Protein Secondary Structure Prediction

part of “Bioinformatik von RNA- und Proteinstrukturen”

Sonja Prohaska

Computational EvoDevo
University Leipzig

Leipzig, SS 2011

Protein Secondary Structure Prediction

- the goal is the prediction of the secondary structure conformation which is local
- each amino acid will be assigned one of the secondary structure conformations
- the problem is expected to be easier then tertiary structure prediction
- afterwards: predict arrangement of secondary structure elements
- classify the fold according to SCOP (Structural Classification of Proteins)

Secondary Structure Conformations

as easy as...

- 'H' for helix
- 'E' for extended β-sheet
- '-' for "other"

as complicated as...

- 'H' for α-helix (period 4)
- 'I' for π-helix (period 5)
- 'G' for 3_{10}-helix (period 3)
- 'E' for extended β-sheet
- 'B' for β-bridge
- 'T' for turn
- 'S' for bend (region of high curvature)
- 'L' for loop
- ' ' for "other"

Chou-Fasman Method

Chou and Fasman developed a simple prediction algorithm in the 1970s. The parameter table is calculated as follow:

- it is based on the probability with which each an amino acid a of type i will appear in a helix H, strand E or turn T
- propensities are **calculated from** a set of **proteins with known structure**:

$$p(a_i, H) = 1 + \log \frac{f(a_i|H)}{E(a_i|H)} = \frac{\frac{F(a_i,H)}{F(H)}}{\frac{F(a_i)}{\sum_{i=1}^{20} F(a_i)}}$$

- $f(a_i|H)$ is the observed count for amino acid a_i in helices
- $E(a_i|H)$ is the expected count for amino acid a_i in helices
- $F(a_i, H)$ is the count for amino acid a_i in helix conformation
- $F(H)$ is the count for amino acids in helix conformation
- $F(a_i)$ is the count for amino acid a_i
- $\sum_{i=1}^{20} F(a_i)$ is the total number of amino acids

Chou-Fasman Method – conformation parameters

Amino acid	$p(a_i, H)$	$p(a_i, E)$	$p(a_i, T)$	$f(j)$	$f(j+1)$	$f(j+2)$	$f(j+3)$
Alanine	1.42	0.83	0.66	0.06	0.076	0.035	0.058
Arginine	0.98	0.93	0.95	0.070	0.106	0.099	0.085
Aspartic Acid	1.01	0.54	1.46	0.147	0.110	0.179	0.081
Asparagine	0.67	0.89	1.56	0.161	0.083	0.191	0.091
Cysteine	0.70	1.19	1.19	0.149	0.050	0.117	0.128
Glutamic Acid	1.39	1.17	0.74	0.056	0.060	0.077	0.064
Glutamine	1.11	1.10	0.98	0.074	0.098	0.037	0.098
Glycine	0.57	0.75	1.56	0.102	0.085	0.190	0.152
Histidine	1.00	0.87	0.95	0.140	0.047	0.093	0.054
Isoleucine	1.08	1.60	0.47	0.043	0.034	0.013	0.056
Leucine	1.41	1.30	0.59	0.061	0.025	0.036	0.070
Lysine	1.14	0.74	1.01	0.055	0.115	0.072	0.095
Methionine	1.45	1.05	0.60	0.068	0.082	0.014	0.055
Phenylalanine	1.13	1.38	0.60	0.059	0.041	0.065	0.065
Proline	0.57	0.55	1.52	0.102	0.301	0.034	0.068
Serine	0.77	0.75	1.43	0.120	0.139	0.125	0.106
Threonine	0.83	1.19	0.96	0.086	0.108	0.065	0.079
Tryptophan	1.08	1.37	0.96	0.077	0.013	0.064	0.167
Tyrosine	0.69	1.47	1.14	0.082	0.065	0.114	0.125
Valine	1.06	1.70	0.50	0.062	0.048	0.028	0.053

- most frequent in **helices**: glutamate, methionine, and alanine
- most frequent in **strands**: valine and isoleucine
- most frequent in **turns**: proline and glycine

