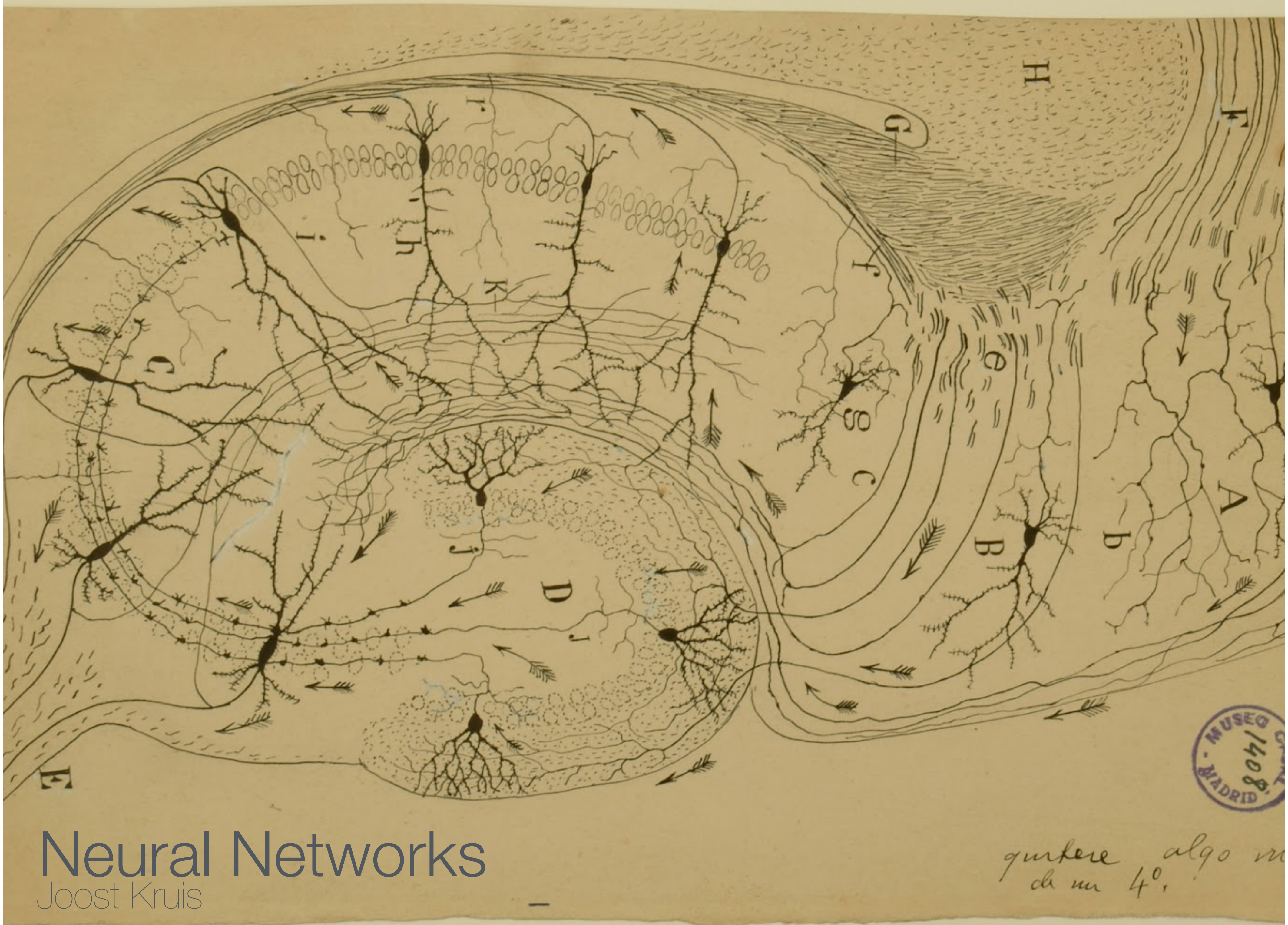


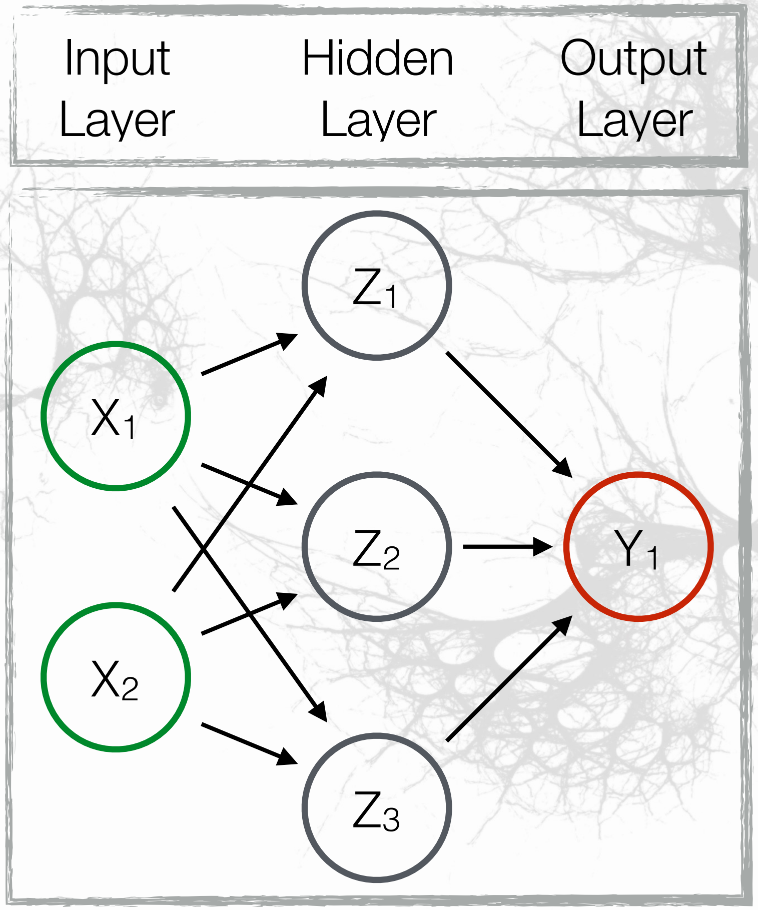
Neural Networks

Joost Kruis

Neural Networks - Basics

- class of statistical models for two-stage regression or classification
- takes non-linear functions of linear combinations of the inputs
- first developed as models for the human brain
- represented by a network diagram

- Google:
  - Alpha Go
  - Google Brain
- Nvidia
  - Drive PX 2



