Identification of conserved patterns

What are sequence motifs?

Sequence motifs are short, conserved elements of a sequence alignment. They can be a short sequence of contiguous residues or a more distributed patterns.

Functionally related sequences will share similar distribution patterns of critical functional residues that are not necessarily contiguous.

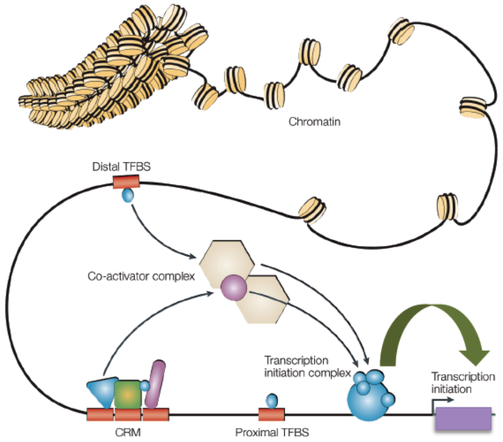
For example, conserved amino acid residues comprising the active site of an enzyme may be distant from each other in the protein sequence, but will still occur in a recognizable pattern because of the constraints imposed by the requirement for them to come together in a particular spatial configuration to form the active site in the 3D structure.

What are Nucleic Acid sequence motifs?

DNA sequence motifs are short, recurring patterns in DNA that are presumed to have a biological function. Often they indicate sequence-specific binding sites for proteins such as nucleases, transcription factors or **restriction enzymes** and transcription termination.

Examples:

- The **Pho4p** transcription factor binds specifically to the **CACGTG** DNA sequence in yeast.
- The **E2F** transcription factor binds the **TTTCGCGC** DNA sequence in eukaryotes.
- The restriction enzyme **EcoRI** specifically cuts DNA at instance of **GAATTC**.
- The **TATA box (TATAAT)** is recognized by TBP (Tata binding protein)/RNA polymerase in prokaryotes.



Transcription factors are proteins that modulate (activate/repress) the expression of **target genes** through the binding on **DNA cis-regulatory elements**

RNA motifs includes:

- Specific protein or ligand binding sites.
- Splicing sites (donor+acceptor sites, in mRNA).
- Motifs involved in transcriptional regulation (in mRNA).

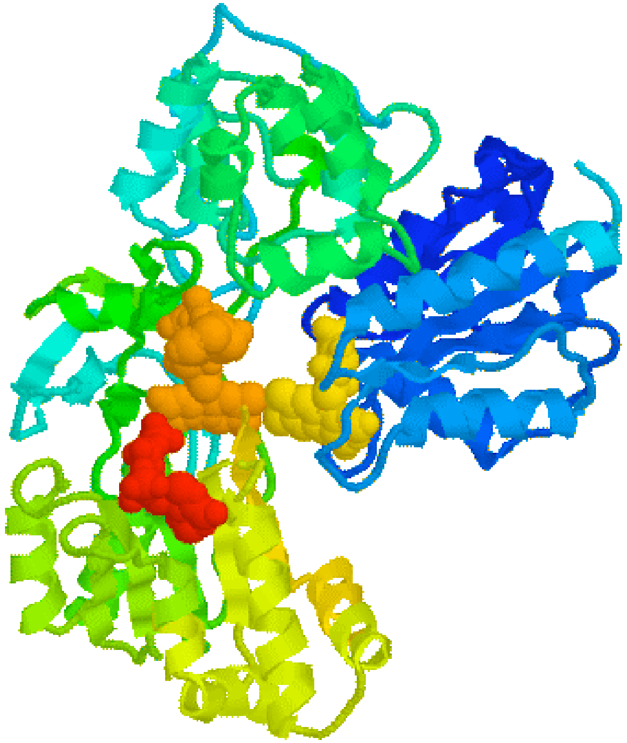
Examples:

- Important processes at the RNA level, including **ribosome binding site (Shine-Dalgarno** sequence **AGGAGGU**).
- mRNA processing e.g. **splicing, polyadenylation**.
- tRNA anticodon

What are protein sequence motifs?

Protein sequence motifs are short, recurring patterns in protein that are presumed to have a biological function or a particular structure.

They can be motifs responsible for **protein-protein interaction**, **nuclear localization signal** (NLS) or they can constitute the **enzyme's active site**.



Examples:

- The sequence **PKKKRKV** is a **NLS** specifically recognized by importin α.
- **Zn-finger** transcription factors share the consensus sequence **X_2-C-$X_{2,4}$-C-X_{12}-H-$X_{3\text{-}5}$-H** (X: any amino acid).
- Many **SH_3-binding epitopes** (part of the protein recognized by the immune system, e.g. antibody or T-cell) of proteins have a the consensus sequence B-P-p-B-P (B: aliphatic amino acids, p: often a Proline).

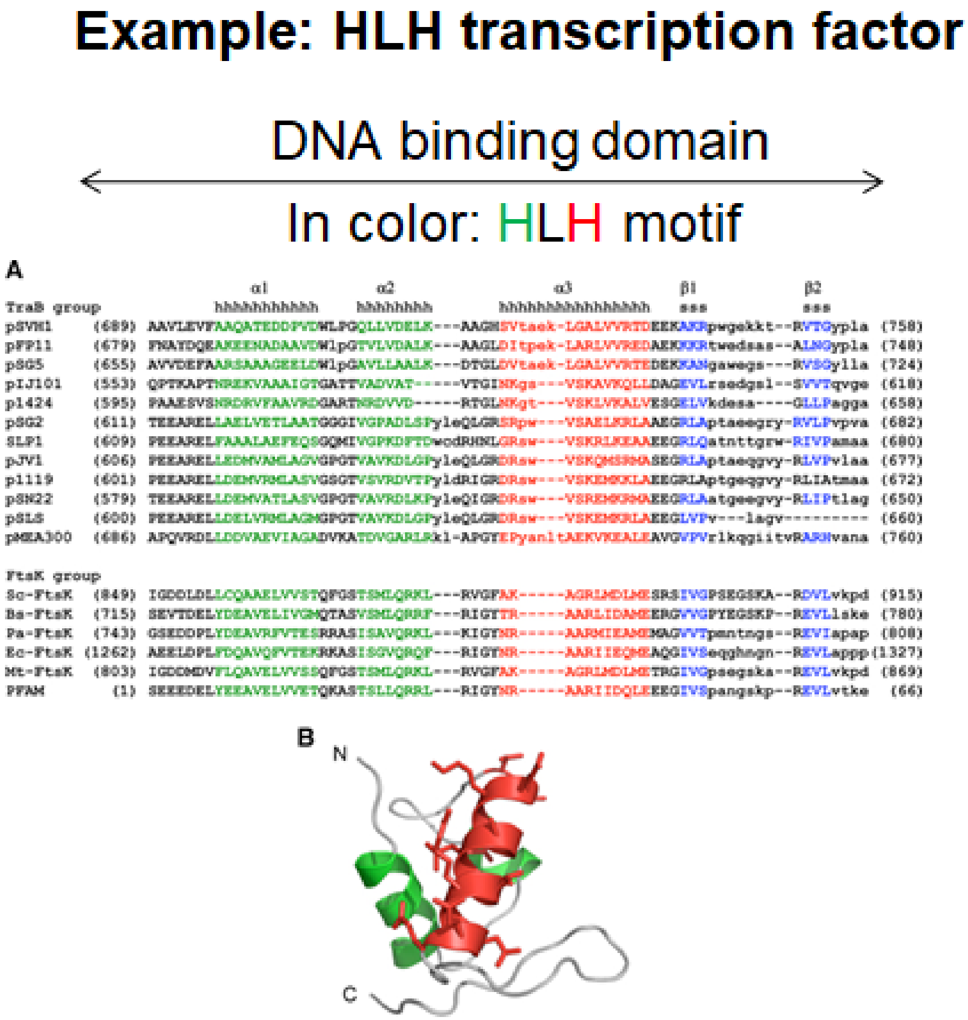
Motif vs Domain vs Fingerprint

Motifs: Any conserved sequence.

Domain: Conserved sequence that can be extracted from the whole protein sequence and that can form a correct fold. It is characterized by a particular 3D structure. **Contrary to domains, motifs may not be stable when they are extracted from the protein sequence**.

