Lecture 18

Monday, March 28, 2022 8:10 AM

Nonlinear Modeling Part III

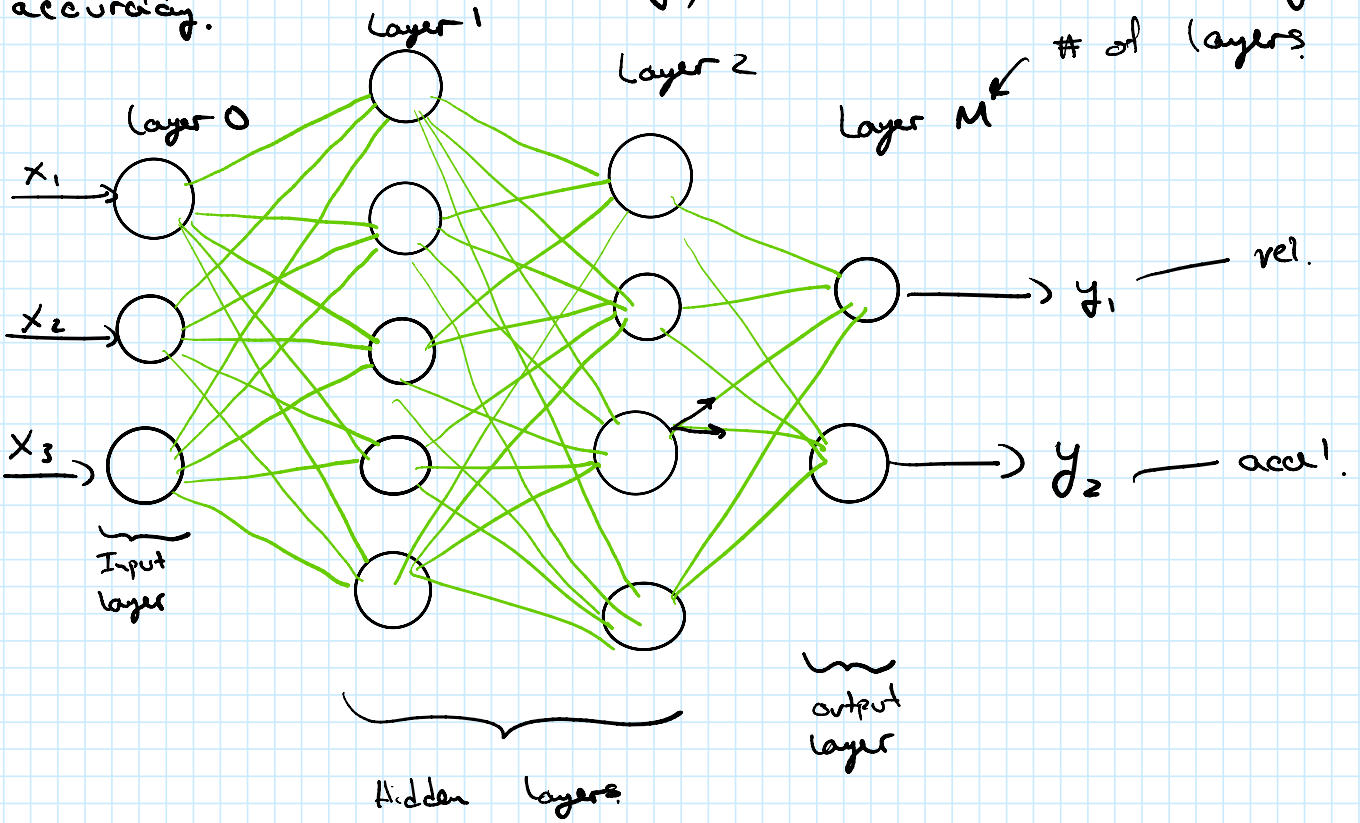
Admin

1. HW2 graded.
2. HW#3 by midnight today.
3. Exam #3 on Weds.

Topics:
1. Hypothesis testing
2. PCA/ICA ←
3. regression ← {ex. regression, multivariation, RLS
4. Kalman.
5. Linear models
   - ARX, ARMA, ARMAX, laguerre.

Artificial Neural Network (ANN)

- ANNs are multi-node structures which enable more complex (non-linear) learning & classification than single perceptrons.
- Multi-layer ANNs are universal approximation funcns. that can approximate an arbitrary, measurable funcn. to any accuracy.



- Input layer: the data presented to the network.
- Output layer: network response (classification/prediction).
- Hidden layers: additional neurons enable extraction/processing/modeling of the data

- Output layer: network response (classification / prediction).
- Hidden layers: additional neurons enable extraction / processing / modeling of the data.

