Artificial Neural Networks 1

Morten Nielsen
Department of Systems Biology,
DTU
Objectives

- Neural network:
  - is a black box that no one can understand
  - over-predicts performance
  - Overfitting - many thousand parameters fitted on few data
HUMAN
NETtalk
(T. Sejnowski and C. Rosenberg, 1987)

Mary had a little lamb

Three of the a's must be pronounced differently! Reading aloud is a context sensitive cognitive skill.
Weight matrices (PSSM)

- A weight matrix is given as
  \[ W_{ij} = \log(p_{ij}/q_j) \]
  where \( i \) is a position in the motif, and \( j \) an amino acid. \( q_j \) is the background frequency for amino acid \( j \).

|   | A  | R  | N  | D  | C  | Q  | E  | G  | H  | I  | L  | K  | M  | F  | P  | S  | T  | W  | Y  | V  |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 0.6| 0.4| -3.5| -2.4| -0.4| -1.9| -2.7| 0.3| -1.1| 1.0| 0.3| 0.0| 1.4| 1.2| -2.7| 1.4| -1.2| -2.0| 1.1| 0.7 |
| 2 | -1.6| -6.6| -6.5| -5.4| -2.5| -4.0| -4.7| -3.7| -6.3| 1.0| 5.1| -3.7| 3.1| -4.2| -4.3| -4.2| -0.2| -5.9| -3.8| 0.4 |
| 3 | 0.2| -1.3| 0.1| 1.5| 0.0| -1.8| -3.3| 0.4| 0.5| -1.0| 0.3| -2.5| 1.2| 1.0| -0.1| -0.3| -0.5| 3.4| 1.6| 0.0 |
| 4 | -0.1| -0.1| -2.0| 2.0| -1.6| 0.5| 0.8| 2.0| -3.3| 0.1| -1.7| -1.0| -2.2| -1.6| 1.7| -0.6| -0.2| 1.3| -6.8| -0.7 |
| 5 | -1.6| -0.1| 0.1| -2.2| -1.2| 0.4| -0.5| 1.9| 1.2| -2.2| -0.5| -1.3| -2.2| 1.7| 1.2| -2.5| -0.1| 1.7| 1.5| 1.0 |
| 6 | -0.7| -1.4| -1.0| -2.3| 1.1| -1.3| -1.4| -0.2| -1.0| 1.8| 0.8| -1.9| 0.2| 1.0| -0.4| -0.6| 0.4| -0.5| -0.0| 2.1 |
| 7 | 1.1| -3.8| -0.2| -1.3| 1.3| -0.3| -1.3| -1.4| 2.1| 0.6| 0.7| -5.0| 1.1| 0.9| 1.3| -0.5| -0.9| 2.9| -0.4| 0.5 |
| 8 | -2.2| 1.0| -0.8| -2.9| -1.4| 0.4| 0.1| -0.4| 0.2| -0.0| 1.1| -0.5| -0.5| 0.7| -0.3| 0.8| 0.8| -0.7| 1.3| -1.1 |
| 9 | -0.2| -3.5| -6.1| -4.5| 0.7| -0.8| -2.5| -4.0| -2.6| 0.9| 2.8| -3.0| -1.8| -1.4| -6.2| -1.9| -1.6| -4.9| -1.6| 4.5 |

- \( W \) is a \( L \times 20 \) matrix, \( L \) is motif length

SLLPAIVEL
YLIPAIIVHI
TLWVDPYEV
Biological Neural network
Biological neuron structure
Artificial neuron

Input signals

Synaptic weights

Threshold

Output signal

\[ O = \sigma \left( \sum_{n=1}^{N} w_n I_n - t \right) \]
Transfer of biological principles to artificial neural network algorithms

- Non-linear relation between input and output
- Massively parallel information processing
- Data-driven construction of algorithms
- Ability to generalize to new data items
Artificial neuron

Input signals

Synaptic weights

Threshold

Output signal

\[ O = \sigma \left( \sum_{n=1}^{N} w_n I_n - t \right) \]

