Protein Structure Prediction
Protein Structure Representation

CPK: hard sphere model
Ball-and-stick
Cartoon
Degrees of Freedom in Proteins

**Bond length**

1

2

**Bond angle**

**Dihedral angle**

1

2

3

4
**Backbone**: 3 angles per residue: $\varphi$, $\phi$ and $\omega$

**Sidechain**: 1 to 7 angles, $\chi$; each $\chi$ has 3 favored values: $60^\circ$, $-60^\circ$, $180^\circ$. 

*Protein Structure Representation*
Ramachandran Plots

All residues, but glycine

Glycine

To compare two sets of points (atoms) \( A = \{a_1, a_2, \ldots a_N\} \) and \( B = \{b_1, b_2, \ldots, b_N\} \):

- Define a 1-to-1 correspondence between \( A \) and \( B \)
  
  for example, \( a_i \) corresponds to \( b_i \), for all \( i \) in \([1,N]\)

- Compute RMS as:

\[
RMS(A, B) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} d(a_i, b_i)^2}
\]

\( d(A_i, B_i) \) is the Euclidian distance between \( a_i \) and \( b_i \).
Root Mean Square Distance (RMSD)

- Simplified problem: we know the correspondence between set A and set B
- We wish to compute the rigid transformation $T$ that best align $a_1$ with $b_1$, $a_2$ with $b_2$, ..., $a_N$ with $b_N$
- The error to minimize is defined as:

$$
\varepsilon = \min_T \sum_{i=1}^{N} \left\| T(a_i) - b_i \right\|^2
$$

Old problem, solved in Statistics, Robotics, Medical Image Analysis, ...

...
Structural Bioinformatics: Proteins

Proteins: Sources of Structure Information

Proteins: Homology Modeling

Proteins: Secondary structure prediction
Structural Bioinformatics: Proteins

Proteins: Sources of Structure Information

Proteins: Homology Modeling

Proteins: Secondary structure prediction

Proteins: Ab initio prediction
Proteins: Finding the Primary Structure

Methods for finding the sequence of a protein:

- Translating gene sequence
  - For proteins from prokaryotes, direct translation
  - For proteins from eukaryotes, we need the sequence of mRNA or cDNA

- Edman degradation
  limited to “small” proteins, up to 50 amino acids for automated sequencer

- Mass spectrometry
Proteins: Finding the Primary Structure

EDMAN DEGRADATION

Phenyl isothiocyanate or PITC

(Phenyl isothiocyanate or PITC)

Trifluoroacetic acid, TFA

(Trifluoroacetic acid, TFA)
Proteins: Finding the Tertiary Structure

Methods for finding the 3D structure of a protein:

- Circular Dichroism
  (low resolution; provides information on secondary structure)

- X-ray crystallography
  (high resolution; finds structure of a protein in a crystal)

- NMR spectroscopy
  (high resolution; finds structure of a protein in solution)
Proteins: Circular Dichroism

Circular dichroism (CD) spectroscopy measures differences in the absorption of left-handed polarized light versus right-handed polarized light which arise due to structural asymmetry.

Different secondary structures in proteins have different CD spectra as they have different asymmetry.

CD therefore can detect secondary structures in proteins.
Proteins: X-ray Crystallography

Bragg’s Law:

\[ 2d \sin(\theta) = n\lambda \]

From the “pattern of diffraction”, i.e. the maximum of intensities observed, we can find the angles of diffraction and for each angle we get the corresponding \( d \) using Bragg’s law.
General principle of X-ray crystallography applied to proteins:

1) We need a crystal

2) From the diffraction pattern, we get the crystal organization

3) From the diffraction intensities, we get the electron densities

4) Once the electron density map we fit a structure that matches with this density

5) From the atomic model, we can compute a theoretical diffraction map; if it matches with the experimental one, we are done; otherwise refine
Getting the Diffraction Pattern

Rosalyn Franklin: Diffraction pattern for DNA
From Diffraction to Electron Density Map

One hidden problem: diffraction patterns provide intensities; for Fourier transform, need intensity and phase. A significant step in X-ray crystallography is the solve the “phase problem”.
Fitting the structure: Influence of the resolution

