Bioinformatics 1 -- lecture 17

Comparing methods

- ROC

How to find motifs, signatures, footprints

- MEME
- Gibbs sampling
- K-means clustering

What to do about low complexity regions: Repeats, Satellites and the role of Transposable Elements in creating them.

- masking repeats
- null models for repeat alignment
- word HMMs for repeats

Follow-up for HW4: smart pseudocounts for profiles

Normal profile calculation uses the sequence weights to sum the amino acid probabilities. If an AA is never observed, then P_{ij} is **zero**.

Sum of sequence weights method:

$$P_{ij} = \frac{\sum_{k \in \left(s_{kj} = aa_i\right)} w_k}{\sum_{k = all\ \ seqs} w_k}$$

s_{kj} is sequence k, position j.

Extrapolated profile method: Use the BLOSUM substitution matrix $S_{i\text{->}j}$ to "extrapolate" from the observed data. Here we are adding *predicted un-observed amino acids*.

$$P_{ij} = \frac{\sum_{k \in \left(s_{kj} = aa_i\right)} w_k + \sum_{k \in \left(s_{kj} = aa_{m \neq i}\right)} \varepsilon w_k S_{m \to j}}{\sum_{k = all\ \ seqs} w_k}$$

Smart pseudocounts: "I didn't see a L, but I saw a V, and L substitutes for V, so let's add some L anyway."

How do you compare two models given T/F data?

Accuracy = percent of the predictions that are correct, of the ones that were made.

Coverage = number of possible predictions that were actually predicted.

Confidence = a score to sort the predictions. A more confident prediction should be a more accurate one. This could be the score itself.

Accuracy = $T^+/(T^+ + F^+)$

Coverage = $T^+/(T^+ + F^-)$

	+	–
≠null	T+	F-
=null	F+	T-

False positive rate

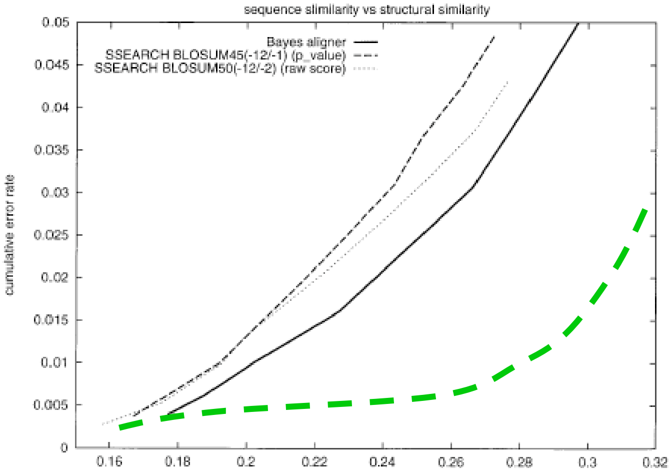
false positive = Type 1 error = error of the first kind

A more detailed description of the method is the rate of *false positive* predictions, which can be a function of the *score*.
A better method has a lower false positive rate.

To calculate, sort the scores and assign T or F to each score. The false positive rate for each score is the percent of the false scores that are above that score.

$$\text{fpr}(x) = \frac{\text{number of false positives above x}}{\text{total number of false positives}}$$

(FPR does not provide one handy number.)



Receiver Operator Characteristic (ROC)

- A way to describe the whole set of scores with a single number.

- Each score has a T or F.

- Sort the scores.

- Starting from the highest scoring, draw a vector **up** for a true, to the **right** for a false.

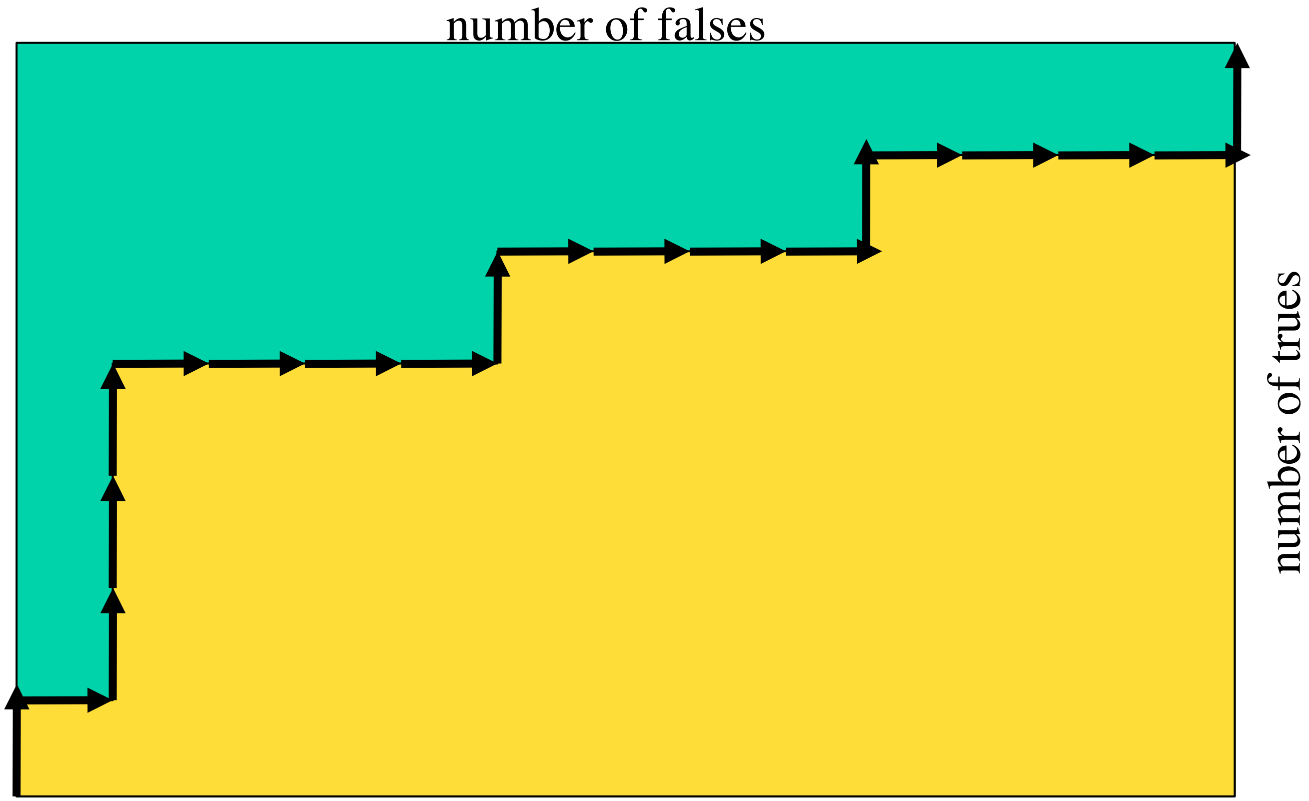
- Calculate ROC = the normalized area under this curve.

- If all of the **true** scores are greater that the greatest **false** score, then ROC = 1.0.

- $0. \leq ROC \leq 1.$

0.990 T
0.978 F
0.972 T
0.966 T
0.951 T
0.902 F
0.880 F
0.811 F
0.803 F
0.792 T
0.766 F
0.751 F
0.723 F
0.696 F
0.688 T
0.666 F
0.651 F
0.623 F
0.596 F
0.488 T

ROC score



Sort the scores, for each score move up one if it is true, right one if it is false.

The area under the curve, divided by the total, is the ROC score. $0 \leq ROC \leq 1$.

In class exercise: calculate ROC score

Which method is better?

0.811	T
0.972	T
0.766	T
0.990	F
0.966	T
0.951	F
0.803	F
0.792	F
0.503	F
0.978	T
0.478	F

Method A

4	T
39	F
44	T
44	T
40	T
1	F
39	F
29	F
10	F
44	F
45	T

Method B

motifs signatures & footprints

Motifs exist due to selective pressure

Selective pressure for:

structure -- protein motifs

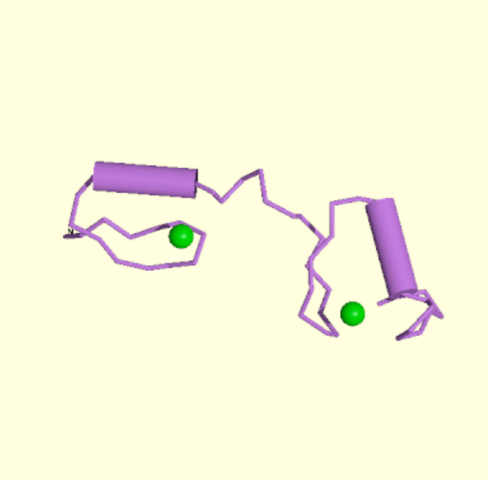
- folding units
- fibrous proteins
- coiled coils
- transmembrane helices

function -- protein motifs

- active site
- binding motifs
- signal sequences

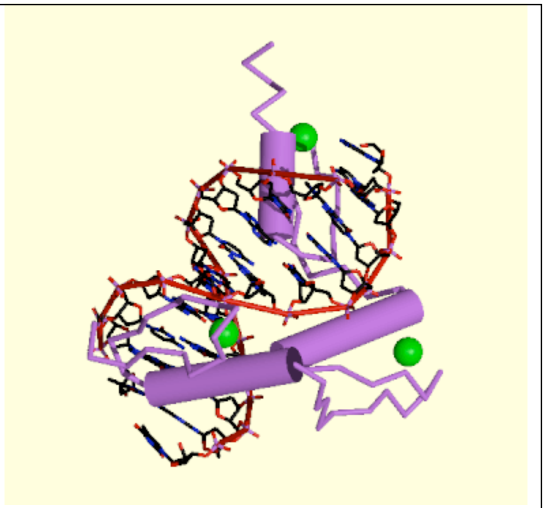
expression -- DNA motifs

- transcription regulation
- chromatin binding



Example: selection for structure

Zinc finger motif



```
            x   x
         x         x
        x           x
        x           x
        x           x
        x           x
         C         H
      x     \   /     x
    x         Zn        x
      x     /   \     x
         C         H
x  x  x  x  x          x  x  x  x  x
```

two Cystines separated by 2 or 4 residues

two Histidines separated by 3 or 5 residues

`C-x(2,4)-C-x(3)-[LIVMFYWC]-x(8)-H-x(3,5)-H`

Loop must be length 12.
4th position in loop must be hydrophobic

Example: selection for function

ER targeting sequence

```
[KRHQSA]-[DENQ]-E-L
```

N-glycosylation

```
N-{P}-[ST]-{P}
```

Tyrosine phosphorylation

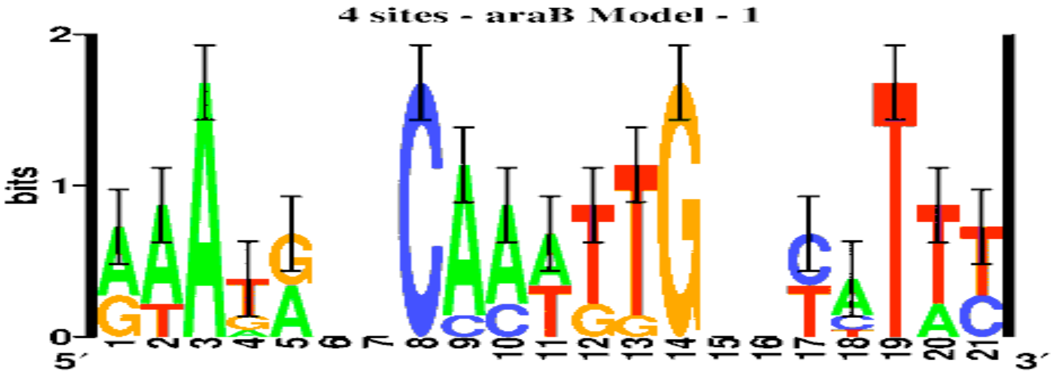
```
[RK]-x(2)-[DE]-x(3)-Y or [RK]-x(3)-[DE]-x(2)-Y
```

C-terminal prenylation

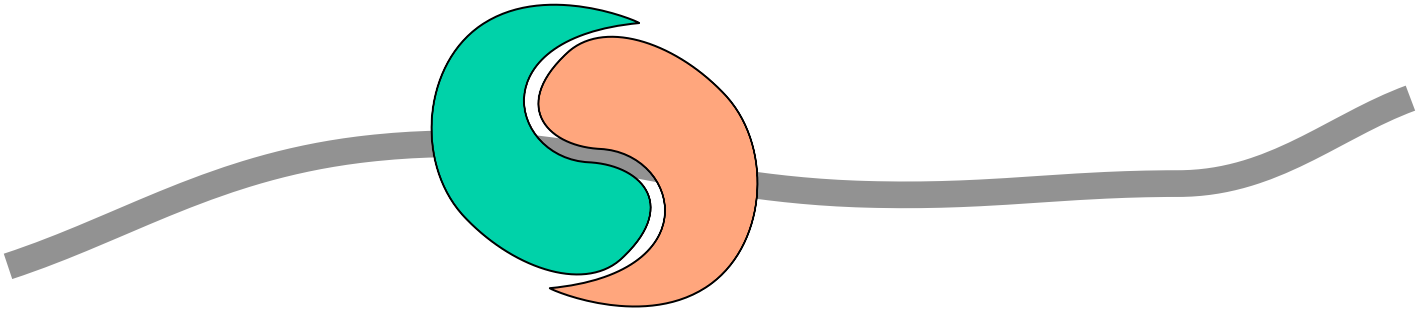
```
C-{DENQ}-[LIVM]-x
```

Example: selection for expression

Transcription factor binding site



Palindromy in TF footprints (binding sites) is due to the symmetry of the TFs, which are almost invariably dimeric.



MEME

motif elucidation by expectation/maximization

How do we, simultaneously, find the <u>motif</u> and the <u>locations</u> of the motif in a set of sequences?

...or...

Where is it, and ... what am I looking for??

Initial guess of motif location

...and therefore of the motif

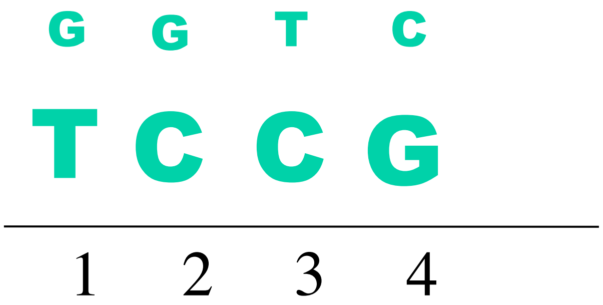
From the motif locations, you make a profile model.

AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

TATTGCTCTCCGCAGC

Motif
Model:
L=4



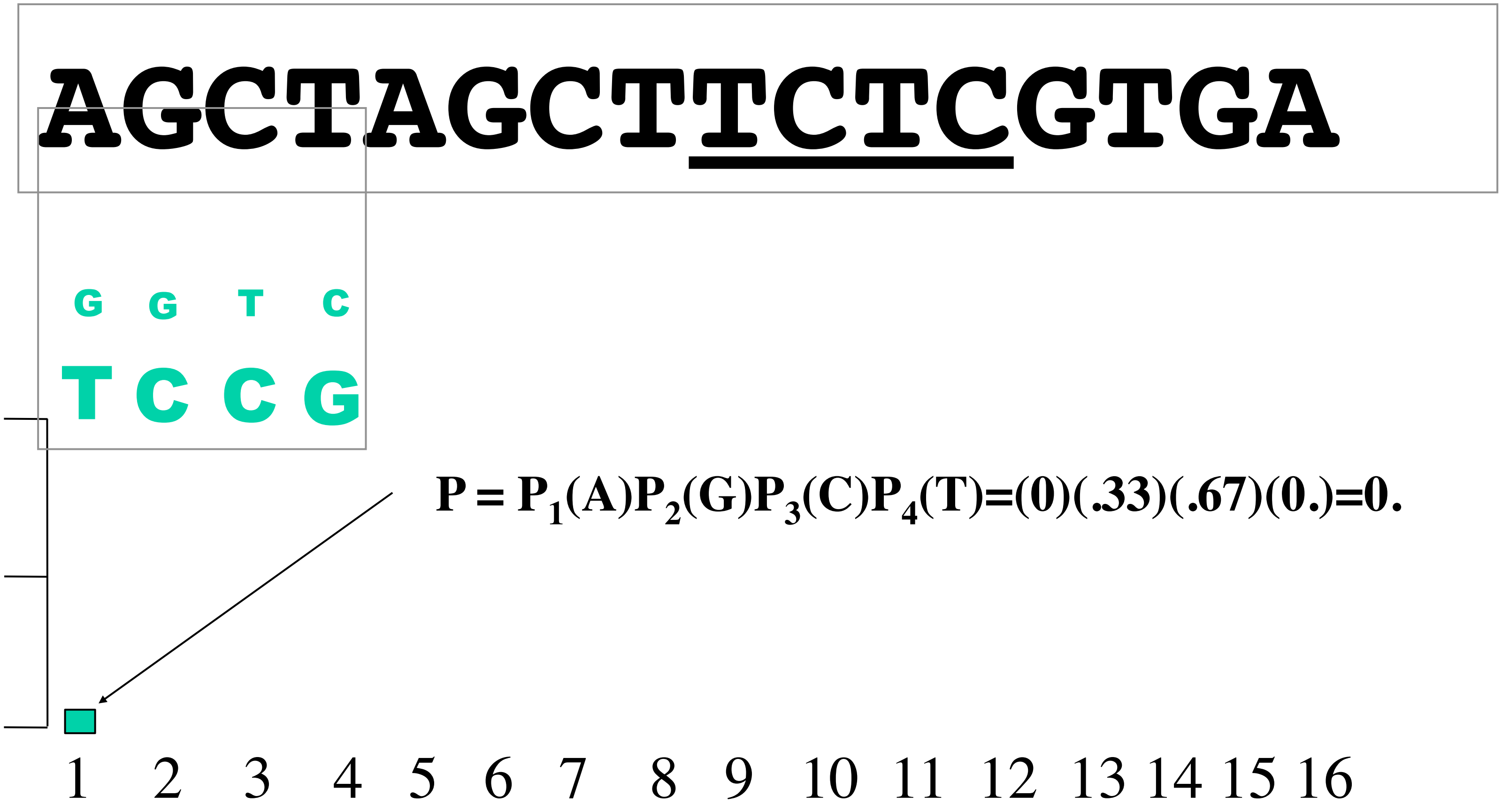
P_1 = 2/3 T, 1/3 G

initial guesses underlined

MEME

Calculate the probability score for each position

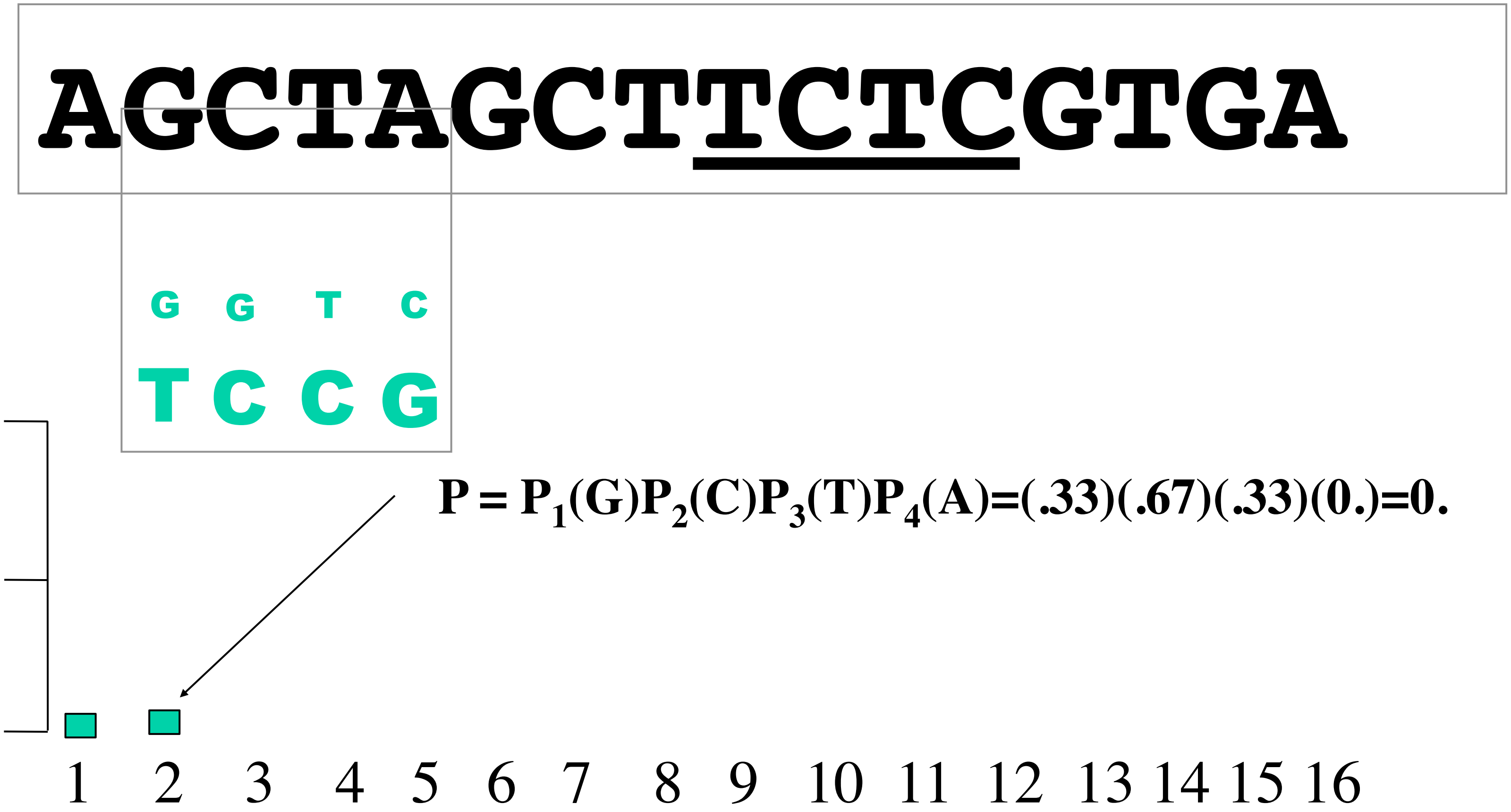
From the profile model and the sequence, get probability scores.



$$P = P_1(A)P_2(G)P_3(C)P_4(T)=(0)(.33)(.67)(0.)=0.$$

Calculate the probability score for each position

Slide the model along the sequence to get the next score.

