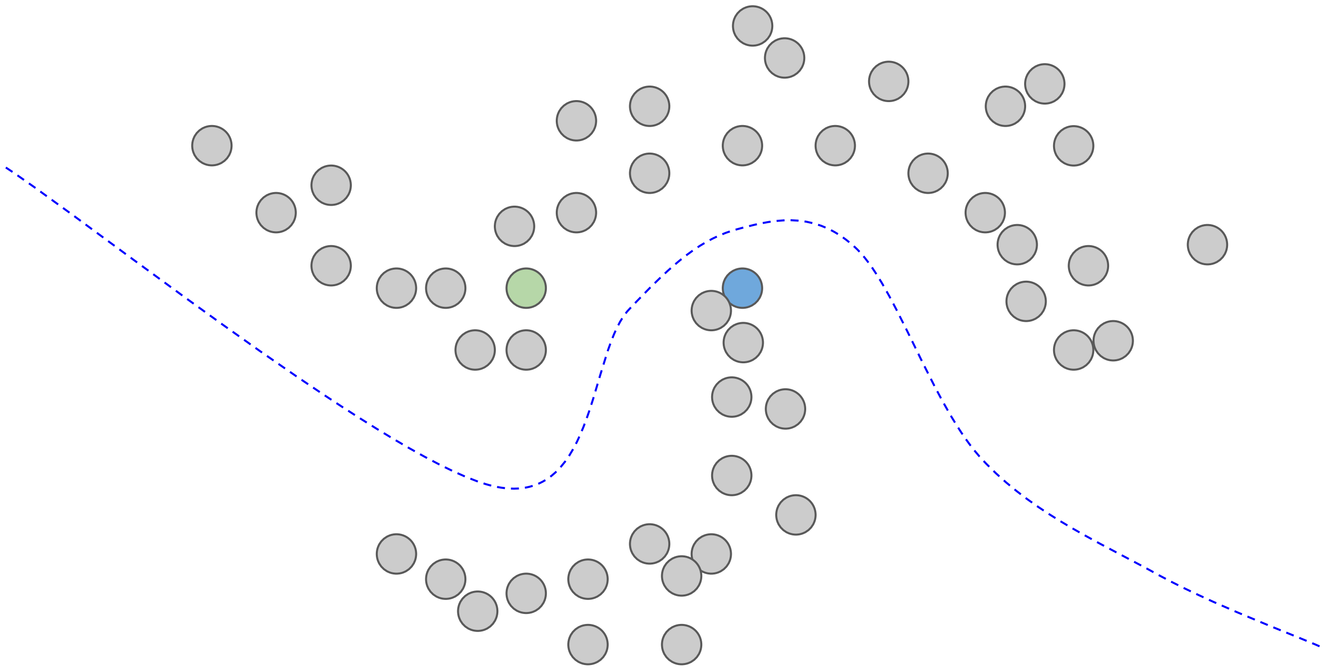
Using unlabelled examples: 1D example



Using unlabelled examples: 2D example



Using unlabelled examples: 2D example



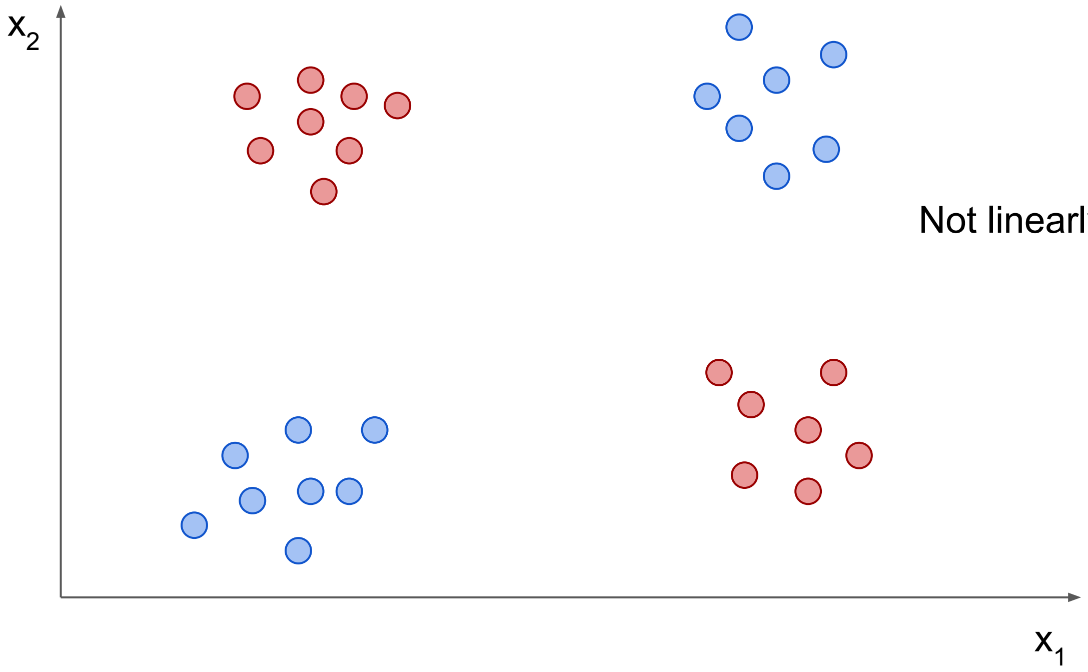
A probabilistic perspective

