Overview

How to solve hard problems?

• Use a lot of good AND labelled training data
• Use a big deep neural network

→ Success is the only possible outcome

powerful models
learnable models

CSC321: Intro to Machine Learning and Neural Networks, Winter 2016
Michael Guerzhoy

Slides from Ilya Sutskever
Deep Neural Networks

• Can perform a wide range of computation
• Can be learned automatically
  • If you tune them right and use a powerful variant of Stochastic Gradient Descent

• Powerful but not (computer) learnable: Python
  • Can’t make a learning algorithm that takes lots of inputs and outputs and produces Python code that generates the outputs
• Learnable but not powerful:
  • Logistic regression
  • Deep Neural Networks that aren’t deep enough
Why are Deep Nets Powerful

• A single neuron can approximately implement Boolean logic
• So by combining multiple neurons we can perform any computation
Features as Computation

- Can think of every layer of a neural network as one step of a parallel computation
- Features are the functions that are applied to the previous layers
- Learning features $\Leftrightarrow$ Learning what function to apply at step $t$ of the algorithm
Problem: input/output are not of fixed size

• Solution: RNNs can process input of any size, and generate output of any size
The *Deep Learning Hypothesis*

- Human perception is fast
  - Neurons fire at most 100 times a second
  - Humans can solve simple perceptual tasks in 0.1 seconds
    - So out neurons fire in a sequence of 10 times at most

Anything a human can do in 0.1 seconds, a big 10-layer neural network can do, too!
Unsupervised Learning

• Need lots of data to train a very big network
  • Big network, not enough data \(\rightarrow\) overfitting

• Humans seem to be able learn from very little labelled data
  • Parents do point at things and tell babies what they are called, but that’s not how you learn the meaning of most words!

• If you can do unsupervised learning and figure out good features of the data, you need a lot less labelled data
  • The functions/features are fixed, only need to figure out how to combine them
  • Partial solution: transfer learning. Like in Project 3, use features learned in one dataset to classify another dataset
“Success is guaranteed”

• Huge amount of progress in supervised learning in recent years
  • With a large enough network, a large enough labelled training set, and a large enough budget for GPUs, *success is guaranteed*

• Unsupervised learning is still very hard