Neural Networks in the Real World

Functional Programming and Intelligent Algorithms

Prof Hans Georg Schaathun

Høgskolen i Ålesund

20th March 2017

Six steps of neural network application

1. Input and output
2. Data preprocessing
3. Split the data
4. Design the Network
5. Train the Network
6. Test the Network

Input and output

ID	Name	Gender	Chest X-ray	Symptoms	Address
1	Anna A.	F		Cough, Breathing trouble	1A Street 1, District A, City AA
2	Bob B.	M		Chest pain, cough	2B Street 2, District B, City BB
3	Cindy C.	F		Wheezing, breathing trouble, chest pain.	3C Street 3, District C, City CC
...	...	...	...	...	...

Input and output

1. Feature vector
    - What does your input look like?
2. Output values
    - Discrete or continuous?
    - How do you code class labels?

Data preprocessing

- Why preprocess the Data?
- Data preprocessing includes:
  - data cleaning
  - data integration
  - data transformation
  - data reduction

Split the data

- Training (e.g. 50%)
- Validation (e.g. 25%)
- Testing (e.g. 25%)

Design the Network

- Training algorithms (Gradient descent backpropagation, Resilient backpropagation, Conjugate gradient backpropagation)
- Error function (e.g. Mean squared error, cross-entropy)
- Activation functions (sigmoid, tanh, softmax)
- Network architecture (number of hidden layers, nodes per layer)

Train the Network

— Train with different configurations and parameters
  - number of epochs
  - different learning rates (η)
  - different architectures
  - different β in the sigmoid?

— Test each configuration on the *validation set*

Test the Network

- Choose *one* design based on tests with the validation set
- Use the *test set* once only
- Final assessment of your chosen configuration

Batch and sequential training

1. Sequential training
   - Considers one training item at a time
   - Updates weights for every individual item
2. Batch training
   - Considers the entire data set
   - Weight updates are added together
   - Weights are updated when all changes are aggregated

What are the advantages and disadvantages of each approach?

Advantages and disadvantages

1. Batch training performs a more accurate estimate of the error gradient
2. Batch training is more likely to get stuck in a local minimum
3. In sequential training, to get full benefits
   - randomise the order of the training items
4. In batch training, order does not matter

— Compromise: minibatches
   - take a subset of the training set for batch training
   - move on to another (disjoint) subset

Accelerated learning

— Use hyperbolic tangent as activation function

$$Y^{tanh} = \frac{2a}{1 + e^{-bX}} - a \tag{1}$$

— Use of Momentum

$$\Delta w_{jk}(p) = \beta \times \Delta w_{jk}(p-1) + \alpha \times y_j(p) \times \delta_k(p) \tag{2}$$

— Adaptive learning rate