From Curriculum Learning to Mollifying Networks

June 24, ICML’2016, Optimization Workshop
Yoshua Bengio
CIFAR Senior Fellow and Program Co-Director
Montreal Institute for Learning Algorithms
Université de Montréal
Busting the myth of local minima

• There are still some researchers who believe that because of the presence of local minima, neural nets should be replaced by kernel machines (Liu, Lee & Jordan, ICML’2016)

• Yet, mounting evidence that this is a myth, for non-tiny nets:
  • (Pascanu, Dauphin, Ganguli, Bengio, arXiv May 2014): On the saddle point problem for non-convex optimization
  • (Dauphin, Pascanu, Gulcehre, Cho, Ganguli, Bengio, NIPS’ 2014): Identifying and attacking the saddle point problem in high-dimensional non-convex optimization
  • (Choromanska, Henaff, Mathieu, Ben Arous & LeCun AISTATS’2015): The Loss Surface of Multilayer Nets
  • (Daniel Soudry, Yair Carmon, arXiv:1605.08361): No bad local minima: Data independent training error guarantees for multilayer neural networks
Saddle Points

- Local minima dominate in low-D, but saddle points dominate in high-D.
- Most local minima are close to the bottom (global minimum error) when the network becomes large.

![Graph showing the relationship between train error and the index of critical point α.](image)
Low Index Critical Points

Choromanska et al & LeCun 2014, ‘The Loss Surface of Multilayer Nets’
Shows that deep rectifier nets are analogous to spherical spin-glass models
The low-index critical points of large models concentrate in a band just above the global minimum, as the number of hidden units increases
Saddle Points During Training

- Oscillating between two behaviors:
  - Slowly approaching a saddle point
  - Escaping it
Yet, training deep / recurrent architectures can be challenging!
Effect of Initial Conditions in Deep Nets with Tanh Units

- (Erhan et al 2009, JMLR)
- Supervised deep net (tanh), with or w/o unsupervised pre-training → very different solutions in fn space

Neural net trajectories in function space, visualized by t-SNE

No two training trajectories end up in the same place → huge number of effective local minima

w/o unsupervised pre-training

with unsupervised pre-training
Guided Training, Intermediate Concepts

• In (Gulcehre & Bengio ICLR’2013) we set up a task that seems almost impossible to learn by shallow nets, deep nets, SVMs, trees, forests, boosting etc

• Breaking the problem in two sub-problems and pre-training each module separately, then fine-tuning, nails it

• Need prior knowledge to decompose the task

• Guided pre-training allows to find much better solutions, escape effective local minima

HINT
Curriculum Learning

Guided learning helps training humans and animals

Start from simpler examples / easier tasks (Piaget 1952, Skinner 1958)
Order & Selection of Examples Matters

(Bengio, Louradour, Collobert & Weston, ICML’2009)

- Curriculum learning
  - (Bengio et al 2009, Krueger & Dayan 2009)
- Start with easier examples
  - Faster convergence to a better local minimum in deep architectures
Curriculum Learning as a Continuation Method

Target objective

Heavily smoothed objective = surrogate criterion

Final solution

Track local minima

Easy to find minimum
Modern Uses of Curriculum Learning in Difficult to Train Architectures

Appropriately Injecting Noise to Improve Training

Noisy Activation Functions, Gulcehre, Moczulski, Denil & Bengio, 2016
Mollifying Networks, Gulcehre, Moczulski& Bengio, 2016 (submitted)

- Injecting noise is like smoothing the objective function, which is easier to optimize (in the limit of high smoothing: convex)

\[ \mathcal{L}_K(\theta) = (\mathcal{L} * K)(\theta) \]
\[ = \int_C \mathcal{L}(\theta - \xi)K(\xi)d\xi \]

- and easy stochastic gradient

\[ \frac{\partial \mathcal{L}_K(\theta)}{\partial \theta} \approx \frac{1}{N} \sum_{i=1}^{N} \frac{\partial \mathcal{L}(\theta - \xi_i)}{\partial \theta} \]
Gradually Decreased Noise = Continuation Method or Annealing

- Start with high noise and gradually converge to the target (noise-free) objective, trying to track the local minimum along the way.
Noisy Activation Functions

- Adding noise where the nonlinearity would saturate to enhance exploration
Noisy Activation Functions

- Improving the ‘Learning to Execute’ architecture
Injecting noise to make the objective function have a single global minimum

- Initially, high noise: layer = linear and is skipped
- Finally, low noise: layer = tanh or sigmoid, not skipped, very useful for gates in LSTM&GRU
Annealed Stochastic Depth

- With some probability $p$, each layer’s output is just the output of the previous layer.
- That probability $p$ is annealed down at the same time as the noise injection level.

\[ b_i^l \sim \text{Bin}(p) \]

\[ h_i^l = b_i^l h_i^{l-1} + (1 - b_i^l) \tilde{h}_i^l \]

- Where $\tilde{h}_i^l$ is the normal non-linear update of the layer.
- Thus the effective depth is gradually increased during training.
Results

- **Difficult to optimize:** parity, pentomino, very deep nets

- **Neural Turing Machine**

- **LSTM language models:** (PennTreeBank)

<table>
<thead>
<tr>
<th>Model</th>
<th>Valid ppl</th>
<th>Test ppl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisy LSTM + Normal Noise</td>
<td>111.7</td>
<td><strong>108.0</strong></td>
</tr>
<tr>
<td>Noisy LSTM + Half Normal Noise</td>
<td>112.6</td>
<td><strong>108.7</strong></td>
</tr>
<tr>
<td>LSTM (Reference)</td>
<td>119.4</td>
<td>115.6</td>
</tr>
</tbody>
</table>
Noisy Activation Functions for NMT

- Beating the SOTA on Chinese-to-English
- 3.45 BLEU score improvement (including changes in architecture)
Conclusions

• Training deep nets may be easier than believed previously, and local minima may not be the main issue

• But optimization can remain an important challenge

• Curriculum learning and other continuation or annealing methods have become a common tool

• Injecting noise = smoothing the objective

• Specific forms of noise injection (in particular noisy activation functions) and annealing seem to be greatly helpful for a variety of deep learning tasks
MILA: Montreal Institute For Learning Algorithms