Neural Networks - Building

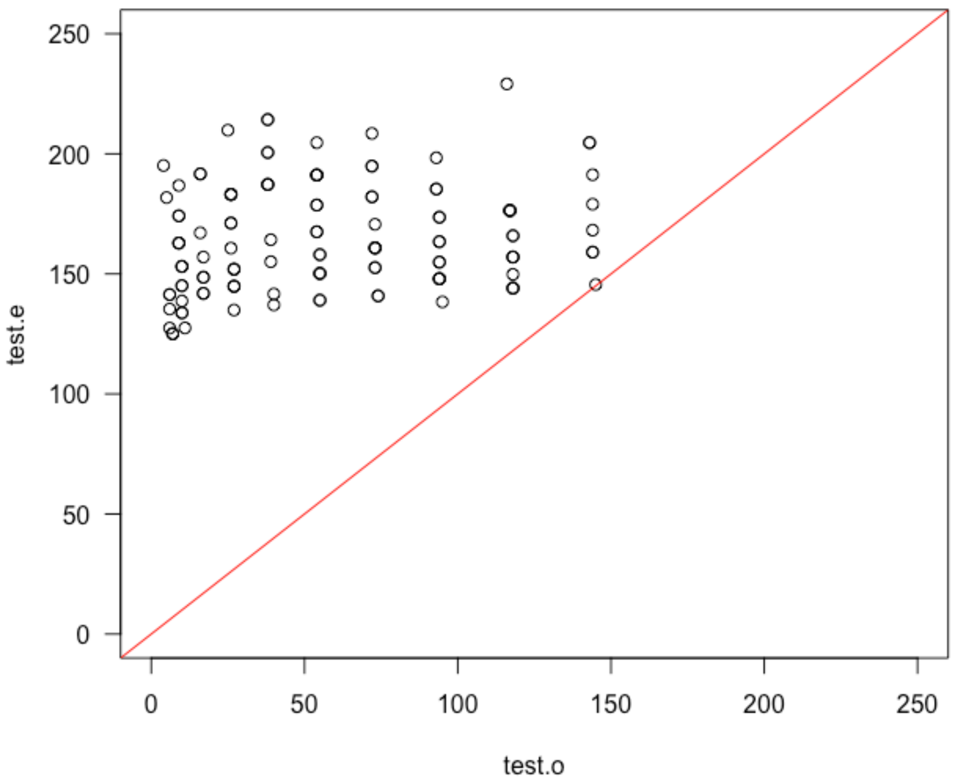
- Nodes
  - Input (X)
  - Hidden (Z)
  - Output (Y)
- Weighted Connection Network
  - from input to hidden [X + 1, Z]
  - from hidden to output [Z + 1,Y]
  - +1 are for bias units (intercepts)
- Functions
  - Activation from input to hidden (sigmoid)
  - Output from hidden to output
    - regression = identity function
    - classification = softmax function