Bayes rule

$$P(Y=y|X)=\frac{P(X|Y)P(Y)}{P(X)}$$

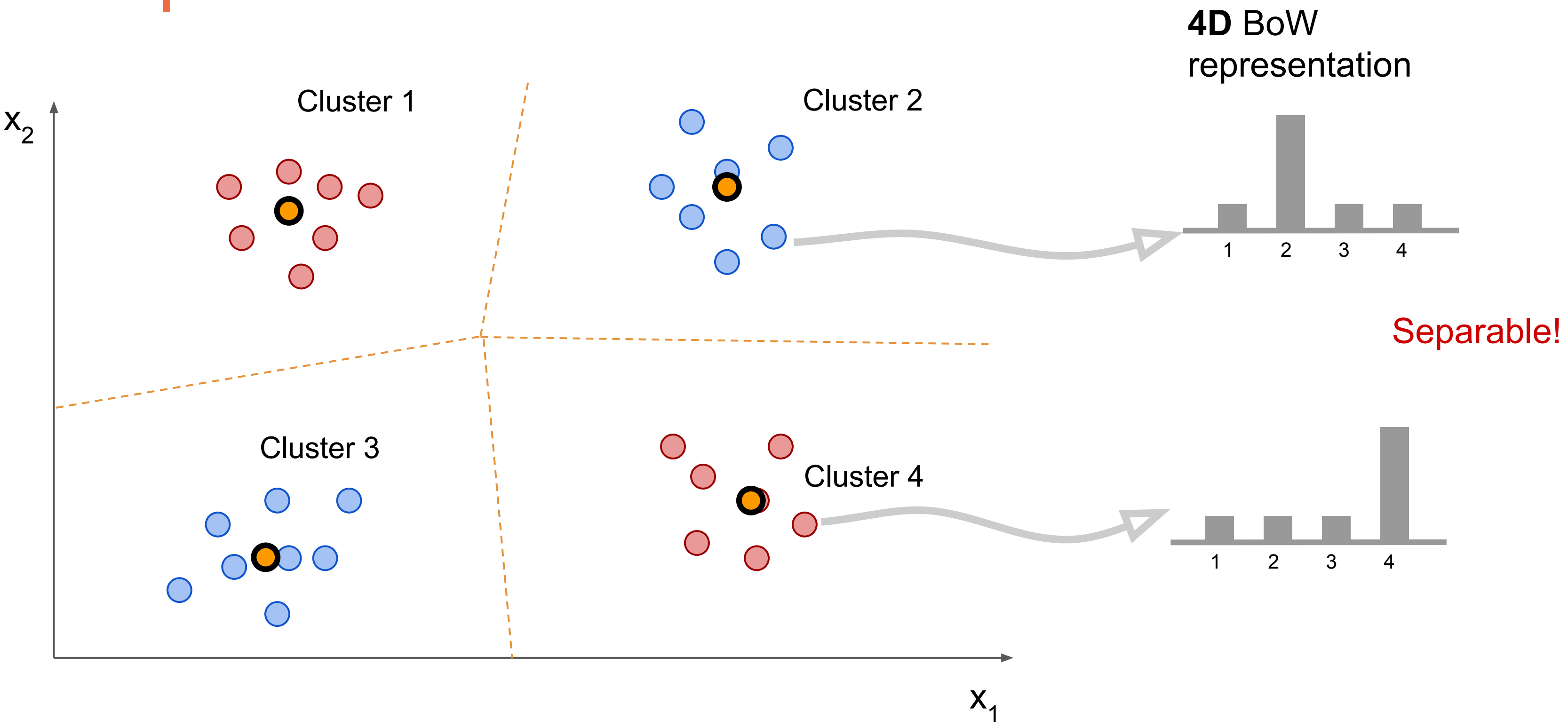
- P(Y|X) depends on P(X|Y) and P(X)
- Knowledge of P(X) can help to predict P(Y|X)
- Good model of P(X) must have Y as an implicit latent variable