For example, a protein can have a DNA binding domain with a helix-turn-helix motif.

Family (fingerprint or print): is a conserved motif (or a group of motifs) that are used to characterize a protein family.



Example: HLH transcription factor

DNA binding domain

In color: HLH motif

A

```
                           α1          α2              α3           β1        β2
TraB group            hhhhhhhhhhh  hhhhhhhh        hhhhhhhhhhhhhhhh sss       sss
pSVH1     (689) AAVLEVFAAQATEDDPVDWLPGQLLVDELK---AAGHSVtaek-LGALVVRTDEEKAKRpwgekkt--RVTGypla (758)
pFP11     (679) FNAYDQEAKEENADAAVDWlpGTVLVDALK---AAGLDItpek-LARLVVREDAEKKKRtwedsas--ALNGypla (748)
pSG5      (655) AVVDEFAARSAAAGEELDWlpGAVLLAALK---DTGLDVtaek-LGALVVRTEDEKKANgawegs---RVSGylla (724)
pIJ101    (553) QPTKAPTNREKVAAAIGTGATTVADVAT-----VTGINKgs---VSKAVKQLLDAGEVLrsedgsl--SVVTqvge (618)
p1424     (595) PAAESVSNRDKVFAAVRDGARTNRDVVD-----RTGLNKgt---VSKLVKALVESGELVkdesa----GLLPagga (658)
pSG2      (611) TEEARELLAELVETLAATGGGIVGPADLSPyleQLGRSRpw---VSAELKRLAAEGRLAptaeegry-RVLPvpva (682)
SLP1      (609) PEEARELFAAALAEFEQSGQMIVGPKDFTDwodRHNLGRsw---VSKRLKEAAEEGRLQatnttgrw-RIVPamaa (680)
pJV1      (606) PEEARELLEDMVAMLAGVGPGTVAVKDLGPyleQLGRDRsw---VSKQMSRMASEGRLAptaeqgvy-RLVPvlaa (677)
p1119     (601) PEEARELLDEMVRMLASVGSGTVSVRDVTPyldRIGRDRsw---VSKEMKKLAEEGRLAptgeqgvy-RLIAtmaa (672)
pSN22     (579) TEEARELLDEMVATLASVGPGTVAVRDLKPyleQIGRDRsw---VSREMKRMAEEGRLAatgeegvy-RLIPtlag (650)
pSLS      (600) PEEARELLDELVRMLAGMGPGTVAVKDLGPyleQLGRDRsw---VSKEMKRLAEEGLVPv---lagv--------- (660)
pMEA300   (686) APQVRDLLDDVAEVIAGADVKATDVGARLKkl-APGYEPyanltAEKVKEALEAVGVPVrlkqgiitvRARHvana (760)

FtsK group
Sc-FtsK   (849) IGDDLDLLCQAAELVVSTQFGSTSMLQRKL---RVGFAK-----AGRLMDLMESRSIVGPSEGSKA--RDVLvkpd (915)
Bs-FtsK   (715) SEVTDELYDEAVELIVGMQTASVSMLQRRF---RIGYTR-----AARLIDAMEERGVVGPYEGSKP--REVLlske (780)
Pa-FtsK   (743) GSEDDPLYDEAVRFVTESRRASISAVQRKL---KIGYNR-----AARMIEAMEMAGVVTpmntngs--REVIapap (808)
Ec-FtsK  (1262) AEELDPLFDQAVQFVTEKRKASISGVQRQF---RIGYNR-----AARIIEQMEAQGIVSeqghngn--REVLappp(1327)
Mt-FtsK   (803) IGDDMDVFLQAVELVVSSQFGSTSMLQRKL---RVGFAK-----AGRLMDLMETRGIVGpsegska--REVLvkpd (869)
PFAM        (1) SEEEDELYEEAVELVVETQKASTSLLQRRL---RIGYNR-----AARIIDQLEEEGIVSpangskp--REVLvtke  (66)
```

B

Why finding motifs in DNA and protein sequences?

Identifying patterns in DNA and protein sequences provides important clues about their possible regulation, structure and/or function.

Identifying particular DNA motifs will also be crucial in deciphering genomic sequences (e.g. gene prediction).

Given a completely sequenced genome, you can find

(a) A favorite gene possessing a particular regulatory element in their promoter.
(b) A set of co-expressed genes and you want to know if they are co-regulated by a common transcription factor.
(c) Genes having a pattern from a set of patterns (i.e. possible targets of a given transcription factor).

How to find motifs: pattern matching vs pattern discovery

Different approaches can be used depending of the question:

You know the genes, you know the pattern... and you want to know if the genes have the given pattern ⇒ **pattern matching**

You know the genes, you don't know the patterns ... and you want to detect possible patterns common to all genes ⇒ **pattern discovery or pattern matching with library**

You know the patterns, you don't know the genes ... and you want to detect possible genes having a given pattern ⇒ **classification**

Different Sequence Motif Finding Algorithms

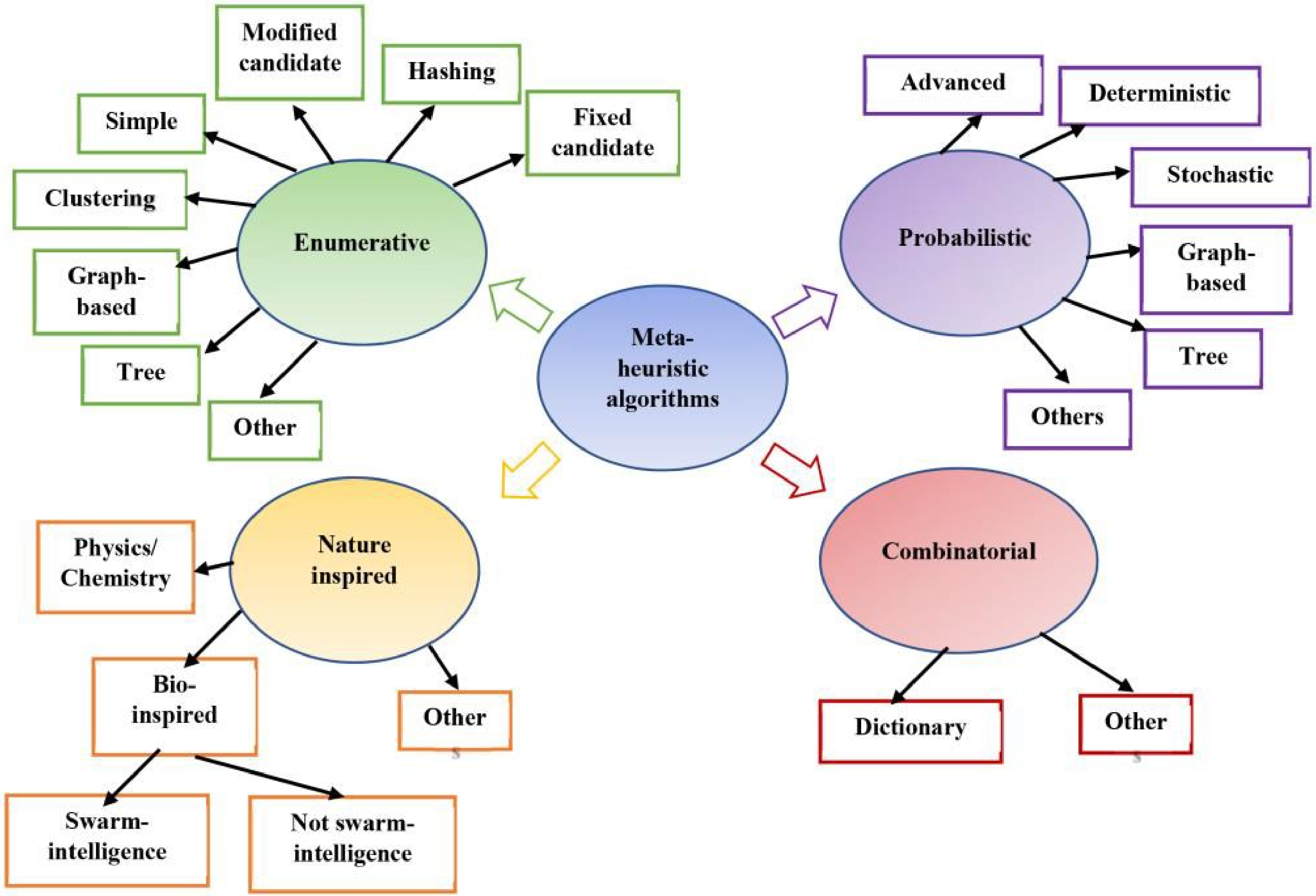
(A) Enumeration-based approaches
This approach searches for **consensus sequences**; motifs are predicted based on the enumeration of words and computing word similarities so this approach is sometimes called the **word enumeration approach**. Popular algorithms based on this approach are DREME, CisFinder, Weeder, FMotif, and MCES.

