Introduction to Deep Learning

M S Ram

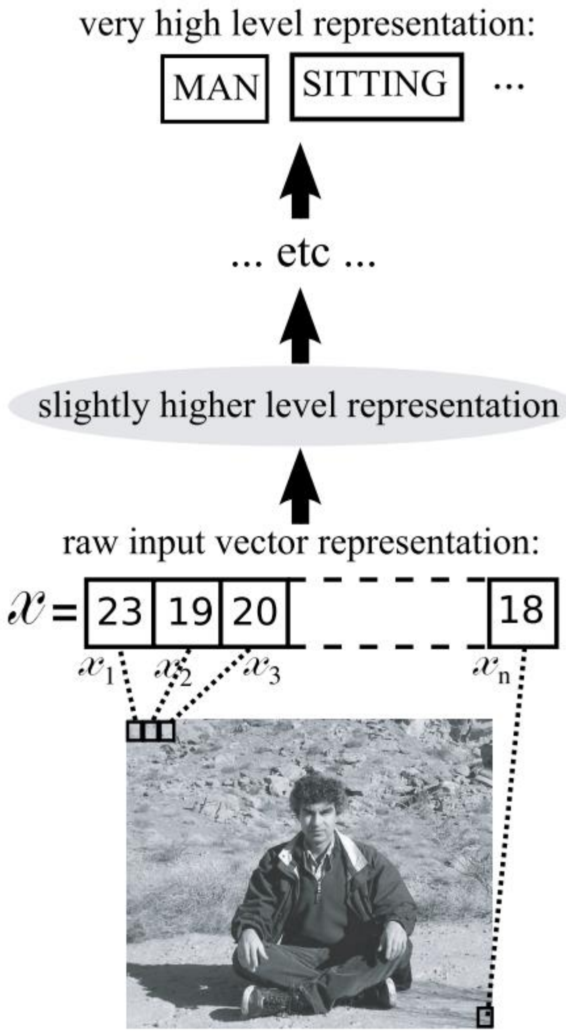
Dept. of Computer Science & Engg.

Indian Institute of Technology Kanpur

Reading of Chap. 1 from "Learning Deep Architectures for AI"; Yoshua Bengio; FTML Vol. 2, No. 1 (2009) 1–127

A Motivational Task: **Percepts → Concepts**

- Create algorithms
  - that can understand scenes and describe them in natural language
  - that can infer semantic concepts to allow machines to interact with humans using these concepts
- Requires creating a series of abstractions
  - Image (Pixel Intensities) → Objects in Image → Object Interactions → Scene Description
- Deep learning aims to automatically learn these abstractions with little supervision



Courtesy: Yoshua Bengio, Learning Deep Architectures for AI

Deep Visual-Semantic Alignments for Generating Image Descriptions (Karpathy, Fei-Fei; CVPR 2015)

“two young girls are playing with lego toy.”

"boy is doing backflip on wakeboard."

"construction worker in orange safety vest is working on road."

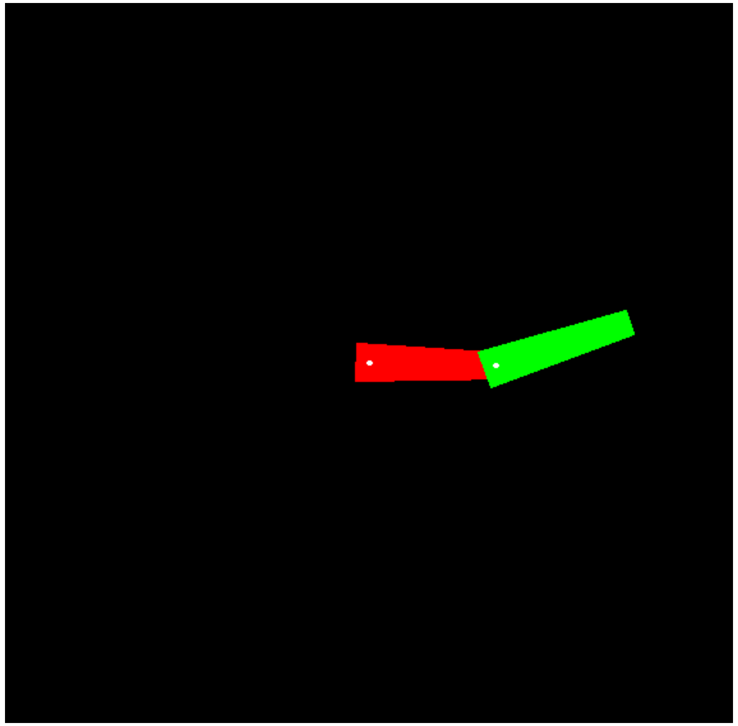
"man in black shirt is playing guitar."