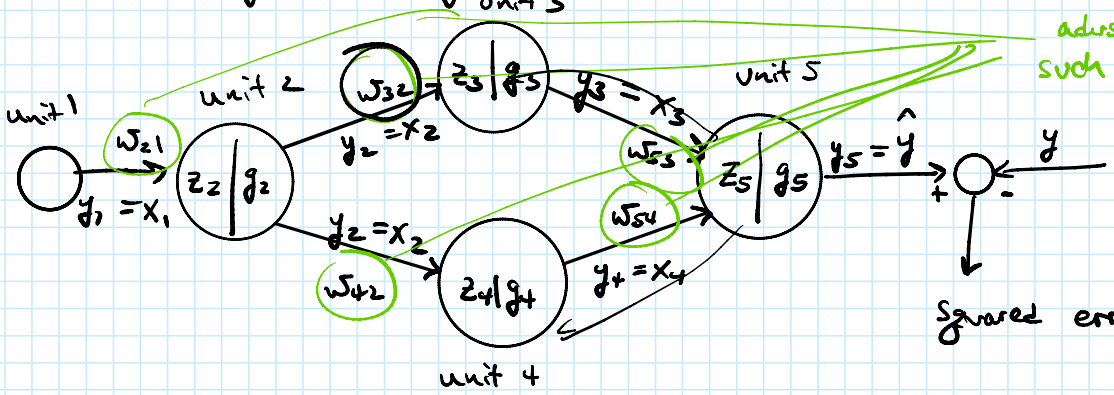
- Network variables:
  - w_{ji}^m: weight of connection from unit i to unit j in layer m.
  - b_i^m: bias of unit i in layer m.
  - y_j^m: output of unit j in layer m.
  - x_i^m: input to a unit in layer m.

- Forward:
  - $z_j^m = \sum_i w_{ji}^m x_i^m + b_i^m$
  - $y_j^m = g_j(z_j^m) = x_j^{m+1}$

The outputs of all units can be computed ~~recursively~~ successively. from layer 0 to layer M. This is called Forward Propogation

- Training an ANN (multilayer perceptron) is done by a process called Error Backpropogation, similar to how we trained a single neuron / perceptron.
  - The goal is to minimize prediction error.
  - Gradient Descent Method can be used:
    - Scaled Conjugate Descent.
    - Levenberg-Marquardt method.
    - Bayesian regularization.

Error Backpropogation Algorithm example: Three ANN.



- Start the gradient descent calculation at the output layer, then recursively back-calculate to the input layer

- start the gradient descent calculation at the output layer, then recursively back-calculate to the input layer

Learning rate. $E = \frac{1}{2}(\hat{y} - y)^2$

1) $\Delta w_{53} = -\rho \, \text{grad}_{w_{53}} E$

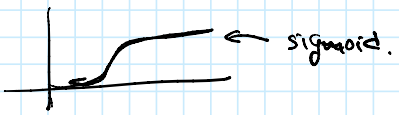
$= -\rho \frac{\partial E}{\partial w_{53}}$ $\quad z_5 = w_{53}x_3 + w_{54}x_4$

$= -\rho \frac{\partial E}{\partial y_5} \frac{\partial y_5}{\partial z_5} \frac{\partial z_5}{\partial w_{53}}$ (chain rule)

$= -\rho (\hat{y} - y) g_5'(z_5) x_3 = \rho \delta_5 x_3$

$= -\delta_5$

(can use for next layer.)



2) $\Delta w_{32} = -\rho \, \text{grad}_{w_{32}} E$

$= -\rho \frac{\partial E}{\partial y_5} \frac{\partial y_5}{\partial z_5} \frac{\partial z_5}{\partial x_3} \frac{\partial x_3}{\partial z_3} \frac{\partial z_3}{\partial w_{32}}$

$= \rho \delta_5 w_{53} g_3'(z_3) x_2$

δ_3

3) $\Delta w_{42} = \rho \delta_5 w_{54} g_4'(z_4) x_2$

δ_4

4) $\Delta w_{21} = \rho \, \text{grad}_{w_{21}} E$ ← must consider both routes to the output.

$= -\rho \frac{\partial E}{\partial z_5} \left[\frac{\partial z_5}{\partial x_3} \frac{\partial x_3}{\partial x_2} \frac{\partial x_2}{\partial w_{21}} + \frac{\partial z_5}{\partial x_4} \frac{\partial x_4}{\partial x_2} \frac{\partial x_2}{\partial w_{21}} \right]$

$= \rho \left[\delta_5 w_{53} g_3'(z_3) w_{32} + \delta_5 w_{54} g_4'(z_4) w_{42} \right] \frac{\partial y_2}{\partial w_{21}}$

δ_3 $\qquad$ δ_4

- The entire process is streamlined by computing δ_j starting from the last layer and propagating to the first.

[Algorithm]

Types of Neural Networks

1. Feed forward neural network.
2. Recurrent neural network: hidden layers store time-history.
3. Convolution Neural Network. -> uses Kernels to reduce dimensionality.
4. Radial Basis Funcn. Network -> use RBFs as activation funcns.
5. Self organizing Feature MAPS -> reduce dimensionality of data.