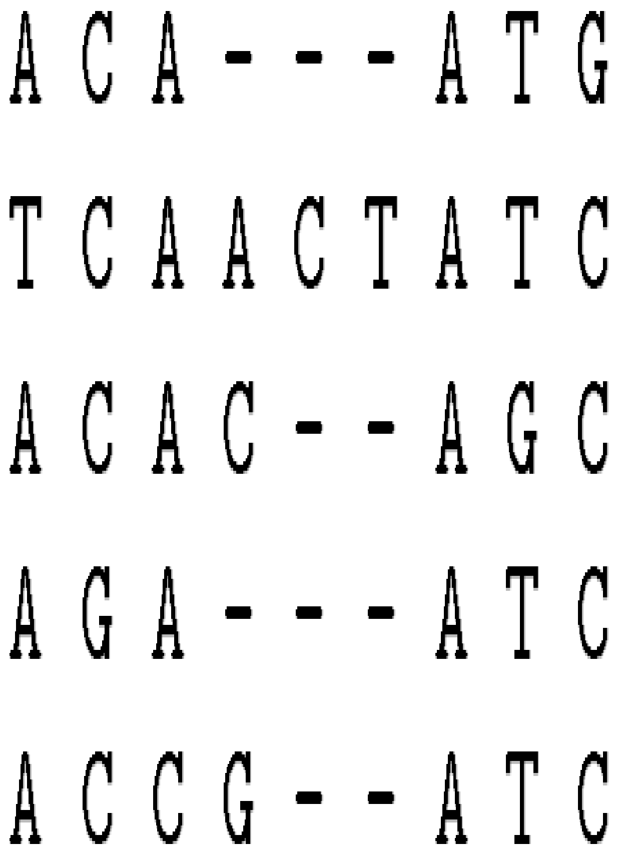
Profile Hidden Markov Models

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

A Characterization Example



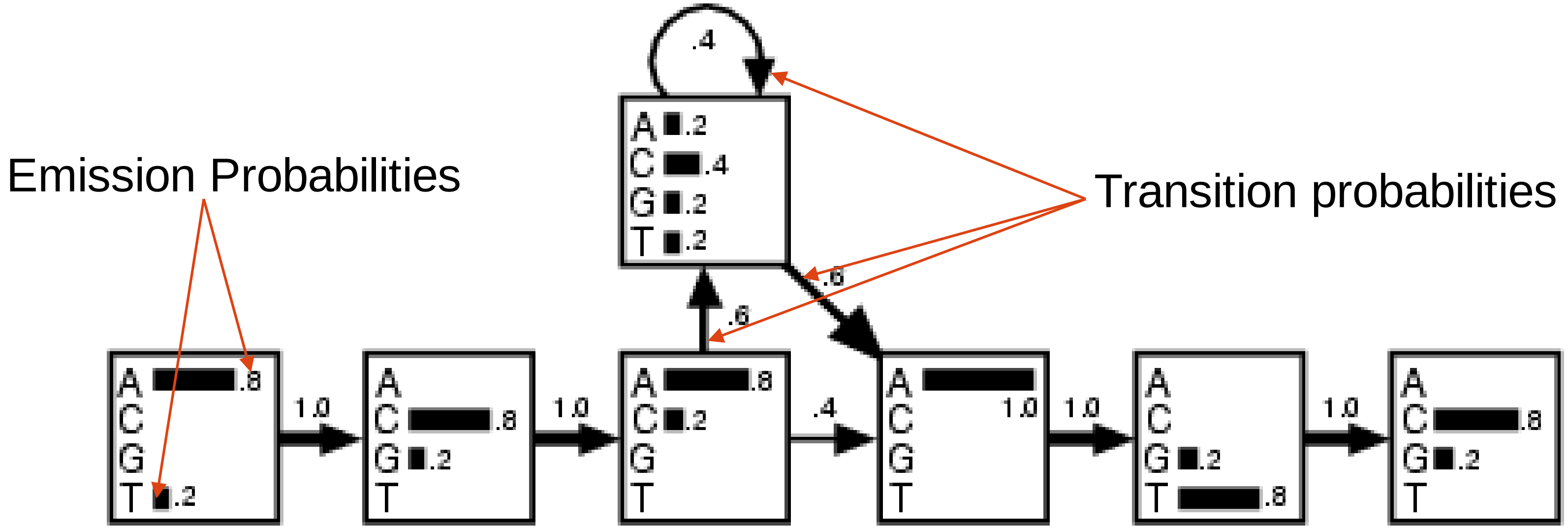
```
A C A - - - A T G
T C A A C T A T C
A C A C - - A G C
A G A - - - A T C
A C C G - - A T C
```