Neural Networks - Functions

X	$= [N_p , X_i]$	—> Input Matrix
W^{xz}	$= [X_i , Z_k]$	—> Weights Matrix (x to z)
A^{xz}	$= X \times W^{xz}$	—> Weighted Input Matrix
A^{f}	$= f(A^{xz})$	—> Activation Matrix
$f(A^{xz})$	$= 1/(1+e^{-A\wedge xz})$	—> Sigmoid Activation Function
W^{zy}	$= [Z_k , Y_j]$	—> Weights Matrix (z to y)
A^{zy}	$= A^{f} \times W^{zy}$	—> Weighted Hidden Matrix
$\hat{Y}$	$= f(A^{zy})$	—> Output Matrix
$f(A^{zy})$	$= A^{zy}$	—> Identity Output Function
$f(A^{zy})$	$= (e^{A\wedge zyk})/\Sigma^{K}\_{(L=1)}\, e^{A\wedge zyl}$	—> Softmax Output Function

Neural Networks - Untrained Network

- all connection weights as randomly sampled from standard normal distribution
- predictions for expected value rather bad



$$x_1 = U(0,10)$$
$$x_2 = U(0,10)$$
$$\epsilon = U(0,5)$$
$$y = x1 * .25 + (x2)^2 * 1.4 + \epsilon$$

