Machine Learning in Structural Biology: Interpreting 3D Protein Images

By

ALOYSIOUS KIKOMEKO – 2019/HD07/24856U

CHISOM SOREMEKUN - 2020/HD07/24246U

Determining/ Interpreting of 3D Protein Images

- Given an electron density map, a three-dimensional "image" of a protein produced from crystallography to identify the chain of protein atoms contained within the image.

- Traditionally, a human performs this interpretation, perhaps aided by a graphics terminal.

- However, over the past 15 years, anumber of research groups have used machine learning to automate densitymap interpretation.

Determining/ Interpreting of 3D Protein Images ….

- Early methods of machine Learning had much success, saving thousands of crystallographer-hours, but required extremely high-quality density maps towork.

- Newer methods aim to automatically interpret poorer and poorer qualitymaps, using state-of-the-art machine learning and computer vision algorithms.

Machine learning in Structural Biology Alogarithimic Models

Algorithmic background

- Algorithms for automatically interpreting electron density maps draw heavily from the machine learning and statistics communities.

- These communities have developed powerful frameworks for modeling uncertainty, reasoning from prior examples and statistically modeling data, all of which have been used by researchers in crystallography.

Machine learning Alogarithimic Models

- Amodel here refers to a system that simulates a real-world event or process.

1. **Probabilistic models**

- Probabilistic models simulate uncertainty by producing different outcomes with different probabilities.
- In such models, the probabilities associated with certain events are generally not known, and instead have to be estimated from a training set, a set of previously solved problem instances.
- Using **maximum likelihood** estimation, the probability of a particular outcome is estimated as the frequency at which that outcome occurs in the training set.

1 . Probabilistic models …… (by type)

- The **unconditional or prior probability** of some outcome A is denoted P(A). This means that inthe absence of any other information, the best assignment of probability of outcome A

- The **conditional or posterior probability** is used when other, previously un-known, information becomes available (probability of A given B" == P(A|B)

- The **joint probability** of two or more events is the probability of both events occurring,

(probability of A and B" == P(A,B)

$$P(A,B) = P(A|B)P(B) = P(B|A)P(A)$$

• 2. Probabilistic models …… (Bayes' theorem)

- One computes the marginal probability by taking the joint probability and summing out one or more variables P(A,B,C),

- Marginal distribution of A is computed:

$$P(A) = \Sigma_B \Sigma_C P(A,B,C)$$

With-out requiring one to explicitly compute the (possibly intractable) full joint distribution.

- Finally, **Bayes'** rule allows one to reverse the direction of a conditional:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

• 3. Probabilistic models …… (Bayes' theorem)

- **Bayes'** rule is useful for computing a conditional P(A|B) when direct estimation (using frequencies from a training set) is difficult, but when P(B|A) can be estimated accurately.

- Often, one drops the denominator, and instead computes the relative likelihood of two outcomes, for example, P(A=a1|B) versus P(A=a2|B). If a1 and a2 are the only possible outcomes for A, then exact probabilities can be determined by normalization; there is no need to compute the prior P(B).

Case-based reasoning (CBR)

- case-based reasoning(CBR) attempts to solve a new problemby using solutions to similar past problems.

- Algorithms for case-based reasoning require a database of previously solved problem instances, and some distance function to calculate how “different” two problem instances are.

Case-based reasoning (CBR) …..

- There are two key aspects of CBR systems.
- **First,** learning in such systems is lazy: the models only generalize to unseen instances when presented with such a new instance.
- **Second,** they only use instances "close" to the unseen instance when categorizing it.

Case-based reasoning (CBR) …..

- The most common CBR algorithm is **k-nearest neighbor (kNN).**
- In kNN, problem instances are feature vectors, that is, points in some n-dimensional space.
- With a new problem instance X**=** (x_i,…,x_n) for classification or regression, finds the k previously solved problem instances closest to the query in Euclidean space.

$$d(X||Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}.$$

Case-based reasoning (CBR) …..