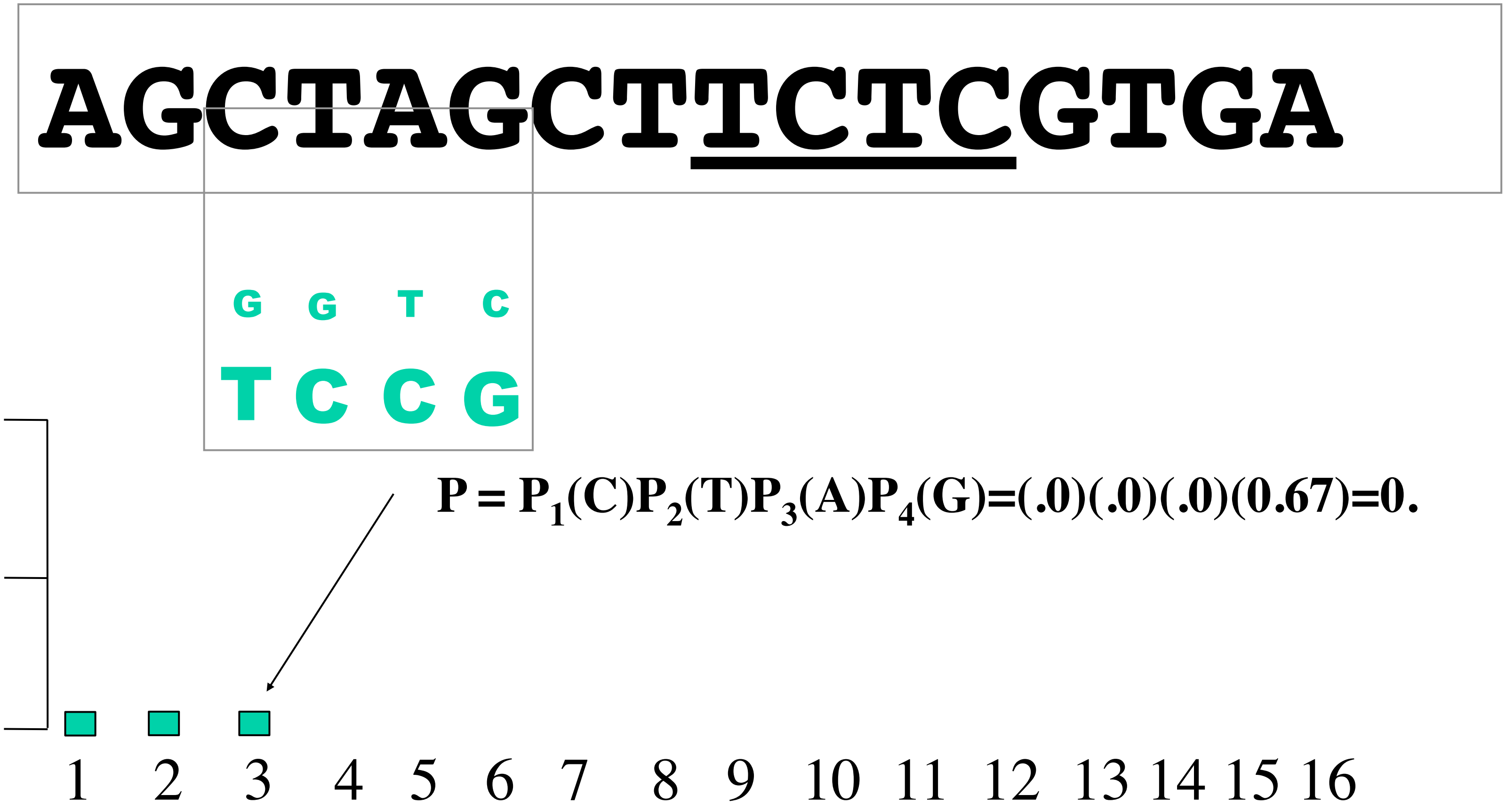


$P = P_1(G)P_2(C)P_3(T)P_4(A)=(.33)(.67)(.33)(0.)=0.$

MEME

Calculate the probability score for each position

Slide the model along the sequence to get the next score.

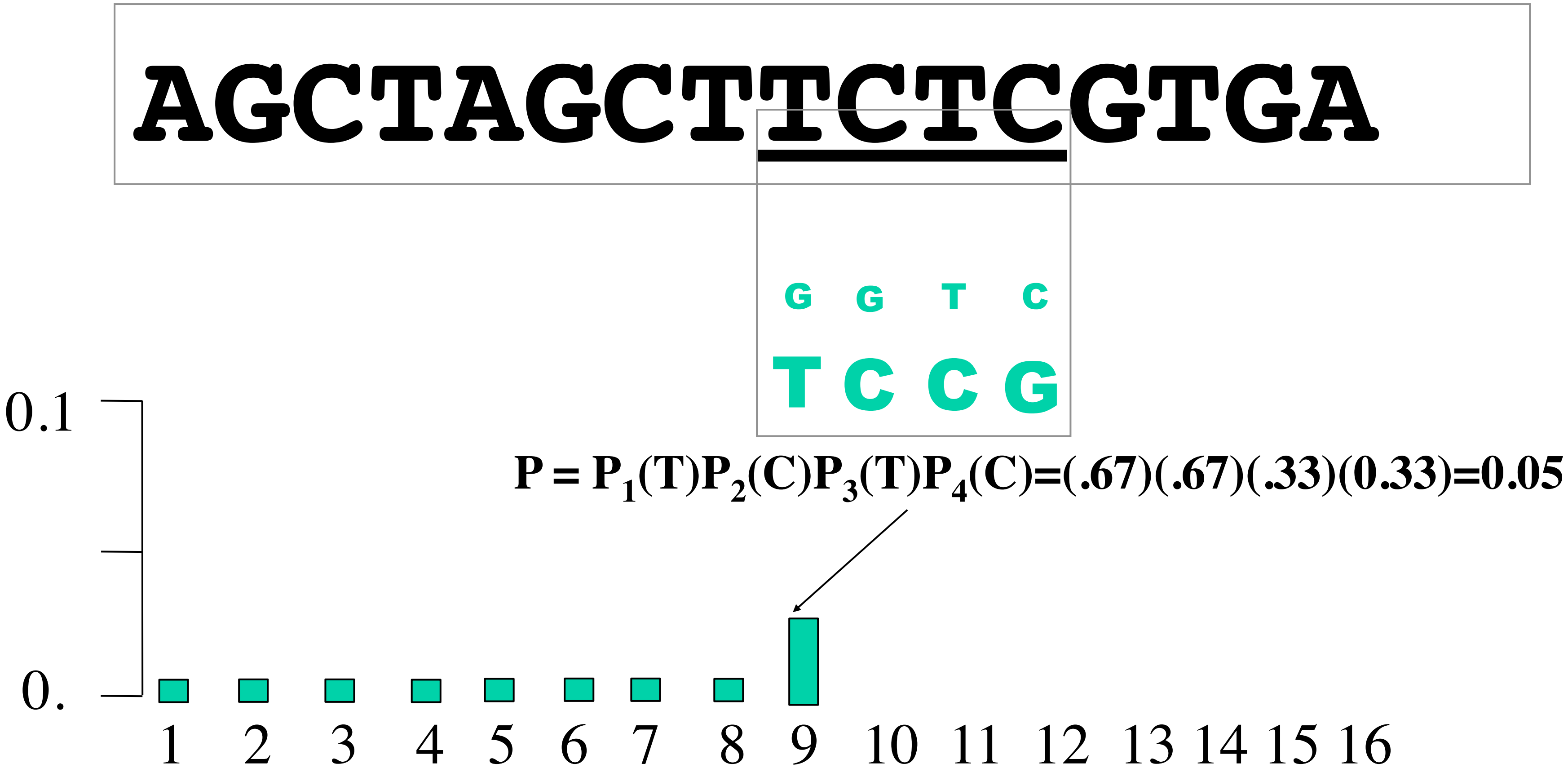


$P = P_1(C)P_2(T)P_3(A)P_4(G)=(.0)(.0)(.0)(0.67)=0.$

MEME

Calculate the probability score for each position

Slide the model along the sequence to get the next score.

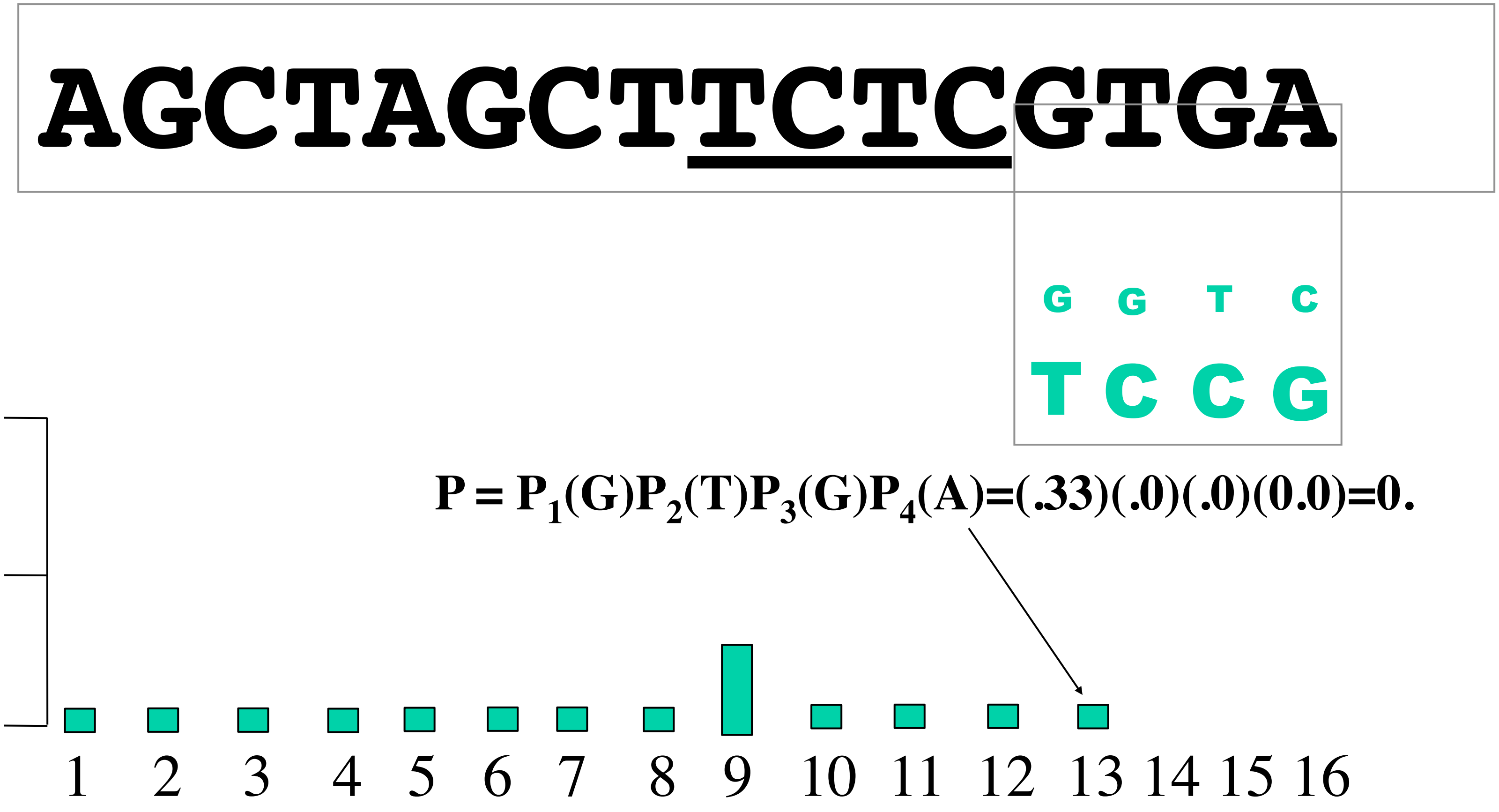


$$P = P_1(T)P_2(C)P_3(T)P_4(C)=(.67)(.67)(.33)(0.33)=0.05$$

MEME

Calculate the probability score for each position

Slide the model along the sequence to get the next score.



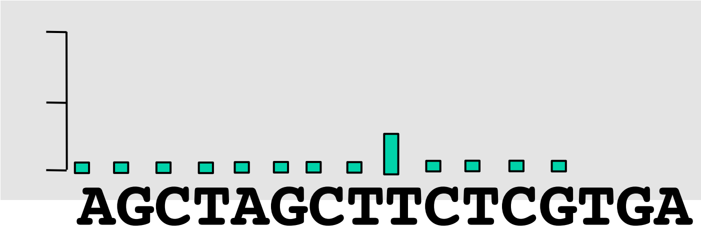
AGCTAGCT<u>TCTC</u>GTGA

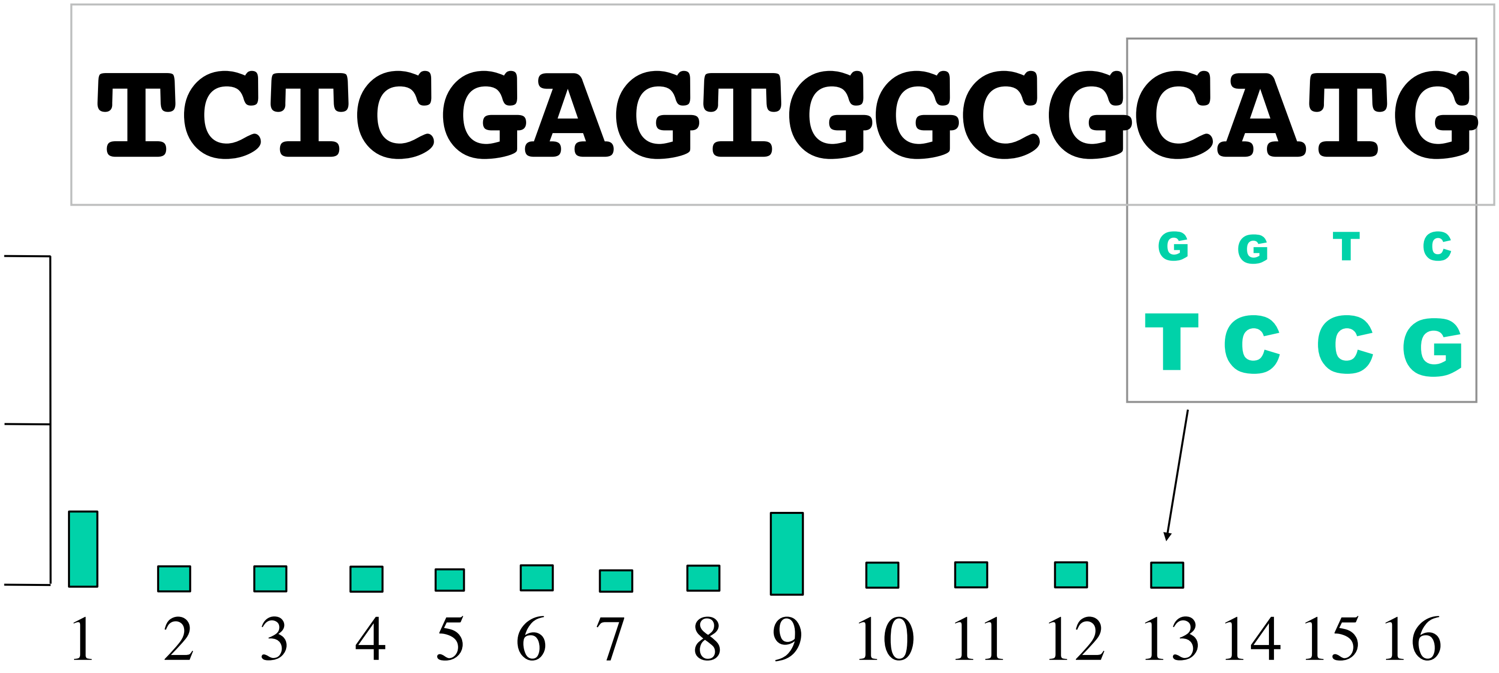
$P = P_1(G)P_2(T)P_3(G)P_4(A)=(.33)(.0)(.0)(0.0)=0.$

MEME

Calculate the probability score for each position

Do every sequence.

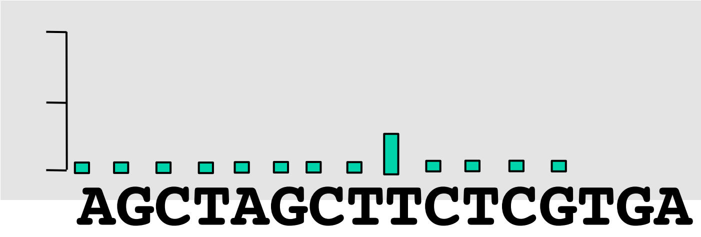


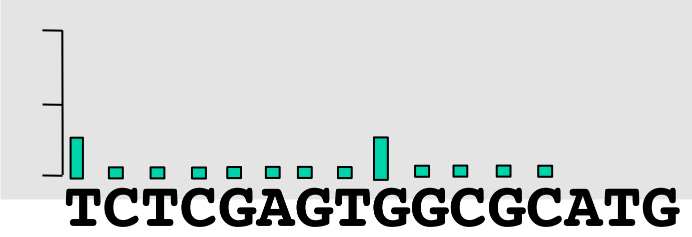


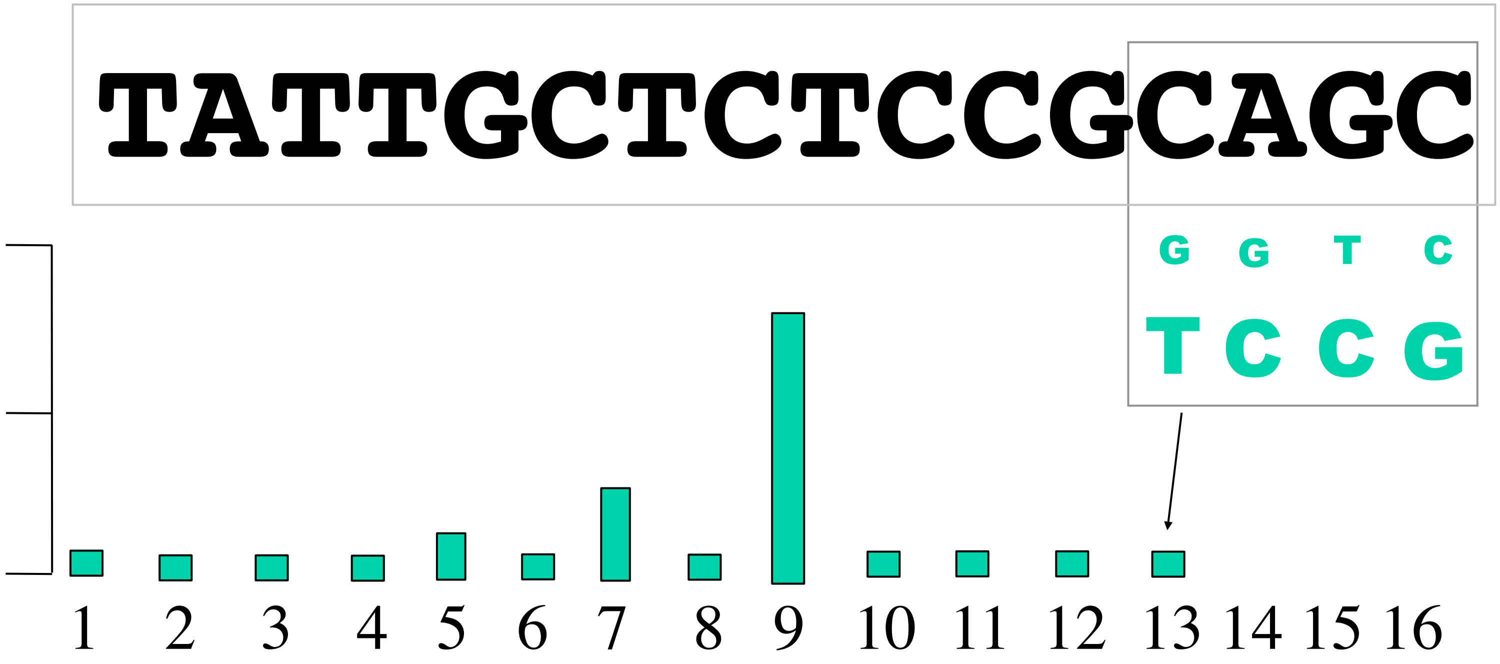
MEME

Calculate the probability score for each position

Do every sequence.

