KU LEUVEN

Avoiding local minima in Deep Learning: a nonlinear optimal control approach

Midterm presentation

Jan Scheers
Promotor: Prof. Panos Patrinos
Supervisors: Masoud Ahookhosh, Andreas Themelis

December 2018

1 Outline

1 Neural Networks and Deep Learning

- Very popular these days
- Very expressive / low bias
- Good for big data sets/unlabeled data sets
- Can detect complex nonlinear relationships
- Focus on deep feedforward neural networks in this thesis

1 Neural Network

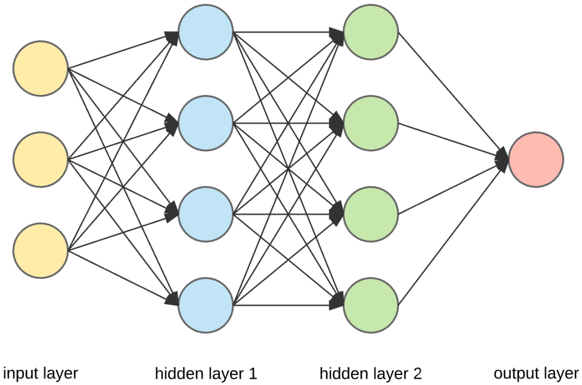


Figure: Feedforward Deep Neural Network. (Retrieved from https://towardsdatascience.com, 2018)

1 Neuron

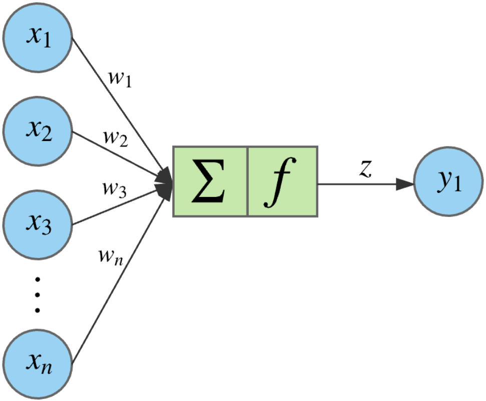


Figure: Single Neuron - McCulloch–Pitts model. (Retrieved from https://towardsdatascience.com, 2018)

1 Activation Function

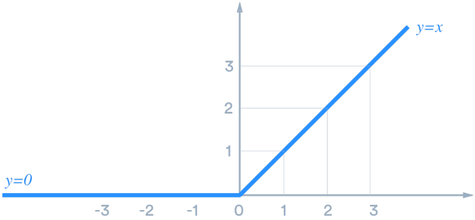
- McCulloch–Pitts neuron model

$$y = \sigma(w_1x_1 + w_2x_2 + ... + w_nx_n)$$

- Rectified Linear Unit (ReLU)

$$\sigma(x) = x^+ = \max(0, x)$$

- Most commonly used activation function



1 Training

- Matrix notation of network

$$f(W,x) = W_H\sigma(W_{H-1}\sigma(...W_1\sigma(W_0x)...))$$

- Training of neural network optimizes cost function for dataset

$$\underset{W}{\text{minimize}} \quad L(W) = \sum_{j=0}^{N} ||f(W,x^j) - y^j||^2$$

- Usual algorithm is backpropagation
  - Calculate output of network
  - Propagate error backwards
  - Calculate gradient

2 Outline

2 Neural Network as Dynamical System

- Matrix notation of network

$$f(W,x) = W_H \sigma(W_{H-1}\sigma(...W_1\sigma(W_0 x)...))$$

- As a dynamical system

$$\begin{aligned} x_0 &= x \\ x_{k+1} &= \sigma(W_k x_k), \quad k = 0, ..., H-1 \\ y &= W_H x_H \end{aligned}$$

- Every layer is a state
- Weight matrices are inputs

2 Linear Neural Networks

- Matrix notation

$$\underset{W}{\text{minimize}} \quad L(W) = ||W_H W_{H-1} \dots W_1 X - Y||^2$$

- Proof exists that these networks have no bad local minima (Kawaguchi, 2017)[2]
- System Equations

$$\begin{aligned} x_0 &= x \\ x_{k+1} &= W_k x_k, \quad k = 0, ..., H-1 \\ y &= W_H x_H \end{aligned}$$

