15.095: Machine Learning under a Modern Optimization Lens

Lecture 11: Deep Learning

Overview

> *"I am really confused. I keep changing my opinion on a daily basis, and I cannot seem to settle on one solid view of this puzzle. No, I am not talking about world politics or the current U.S. president, but rather something far more critical to humankind, and more specifically to our existence and work as engineers and researchers. I am talking about...* **deep learning***."—M. Elad*

Deep learning can often handle complex prediction tasks, but it requires a lot of data.

Logistic Regression

Given features $\mathbf{x} \in \mathbb{R}^p$, predict outcome $y \in \{0, 1\}$.

Build predictive model $f_{\mathbf{w}}(\mathbf{x})$ of $\mathbb{P}(y|\mathbf{x})$ using maximum likelihood and the data $(\mathbf{x}_i, y_i)_{i=1}^n$:

$$\mathbb{P}(y = 1|\mathbf{x}) = f_{\mathbf{w}}(\mathbf{x})$$

Under this setup, likelihood function (to maximize) is

$$L(\mathbf{w}) = \left(\prod_{i:y_i=1} f_{\mathbf{w}}(\mathbf{x}_i) \right) \left(\prod_{i:y_i=0} (1 - f_{\mathbf{w}}(\mathbf{x}_i)) \right).$$

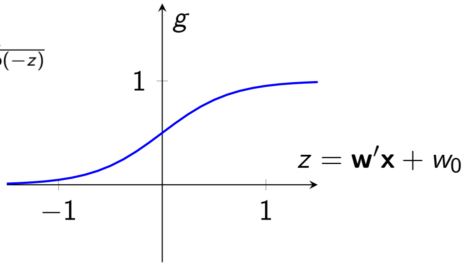
Logistic Regression

Criterion:

$$\min_{\mathbf{w}} (-\log(L)) = -\sum_i \left[y_i \log f_{\mathbf{w}}(\mathbf{x}_i) + (1 - y_i) \log(1 - f_{\mathbf{w}}(\mathbf{x}_i)) \right].$$

Logistic regression focuses on f of the form $f_{\mathbf{w}}(\mathbf{x}) = g(\mathbf{w}'\mathbf{x} + w_0)$, where g is the *logistic function* (sometimes called sigmoid function).

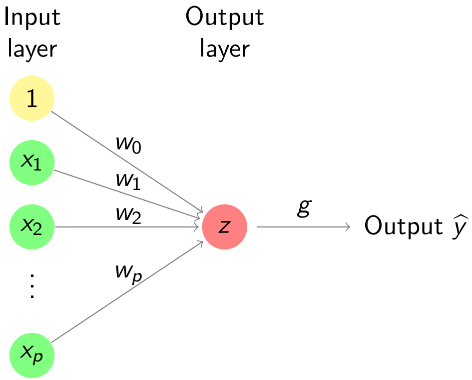
$$g(z) = \frac{\exp(z)}{1+\exp(z)} = \frac{1}{1+\exp(-z)}$$



Leads to classifiers with a "linear decision boundary": predict $y = 1$ if and only if $\widehat{\mathbf{w}}'\mathbf{x} >$ some threshold.

Abstracting Away

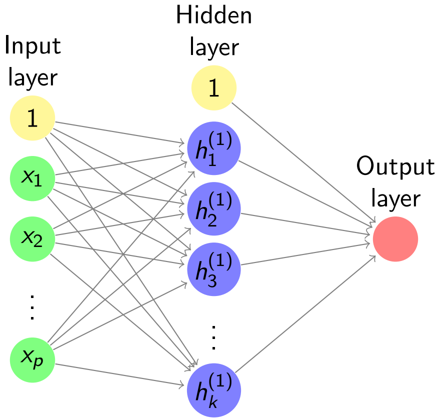
This procedure takes features $\mathbf{x}$, computes some linear combination $(\mathbf{w}'\mathbf{x} + w_0)$, then applies a nonlinear transform (logistic function g) to estimate $\mathbb{P}(y = 1|\mathbf{x})$.



What if we consider intermediate, *hidden* layers?

Neural Networks

This graph describes the network's **architecture**.



This is a **feed-forward neural network**. Note that not all edges from one layer to the next need to be present.