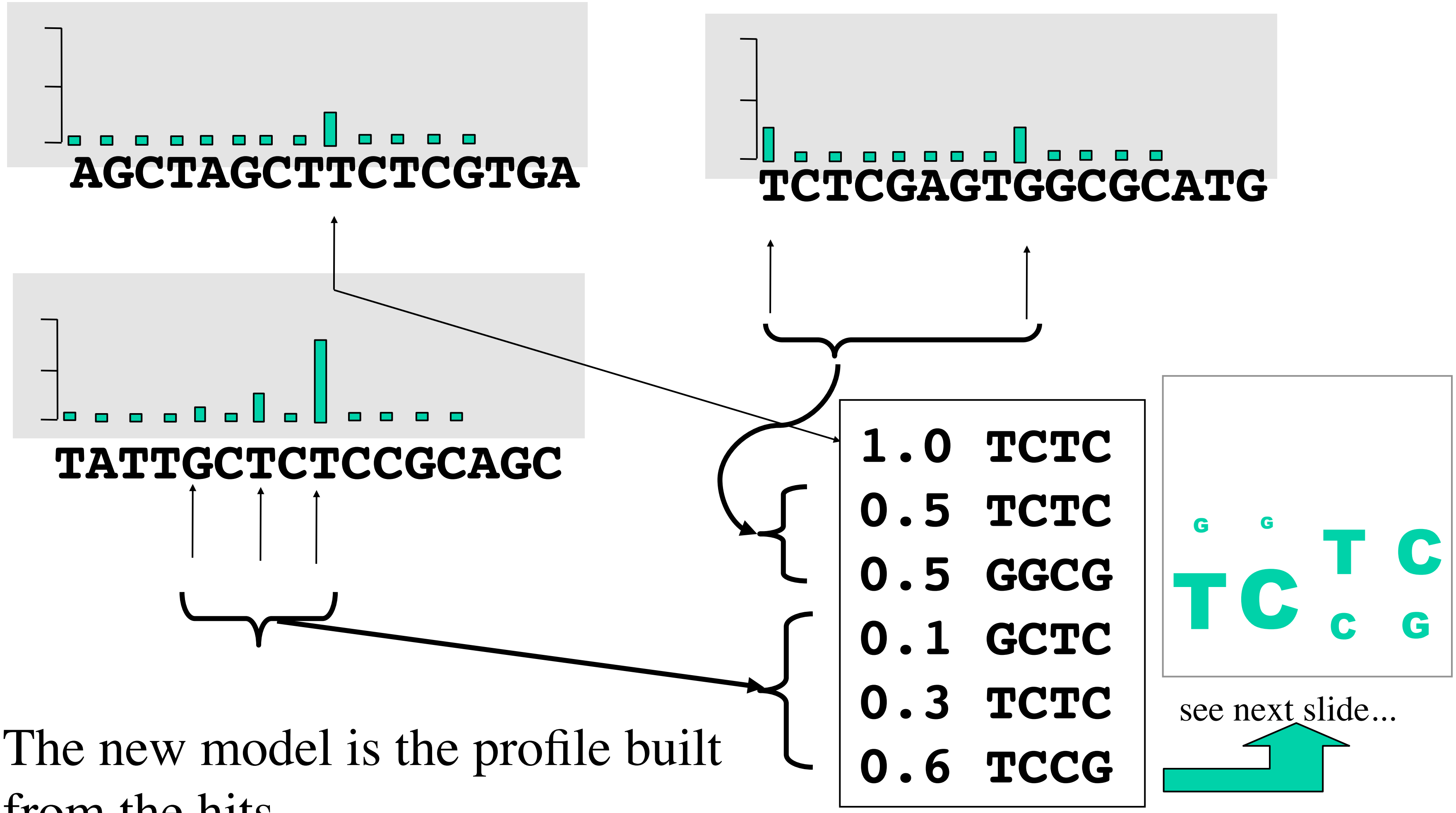






Re-Calculate the motif model

Probabilities are normalized to sum to one for each sequence, since we expect exactly one motif per sequence.



AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

TATTGCTCTCCGCAGC

1.0 TCTC
0.5 TCTC
0.5 GGCG
0.1 GCTC
0.3 TCTC
0.6 TCCG

see next slide...

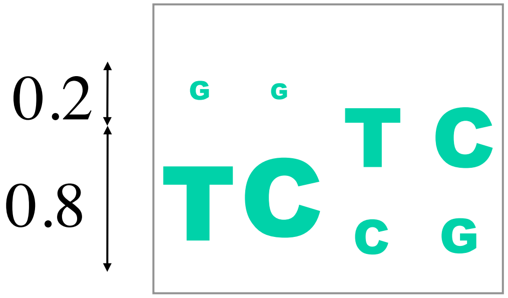
The new model is the profile built from the hits.

Recalculating the profile from the hits

1.0	TCTC
0.5	TCTC
0.5	GGCG
0.1	GCTC
0.3	TCTC
0.6	TCCG

$P_1(T)$ = the probability of T in the first position = the sum of the scores for sequences with T in the first position, normalized.

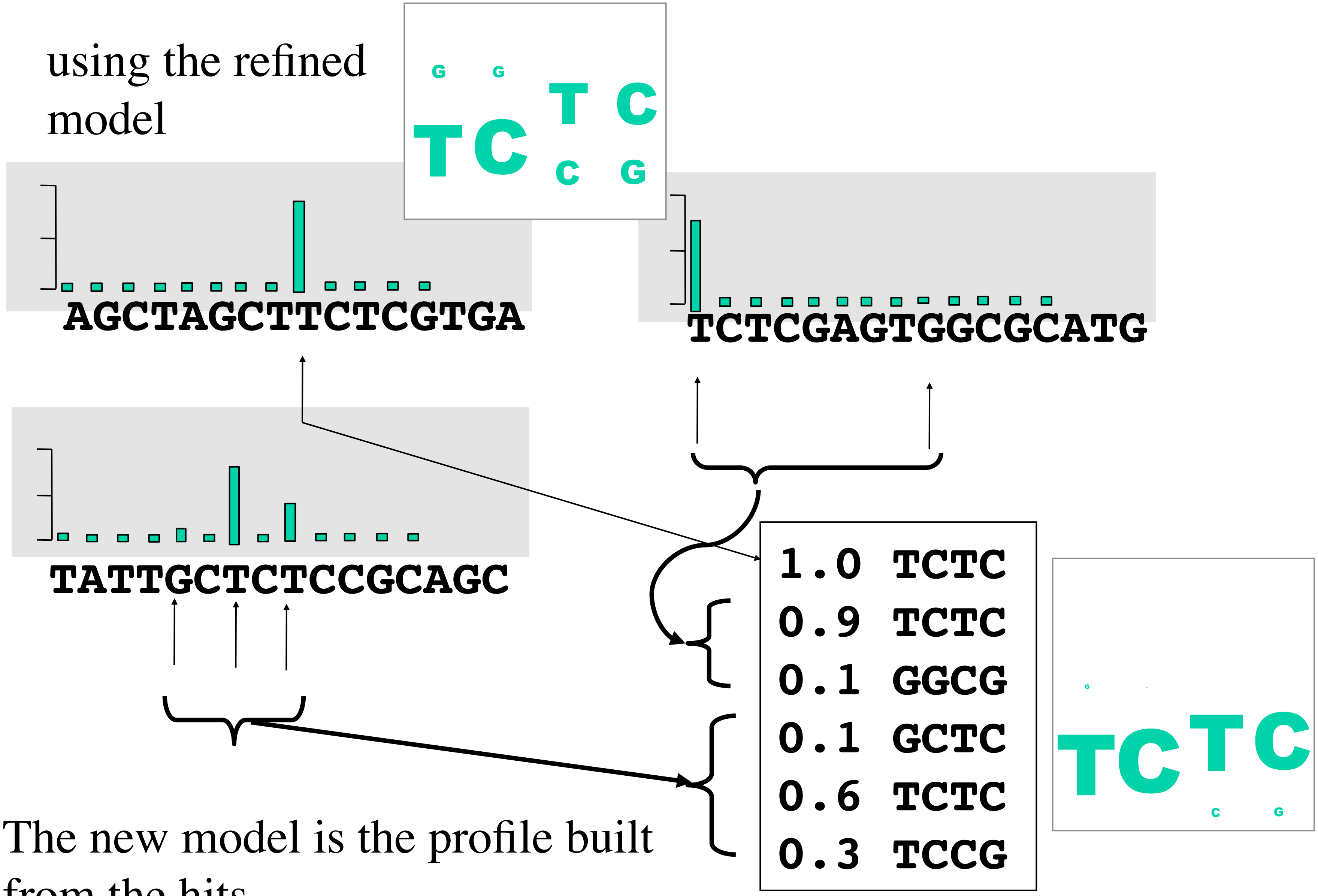
$$P_1(T) = \frac{1.0+0.5+0.3+0.6}{1.0+0.5+0.5+0.1+0.3+0.6} = 0.8$$



MEME

Do it again: Re-Calculate the probability scores

using the refined model



AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

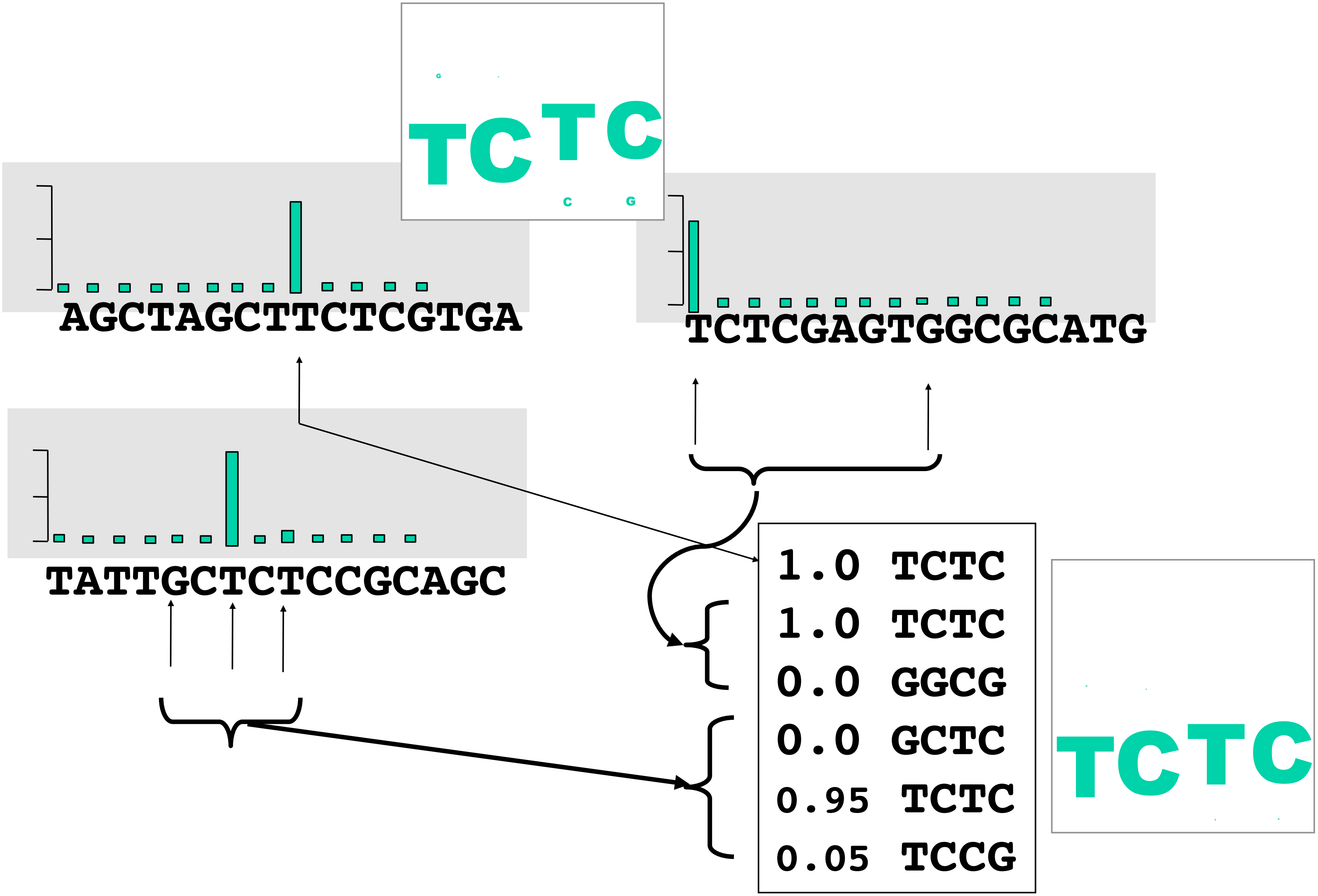
TATTGCTCTCCGCAGC

```
1.0 TCTC
0.9 TCTC
0.1 GGCG
0.1 GCTC
0.6 TCTC
0.3 TCCG
```

The new model is the profile built from the hits.

MEME

...and again, until converged.



AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

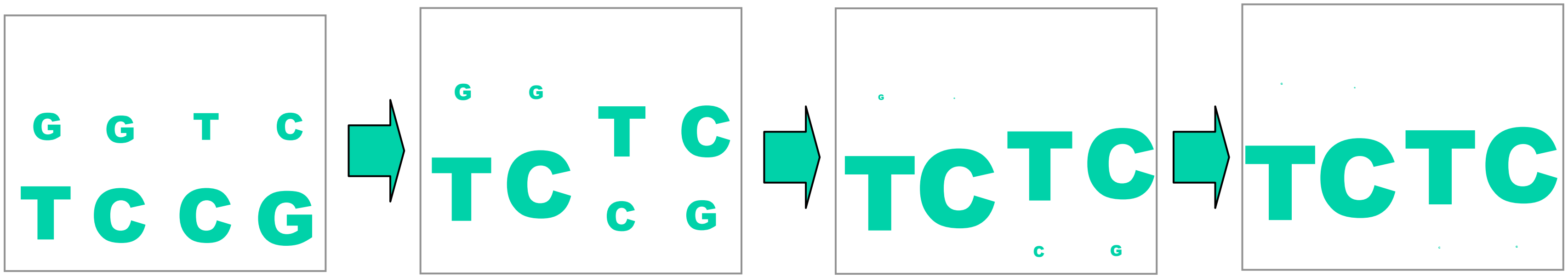
TATTGCTCTCCGCAGC

1.0	TCTC
1.0	TCTC
0.0	GGCG
0.0	GCTC
0.95	TCTC
0.05	TCCG

MEME

EM converges on the conserved pattern if the initial guess was not too far off.

A summary of the exercise:



AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

TATTGCTCTCCGCAGC

If the true motif was not one of the initial guesses, or some combination of the initial guesses, then EM would never find the true motif.

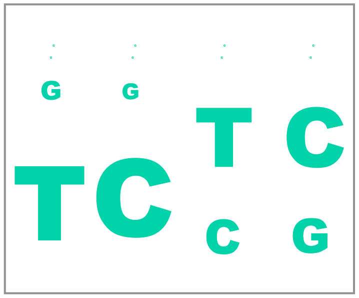
Pseudocounts, just in case

1.0	TCTC
0.5	TCTC
0.5	GGCG
0.1	GCTC
0.3	TCTC
0.6	TCCG

No A is observed in the first position, but if we set P(A) = 0, then we “rule out” a motif with A in the first position. Instead, $P_1(A)$ = a small pseudocount value / sum of the weights.

This is especially important in the initial guesses, so that the true motif is not missed.

$$P_1(T) = \frac{\varepsilon}{1.0+0.5+0.5+0.1+0.3+0.6} = 0.8$$



Pseudocounts may be decreased or removed (ε=0) in later stages.

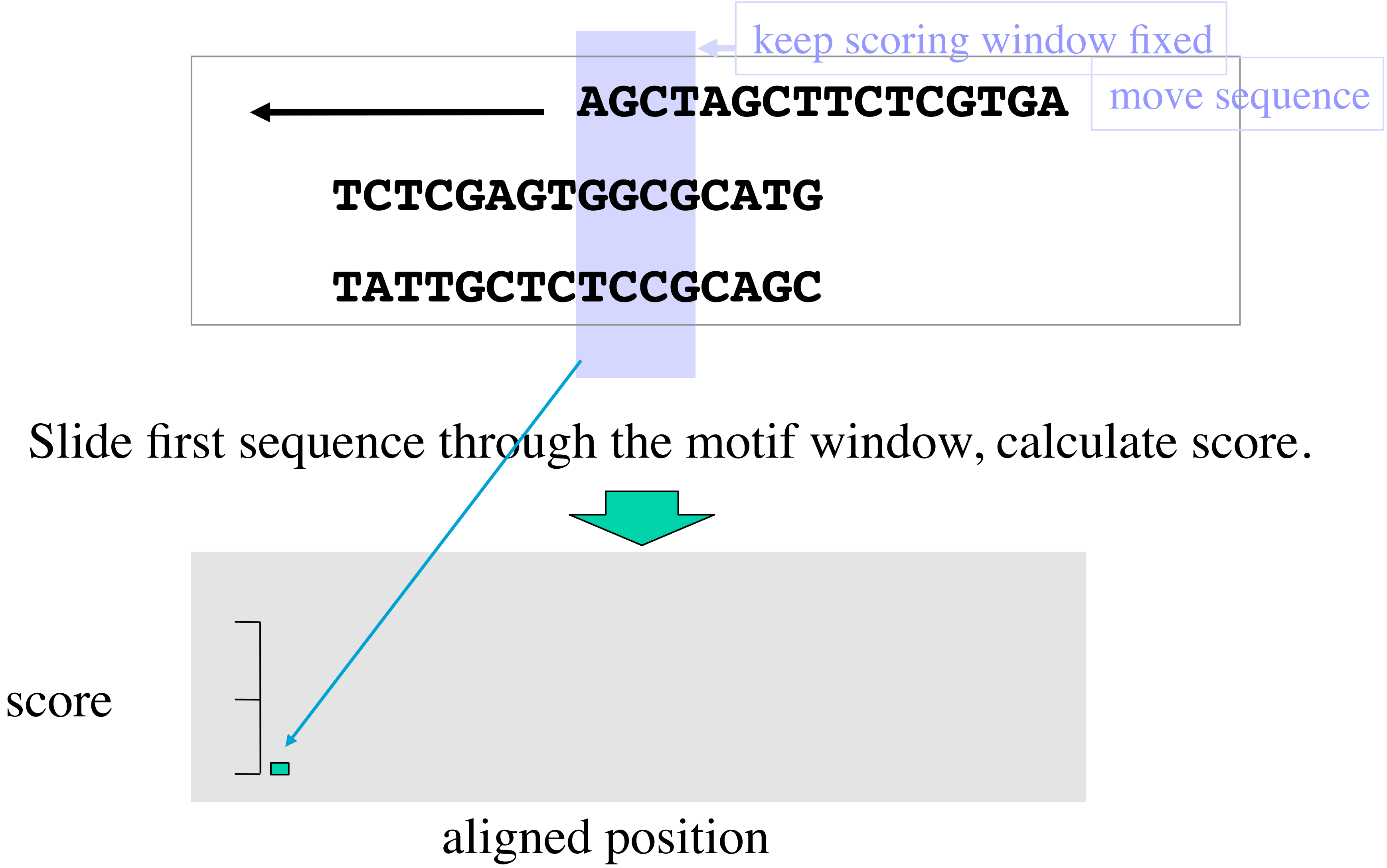
Gibbs Sampling

Stochastic version of MEME.

Radius of convergence is wider than MEME.
Doesn’t need to start with one correct guess.

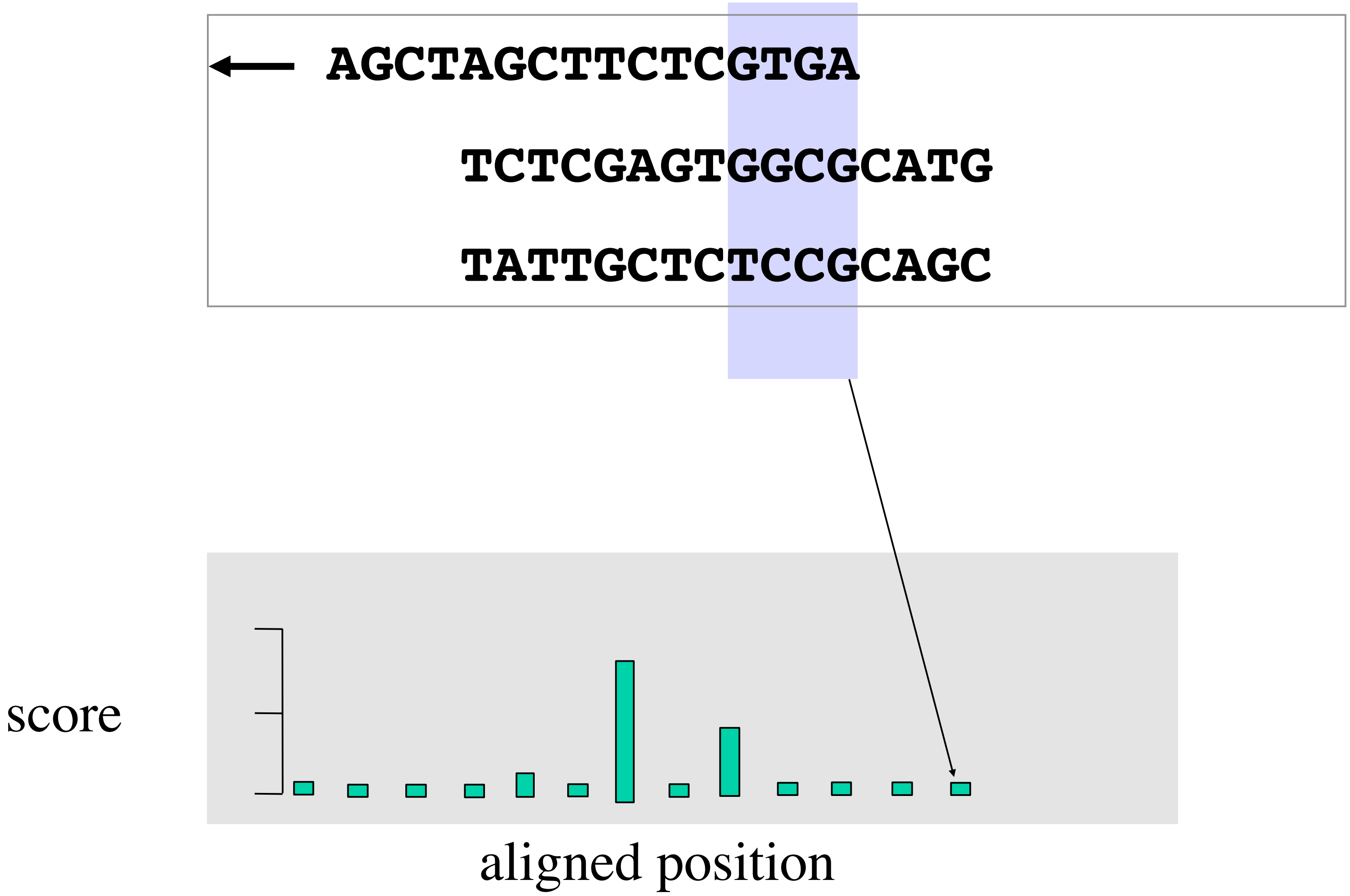
Expectation step

Start from random alignment. Select window size and position. Slide one sequence through window. Calculate scores.