2.6 Å resolution

1.2 Å resolution
Resolution of X-ray structures

<table>
<thead>
<tr>
<th>Resolution (Å)</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;4.0</td>
<td>Individual coordinates meaningless</td>
</tr>
<tr>
<td>3.0 - 4.0</td>
<td>Fold possibly correct, but errors are very likely.</td>
</tr>
<tr>
<td>2.5 - 3.0</td>
<td>Fold likely correct except that some surface loops might be mismodelled.</td>
</tr>
<tr>
<td>2.0 - 2.5</td>
<td>Many small errors can normally be detected. Fold normally correct and number of errors in surface loops is small. Water molecules and small ligands become visible.</td>
</tr>
<tr>
<td>1.5 - 2.0</td>
<td>Many small errors can normally be detected. Folds are extremely rarely incorrect, even in surface loops.</td>
</tr>
<tr>
<td>0.5 - 1.5</td>
<td>In general, structures have almost no errors at this resolution. geometry studies are made from these structures.</td>
</tr>
</tbody>
</table>

(http://en.wikipedia.org/wiki/Resolution_(electron_density)
Large molecular assemblies: X-ray crystallography and Cryo-EM

**X-ray structure**

(180 copies of the same protein)

(Norwalk virus: http://www.bcm.edu/molvir/norovirus)
Large molecular assemblies: X-ray crystallography and Cryo-EM

**Cryo-EM:**

- Microscopy technique; as such, do not need crystal (closer to physiological conditions)

- Not high-resolution enough to provide atomic details; used in combination with modeling
Structural Bioinformatics: Proteins

Proteins: Sources of Structure Information

Proteins: Homology Modeling

Proteins: Secondary structure prediction
Why we need Homology Modeling?

❖ Aim to solve the structure of all proteins: this is too much work experimentally!

❖ Solve enough structures so that the remaining structures can be inferred from those experimental structures

❖ The number of experimental structures needed depend on our abilities to generate a model.
Sequence Space

Structure Space
Proteins with known structures

Unknown proteins

Why we need Homology Modeling?
Why does Homology Modeling Work?


High sequence identity

High structure similarity
Homology Modeling:
How it works
Homology Modeling: How it works

- Find template
- Align target sequence with template
- Generate model:
  - add loops
  - add sidechains
- Refine model
Homology Modeling: Input

The query, also called target sequence

The template structure and sequence  
(need high quality structure)

The sequence alignment between query and template sequence  
(Probably the most important input!)
Wallner B, Elofsson A.
All are not equal: a benchmark of different homology modeling programs.
Homology Modeling: Which program to use?

1) **Web service: SwissModel**
   http://swissmodel.expasy.org/
   
   3 modes:
   - fully automatic
   - “Alignment mode”: you provide your own target-template alignment
   - “Project mode”: provides an environment to edit alignment

2) **Software: Modeller**
   http://www.salilab.org/modeller/

   Probably the best maintained software the homology modeling
Do not start a homology modeling project before checking...

- **Swiss-Model** repository
  (http://swissmodel.expasy.org/repository/)
  - Companion to the Swiss-Model tools – over 2.0 million models of protein domains

- **ModBase**
  (http://modbase.compbio.ucsf.edu/)
  - Companion to **Modeller**

<table>
<thead>
<tr>
<th>ModBase Contents</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Models</td>
<td>38,284,888</td>
</tr>
<tr>
<td>Number of Unique Sequences modeled</td>
<td>6,492,640</td>
</tr>
<tr>
<td>Unique sequences attempted</td>
<td>8,072,845</td>
</tr>
<tr>
<td>Number of PDB chains</td>
<td>85,448</td>
</tr>
<tr>
<td>Statistics updated at Fri Jan 14 22:01:34 PST 2022</td>
<td></td>
</tr>
</tbody>
</table>
Structural Bioinformatics: Proteins

Proteins: Sources of Structure Information

Proteins: Homology Modeling

Proteins: Secondary structure prediction
Secondary Structure Prediction

- Given a protein sequence $a_1a_2…a_N$, secondary structure prediction aims at defining the state of each amino acid $a_i$ as being either H (helix), E (extended=strand), or O (other) (Some methods have 4 states: H, E, T for turns, and O for other).