Example borrowed from Salzberg, 1998

How could we characterize this (hypothetical) family of nucleotide sequences?

- Keep the Multiple Alignment
- Try a regular expression
  [AT] [CG] [AC] [ACTG]* A [TG] [GC]
  - But what about?
    - T G C T - - A G G *vrs*
    - A C A C - - A T C
- Try a consensus sequence:
  A C A - - - A T C
  - Depends on distance measure

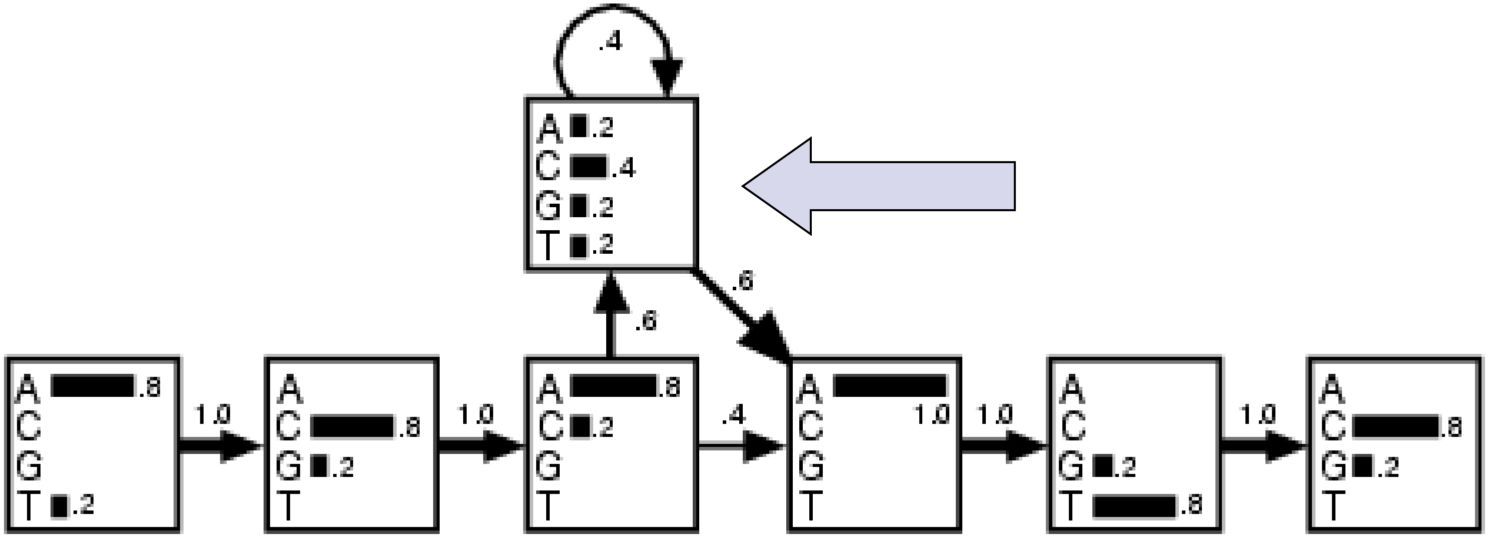
HMMs to the rescue!