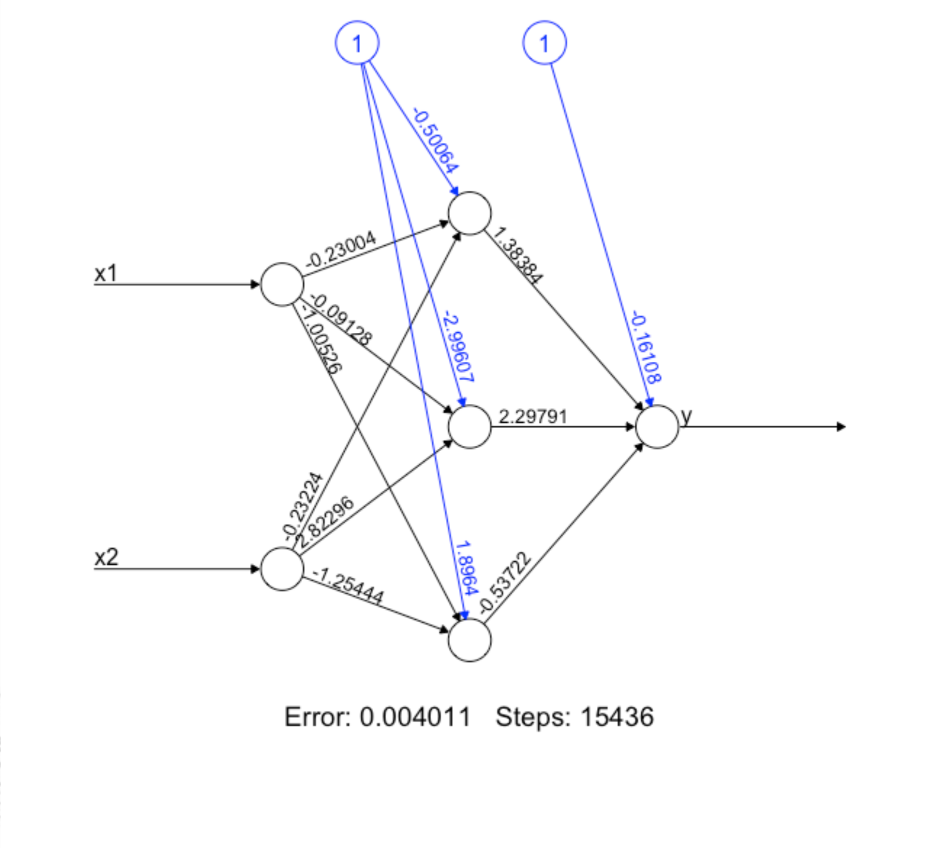
- we need to train the network

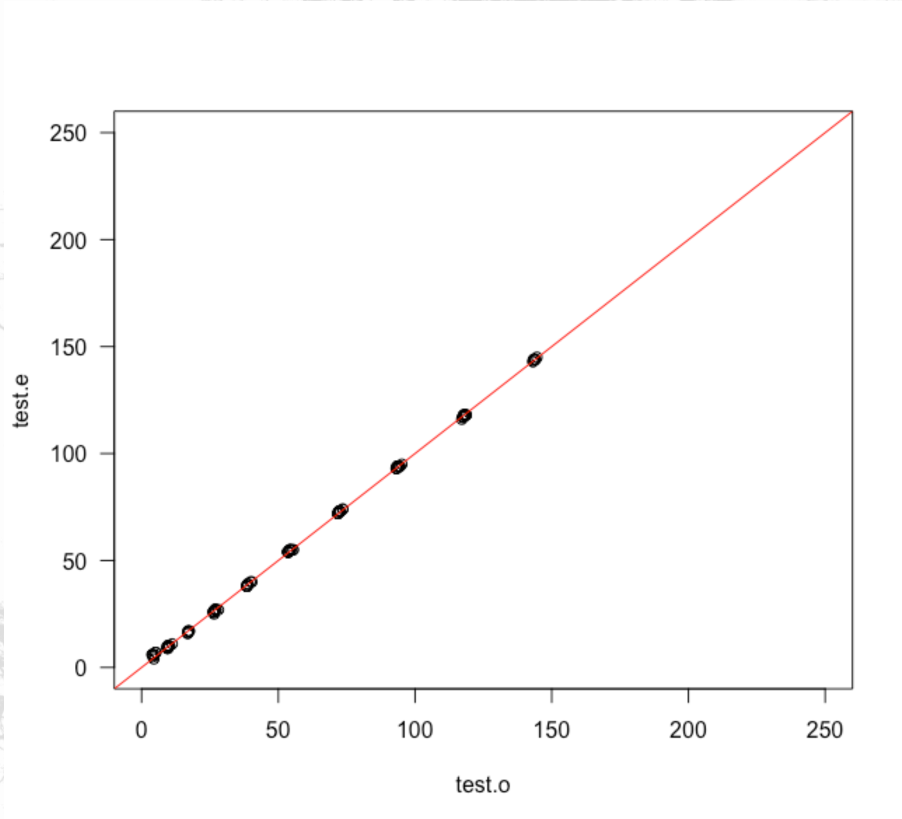
Neural Networks - Training the Network

- Optimisation of connection weights
- Minimisation of loss function
    - regression: SSE
    - classification: SSE or cross-entropy
- Global minimiser —> overkill solution
    - Penalisation
    - Stopping Rule
- Gradient descent —> back-propagation
    - optimisation using the partial derivatives
        - forward pass —> weights fixed and predicted values are computed
        - backward pass —> errors are computed and used for calculation of gradient for update
- Scaling
    - Variables must be scaled