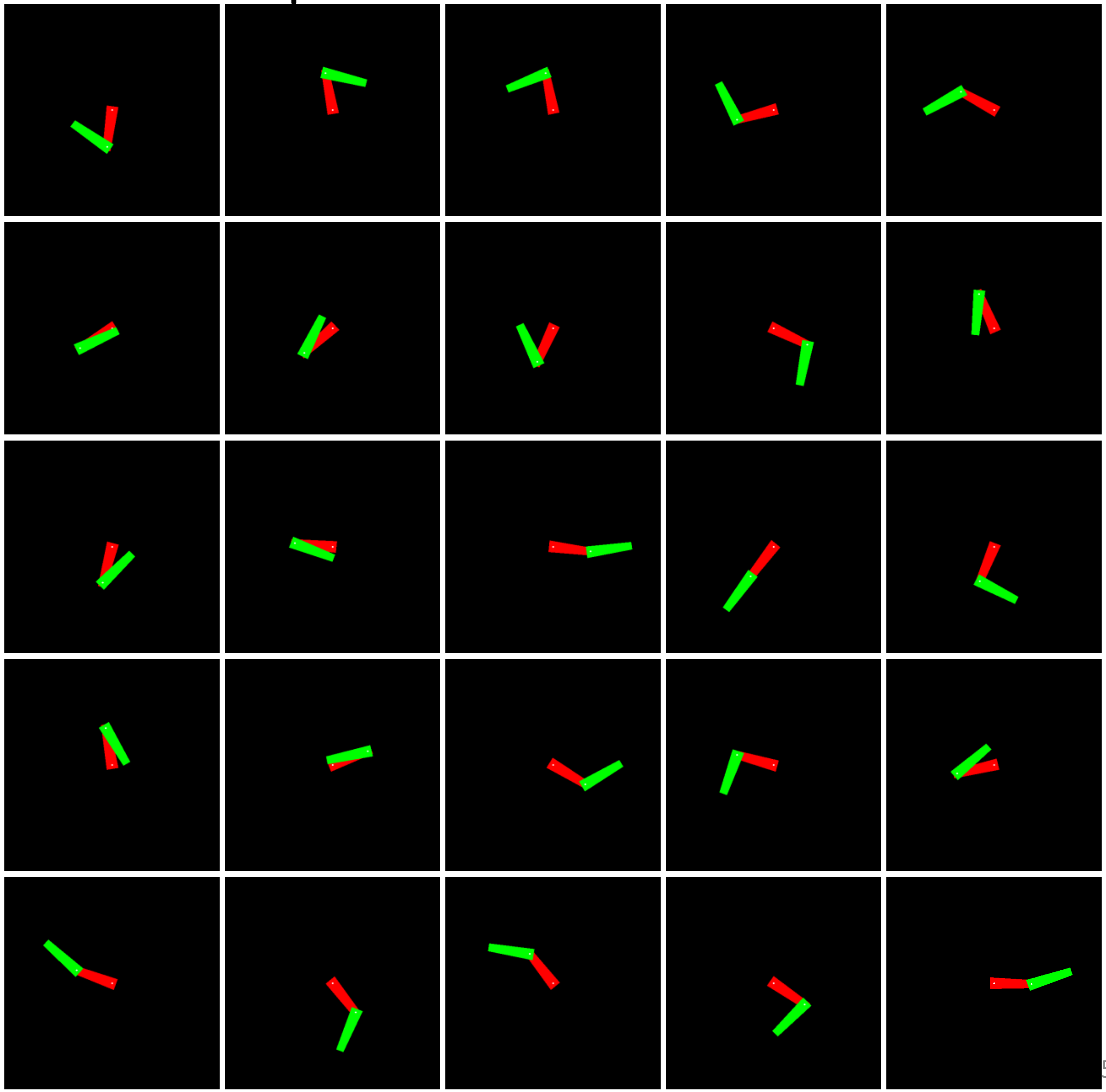
http://cs.stanford.edu/people/karpathy/deepimagesent/