Chou-Fasman Method – continued

- find *nucleation* region
  - **helix**: 4 out of 6 contiguous residues with $p(H) > 1.00$
  - **strand**: 3 out of 5 contiguous residues with $p(E) > 1.00$
- extend the nucleation regions in both directions
  - **helix**: until four-residue window from position j to $j+3$ has $\frac{p_j(H)+p_{j+1}(H)+p_{j+2}(H)+p_{j+3}(H)}{4} < 1.0$
  - **strand**: analogous
- minimum length for structure conformations
  - **helix**: 5 residues
  - **strand**: 3 residues
- resolve regions with helix and strand annotation
  - **helix**: if average $p(H)$ for the region $>$ average $p(E)$
  - **strand**: if average $p(E)$ for the region $>$ average $p(H)$

Chou-Fasman Method – continued

- predict β-turns
  - calculate $p(t) = f(j) \times f(j+1) \times f(j+2) \times f(j+3)$ where $f(x)$ are the bend frequences in the position x of the β-turn (see parameter table)
  - calculate $p(S) = \frac{p_j(S)+p_{j+1}(S)+p_{j+2}(S)+p_{j+3}(S)}{4}$ for $S \in \{H, E, T\}$
  - if $p(t) > 0.000075$ and
  - if average $p(T) > 1.0$ and
  - if $p(H) < p(T) > p(E)$ for the same window
  - then a β-turn is predicted

The GOR Method

by **G**arnier, **O**sguthorpe and **R**obson (late 1970s) improves the secondary structure prediction by introducing the **conditional probabilities** of an amino acid to form a particular secondary structure element, given that its neighbors already possess that structure.

- works on windows of size 17, $a_{j-8}, ..., a_j, ..., a_{j+8}$
- based on data: collecting statistics for 20^{17} sequences
- instead assume: the central residue depends on its neighbors but the neighbors are independent of each other
- four matrices, each 17×20, reflect the probabilities of the central residue being in a helical, sheet, turn, or coil conformation
- uses Bayesian statistics $P(S|a) = P(S, a)/P(a)$

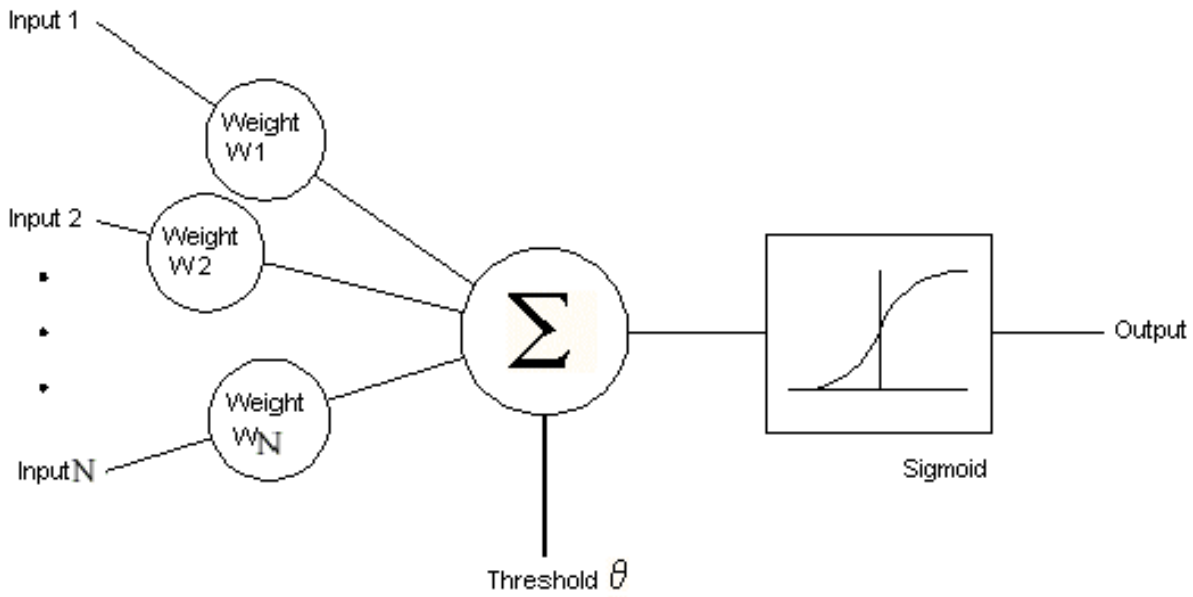
The GOR Method – continued

- calculates the information for the joint occurence of the structure S and the amino acid a
- $I(S,a) = \log P(S|a)/P(S)$ gives $I(S,a) = \log f_{S,a}/f_S$
- hypothesis residue a is in structure S: $I(S,a)$
- hypothesis residue a is in a different conformation: $I(notS,a)$
- the difference is