Neural Networks - Training the Network

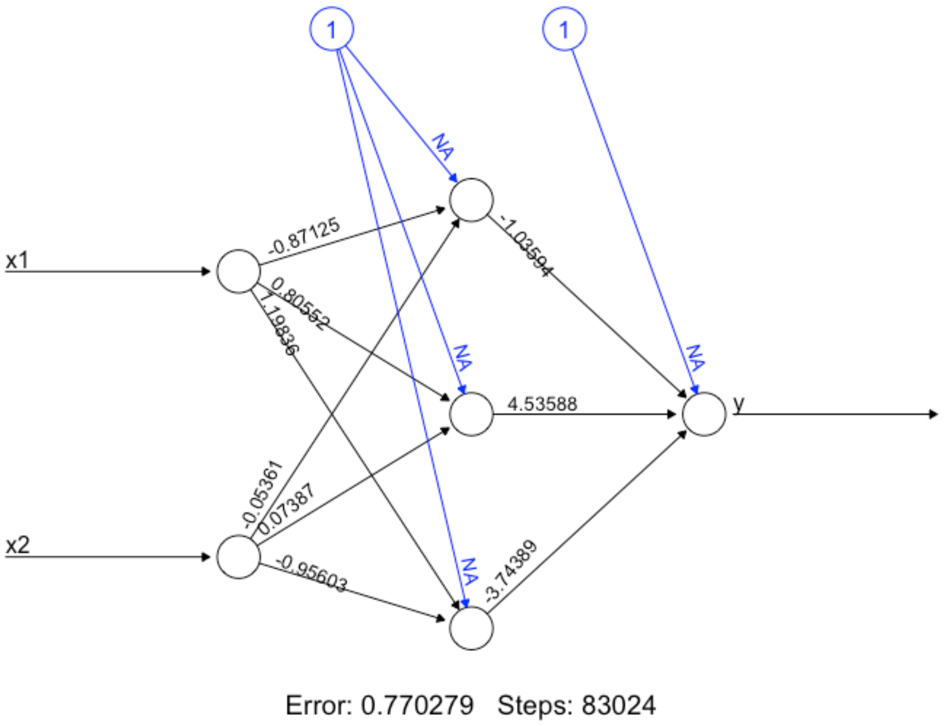
- neuralnet package in R
- training- and test data
- better predictions after training