- Strongly simplified in relation to the biology.
- Dates back to McCulloch and Pitts, 1943.
- Linear combination of input, weights and threshold.
- Non-linear (sigmoid) response function.
- Typical choice: \( \sigma(x) = 1/(1 + e^{-x}) \)
Linear separation by simple neural network

Two input features and one output.

\[ O = \begin{cases} 
  1 & \text{for } w_1I_1 + w_2I_2 > t \\
  0 & \text{otherwise}
\end{cases} \]

Similar to SMM, except for delta function!
Linear separation by simple neural network

Two input features and one output.

\[ O = \begin{cases} 
1 & \text{for } w_1 I_1 + w_2 I_2 > t \\
0 & \text{otherwise} 
\end{cases} \]

Equation \( w_1 I_1 + w_2 I_2 = t \) is straight line in \( I_1 I_2 \)-plane:
Linear separation by simple neural network

Two input features and one output.

\[ O = \begin{cases} 
1 & \text{for } w_1 I_1 + w_2 I_2 > t \\
0 & \text{otherwise}
\end{cases} \]

Equation \( w_1 I_1 + w_2 I_2 = t \) is straight line in \( I_1 I_2 \)-plane:

\[ \text{AND}(I_1 + I_2 > \frac{3}{2}) \]

\[ \text{OR}(I_1 + I_2 > \frac{1}{2}) \]
How to predict

• The effect on the binding affinity of having a given amino acid at one position can be influenced by the amino acids at other positions in the peptide (sequence correlations).
  - Two adjacent amino acids may for example compete for the space in a pocket in the MHC molecule.

• Artificial neural networks (ANN) are ideally suited to take such correlations into account
MHC peptide binding

