Parameter Initialization Strategies

Sargur N. Srihari
srihari@cedar.buffalo.edu

Topics

- Importance of Optimization in machine learning
- How learning differs from optimization
- Challenges in neural network optimization
- Basic Optimization Algorithms
- Parameter initialization strategies
- Algorithms with adaptive learning rates
- Approximate second-order methods
- Optimization strategies and meta-algorithms

Types of Initialization

1. Non-iterative optimization requires no initilization
   – Simply solve for solution point
2. Iterative but converge regardless of initialization
   – Acceptable solutions in acceptable time
3. Iterative but affected by choice of Initialization
   – Deep learning training algorithms are iterative
      - Initialization determines whether it converges at all
      - Can dtermine how quickly learning converges

Modern Initialization Strategies

- They are simple and heuristic
- Based on achieving nice properties
- But problem is a difficult one
  - Some initial points are beneficial for optimization but detrimental to generalization

Known property: Break Symmetry

- Only property known with certainty: Initial parameters must be chosen to break symmetry
- If two hidden units have the same inputs and same activation function then they must have different initial parameters
- Usually best to initialize each unit to compute a different function
- This motivates use random initialization of parameters

Choice of biases

- Biases for each unit are heuristically chosen constants
- Only the weights are initialized randomly
- Extra parameters such as conditional variance of a prediction are constants like biases

Weights drawn from Gaussian

- Weights are almost always drawn from a Gaussian or uniform distribution
  – Choice of Gaussian or uniform does not seem to matter much but not studied exhaustively
- Scale of the initial distribution does have an effect on outcome of optimization and ability to generalize
  – Larger initial weights will yield stronger symmetry-breaking effect, helping avoid redundant units
  – Too large may result in exploding values

Heuristics for initial scale of weights

- One heuristic is to initialize the weights of a fully connected layer with N_{in} inputs and N_{out} outputs by sampling each weights from $\mathrm{Uniform}(-r, r)$ where $r = \frac{1}{\sqrt{N_{in}}}$
- Another heuristic is normalized initiation with

$$r = \sqrt{\frac{6}{N_{in} + N_{out}}}$$

  – Which is a compromise between the goal of initializing all layers to have the same *activation* variance and the goal of having all layers having the same *gradient* variance

Initialization for the biases

- Bias settings must be coordinated with setting weights
- Setting biases to zero is compatible with most weight initialization schemes
- Situations for nonzero biases:
  - Bias for an output unit: initialize to obtain right marginal statistics for output
    - Set bias to inverse of activation function applied to the marginal statistics of the output in the training set
  - Choose bias to causing too much saturation at initialization