- The quality of secondary structure prediction is measured with a “3-state accuracy” score, or $Q_3$. $Q_3$ is the percent of residues that match “reality” (X-ray structure).
Determine Secondary Structure positions in known protein structures using DSSP or STRIDE:


Early methods for Secondary Structure Prediction

- **Chou and Fasman**
  

- **GOR**
  
Start by computing amino acids propensities to belong to a given type of secondary structure:

\[
\begin{align*}
\frac{P(i \mid \text{Helix})}{P(i)} & \quad \frac{P(i \mid \text{Beta})}{P(i)} & \quad \frac{P(i \mid \text{Turn})}{P(i)}
\end{align*}
\]

Propensities > 1 mean that the residue type I is likely to be found in the corresponding secondary structure type.
Chou and Fasman

<table>
<thead>
<tr>
<th>Amino Acid</th>
<th>α-Helix</th>
<th>β-Sheet</th>
<th>Turn</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ala</td>
<td>1.29</td>
<td>0.90</td>
<td>0.78</td>
</tr>
<tr>
<td>Cys</td>
<td>1.11</td>
<td>0.74</td>
<td>0.80</td>
</tr>
<tr>
<td>Leu</td>
<td>1.30</td>
<td>1.02</td>
<td>0.59</td>
</tr>
<tr>
<td>Met</td>
<td>1.47</td>
<td>0.97</td>
<td>0.39</td>
</tr>
<tr>
<td>Glu</td>
<td>1.44</td>
<td>0.75</td>
<td>1.00</td>
</tr>
<tr>
<td>Gln</td>
<td>1.27</td>
<td>0.80</td>
<td>0.97</td>
</tr>
<tr>
<td>His</td>
<td>1.22</td>
<td>1.08</td>
<td>0.69</td>
</tr>
<tr>
<td>Lys</td>
<td>1.23</td>
<td>0.77</td>
<td>0.96</td>
</tr>
<tr>
<td>Val</td>
<td>0.91</td>
<td>1.49</td>
<td>0.47</td>
</tr>
<tr>
<td>Ile</td>
<td>0.97</td>
<td>1.45</td>
<td>0.51</td>
</tr>
<tr>
<td>Phe</td>
<td>1.07</td>
<td>1.32</td>
<td>0.58</td>
</tr>
<tr>
<td>Tyr</td>
<td>0.72</td>
<td>1.25</td>
<td>1.05</td>
</tr>
<tr>
<td>Trp</td>
<td>0.99</td>
<td>1.14</td>
<td>0.75</td>
</tr>
<tr>
<td>Thr</td>
<td>0.82</td>
<td>1.21</td>
<td>1.03</td>
</tr>
<tr>
<td>Gly</td>
<td>0.56</td>
<td>0.92</td>
<td>1.64</td>
</tr>
<tr>
<td>Ser</td>
<td>0.82</td>
<td>0.95</td>
<td>1.33</td>
</tr>
<tr>
<td>Asp</td>
<td>1.04</td>
<td>0.72</td>
<td>1.41</td>
</tr>
<tr>
<td>Asn</td>
<td>0.90</td>
<td>0.76</td>
<td>1.23</td>
</tr>
<tr>
<td>Pro</td>
<td>0.52</td>
<td>0.64</td>
<td>1.91</td>
</tr>
<tr>
<td>Arg</td>
<td>0.96</td>
<td>0.99</td>
<td>0.88</td>
</tr>
</tbody>
</table>

- Favors α-Helix
- Favors β-strand
- Favors turn
Predicting helices:
- find nucleation site: 4 out of 6 contiguous residues with $P(\alpha)>1$
- extension: extend helix in both directions until a set of 4 contiguous residues has an average $P(\alpha) < 1$ (breaker)
- if average $P(\alpha)$ over whole region is $>1$, it is predicted to be helical

Predicting strands:
- find nucleation site: 3 out of 5 contiguous residues with $P(\beta)>1$
- extension: extend strand in both directions until a set of 4 contiguous residues has an average $P(\beta) < 1$ (breaker)
- if average $P(\beta)$ over whole region is $>1$, it is predicted to be a strand
Position-specific parameters for turn:
Each position has distinct amino acid preferences.