SLLPAIVEL YLLPAIVHI TLWVDPYEV GLVPFLVSV KLLEPVLLL LLDVPTAAV LLDVPTAAV LLDVPTAAV
LLDVPTAAV VLFRGGPRG MVDGTLLLL YMGNTMSQV MLLSVPLL LLLGLLVEV ALLPPINIL TLIKIQHTL
HLIDYLVTS ILAPPVVKL ALFPQLVLIL GILGFVFIAL NLTRQSGRQ GLDVLTAKV RILGAVAKV QVCEPI
ILFGENERV ILMEHIKHL ILDQKINEV SLAGGIIGV LLIENVASI FLLWATAEA SLPDFGISY KKRAPERL
LREPPGGNEI ALSNLEVKL ALNELLQHV DLERKVESL FLGENISNF ALSDDHIYL GLSEFTEYL STAPPAHV
PLDEYFTFL GLVGVVALI RTLDKVEV GLSTAFARV RLDGYRSL YMNGTMSQV GILGFVFGLILKEPVHG
ILGFVFITL LLDFYVPVV GLSPTVWLS WLSLLPVFVL PLGDFDFPS LLGGLLTMV FIAGNSAYE KLGEEFEQ
KLVALGINA DLGMYIPVL RLVTLLDIV MLLAVLYCL AAGIGILTV YLEPGPVTA LLGLTATL TLTDQVFPSV
KTVGWQYWQV TIDQVFPS AFHHVAREL YLNKIQNSL MMRKLIALS AIDMKMIL IMDKNIILK SMVGWAQ
SLLAPAGAKQ KIFGSLAFL ELVSEFSSM KLPQCVTL VLYRGFSL SIEVGLVSV CINGVWTW VMMKILQV
ILTVILGVTKLEYVIKV MLEWPRALV GLSRYVARL FLTRITLTI HLCWNVKLV GIAAGLALL GLQDCTML
TGAPVHYST LLDFYQMDL VLDPVFIRC VLPDFIRC AGGIGAVV VLNGLLALV ALGLLLPV GIGGVLAA
GAGIGVAVL IAGIGILAI LILIGILAAV VDGIGITL GAGIGAVL AAGIGIIQI QAGIGILLA
KARDPHSGH KACDPHSGH ACDPHSGV SLYNTVATL RLGPRAFVT NLVPVMATV GLHCYQEVQ PLQHQPQIV
AVFDKSVTL LLDVFVPSV FLVKSPPHV NLLGQPQHI LLGLNQFEV LTDFGSQW VLEWFQDSR TINAWKVVV
GLCLTVAML FIDSYICQV LSSAVGVQW VGMAGVQSF LLLWTLVVL SVDRDRLARL LLMDCSGSI CLTSTQVL
VLUHDLLEA LMWITQCPF SLLMWITC QLSSLLWIT LLGATCMFV RLPFLTSLVD YMDGTSQV FLTPKLC
ISNDCAQV KVTGNDNEP SBYDFVWV FLGALLLA VLPDFVIRFWM MAKGPIV SLLELEEEV SLRSFSGA
YTAFTIPSJ LRMKFQDSV LTRFRFCSC GLWPGARAYA RLDQETELV SLD KGIDFY VLDQSVVSL RLNMFTPQI
NMPFTYIGV LMIPIQNDL TILQSHHV SVLVTITTV LQWASLAV LLLKFLPHL STAPPHNV MNLLTQL
VVLGNVFGI ILHNGAYSL MIVKGCWMI MGLHTMVEV MGLHTMVEV SLADTNLSA LLWAARPRL GVQMTMQ
GLYGDMEHML KMSLHVFSL YQLQVFGE MLEMAQELAVL IMQEMALAF VYDGREHTV YLSGANNLNL RMFTPAPYL
EAAAGIULTL TLDSQVMSL STPPPTGRV KVAELHHFL IMIGVLVGV ALCRGWGLL LLFGAQVQO VLLCESTAV
YLSTAFARV YLLELMDVLL SLDDYNHLV RTLDKVEV GLPVEVQLQV KLIANNTRF FIYAGSLSA KLSCNTRL
FLDEFMGEL ALQPGTALL VLDGLDVLV SLYSFEPEE ALYVDLSSF SLLQHLIGL ELITLHSFLLL MINAYDLK
AAGIGILTV FLPSDFDFS SVRDRILARL SLREWLLRLI LLASWLITL AAGIGILTV AVPDIEFPL QAYDGKYI
AAGIGILTV FLPSDFDFS AAGIGILTV FLPSDFDFS AAGIGILTV FLWGPRAFLV ETVEQSVN VTLWQRPLV
How is mutual information calculated?
- Information content was calculated as:
  - Gives information in a single position
    \[
    I = \sum_a p_a \log\left(\frac{p_a}{q_a}\right)
    \]
- Similar relation for mutual information:
  - Gives mutual information between two positions
    \[
    I = \sum_{a,b} p_{ab} \log\left(\frac{p_{ab}}{p_a \cdot p_b}\right)
    \]
Mutual information. Example

Knowing that you have $G$ at $P_1$ allows you to make an educated guess on what you will find at $P_6$. $P(V_6) = 4/10$. $P(V_6 | G_1) = 1.0!$

\[ I = \sum_{a,b} p_{ab} \log \left( \frac{p_{ab}}{p_a \cdot p_b} \right) \]

- $P(G_1) = 2/10 = 0.2$, ..
- $P(V_6) = 4/10 = 0.4$, ..
- $P(G_1, V_6) = 2/10 = 0.2$
- $P(G_1) * P(V_6) = 8/100 = 0.08$

\[ \log(0.2/0.08) > 0 \]
Mutual information

313 binding peptides

313 random peptides
Higher order sequence correlations

- Neural networks can learn higher order correlations!
  - What does this mean?