(B) Probabilistic approaches
It constructs a probabilistic model called **position-specific weight matrix (PSWM)** or **motif matrix** that specifies a distribution of bases for each position in sites to distinguish motifs *vs*. non-motifs and it requires few search parameters. The most popular methods based on probabilistic approach are MEME, STEME, EXTREME, AlignACE, and BioProspector.

(C) Nature-inspired methods
Evolutionary algorithms can over-come the disadvantages of local search and synthesize local search and global search. Examples of evolutionary algorithms are: Genetic Algorithm (GA), Differential Evolution (DE), Evolution Strategy, Multi-modal Optimization, Cuckoo-Search (CS), Levy flight, Bacterial Colony Optimization, and Intelligent Water Drops algorithm. Swarm intelligence is a special class of evolutionary algorithm including Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC) algorithm, and Ant Colony Optimization (ACO) algorithm.

Hashim et al., 2019, Avicenna Journal of Medical Biotechnology.

Different Sequence Motif Finding Algorithms



Hashim et al., 2019, Avicenna Journal of Medical Biotechnology.

Different Sequence Motif Finding Algorithms

(A) Enumerative approaches

1. ***Simple word enumeration:*** The first class is based on simple word enumeration.

2. ***Clustering-based method:*** Sharov *et al* proposed word clustering method called CisFinder to detect short motif with high processing speed in large sequences (up to 50 *Mb*).

3. ***Tree-based method:*** Pavesi *et al* presented Weeder algorithm based on count matching patterns with specific and most extreme mismatches.

4. ***Graph theoretic-based method:*** The graph-theoretic method represents a motif in-stance, as a clique; the graph G is built by representing each l-mer in the input sequences by vertex and the edge between a pair of vertices representing a pair of l-mer in different input sequences having the Hamming distance between the substrings which is less than or equal to 2d. Then, cliques of size N are searched for in this graph. Popular graph-theoretic methods are WIN-NOWER, Pruner, and cWINNOWER.

5. ***Hashing-based method:*** Buhler *et al* developed random projection algorithm for a PMP that projects every l-mer in the input data into a smaller space by hashing. Initially, a projection of l-dimensional space onto a k-dimensional sub-space for all subsequences in the input set is developed, and random projection is constructed by choosing random k positions from l position. Using this projection, each l-mer is hashed to its corresponding bucket.

6. ***Fixed candidates and modified candidate-based methods:*** The sixth class is fixed candidates that select candidate motifs from input sequences and use them for motif scanning while the seventh class is modified candidate that selects one candidate from the input sequence and modifies it letter by letter.

Hashim et al., 2019, Avicenna Journal of Medical Biotechnology.

Different Sequence Motif Finding Algorithms

B. Probabilistic approach

1. ***Deterministic approach:*** Expectation-Maximization (EM) is the famous example of deterministic approach. EM for motif finding was first introduced by Lawrence *et al* and it consists of two main steps, the first called "Expectation step" that estimates the values of some set of unknowns based on a set of parameters. The second step is "Maximization step" that uses those estimated values to refine the parameters over several iterations.

2. ***Stochastic approach:*** **Gibbs sampling** is a famous stochastic approach, similar to EM algorithm. Pseudocode of the Gibbs sampling algorithm for motif detection follows these steps:
   - Random initializing of motif positions in the input N sequences with an assumption of the presence of one motif per sequence,
   - Choosing one sequence at random,
   - Computing PWM for the other N-1 sequences using staring positions of motifs and background probabilities for each base using the non-motif positions,
   - Calculating probability of each possible motif location in the removed sequence using PWM and back-ground probabilities,
   - For the removed sequence, choosing a new starting position based on step 4.

3. ***Advanced approach:*** Different algorithms were proposed based on Bayesian approach. Jensen *et al* proposed an algorithm based on Bayesian approach with Markov chain Monte Carlo. Xing *et al* proposed LOGOS (Integrated LOcal and GlObal motif sequence model) algorithm that combines between HMDM (Hidden Markov Dirichlet-Multinomial) for local alignment model for each different motif and HMM (Hidden Markov model) for global motif distribution model for the occurrence of multiple motifs.

Hashim et al., 2019, Avicenna Journal of Medical Biotechnology.

Different Sequence Motif Finding Algorithms

C. Nature-inspired algorithms

1. ***Genetic Algorithm:*** GA is a probabilistic optimization algorithm based on evolutionary computing. GA is inspired from biological evolution processes like selection, crossover, and mutation.

2. ***Particle Swarm Optimization:*** PSO is a new global optimization technique for solving continuous optimization problems. PSO algorithm is characterized by its simple computations and information sharing within the algorithm.

3. ***Artificial Bee Colony algorithm:*** ABC algorithm is a type of swarm-based algorithm proposed by Karaboga. It simulates the behavior of honey bees to find a food source. Two fundamental properties to obtain swarm intelligent behavior in honey bee colonies are self-organizing and division of labor.

4. ***Ant Colony Optimization algorithm:*** The ACO algorithm is a metaheuristic optimization technique that mimics the behavior of real ants, which try to find the shortest path to the food from their nest.

5. ***Cuckoo Search algorithm:*** CS is a new simple heuristic search algorithm that is more efficient than GA and PSO. CS is inspired from brood parasitism reproduction behavior of some cuckoo species in combination with Lévy flight behavior.

Hashim et al., 2019, Avicenna Journal of Medical Biotechnology.

Some examples of identifying motifs

Finding Motifs with Gibbs Sampling Method

B. Probabilistic approach: Stochastic approach: **Gibbs sampling**

Aim: To identify four letter motif.

Step 1: initialization- choose randomly at each sequence a candidate position for the motif.

S1: ACGTATAG
S2: TATACAGT
S3: CTATAGCA
S4: AAGCTATA

→

S1: ACGTATAG
S2: TATACAGT
S3: CTATAGCA
S4: AAGCTATA

Step 2: prepare the count matrix for motif and non-motif regions.

Non-motif
S1: ATAG
S2: TATA
S3: CTAA
S4: AAGA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	9	1	2	1	1
C	1	2	1	0	1
G	2	0	0	3	0
T	4	1	1	0	2
Total	**16**	**4**	**4**	**4**	**4**

Motif
S1: ACGT
S2: CAGT
S3: TAGC
S4: CTAT

Finding Motifs with Gibbs Sampling Method

Step 3: calculate the probability matrix for motif and non-motif regions.

Probability matrix for motif:

$$p_{c,k} = \frac{n_{c,k} + dc}{(N-1) + db}$$

Where $n_{c,k}$: count of character c at position k, d_c: pseudocount of a single character (=1), N: number of sequences, d_b: pseudocount of all characters (=4 for DNA, =20 for protein).

Probability matrix for non-motif:

$$p_{c,k} = \frac{n_{c,0} + dc}{(N-1)(L-W) + db}$$

Where $n_{c,k}$: count of character c at position k, d_c: pseudocount of a single character (=1), N: number of sequences, L: length of the sequence, W: length of the motif, d_b: pseudocount of all characters (=4 for DNA, =20 for protein).

Finding Motifs with Gibbs Sampling Method

Step 3: calculate the probability matrix for motif and non-motif regions.