- Then, the **k neighbors "vote" label:** usually the majority class label **(for classification)** or average label **(for regression).**

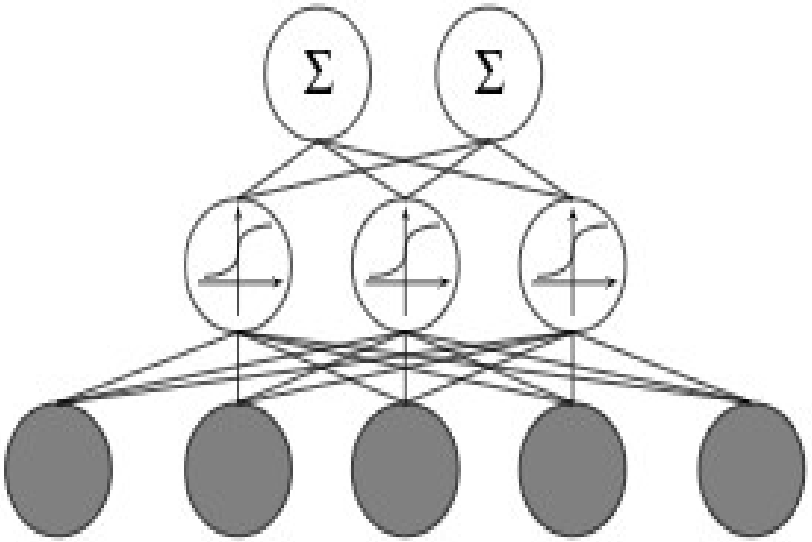
- One variant of kNN weights each neighbor's vote by its similarity to the query. Another variant learns weights for each dimension, to be used when computing the distance between two instances.

Neural networks

- An Artificial Neural Network (ANN) is a non linear function estimator used to approximate real or discrete target functions.
- Inspired by neurons in the brain, ANNs consist of a number of units connected in a network.
- Each unit is connected with multiple inputs and a single output.

Neural networks …..

- A given unit's output is some function of the weighted sum of the inputs.
- This function is known as the unit's activation function.
- For aperceptron– a simple, one-layer network– this function is usually a step function.



A multilayer, feed-forward neural network. The network consists of an input layer fully connected to a hidden layer of sigmoid units, fully connected to an output layer

Neural networks …..

- This network consists of a hidden layer which fully connects the input and outputs.

- Learning the weights in the network requires a differentiable activation function; often one uses a sigmoidal function:

$$\sigma(y) = \frac{1}{1 + e^{-y}}.$$

- Above, y is the weighted sum of inputs, $y = \sum_{i=0}^{N} w_i x_i.$

Neural networks …. (back propagation algorithm)

- The back propagation algorithm learns the weights for a multilayer network, given a network structure.

- Back propagation uses gradient descent over the network weight space to minimize the squared error between computed out-put values and desired output values over some training set.

- The goal of back propagation is to find some point in weight space – that is, some setting of all the weights in the network that (locally) minimizes this squared error.

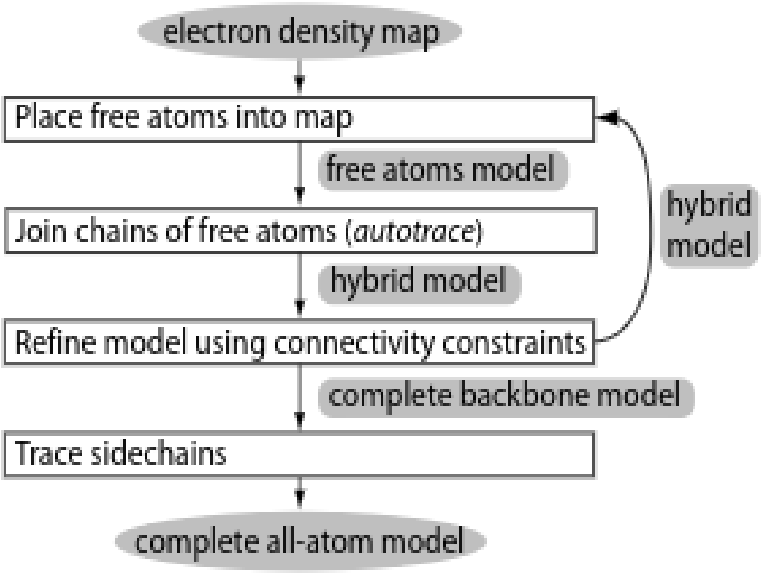
Approaches to Automatic Density Map Interpretation

ARP/WARP

- The ARP/WARP **(Automated Refinement Procedure)** software suite is a crystal-lographic tool for the interpretation and refinement of electron density maps.
- ARP/WARP's warpntrace procedure was the first automatic interpretationtool successfully used for protein models.
- Today, it remains one of the most used tools in the crystallographic community for 3D protein-image interpre-tation.

ARP/WARP … (Free-atom placement)

- ARP/WARP contains a atom placement method based on arp, an interpretation method for general molecular models.
- ARP randomly places unconnected atoms into the density map, producing a free atom model.
- To initialize the model,arpbegins with a small set of atoms in the density map.



