Utvärdering av sensorsignaler - bilinjära modeller och neurala nät

Tomas Öberg

www.tomasoberg.com

Innehåll

- *Bilinjära modeller*
  - *Allmänt*
  - *PCA*
  - *PLS*
- *Neurala nät*
  - *Allmänt*
  - *Feed-forward*
  - *Applikationer*
- *Jämförelse mellan neurala nät och bilinjära modeller*

What is a neural net / ”artificial neural net” (ANN)?

- *Definition*
  - *Artificial neural networks are an information processing technology inspired by studies of the brain and nervous system.*
- *Characteristics*
  - *They approximate biogical systems in their abilities to do pattern recognition, signal processing, etc.*

History

- ***1943 McCulloch & Pitts*** *"A logical Calculus of Ideas Immanent in Nervous Activity"*
- *Developments*
  - *Macroscopic intelligence*
    - *John von Neuman et al: Computers.*
    - *Marvin Minsky et al: Black box AI, expert systems.*
  - *Microscopic intelligence*
    - *Frank Rosenblatt et al: Perceptron, neural nets.*

History cont

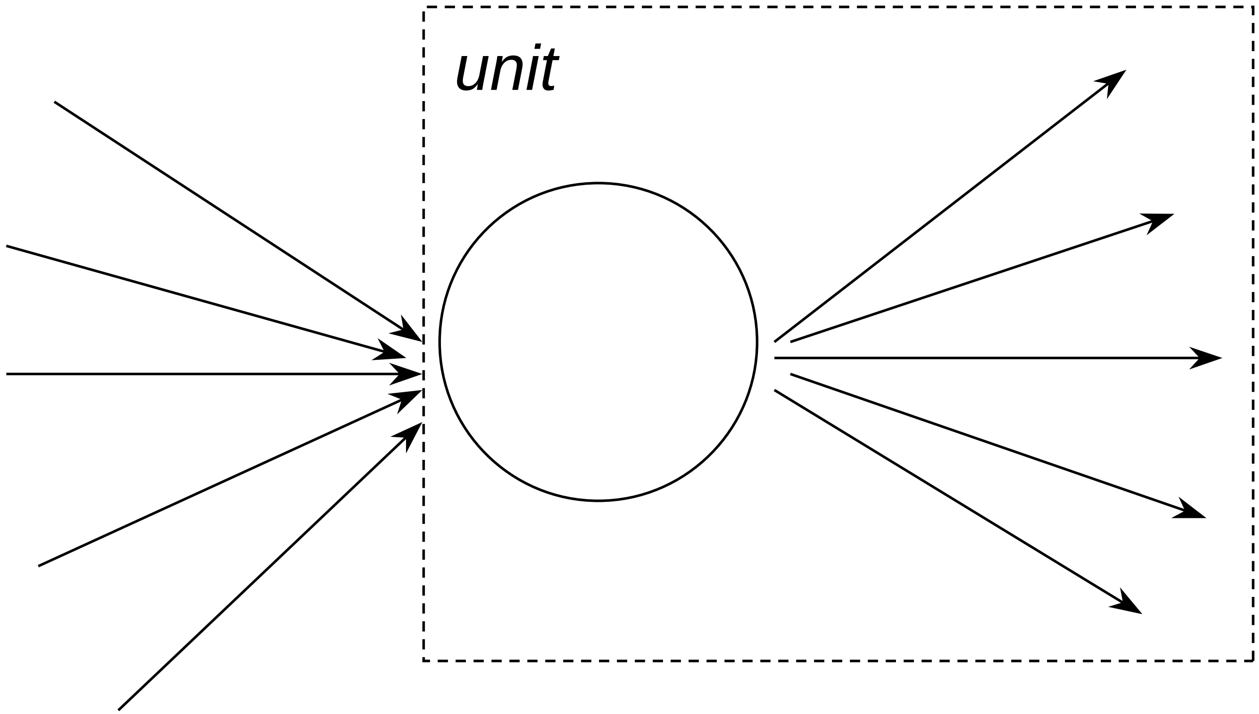
- ***Re-evaluation***
  - *Late 1960s.*
  - *Evaluation of funding agencies, funding shifted to expert systems.*
- *New beginnings*
  - *Computing revolution.*
  - *David E. Rummelhart et al solved key problems encountered in the late 1960s.*

Suitable tasks (”human-like”)

- *Pattern recognition.*
- *Generalization.*
- *Decision making and estimating (robust to noise!)*

The ”neurons”

Networks are built up out of units/neurons, connected with eachother and/or the outside world.