Slide first sequence through the motif window, calculate score.

Expectation step

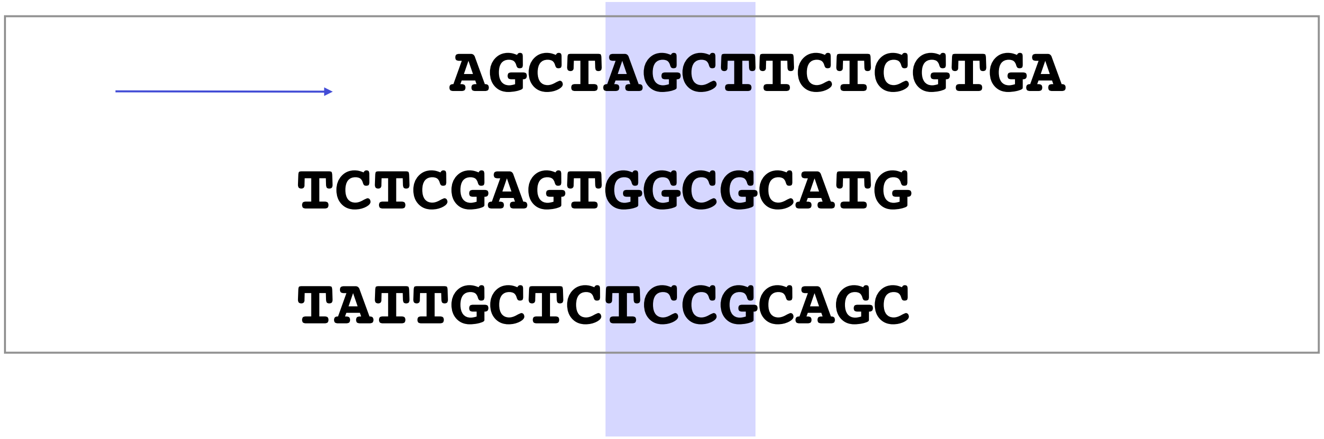


AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

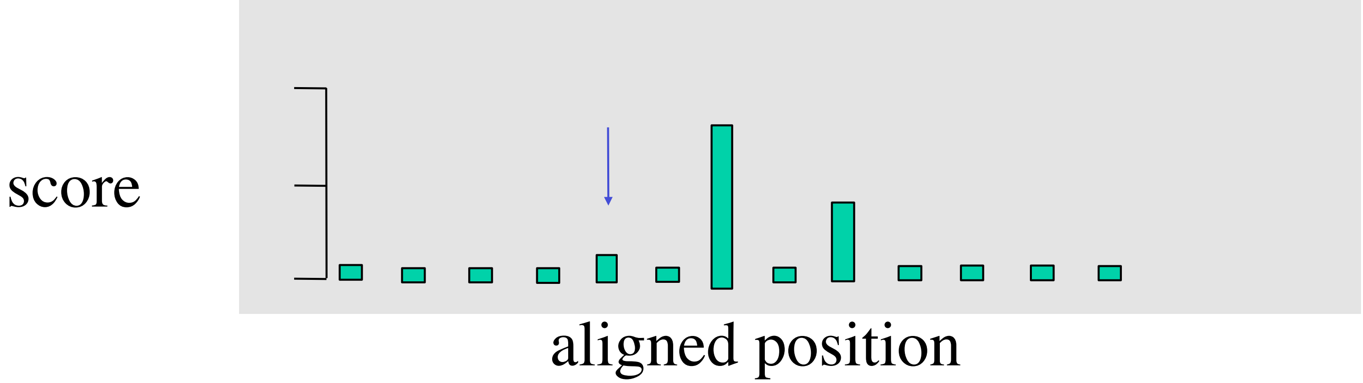
TATTGCTCTCCGCAGC

Example



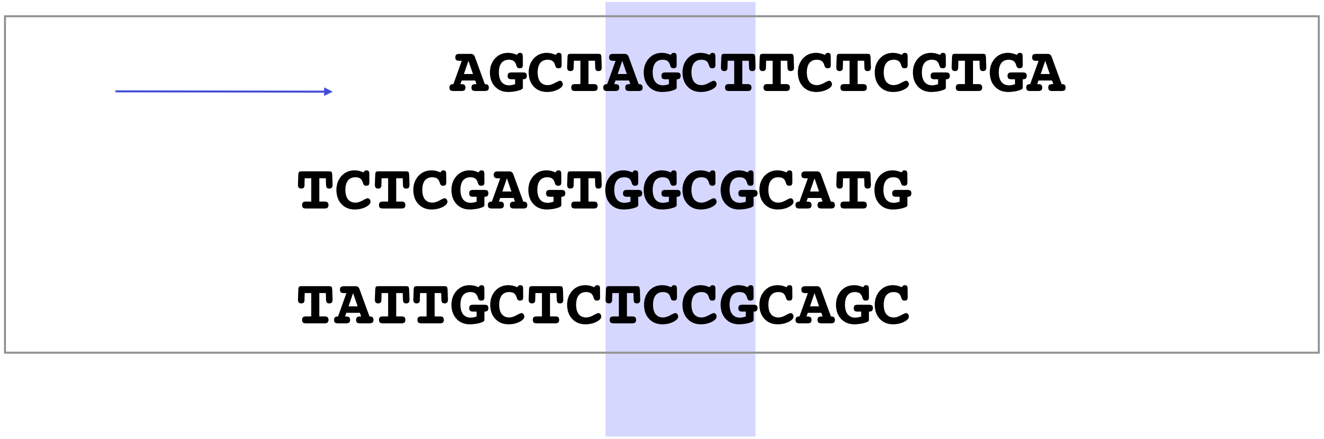
```
        AGCTAGCTTCTCGTGA
TCTCGAGTGGCGCATG
TATTGCTCTCCGCAGC
```

Select an aligned position at random from the score distribution.



Do next sequence, and so on, cycling through the sequences many times.

Example

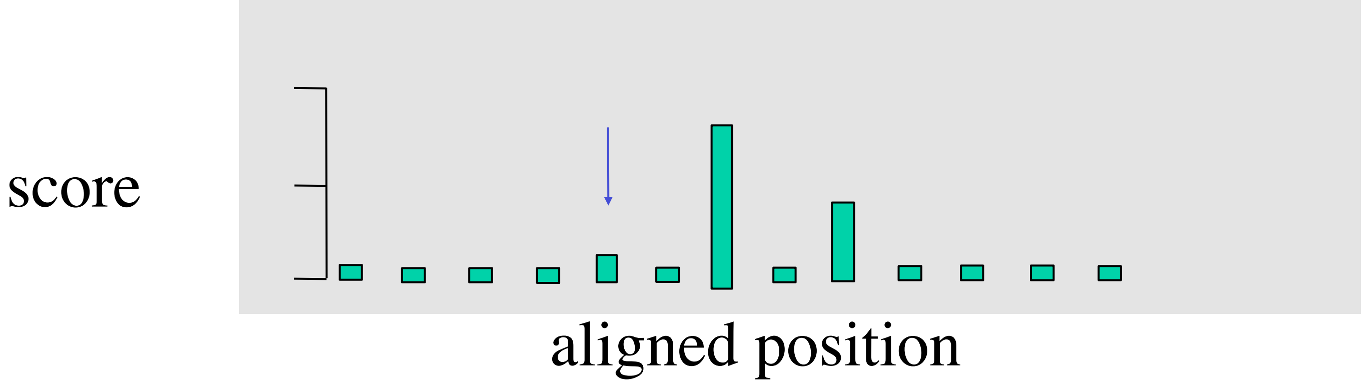


AGCTAGCTTCTCGTGA

TCTCGAGTGGCGCATG

TATTGCTCTCCGCAGC

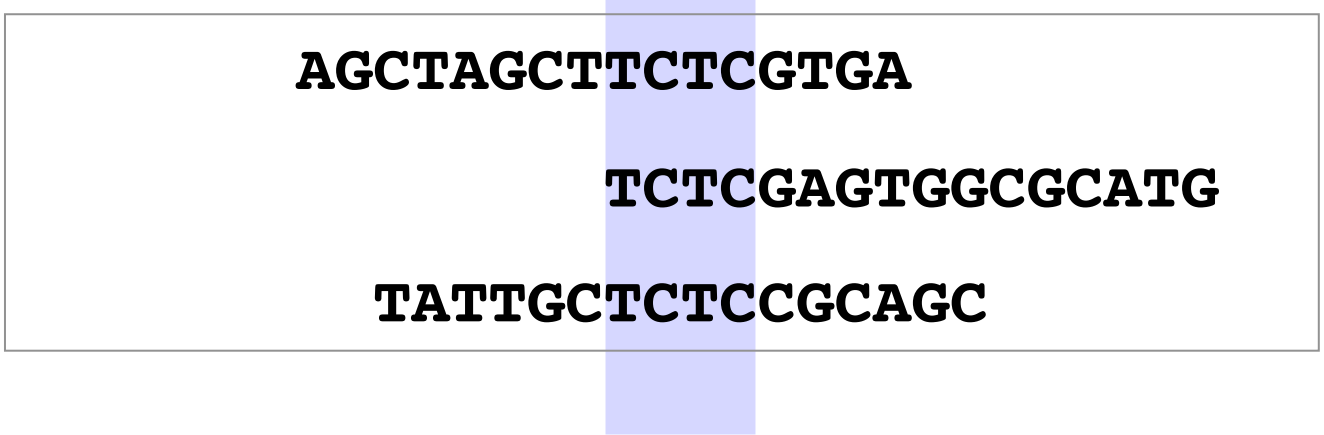
Select an aligned position at random from the score distribution.



Do next sequence, and so on, cycling through the sequences many times.

GIBBS

Convergence is when there are no more changes.



```
AGCTAGCTTCTCGTGA
        TCTCGAGTGGCGCATG
  TATTGCTCTCCGCAGC
```

Exactly one segment is aligned to the motif region at each step.

Gibbs Sampling

Stochastic version of MEME.

(1) Choose length and initial (or random) guesses of motif locations.

(2) Sum the motif profile (w/ or w/o pseudocounts/noise) from the current motif positions.

(3) Remove one sequence. Calculate probability scores for each possible motif position.

(4) Randomly choose a motif position from the probability distribution.

(5) Repeat (2)-(4) until convergence.

Radius of convergence is wider than MEME.
Doesn’t need to start with one correct guess.

What is Expectation/ Maximization ?

EM is any method that iterates between an “**expectation**” step and a “**maximization**” step. Starting with a statistical model and a set of data.

•**Expectation**
Calculate the expected values for the parameters of the model, using the current model and the data.

•**Maximization**
Replace the parameters of the model with their expected values.

MEME is an EM algorithm

K-means clustering

(1) Choose K.

(2) Randomly select K centers in the metric space.

(3) Get the distance from each center to each data point.

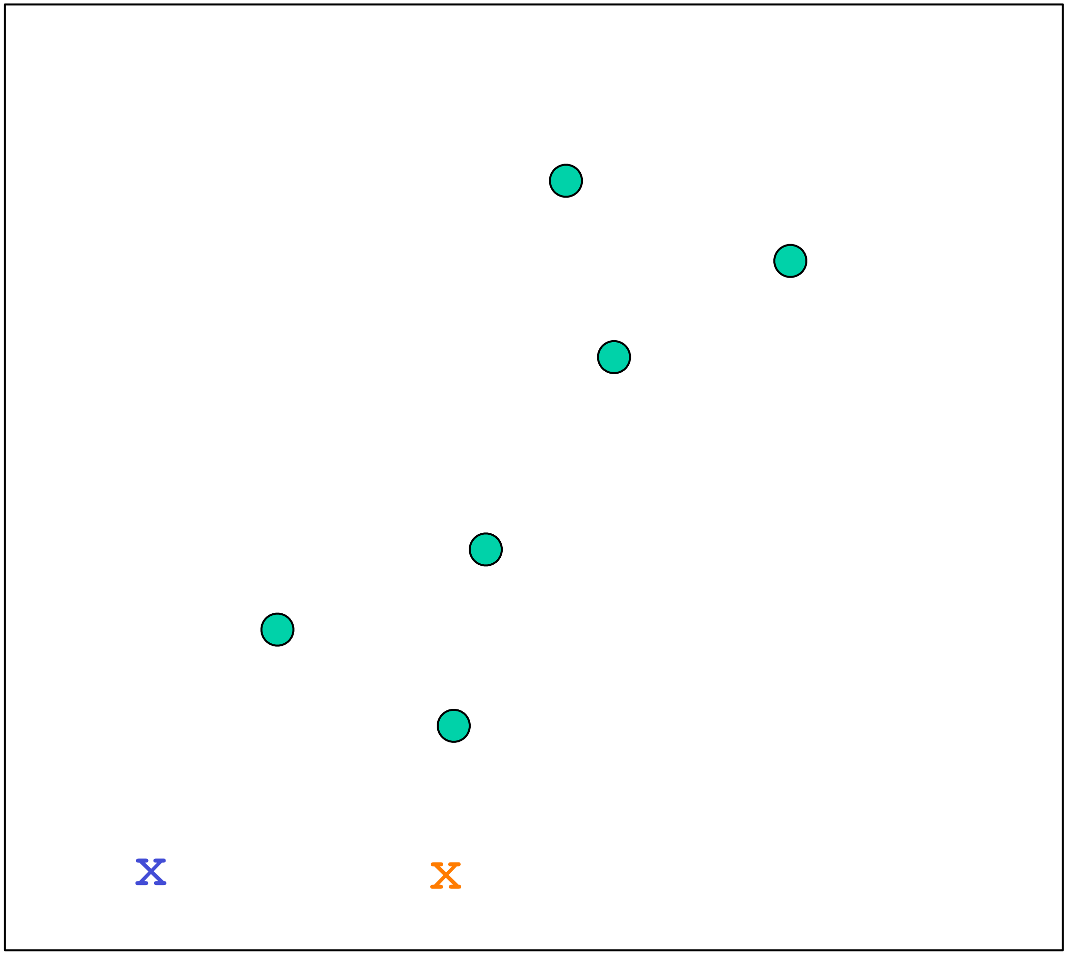
(4) Assign each data point to the nearest center.

(5) Calculate the new centers using the center-of-mass of the data points.

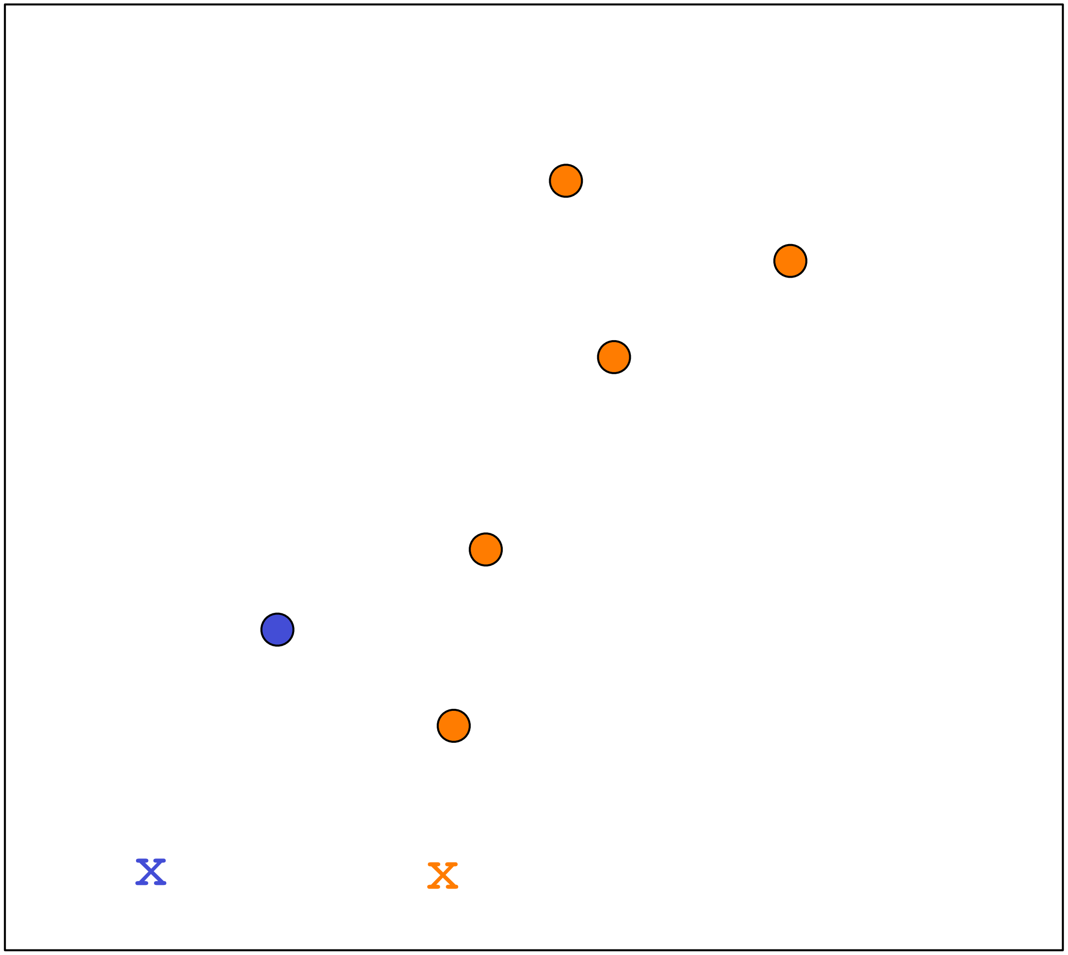
(6) *Repeat from Step 3 until converged*.

Final positions of the centers define **K** clusters of data points.