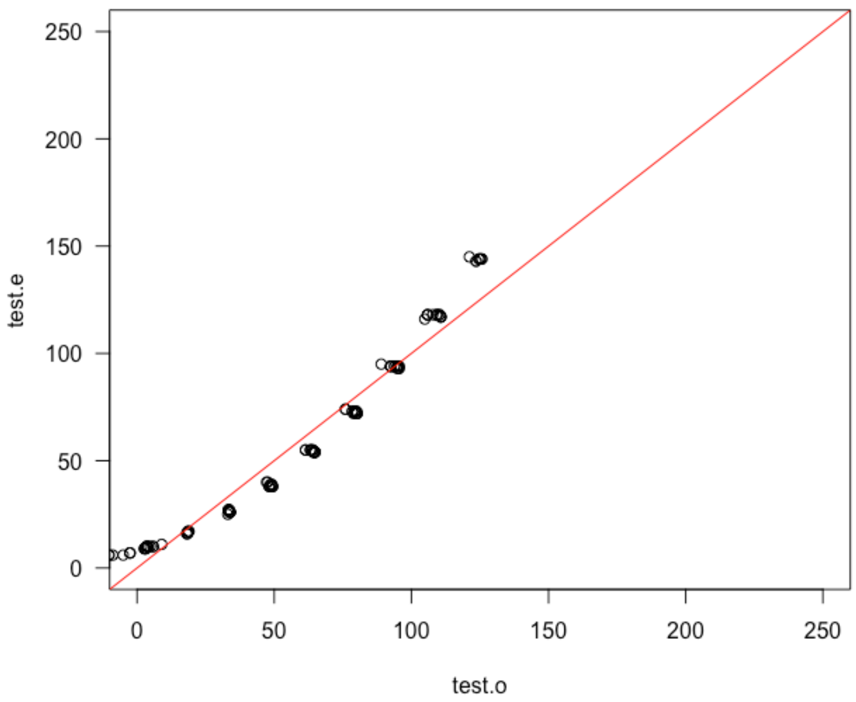




Neural Networks - Training the Network

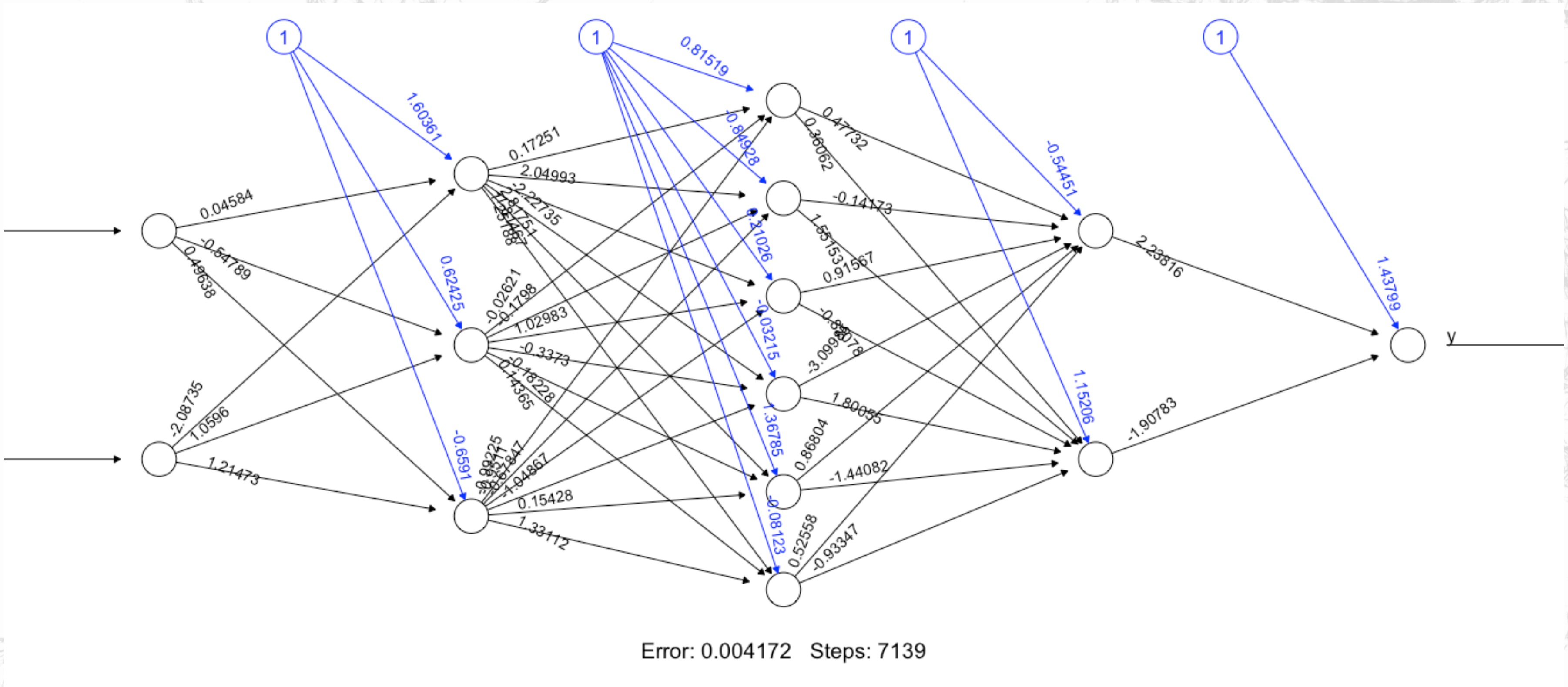
- without intercepts





Neural Networks - Extensions and Applications

- neural networks can be huge and have multiple hidden layers
- neural networks can be combined with other ML techniques



Neural Networks - Applications for Psychology

- excellent and easy to use tool for prediction
  - will a student complete the first year successfully?
- hard to interpret hidden layer, especially when size goes up

The End