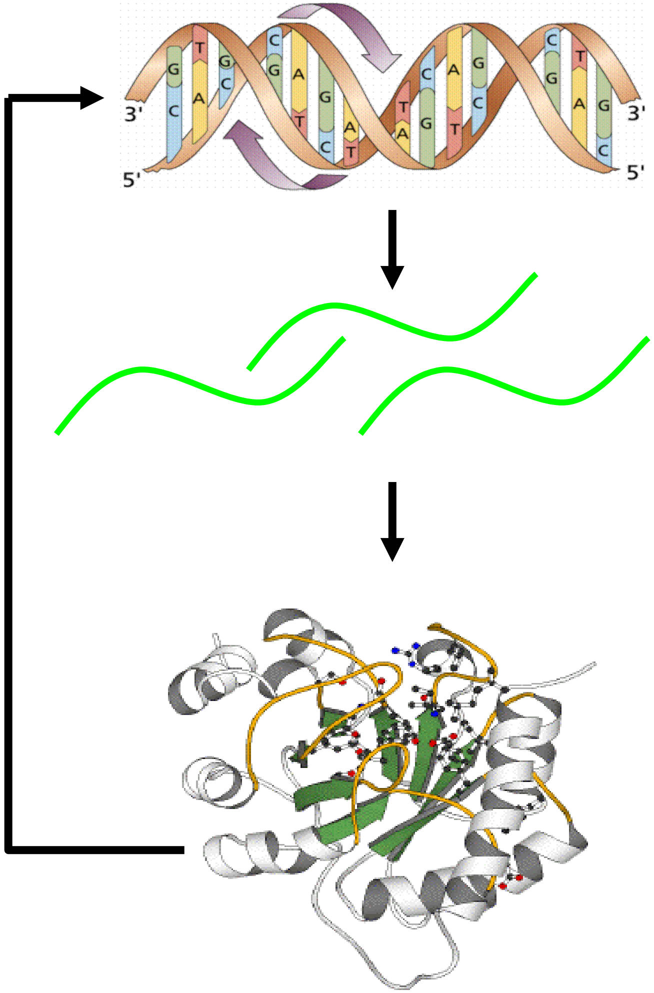
15-780: Graduate Artificial Intelligence

Computational biology: Sequence alignment and profile HMMs

Central dogma



DNA

transcription

mRNA

translation

Protein

CCTGAGCCAACTATTGATGAA

CCUGAGCCAACUAUUGAUGAA

PEPTIDE

Assigning function to proteins

- One of the main goals of molecular (and computational) biology.
- There are 25000 human genes and the vast majority of their functions is still unknown
- Several ways to determine function

  - Direct experiments (knockout, overexpression)
  - Interacting partners
  - 3D structures

  Hard

  - Sequence homology

  Easier

Comparison of Different Organisms

	Genome size	Num. of genes
E. coli	$.05*10^8$	4,200
Yeast	$.15*10^8$	6,000
Worm	$1*10^8$	18,400
Fly	$1.8*10^8$	13,600
Human	$30*10^8$	30,000
Plant	$1.3*10^8$	25,000

Function from sequence homology

- We have a query gene: ACTGGTGTACCGAT
- Given a database with genes with a known function, our goal is to find another gene with similar sequence (possibly in another organism)
- When we find such gene we predict the function of the query gene to be similar to the resulting database gene
- Problems

  - How do we determine similarity?

Sequence analysis techniques

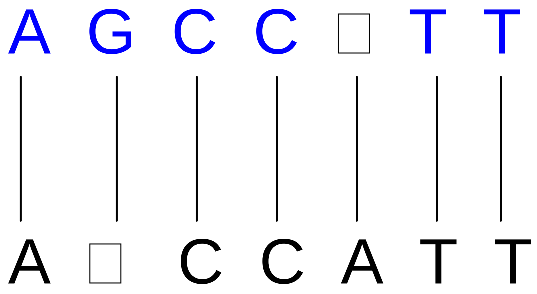
- A major area of research within computational biology.
- Initially, based on deterministic or heuristic alignment methods
- More recently, based on probabilistic inference methods

Sequence analysis

- Traditional
  - Dynamic programming
  - Blast
- Probabilsitic
  - Profile HMMs

Pairwise sequence alignment

AGCCTT
ACCATT



A G C C □ T T

A □ C C A T T

- We cannot expect the alignments to be perfect.
- Major reasons include insertion, deletion and substitutions.
- We need to allow gaps in the resulting alignment.

Scoring Alignments

- Alignments can be scored by comparing the resulting alignment to a background (random) model.

Independent

$$P(x,y\,|\,I) = \prod_i q_{x_i} \prod_j q_{x_j}$$

Related

$$P(x,y\,|\,M) = \prod_i p_{x_i y_i}$$

Score for alignment:

$$S = \sum_i s(x_i, y_i)$$

where:

$$s(a,b) = \log\left(\frac{p_{a,b}}{q_a q_b}\right)$$

Scoring Alignments

- Alignments can be scored by comparing the resulting alignment to a background (random) model.