Challenge in Modelling Complex Behaviour

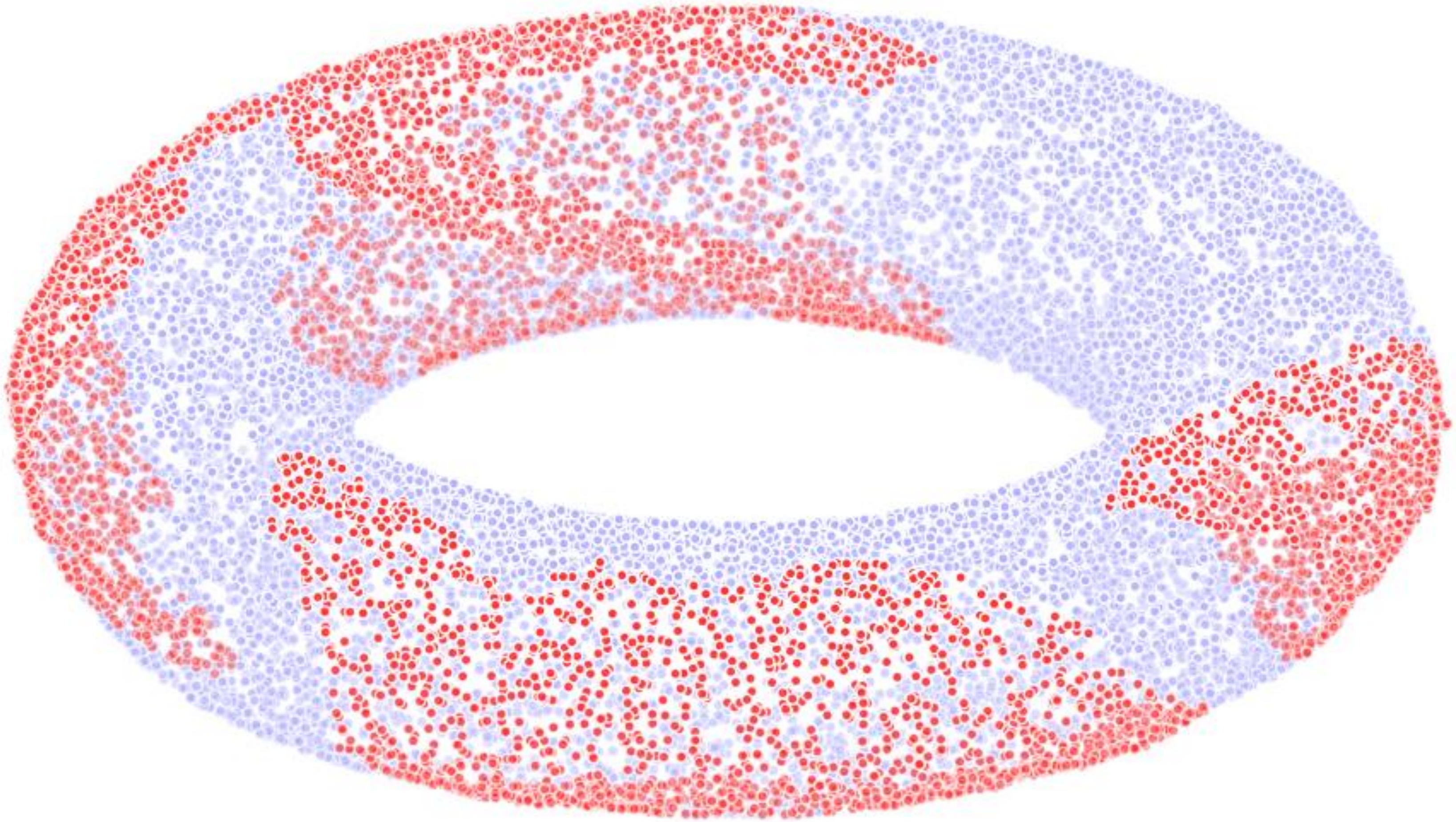
- Too many concepts to learn
  - Too many object categories
  - Too many ways of interaction between objects categories
- Behaviour is a highly varying function underlying factors
  - f: L → V
  - L: latent factors of variation
    - low dimensional latent factor space
  - V: visible behaviour
    - high dimensional observable space
  - f: highly non-linear function