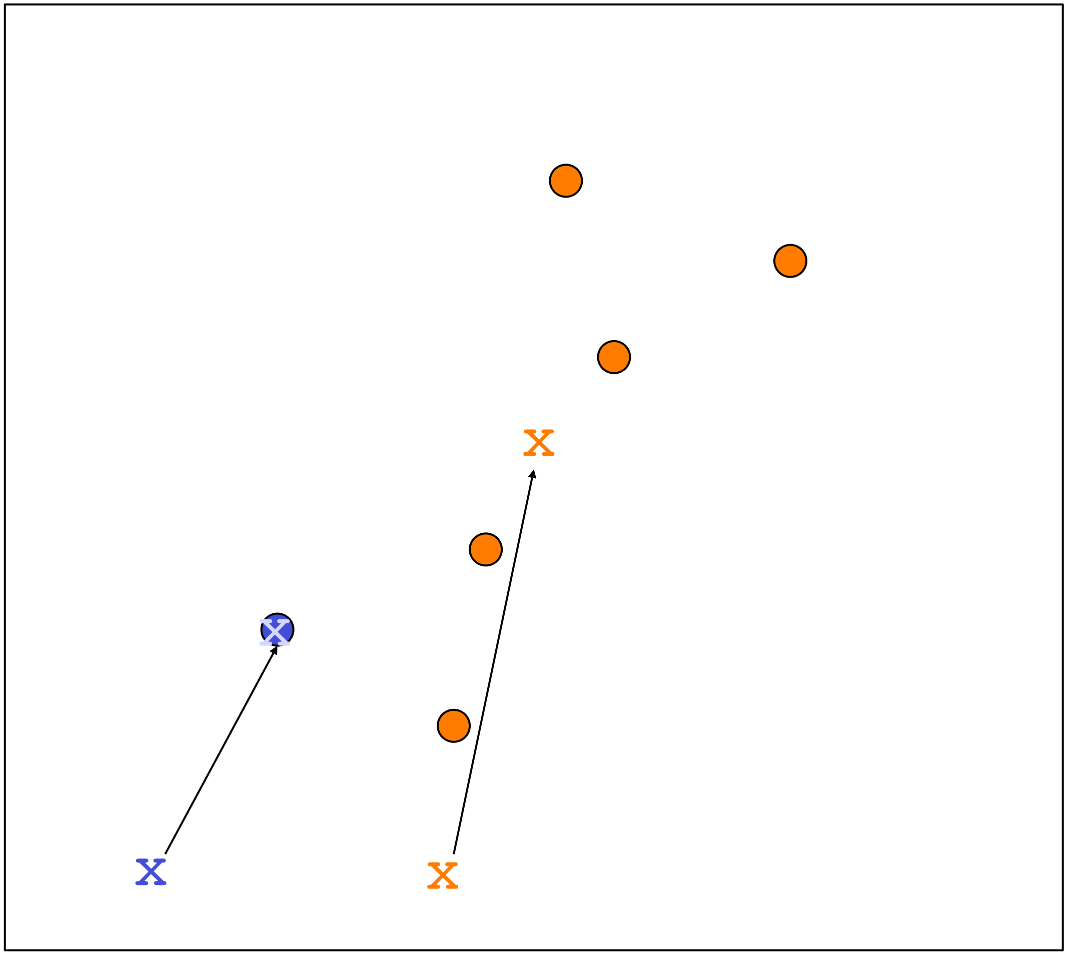
Example: K=2



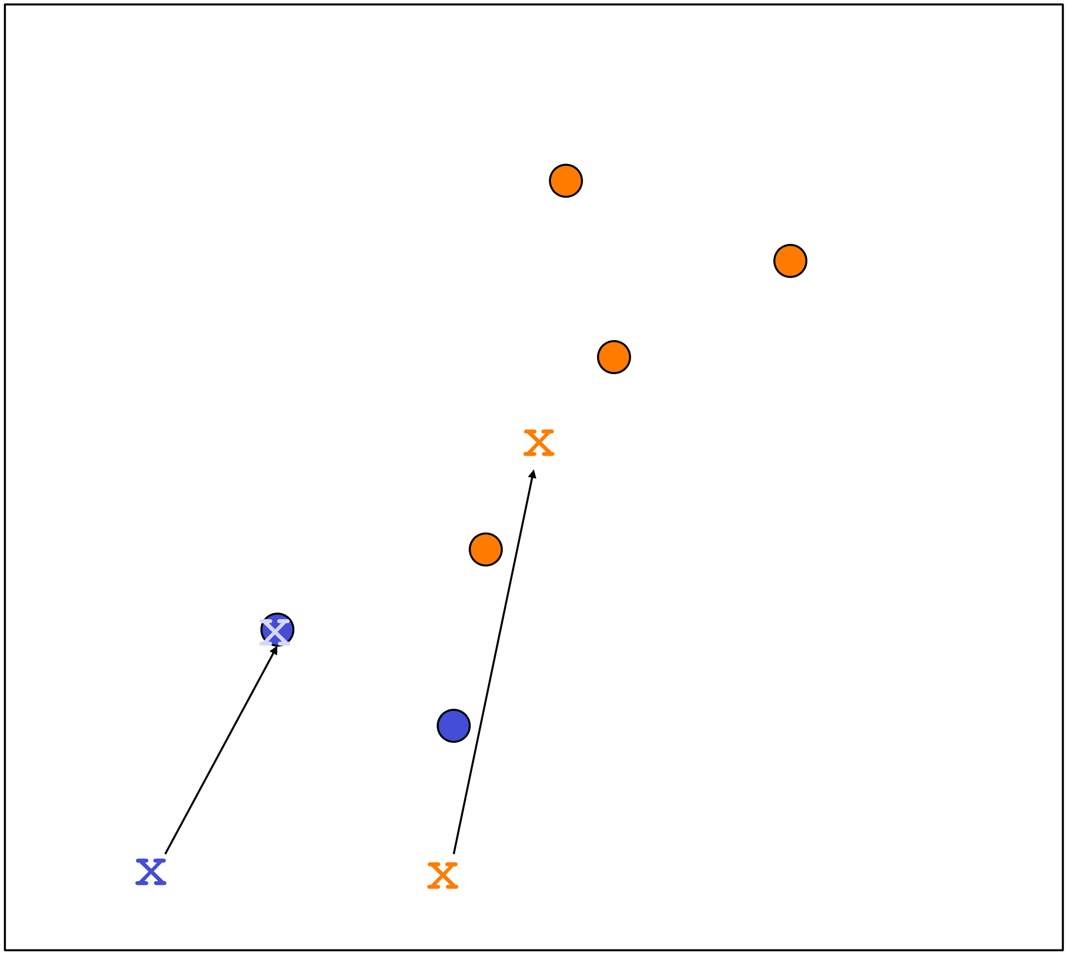
Example: K=2



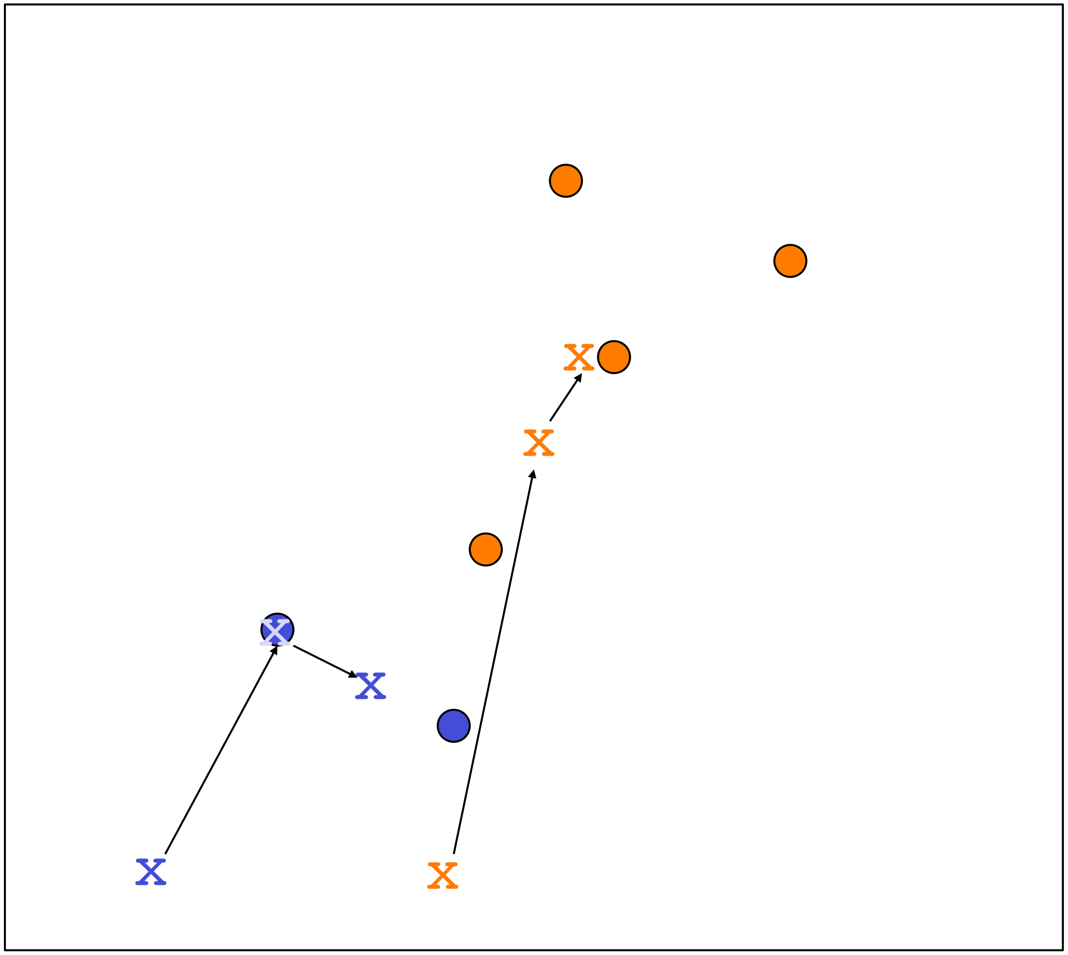
Example: K=2



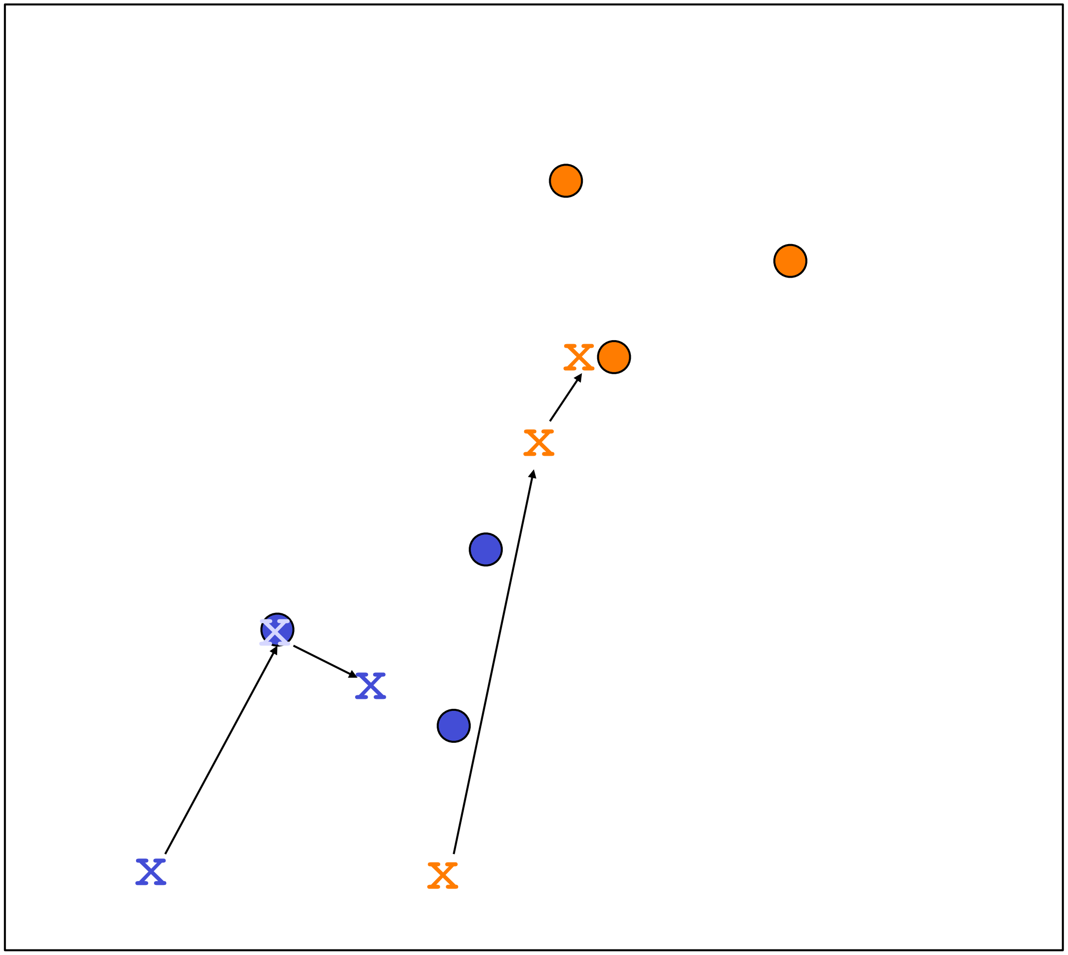
Example: K=2



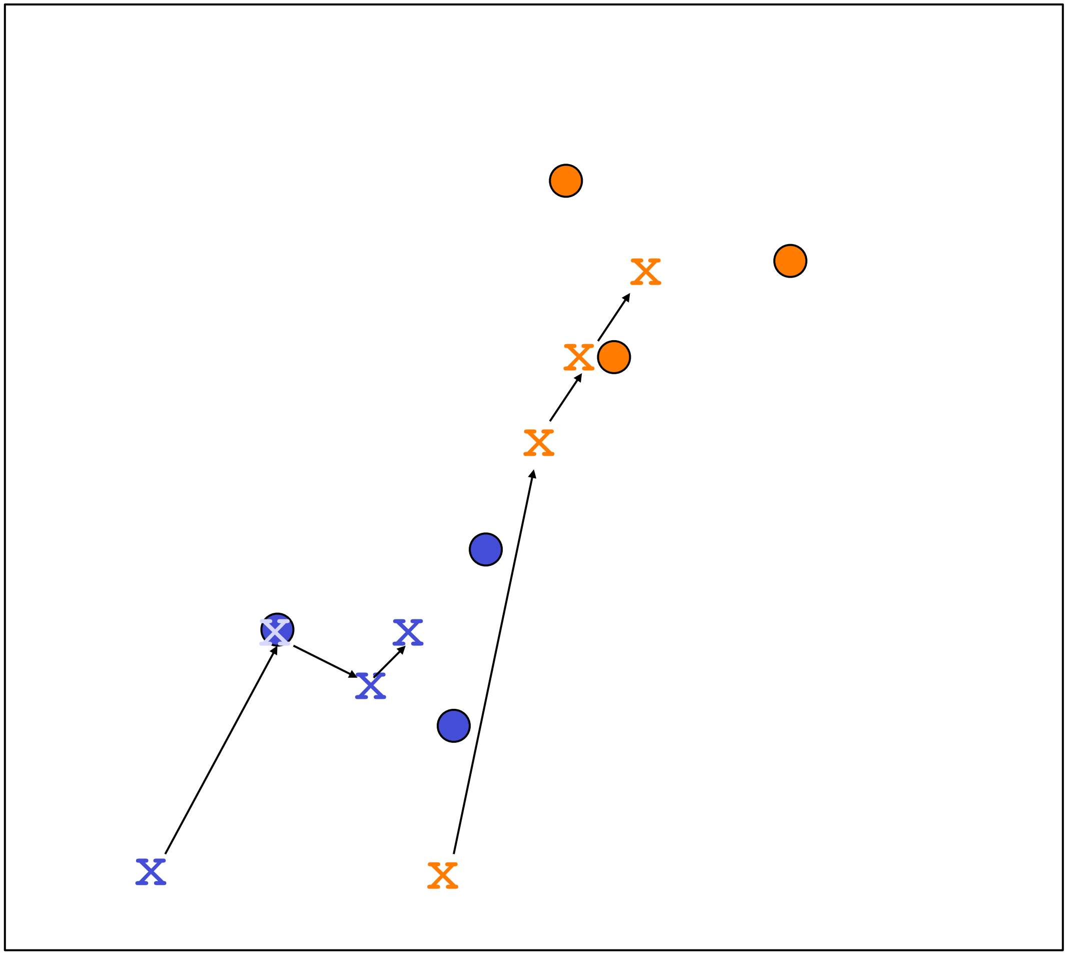
Example: K=2



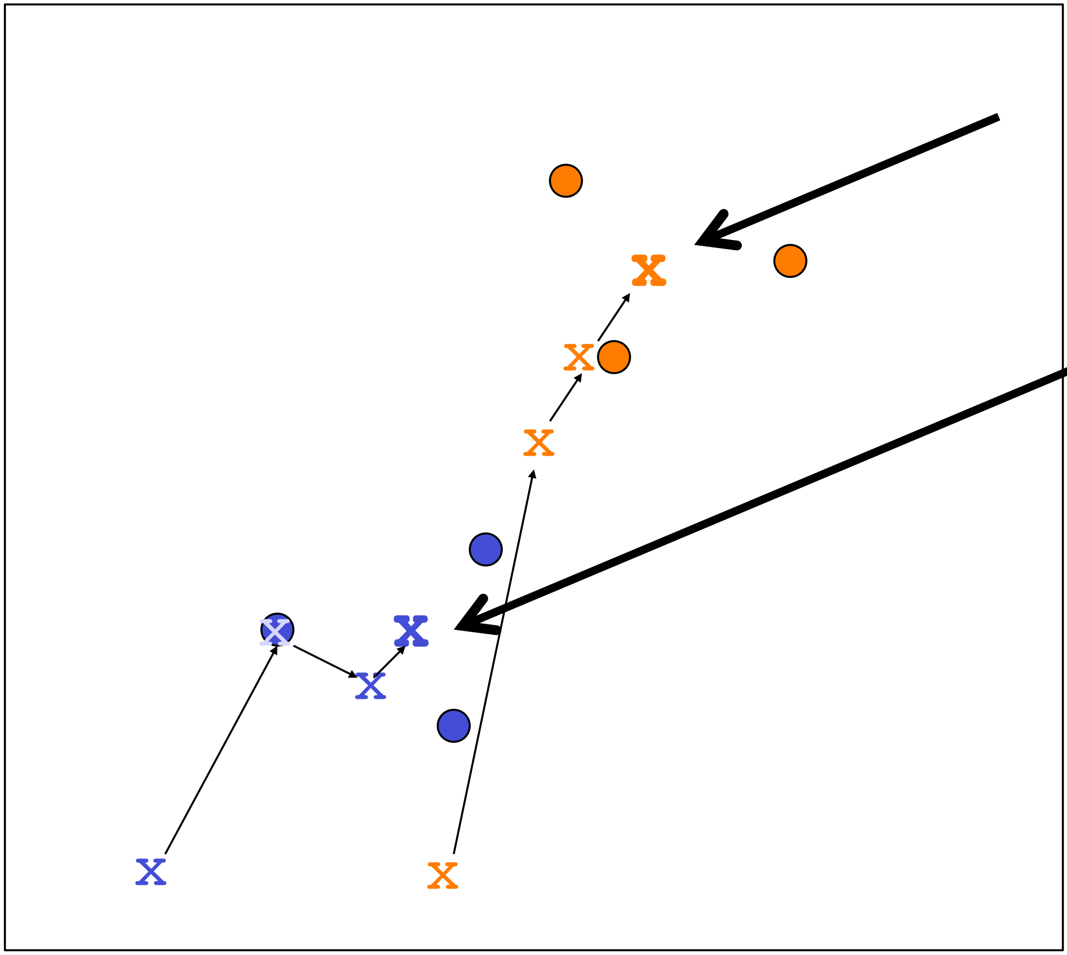
Example: K=2



Example: K=2



Example: K=2

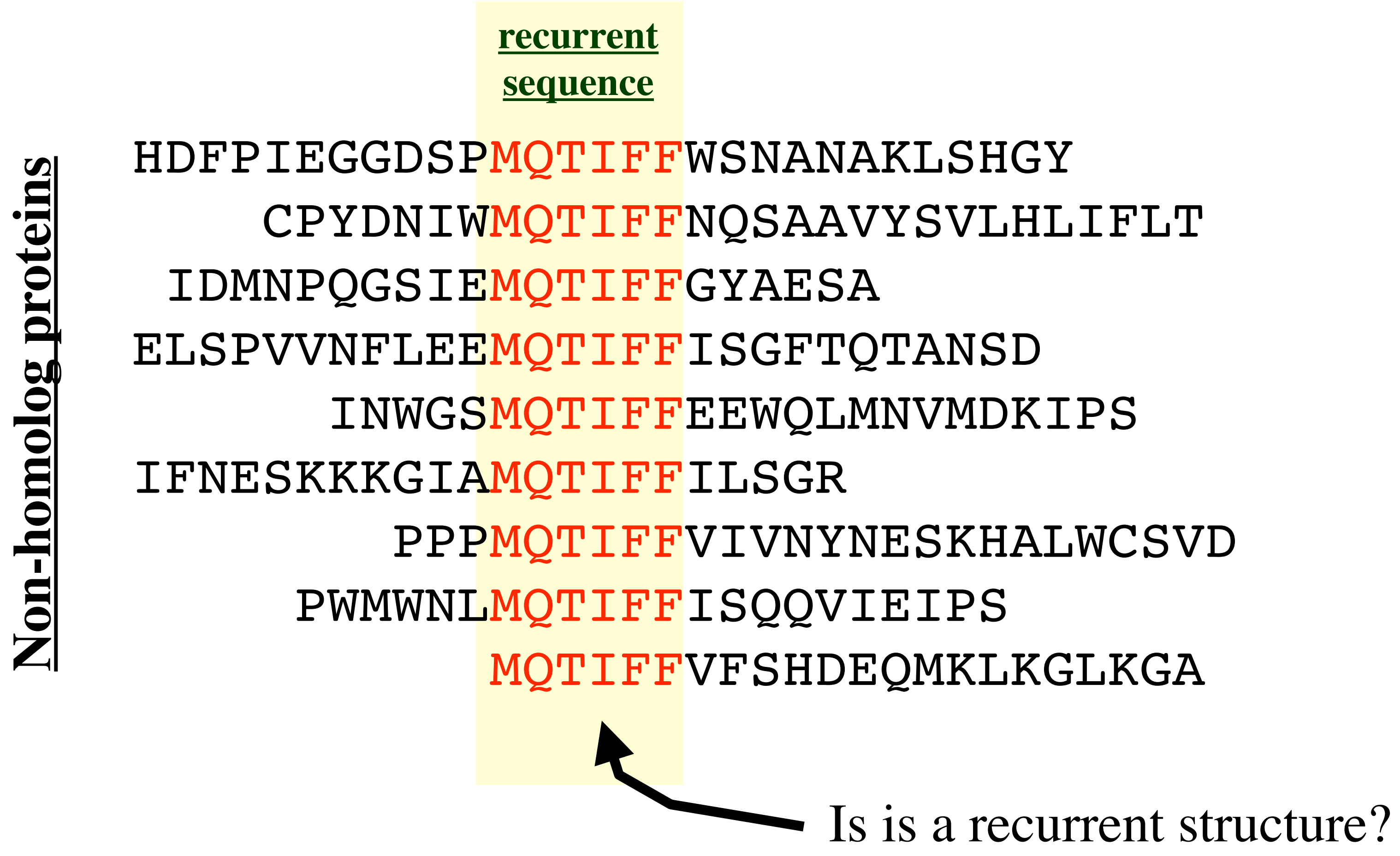


Final cluster centers

no change. Converged.