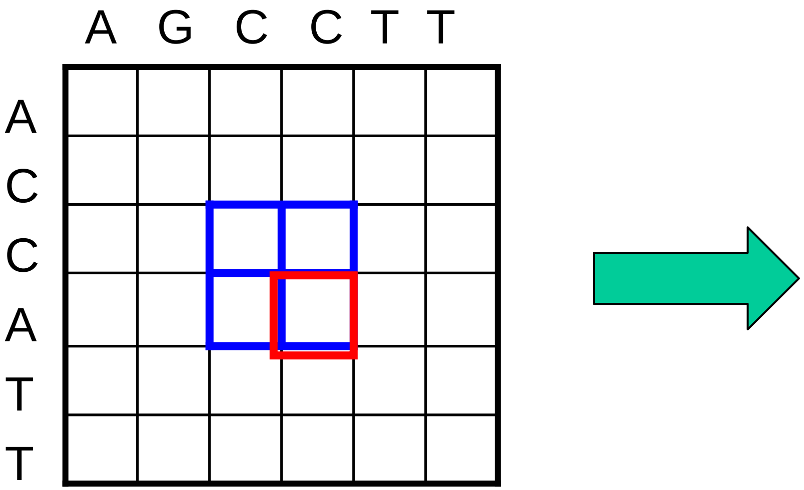
In other words, we are trying to find an alignment that maximizes the likelihood ratio of the aligned pair compared to the background model

Score for alignment: $S = \sum_i s(x_i, y_i)$ where: $s(a,b) = \log(\frac{p_{a,b}}{q_a q_b})$

Computing optimal alignment: The Needham-Wuncsh algorithm

$$F(i,j) = \max \begin{cases} F(i-1,j-1)+s(x_i,x_j) \\ F(i-1,j)-d \\ F(i,j-1)-d \end{cases}$$

d is a penalty for a gap



F(i-1,j-1)	*F(i-1,j)*
F(i,j-1)	*F(i,j)*

Example

Assume a simple model where *S(a,b) = 1* if *a=b* and -5 otherwise.

Also, assume that d = -1

		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1						
C	-2						
C	-3						
A	-4						
T	-5						
T	-6						

Example

Assume a simple model where *S(a,b) = 1* if *a=b* and -5 otherwise.

Also, assume that d = -1

$$F(i,j) = \max \begin{cases} F(i-1,j-1)+s(x_i,x_j) \\ F(i-1,j)-d \\ F(i,j-1)-d \end{cases}$$

		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1	1					
C	-2						
C	-3						
A	-4						
T	-5						
T	-6						

Example

Assume a simple model where *S(a,b) = 1* if *a=b* and -5 otherwise.

Also, assume that d = -1

$$F(i,j) = \max \begin{cases} F(i-1,j-1)+s(x_i,x_j) \\ F(i-1,j)-d \\ F(i,j-1)-d \end{cases}$$

		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1	1	0				
C	-2	0					
C	-3						
A	-4						
T	-5						
T	-6						

Example

Assume a simple model where *S(a,b)* = *1* if *a*=*b* and -5 otherwise.

Also, assume that d = -1

$$F(i,j) = \max \begin{cases} F(i-1,j-1)+s(x_i,x_j) \\ F(i-1,j)-d \\ F(i,j-1)-d \end{cases}$$

		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1	1	0	-1	-2	-3	-4
C	-2	0	-1				
C	-3	-1					
A	-4	-2					
T	-5	-3					
T	-6	-4					

Example

Assume a simple model where *S(a,b) = 1* if *a=b* and -5 otherwise.

Also, assume that d = -1

$$F(i,j) = \max \begin{cases} F(i-1,j-1)+s(x_i,x_j) \\ F(i-1,j)-d \\ F(i,j-1)-d \end{cases}$$

		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1	1	0	-1	-2	-3	-4
C	-2	0	-1	1	0	-1	-2
C	-3	-1	-2	0	2	1	0
A	-4	-2	-3	-1	1	0	-1
T	-5	-3	-4	-2	0	2	1
T	-6	-4	-5	-3	-1	1	3

Example

Assume a simple model where *S(a,b)* = *1* if *a*=*b* and -5 otherwise.

Also, assume that d = -1

$$F(i,j) = \max \begin{cases} F(i-1,j-1)+s(x_i,x_j) \\ F(i-1,j)-d \\ F(i,j-1)-d \end{cases}$$

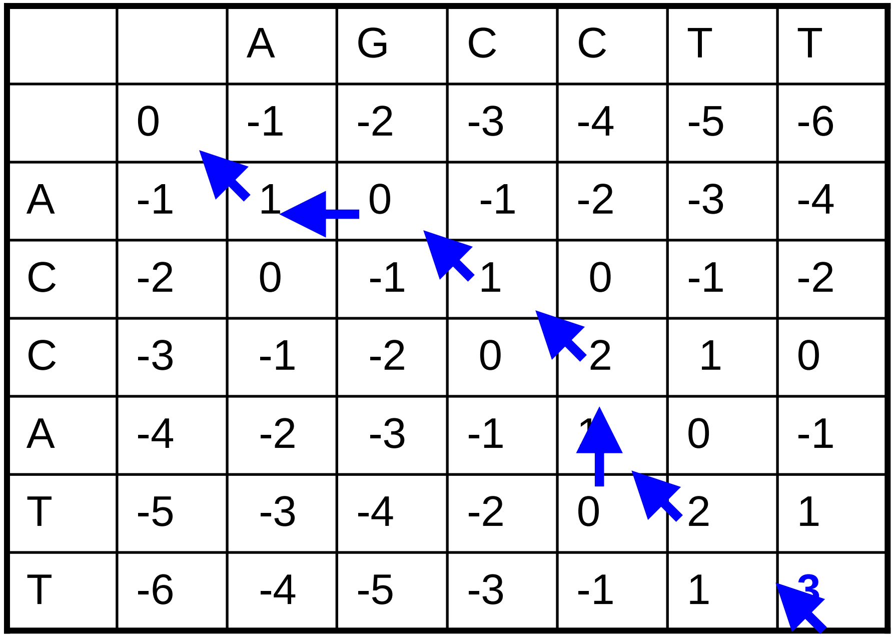
		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1	1	0	-1	-2	-3	-4
C	-2	0	-1	1	0	-1	-2
C	-3	-1	-2	0	2	1	0
A	-4	-2	-3	-1	1	0	-1
T	-5	-3	-4	-2	0	2	1
T	-6	-4	-5	-3	-1	1	↖**3**