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	0.625	0.286	0.429	0.286	0.286
C	0.125	0.429	0.286	0.143	0.286
G	0.1875	0.143	0.143	0.571	0.143
T	0.3125	0.286	0.286	0.143	0.429
Total	**1.25**	**1.144**	**1.144**	**1.145**	**1.144**

Finding Motifs with Gibbs Sampling Method

Step 4: Remove one sequence randomly and re-calculate the probability matrix for motif and non-motif regions. Let us take out, e.g., sequence S1. Then

Non-motif
S2: TATA
S3: CTAA
S4: AAGA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	7	0	2	1	1
C	1	2	0	0	1
G	1	0	0	2	0
T	3	1	1	0	1
Total	**12**	**3**	**3**	**3**	**3**

Motif
S2: CAGT
S3: TAGC
S4: CTAT

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	0.667	0.167	0.5	0.333	0.333
C	0.167	0.5	0.167	0.167	0.333
G	0.167	0.167	0.167	0.5	0.167
T	0.333	0.333	0.333	0.167	0.333
Total	**1.334**	**1.167**	**1.167**	**1.167**	**1.166**

Finding Motifs with Gibbs Sampling Method

Step 5: Score each word (of length W=4) from the removed sequence.

$$\text{Word score} = \frac{p\,(word\ from\ the\ probability\ matrix\ for\ motif)}{p\,(word\ from\ the\ probability\ matrix\ for\ non{-}motif)}$$

S1: ACGTATAG

Word No.	Word	Probability for motif	Probability for non-motif	Word score
1	ACGT	0.167x0.167x0.5x0.333 = 0.0046	0.667x0.167x0.167x0.333 = 0.0062	0.74
2	**CGTA**	**0.5x0.167x0.167x0.333 = 0.0046**	**0.167x0.167x0.333x0.667 = 0.0062**	**0.74**
3	GTAT	0.167x0.333x0.333x0.333 = 0.0062	0.167x0.333x0.667x0.333 = 0.0124	0.5
4	TATA	0.333x0.5x0.167x0.333 = 0.0093	0.333x0.667x0.333x0.667 = 0.0493	0.19
5	ATAG	0.167x0.333x0.333x0.167 = 0.0031	0.667x0.333x0.667x0.167 = 0.0247	0.13

Finding Motifs with Gibbs Sampling Method

Repeat Step 4: Include the highest scoring motif, remove one more sequence randomly (e.g. S2) and re-calculate the probability matrix for motif and non-motif.

Non-motif
S1: ATAG
S3: CTAA
S4: AAGA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	7	0	1	1	1
C	1	2	0	0	1
G	2	0	1	1	0
T	2	1	1	1	1
Total	**12**	**3**	**3**	**3**	**3**

Motif
S1: CGTA
S3: TAGC
S4: CTAT

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	0.667	0.167	0.333	0.333	0.333
C	0.167	0.5	0.167	0.167	0.333
G	0.25	0.167	0.333	0.333	0.167
T	0.25	0.333	0.333	0.333	0.333
Total	**1.334**	**1.167**	**1.166**	**1.166**	**1.166**

Finding Motifs with Gibbs Sampling Method

Repeat Step 5: Score each word (of length W=4) from the removed sequence.

$$\text{Word score} = \frac{p\,(word\ from\ the\ probability\ matrix\ for\ motif)}{p\,(word\ from\ the\ probability\ matrix\ for\ non{-}motif)}$$

S2: TATACAGT

Word No.	Word	Probability for motif	Probability for non-motif	Word score
1	**TATA**	**0.333x0.333x0.333x0.333 = 0.0123**	**0.25x0.667x0.25x0.667 = 0.0278**	**0.44**
2	ATAC	0.167x0.333x0.333x0.333 = 0.0062	0.667x0.25x0.667x0.167 = 0.0186	0.33
3	TACA	0.333x0.333x0.167x0.333 = 0.0062	0.25x0.667x0.167x0.667 = 0.0186	0.33
4	ACAG	0.167x0.167x0.333x0.167 = 0.0016	0.667x0.167x0.667x0.25 = 0.0186	0.09
5	CAGT	0.5x0.333x0.333x0.333 = 0.0185	0.167x0.667x0.25x0.25 = 0.0070	2.64

Although the highest score is 2.64 for the motif CAGT, let us consider the motif TATA (as it is expected) with score 0.44.

Finding Motifs with Gibbs Sampling Method

Repeat Step 4: Include the highest scoring motif, remove one more sequence randomly (e.g. S3) and re-calculate the probability matrix for motif and non-motif.

Non-motif
S1: ATAG
S2: CAGT
S4: AAGA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	6	0	1	1	2
C	1	2	0	0	0
G	3	0	1	0	0
T	2	1	1	2	1
Total	**12**	**3**	**3**	**3**	**3**

Motif
S1: CGTA
S2: TATA
S4: CTAT

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	0.583	0.167	0.333	0.333	0.5
C	0.167	0.5	0.167	0.167	0.167
G	0.333	0.167	0.333	0.167	0.167
T	0.25	0.333	0.333	0.5	0.333
Total	**1.333**	**1.167**	**1.166**	**1.167**	**1.167**

Finding Motifs with Gibbs Sampling Method

Repeat Step 5: Score each word (of length W=4) from the removed sequence.

$$\text{Word score} = \frac{p\,(word\ from\ the\ probability\ matrix\ for\ motif)}{p\,(word\ from\ the\ probability\ matrix\ for\ non{-}motif)}$$

S3: CTATAGCA

Word No.	Word	Probability for motif	Probability for non-motif	Word score
1	CTAT	0.5x0.333x0.333x0.333 = 0.0185	0.167x0.25x0.583x0.25 = 0.0061	3.03
2	**TATA**	**0.333x0.333x0.5x0.5 = 0.0277**	**0.25x0.583x0.25x0.583 = 0.0212**	**1.31**
3	ATAG	0.167x0.333x0.333x0.167 = 0.0031	0.583x0.25x0.583x0.333 = 0.0283	0.11
4	TAGC	0.333x0.333x0.167x0.167 = 0.0031	0.25x0.583x0.333x0.167 = 0.0081	0.38
5	AGCA	0.167x0.333x0.167x0.5 = 0.0046	0.583x0.333x0.167x0.583 = 0.0189	0.24

Although the highest score is 3.03 for the motif CTAT, let us consider the motif TATA (as it is expected) with score 1.31.

Finding Motifs with Gibbs Sampling Method

Repeat Step 4: Include the highest scoring motif, remove one more sequence randomly (e.g. S4) and re-calculate the probability matrix for motif and non-motif.

Non-motif
S1: ATAG
S2: CAGT
S3: CGCA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	4	0	2	0	3
C	3	1	0	0	0
G	3	0	1	0	0
T	2	2	0	3	0
Total	**12**	**3**	**3**	**3**	**3**

Motif
S1: CGTA
S2: TATA
S3: TATA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	0.417	0.167	0.5	0.167	0.667
C	0.333	0.333	0.167	0.167	0.167
G	0.333	0.167	0.333	0.167	0.167
T	0.25	0.5	0.167	0.667	0.167
Total	**1.333**	**1.167**	**1.167**	**1.168**	**1.168**

Finding Motifs with Gibbs Sampling Method

Repeat Step 5: Score each word (of length W=4) from the removed sequence.

$$\text{Word score} = \frac{p\ (word\ from\ the\ probability\ matrix\ for\ motif)}{p\ (word\ from\ the\ probability\ matrix\ for\ non{-}motif)}$$

S4: AAGCTATA

Word No.	Word	Probability for motif	Probability for non-motif	Word score
1	AAGC	0.167x0.5x0.167x0.167 = 0.0023	0.417x0.417x0.333x0.333 = 0.0193	0.12
2	AGCT	0.167x0.333x0.167x0.167 = 0.0016	0.417x0.333x0.333x0.25 = 0.0116	0.14
3	GCTA	0.167x0.167x0.667x0.667 = 0.0124	0.333x0.333x0.25x0.417 = 0.0116	1.07
4	CTAT	0.333x0.167x0.167x0.167 = 0.0016	0.333x0.25x0.417x0.25 = 0.0087	0.18
5	**TATA**	**0.5x0.5x0.667x0.667 = 0.1112**	**0.25x0.417x0.25x0.417 = 0.0109**	**10.20**

Finding Motifs with Gibbs Sampling Method

Repeat Step 4: Include the highest scoring motif, remove one more sequence randomly (e.g. S1) and re-calculate the probability matrix for motif and non-motif.