Say that the peptide needs one and only one large amino acid in the positions P3 and P4 to fill the binding cleft

How would you formulate this to test if a peptide can bind?

\[
\begin{align*}
S \text{ S} & \Rightarrow 0 \\
L \text{ S} & \Rightarrow 1 \\
S \text{ L} & \Rightarrow 1 \\
L \text{ L} & \Rightarrow 0
\end{align*}
\Rightarrow \text{ XOR function}
Neural networks

- Neural networks can learn higher order correlations

XOR function:
- 0 0 => 0
- 1 0 => 1
- 0 1 => 1
- 1 1 => 0

No linear function can separate the points
Error estimates

<table>
<thead>
<tr>
<th>XOR</th>
<th>Predict</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 =&gt; 0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 0 =&gt; 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0 1 =&gt; 1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1 1 =&gt; 0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Mean error: 1/4
Neural networks

Linear function

\[ y = x_1 \cdot v_1 + x_2 \cdot v_2 \]
Neural networks with a hidden layer

\[ O = \frac{1}{1 + \exp(-o)} \]

\[ O = \sum_{i=1}^{N} x_i \cdot w_i + t = \sum_{i=1}^{N+1} x_i \cdot w_i \]

\[ x_N = 1 \]
Neural networks
How does it work?
Ex. Input is (0 0)

\[
O = \frac{1}{1 + \exp(-o)}
\]

\[
O = \sum x_i \cdot w_i
\]
Neural networks. How does it work?

Hand out
Neural networks \((1\ 0 \&\& \ 0\ 1)\)

\[
O = \frac{1}{1 + \exp(-o)}
\]

\[
O = \sum x_i \cdot w_i
\]
Neural networks (1 1)

\[
O = \frac{1}{1 + \exp(-o)}
\]

\[
O = \sum x_i \cdot w_i
\]
What is going on?

\[ f_{\text{XOR}}(x_1, x_2) = -2 \cdot x_1 \cdot x_2 + (x_1 + x_2) = -y_2 + y_1 \]

XOR function:
0 0 \Rightarrow 0
1 0 \Rightarrow 1
0 1 \Rightarrow 1
1 1 \Rightarrow 0
What is going on?

\[ y_1 = x_1 + x_2 \]

\[ y_2 = 2 \cdot x_1 \cdot x_2 \]
Network with more inputs and hidden units

Input layer: \( I_1, I_2, \ldots, I_N \)

Hidden layer: \( H_1, H_2, \ldots, H_M \)

Output layer: \( O_1, \ldots, O_L \)

Connections and Feed forward
Pattern Association

Pattern association.
Input is associated with output.
Classification, categorization, discrimination.

Goal: Find weights and thresholds.
Method: Training, not programming.

Training examples: \( I_j^\alpha (\alpha = 1, 2, \ldots; j = 1, 2, \ldots, N) \).

Desired targets: \( T_i^\alpha (\alpha = 1, 2, \ldots; i = 1, 2, \ldots, M) \).

Actual output: \( O_i^\alpha (\alpha = 1, 2, \ldots; i = 1, 2, \ldots, M) \).

Define quadratic error

\[
E = \frac{1}{2} \sum_{\alpha, i} (O_i^\alpha - T_i^\alpha)^2
\]

Measures least square deviation between desired result and actual output.

Minimize error by varying weights and thresholds.

\[
\delta w = -\epsilon \frac{\partial E}{\partial w}
\]

Gradient descent method.
Training and error reduction

\[ \delta w = -\epsilon \frac{\partial E}{\partial w} \]

E

W

local minimum

global minimum

\( \epsilon \)
Training and error reduction

\[ \delta w = -\varepsilon \frac{\partial E}{\partial w} \]

local minimum

global minimum
Training and error reduction

\[ \delta w = -\epsilon \frac{\partial E}{\partial w} \]

Size matters
Neural network training

- A Network contains a very large set of parameters
  - A network with 5 hidden neurons predicting binding for 9meric peptides has $9 \times 20 \times 5 = 900$ weights
  - 5 times as many weights as a matrix-based method
- Over fitting is a problem
- Stop training when test performance is optimal (use early stopping)
Neural network training. Cross validation

Cross validation

Train on 4/5 of data
Test on 1/5

=>
Produce 5 different neural networks each with a different prediction focus
Neural network training curve