Emission Probabilities

Transition probabilities

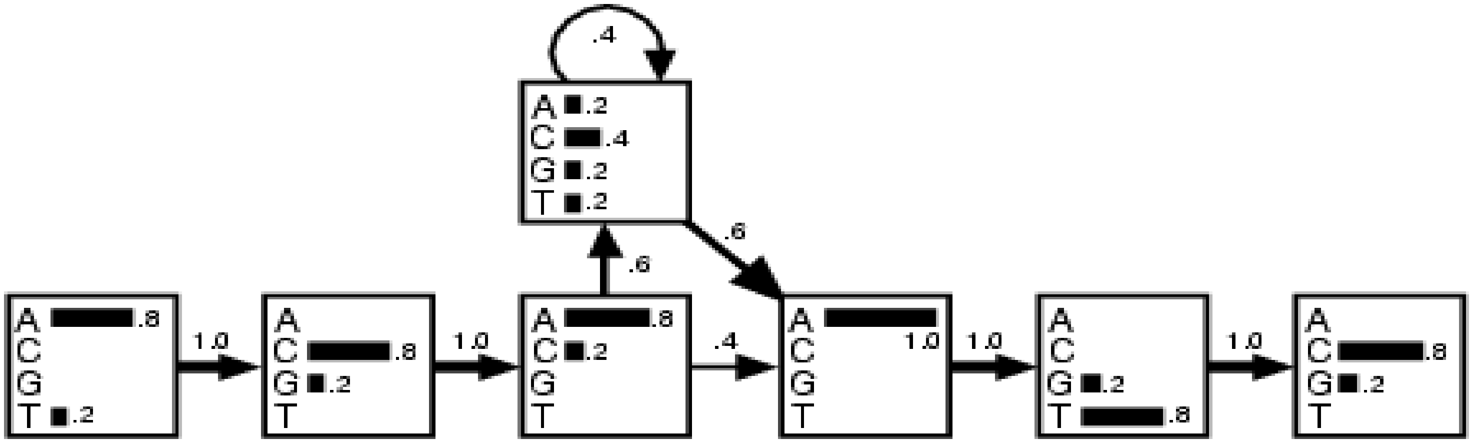
A	C	A	–	–	–	A	T	G
T	C	A	A	C	T	A	T	C
A	C	A	C	–	–	A	G	C
A	G	A	–	–	–	A	T	C
A	C	C	G	–	–	A	T	C

Insert (Loop) States



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

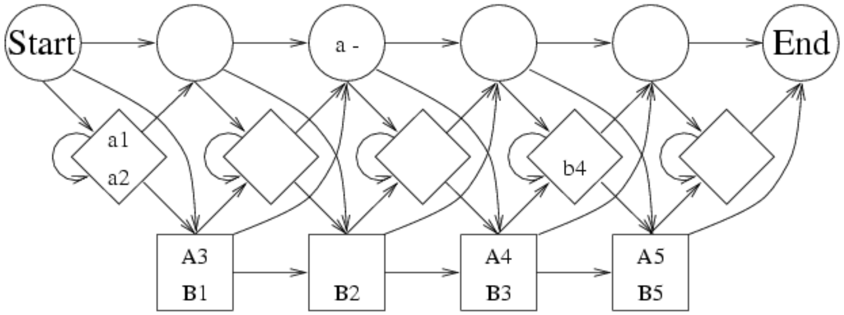
Scoring our simple HMM



- #1 - “T G C T - - A G G” *vrs:* #2 - “A C A C - - A T C”
  - Regular Expression ([AT] [CG] [AC] [ACTG]* A [TG] [GC]):
    - #1 = Member #2: Member
  - HMM:
    - #1 = Score of 0.0023% #2 Score of 4.7% (Probability)
    - #1 = Score of -0.97 #2 Score of 6.7 (Log odds)