Aflow chart of warpntrace

ARP/WARP … (Main-chain tracing)

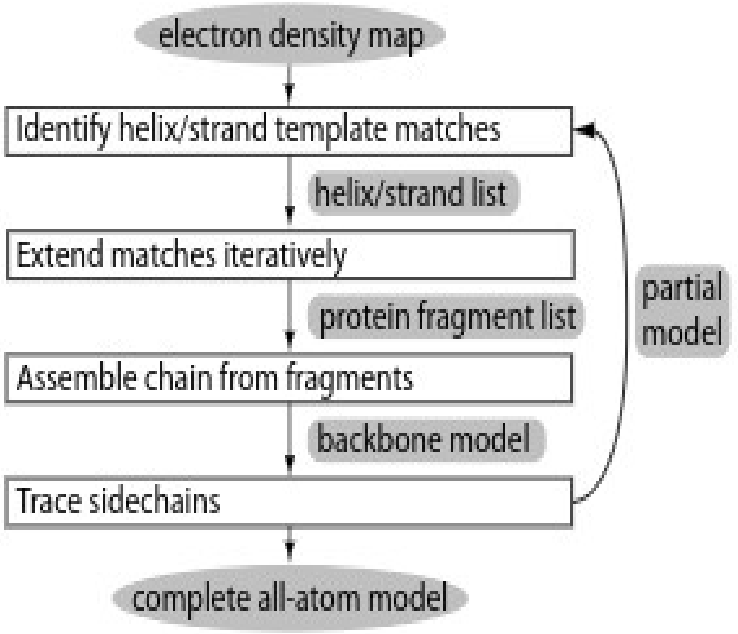
- Given a free-atom model of a protein, one can form a crude backbone trace by looking for pairs of free atoms the proper distance apart.
- Warpntrace formalizes this procedure, called autotracing, using a heuristic method.
- After computing a list of C_α pairs,warpntrace constructs the backbone using a database of known backbone conformations (taken from solved protein structures).
- Given a chain of candidate C_α pairs, warpntrace considers a backbone conformations in the database with matching C_α positions, ordered by length.

Resolve

- Resolve's method hinges upon the construction of two model secondary structure fragments a short α-helix and β-strand for the initial matching.

- Resolve first searches over all of rotation and translation space for these fragments; after placing a small set of over-lapping model fragments into the map, the algorithm considers a much larger template set as potential extensions.

- Resolve joins overlapping fragments and, finally, identifies side chains corresponding to each Cα conditioned on the input sequence and places individual atoms into the model.

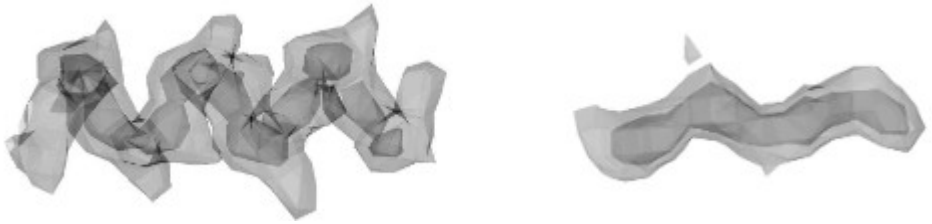


Aflow chart of Resolve.

Resolve _ _secondary structure search

- Given an electron density map, resolve begins its interpretation by searching all translations and rotations in the map for a model 6-residue α-helix and a model 4-residue β-strand.

- Resolve constructs these fragments by aligning a collection of helices (or strands) from solved structures; it computes the electron density for each at 3A resolution, and averages the density across all examples.



The averaged helix (left) and strand (right) fragment used in resolve's initial matching step

Resolve _ _secondary structure search ……

- Resolve considers placing each model fragments, at position in the map. At each position it considers all possible rotations (at a 30° or 40° discretization) of the fragment, and computes a standardized squared-densitydifference between the fragment's electron density and the map

  - is the map in which we ar $t(\vec{x}) = \sum_{\vec{y}} \epsilon_f(\vec{y}) \left(\rho'_f(\vec{y}) - \frac{1}{\sigma_f(\vec{x})} [\rho(\vec{y} - \vec{x}) - \bar{\rho}(\vec{x})] \right)$
  - $\rho(\vec{x})$ is thestandardizedfragment electron density,
  - $\rho'_f(\vec{x})$ is amasking function that is nonzero only for points near the fragment
  - $\epsilon_f(\vec{x})$ standardize the map in themasked region ϙf centered at~x