Maximum test set performance
Most capable of generalizing
Network training

• Encoding of sequence data
  - Sparse encoding
  - Blosum encoding
  - Sequence profile encoding
Sparse encoding

<table>
<thead>
<tr>
<th>Inp Neuron</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAcid</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>R</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>N</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Q</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
BLOSUM encoding (Blosum50 matrix)

|   | A | R | N | D | C | Q | E | G | H | I | L | K | M | F | P | S | T | W | Y | V |
| A | 4 | -1 | -2 | -2 | 0 | -1 | -1 | 0 | -2 | -1 | -1 | -1 | -1 | -2 | -1 | 1 | 0 | -3 | -2 | 0 |
| R | -1 | 5 | 0 | -2 | -3 | 1 | 0 | -2 | 0 | -3 | -2 | 2 | -1 | -3 | -2 | -1 | -1 | -3 | -2 | -3 |
| N | -2 | 0 | 6 | 1 | -3 | 0 | 0 | 0 | 0 | 1 | -3 | -3 | 0 | -2 | -3 | -2 | 1 | 0 | -4 | -2 | -3 |
| D | -2 | -2 | 1 | 6 | -3 | 0 | 2 | -1 | -1 | -3 | -4 | -1 | -3 | -3 | -1 | 0 | -1 | -4 | -3 | -3 |
| C | 0 | -3 | -3 | -3 | 9 | -3 | -4 | -3 | -1 | -1 | -3 | -1 | -2 | -3 | -1 | -1 | -2 | -2 | -1 |
| Q | -1 | 1 | 0 | 0 | -3 | 5 | 2 | -2 | 0 | -3 | -2 | 1 | 0 | -3 | -1 | 0 | -1 | -2 | -1 | -2 |
| E | -1 | 0 | 0 | 2 | -4 | 2 | 5 | -2 | 0 | -3 | -3 | 1 | -2 | -3 | -1 | 0 | -1 | -3 | -2 | -2 |
| G | 0 | -2 | 0 | -1 | -3 | -2 | -2 | 6 | -2 | -4 | -4 | -2 | -3 | -2 | 0 | -2 | -2 | -3 | -3 |
| H | -2 | 0 | 1 | -1 | -3 | 0 | 0 | -2 | 8 | -3 | -3 | -1 | -2 | -1 | -2 | -1 | -2 | -2 | 2 | -3 |
| I | -1 | -3 | -3 | -3 | -1 | -3 | -3 | -4 | -3 | 4 | 2 | -3 | 1 | 0 | -3 | -2 | -1 | -3 | -1 | 3 |
| L | -1 | -2 | -3 | -4 | -1 | -2 | -3 | -4 | -3 | 2 | 4 | -2 | 2 | 0 | -3 | -2 | -1 | -2 | -1 | 1 |
| K | -1 | 2 | 0 | -1 | -3 | 1 | 1 | -2 | -1 | -3 | -2 | 5 | -1 | -3 | -1 | 0 | -1 | -3 | -2 | -2 |
| M | -1 | -1 | -2 | -3 | -1 | 0 | -2 | -3 | -2 | 1 | 2 | -1 | 5 | 0 | -2 | -1 | -1 | -1 | -1 | 1 |
| F | -2 | -3 | -3 | -3 | -2 | -3 | -3 | -1 | 0 | 0 | -3 | 0 | 6 | -4 | -2 | -2 | 1 | 3 | -1 |
| P | -1 | -2 | -2 | -1 | -3 | -1 | -1 | -2 | -2 | -3 | -3 | -1 | -2 | -4 | 7 | -1 | -1 | -4 | -3 | -2 |
| S | 1 | -1 | 1 | 0 | -1 | 0 | 0 | 0 | 0 | -1 | -2 | -2 | 0 | -1 | -2 | -1 | 4 | 1 | -3 | -2 | -2 |
| T | 0 | -1 | 0 | -1 | -1 | -1 | -2 | -2 | -1 | -1 | -1 | -1 | -2 | -1 | 1 | 5 | -2 | -2 | 0 |
| W | -3 | -3 | -4 | -4 | -2 | -2 | -3 | -2 | -2 | -3 | -2 | -3 | -1 | 1 | -4 | -3 | -2 | 1 | 1 | 2 | -3 |
| Y | -2 | -2 | -2 | -3 | -2 | -1 | -2 | -3 | 2 | -1 | -1 | -2 | -1 | 3 | -3 | -2 | -2 | 2 | 7 | -1 |
| V | 0 | -3 | -3 | -3 | -1 | -2 | -2 | -3 | -3 | 3 | 1 | -2 | 1 | -1 | -2 | -2 | 0 | -3 | -1 | 4 |
Sequence encoding (continued)