Standard Profile HMM Architecture

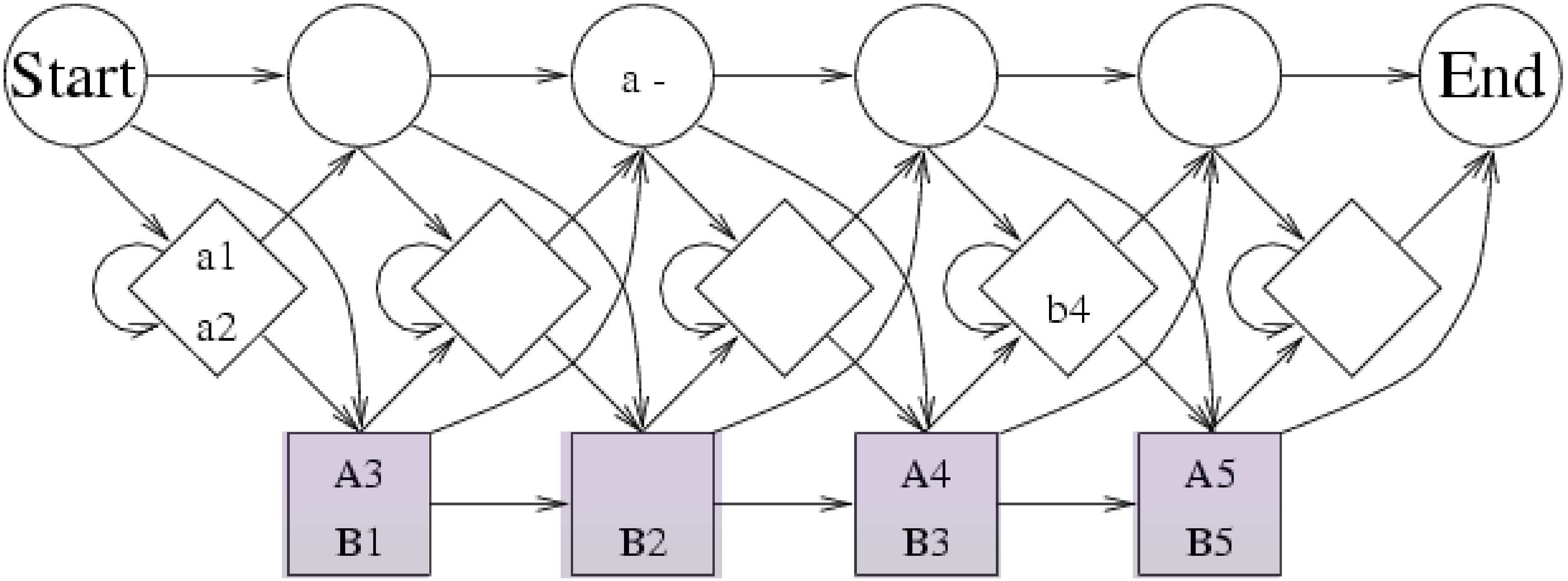
- Three types of states:
  - Match
  - Insert
  - Delete
- One delete and one match per position in model
- One insert per transition in model
- Start and end "dummy" states



```
a1  a2  A3      -     A4  .  A5
 .   .  B1      B2    B3  b4 B5
```