Example: Learning the Configuration Space of a Robotic Arm





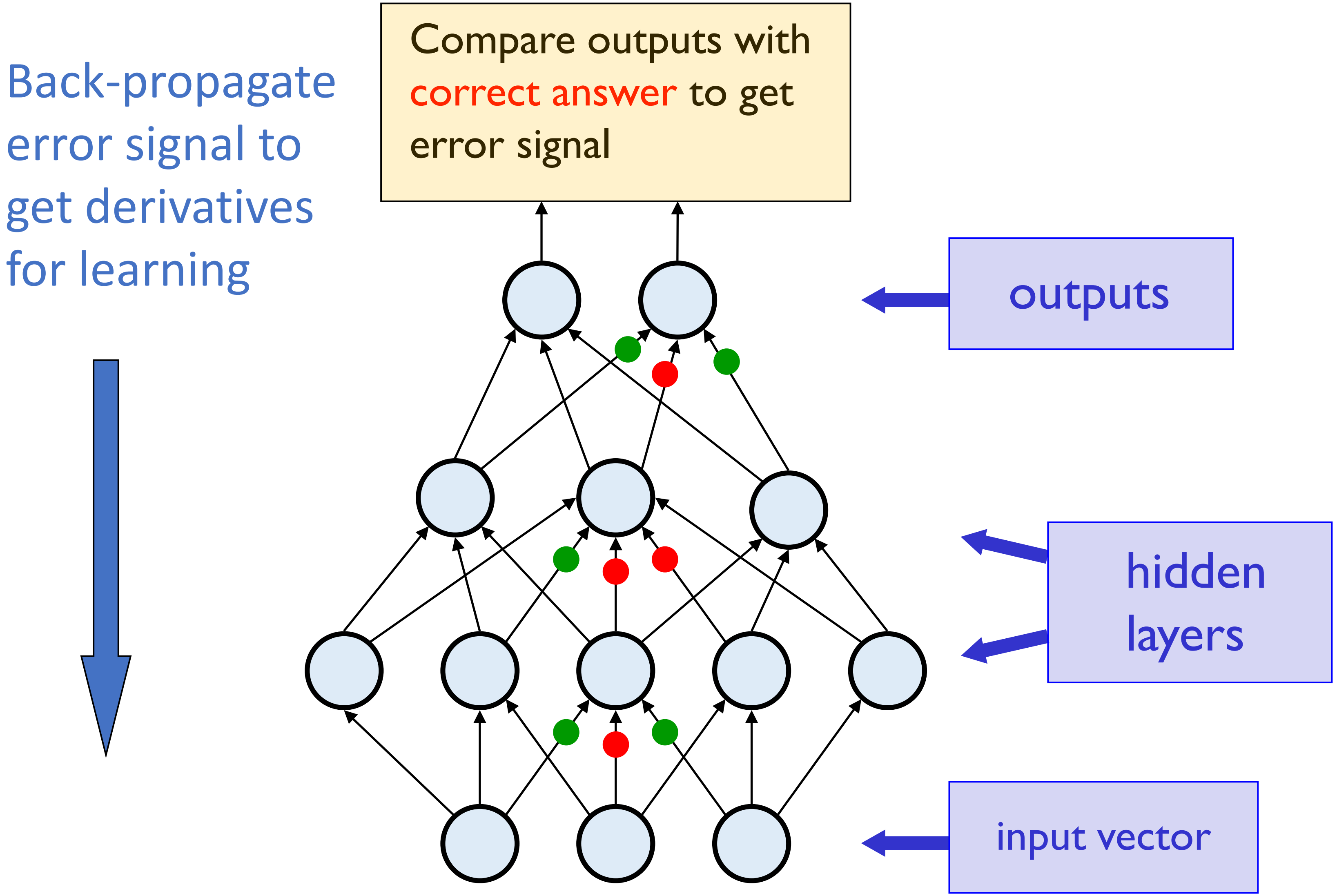
C-Space Discovery using Isomap



How do We Train Deep Architectures?