Example



Not linearly separable :(

Example



https://github.com/kevinmcguinness/ml-examples/blob/master/notebooks/Semi_Supervised_Simple.ipynb

Assumptions

To model P(X) given data, it is necessary to make some assumptions

"You can't do inference without making assumptions"

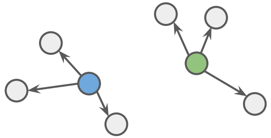
-- David MacKay, Information Theory, Inference, and Learning Algorithms

Typical assumptions:

- Smoothness assumption
    - Points which are close to each other are more likely to share a label.
- Cluster assumption
    - The data form discrete clusters; points in the same cluster are likely to share a label
- **Manifold assumption**
    - The data lie approximately on a manifold of much lower dimension than the input space.

Examples

Smoothness assumption



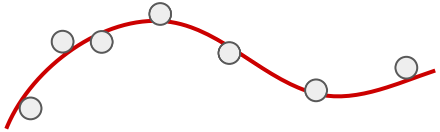
- Label propagation
  - Recursively propagate labels to nearby points
  - Problem: in high-D, your nearest neighbour may be very far away!

Cluster assumption

- Bag of words models
  - K-means, etc.
  - Represent points by cluster centers
  - Soft assignment
  - VLAD
- Gaussian mixture models
  - Fisher vectors

Manifold assumption

- Linear manifolds
  - PCA
  - Linear autoencoders
  - Random projections
  - ICA
- Non-linear manifolds:
  - Non-linear autoencoders
  - Deep autoencoders
  - Restricted Boltzmann machines
  - Deep belief nets