$$I(\Delta S,a) = I(S,a) - I(notS,a) = \log \frac{f_{S,a}}{f_S} - \log \frac{1-f_{S,a}}{1-f_S}$$

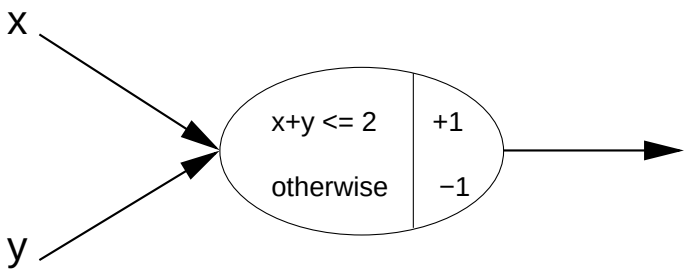
- The values $I(\Delta S,a)$ are summed over $a_{j-8},...,a_{j-1},a_{j+1},...,a_{j+8}$ to approximate the value for $I(\Delta S_j,(a_{j-8},...,a_{j+8}))$

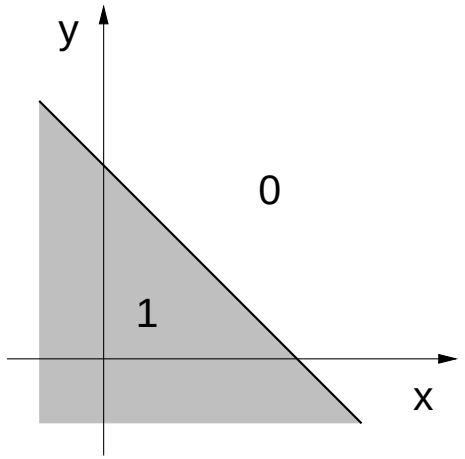
Neural Networks – the Perceptron

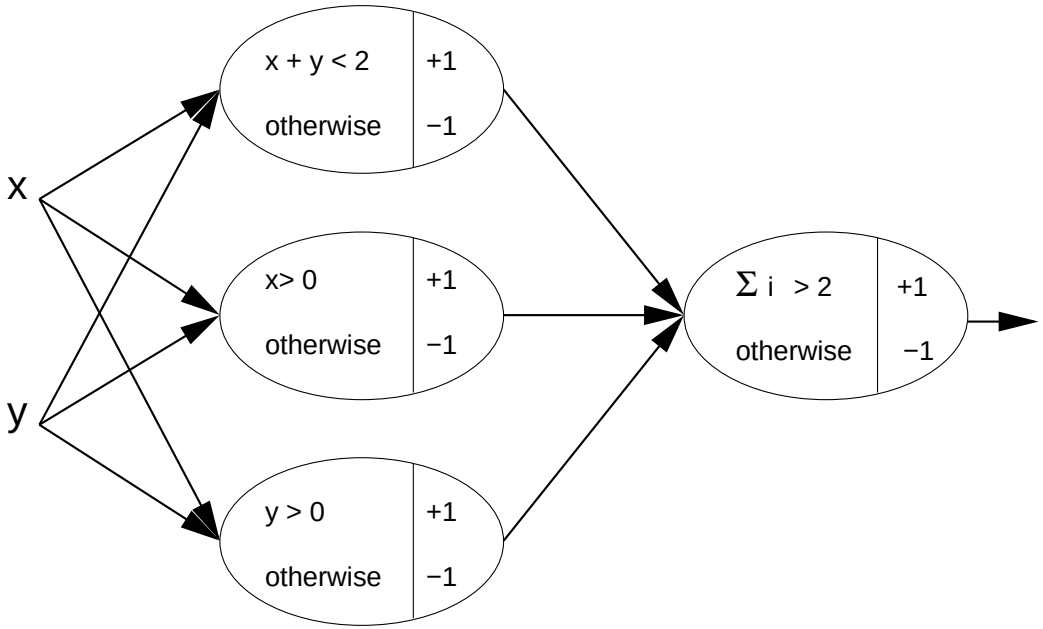


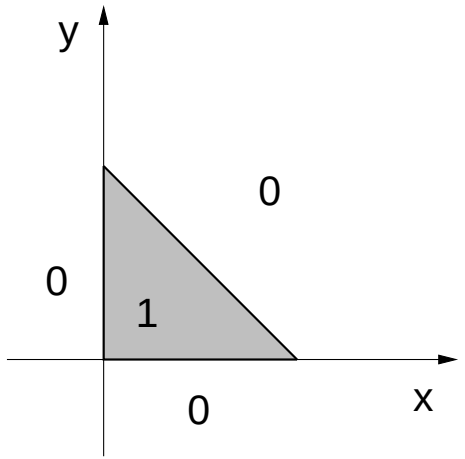
- inputs: $x_1, ..., x_n$
- weigths: $w_1, ..., w_n$
- the perceptron calculates the sum: $a = \sum_{i=1}^{n} w_i x_i$
- if $a >= \theta$ the output is $+1$
- if $a < \theta$ the output is -1

Classification by Perceptrons









Goal and Training