Example

Assume a simple model where *S(a,b) = 1* if *a=b* and -5 otherwise.

Also, assume that d = -1

A G C C □ T T
| | | | | | |
A □ C C A T T



		A	G	C	C	T	T
	0	-1	-2	-3	-4	-5	-6
A	-1	1	0	-1	-2	-3	-4
C	-2	0	-1	1	0	-1	-2
C	-3	-1	-2	0	2	1	0
A	-4	-2	-3	-1	1	0	-1
T	-5	-3	-4	-2	0	2	1
T	-6	-4	-5	-3	-1	1	3

Running time

- The running time of an alignment algorithms if $O(n^2)$
- This doesn't sound too bad, or is it?

- The time requirement for doing global sequence alignment is too high in many cases.
- Consider a database with tens of thousands of sequences. Looking through all these sequences for the best alignment is too time consuming.
- In many cases, a much faster heuristic approach can achieve equally good results.

BLAST: Basic Local Alignment Search Tool

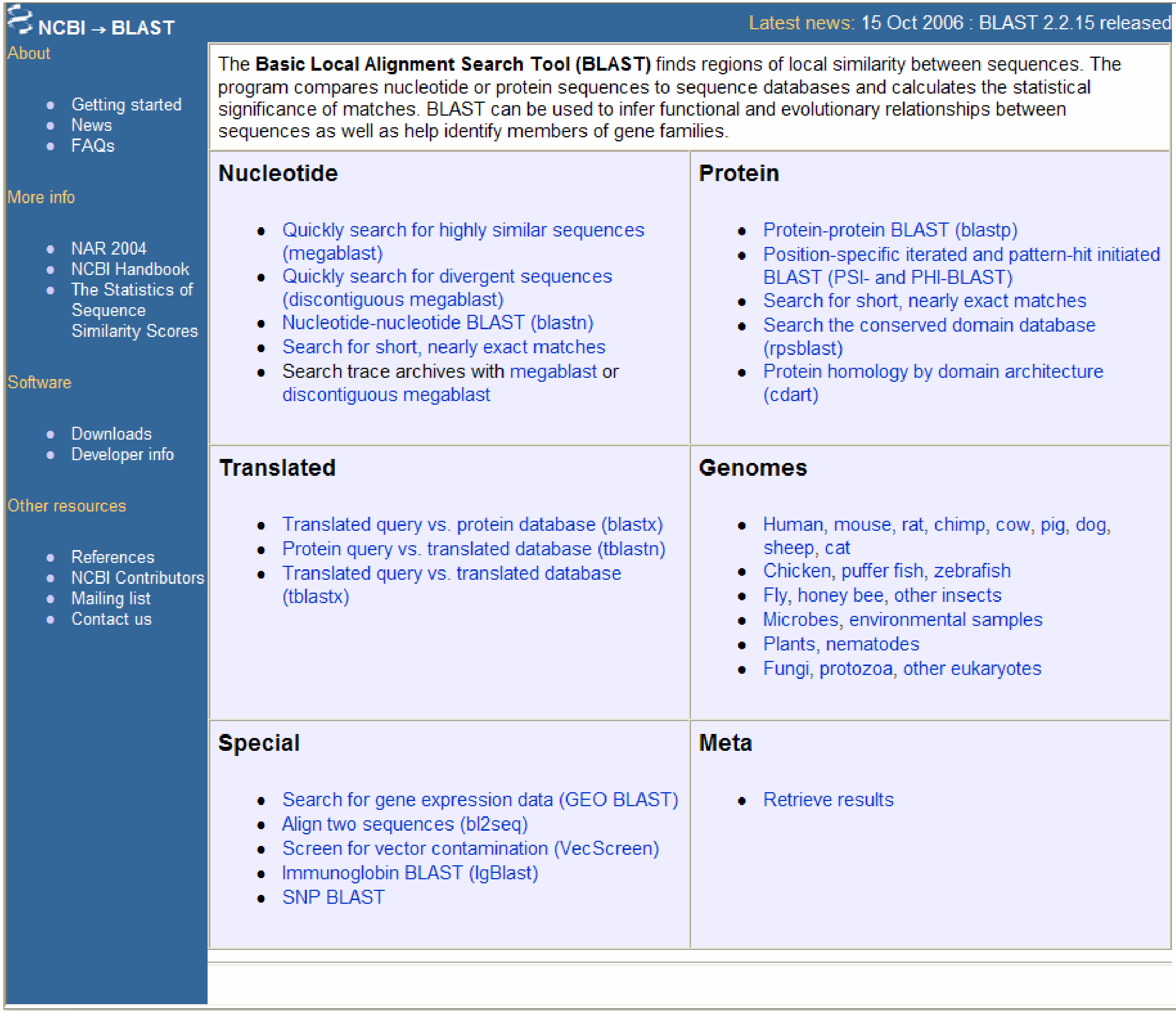
- Heuristic alignment method, first presented in 1990.
- The biggest success of computational biology to date.
- Since it was suggested, a number of new and improved version where presented (psi-BLAST).
- Currently available with almost all public databases.