$\bar{\rho}(\vec{x})$ and $\sigma_f(\vec{x})$

$$\bar{\rho}(\vec{x}) = \frac{\sum_{\vec{y}} \epsilon_f(\vec{y}) \rho(\vec{y} - \vec{x})}{\sum_{\vec{y}} \epsilon_f(\vec{y})}$$

$$\sigma_f^2(\vec{x}) = \frac{\sum_{\vec{y}} \epsilon_f(\vec{y}) [\rho(\vec{y} - \vec{x}) - \bar{\rho}(\vec{x})]^2}{\sum_{\vec{y}} \epsilon_f(\vec{y})}.$$

Resolve _ _secondary structure search ……

- RESOLVE refines each fragment's position and rotation to maximize the real-space correlation coefficient (RSCC) between template and map:

$$RSCC(\rho_f, \rho) = \frac{\langle \rho_f \cdot \rho \rangle - \langle \rho_f \rangle \langle \rho \rangle}{\sqrt{\langle \rho_f^2 \rangle - \langle \rho_f \rangle^2} \sqrt{\langle \rho^2 \rangle - \langle \rho \rangle^2}}.$$

Here, ⟨ρ⟩ indicates the map mean over a fragment mask.Resolveonly con-siders refined matches with an RSCC above some threshold.

Resolve _ _Terative Fragment Extension

- At this point, resolvehas a set of putative helix and strand locations inthe density map.

Specifically, resolve makes use of four such libraries for fragment extension:

- (a) 17α-helices between 6 and 24 amino acids in length
- (b) 17 β-strands between 4 and 9 amino acids in length
- (c) 9,232 tripeptides containing backbone atoms onlyfor N-terminus extension
- (d) 4,869 tripeptides containing a partial backbone (the chain $C_\alpha - C - O$ with no terminal N) plus two full residues for C-terminus extension
- Resolve learns these fragment libraries from a set of previously solved protein structures. It constructs the two tripeptide libraries by clustering a larger dataset of tripeptides

- **Resolve _ _ Chain assembly**

- Given this set of candidate model segments, resolve’s next step is assembling a continuous chain.
- To do so, it uses an iterative method,
- The outermost loop repeats until no more candidate segments

• Resolve _ _ Sidechain trace

- Resolve's final step is, given a set of C_α positions in some density map, to identify the corresponding residue type, and to trace all the sidechain atoms.

- This sidechain tracing is the first time that resolve makes use of the analyzed protein's sequence. Resolve's sidechain tracing uses a probabilistic method, finding the most likely layout conditioned on the input sequence.

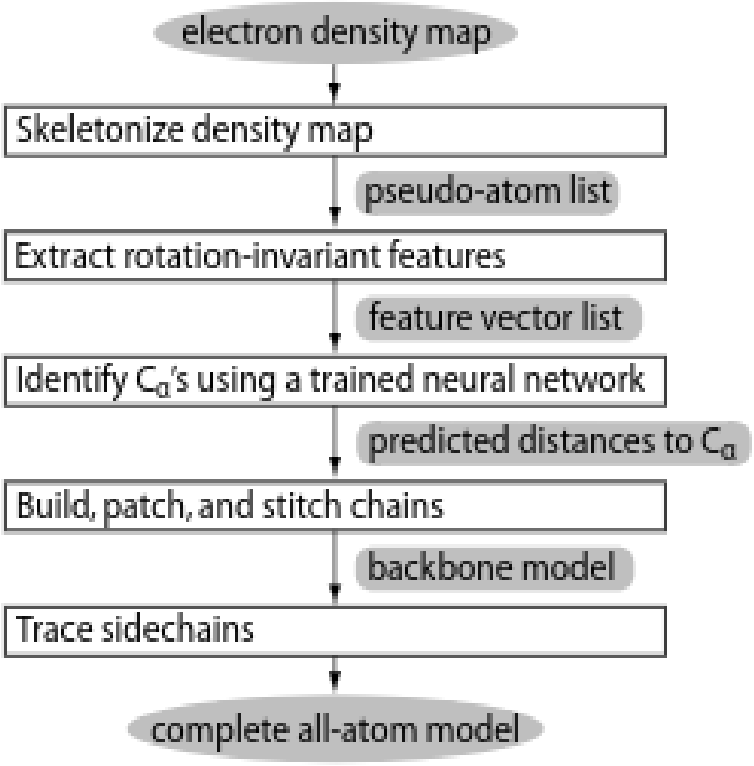
- For each rotamerj, the correlation coefficient at thekth C_α is given by cc_{jk}. A Z-score is computed, based on rotamer-j's correlation at every other $C_{\alpha.}$

