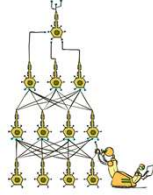


CSE 473: Artificial Intelligence

Multilayer Perceptrons



Steve Tanimoto --- University of Washington

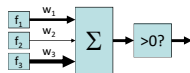
[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at <http://ai.berkeley.edu>.]

How Many Computers to Identify a Cat?

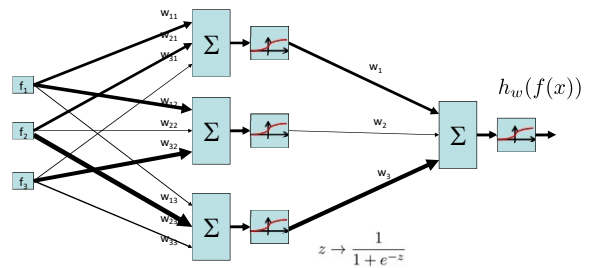


"Google Brain"
[Le, Ng, Dean, et al, 2012]

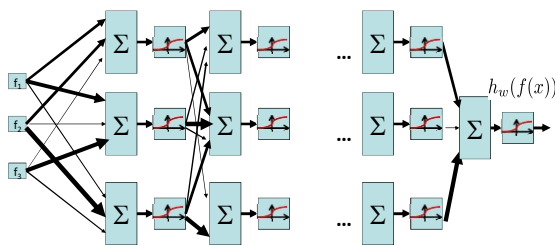
Perceptron



Two-Layer Neural Network

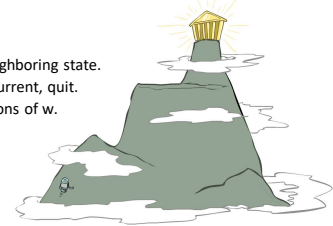


N-Layer Neural Network



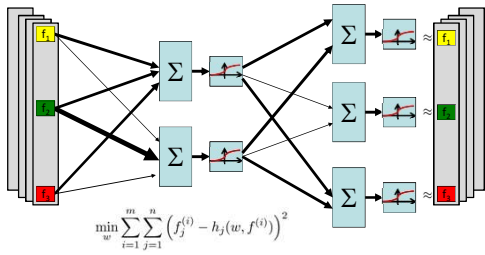
Hill Climbing

- Simple, general idea:
 - Start wherever
 - Repeat: move to the best neighboring state.
 - If no neighbors better than current, quit.
 - Neighbors = small perturbations of w .
- Property
 - Many local optima



-> How to find a good local optimum?

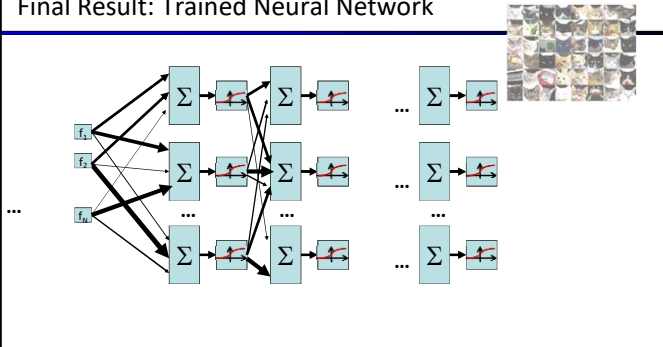
Auto-Encoder (Crude Idea Sketch)



Training Procedure: Stacked Auto-Encoder

- **Auto-encoder**
 - Layer 1 = "compressed" version of input layer
- **Stacked Auto-encoder**
 - For every image, make a compressed image (= layer 1 response to image)
 - Learn Layer 2 by using compressed images as input, and as output to be predicted
 - Repeat similarly for Layer 3, 4, etc.
- **Some details left out**
 - Typically in between layers responses get agglomerated from several neurons ("pooling" / "complex cells")

Final Result: Trained Neural Network



Final Result: Trained Neural Network