BLAST (cont.)

- Sequence is composed of a list of ‘words’.
- Uses a dictionary (3 for AA and 11 for nucleotides).
- All matches to database are recorded.

BLAST

- Hits are extended in both direction if they are less than A bases away from each other.
- All sequences reaching a certain score are returned, and a complete alignment is performed.



NCBI → BLAST

Latest news: 15 Oct 2006 : BLAST 2.2.15 released

About

- Getting started
- News
- FAQs

More info

- NAR 2004
- NCBI Handbook
- The Statistics of Sequence Similarity Scores

Software

- Downloads
- Developer info

Other resources

- References
- NCBI Contributors
- Mailing list
- Contact us

The **Basic Local Alignment Search Tool (BLAST)** finds regions of local similarity between sequences. The program compares nucleotide or protein sequences to sequence databases and calculates the statistical significance of matches. BLAST can be used to infer functional and evolutionary relationships between sequences as well as help identify members of gene families.

Nucleotide

- Quickly search for highly similar sequences (megablast)
- Quickly search for divergent sequences (discontiguous megablast)
- Nucleotide-nucleotide BLAST (blastn)
- Search for short, nearly exact matches
- Search trace archives with megablast or discontiguous megablast

Protein

- Protein-protein BLAST (blastp)
- Position-specific iterated and pattern-hit initiated BLAST (PSI- and PHI-BLAST)
- Search for short, nearly exact matches
- Search the conserved domain database (rpsblast)
- Protein homology by domain architecture (cdart)

Translated

- Translated query vs. protein database (blastx)
- Protein query vs. translated database (tblastn)
- Translated query vs. translated database (tblastx)

Genomes

- Human, mouse, rat, chimp, cow, pig, dog, sheep, cat
- Chicken, puffer fish, zebrafish
- Fly, honey bee, other insects
- Microbes, environmental samples
- Plants, nematodes
- Fungi, protozoa, other eukaryotes

Special

- Search for gene expression data (GEO BLAST)
- Align two sequences (bl2seq)
- Screen for vector contamination (VecScreen)
- Immunoglobin BLAST (IgBlast)
- SNP BLAST

Meta

- Retrieve results

Sequence analysis

- Traditional
  - Dynamic programming √
  - Blast √
- Probabilsitic
  - Profile HMMs

Protein families

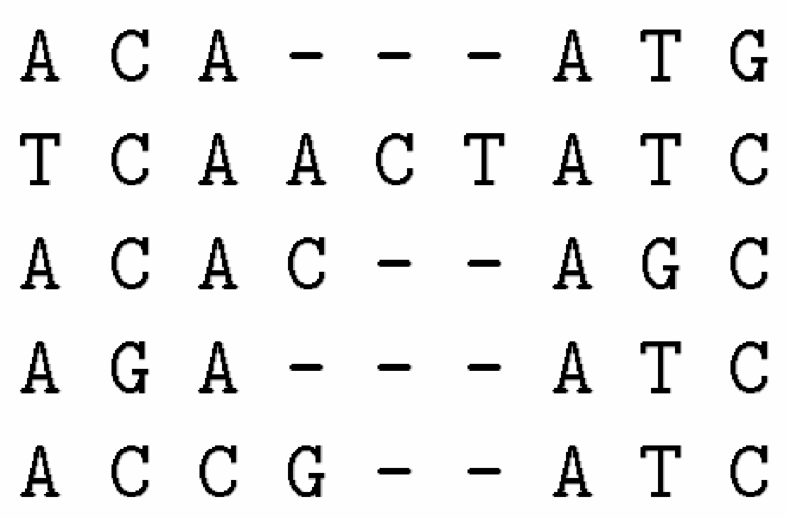
- Proteins can be classified into families (and further into sub families etc.)
- A specific family includes proteins with similar high level functions
- For example:

  - Transcription factors
  - Receptors
  - Etc.