Application of K-means: I-sites motifs
Findng “words” within protein sequences

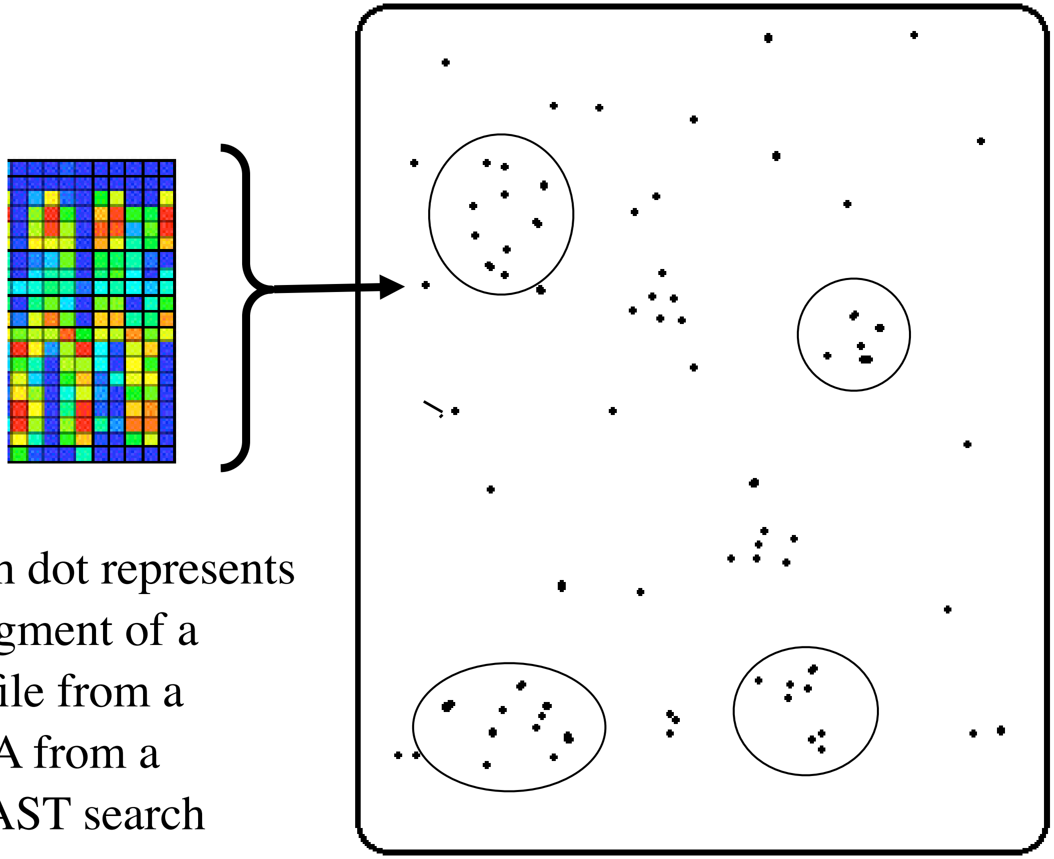
Short, recurrent sequence patterns may exist in different protein because they are required to initiate folding



```
HDFPIEGGDSPMQTIFFWSNANAKLSHGY
    CPYDNIWMQTIFFNQSAAVYSVLHLIFLT
 IDMNPQGSIEMQTIFFGYAESA
ELSPVVNFLEEMQTIFFISGFTQTANSD
      INWGSMQTIFFEEWQLMNVMDKIPS
IFNESKKKGIAMQTIFFILSGR
        PPPMQTIFFVIVNYNESKHALWCSVD
     PWMWNLMQTIFFISQQVIEIPS
           MQTIFFVFSHDEQMKLKGLKGA
```

Is is a recurrent structure?

Clustering protein sequence profiles (Bystroff&Baker, 1998)



Each dot represents a segment of a profile from a MSA from a BLAST search

distance/similarity metrics for clustering **profiles**.

(1) Manhattan, or City-Block metric
(distance metric)

$$D(p,q) = \sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} \left| P\left(p_{ij}\right) - P\left(q_{ij}\right) \right|$$

(2) Entropy (similarity metric)
__not symmetrical__!

$$S(p,q) = \sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} p_{ij} \log\left(q_{ij}\right)$$

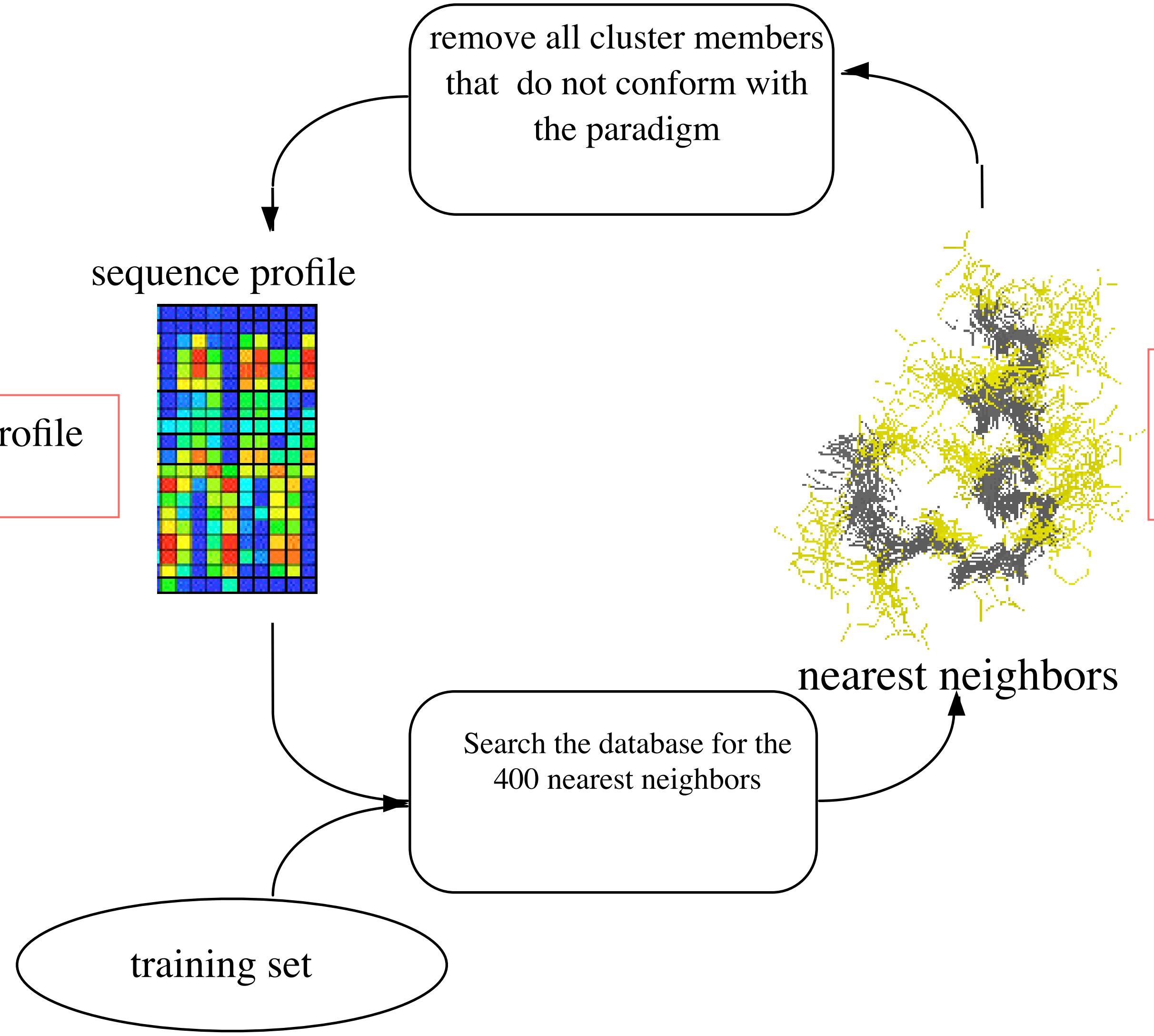
(3) Correlation (similarity metric)

$$S(p,q) = \frac{\displaystyle\sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} \left(p_{ij} - \langle p \rangle\right)\left(q_{ij} - \langle q \rangle\right)}{\sqrt{\displaystyle\sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} \left(p_{ij} - \langle p \rangle\right)^2 \sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} \left(q_{ij} - \langle q \rangle\right)^2}} = \frac{\displaystyle\sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} \left(p_{ij} - \langle p \rangle\right)\left(q_{ij} - \langle q \rangle\right)}{\sigma_p \sigma_q}$$

(4) Dpq (similarity metric)

$$D(p,q) = \sum_{\substack{\text{positions}\\ j}} \sum_{\substack{\text{amîno}\\ \text{acids}\\ i}} LLR\left(p_{ij}\right) LLR\left(q_{ij}\right)$$

Supervised learning is like co-clustering



remove all cluster members that do not conform with the paradigm

sequence profile

We want this profile to predict...

…as long as it is consistent with this structure.

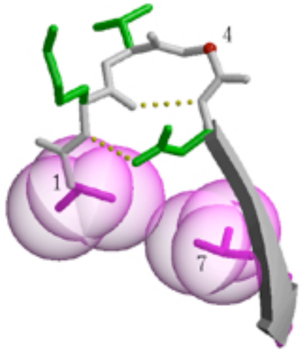
nearest neighbors

Search the database for the 400 nearest neighbors

training set

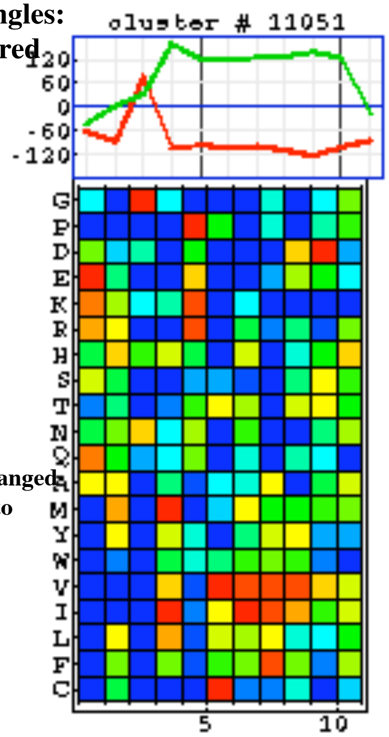
Supervised learning finds predictive correlations between two spaces (sequence and structure)

I-sites motifs

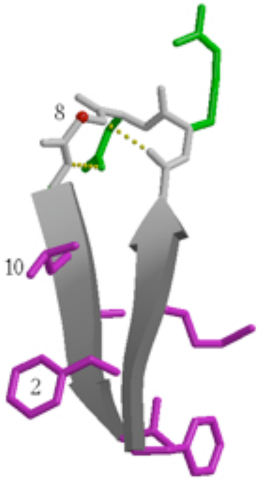


diverging type-2 turn

Backbone angles:
ψ=green, φ=red

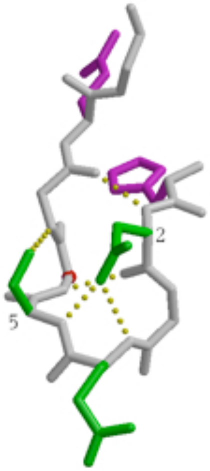


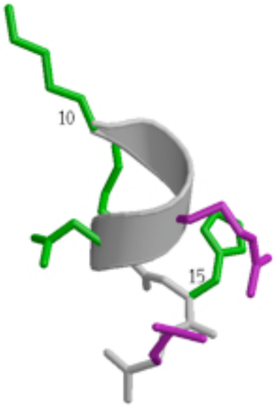
Amino acids arranged from non-polar to polar



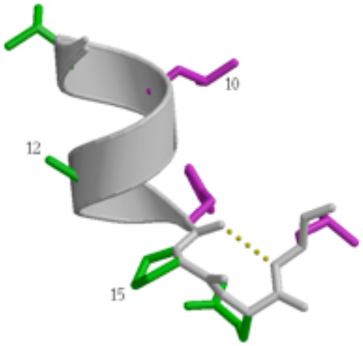
Type-I hairpin

Serine hairpin

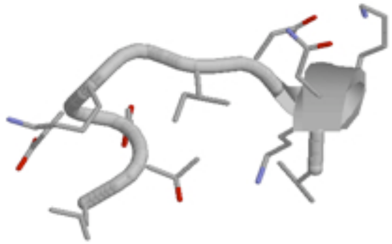




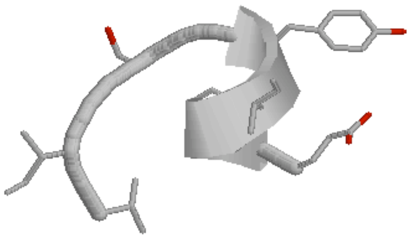
Frayed helix



Proline helix C-cap



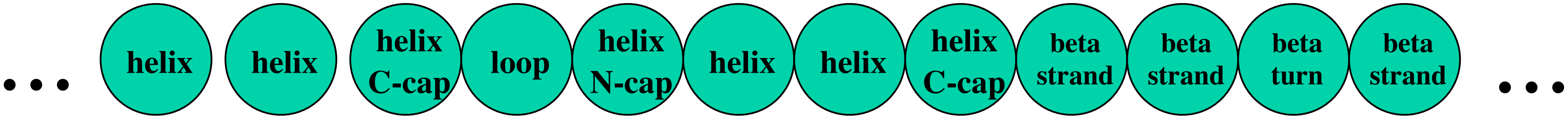
alpha-alpha corner



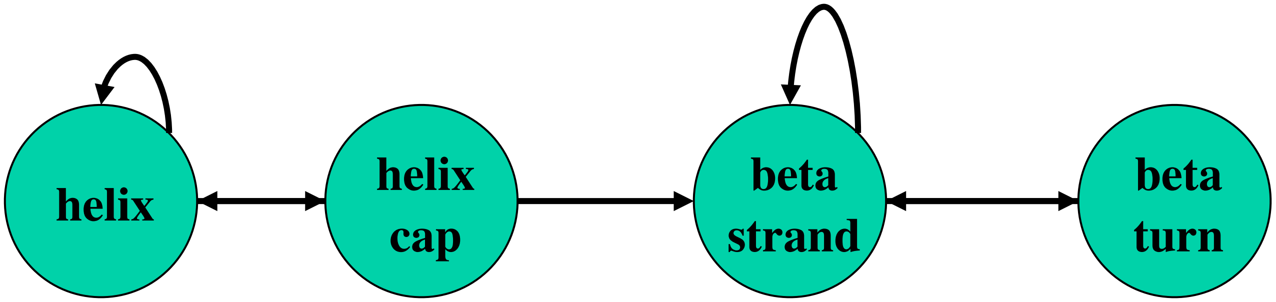
glycine helix N-cap

I-sites ---> HMM

I-sites are arranged in predictable non-random order in proteins:

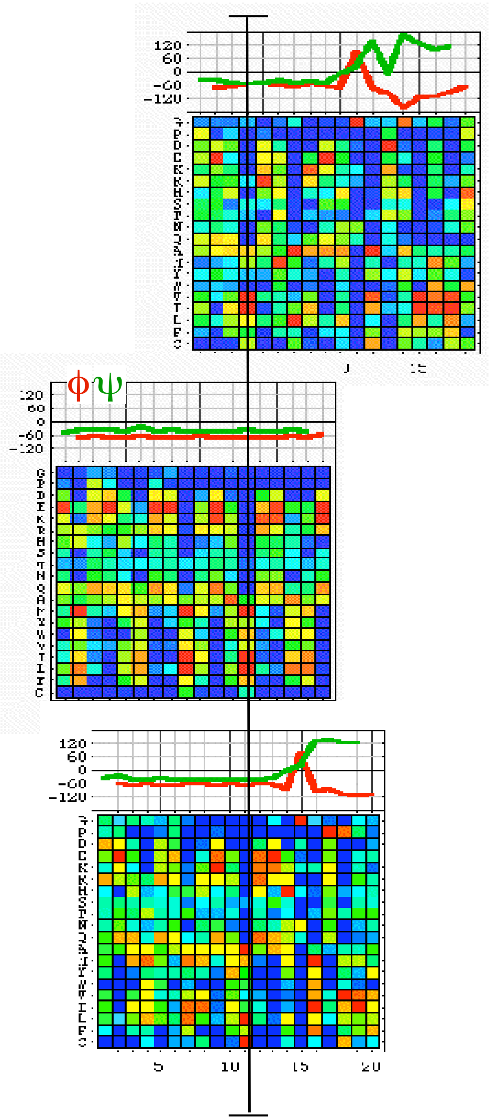


...therefore they can be modeled as a HMM.

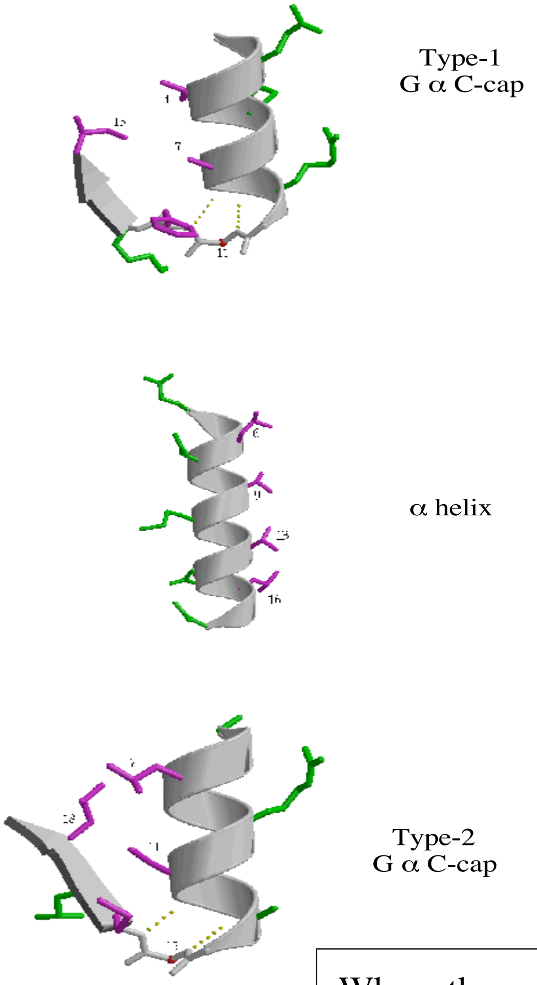


State-state transitions are defined wherever I-sites have overlaps.

aligned profiles

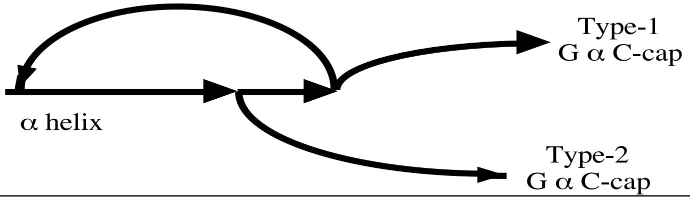


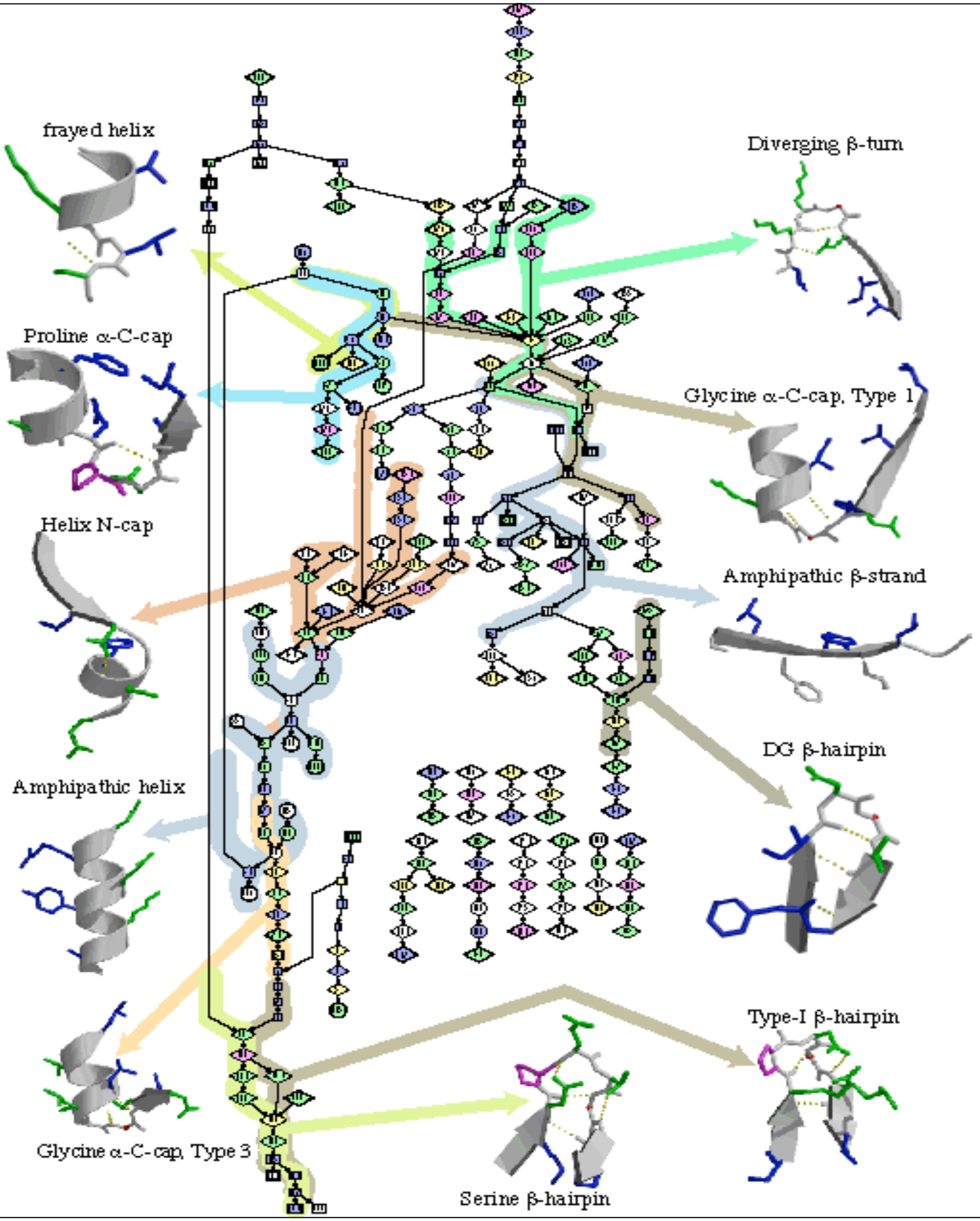
aligned structures



Where the motifs align, we call each positions a state. Where they stop aligning, we split the state path.

state topology:





I-sites HMM
=
HMMSTR!