Neural Networks

The values of $h_i^{(1)}$ in the hidden layer are defined by

$$\mathbf{h}^{(1)} = g\left(\mathbf{W}'_{(1)}\mathbf{x} + \mathbf{w}_{(1),0}\right).$$

If you have more than one hidden layer, further layers would be defined similarly (using weighted input from previous layer).

Many choices: network architecture, activation functions, loss function. . .
$\hookrightarrow$ allows for much more complex decision boundaries.

Moment of Reflection

Let's put all of this in context:

$$\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \ell\left(f; \mathbf{x}_i, y_i\right),$$

where $\mathcal{F}$ is

- Linear model
- (Classification and regression) trees
- $f = f_1 + \cdots + f_T$, where f_t is "simple" (boosting)

Neural networks (and variants) fit into the same approach—the only difference is $\mathcal{F}$ (or how the problem is solved).

Algorithms

Given fixed architecture, activation functions, and loss function, how to solve the optimization problem?

Typically procedures like (stochastic) gradient descent are used (*back propagation*), with many random restarts.

Observations:

- Many parameters $\implies$ need large quantity of data for tuning.
- Lots of research to understand why/when neural networks (and algorithms to solve the optimization criteria) work well.

Applications

Deep learning has seen no shortage of news headlines and applications

Applications

Deep learning tends to excel in applications where you can easily access or cheaply create massive quantities of data.

Some applications to discuss today:

- Image recognition
- Healthcare operations

Image Net Challenge

Which of 1,000 categories does a given image fall into?
Data: ~ 1.2 million categorized images (2014 competition)
Task: rank top 5 possible categories

GoogLeNet

GoogLeNet:

- 2014 Challenge's winning approach
- Top-5 list error rate around 7%

GoogLeNet

Image classification tasks often rely on *convolutional neural networks*.

Input image: $\sim 500 \times 500 \times 3$ (RGB) = **750,000 pixel values**
$\hookrightarrow$ hundreds of millions of parameters to train using standard neural nets

Convolutions help to capture inherent structure in images:

130	155	165	180	220
120	150	160	200	210
110	100	120	130	170
75	50	30	80	110
55	75	40	100	150

Filter

×-1	×0	×1
×0	×2	×-1
×-1	×0	×1

$\xrightarrow{\Sigma}$

255	200	240
320	190	60
30	-30	380

$5 \times 5 \mapsto 3 \times 3$ . . . and then combine across colors!

The Good, The Bad, . . .

While neural nets can work extremely well, they can also perform exceptionally poorly at times. . .

Has led to whole area of exploration into this behavior

Example from Goodfellow et al. (2015); image from Image Net.

Visualization

There has been a variety of work on interpreting neural networks.

What if we invert the process?

$$\begin{tikzcd} \text{Image} \arrow[r] & \text{Class} \arrow[l, bend right] \end{tikzcd}$$

Application in Healthcare

Not surprisingly, deep learning is being explored extensively in medical applications... but what about for operational changes in healthcare?

One of the central challenges in hospitals is *bed management* and *capacity planning*.

- Planning for scheduled versus unscheduled arrivals
- Planning for patients already admitted to the hospital

Discharge prediction at MGH

One of the most fundamental questions you can ask: *which patients do we expect will be discharged today?*

Multidisciplinary team from Massachusetts General Hospital and MIT

Primary questions:

- At 5am, predict which patients will go home today.
- Who are the patients most likely to home?
- What are the *barriers* to a patient not being discharged?
- How do you intervene for those patients?

Summary

Deep learning allows for the creation of complex prediction models that can outperform many other competing techniques.

There have been many recent innovative applications, and that list will likely continue to grow with things like IoT.

Keep in mind *decisions* that need to be made—prediction is not the only important thing!

References

1. "Deep learning," Goodfellow, Bengio, and Courville
2. "Inceptionism" and Deep Dream: https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html
3. Image Net: image-net.org
4. "Explaining and harnessing adversarial examples," Goodfellow, Schlens, and Szegedy
5. "Going Deeper with Convolutions," Szegedy et al.
6. Discharge prediction at Mass General Hospital, work in progress by Safavi et al.