$$Z_{jk}^{rot} = \frac{cc_{jk} - \langle cc_j \rangle}{\sigma_j}$$

The algorithm only keeps as ingle best-matching rotamer of each residue type

Textal

- Textal– another method for density map interpretation
- Textal uses of a set of rotationally invariant numerical features to describe regions of density.
- The electron density around each point in the map is converted to a vector of 19 features sampled at several radii that remain constant under rotations of the sphere.



A flowchart oftextal

Textal _ _ Feature extraction

- The most important component of Textal is its extraction of a set of numerical features from a region of density.These numerical features allow rapid identification of similar regions from different (solved) maps.

- A key aspect of Textal's feature set is invariance to arbitrary rotations of the region's density

- A third class of descriptors includes moments of inertia (MOI), which provides six features describing how elliptical is the density distribution. Moments of inertia are calculated as the Eigenvectors of the inertia matrix I $$I = \sum_i \left(\rho_i \begin{vmatrix} y_i^2 + z_i^2 & -x_i y_i & -x_i z_i \\ -x_i y_i & x_i^2 + z_i^2 & -y_i z_i \\ -x_i z_i & -y_i z_i & x_i^2 + y_i^2 \end{vmatrix} \right)$$

- Above, ρ_i is the density at point (x_i,y_i,z_i). As a rotation-invariant description, Textal only considers the moments and the ratios between moments, not theaxes themselves (the Eigenvectors of the inertia matrix).

Textal _ _Backbone tracing

- A feed-forward neural network – a non linear function approximator used for both classification and regression – is trained to learn which pseudo-atoms correspond to actual Cα's.
- Specifically, the network is trained on a set of previously solved maps to predict the distance of each pseudo-atom to the nearest C_α.
- The rotation invariant features are inputs to the network; a single output node estimates the distance to the closest C_α.
- A hidden layer of 20 sigmoidal units fully connects input and output layers

Textal _ _Sidechain placement

- After capra returns its predicted backbone trace, textal must next identify the residue type associated with each C_α.
- Essentially, the subroutine compares the density around each C_α to a data base of solved maps to identify the residue type.
- Lookup uses Textal's rotation invariant features, and builds a database of feature vectors corresponding to C_α's in solved maps.

Textal _ _Sidechain placement …..

- To determine the residue type of an unknown region of density, lookup finds the nearest neighbors in the database, using weighted Euclidian distance

$$D(\rho_1||\rho_2) = \left[\sum_i \lambda_i \cdot \left(F_i(\rho_1) - F_i(\rho_2)\right)^2\right]^{1/2}.$$

- F_i refers to the ith feature in the vector, while ρ1 and ρ2 are two regions of density.

- Feature weights λi are optimized to maximize similarity between matching regions and minimize between nonmatching regions, where ground truth is the optimally-aligned RSCC.

- Textal sets weights using the slider algorithm which considers features pairwise to determine a good set of weights

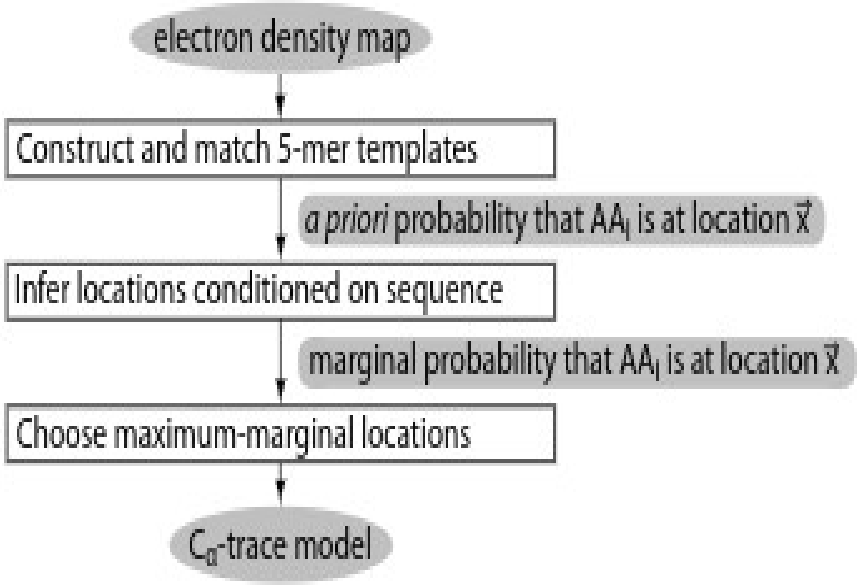
Textal _ _Post-processing routines

- Textal’s post-processing makes sure that all chains are oriented in a consistent direction.
- Refinement, like that of ARP/WARP, corrects improper bond lengths and bond angles, iteratively moving individual atoms to fit the density map better.
- Finally, textal takes into account the target protein’s sequence to fix mismatched residues.