Introduction to multi-layer feedforward neural net

- *The most commonly used neural architecture for real applications.*
- *”Back-propagation” is a misnormer because it is one out of many learning algorithms.*
- *Application areas*
  - *Prediction or forecasting.*
  - *Pattern recognition.*

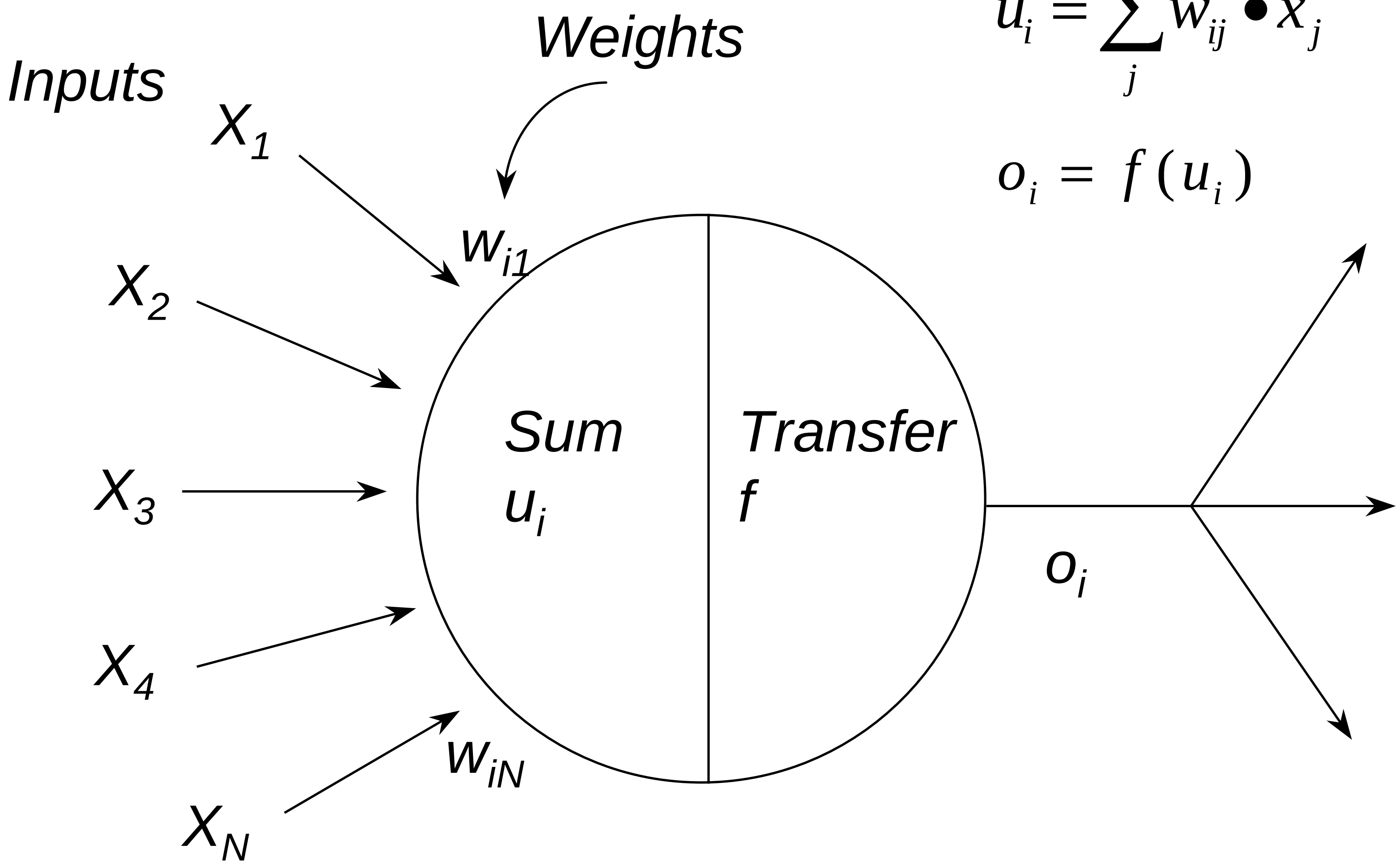
MLF-net cont.

