XOR with intermediate ("hidden") units

- Intermediate units can re-represent input patterns as new patterns with altered similarities
- Targets which are not linearly separable in the input space can be linearly separable in the intermediate representational space
- Intermediate units are called “hidden” because their activations are not determined directly by the training environment (inputs and targets)

Hidden-to-output weights can be trained with the Delta rule

How can we train input-to-hidden weights?
- Trick: We don’t need targets, we just need to know how hidden activations affect error (i.e., error derivatives)

Note: Need nonlinear unit function for hidden units
- Linear units cannot alter the relative similarities of patterns
- Linear algebra: Each layer of weights is a matrix multiplication; with linear units, can just multiply the matrices to get equivalent 1-layer transformation (which we know is insufficient)

Delta rule as gradient descent in error (sigmoid units)

\[
\begin{align*}
    n_j &= \sum_i a_i w_{ij} \\
    a_j &= \frac{1}{1 + \exp(-n_j)} \\
    E &= \frac{1}{2} \sum_j (t_j - a_j)^2
\end{align*}
\]

Gradient descent:
\[
\Delta w_{ij} = -\epsilon \frac{\partial E}{\partial w_{ij}}
\]

Generalized Delta rule ("back-propagation")

\[
\begin{align*}
    n_j &= \sum_i a_i w_{ij} \\
    a_j &= \frac{1}{1 + \exp(-n_j)} \\
    E &= \frac{1}{2} \sum_j (t_j - a_j)^2
\end{align*}
\]

Gradient descent:
\[
\Delta w_{ij} = -\epsilon \frac{\partial E}{\partial w_{ij}}
\]
Back-propagation

Forward pass (⇑)

\[ a_j = \frac{1}{1 + \exp(-n_j)} \]

\[ n_j = \sum_i a_i w_{ij} \]

\[ a_i = \frac{1}{1 + \exp(-n_i)} \]

Backward pass (⇓)

\[ \frac{\partial E}{\partial a_j} = -(t_j - a_j) \]

\[ \frac{\partial E}{\partial n_j} = \frac{\partial E}{\partial a_j} a_j (1 - a_j) \]

\[ \frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial n_j} a_i \]

\[ \frac{\partial E}{\partial a_i} = \sum_j \frac{\partial E}{\partial n_j} w_{ij} \]

Accelerating learning: Momentum descent

\[ \Delta w_{ij}[t] = -\epsilon \frac{\partial E}{\partial w_{ij}} + \alpha \left( \Delta w_{ij}[t-1] \right) \]

What do hidden representations learn?

- Mapped orthography to semantics (unrelated similarities)
- Compared similarities among hidden representations to those among orthographic and semantic representations (over settling)

"Auto-encoder" network (4–2–4)
Projections of error surface in weight space

- Asterisk: error of current set of weights
- Tick mark: error of next set of weights
- Solid curve (0): Gradient direction
- Solid curve (21): Integrated gradient direction (including momentum)
  - This is actual direction of weight step (tick mark is on this curve)
  - Number is angle with gradient direction
- Lens: “Grad lin” (gradient linearity) is (normalized) dot product of gradient direction and weight change direction
- Dotted curves: Random directions (each labeled by angle with gradient direction)

Epochs 1-2

Epochs 3-4

Epochs 5-6
High momentum (epochs 1-2)

High learning rate (epochs 1-2)

High momentum (epochs 3-4)

High learning rate (epochs 3-4)