The Knowledge Content of Neural Networks

Keith L. Downing

The Norwegian University of Science and Technology (NTNU)
Trondheim, Norway
keithd@idi.ntnu.no

March 25, 2014
- Linear Separability
- Saliency
- Principle Components Analysis
- Hierarchical Clustering based on ANN Layer Behavior
- Topographic Maps
Neurons as Detectors

\[ 2x + 5y \geq 1 \iff y \geq -\frac{2}{5}x + \frac{1}{5} \]
If each data case has $n$ features, then, when plotted in $n$-dimensional space, can the positive and negative instances be separated by a hyperplane of $n-1$ dimensions. E.g. If $n = 2$, the hyperplane = a line.

If so, then a single-neuron detector can easily be reverse-engineered to detect the positive instances.
AND and OR are linearly separable for any input-vector size.
This simple, non-linearly-separable example nearly killed neural network research.


Detecting non-linearly-separable classes requires more than 2 layers of neurons, but weights in multi-layer nets could not be learned prior to the popularization of backprop in the mid 1980’s.
XOR requires 3 Layers

\[
\begin{align*}
    x &= 0.5, -0.5 \\
    y &= 0.5, -0.5 \\
    t_u &= 1, t_v = 1 \\
    v &= 0.5, 0.5 \\
    u &= 0.5, -0.5 \\
    t_z &= 0 \\
    z &= 0 \\
    AND &= AND \\
    OR &= OR \\
\end{align*}
\]

\[y = x + 2\]
\[y = x - 2\]
ANNs can perform mappings of any complexity, whether linearly separable or not. Although, it may require a lot of hidden layers and neurons. However, for a k-layered ANN (with $k > 3$) an equivalent ANN with $k = 3$ can be designed.
Each of the 3 borderlines is expressed by a simple line, which translates into the weights of three detector neurons.

\[ y - x > 0 \]
\[ y + x > 5 \]
\[ y + 4x > 30 \]

These fire on all input vectors \((x,y)\) that are above the line.
Each region of positive training instances is expressed as a conjunction of above and below relationships w.r.t. the borderlines.

Region 3 is above border 2 and below borders 1 and 3.
A positive instance of the concept is an (x, y) case in any of the 3 regions, so the high-level detector, M, represents the disjunct of the 3 regions.
Neurons Detect Salient Contexts

- Three-spined stickleback experiments (Tinbergen, 1951)
- Males develop red bellies when establishing territory.
- Sight of the salient concept, a red belly, makes male's aggressive, even on abstract mock-up figures.

Keith L. Downing
The Knowledge Content of Neural Networks
Mock-ups resembling hawks elicit fear.
Those resembling a goose do not.
What Excites a Toad??

- Worms or moving rectangles resembling worms (Ewert, 1980).
- Neurons in area T5(2) of the toad brain detect worm-ness.

<table>
<thead>
<tr>
<th>Strong Response</th>
<th>Weak Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Worm&quot;</td>
<td>&quot;Partial Worm&quot;</td>
</tr>
<tr>
<td>&quot;Anti-Worm&quot;</td>
<td></td>
</tr>
</tbody>
</table>

No Response
What Excites an Artificial Neuron??

Keith L. Downing
The Knowledge Content of Neural Networks
Two Keys to Intelligent Behavior

1. Knowing when to **differentiate** between two situations based on salient features (for which the situations have unequal values), and thus act **differently** in each.

2. Knowing when to **generalize** over two situations based on salient **similarities**, and thus treat each the **same**.

Salient features are very task dependent.

- Easy task $\rightarrow$ salient feature(s) have **high** variance among the cases.
- Hard task $\rightarrow$ salient feature(s) have **low** variance among the cases (e.g. Where’s Waldo?)
Principal Component Analysis (PCA) with ANNs

Principle components of a data set = vector that captures the highest amounts of variance among the features.

**Important ANN Property**

**If:**

- the values of a data set are scaled (to a common range for each feature such as $[0, 1]$) and normalized by subtracting the mean vector from each element,
- these values are fed into a single output neuron, $z$, and
- the incoming weights to $z$ are modified by correlation-based Hebbian means

$\Rightarrow$

- $z$’s input-weight vector will reflect the principle components of the data set.
The border between regions carved out by a single output neuron is perpendicular to that neuron’s weight vector

\[ xw_x + yw_y \geq t_z \iff y \geq -\frac{w_x}{w_y} x + \frac{t_z}{w_y} \]

- The border is a line with slope \( -\frac{w_x}{w_y} \).
- So, any vector with slope \( +\frac{w_y}{w_x} \) is perpendicular to that border.
- Since neuron z’s incoming-weight vector is \( \langle w_x, w_y \rangle \), it has slope \( +\frac{w_y}{w_x} \) and is therefore perpendicular to the borderline.
Of Mice and Elephants

Keith L. Downing

The Knowledge Content of Neural Networks

### Raw Data Points

- **Mouse**: (0.05, 60), (0.04, 62), (0.06, 68)
- **Elephant**: (5400, 61), (5250, 66), (5300, 69)

### Scaled Data

- **Mouse**: (0, 0.6), (0, 0.62), (0, 0.68)
- **Elephant**: (0.54, 0.61), (0.53, 0.66), (0.53, 0.69)

### Normalized Data

- **Mouse**: (-0.27, -0.04), (-0.27, -0.02), (-0.27, 0.04)
- **Elephant**: (0.27, -0.03), (0.26, 0.03), (0.26, 0.05)
Hebbian Learning ⇒ Principle Components

\[ \Delta w_i = \lambda x_i y \]