ACMI

- ACMI (Automatic Crystallographic Map Interpreter) is a recent method for tracing protein backbones in poor-quality density maps.

- Acmi takes a probabilistic approach to electron density map interpretation, finding the most likely layout of the backbone under some likelihood function.



Aflow chart of ACMI

• ACMI _ _ Local matching

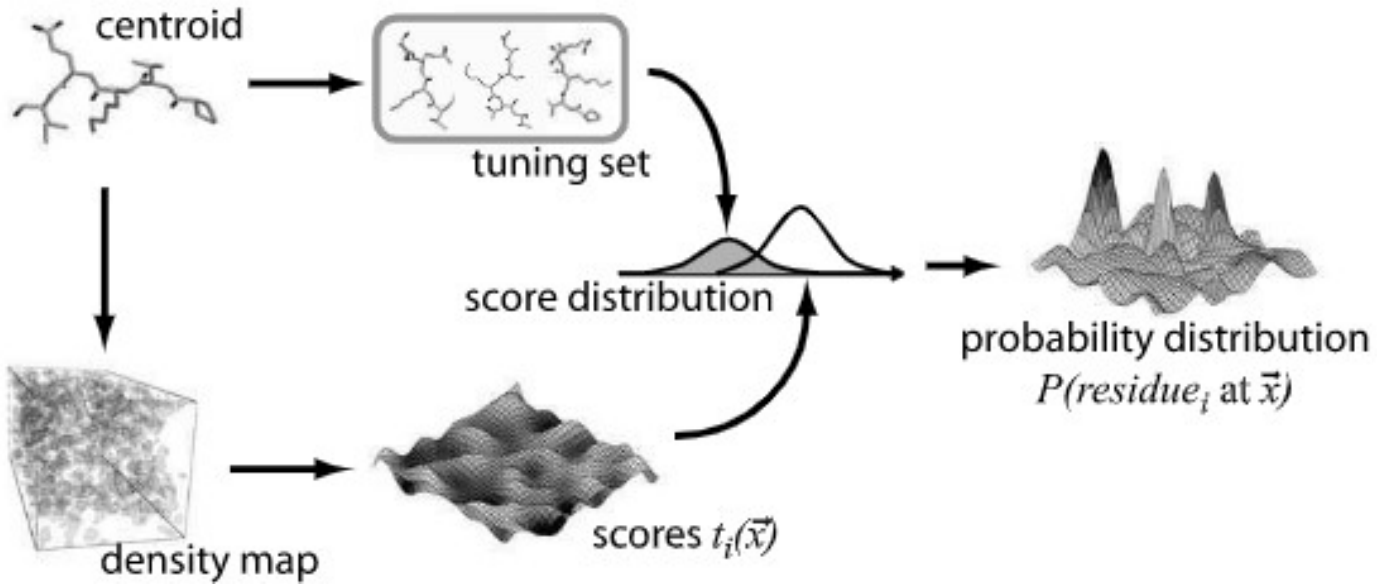
- The local matching component of ACMI is provided the density map and the protein's amino acid sequence.

- It computes, for each residue in the protein, a probability distribution $P_i(\vec{w}_i)$ over all locations $\vec{w}_i$ in the unit cell.

- This probability distribution reflects the probability that residue i is at position and orientation $\vec{w}_i$ in the unit cell

- Using a set ofpreviously solved structures,acmifirst constructs a basis set of structures de-scribing the conformational space of each 5-mer in the protein

ACMI _ _ Constructing a sequence-specific 5-mer basis set



An overview of the 5-mer template matching process. Given a clusterof 5-mers for residue i, ACMI performs a 6D search for the fragment in the density map. ACMI also matches the fragment to a tuning set of known structures, using Bayes' rule to determine a probability distribution from the match scorest $t(\vec{w})$

Constructing a sequence-specific 5-mer basis set

- To compute probabilities, Bayes' rule is used:

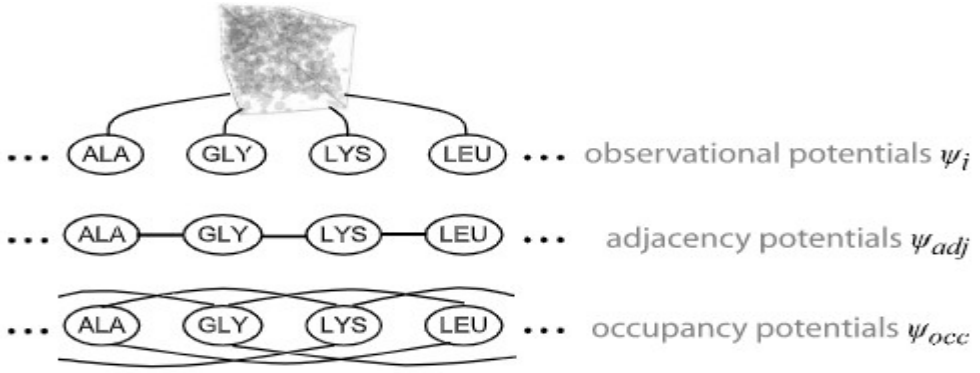
$$P(\text{res. } i \text{ at } \vec{x}_i | t(\vec{x}_i)) = \frac{P(t(\vec{x}_i) | \text{res. } i \text{ is at } \vec{x}_i) \cdot P(\text{res. } i \text{ at } \vec{x}_i)}{P(t(\vec{x}_i))}.$$

ACMI _ _ Global constraints

- Given each residue's independent probability distribution over the unit cell, $P_i(\vec{x}_i)$, ACMI accounts for global constraints (enforced by physical feasibility) of a formation by modeling the protein using apairwise Markov field.

- Apairwise Markov field defines the joint probability distribution over a set of variables as a product of potential functions defined over vertices and edgesin an undirected graph

ACMI _ _ Markov field model ...



ACMI protein backbone model.

(a) The joint probability of a conformationof residues is the product of an observation potentialpsiiat each node,

(b) an ad-jacency potential between adjacent residues

(c) an occupancy potential between all pairs of nonadjacent residues

• ACMI _ _Adjacency potentials

- Adjacency potentials, which connect every neighboring pair of residues, are the product of two constraining functions, a distance constraint function and a rotational constraint function

$$\psi_{adj}(\vec{w}_i, \vec{w}_j) = p_x(||\vec{x}_i - \vec{x}_j||) \cdot p_\theta(\vec{w}_i, \vec{w}_j).$$

- In proteins, the Cα- Cα distance is a nearly invariant 3.8A. Thus, the potential p_x takes the form of a tight Gaussian (σ= 0.03A) around this ideal value, softened a bit $\hat{\vec{q}_i}$r grid effects. Acmi defines the potential pθ using an alternate parameterization of the angular parameters .

ACMI _ _ Occupancy potentials

- Occupancy potentials ensure that two residues do not occupy the same loca-tion in space.

- They are defined independently of orientation, and are merelya step function that constrains two (nonadjacent) Cα's be at least 3A apart.

- It is in this structural potential function thatacmideals with crystallo-graphic symmetry, by ensuring that all crystallographic copies are at least 3A apart

$$\psi_{occ}(\vec{w}_i, \vec{w}_j) = \begin{cases} 1 & \left(\min_{\substack{symmetric \\ transforms\ K}} ||x_i - K(x_j)|| \right) \geq 3.0\mathbf{A} \\ 0 & \text{otherwise.} \end{cases}$$

ACMI _ _ Tracing the backbone

- Since the graph over which the joint probability is defined contains loops, finding an exact solution is infeasible (dynamic programming can solve this in quadratic time for graphs with no loops).

$$\begin{aligned}\vec{w}_i^{*} &= \arg\max_{\vec{w}_i} \hat{b}_i(\vec{w}_i) \\ &= \arg\max_{\vec{w}_i} \psi_i(\vec{w}_i, \mathbf{M}) \times \prod_{j \in \Gamma(i)} m_{j \to i}^{n}(\vec{w}_i).\end{aligned}$$

- Since ACMI computes a complete probability distribution, it can return not only a putative trace, but also a likelihood associated with each amino acid. This likelihood provides the crystallographer with information about what areas in the trace are likely flawed; ACMI can also produce a high-likelihood partial trace suitable for phase improvement.