- Inspiration from mammal brain
- Multiple Layers of “neurons” (Rumelhart et al 1986)
- Train each layer to compose the representations of the previous layer to learn a higher level abstraction
  - Ex: Pixels → Edges → Contours → Object parts → Object categories
  - Local Features → Global Features
- Train the layers one-by-one (Hinton et al 2006)
  - Greedy strategy

Multilayer Perceptron with Back-propagation
First deep learning model (Rumelhart, Hinton, Williams 1986)



Source: Hinton's 2009 tutorial on Deep Belief Networks

Drawbacks of Back-propagation based Deep Neural Networks

- They are discriminative models
  - Get all the information from the labels
    - And the labels don't give so much of information
  - Need a substantial amount of labeled data

- Gradient descent with random initialization leads to poor local minima

Hand-written digit recognition

- Classification of **MNIST** hand-written digits
- 10 digit classes
- Input image: 28x28 gray scale
    - 784 dimensional input

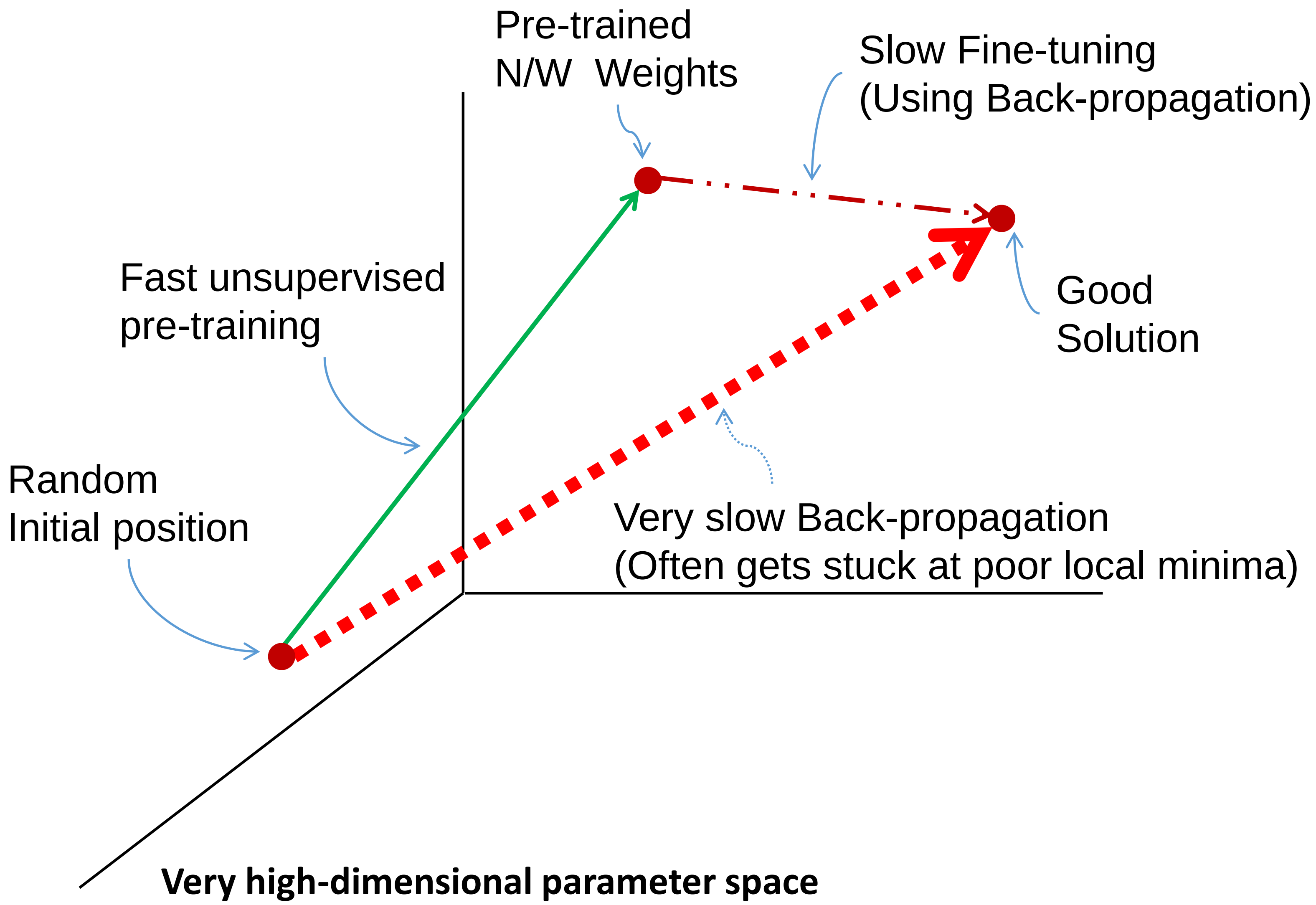
A Deeper Look at the Problem

- One hidden layer with 500 neurons

  => 784 * 500 + 500 * 10

  ≈ **0.4 million weights**
- Fitting a model that best explains the training data is an optimization problem in a 0.4 million dimensional space
- It’s almost impossible for Gradient descent with random initialization to arrive at the global optimum