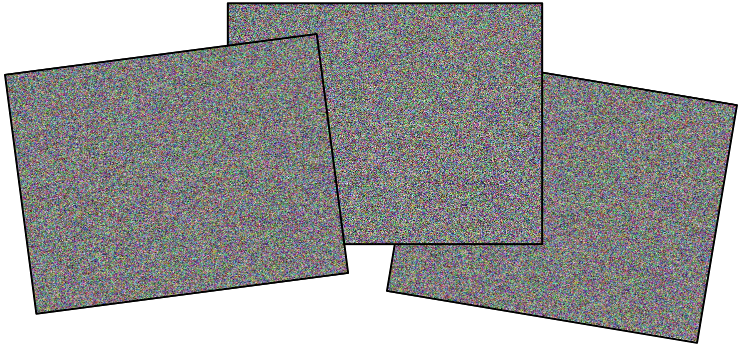
The manifold hypothesis

The data distribution lie close to a low-dimensional manifold

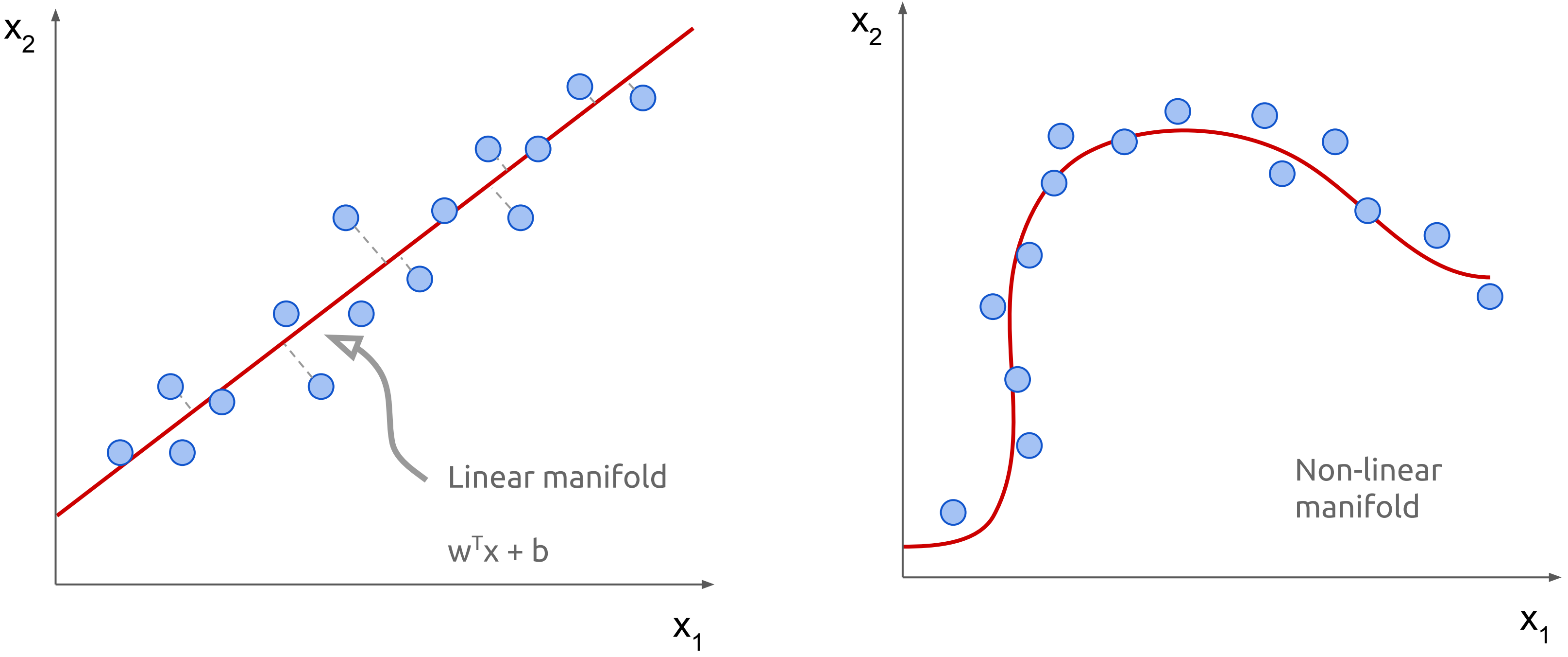
Example: **consider image data**

- Very high dimensional (1,000,000D)
- A randomly generated image will almost certainly not look like any real world scene
  - The space of images that occur in nature is almost completely empty
- Hypothesis: real world images lie on a smooth, low-dimensional manifold
  - Manifold distance is a good measure of similarity

Similar for audio and text



The manifold hypothesis



The Johnson–Lindenstrauss lemma

Informally:

“A small set of points in a high-dimensional space can be embedded into a space of much lower dimension in such a way that distances between the points are nearly preserved. The map used for the embedding is at least Lipschitz continuous.”

Intuition: Imagine threading a string through a few points in 2D

The manifold hypothesis guesses that such a manifold generalizes well to unseen data

Energy-based models

Often intractable to explicitly model probability density

Energy-based model: high energy for data far from manifold, low energy for data near manifold of observed data

Fitting energy-based models

- Push down on area near observations.
- Push up everywhere else.