References

- [1] B. Rost and C. Sander (1993). Prediction of protein secondary structure at betterthan 70% accuracy.J Mol Biol.
- [2] G. Rhodes (2000).Crystallography Made Crystal Clear. Academic Press.
- [3] J. Abrahams and R. De Graaff (1998). New developments in phase refinement.Curr Opin Struct Biol.
- [4] S. Russell and P. Norvig (1995).Artificial Intelligence: A Modern Approach.Prentice Hall.
- [5] T. Mitchell (1997).Machine Learning. McGraw-Hill.
- [6] V. Lamzin and K. Wilson (1993). Automated refinement of protein models.ActaCryst

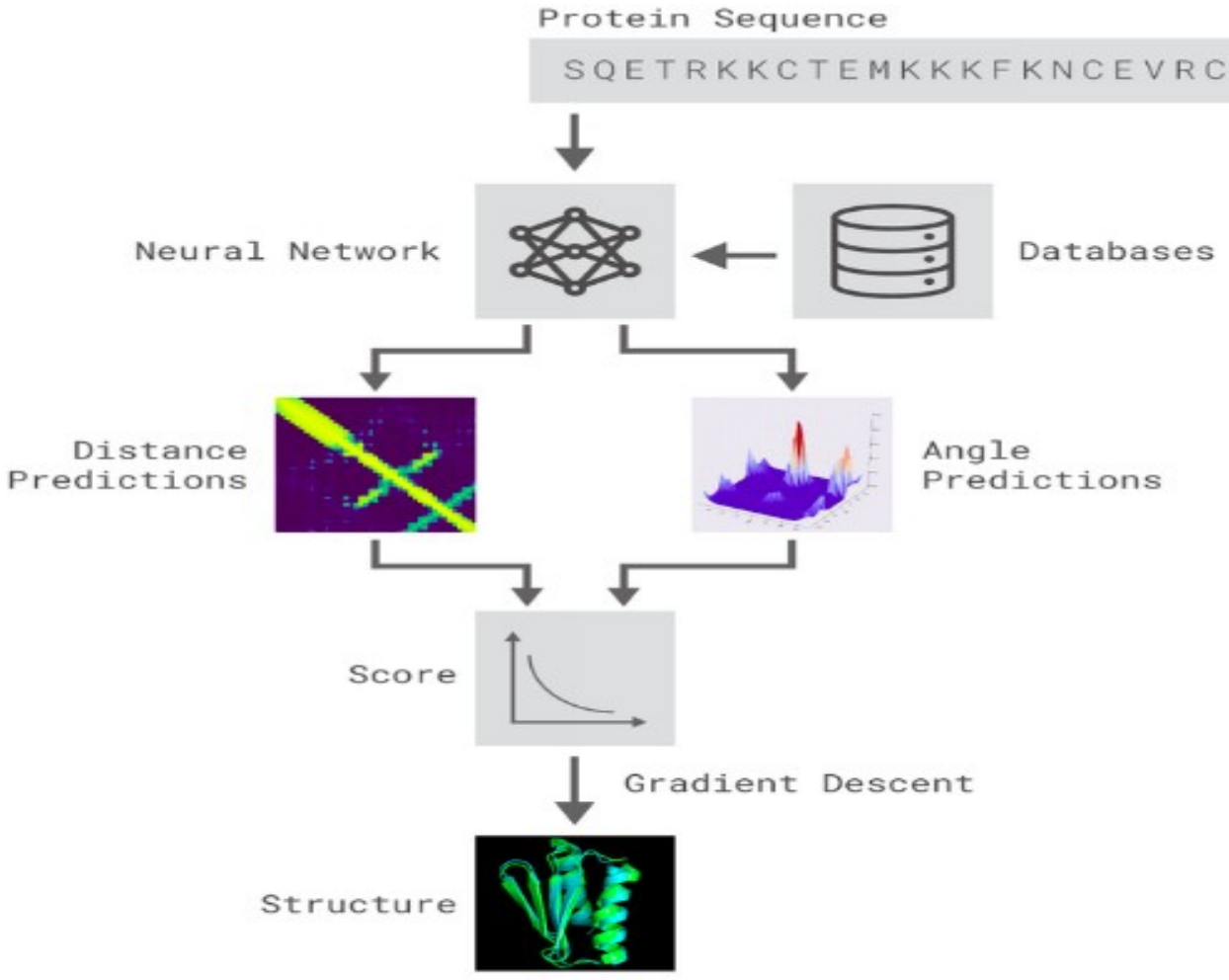


Image from DeepMind's original blogpost.