Family assignment is an important first step towards function prediction

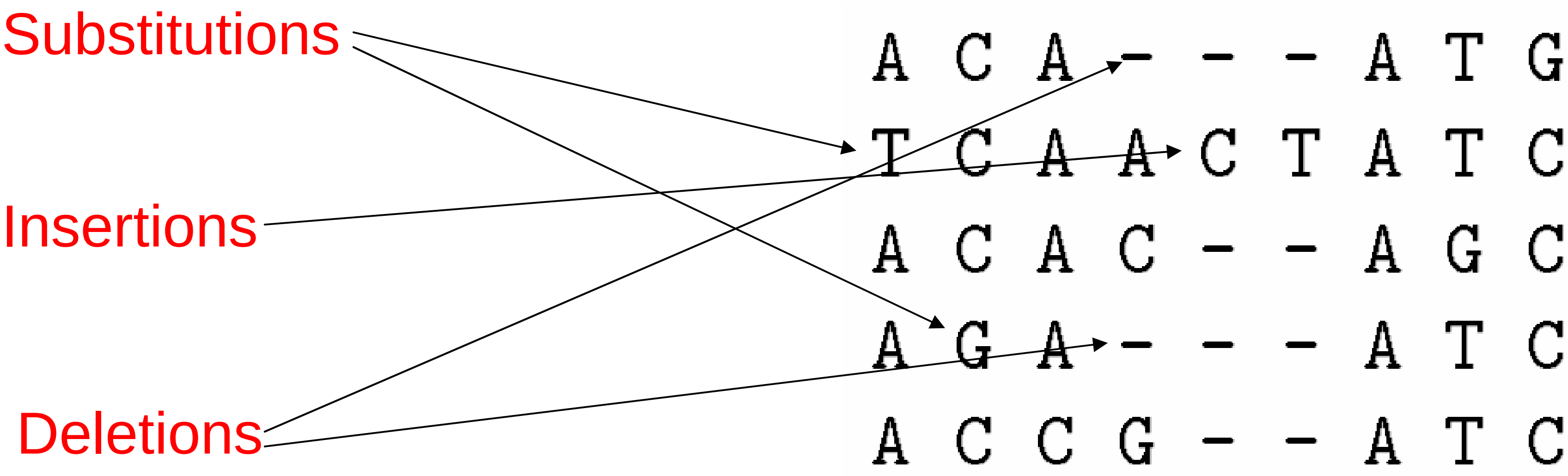
Multiple Alignment Process

- Process of aligning three or more sequences with each other
- We can determine such alignment by generalizing the algorithm to align two sequences
- What’s the complexity of this?



```
A C A - - - A T G
T C A A C T A T C
A C A C - - A G C
A G A - - - A T C
A C C G - - A T C
```

Multiple Alignment: Reasons for differences



Substitutions

Insertions

Deletions

```
A C A - - - A T G
T C A A C T A T C
A C A C - - A G C
A G A - - - A T C
A C C G - - A T C
```

Biological Motivation:

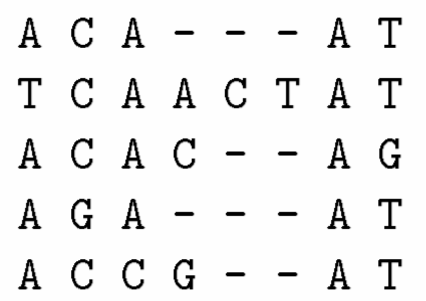
- Given a single amino acid target sequence of unknown protein we want to infer the family of the resulting protein.

Methods for Characterizing a Protein Family

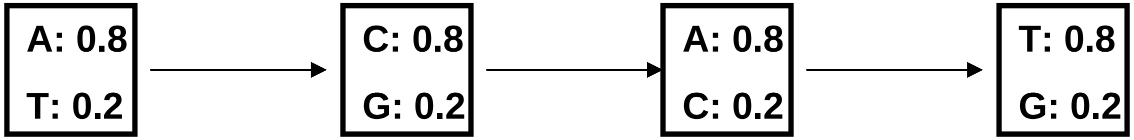
- Objective: Given a number of related sequences, encapsulate what they have in common in such a way that we can recognize other members of the family.
- Some standard methods for characterization:
  - Multiple Alignments
  - Regular Expressions
  - Consensus Sequences
  - Hidden Markov Models

Designing HMMs: Consensus (match) states

We first include states to output the consensus sequence



```
A C A - - - A T
T C A A C T A T
A C A C - - A G
A G A - - - A T
A C C G - - A T
```