- *The non-linear feedforward network with one "hidden layer" can accurately approximate any measurable relationship.*
- *This means one computional model can be used in almost any prediction or pattern recognition application.*
- *There are numerous methods to adjust the connection weights.*

Neural network processing

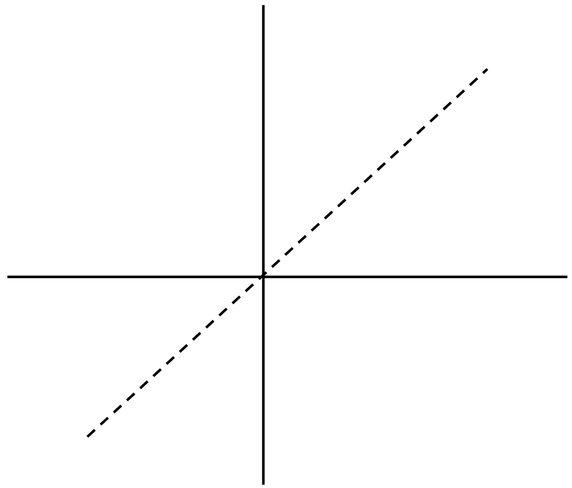
$$u_i = \sum_j w_{ij} \bullet x_j$$

$$o_i = f(u_i)$$

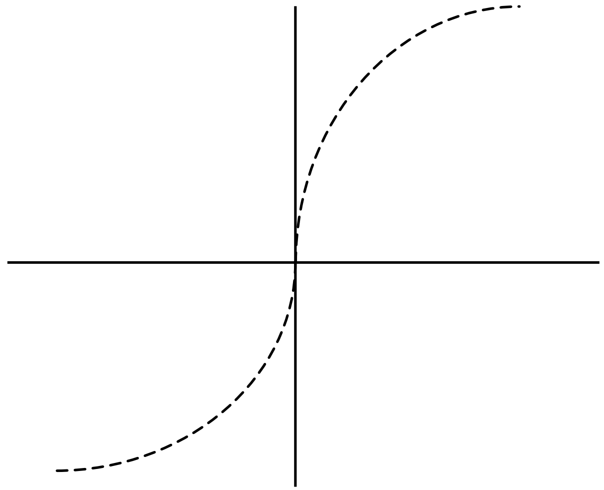


Transfer functions

- *Linear*
  - *f(u)=u*



- *Nonlinear*
  - *e.g. f(u)=tanh(u) or f(u)=1/(1+e^{-u})*



Training of the network

- *Weights are important for the adequate functioning of the network.*
- *Proper settings of weights are not known beforehand.*
- *Start random and adjust weights with small steps.*
- *Supervised: Present examples to the network (=CALIBRATION!)*