Non-motif
S2: CAGT
S3: CGCA
S4: AAGC

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	4	0	3	0	3
C	4	0	0	0	0
G	3	0	0	0	0
T	1	3	0	3	0
Total	**12**	**3**	**3**	**3**	**3**

Motif
S2: TATA
S3: TATA
S4: TATA

	Non-motif	Motif			
	Background count (k=0)	Position 1 (k=1)	Position 2 (k=2)	Position 3 (k=3)	Position 4 (k=4)
A	0.417	0.167	0.667	0.167	0.667
C	0.417	0.167	0.167	0.167	0.167
G	0.333	0.167	0.167	0.167	0.167
T	0.167	0.667	0.167	0.667	0.167
Total	**1.334**	**1.168**	**1.168**	**1.168**	**1.168**

Finding Motifs with Gibbs Sampling Method

Repeat Step 5: Score each word (of length W=4) from the removed sequence.

$$\text{Word score} = \frac{p\ (word\ from\ the\ probability\ matrix\ for\ motif)}{p\ (word\ from\ the\ probability\ matrix\ for\ non{-}motif)}$$

S1: ACGTATAG

Word No.	Word	Probability for motif	Probability for non-motif	Word score
1	ACGT	0.167x0.167x0.167x0.167 = 0.0008	0.417x0.417x0.333x0.167 = 0.0097	0.08
2	CGTA	0.167x0.167x0.667x0.667 = 0.0124	0.417x0.333x0.167x0.417 = 0.0097	1.28
3	GTAT	0.167x0.167x0.167x0.167 = 0.0008	0.333x0.167x0.417x0.167 = 0.0039	0.21
4	**TATA**	**0.667x0.667x0.667x0.667 = 0.1979**	**0.167x0.417x0.167x0.417 = 0.0048**	**41.23**
5	ATAG	0.167x0.167x0.167x0.167 = 0.0008	0.417x0.167x0.417x0.333 = 0.0097	0.08

Now the expected motif was found with the highest score and including it in the motif (for the next round) would not change the probability matrix for motif and non-motif. Thus, the iteration can now be terminated the motif TATA found.

Different Sequence Motif Finding Algorithms

C. Nature-inspired algorithms: Artificial Bee Colony algorithm

Aim: To identify five letter motif.

Sequence number	Sequence														
1	C	A	G	G	C	C	A	T	A	A	G	C	C	G	A
2	A	C	A	T	A	A	A	C	G	G	C	T	A	T	A
3	T	T	C	A	T	A	A	G	A	G	G	C	A	T	C
4	G	C	G	C	A	A	G	C	A	T	A	A	A	T	T
Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Step 1: Randomly generate position vectors using all sequences

Sequence	1	2	3	4
Position	6	6	2	11

Sequence	1	2	3	4
Position	6	2	3	10

Hashim et al., 2019, Avicenna Journal of Medical Biotechnology.

Different Sequence Motif Finding Algorithms

Step 2: Generate the alignment matrix for the selected windows from all sequences

Alignment matrix for [6,6,2,11] vector.

Sequence \ Position	1	2	3	4	5
1	C	A	T	A	A
2	A	A	C	G	G
3	T	C	A	T	A
4	A	A	A	T	T

Alignment matrix for [6,3,2,10] vector.

Sequence \ Position	1	2	3	4	5
1	C	A	T	A	A
2	C	A	T	A	A
3	C	A	T	A	A
4	T	A	A	A	T

Profile matrix for [6,6,2,11] vector.

Letter\ Position	1	2	3	4	5
A	2	3	2	1	2
C	1	1	1	0	0
G	0	0	0	1	1
T	1	0	1	2	1

Profile matrix for [6,3,2,10] vector.

Letter\ Position	1	2	3	4	5
A	0	4	1	4	3
C	3	0	0	0	0
G	0	0	0	0	0
T	1	0	3	0	1

Different Sequence Motif Finding Algorithms

Step 3: Generate the consensus sequence and frequency for the selected windows from all sequences.

Consensus sequence and frequency for [6,6,2,11] vector.

Position	1	2	3	4	5
Sequence	A	A	A	T	A
Max. frequency	2/4	3/4	2/4	2/4	2/4

Consensus sequence and frequency for [6,3,2,10] vector.

Position	1	2	3	4	5
Sequence	C	A	T	A	A
Max. frequency	3/4	4/4	3/4	4/4	3/4

Step 4: Calculate the similarity score.

$$Sim(P) = \sum_{i=1}^{l} \frac{\max(p(i))}{l \times N}$$

OR

$$Sim(P) = \sum_{j=1}^{l} \sum_{i=1}^{4} \frac{p(i,j)}{N} \log \frac{p(i,j)}{N}$$

Where *max(p(i))* and *l* represent the frequency of the dominant nucleotide of the *i*th column in the profile matrix and the length of the motif, respectively. Given that *P* is a *4×l* profile matrix and *p(i,j)* is the element of *i*th row and *j*th column of *P. N:* number of sequences.

Different Sequence Motif Finding Algorithms

Step 4: Calculate the similarity score.

Similarity score for the vector [6,6,2,11]

2/4+3/4+2/4+2/4+2/4 = 2.75/20 for the motif AAATA

Similarity score for the vector [6,3,2,10]

3/4+4/4+3/4+4/4+3/4 = 4.25/20 for the motif CATAA

Similarity (entropy) score for the vector [6,6,2,11]

1/4 x [2/4 log (2/4/4)+1/4 log (1/4/4) + 0 + 1/4 log (1/4/4) + 3/4 log (3/4/4) + 1/4 log (1/4/4) + 0 + 0 + 2/4 log (2/4/4) + 1/4 log (1/4/4) + 0 + 1/4 log (1/4/4) + 1/4 log (1/4/4) + 0 + 1/4 log (1/4/4) + 2/4 log (2/4/4)+ 2/4 log (2/4/4) + 0 + 1/4 log (1/4/4) + 1/4 log (1/4/4)] = ?? for the motif AAATA

Similarity (entropy) score for the vector [6,3,2,10]

1/4 x [0 + 3/4 log (3/4/4) + 0 + 1/4 log (1/4/4) + 4/4 log (4/4/4) + 0 + 0 + 0 + 1/4 log (1/4/4) + 0 + 0 + 3/4 log (3/4/4) + 4/4 log (4/4/4) + 0 + 0 + 0 + 3/4 log (3/4/4) + 0 + 0 + 1/4 log (1/4/4)] = ?? for the motif CATAA

Representing a sequence motif

How to represent a sequence motif?

In some cases, a pattern can be represented by a single short string of nucleotides e.g. **GAATTC**

Unfortunately, this representation is very restrictive.

Indeed, most of the time, **variability in the pattern is allowed. How to account for the variability of the patterns?**

For example: HindII binds to the sequences GTYRAC, where Y stands for ‘C or T’ (pYrimidine) and R stands for ‘A or G’ (puRine).

The variability allowed in the patterns makes the representation and the identification of patterns challenging.

