Laboratory of Bioinformatics 1 part B - Capriotti

Saul Pierotti

March 27, 2020

Introduction

- There will be a project for the final, to be submitted for may 18
- Protein structure is more conserved than sequence
- When sequence identity is sufficiently high, we can tranfer structural information
- A structural alignment is a rigid body transformation of 2 subsets from 2 sets of points that maximizes a given distance metric
  - The subsets need to have the same number of elements and define the corrispondence set
  - Finding the corrispondece set is an NP-hard problem
  - Finding the optimal rigid transformation of the corrispondence set is $\Theta(n)$
- The distance of 2 sequences can be evaluated with a substitution matrix and a gap penalty
- Global alignments are computed with the NW algorithm, local alignments with the SW
- The significance of an alignment score can be evaluated by comparing with the score distribution for random alignments
- Over 100 residues, under 25% of sequence identity only 10% of the sequences are homologous, while above 20% 90% of them are
- Over 30% idnetity sequences longer than 100 residues have similar structures, but this does NOT mean that under 30% the structure is necessarily different!
- Proteins with low sequence identity but high structural similarity are referred to as remote homologs
- Structures can be predicted by comparative modeling, threading, ab initio
- If I want to use sequence identity for trasferring annotation features, I need to identify the problem-specific twilight region
- The sequence identity needed for transferring subcellular localization is higher than that required for structure
- Function of proteins with really high sequence identity can be completely different
- In remote homologs the sequence alignment is often wrong
- Important residues in a sequence can be identified by comparing conservation levels

Structural alignment

- Structural alignment is different from superimposition
- Superimposition assumes that I already have the correspondence set, and it is relatively easy
- Structural alignment requires the identification of the correspondence set, which is hard
- The definition of domain is often heuristic and questionable
- Proteins with similar spatial distribution but different topology are difficult to align
- Alignment methods can be classified in different ways
  - Pairwise or multiple
  - Depending on the descriptor used
    - Backbone
    - All atoms
    - Sequence-based

        - * Contact map
        - * Surface
    - Rigid body or flexible
- The comparison of torsion angles is $\Theta(n)$
    - They are invariant for rotation and translation
    - It is good for local regions but problematic for whole structures
- A distance matrix is also invariant for rotaion and translation
    - Comparing matrices is hard, $\Theta(n^2)$
    - It is not sensitive to chirality
- At the moment, all methods are able to identify obvious similarities
- Remote similarities are detected by a subset of methods, and different methods recognize different similarities
- Speed is an issue in many algorithms
- We want our method to be biologically meaningful, not only geometrically
- The expected score or random pairwise alignments is an extreme value distribution
    - I would have a gaussian if there was no evolution
    - In real databases I have an excess of good-scoring pairs
- When I want to determine the distribution of scores, it is better to have an analitycal distribution than an empirical one
    - I don't have tools for working with empirical distributions (!)

CE algorthm

- Compares AFPs composed of 8 residues, stiches them together and finds an optimal path trough them with dynamic programming
- It gives a statistical score
- The alignment is the longest continuous path of AFPs in a similarity matrix S
- The similarity matrix S is composed represent all AFPs conforming to a similaritt criterion
- The dimensions of S are (na-m)(nb-m), where na and nb are the length of the sequences and m the size of the AFPs
- The matrix is large to compute, therefore we need constraints
- Two consecutive AFPs can be aligned with a gap in protein A, a gap in protein B or without gaps
- The AFP lenght is set to 6 and the maximum possible gap to 30
- Similarity measures are RMSD, full set of distances, and others
- The best 20 alignments with Z score above 3.5 are compared based on RMSD and the best one is kept
    - I get an error in 1000 comparisons
- Each gap is assessed for relocation up to m/2 times
- Iteritive optimization with dynamic programming
- It cannot find non-topological alignments
- The unit of comparison was originally the protein chain, but domains are optimal
    - Domains are difficult to define (!)
- The statistical distribution of alignment scores can be used to evaluate the Z score of an alignment

PDBe Fold

- It uses secondarys structure elements (SSEs)
- Secondary structure is typically conserved
- SSE are represented as vectors that connected in a graph by edges
    - 2 vertices and an edge describe position and orientation of the SSEs
    - SSEs are helices and strands
- Each edge is labelled by a property vector containing information on edge-vertices angles, torsion angles between vertices, lenght of the edge
- The set of vertices, edges and labels defines the graph that is then matched with an algorthm
- Vertex and edge lenghts are compare both in absolute and relative terms

  - In relative terms, the same absolute difference is less significative for longer edges
- Torsion angles are used for distinguishing mirror simmetries
- The SSE matching gives correspondences among SSEs, and can be used to yeld an initial sequence alignment
- Connectivity (topology) can be neglected, considered but allow for any number of missing SSEs (soft connectivity) or allow only for an equal number of unmatched SSEs (strict connectivity)

MAMMOTH algorithm

- Matching molecular models obtained from theory (MAMMOTH) is one of the fastest algorithms
- The protein is represented as a set of unit vectors among Ca
- It is based on dynamic programming
- An unit vector is the normalized vector among Ca atoms
  - For each position, k consecutive vectors are mapped into a unit sphere that represents the local structure of k residues
- Each set of unit vectors is compare to all the sets in the other structure, building a matrix
- Each comparison yelds a unit root mean square distance (URMS)
  - This is compared against the expected random URMS
  - THe alignment score is obtained by normalizing the URMS with its expected value
- The path trough the matrix is found with dynamic programming by a global alignment without end-gap penalties