Examples:
- At position 2, Pro is highly preferred; Trp is disfavored
- At position 3, Asp, Asn and Gly are preferred
- At position 4, Trp, Gly and Cys preferred

<table>
<thead>
<tr>
<th></th>
<th>f(i)</th>
<th>f(i+1)</th>
<th>f(i+2)</th>
<th>f(i+3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ala</td>
<td>0.060</td>
<td>0.076</td>
<td>0.035</td>
<td>0.058</td>
</tr>
<tr>
<td>Arg</td>
<td>0.070</td>
<td>0.106</td>
<td>0.099</td>
<td>0.085</td>
</tr>
<tr>
<td>Asp</td>
<td>0.147</td>
<td>0.110</td>
<td>0.179</td>
<td>0.081</td>
</tr>
<tr>
<td>Asn</td>
<td>0.161</td>
<td>0.083</td>
<td>0.191</td>
<td>0.091</td>
</tr>
<tr>
<td>Cys</td>
<td>0.149</td>
<td>0.050</td>
<td>0.117</td>
<td>0.128</td>
</tr>
<tr>
<td>Glu</td>
<td>0.056</td>
<td>0.060</td>
<td>0.077</td>
<td>0.064</td>
</tr>
<tr>
<td>Gln</td>
<td>0.074</td>
<td>0.098</td>
<td>0.037</td>
<td>0.098</td>
</tr>
<tr>
<td>Gly</td>
<td>0.102</td>
<td>0.085</td>
<td>0.190</td>
<td>0.152</td>
</tr>
<tr>
<td>His</td>
<td>0.140</td>
<td>0.047</td>
<td>0.093</td>
<td>0.054</td>
</tr>
<tr>
<td>Ile</td>
<td>0.043</td>
<td>0.034</td>
<td>0.013</td>
<td>0.056</td>
</tr>
<tr>
<td>Leu</td>
<td>0.061</td>
<td>0.025</td>
<td>0.036</td>
<td>0.070</td>
</tr>
<tr>
<td>Lys</td>
<td>0.055</td>
<td>0.115</td>
<td>0.072</td>
<td>0.095</td>
</tr>
<tr>
<td>Met</td>
<td>0.068</td>
<td>0.082</td>
<td>0.014</td>
<td>0.055</td>
</tr>
<tr>
<td>Phe</td>
<td>0.059</td>
<td>0.041</td>
<td>0.065</td>
<td>0.065</td>
</tr>
<tr>
<td>Pro</td>
<td>0.102</td>
<td>0.301</td>
<td>0.034</td>
<td>0.068</td>
</tr>
<tr>
<td>Ser</td>
<td>0.120</td>
<td>0.139</td>
<td>0.125</td>
<td>0.106</td>
</tr>
<tr>
<td>Thr</td>
<td>0.086</td>
<td>0.108</td>
<td>0.065</td>
<td>0.079</td>
</tr>
<tr>
<td>Trp</td>
<td>0.077</td>
<td>0.013</td>
<td>0.064</td>
<td>0.167</td>
</tr>
<tr>
<td>Tyr</td>
<td>0.082</td>
<td>0.065</td>
<td>0.114</td>
<td>0.125</td>
</tr>
<tr>
<td>Val</td>
<td>0.062</td>
<td>0.048</td>
<td>0.028</td>
<td>0.053</td>
</tr>
</tbody>
</table>
Chou and Fasman

Predicting turns:
- for each tetrapeptide starting at residue i, compute:
  - $P_{\text{Turn}}$ (average propensity over all 4 residues)
  - $F = f(i)*f(i+1)*f(i+2)*f(i+3)$

- if $P_{\text{Turn}} > P_{\alpha}$ and $P_{\text{Turn}} > P_{\beta}$ and $P_{\text{Turn}} > 1$ and $F > 0.000075$
  tetrapeptide is considered a turn.

Chou and Fasman prediction:

http:// fasta.bioch.virginia.edu/fasta_www/chofas.htm
The most successful methods for predicting secondary structure are based on neural networks. The overall idea is that neural networks can be trained to recognize amino acid patterns in known secondary structure units, and to use these patterns to distinguish between the different types of secondary structure.

Neural networks classify “input vectors” or “examples” into categories (2 or more). They are loosely based on biological neurons.
The perceptron classifies the input vector $X$ into two categories.