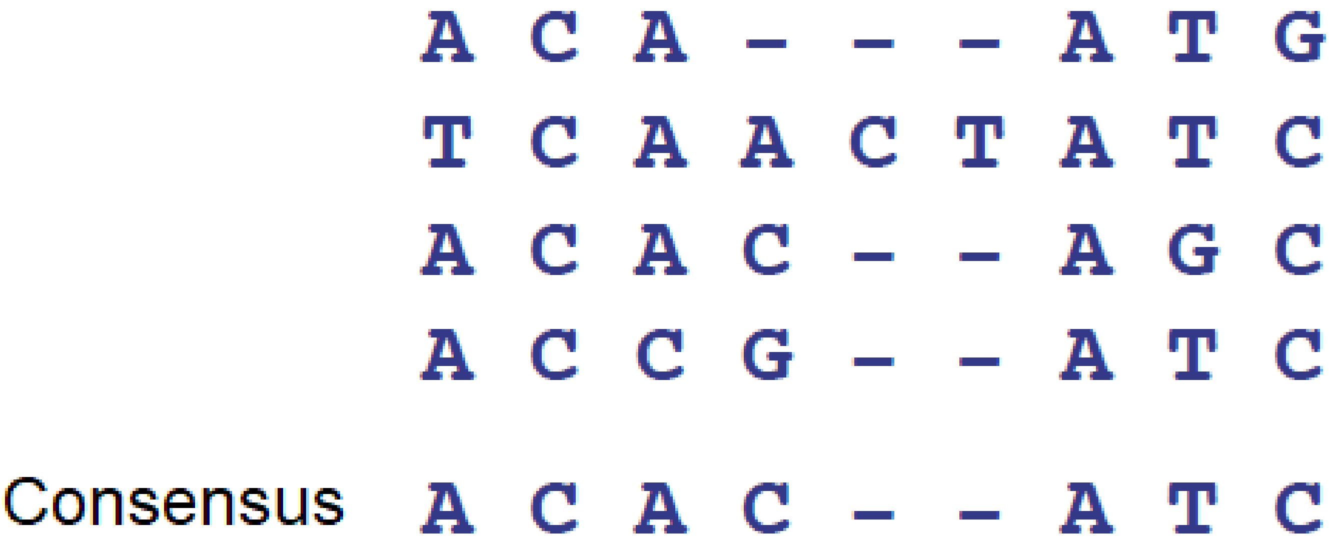
We can calculate how often we would expect these consensus sequences to occur, based on their length and degeneracy. The probability that a random 6-mer matches the EcoRI binding site is $(1/4)^6$, so the site occurs about once every 4^6 (= 4,096) bp in a random DNA sequence. The HindII binding site, containing two positions where two out of four bases can match, would occur once per $4^4 \times 2^2$ (= 1,024) bp.

Three common motif representations

- Consensus sequence / regular expressions
- Profile matrices (PWM, PSSM)
- Hidden Markov models (HMM)

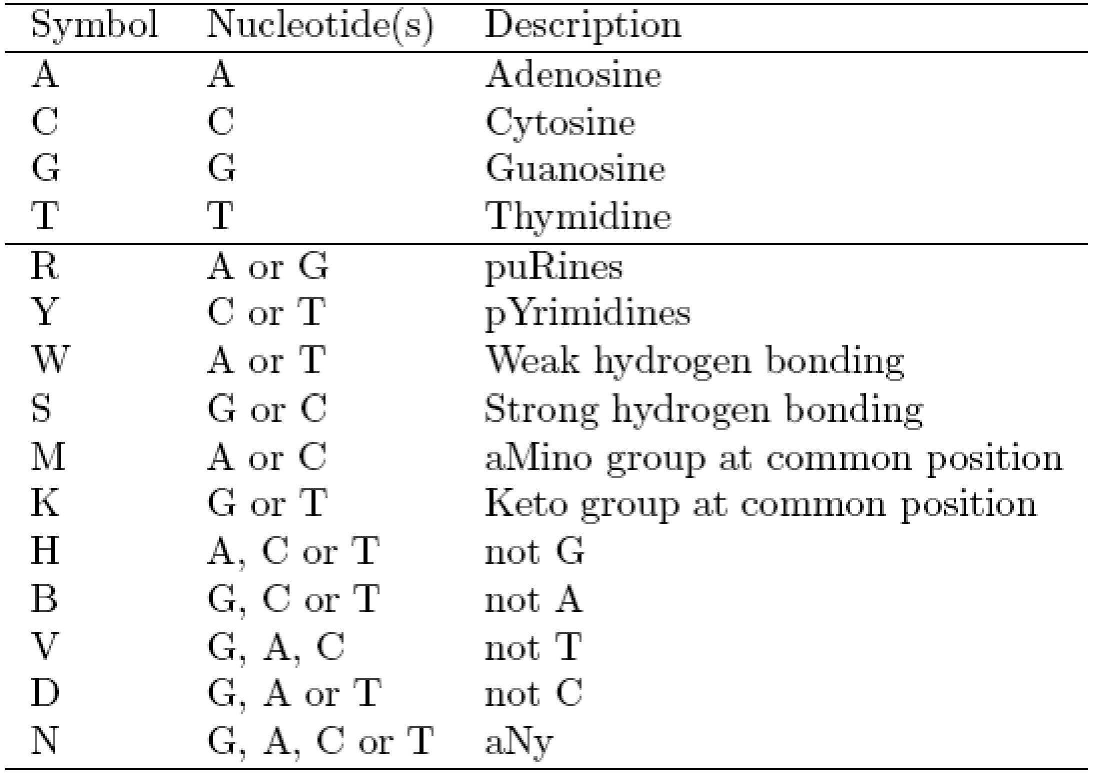
1. Consensus sequence

A consensus sequence is a string that summarizes a pattern the best.



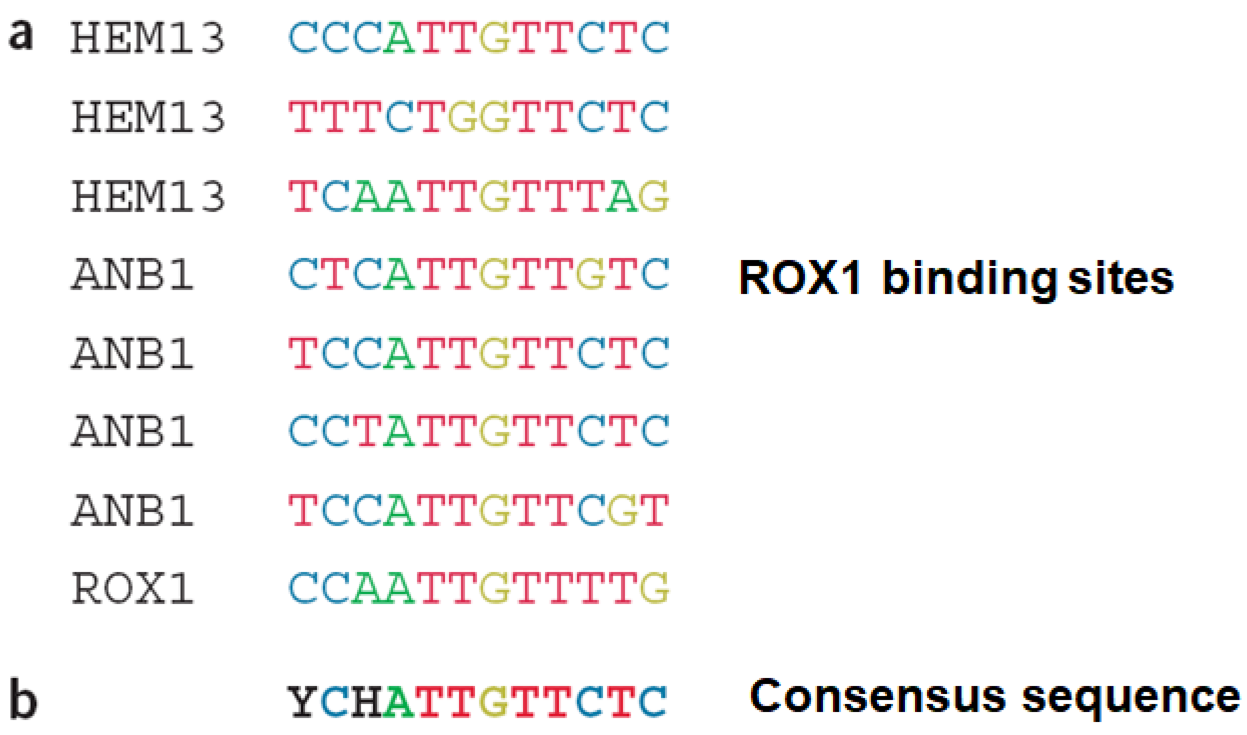
	A	C	A	-	-	-	A	T	G
	T	C	A	A	C	T	A	T	C
	A	C	A	C	-	-	A	G	C
	A	C	C	G	-	-	A	T	C
Consensus	A	C	A	C	-	-	A	T	C

The **IUPAC code can be used to refine the consensus sequence**, allowing to assign to a given position with 2 or 3 possible nucleotides. For example the letter **Y means either C or T (pYrimidine).**



Symbol	Nucleotide(s)	Description
A	A	Adenosine
C	C	Cytosine
G	G	Guanosine
T	T	Thymidine
R	A or G	puRines
Y	C or T	pYrimidines
W	A or T	Weak hydrogen bonding
S	G or C	Strong hydrogen bonding
M	A or C	aMino group at common position
K	G or T	Keto group at common position
H	A, C or T	not G
B	G, C or T	not A
V	G, A, C	not T
D	G, A or T	not C
N	G, A, C or T	aNy

Example: ROX1 is a transcription factor in *S. cerevisiae* involved in the regulation of heme-repressed and heme-induced genes.



