Applying Neural Networks

Michael J. Watts

http://mike.watts.net.nz

Lecture Outline

- When to use an ANN
- Preparing the data
- Apportioning data
- What kind of ANN to use
- Training ANN

When to use an ANN

- When the rules of the problem aren't known
- When there is an underlying function
  - ANN can't model random processes!
- ANN learn from existing data
- Careful preparation of the data is essential!

Preparing the Data

- Need a numerical representation of inputs
- Beware when encoding discrete items!
  - nominal vs. ordinal
- Binary coding should be -1,1 not 0,1
  - hyperplane geometry & small initial weights
  - most published application papers ignore this, however

Preparing the Data

- Identify essential variables
  - data analysis
- Redundant variables complicate the learning process
  - curse of dimensionality
- Need more training examples than inputs
- Remove / transform outliers

Preparing the Data

- Normalisation
  - not strictly necessary
  - speeds training
    - sigmoid, tanh functions
  - again, either -1,1 or 0,1
  - set mean 0 STD 1
- MATLAB MLP simulator normalises inputs automatically

Preparing the Data

- Two kinds of problems
  - classification
  - function approximation / time series
- Classification has binary outputs
- Function approximation has continuous outputs

Preparing the Data

- Two ways of encoding multiple classes
- One output per class
  - sparse encoding
  - e.g. 1 0 0, 0 1 0, 0 0 1
- Combining outputs
  - dense encoding
  - e.g. 1 0, 0 1, 0 0

Preparing the Data

- Some people use 0.1 & 0.9 instead of 0 & 1
- Used to speed up training
  - limits of sigmoid function
- This is a gimmick
- Cannot interpret outputs as probabilities
- Better training algorithm should be used

Apportioning Data

- Splitting up the available data into multiple sets
- Two or three sets
  - training
  - testing
  - validation
- Each set has a different purpose

Apportioning Data

- Training data set is used to train the network
- Larger of the three sets
- Usually 50-75% of the data
- Should represent a wide range of output classes / values
  - mustn't omit any target classes

Apportioning Data

- Testing data set is used to evaluate the performance of the trained network
- Network has not been trained on testing data
- Accuracy over testing set is an indicator of generalisation ability of the network

Apportioning Data

- 25% of the total data if no validation set it used
- Otherwise, half of the remaining data
- Must also represent an adequate range of output classes / values

Apportioning Data

- Validation data set is used as a final test
- Performance over this set is evaluated once
- Guards against optimising the network to the training set

Apportioning Data

- Similar to over-training
- Validation set consists of all remaining data
- As before, must have a wide range of output classes / values

Apportioning Data

- Can't just split a data set into three to form training, testing and validation sets
- Statistical parameters of each set should be similar
- Distribution of target classes / values should be similar
  - can't have all of class A in the training set and all of class B in the testing set

What Kind of ANN to use

- Perceptrons are useful for linearly separable problems
- MLP handle problems perceptrons cannot
  - non-linearly separable
- Both use supervised learning
  - require known outputs for each example

What Kind of ANN to use

- Kohonen SOMs are good for classification
- Can't handle function approximation very well
- Unsupervised learning
  - no outputs required
- Find clusters in the data

Training ANN

- Goal is to maximise generalisation accuracy
- Problems with over-training
  - low training error
  - high generalisation error
- Common practice is to train until a low training error, then evaluate generalisation on test set

Training ANN

- Can lead to optimising on test set as well
- If the network is optimised for performance on the testing set, then the network is only good for that data set
- Use of validation data set is intended to counter this

Training ANN

- Number of hidden neurons must be selected for MLPs
- Some heuristics exist for this
  - most are not reliable
- To avoid over-training, number of connections should be less than the number of training examples
  - this is itself a heuristic, though

Training ANN

- Selection of parameters for backpropagation training
- Learning rate
  - must be determined by trial and error
- Momentum
  - less critical
- Number of epochs
  - effects generalisation

Training ANN

- Heuristics exist for each of these parameters
- Trial and error is the only reliable method
- Care must be taken in assessing generalisation
  - selection of validation set

Summary

- Applying ANN is something of a “black art”
- Each problem is different
- Careful preparation of data is critical
- Trial and error is often the only way to find an optimal network