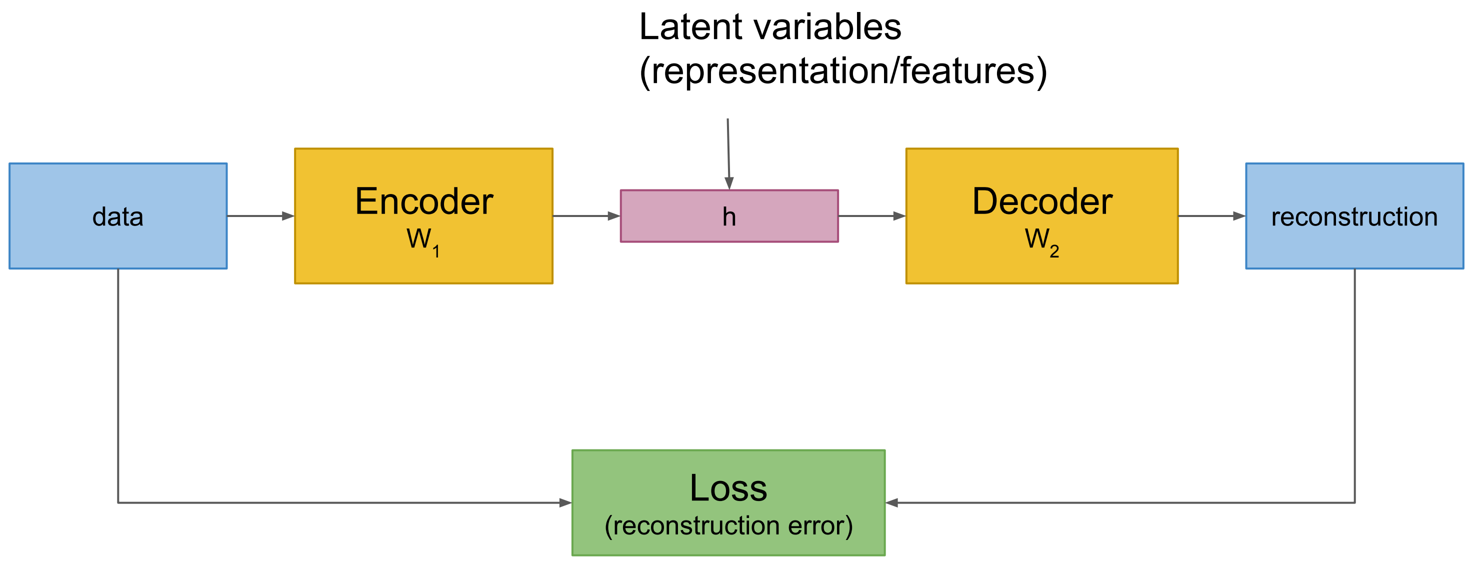
Examples

Encoder-decoder models: measure energy with reconstruction error

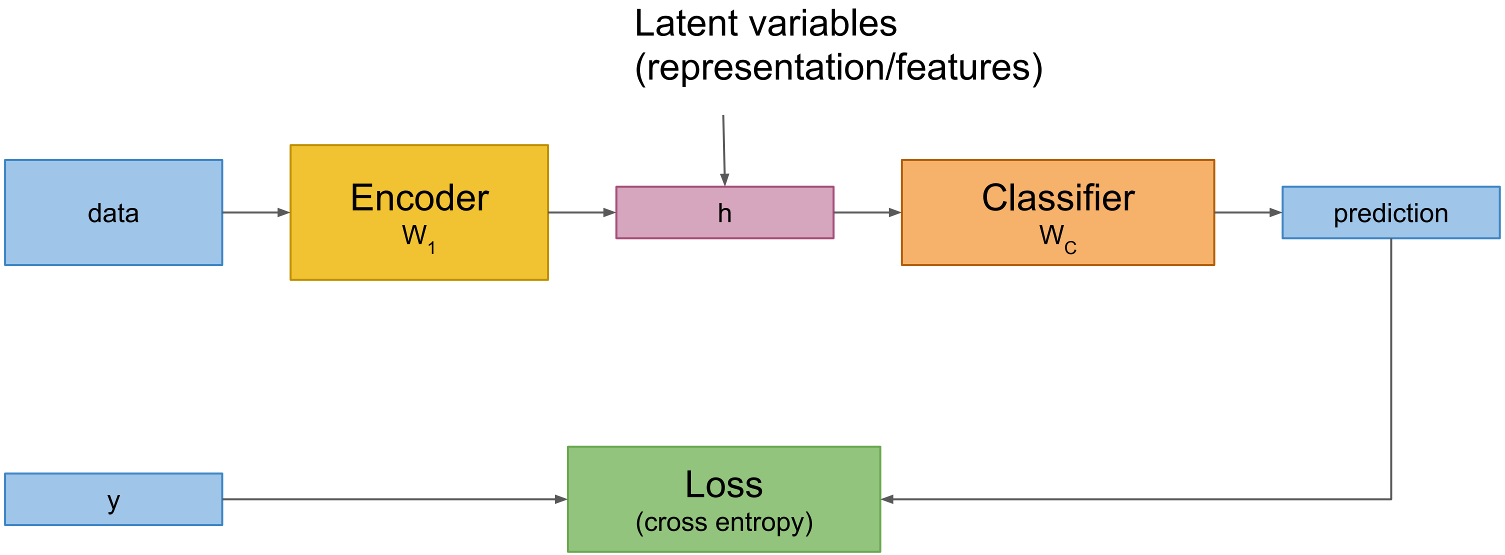
- **K-Means**: push down near prototypes. Push up based on distance from prototypes.
- **PCA**: push down near line of maximum variation. Push up based on distance to line.
- **Autoencoders**: non-linear manifolds...