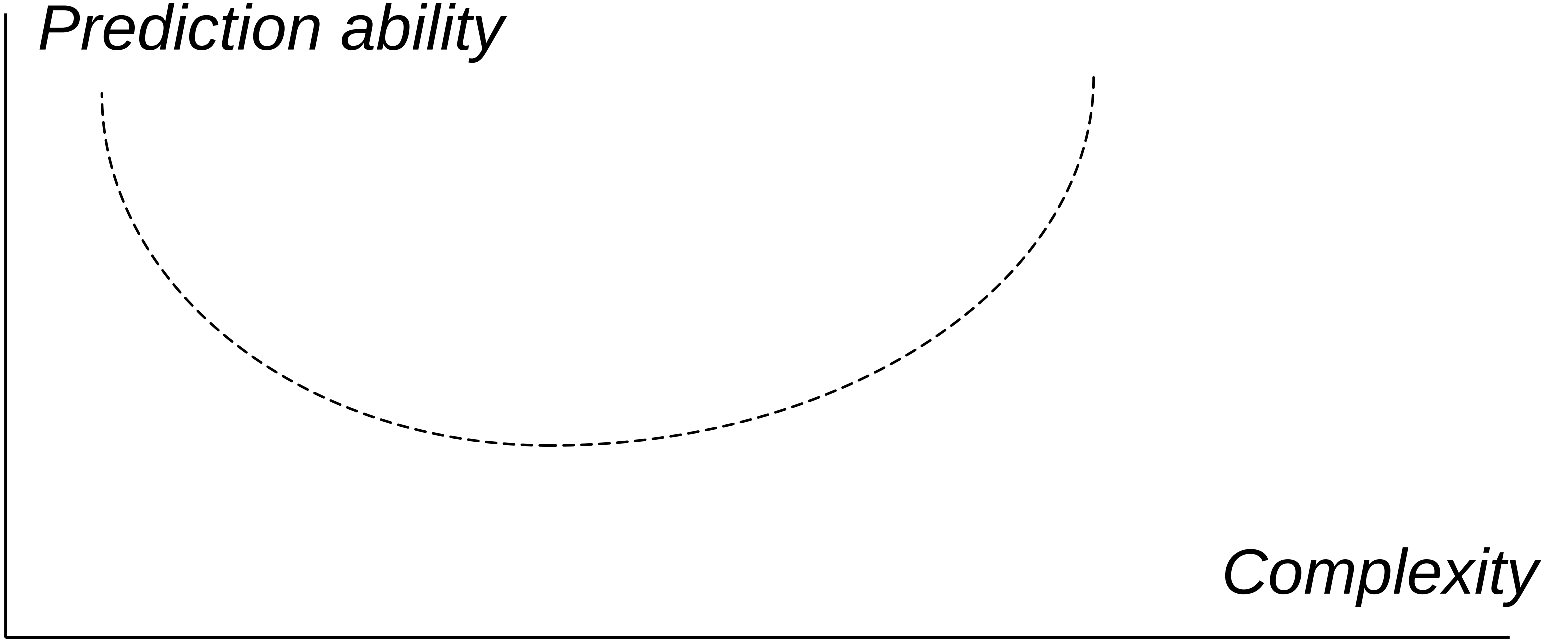
Testing of the network

- *Independent set of examples - test ”generalization capability”*

  (=PREDICTION ABILITY!).
- *Cross validation if not enough examples.*
- *Recognition: Examples used for training.*
- *Prediction: Examples not used for training.*

”Overtraining the network”

Loss of generalization (prediction) capability.



Preprocessing & statistics - a key to success

- *Use regression techniques to give insights in choice of explanatory variables.*
- *Use descriptive statistics and graphics to understand data distributions, then transform as needed, e.g.* $log(x_i)$*.*
- *Use principal component analysis to understand complexity and interrelationships.*

Outliers

Outliers can significantly impact a model's result, and must therefore be indentified. Always check with appropriate techniques!

Network design

- *Input units/neurons:*
  - *Determinded by representation of the problem (normalized data).*
- *Output units/neurons:*
  - *Determinded by representation of the solution.*
- *Hidden units/neurons:*
  - *Trial and error, or systematic search.*

Other design factors

- *Transfer functions*
  - *input units: flow through f(x)=x*
  - *hidden units: sigmoid or tanh*
  - *output units: problem dependent*
- *Weights*
  - *random, in the dynamic range of the transfer function*
- *Learning algorithm, rate, no of iterations, etc. EXPERIMENT!*

Interpretation

- *Non-linear feedforward networks do not have a current way to explain input variables.*
- *Interpretation of the models is difficult or impossible.*
- *Simulations, contour plots, etc. can aid the interpretation, but is valid only within the calibration range!!!*

Successful applications

- *Financial applications:*
  - *Credit card fraud.*
  - *Credit approval*
  - *Stock picking*
  - *Target marketing*
- *Industrial*
  - *Engine diagnostics*
  - *Signal processing*
  - *Adaptive control*

Applications in chemistry

Data from presentation by Jure Zupan.