A Solution – Deep Belief Networks (Hinton et al. 2006)



A Solution – Deep Belief Networks (Hinton et al. 2006)

- Before applying back-propagation, pre-train the network as a series of generative models
- Use the weights of the pre-trained network as the initial point for the traditional back-propagation
  - This leads to quicker convergence to a good solution
- Pre-training is fast; fine-tuning can be slow

Quick Check: MLP vs DBN on MNIST

- MLP (1 Hidden Layer)
    - 1 hour: 2.18%
    - 14 hours: 1.65%

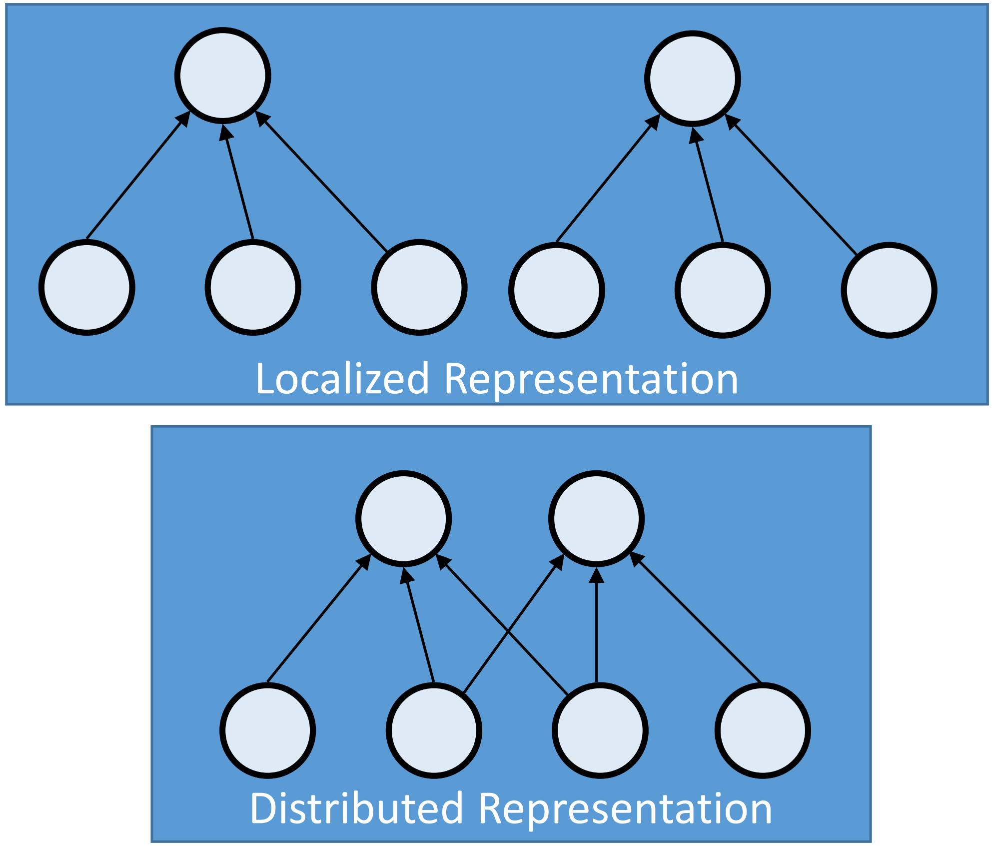
- DBN
    - 1 hour: 1.65%
    - 14 hours: 1.10%
    - 21 hours: 0.97%