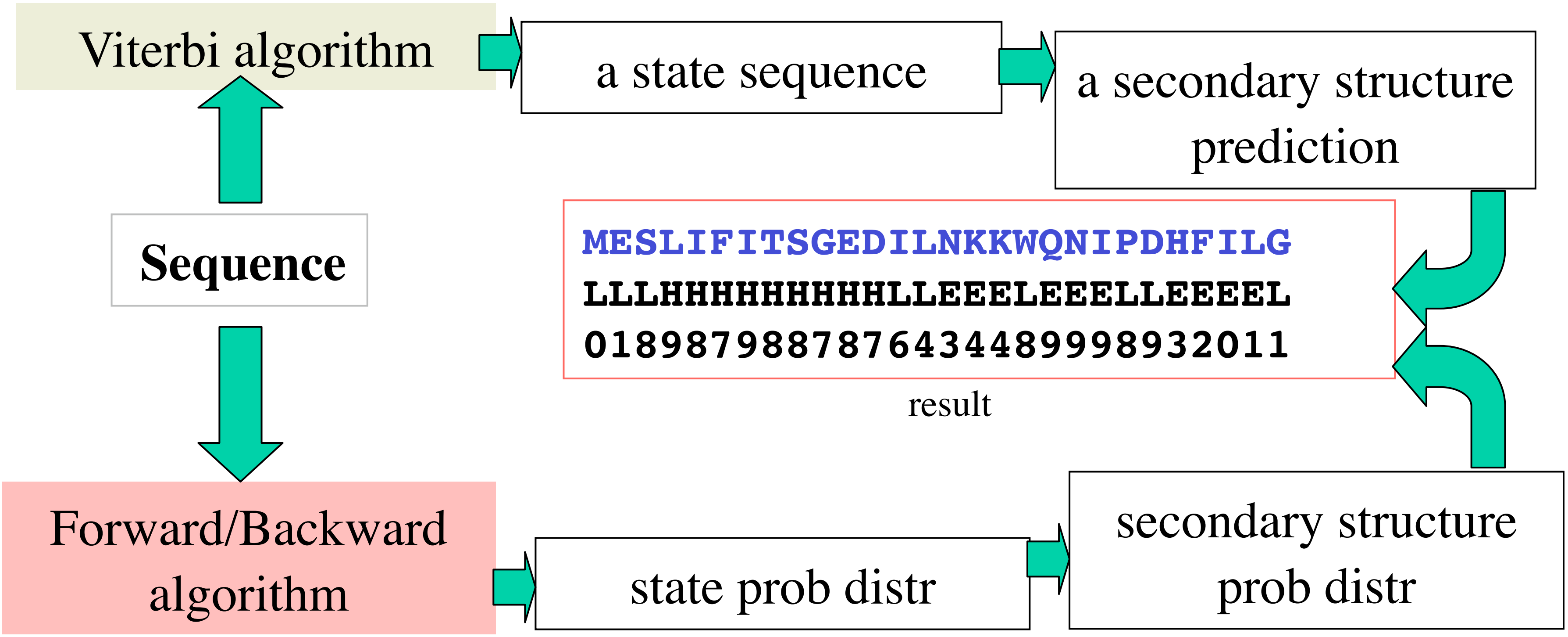
Hidden Markov Model for local protein STRucture

HMM of linked I-sites motifs. Each node is one amino acid.

Size of HMM:
282 nodes
317 transitions

(Bystroff et al., JMB 2000)

HMMSTR server



Viterbi algorithm

a state sequence

a secondary structure prediction

Sequence

```
MESLIFITSGEDILNKKWQNIPDHFILG
LLLHHHHHHHHLLEEELEEELLEEEEL
0189879887876434489998932011
```

result

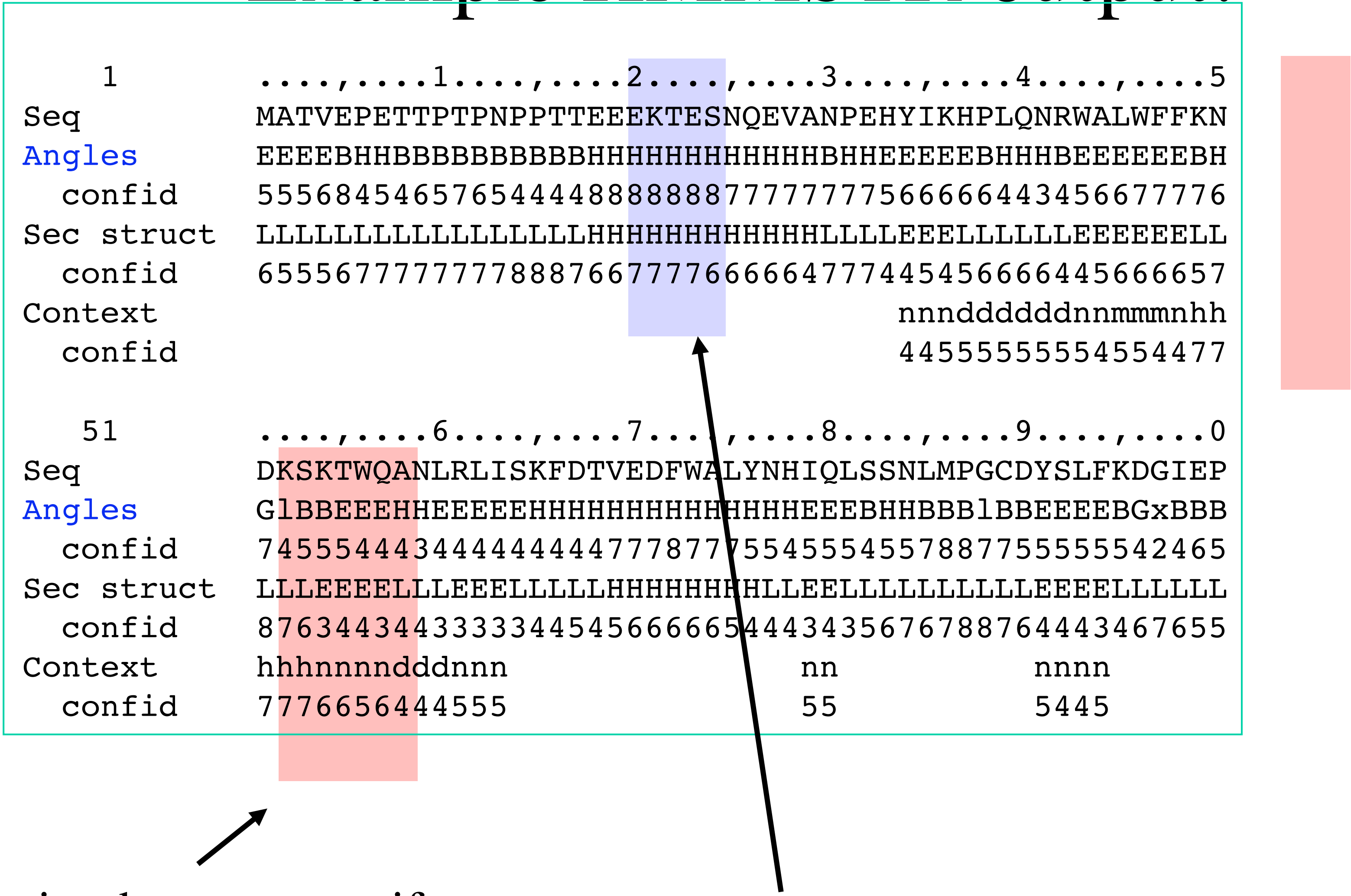
Forward/Backward algorithm

state prob distr

secondary structure prob distr

www.bioinfo.rpi.edu/bystrc/hmmstr/server.php

Example HMMSTR output.



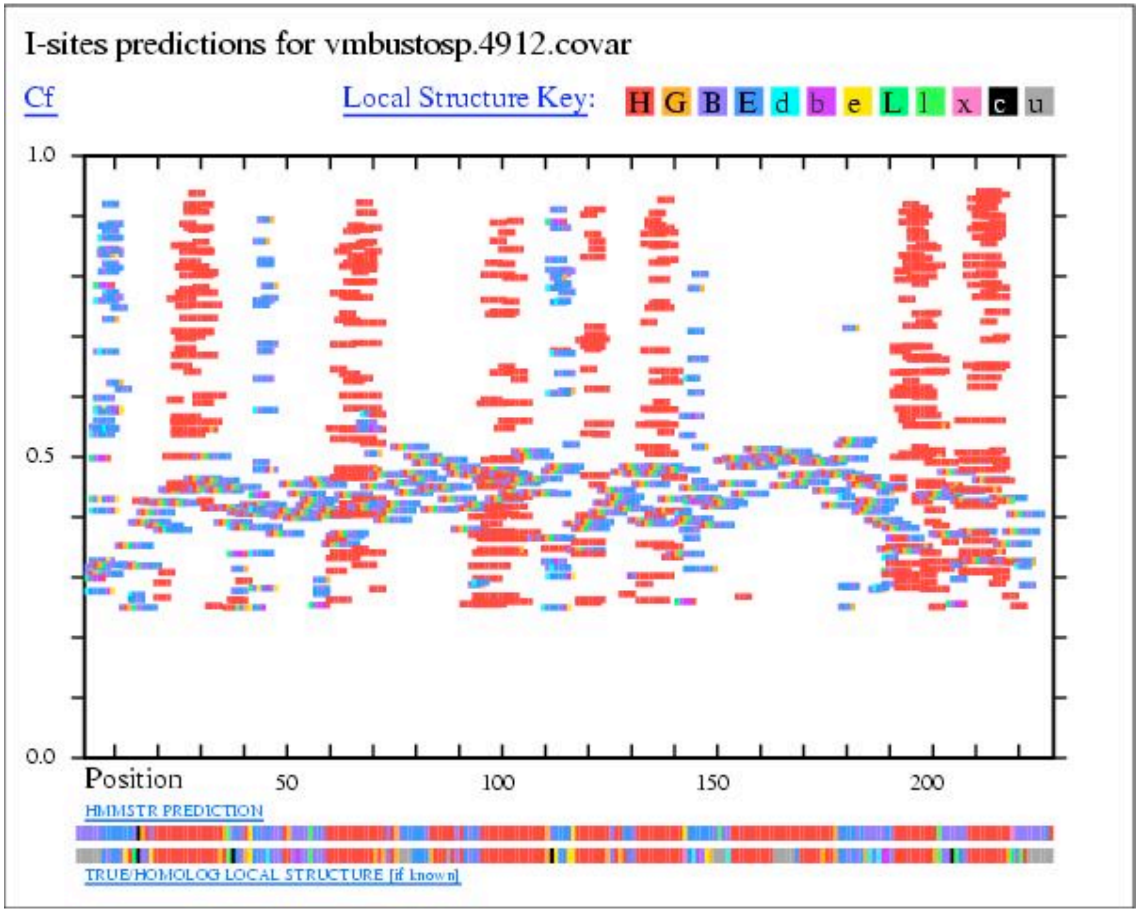
```
    1       ....,....1....,....2....,....3....,....4....,....5
Seq         MATVEPETTPTPNPPTTEEEKTESNQEVANPEHYIKHPLQNRWALWFFKN
Angles      EEEEBHHBBBBBBBBBBHHHHHHHHHHHBHHEEEEEBHHHBEEEEEEEBH
  confid    55568454657654444888888877777777566666443456677776
Sec struct  LLLLLLLLLLLLLLLLLHHHHHHHHHHHLLLLEEELLLLLLEEEEEEELL
  confid    65556777777778887667777666664777445456666445666657
Context                                          nnnddddddnnmmmnhh
  confid                                         44555555554554477

   51       ....,....6....,....7....,....8....,....9....,....0
Seq         DKSKTWQANLRLISKFDTVEDFWALYNHIQLSSNLMPGCDYSLFKDGIEP
Angles      GlBBEEEHHEEEEEHHHHHHHHHHHHHHEEEBHHBBBlBBEEEEEBGxBBB
  confid    74555444344444444477787775545554557887755555542465
Sec struct  LLLEEEELLLEEELLLLLHHHHHHHHLLEELLLLLLLLLLEEEEELLLLLL
  confid    87634434433333445456666654443435676788764443467655
Context     hhhnnnnddnnn                nn          nnnn
  confid    777665644555                55          5445
```

This is a beta turn motif.

This is a helix N-cap motif.

I-sites/HMMSTR graphical output



I-sites predictions for vmbustosp.4912.covar

summary

MEME -- deterministic EM algorithm for motif finding, starting with initial guess

Gibbs sampling -- stochastic EM algorithm for motif finding, doesn’t need initial guess

K-means -- unsupervised learning of recurrent patterns, requires a metric space (distance or similarity).

Supervised learning -- EM in two spaces. Expectation in one space, maximization in the other.

I-sites/HMMSTR -- motifs and HMM based on linked motifs. For sec struct prediction in proteins.

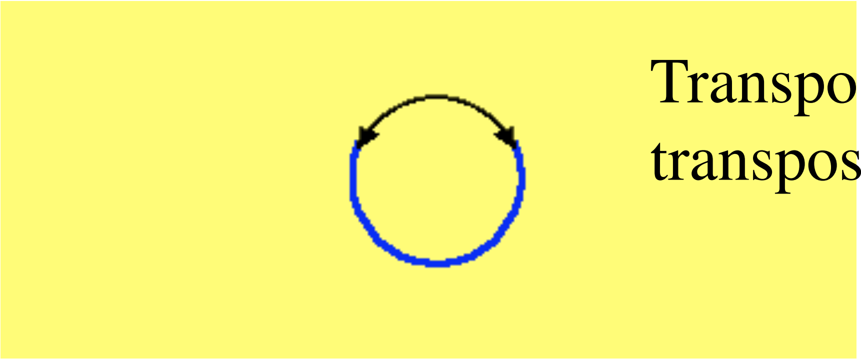
Repeats, Satellites & Transposable Elements

Transposable elements: junk dealers

Barbara McClintock

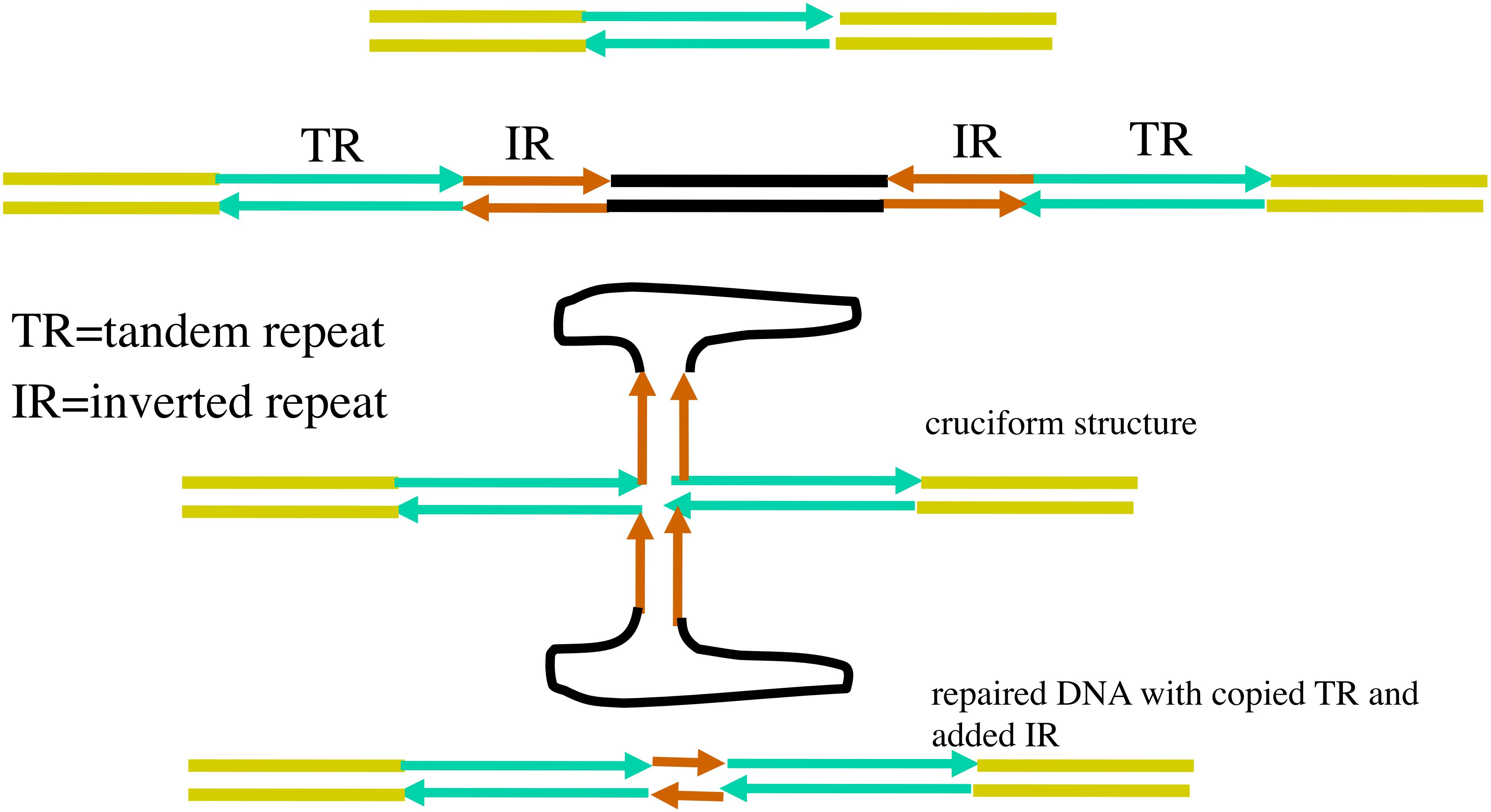
"Out standing in her field"

Transposable elements "jumping genes" lead to rapid germline variation.



Transposase, transposasome

Excision of transposon may leave a “scar”.