RNA structure

- Most RNAs are around 50 bp
- Secondary structure of RNAs is usually represented with parenteses
  - I cannot represent pseudo-knots in this way
- For RNA, the secondary strucutre is much more informative than for proteins
  - A certain secondary structure constraints a lot the tertiary structure
- There is less variability in RNA strucutures than in proteins
- The best atom for representing the backbone is C3', since it has the most constant inter-nuclotide distance
- The professor adapted MAMMOTH to work with RNA C3' atoms instead of Ca in proteins: SARA
  - The statistics of the score had to be re-evaluated
  - They still used the extreme value distribution, which is defined by μ and σ
  - They selected how the parameters change when RNA size changes
  - The set of unit vectors was 3 instead of 7
  - The method gives a -log(p-value) score
  - By comparing RNAs of known function, I can determine a score threshold that gives correct functional annotation
- Another method was developped in Israel: ARTS
- Few people are working in RNA: not so many methods
- The twilight zone of RNA sequence alignment is around 60%
- Secondary structure identity (PSS) correlates well with tertiary structure identity (PSI) but not with sequence identity

Multiple sequence alignment

- We can observe blocks of conservation in MSAs
- In MSA it is easier that in pariwise alignments to identify conserved regions
- Conserved regions could be functionally important
- I can transform a MSA in a profile of the sequences
- A profile is a matrix with a row for each possible residue and a column for each position

  - The value of each element reflects the frequency of a residue in a specific position
  - Each position is therefore a vector of 20 elements
  - I can also have a row for the presence of a gap in the position
- Shannon entropy: information content of a message
  - Far a single colum $S(p) = \sum_{i=1}^{20} -p_i \ln p_i$
  - Total conservation: $S(p) = 0$
  - All residues are equally probable: $S(p) = ln(20)$
  - There are more sofisticated models that take into account the expected frequency of residues
  - The entropy of an alignment si obtained by summing the Shannon entropy over the all alignment
- Scoring an MSA: sum of pairwise scores or entropy score
  - I can obtain the MSA so to minimise its entropy
  - I can score each pairwise alignment and sum it
- I can align a sequence to a profile
  - Each position is aligned to a vector for the position
  - The score for the position of the residue in the sequence with every possible residue is summed and weighted for the frequency encoded in the vector
    - This is a matrix by vector multiplication (!)
  - These scores can be used with a dynamic programming algortihm
- To calculate an MSA, I want to optimise its score
- Dynamic programming approaches exist, but they are $O(N^M)$ and they are np-hard
- A possible solution is to do a progressive MSA, like with Clustal
  - I allign sequences in pairs, one after the other
  - The result depends on the order of how I pair sequences (!)
    - I usually pair the most similar sequences first
  - From each pairwise alignment, I build a profile
  - I iterate until there are no sequences left, by aligning pairwise sequences and profiles
  - In order to do this I need to be able to align profiles (!)
  - I want to be conservative with gaps with the initial pairwise alignments, and introduce them later on profiles
    - When I get to profiles I have info about conservation (!)
    - Errors in the first alignments are propagated
    - If I am not conservative I can become full of gaps
  - I can improve the alignment by changing the sequence tree
    - By default Clustal uses NJ
    - Maybe I have a tree available (!)
- A profile-to-profile alignments involve the pairwise comparison of same-dimentional vectors
  - I do a double sum all against all elements weighted with a substitution matrix
  - This is done via a simple vector to matrix multiplication, followed by a multiplicatio for the remaining vector (!)
  - Adding gaps is tricky, since their penalty logically depends on the position and conservation
- An MSA method can be evaluated from the functionally important residues that are correctly aligned
- MUSCLE is an iterative MSA method
  - It is based on kmers
    - If a rare kmer is present in 2 sequences maybe they are related
  - It creates a distance matrix with all sequences against each other
  - From the matrix, the tree is calculated with UPGMA
  - It creates the alignment progressively
  - From the alignment evaluates the pairwise distances using the Kimura distance
  - It creates a new distance matrix and a new tree, and from this a new MSA
  - It splits the tree and aligns the profiles
  - It iterates this last step
- An MSA should be consistent: if residue X is aligned with Y and Y is aligned with Z, then X is aligned to Z
  - I can use this property backwards by assigning an higher score to MSA that respect this property

- The consistency refers to the respective pairwise alignments
  - Progressive MSA methods frequently are not consistent
- T-coffe is an MSA method based on consistency
  - I do all the possible pairwise and I measure sequence identity
    - This is the primary library
  - Every pairwise has a weight equal to its identity

MSA benchmark

- BaliBASE is used for MSA benchmark
  - It is a dataset with manually cureted alignments deriving from structural superimposition
  - SP score: 1 if a pair of residues is aligned corrextly, 0 if not and I sum all the scores
- No single method is perfect in all cases (!)
  - On average, consistency-based methods are better but slower
  - Many algos take advantage of parallel processing

Probabilistic models of protein families

- I can define the probability of a sequence s_i to be generated by a family described by a MSA (and hence a profile as a matrix) M
  - $p(s_i|M)$