If the weights and threshold $T$ are not known in advance, the perceptron must be trained. Ideally, the perceptron must be trained to return the correct answer on all training examples, and perform well on examples it has never seen.

The training set must contain both type of data (i.e. with “1” and “0” output).
The perceptron

Notes:

- The input is a vector $X$ and the weights can be stored in another vector $W$.

- The perceptron computes the dot product $S = X.W$

- The output $F$ is a function of $S$: it is often set discrete (i.e. 1 or 0), in which case the function is the step function.

For continuous output, often use a sigmoid:

$$F(X) = \frac{1}{1 + e^{-X}}$$

- Not all perceptrons can be trained! (famous example: XOR)
Training a perceptron:

Find the weights $W$ that minimizes the error function:

$$E = \sum_{i=1}^{P} \left( F(X^i, W) - t(X^i) \right)^2$$

- $P$: number of training data
- $X^i$: training vectors
- $F(W, X^i)$: output of the perceptron
- $t(X^i)$: target value for $X^i$

Use steepest descent:

- compute gradient:

$$\nabla E = \left( \frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \frac{\partial E}{\partial w_3}, \ldots, \frac{\partial E}{\partial w_N} \right)$$

- update weight vector:

$$W_{\text{new}} = W_{\text{old}} - \varepsilon \nabla E$$

($\varepsilon$: learning rate)
A complete neural network is a set of perceptrons interconnected such that the outputs of some units becomes the inputs of other units. Many topologies are possible!

Neural networks are trained just like perceptron, by minimizing an error function:

$$E = \sum_{i=1}^{N_{data}} \left[ NN(X^i) - t(X^i) \right]^2$$
### PHD: Secondary structure prediction using NN

#### Biophysics: Rost and Sander

![Diagram](image)

<table>
<thead>
<tr>
<th>Protein</th>
<th>Alignments</th>
<th>Profile table</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>GGGGG</td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td>YYYY</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>IEE</td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td>YYYY</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>DDDD</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>PPPP</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>AA</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>VVEE</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>GGGG</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>PPPP</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>DTDD</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>NQNN</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>GNGG</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>VV</td>
<td></td>
</tr>
<tr>
<td>W</td>
<td>BKKK</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>PPPP</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>GGGG</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>TTTT</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>EKSA</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>PPPP</td>
<td></td>
</tr>
</tbody>
</table>

---

*Proc. Natl. Acad. Sci. USA 90 (1993) 7559*
For each residue, consider a window of size 13:

13x20 = 260 values
PHD: Network 1
Sequence ➔ Structure

13x20 values ➔ Network1 ➔ 3 values

\( P_{\alpha(i)} \) \( P_{\beta(i)} \) \( P_{c(i)} \)
For each residue, consider a window of size 17:

\[ \text{Network2} \]
One popular model for protein folding assumes a sequence of events:

- Hydrophobic collapse
- Local interactions stabilize secondary structures
- Secondary structures interact to form motifs
- Motifs aggregate to form tertiary structure
Protein Structure Prediction

A physics-based approach:

- find conformation of protein corresponding to a thermodynamics minimum (free energy minimum)
- cannot minimize internal energy alone! Needs to include solvent
- simulate folding…a very long process!
- Folding time are in the ms to second time range; however, Folding simulations at best run 1 ns in one day…
PHD: Secondary structure prediction using NN

• **Sequence-Structure network**: for each amino acid $a_j$, a window of 13 residues $a_{j-6} \ldots a_j \ldots a_{j+6}$ is considered. The corresponding rows of the sequence profile are fed into the neural network, and the output is 3 probabilities for $a_j$: $P(a_j,\text{alpha})$, $P(a_j,\text{beta})$ and $P(a_j,\text{other})$

• **Structure-Structure network**: For each $a_j$, PHD considers now a window of 17 residues; the probabilities $P(a_k,\text{alpha})$, $P(a_k,\text{beta})$ and $P(a_k,\text{other})$ for $k$ in $[j-8,j+8]$ are fed into the second layer neural network, which again produces probabilities that residue $a_j$ is in each of the 3 possible conformations

• **Jury system**: PHD has trained several neural networks with different training sets; all neural networks are applied to the test sequence, and results are averaged

• **Prediction**: For each position, the secondary structure with the highest average score is output as the prediction