Multiple Layered Neural Networks

Sattiraju Prabhakar
CS898R: MultiLayeredNN
Wichita State University

Topics

- Non-linear classification tasks
- Sigmoid Threshold Unit
- Backpropagation Algorithm

Non-Linear Classification Tasks

Classification of Phonemes
Hand-written Digit Recognition

- 3-nearest-neighbor = 2.4% error
- 400-300-10 unit MLP = 1.6% error
- LeNet: 768-192-30-10 unit MLP = 0.9%

Sigmoid Threshold Units

Sigmoid Unit

\[ s_w = \sum_{j} w_j x_j \]
\[ \sigma = \sigma(s_w) = \frac{1}{1 + e^{-s_w}} \]

Sigmoid Function
**Sigmoid Unit**

\[ \sigma(x) = \frac{1}{1 + e^{-x}} \]

Nice property: \( \frac{d\sigma(x)}{dx} = \sigma(x)(1 - \sigma(x)) \)

We can derive gradient decent rules to train
- One sigmoid unit
- *Multilayer networks* of sigmoid units → Backpropagation

**Multi-layer Sigmoid Units**

**MultiLayer Network of Sigmoid Units**

![MultiLayer Network of Sigmoid Units](image)

**Multi-layered Network**

Layers are usually fully connected; numbers of hidden units typically chosen by hand

- Output units \( a_\text{out} \)
- Hidden units \( a_h \)
- Input units \( a_i \)
Example of Feed-forward Network

Learning Problem

- To find $w$ (vector)
- Hypothesis space:
  - Defined by all possible weights for all possible units in the network
- The error surface is similar to linear units, defined by $E$ (on slide 13)
- Gradient descent is used to attempt to find a hypothesis to minimize $E$.

Backpropagation Algorithm

- Error over all of the network output units:
  \[
  E(w) \equiv \frac{1}{2} \sum_{d \in D} \sum_{k \in \text{outputs}} (t_{kd} - o_{kd})^2
  \]
  - $E(w)$: Error function
  - $t_{kd}$: Target values associated with $k^{th}$ output unit and training example $d$
  - $o_{kd}$: Output value associated with $k^{th}$ output unit and training example $d$

Explanation of Terms

- A node is either an input to the network or the output of some unit
- An index is assigned to each node in the network
- $x_i$ (or $x_{id}$) denotes the input from node $i$ to unit $j$
- $w_{ij}$ (or $w_{ij}$) denotes the weight associated with connection between node $i$ and unit $j$
- $\delta_n$ denotes the error term associated with unit $n$
- Each training example is $<x, t>$, both are vectors.
### Backpropagation Algorithm

Initialize all weights to small random numbers.

Until satisfied, Do
- For each training example, Do
  1. Input the training example to the network and compute the network outputs
  2. For each output unit $k$
     \[ \delta_k = o_k(1 - o_k)(t_k - o_k) \]
  3. For each hidden unit $h$
     \[ \delta_h = o_h(1 - o_h) \sum_{k \in \text{next layer}} w_{h,k} \delta_k \]
  4. Update each network weight $w_{i,j}$
     \[ w_{i,j} = w_{i,j} + \Delta w_{i,j} \]
     where
     \[ \Delta w_{i,j} = \eta \delta_i x_{i,j} \]

### Backpropagation Learning

At each epoch, sum gradient updates for all examples and apply

Usual problems with slow convergence, local minima

### Properties of Backpropagation

- Gradient descent over entire network weight vector
- Easily generalized to arbitrary directed graphs
- Will find a local, not necessarily global error minimum
  - In practice, often works well (can run multiple times)
- Often include weight momentum $\alpha$
  \[ \Delta w_{i,j}(t) = \eta \delta_i x_{i,j} + \alpha \Delta w_{i,j}(t-1) \]

### Properties of Backpropagation

- Minimizes error over training examples
  - Will it generalize well to subsequent examples?
- Training can take thousands of iterations $\rightarrow$ slow!
- Using network after training is very fast
Face Recognition Example

Face Recognition Task (1)

- **Task:**
  - Classifying camera images of faces of various people in various poses.

- **Input Data:**
  - Images of 20 different people
  - 32 images per person, varying in:
    - expression (happy, sad, angry, neutral)
    - direction they are looking (left, right, straight ahead, up)
    - whether or not they are wearing sunglasses
    - variation in background
    - clothing worn by person
    - position of the person's face within the image

Typical Input Images

Typical input images
Face Recognition Task (2)

- Target Functions:
  - Identity of the person
  - Direction in which the person is facing
  - Gender of the person
  - Whether the person is wearing sunglasses or not

Learned Hidden Unit Weights

Learned Weights