If the weigths are not know in advance, the perceptron(s) must be trained.

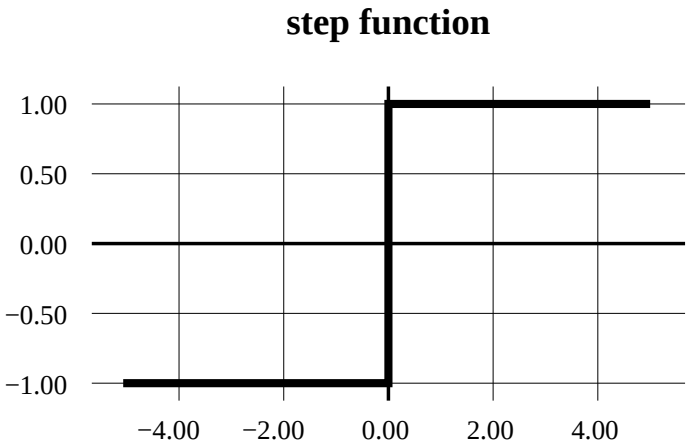
Goal: find weigthes such that the perceptron returns the correct answere for all cases in the training set.

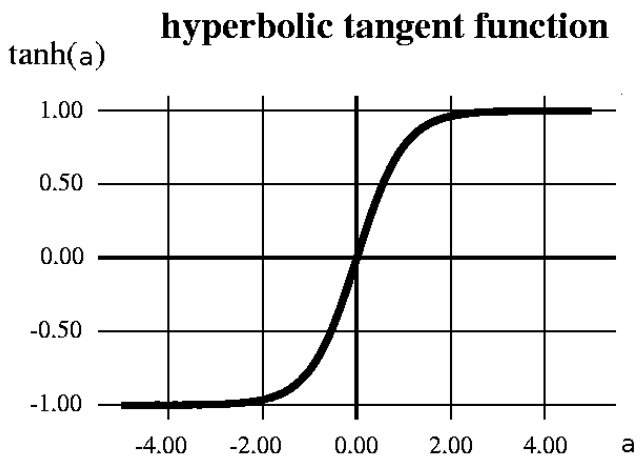
- take the input vector $\vec{x}$ and the weigths $\vec{w}$
- multiply the vectors and use θ to categorize the output as positive or negative
- if the perceptron is wrong, update the weigths:
  - calculate $\vec{w} - \vec{x}$ if false positive
  - calculate $\vec{w} + \vec{x}$ if false negative
- repeat until the perceptron classifies all examples in the training set correctly

Proof that the training procedure converges under the assumption that a set of weigths exists that labels all examples correctly.