- Objective: recreate proof using dynamical system interpretation

2 Linear Neural Network Proof Problems

- Even though network is linear, system is nonlinear
- Inputs and states are not separated
- Understood previous proof
  - Did not help much in formulating new proof
- System Equations

$$\begin{aligned} x_0 &= x \\ x_{k+1} &= W_k x_k, \quad k = 0, ..., H-1 \\ y &= W_H x_H \end{aligned}$$

$\Rightarrow$ Change objective to training networks using Control Theory

3 Outline

KU LEUVEN

3 Training as Optimal Control Problem

- Training a network with ReLU activation functions is equivalent to following Optimal Control Problem (OCP)

$$\underset{W}{\text{minimize}} \quad \sum_{j=0}^{N} ||W_H x_H^j - y^j||^2$$

$$\text{subject to} \quad x_{k+1}^j = \max(W_k x_k^j, 0), \quad k = 0, \dots, H-1, j = 1, \dots, N$$

3 Transforming ReLU Constraints

$$x_{k+1}^j = \max(W_k x_k^j, 0)$$
$$\Updownarrow$$
$$x_{k+1}^j = -\min(-W_k x_k^j, 0)$$
$$\Updownarrow$$
$$\min(x_{k+1}^j - W_k x_k^j) = 0$$
$$\Updownarrow$$
$$(x_{k+1}^j - W_k x_k^j)^\top x_{k+1}^j = 0,$$
$$x_{k+1}^j \geq 0, x_{k+1}^j - W_k x_k^j \geq 0$$

$\Rightarrow$ Now constraint function is smooth

3 Solving the OCP

- Problem formulation with relaxed constraints

$$
\begin{aligned}
\underset{W}{\text{minimize}} \quad & \sum_{j=0}^{N} ||W_H x_H^j - y^j||^2 \\
\text{subject to} \quad & (x_{k+1}^j - W_k x_k^j)^\top x_{k+1}^j \leq 0, \qquad k = 0, \ldots, H-1, j = 1, \ldots \\
& x_{k+1}^j \geq 0, x_{k+1}^j - W_k x_k^j \geq 0, \quad k = 0, \ldots, H-1, j = 1, \ldots
\end{aligned}
$$

KU LEUVEN

3 Solving the OCP

- Two ways of transforming OCP to standard optimization problem
- Sequential approach: eliminate states using dynamics
  - This is Backpropagation algorithm
  - Standard approach for training networks
- Simultaneous approach: Keep states as variables, dynamics as constraints
  - Often works better for highly nonlinear problems
  - Novel approach for Neural Networks
  - Topic of thesis

3 Penalty Method

- Penalty method is simplest
- Take constrained problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, \ldots, m \end{aligned}$$

- Solve series of unconstrained problems

$$\min_x \quad \Phi_k(x) = f(x) + \sigma_k \sum_i \max(0, g_i(x))^2$$

- In each iteration increase size of penalty parameter σ_k

3 Augmented Lagrangian Method

- Similar to penalty method
- For constrained problem

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, \ldots, m \end{aligned}$$

- Add Langrange multipliers
- Solve series of unconstrained problems

$$\min_{x} \quad \Phi_k(x) = f(x) + \frac{\sigma_k}{2} \sum_i \max(0, g_i(x))^2 - \sum_i \lambda_i g_i(x)$$

- In each iteration increase size of penalty parameter σ_k and update Lagrange parameters λ_i

3 Comparison to Backpropagation

- Including states as variables increases problem size greatly
- Problem becomes less nonlinear
- In Optimal Control Problems simultaneous approach usually better then sequential approach when problem is highly nonlinear
- Might be easier to parallellize

4 Outline

4 Planning of Future Work

- Explore Penalty Method and Augmented Lagrangian Method compared to Backpropagation
- In Matlab
- If promising, switch to language such as C++/Fortran
- Maybe explore other optimization methods
- Explore Online/Offline training
- ...

5 Outline

5 References

D. P. Bertsekas.
Nonlinear programming.
Athena scientific, Belmont, 2nd ed. edition, 1999.

H. Lu and K. Kawaguchi.
Depth creates no bad local minima.
CoRR, abs/1702.08580, 2017.

J. Nocedal and S. J. Wright.
Numerical optimization.
Springer series in operations research and financial engineering.
Springer, New York (N.Y.), 2nd ed. edition, 2006.

Questions?