Example borrowed from Cline, 1999

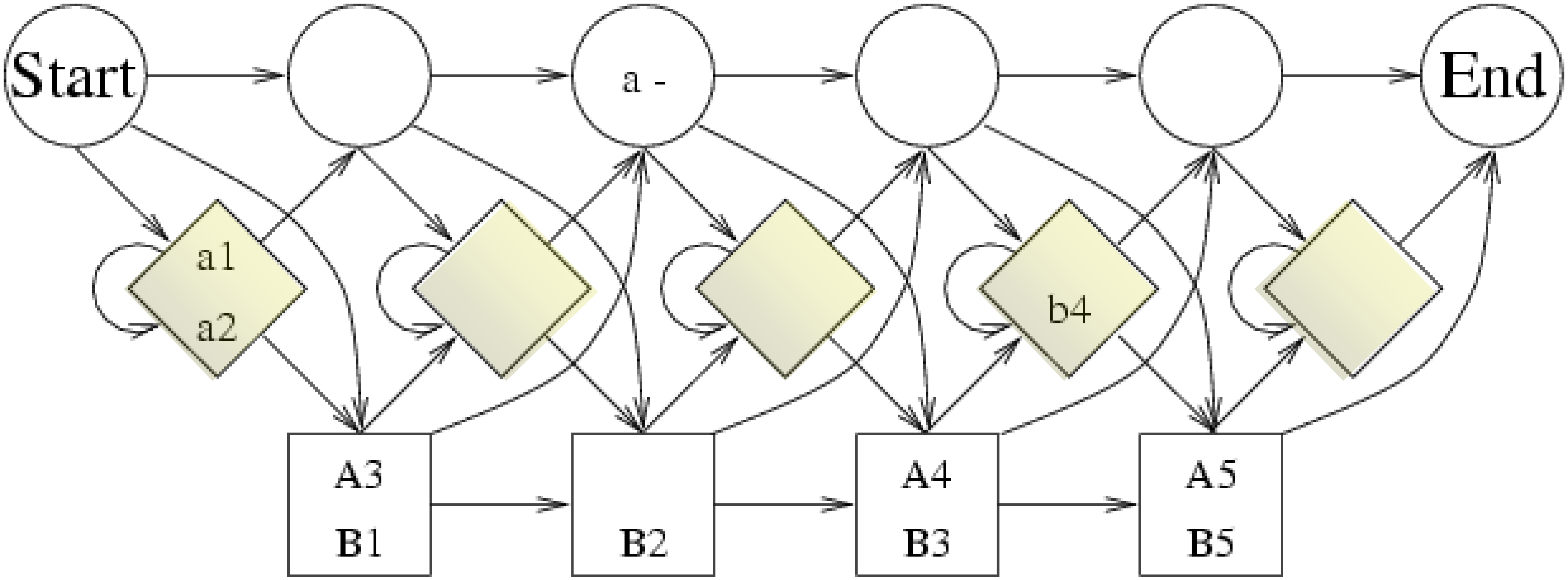
Match States



a1	a2	A3	–	A4	.	A5
.	.	B1	B2	B3	b4	B5

Example borrowed from Cline, 1999

Insert States

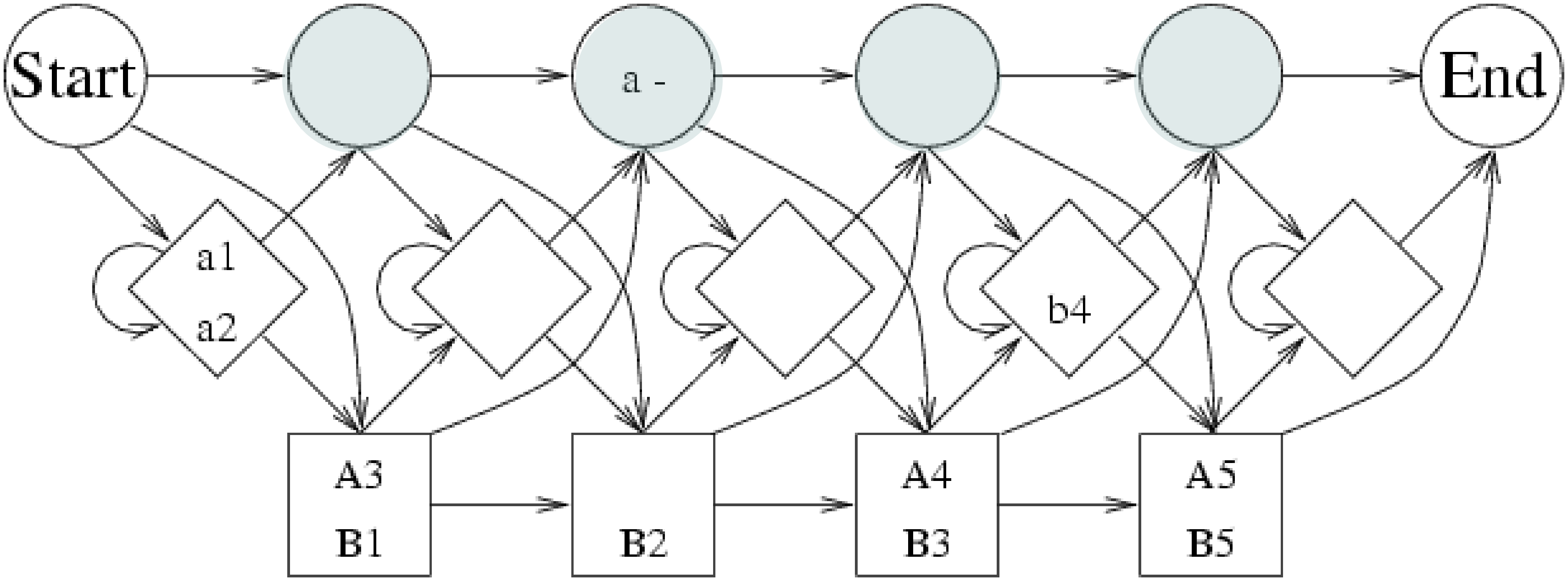


a1 a2 A3 – A4 . A5

. . B1 B2 B3 b4 B5

Example borrowed from Cline, 1999

Delete States



```
a1  a2  A3   –   A4  .   A5
 .   .  B1   B2  B3  b4  B5
```

Example borrowed from Cline, 1999

Aligning and Training HMMs

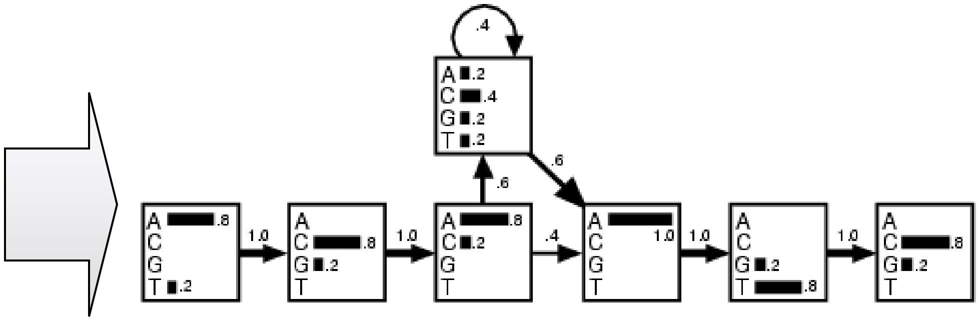
- Training from a Multiple Alignment
- Aligning a sequence to a model
  - Can be used to create an alignment
  - Can be used to score a sequence
  - Can be used to interpret a sequence
- Training from unaligned sequences

Training from an existing alignment

- This process what we've been seeing up to this point.
  - 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!
- Try every possible path through the model that would produce the target sequence
  - Keep the best one and its probability.
- Viterbi alg. has been around for a while
  - Dynamic Programming based method
  - Time: O(N*M) Space: O(N*M)
    - (Assuming a constant # of transitions per state)
    - N = Length of sequence, M = # of states in HMM

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

- Align a bunch of sequences to the model, and get a new multiple alignment.
- Align a single sequence to the model and get a numerical score stating how well it fits the model
  - “Find me all sequences in the database that match this family profile X with a log odds score of at least Y”
- Align a single sequence to the model, and get a description of its columns
  - “Columns 124 and 125 map to insert states of family Y, I wonder what that means?”