• Sparse encoding
  - $V$: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
  - $L$: 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
  - $V \cdot L = 0$ (unrelated)

• Blosum encoding
  - $V$: 0 -3 -3 -3 -1 -2 -2 -3 -3 3 1 -2 1 -1 -2 -2 0 -3 -1 4
  - $L$: -1 -2 -3 -4 -1 -2 -3 -4 -3 2 4 -2 2 0 -3 -2 -1 -2 -1 1
  - $V \cdot L = 0.88$ (highly related)
  - $V \cdot R = -0.08$ (close to unrelated)
The Wisdom of the Crowds

• The Wisdom of Crowds. Why the Many are Smarter than the Few. James Surowiecki

One day in the fall of 1906, the British scientist Fracis Galton left his home and headed for a country fair... He believed that only a very few people had the characteristics necessary to keep societies healthy. He had devoted much of his career to measuring those characteristics, in fact, in order to prove that the vast majority of people did not have them. ... Galton came across a weight-judging competition...Eight hundred people tried their luck. They were a diverse lot, butchers, farmers, clerks and many other no-experts...The crowd had guessed ... 1.197 pounds, the ox weighted 1.198
Network ensembles

• No one single network with a particular architecture and sequence encoding scheme, will constantly perform the best

• Also for Neural network predictions will enlightened despotism fail
  - For some peptides, BLOSUM encoding with a four neuron hidden layer can best predict the peptide/MHC binding, for other peptides a sparse encoded network with zero hidden neurons performs the best
  - Wisdom of the Crowd
    • Never use just one neural network
    • Use Network ensembles
Evaluation of prediction accuracy

<table>
<thead>
<tr>
<th></th>
<th>Motif</th>
<th>Sparse</th>
<th>BLOSUM</th>
<th>ENS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pear</td>
<td>0.76</td>
<td>0.88</td>
<td>0.91</td>
<td>0.92</td>
</tr>
<tr>
<td>Aroc</td>
<td>0.92</td>
<td>0.97</td>
<td>0.97</td>
<td>0.98</td>
</tr>
</tbody>
</table>

**ENS**: Ensemble of neural networks trained using sparse, Blosum, and weight matrix sequence encoding
Applications of artificial neural networks

• Talk recognition
• Prediction of protein secondary structure
• Prediction of Signal peptides
• Post translation modifications
  - Glycosylation
  - Phosphorylation
• Proteasomal cleavage
• MHC:peptide binding
NETtalk
(T. Sejnowski and C. Rosenberg, 1987)

Mary had a little lamb

Three of the a’s must be pronounced differently! Reading aloud is a context sensitive cognitive skill.
Prediction of protein secondary structure

• Benefits
  - General applicable
  - Can capture higher order correlations
  - Inputs other than sequence information

• Drawbacks
  - Needs many data (different solved structures).
    • However, these does exist today (more than 2500 solved structures with low sequence identity/high resolution)
  - Complex method with several pitfalls
Secondary Structure Elements

\[ \beta \text{-strand} \]

Helix

Bend

Turn
Sparse encoding of amino acid sequence windows
Why so many networks?

Q3 is the overall accuracy
Why not select the best?
What have we learned?

- Neural networks are not so bad as their reputation
- Neural networks can deal with higher order correlations
- Be careful when training a neural network
  - Overfitting is an important issue
  - Always use cross validated training