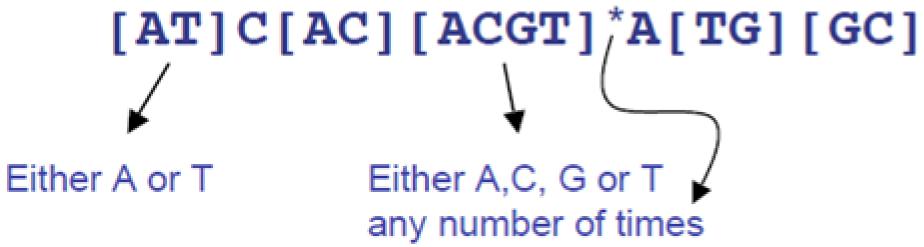
a	HEM13	CCCATTGTTCTC	
	HEM13	TTTCTGGTTCTC	
	HEM13	TCAATTGTTTAG	
	ANB1	CTCATTGTTGTC	ROX1 binding sites
	ANB1	TCCATTGTTCTC	
	ANB1	CCTATTGTTCTC	
	ANB1	TCCATTGTTCGT	
	ROX1	CCAATTGTTTTG	
b		YCHATTGTTCTC	Consensus sequence

Such consensus sequence can be used to search in any DNA sequences "strings" that match the consensus. A certain number of mismatch may also be allowed.

2. Regular expression

A regular expression (regexp) is a notational algebra that describes a string.

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



[AT]C[AC][ACGT]*A[TG][GC]

Either A or T

Either A,C, G or T any number of times

Example: C_2H_2 Zinc-finger motif

$$X_2\text{-}C\text{-}X_{2,4}\text{-}C\text{-}X_{12}\text{-}H\text{-}X_{3\text{-}5}\text{-}H$$

Regular expressions do not capture the statistics of the variation in sequence patterns - they just tell what letters are permissible at each position in the pattern.

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

Regular expression [AT]C[AC][ACGT]*A[TG][GC]

```
Consensus sequence     A C A C - - A T C
Improbable sequence    T C C T - - A G G
```

They are not distinguished in the regular expression

3. Position specific scoring matrices (PSSM)

Profiles capture the frequency of each letter at each position in the pattern so you can tell how well a potential site matches the pattern (the **probability of the site)**.

Starting from a multiple alignment, one can build a matrix which reflects the preferred residues at each position:

- Each column represents a position
- Each row represents a residue (20 rows for proteins, 4 rows for DNA)
- The cells indicate the frequency of each residue at each position of the multiple alignment.

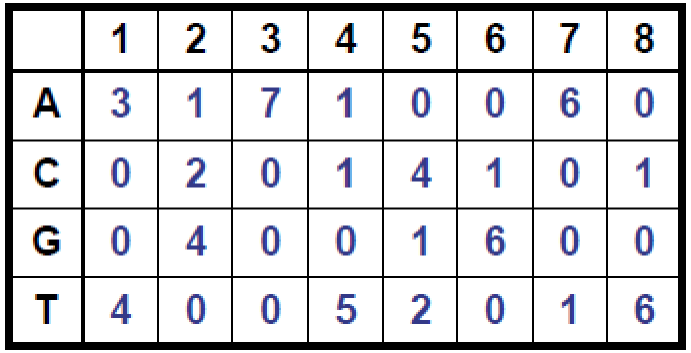
Profile matrices (PFM)

Site (1)	A	G	A	T	C	C	A	T
Site (2)	T	G	A	C	T	G	A	T
Site (3)	T	C	A	T	C	G	T	T
Site (4)	A	G	A	T	T	G	A	T
Site (5)	T	C	A	A	G	G	A	T
Site (6)	T	G	A	T	C	G	A	C
Site (7)	A	A	A	T	C	G	A	T
Consensus:	T	G	A	T	C	G	A	T
IUPAC consensus:	W	V	A	H	B	S	W	Y

Position-specific frequency matrix (counts)

	1	2	3	4	5	6	7	8
A	3	1	7	1	0	0	6	0
C	0	2	0	1	4	1	0	1
G	0	4	0	0	1	6	0	0
T	4	0	0	5	2	0	1	6

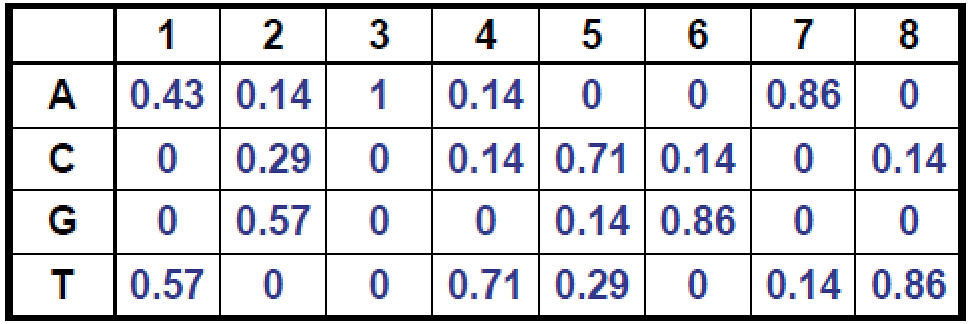
Profile matrices (PFM)



	1	2	3	4	5	6	7	8
A	3	1	7	1	0	0	6	0
C	0	2	0	1	4	1	0	1
G	0	4	0	0	1	6	0	0
T	4	0	0	5	2	0	1	6

Position-specific frequency matrix (frequencies)

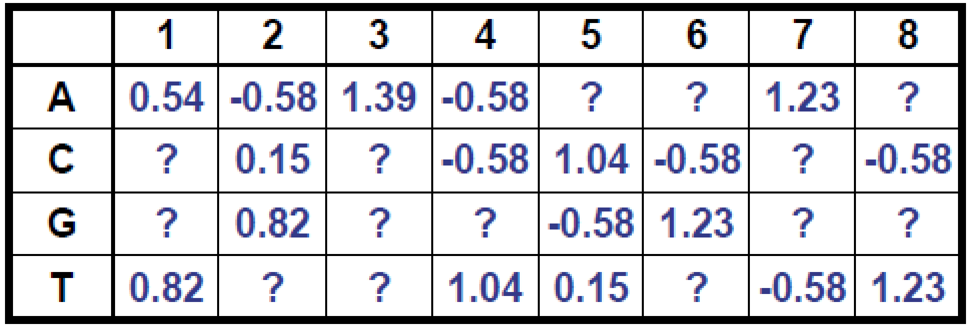
$$f_{i,j} = \frac{n_{i,j}}{\sum_{r=1}^{A} n_{r,j}}$$



	1	2	3	4	5	6	7	8
A	0.43	0.14	1	0.14	0	0	0.86	0
C	0	0.29	0	0.14	0.71	0.14	0	0.14
G	0	0.57	0	0	0.14	0.86	0	0
T	0.57	0	0	0.71	0.29	0	0.14	0.86

j = 1,2,... L (number of positions)
i = 1,2,... A (A = alphabet size; 4 for nucleic acids, 20 for amino acids)

Position-weight matrix (PWM)