<table>
<thead>
<tr>
<th>Input (Size, Color)</th>
<th>Output</th>
<th>( \delta w_{size} )</th>
<th>( \delta w_{color} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-0.27, -0.04)</td>
<td>-0.031</td>
<td>+0.0017</td>
<td>+0.0002</td>
</tr>
<tr>
<td>(-0.27, -0.02)</td>
<td>-0.029</td>
<td>+0.0016</td>
<td>+0.0001</td>
</tr>
<tr>
<td>(-0.27, 0.04)</td>
<td>-0.023</td>
<td>+0.0012</td>
<td>-0.0002</td>
</tr>
<tr>
<td>(0.27, -0.03)</td>
<td>+0.024</td>
<td>+0.0013</td>
<td>-0.0001</td>
</tr>
<tr>
<td>(0.26, 0.03)</td>
<td>+0.029</td>
<td>+0.0015</td>
<td>+0.0002</td>
</tr>
<tr>
<td>(0.26, 0.05)</td>
<td>+0.031</td>
<td>+0.0016</td>
<td>+0.0003</td>
</tr>
<tr>
<td>Sum weight change:</td>
<td></td>
<td>+0.0089</td>
<td>+0.0005</td>
</tr>
</tbody>
</table>

Keith L. Downing  The Knowledge Content of Neural Networks
If the detectors of a network modify their input-weight vectors according to basic Hebbian principles, then, after training, the activation levels of detectors can be used to differentiate the input patterns along the dimensions of highest variance. Hence, those detectors will differentiate between objects (or situations) that are most distinct relative to the space of feature values observed in the training data.

- Train on animal pictures ⇒ Differentiate birds from horses better than horses from donkeys.
- Train on human faces ⇒ Differentiate males from females better than Swedes from Norwegians.

The network figures out the most salient features on its own, via simple Hebbian means.
Assessing Generality of an ANN

- Generalization: Ability to handle similar cases with similar actions.
- In ANNs, measure the correlation between input patterns and activity patterns of output- or hidden-layer neurons, giving a **coarse** indicator of generalization.
- Hierarchical clustering (using dendograms) gives a more detailed, case-by-case assessment.
- A quick look at the hierarchical tree usually indicates whether or not the ANN has learned useful similarities and distinctions between the inputs.

<table>
<thead>
<tr>
<th>Animal</th>
<th>Name</th>
<th>Hidden-Layer Activation Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cat</td>
<td>Felix</td>
<td>11000011</td>
</tr>
<tr>
<td>Dog</td>
<td>Max</td>
<td>00111100</td>
</tr>
<tr>
<td>Cat</td>
<td>Samantha</td>
<td>10001011</td>
</tr>
<tr>
<td>Dog</td>
<td>Fido</td>
<td>00011101</td>
</tr>
<tr>
<td>Cat</td>
<td>Tabby</td>
<td>11011001</td>
</tr>
<tr>
<td>Dog</td>
<td>Bruno</td>
<td>10110101</td>
</tr>
</tbody>
</table>
Hierarchical Clustering

Begin with N items, each of which includes a *tag*, which in this example is the hidden-layer activation pattern that it evokes.

Encapsulate each item in a *singleton cluster* and form the cluster set, C, consisting of all these clusters.

Repeat until size(C) = 1

Find the two clusters, c₁ and c₂, in C that are closest, using distance metric D.
Form cluster c₃ as the union of c₁ and c₂; it becomes their parent on the hierarchical tree.
Add c₃ to C.
Remove c₁ and c₂ from C
Distance metric for clustering:

$$D(c_1, c_2) = \frac{1}{M_1 M_2} \sum_{x \in c_1} \sum_{y \in c_2} d_{ham}(\text{tag}(x), \text{tag}(y))$$

where $M_1$ and $M_2$ = sizes of clusters $c_1$ and $c_2$, respectively.

We do not need to understand the concepts represented by the hidden nodes, only the similarities of the hidden-layer activation patterns.
Topographic Neural Maps

Star-nosed Mole Brain

Body scaled to match brain proportions

Keith L. Downing

The Knowledge Content of Neural Networks
If similar inputs map to neighboring neurons, and those in turn map to neighboring neurons, etc., then:

- **Generalization** occurs naturally
- Small errors in perception still lead to the correct action
- Neural wiring can be reduced.
Topographic Maps in the Brain

Isomorphism between 2 Spaces

Spaces: Sound Frequencies + A Layer of Neurons

If points p and q are close (distant) in the sound frequency space, then the neurons that detect frequencies p and q, \( n_p \) and \( n_q \), are also close neighbors (distant) in the neuron layer.
Artificial Self-Organizing Maps (SOMs)

Competition + Cooperation

Nodes **compete** for input patterns, but then **share** the win by allowing grid neighbors to also update their input weights to more closely match the input pattern.
There Goes the Neighborhood

Neuron Space

Self-Organizing Learning

Keith L. Downing
The Knowledge Content of Neural Networks
Spaces

- Euclidean: City Locations
- Neural: A **ring** of neurons ⇒ Each neuron has 2 neighbors.
Axons *read* morphogen concentrations in their source layer and then search for similar chemical signatures in the target layer.
Hebbian Fine-Tuning of Topographic Maps

Lower-layer neurons require 2 or more simultaneous inputs to fire.
Hebbian Learning with STDP: fire together, wire together...fire apart, weaken.
More Fine Tuning

D fires but W does not, so D-W synapse weakens.

This is a noncontinuous stimulus (less common in the real world), but it does suffice to fire W and D, so the D-W synapse strengthens.
When learning begins with many topographic links, the contiguous nature of most real-world stimuli will strongly bias the training set, leading to depression and disappearance of the non-topographic links.