Step function or sigmoidal curve?

Use sigmoidal curve instead of step function because it is differentiable.





$$\Theta(a) = \begin{cases} +1 & \text{if } a >= \theta \\ -1 & : \text{otherwise} \end{cases}$$

$$\sigma(a) = \tanh(a)$$

The output perceptron therefore returns a real number between -1 and $+1$ instead of either -1 or $+1$.

Error Minimization

Goal: Find a $\vec{w}$ that minimizes $E(\vec{w})$.

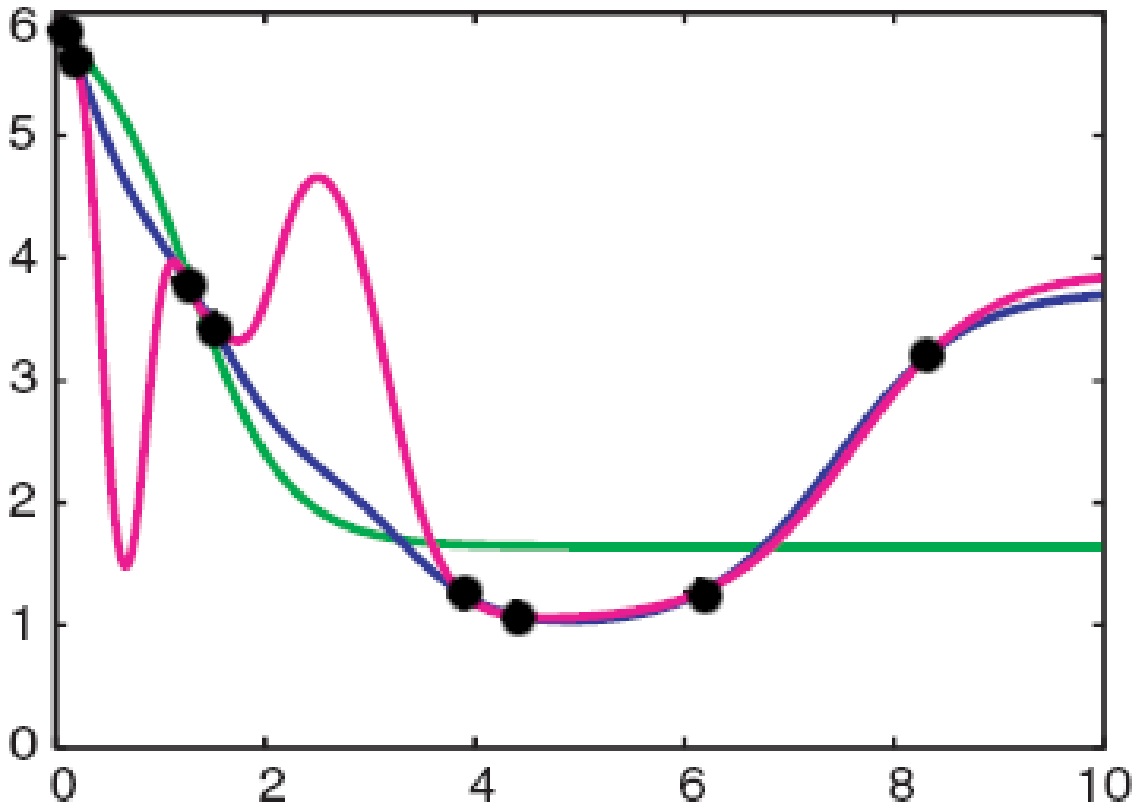
Let T be the set of trainings examples, and $t(\vec{x})$ the target value of example $\vec{x} \in T$

$$E(\vec{w}) = \sum_{\vec{x} in T} \frac{1}{2}(\sigma(\vec{w}\vec{x}) - t(\vec{x}))^2$$

This is the formular for “squared error” which is used to quantify how close the observed value $\sigma(\vec{w}\vec{x})$ is to the target value $t(\vec{x})$.