Intel QuadCore 2.83GHz, 4GB RAM

MLP: Python :: DBN: Matlab

Intermediate Representations in Brain

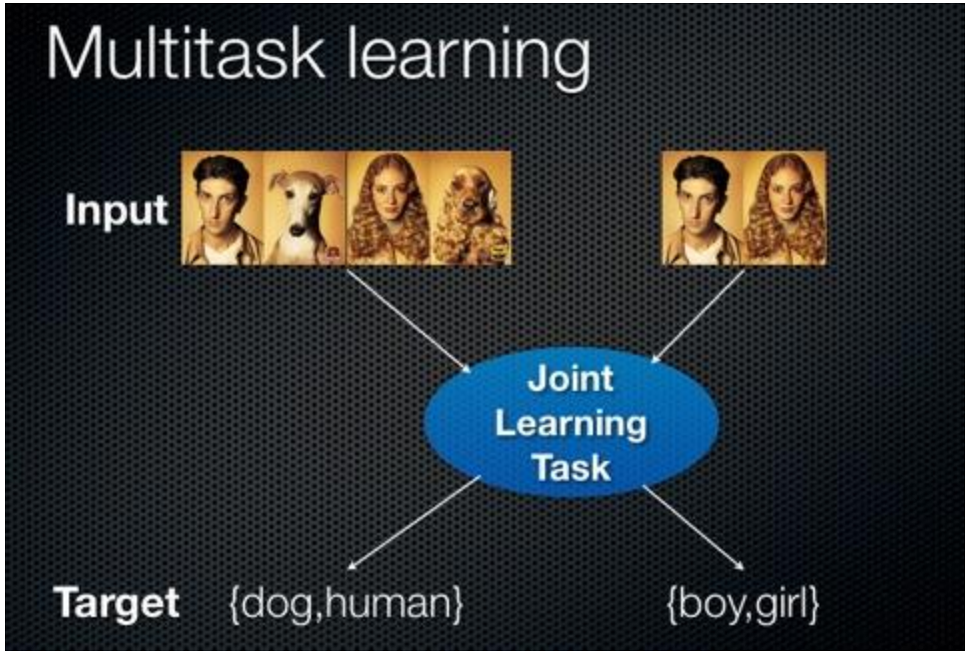
- Disentanglement of factors of variation underlying the data
- Distributed Representations
  - Activation of each neuron is a function of multiple features of the previous layer
  - Feature combinations of different neurons are not necessarily mutually exclusive
- Sparse Representations
  - Only 1-4% neurons are active at a time



Local vs. Distributed in Input Space

- Local Methods
  - Assume smoothness prior
    - $g(x) = f(g(x_1), g(x_2), \ldots, g(x_k))$
    - $\{x_1, x_2, \ldots, x_k\}$ are neighbours of x
  - Require a metric space
    - A notion of distance or similarity in the input space
  - Fail when the target function is highly varying
  - Examples
    - Nearest Neighbour methods
    - Kernel methods with a Gaussian kernel
- Distributed Methods
  - No assumption of smoothness ➔ No need for a notion of similarity
  - Ex: Neural networks

Multi-task Learning



Source: https://en.wikipedia.org/wiki/Multi-task_learning

Desiderata for Learning AI

- Ability to learn complex, highly-varying functions
- Ability to learn multiple levels of abstraction with little human input
- Ability to learn from a very large set of examples
  - Training time linear in the number of examples
- Ability to learn from mostly unlabeled data
  - Unsupervised and semi-supervised
- Multi-task learning
  - Sharing of representations across tasks
- Fast predictions

References

- Primary

  - Yoshua Bengio, **Learning Deep Architectures for AI**, Foundations and Trends in Machine Learning Vol. 2, No. 1 (2009) 1–127
  - Hinton, G. E., Osindero, S. and Teh, Y. **A fast learning algorithm for deep belief nets**. Neural Computation 18 (2006), pp 1527-1554
  - Rumelhart, David E., Geoffrey E. Hinton, and R. J. Williams. **Learning Internal Representations by Error Propagation**. David E. Rumelhart, James L. McClelland, and the PDP research group. (editors), Parallel distributed processing: Explorations in the microstructure of cognition, Volume 1: Foundations. MIT Press, 1986.

- Secondary

  - Hinton, G. E., **Learning Multiple Layers of Representation**, Trends in Cognitive Sciences, Vol. 11, (2007) pp 428-434.
  - Hinton G.E., Tutorial on Deep Belief Networks, Machine Learning Summer School, Cambridge, 2009
  - Andrej Karpathy, Li Fei-Fei. **Deep Visual-Semantic Alignments for Generating Image Descriptions**. CVPR 2015.