LeCun et al, **A Tutorial on Energy-Based Learning**, Predicting Structured Data, 2006 http://yann.lecun.com/exdb/publis/pdf/lecun-06.pdf

Autoencoders



Autoencoders



Autoencoders

Need to somehow **push up** on energy far from manifold

- **Standard**: limit the dimension of the hidden representation
- **Sparse autoencoders:** add penalty to make hidden representation sparse
- **Denoising autoencoders**: add noise to the data, reconstruct without noise.

Can **stack** autoencoders to attempt to learn higher level features

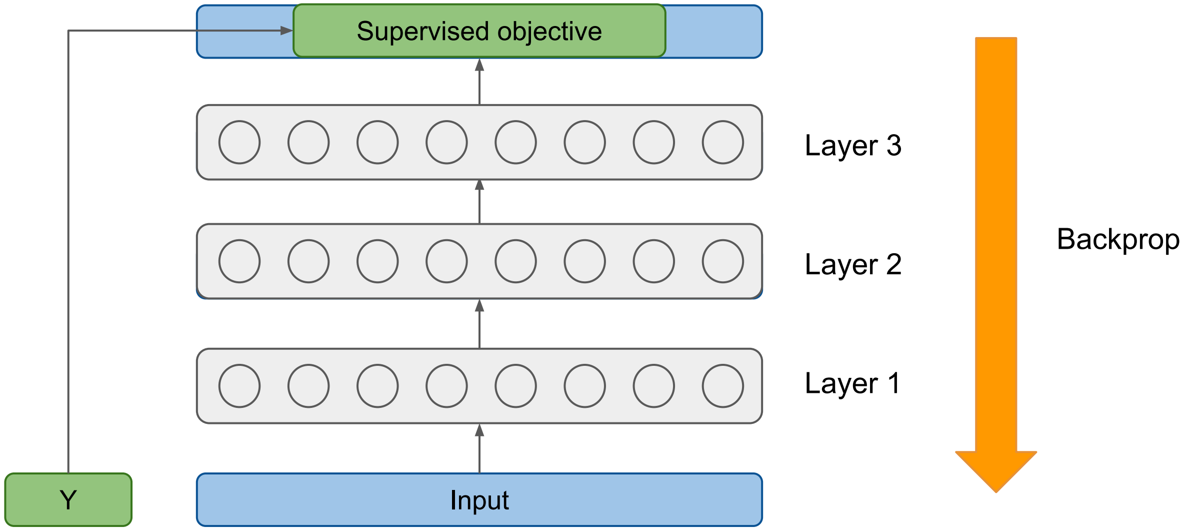
Can train stacked autoencoders by **greedy layerwise training**

Finetune for classification using backprop

Denoising autoencoder example

https://github.com/kevinmcguinness/ml-examples/blob/master/notebooks/DenoisingAutoencoder.ipynb

Greedy layerwise training



Unsupervised learning from video

Slow feature analysis

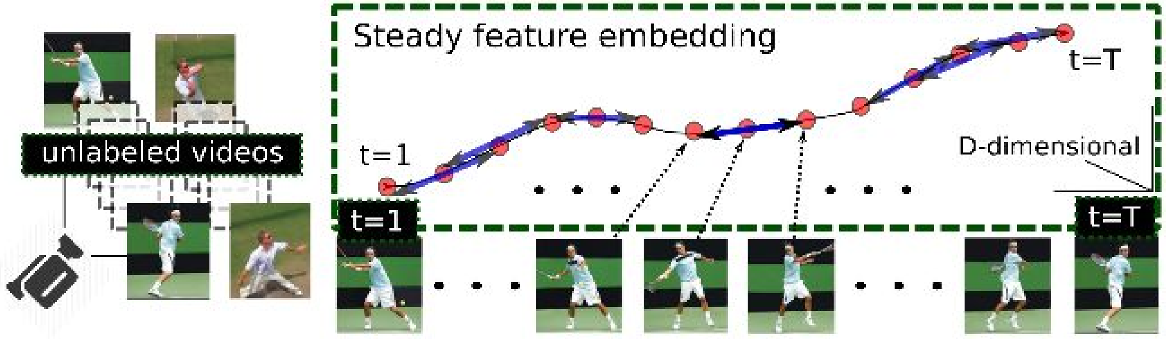
- Temporal coherence assumption: features should change slowly over time in video