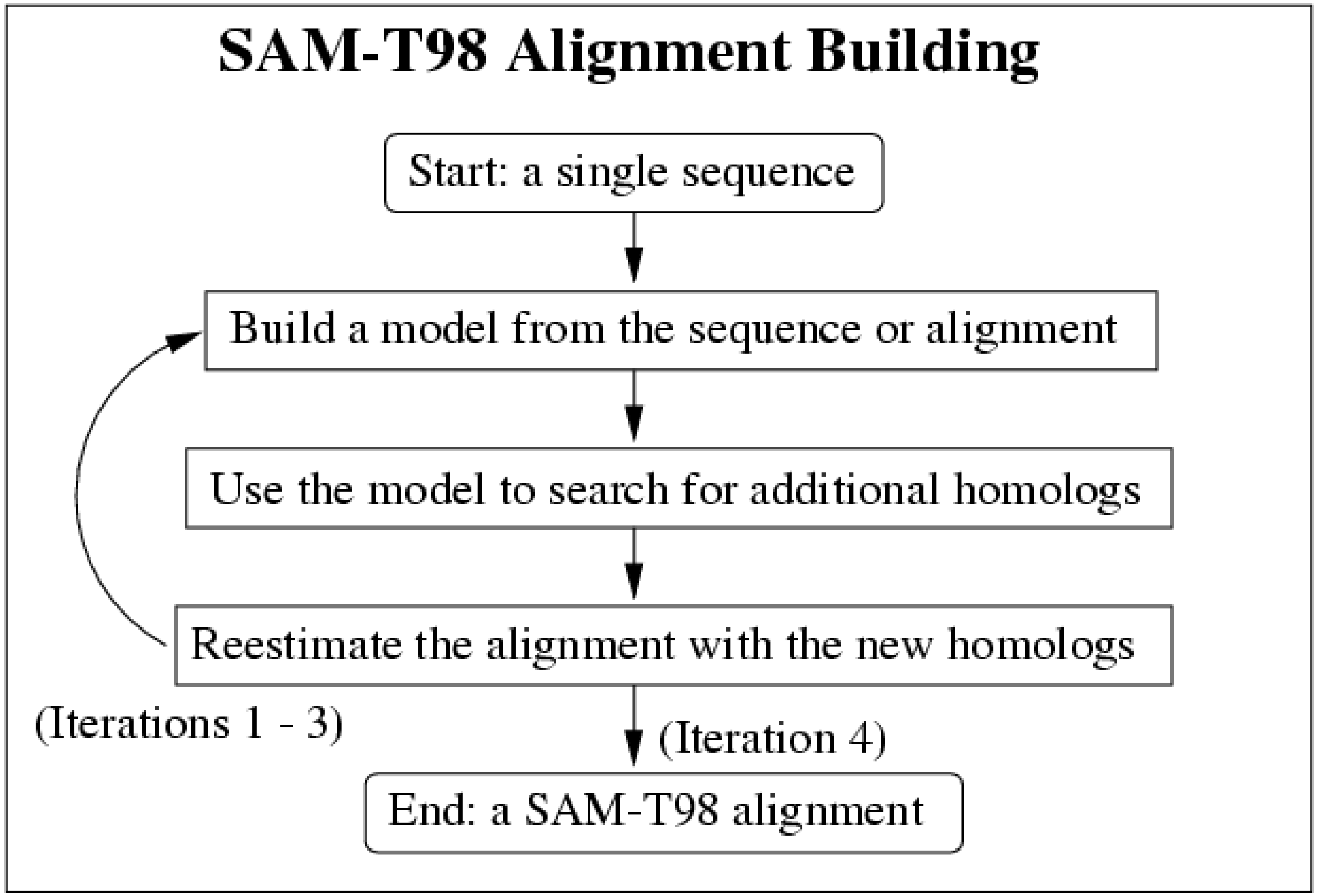
Training from unaligned sequences

- One method:
  - 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 produced 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 (not HMMER 3)
  - Have to avoid over-fitting and locally optimal models

pHMMS tools

- Tools
  - SAM
  - HMMER
- GUI
  - HMMVE
  - UGENE (plugin)
- Database
  - Pfam