Overfitting

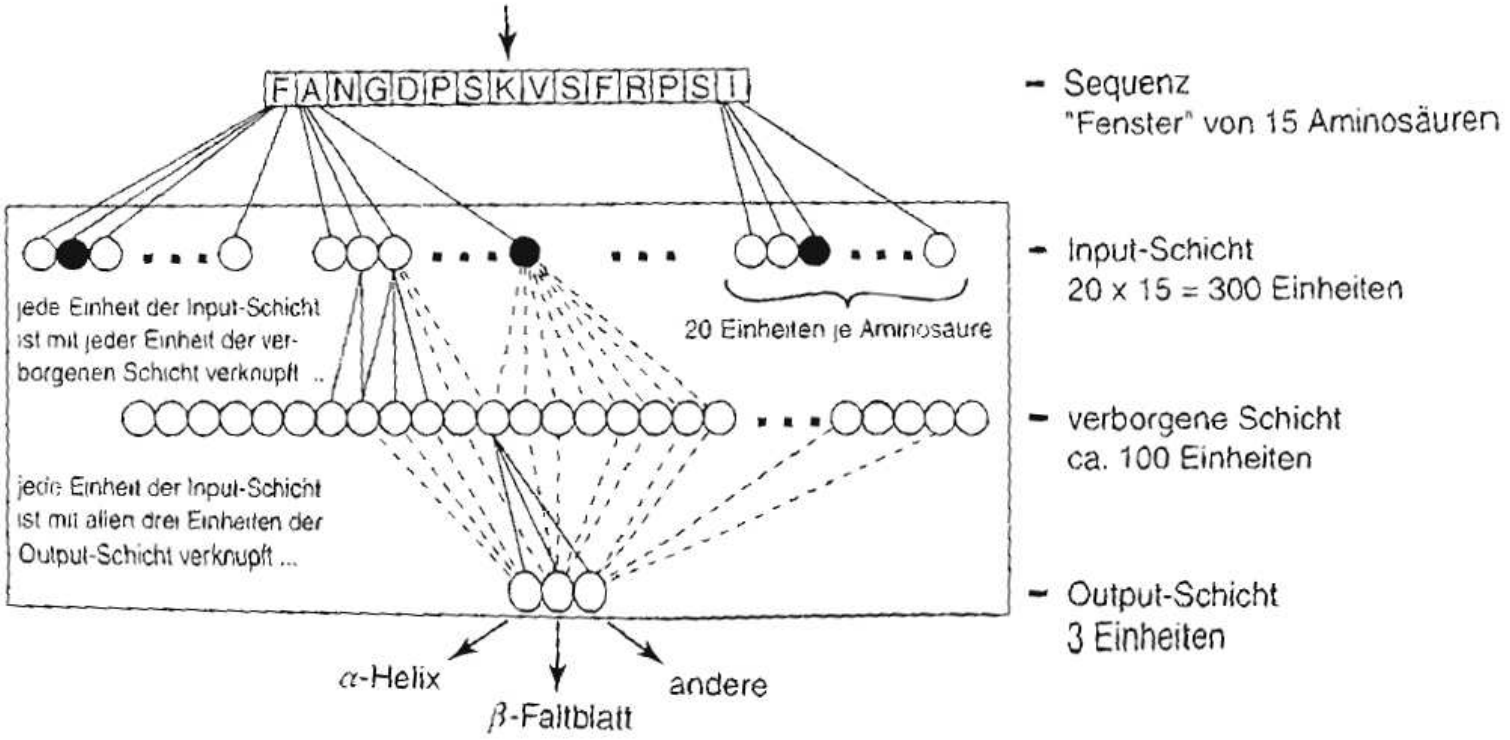
Task: fit a line to the set of points with a neural network



- green line – not a good fit
- blue line – good fit
- purple line – overfitted, no data points support the fancy shape between 0 and 4

Neuronal Network Method

How are neuronal networks applied to predict protein secondary structures?



Secondary Structure Prediction from Multiple Alignments

How good are the methods?

- Chou & Fasman – 55%
- GOR – 65%
- neural network on multiple sequences (PHD) – 72%
- weigthed consensus of many methods – moderate improvement
- database search method (PSI-BLAST) – 77%

Example

Literature