Steady feature analysis

- Second order changes also small: changes in the past should resemble changes in the future

Train on triples of frames from video

Loss encourages nearby frames to have slow and steady features, and far frames to have different features



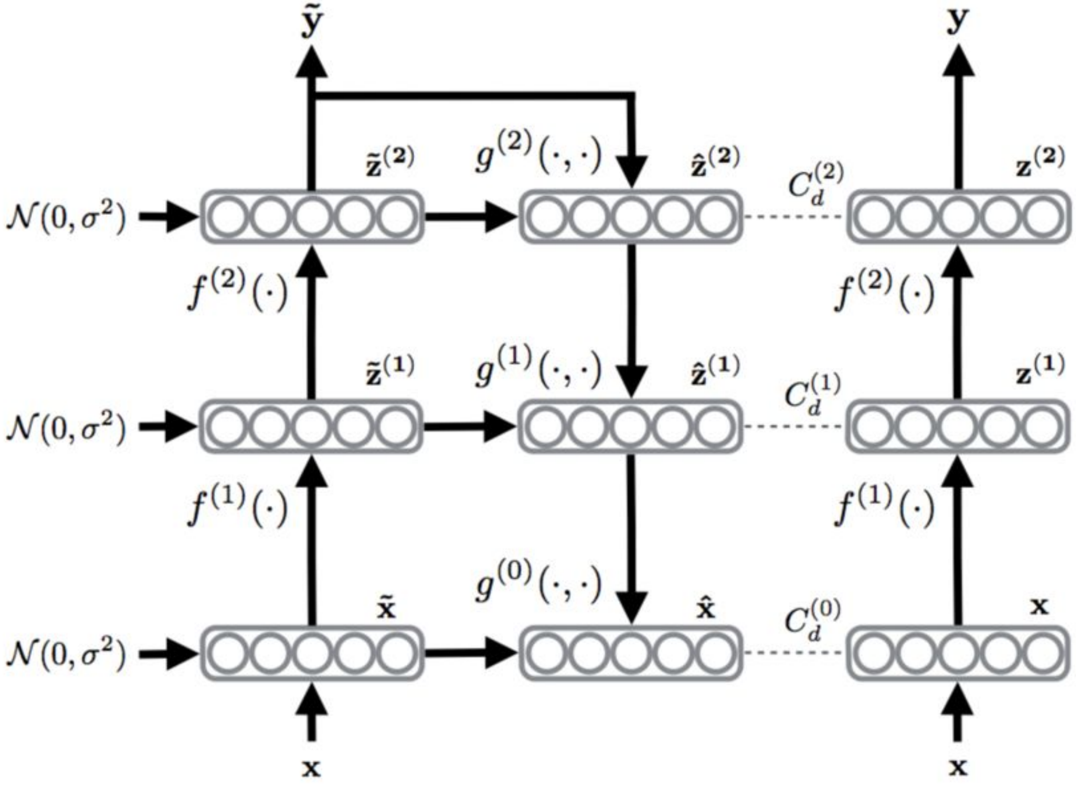
Jayaraman and Grauman. **Slow and steady feature analysis: higher order temporal coherence in video** CVPR 2016. https://arxiv.org/abs/1506.04714

Ladder networks

Combine supervised and unsupervised objectives and train together

- Clean path and noisy path
- Decoder which can invert the mappings on each layer
- Loss is weighted sum of supervised and unsupervised cost

1.13% error on permutation invariant MNIST with only 100 examples



Rasmus et al. **Semi-Supervised Learning with Ladder Networks.** NIPS 2015. http://arxiv.org/abs/1507.02672

Summary

Many methods available for learning from unlabelled data

- Autoencoders (many variations)
- Restricted boltzmann machines
- Video and ego-motion
- Semi-supervised methods (e.g. ladder networks)

Very active research area!