- *Classification of olive oils.*
- *Classification of chemical bonds.*
- *Fault detection in chemical processes.*
- *Modelling a liquid phase compsition for HPLC.*
- *Prediction of structural fragments from infrared spectra.*
- *Etc., etc.*

Practical experience from application to a problem in food chemistry

Reference:

Claus Borggard: *Modelling non-linear data using neural networks regression in connection with PLS or PCA.* Spectroscopy Europe 6/3 (1994).

Software:

The Unscrambler and Neural-Unscrambler.

Data set

- *81 NIR spectra, from different mixtures of water, fish meal and starch.*
- *19 wavelengths in the region 1200-2400 nm.*
- *Data from Tormod Næs, MATFORSK.*

Experience with PLS and PCR

- *PLS: 12 factors optimal. SEC=1.32%, SEP=2.08% and R=0.995.*
- *PCR: 15 factors optimal. SEC=1.49%, SEP=2.72% and R=0.992.*

The many factors indicated strong non-linearities in the data

Neural network model

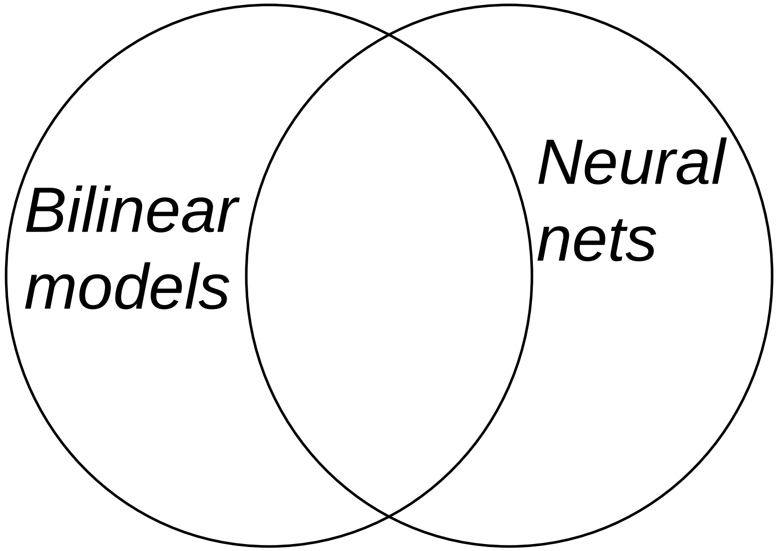
- *Input data: First four scores calculated with the Unscrambler in the PCA models.*

- *SEP on average (6 test segments) improved by 66% compared to standard PLS.*

Two advantages of PCA preprocessing

- *Faster modelling (fewer inputs).*
- *Tools for outlier detection and vizualisation prior to NN-modelling.*

Relation between neural nets and bilinear models



What can we achieve?

- *Neural nets:*
  - *Models of all kind of relationships.*
  - *Optimal prediction.*
- *Bilinear models:*
  - *Satisfactory modelling results for most linear and non-linear relationships.*
  - *Graphical vizualisation of data.*
  - *Optimal interpretation (?).*

When to use bilinear modelling?

My suggestions:

- *Always the first alternative because of*
  - *Ease of computation.*
  - *Ease of interpretation.*
- *Extended to model non-linear data*
  - *With transformations of X- and Y-varaibles.*
  - *Polynomial PLS, locally weighted, etc.*

When to use neural nets?

My suggestions:

- *To obtain an optimal model with regard to prediction ability.*
- *Use as a bench-mark for other models.*

Combine both!

My suggestions:

- *Ordinary statistics and bilinerar modelling should always be used to get a preliminary description of the data.*
- *Bilinear modelling can in may cases be useful preprocessing tool.*

Conclusions

- *No one model is optimal for every situation.*
- *Neural nets are just another type of mathematical model (NOT AI!).*
- *Use as simple model as possible (Occams razor).*
- *Don’t be afraid of neural nets, but don’t forget what you have previously learnt about multivariate data analysis.*