A: 0.8
T: 0.2

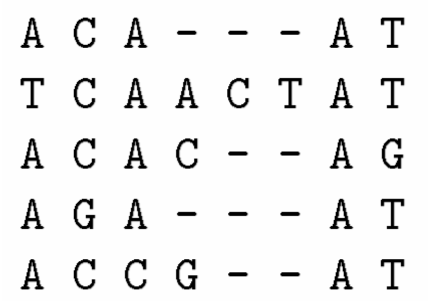
C: 0.8
G: 0.2

A: 0.8
C: 0.2

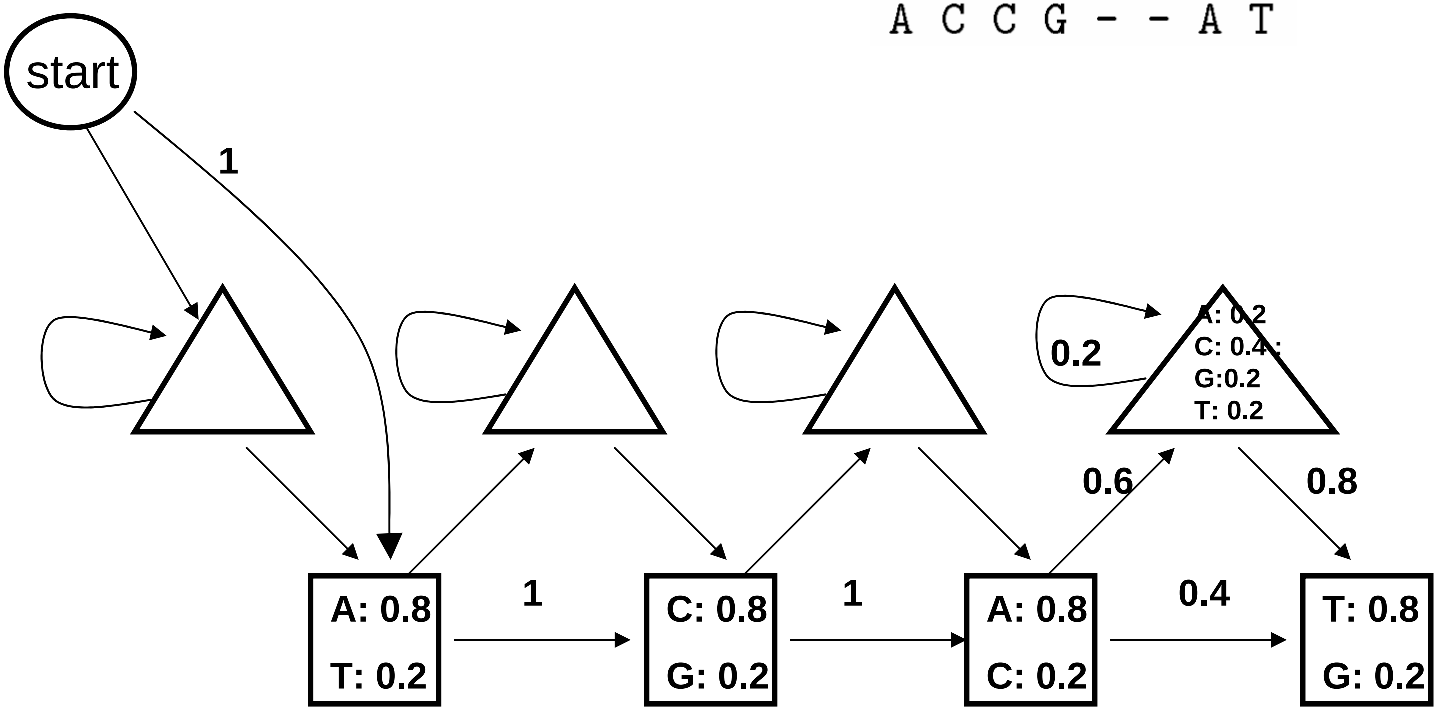
T: 0.8
G: 0.2

Designing HMMs: Insertions

We next add states to allow insertions



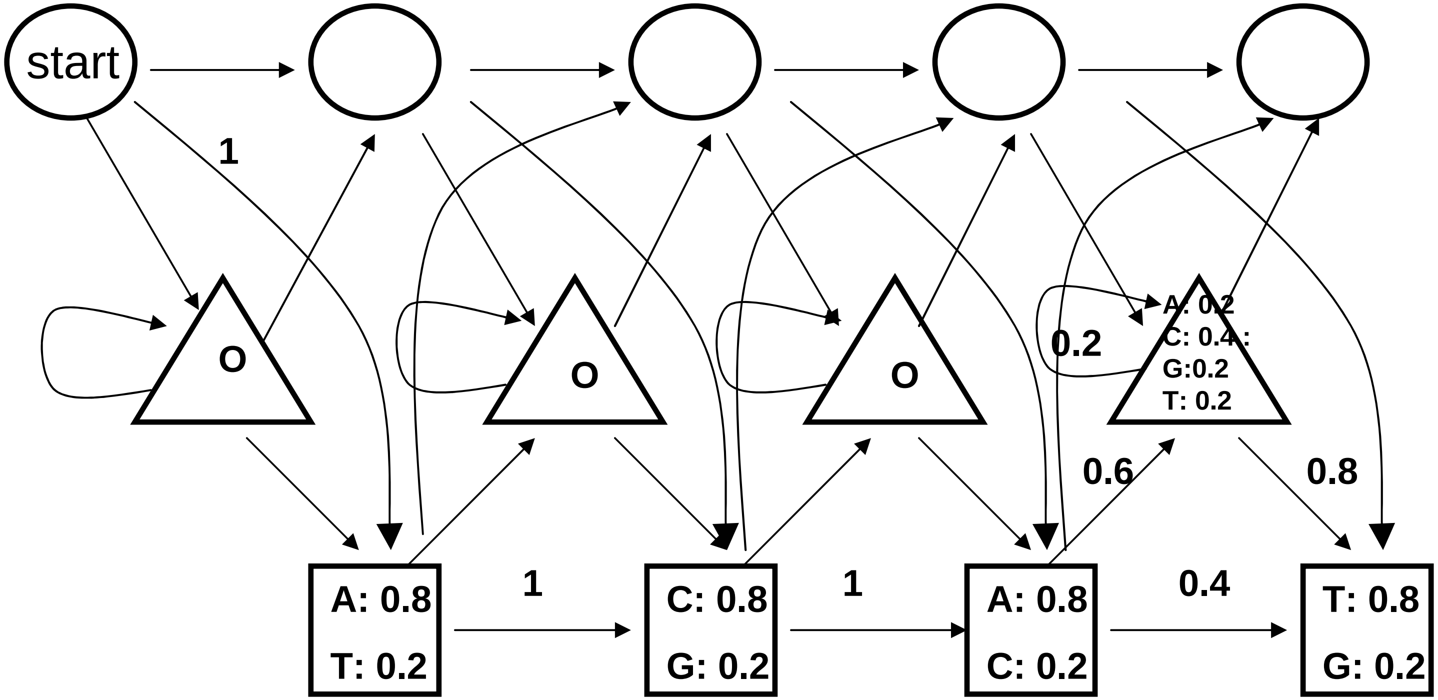
```
A C A - - - A T
T C A A C T A T
A C A C - - A G
A G A - - - A T
A C C G - - A T
```