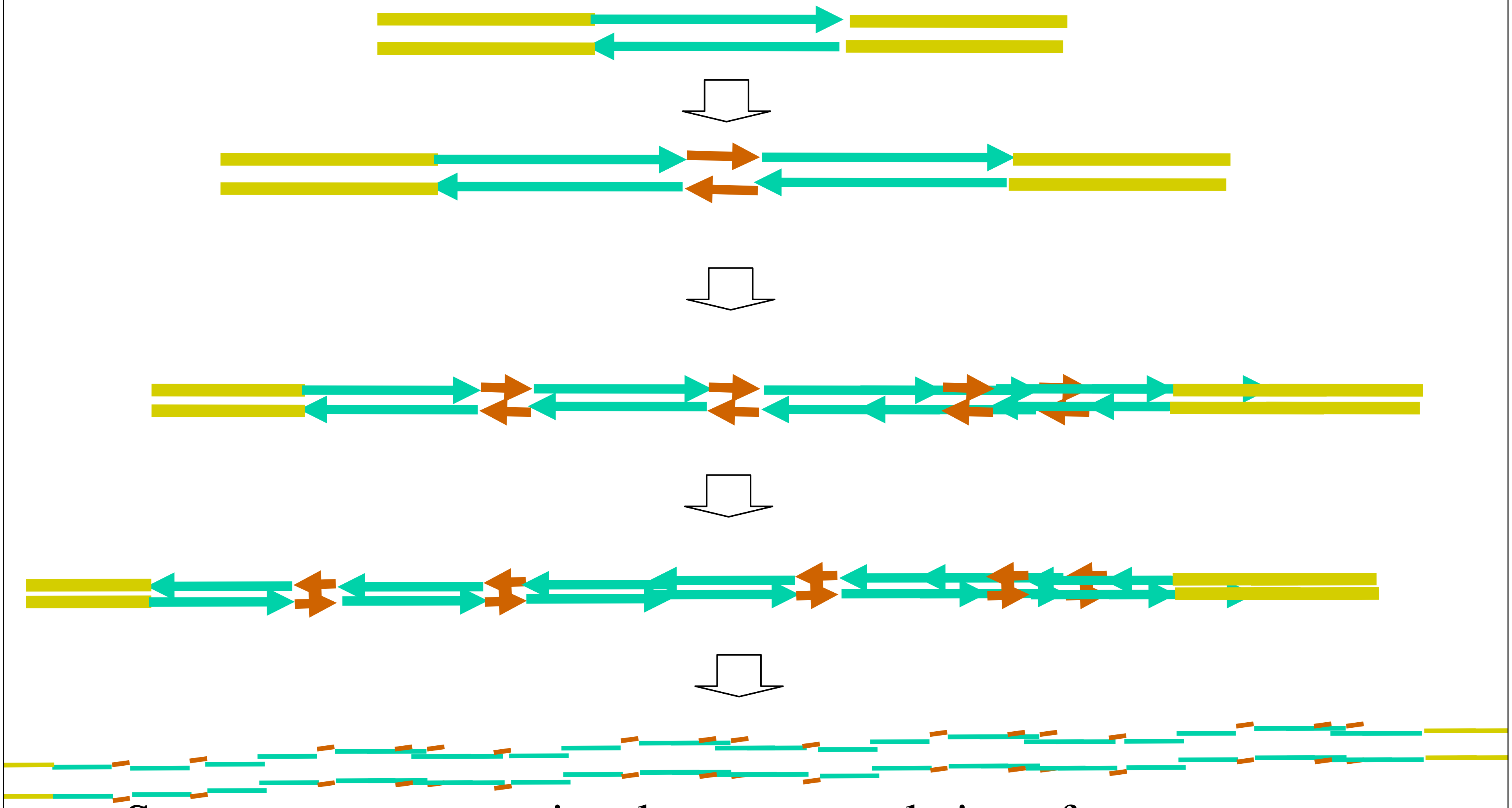
TR=tandem repeat

IR=inverted repeat

cruciform structure

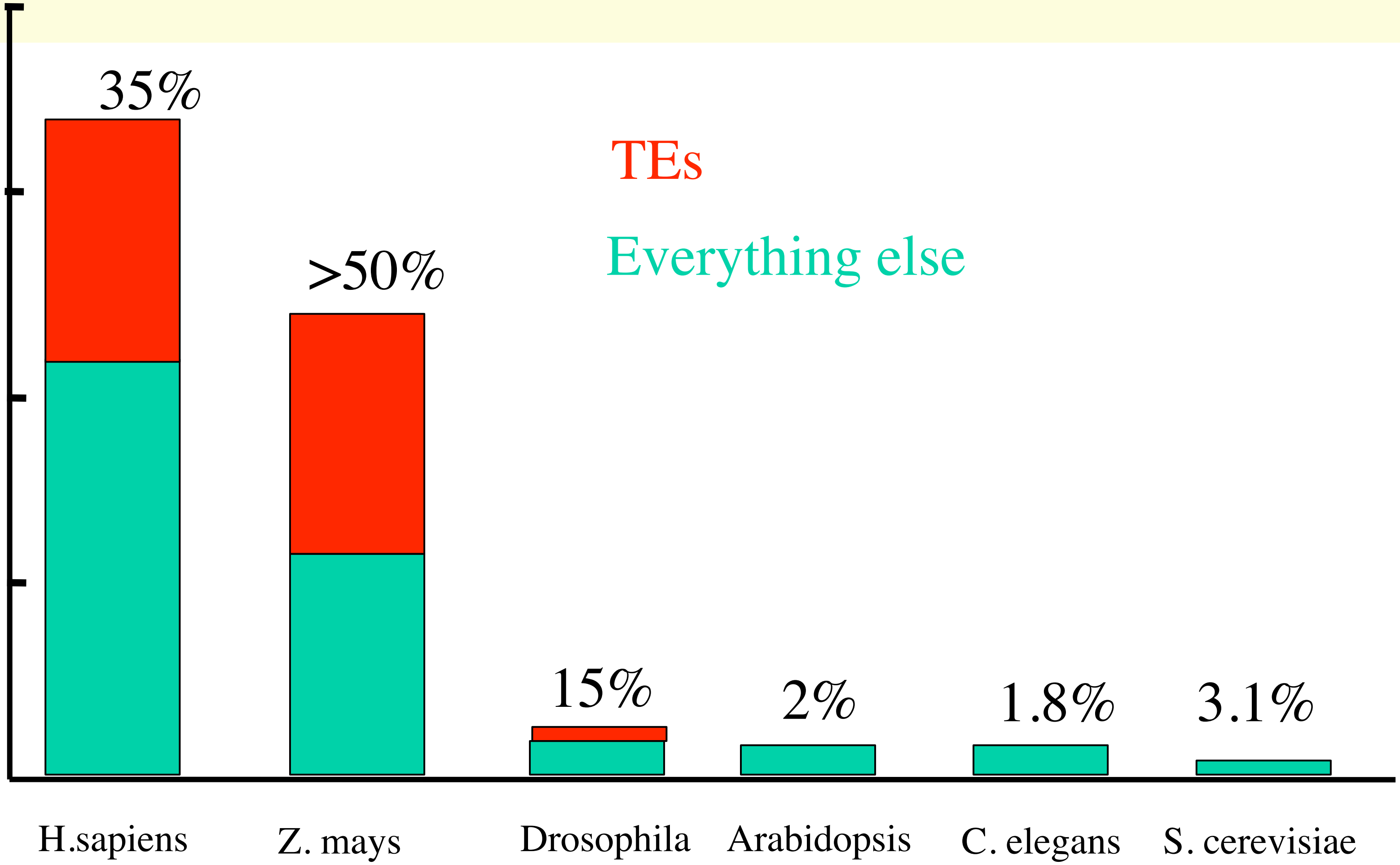
repaired DNA with copied TR and added IR

Millions of years of accumulated TE “scars”



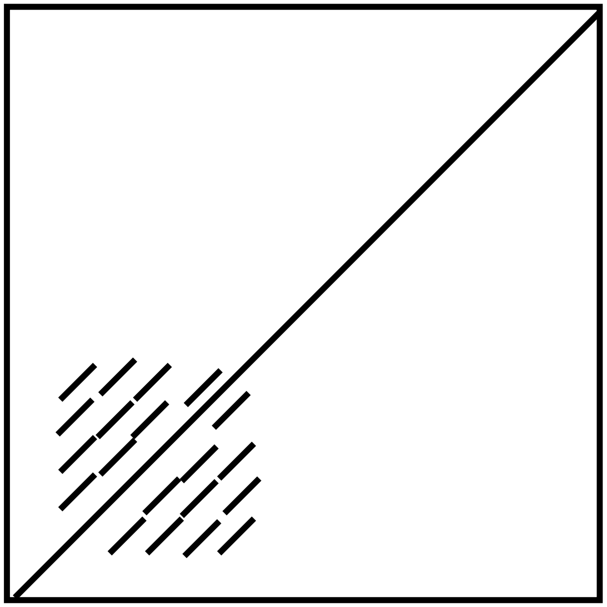
Some genomes contain a large accumulation of transposon scars.

Estimated Transposable element-associated DNA content in selected genomes



How do you recognize a repeat sequence?

- High scoring self-alignments
- High dot plot density
- Compositional bias



A repeat region
in a dot plot.

Types of repeat sequences

Satellites -- 1000+ bp in *heterochromatin*: centromeres, telomeres

Simple Sequence Repeats (SSRs), in *euchromatin* :

Minisatellites -- ~15bp (VNTR)

Microsatellites -- 2-6 bp

heterochromatin=compact, light bands
euchromatin=loose, dark bands.

microsatellite

```
541 gagccactag tgcttcattc tctcgctcct actagaatga acccaagatt gcccaggccc
601 aggtgtgtgt gtgtgtgtgt gtgtgtgtgt gtgtgtgtgt gtatagcaga gatggtttcc
661 taaagtaggc agtcagtcaa cagtaagaac ttggtgccgg aggtttgggg tcctggccct
721 gccactggtt ggagagctga tccgcaagct gcaagacctc tctatgcttt ggttctctaa
781 ccgatcaaat aagcataagg tcttccaacc actagcattt ctgtcataaa atgagcactg
841 tcctatttcc aagctgtggg gtcttgagga gatcatttca ctggccggac cccatttcac
```

a **microsatellite** in a dog (*canis familiaris*) gene.

Minisatellite

```
  1 tgattggtct ctctgccacc gggagatttc cttatttgga ggtgatggag gatttcagga
 61 tttggggat tttaggatta taggattacg ggattttagg gttctaggat tttaggatta
121 tggtatttta ggatttactt gattttggga ttttaggatt gagggatttt agggtttcag
181 gatttcggga tttcaggatt ttaagttttc ttgattttat gattttaaga ttttaggatt
241 tacttgattt tgggatttta ggattacggg attttagggt ttcaggattt cgggatttca
301 ggattttaag ttttcttgat tttatgattt taagatttta ggatttactt gattttggga
361 ttttaggatt acgggatttt agggtgctca ctatttatag aactttcatg gtttaacata
421 ctgaatataa atgctctgct gctctcgctg atgtcattgt tctcataata cgttcctttg
```

This 8bp tandem repeat has a consensus sequence `AGGATTTT`, but is almost never a perfect match to the consensus.

fun with bioinformatics jargon

ACRONYMS for satellites and transposons

SSR	Short Sequence Repeat
STR	Short Tandem Repeat
VNTR	Variable Number Tandem Repeat
LTR	Long Terminal Repeat
LINE	Long Interspersed Nuclear Element
SINE	Short Interspersed Nuclear Element
MITE	Miniature Inverted repeat Transposable Element (class III TE)
TE	Transposable Element
IS	Insertion Sequence
IR	Inverted Repeat
RT	Reverse Transcriptase
TPase	Transposase
Alu	11% of primate genome (SINE)
LINE1	14.6% of human genome
Tn7,Tn3,Tn10,Mu,IS50	transposons or transposable bacteriophage

Class I TE, uses RT.

Class II TE, uses TPase.

Class III TE, MITEs*

*Cl,ass III are now merged with Class II TEs.

Is there an evolutionary advantage of repeat sequences?

Repeat sequences are prone to

(1) locally: errors in replication

(2) non-locally: homologous recombination

Errors in replication (polymerase slippage) can lead to a change in the **reading frame,** eliminating a STOP codon, adding one, or translating to a different sequence entirely.

Neisseriae Gonorrheoae evades the human immune system by periodically (weeks) changing the **reading frame** of the **pilin surface antigen** protein.

(How) do you align repeat sequences?

A: Don’t align. Mask them out instead.

B: Dynamic Programming with special EVD. Align just like any other sequence, but using a special null model to assess the significance of the alignment score. Use EVD to fit random scores.

Remember: Low complexity sequences will have high-scoring alignments *randomly*. For example:

```
ATTTATATAATTAATATATAAATATAATAAATAT
```

aligned to

```
TATTATATATATATATATATTATATATATATATA
```

Random score is likely to have >50% identity!

INSTITUTE FOR Systems Biology

www.repeatmasker.org

Ariana Smit, Phil Green

Compares your seqeunce to a *curated library of known repeats* to a query sequence: Returns: (1) Location and type of each repeat, and/or
(2) Query sequence with repeats masked (set to "N")

Annotation Results

```
position in query                  matching repeat                          position in   repeat
begin     end      (left)          repeat          class/family             begin   end (left)  ID

1031265   1031302 (244491545) + C-rich           Low_complexity             3       41     (0) 624
1031638   1031782 (244491065) + (TG)n            Simple_repeat              1      145     (0) 625
1031794   1031886 (244490961) + (CGTG)n          Simple_repeat              3       97     (0) 626
1031900   1032062 (244490785) + (TG)n            Simple_repeat              1      163     (0) 627
1032330   1032614 (244490233) + AluJo            SINE/Alu                   5      287    (25) 628
```

If you must align repeat sequences, you need significance.

REMINDER: *Significance is what matters*!

[What is the likelihood of getting a score at "*random*".]

Getting e-values requires a **model** for **random scores.**

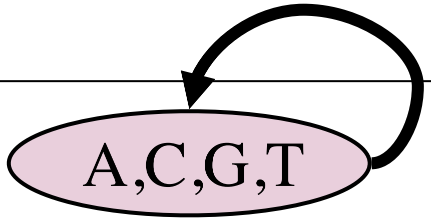
These scores are fit to a EVD. Using the EVD equation, we can convert a score to a *e-value*.

What is a good model for random alignments of low-complexity/repeat sequences?

Simplest null model (1) **Composition-biased model**.

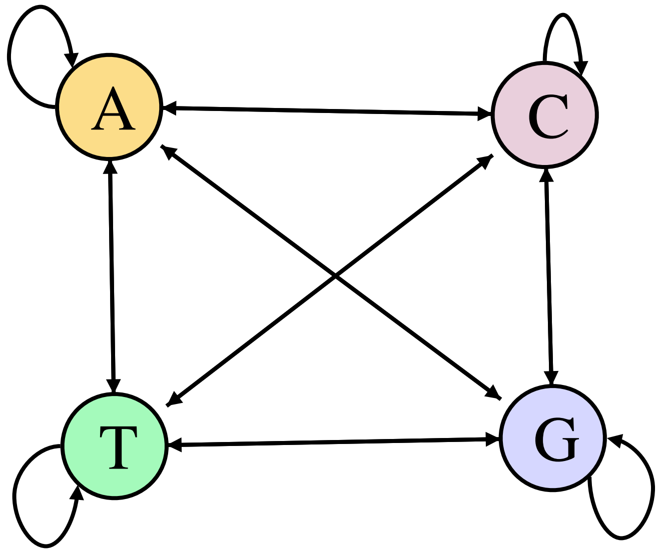
Generate random sequences based on composition. Align them.

Get scores. Fit the scores to the EVD.



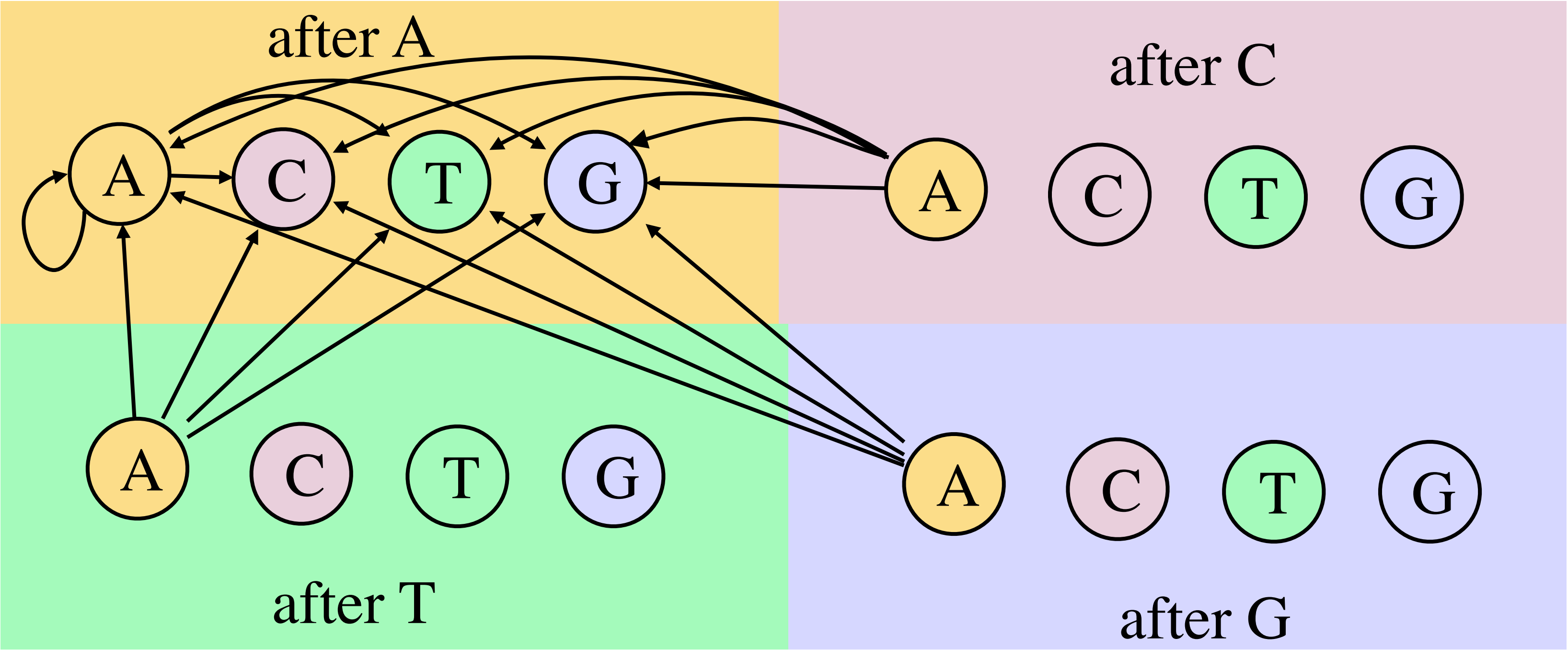
Getting expectation values for low complexity/repeat sequences.

Microrepeat null model (2) **Dinucleotide composition model**. Generate random sequences based on dinucleotide model, such as 4-state Markov chain. Align them. Get scores. Fit the scores to the EVD.



Getting expectation values for low complexity/repeat sequences.

Microrepeat null model (3) **Trinucleotide composition model**. Generate random sequences based on dinucleotide model, such as 16-state HMM. Align them. Get scores. Fit the scores to the EVD.

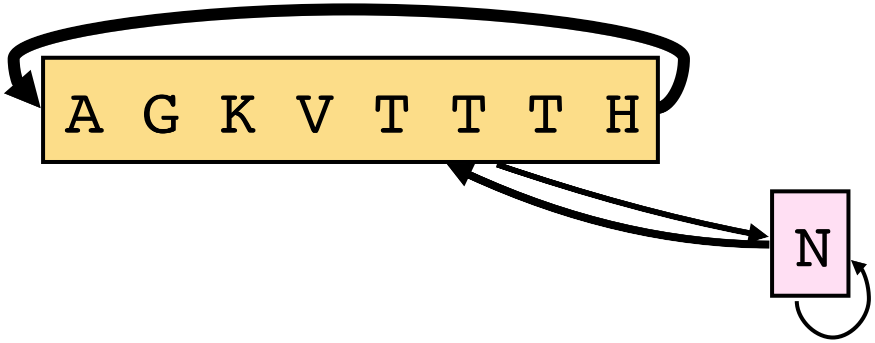


Only the arrows into the 4 “after A” states are shown

Getting expectation values for low complexity/repeat sequences.

Minirepeat null model (4) **Motif model**. (Grammatical model.)
Repeats are (possibly misspelled) words.

Generate sequences. Align them. Get scores. Fit the scores to the EVD.

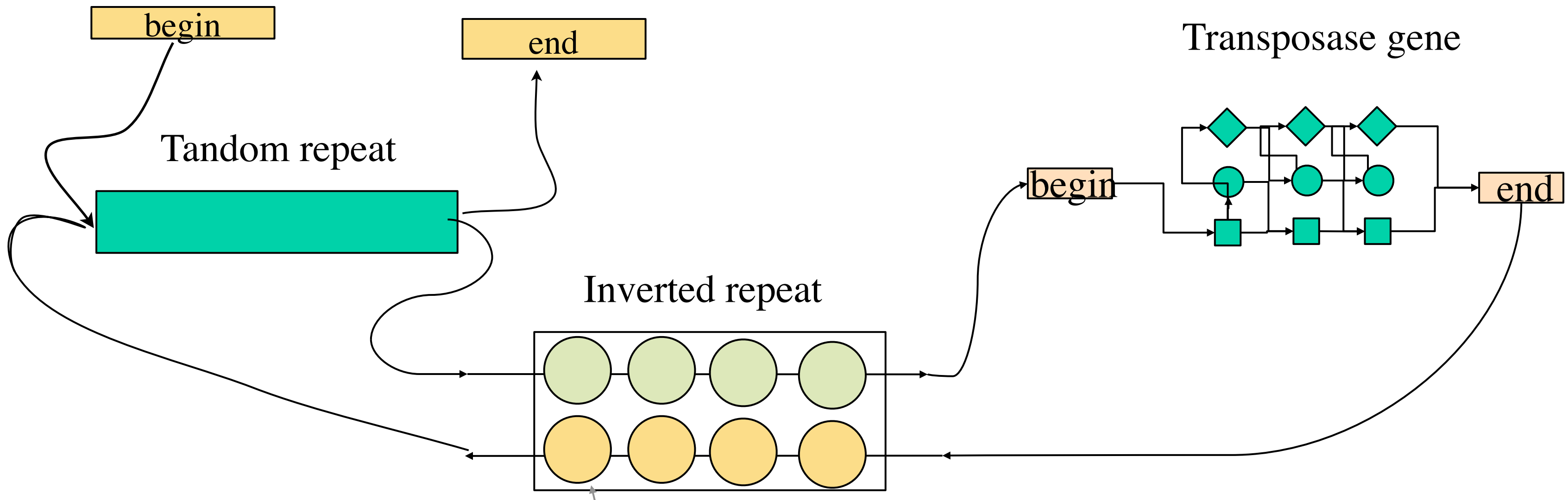


8 character misspelled-word repeat model, with occasional extra character(s).

In class exercise: create a HMM for a microsatellite.

- Using Netscape: Go to the NCBI database and download the nucleotide sequence with GenBank identifier (*gi)* 21912445
- Import it into Geneious.
- Find the microsatellite that starts at around 330.
Draw a motif HMM. Use *ProSite syntax*
- Run your model to generate a random microsatellite sequence.

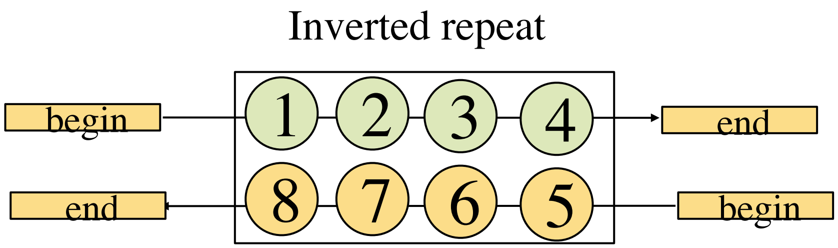
TE HMM?



Complementary base states. Training paired states enforces complementarity.

A heirarchical HMM is made by connecting the end and begin states of HMMs.

Constrained training of HMM states is possible.



In expectation/maximization training, we select the new parameters of the model.

In constrained training, we can enforce:

- identical emission probabilities
- complementary emission probabilities
- identical transition probablities.

For example in the maximization step of E/M: (‘ = expected value)

$$b_3(A) = (b'_3(A) + b'_6(T)) / 2$$