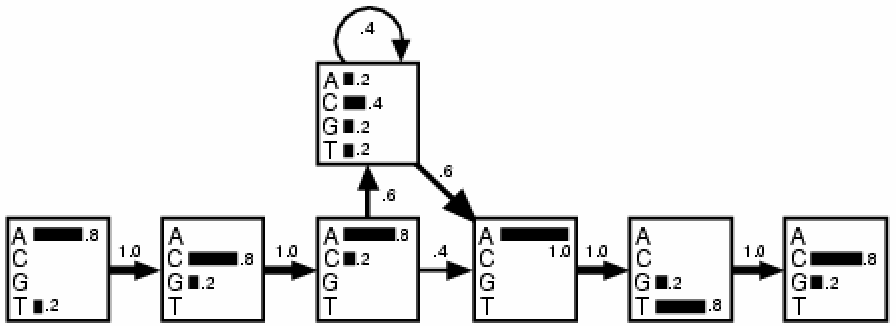
Designing HMMs: Deletions

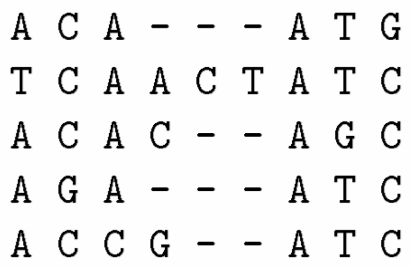
Finally we add states with **no** output to allow for deletions

```
A C A - - - A T
T C A A C T A T
A C A C - - A G
A G A - - - A T
A C C G - - A T
```



Scoring our simple HMM





```
A C A - - - A T G
T C A A C T A T C
A C A C - - A G C
A G A - - - A T C
A C C G - - A T C
```

- #1 - “T G C T - - A G G” *vrs:* #2 - “A C A C - - A T C”
  - HMM:
    - #1 = Score of -0.97 #2 Score of 6.7 (Log odds)

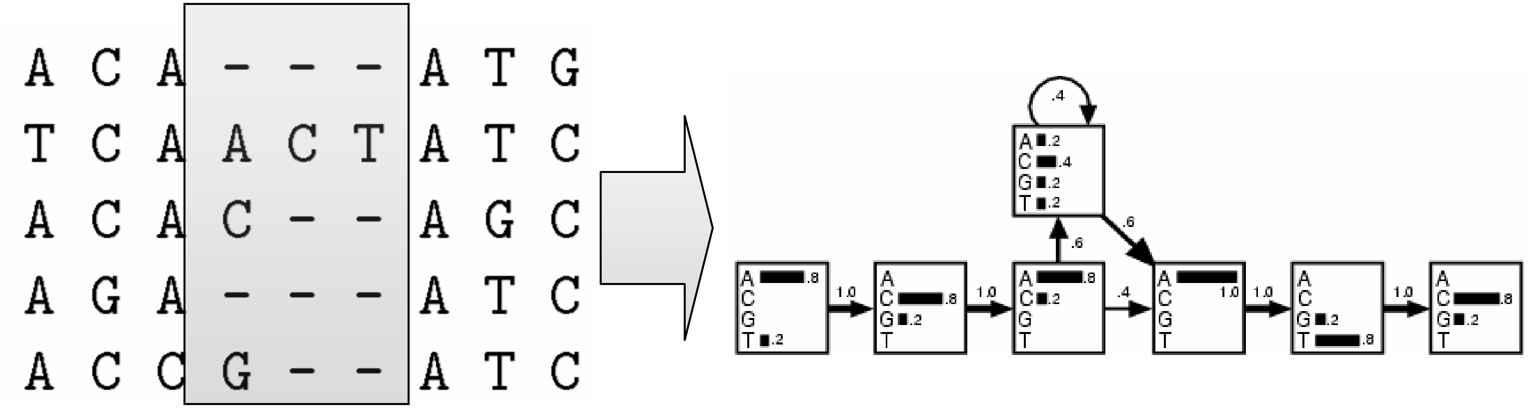
Aligning and Training HMMs

- Training from a Multiple Alignment
- Aligning a sequence to a model
  - Can be used to create an alignment
  - Can be used to score a sequence
  - Can be used to interpret a sequence
- Training from unaligned sequences

Training from an existing alignment

- Start with a predetermined number of states in your HMM.
- For each position in the model, assign a column in the multiple alignment that is relatively conserved.
- Emission probabilities are set according to amino acid counts in columns.
- Transition probabilities are set according to how many sequences make use of a given delete or insert state.

Remember the simple example



```
A C A - - - A T G
T C A A C T A T C
A C A C - - A G C
A G A - - - A T C
A C C G - - A T C
```

- Chose six positions in model.
- Highlighted area was selected to be modeled by an insert due to variability.
- Can also do neat tricks for picking length of model, such as model pruning.

Aligning sequences to a model

- Now that we have a profile model, let’s use it!
- Compute the likelihood of the best set of states for this sequence
- We know how to do this: The Viterbi algortthm
- Time: O(N*M)

So… what do we do with an alignment to a model?

- Design statistical tests to determine how likely it is to get this score from a random (gene) sequence
- Use several protein family models for classifying new proteins, assign protein to most highly scoring family.

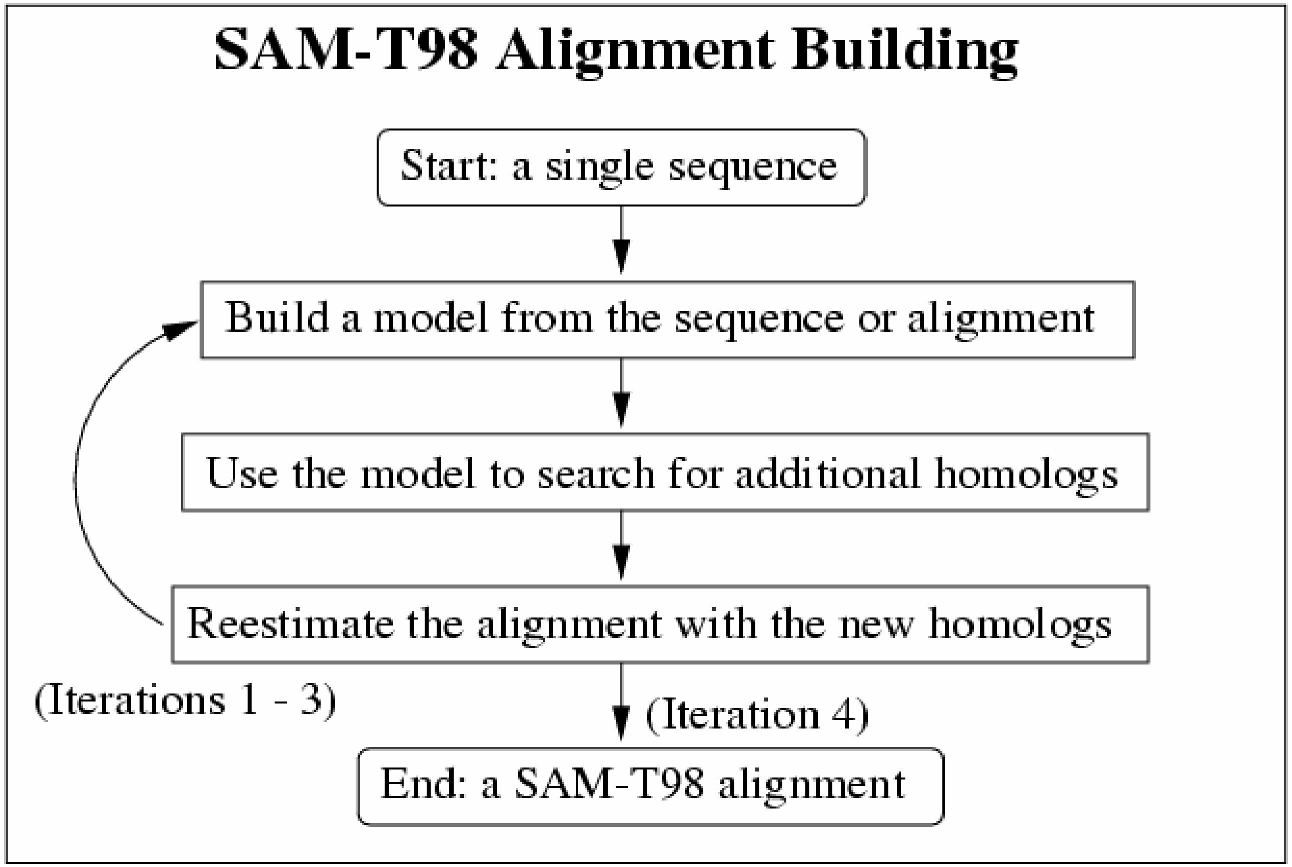
Training from unaligned sequences

- Baum-Welch algorithm
  - Start with a model whose length matches the average length of the sequences and with random emission and transition probabilities.
  - Align all the sequences to the model.
  - Use the alignment to alter the emission and transition probabilities
  - Repeat. Continue until the model stops changing
- By-product: It produces a multiple alignment

Training from unaligned continued

- Advantages:
  - You take full advantage of the expressiveness of your HMM.
  - You might not have a multiple alignment on hand.
- Disadvantages:
  - HMM training methods are local optimizers, you may not get the best alignment or the best model unless you're very careful.
  - Can be alleviated by starting from a logical model instead of a random one.

How do we build a model using only one sequence?

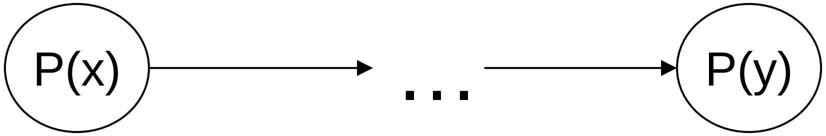

SAM-T98 Alignment Building
Start: a single sequence
Build a model from the sequence or alignment
Use the model to search for additional homologs
Reestimate the alignment with the new homologs
(Iterations 1 - 3)
(Iteration 4)
End: a SAM-T98 alignment

Profile HMM Effectiveness Overview

- Advantages:
  - Very expressive profiling method
  - Transparent method: You can view and interpret the model produced
  - Very effective at detecting remote homologs
- Disadvantages:
  - Slow – full search on a database of 400,000 sequences can take 15 hours
  - Have to avoid over-fitting and locally optimal models

Limitations

- **Markov Chains**
  - Probabilities of states are supposed to be independent



  - P(y) must be independent of P(x), and vice versa
  - This usually isn’t true