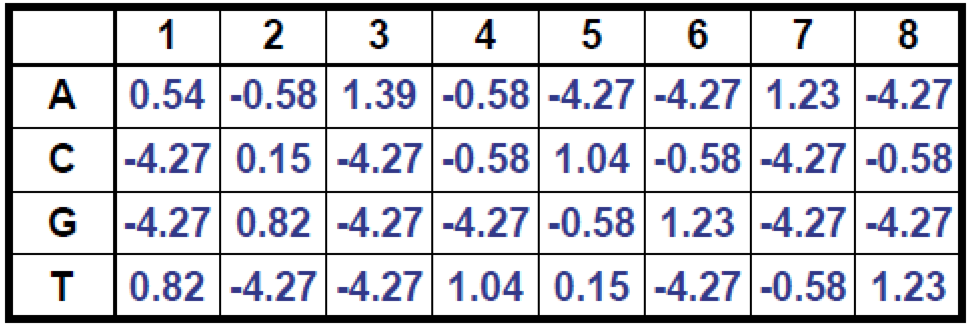


	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	?	?	1.23	?
C	?	0.15	?	-0.58	1.04	-0.58	?	-0.58
G	?	0.82	?	?	-0.58	1.23	?	?
T	0.82	?	?	1.04	0.15	?	-0.58	1.23

$$W_{i,j} = \ln\left(\frac{f_{i,j}}{p_i}\right)$$

p_i = prior probability
(here: p_A=p_C=p_G=p_T=0.25)

Position-weight matrix (PWM) with pseudo-weights



	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	-4.27	-4.27	1.23	-4.27
C	-4.27	0.15	-4.27	-0.58	1.04	-0.58	-4.27	-0.58
G	-4.27	0.82	-4.27	-4.27	-0.58	1.23	-4.27	-4.27
T	0.82	-4.27	-4.27	1.04	0.15	-4.27	-0.58	1.23

$$W_{i,j} = \ln\left(\frac{f'_{i,j}}{p_i}\right)$$

$$f'_{i,j} = \frac{n_{i,j} + p_i k}{\sum_{r=1}^{A} n_{r,j} + k}$$

k = pseudo-count
(here: k=0.1)

The pseudo-weights have been introduced by Hertz & Stormo (1999) to account for the small number of sequences used to build the PWM matrix.

How to choose the appropriate pseudo-count *k?*

Most of the binding-site matrices have been constructed on the basis of a small number of sites, often below 10. The $\mathrm{Pho}_4\mathrm{P}$ matrices, as found in TRANSFAC database, e. g., has been built from 8 known binding sites. Some residues have thus a frequency of 0. This gives a weight of -∞, which means that we consider as completely impossible for the transcription factor to bind this site. However, this might also be an artifact due to an incomplete sampling.

Hertz & Stormo (1999) proposed to correct the frequencies by introducing a pseudoweights
(k):

$$f'_{i,j} = \frac{n_{i,j} + p_i k}{\sum_{r=1}^{A} n_{r,j} + k}$$

The pseudo-weight has to be chosen in order to guaranty that the **sum of the frequencies** is still one.

A typical value for the pseudo-weight is **1/(alphabet size)**, but more elaborated theories have been developed to better choose the pseudo-weight.

$$\sum_{r=1}^{A} f'_{i,j} = 1$$

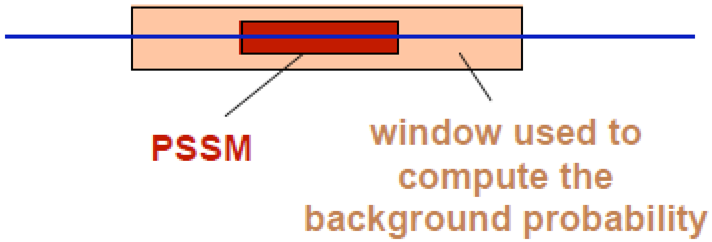
How to choose the appropriate background probabilities p_i

The four bases in DNA sequences do not occur equally often. The GC content strongly varies from one organism to another (51% in *E. coli,* 41% in *human,* 36% in *S. cerevisiae,* 19% in *Plasmodium falciparum,* 72% in *Streptomyces coelicolor).*

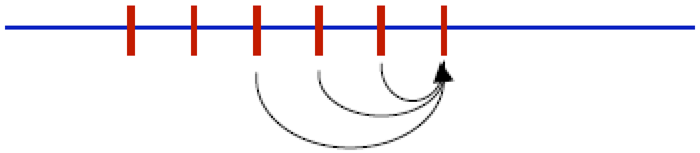
This bias can easily be taken into account in the background probabilities p_i.

Background probabilities can also take into account the local variation in the residues content (moving windows) or dependence between successive residues (Markov chain).

Moving windows

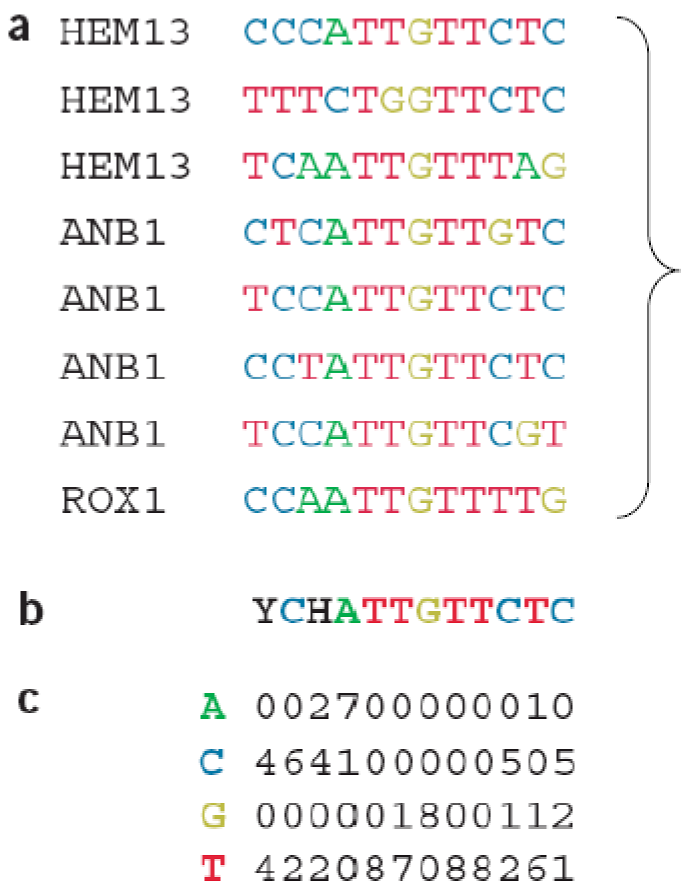


Markov chains



$p(R_j) = f\,(R_{j-1}, R_{j-2}, \ldots)$
where R_j is residue at position j

Profile matrices - example



a

HEM13	CCCATTGTTCTC
HEM13	TTTCTGGTTCTC
HEM13	TCAATTGTTTAG
ANB1	CTCATTGTTGTC
ANB1	TCCATTGTTCTC
ANB1	CCTATTGTTCTC
ANB1	TCCATTGTTCGT
ROX1	CCAATTGTTTTG

ROX1 binding sites

b YCHATTGTTCTC

Consensus sequence

c

A	002700000010
C	464100000505
G	000001800112
T	422087088261

Position-specific frequency matrix

How to score a sequence with a given position-weight matrix?

Site (1)	A	G	A	T	C	C	A	T
Site (2)	T	G	A	C	T	G	A	T
Site (3)	T	C	A	T	C	G	T	T
Site (4)	A	G	A	T	T	G	A	T
Site (5)	T	C	A	A	G	G	A	T
Site (6)	T	G	A	T	C	G	A	C
Site (7)	A	A	A	T	C	G	A	T

	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	-4.27	-4.27	1.23	-4.27
C	-4.27	0.15	-4.27	-0.58	1.04	-0.58	-4.27	-0.58
G	-4.27	0.82	-4.27	-4.27	-0.58	1.23	-4.27	-4.27
T	0.82	-4.27	-4.27	1.04	0.15	-4.27	-0.58	1.23

...	A	G	T	C	G	T	A	C	T	C	T	A	C	...

How to score a sequence with a given position-weight matrix?

	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	-4.27	-4.27	1.23	-4.27
C	-4.27	0.15	-4.27	-0.58	1.04	-0.58	-4.27	-0.58
G	-4.27	0.82	-4.27	-4.27	-0.58	1.23	-4.27	-4.27
T	0.82	-4.27	-4.27	1.04	0.15	-4.27	-0.58	1.23

...	A	C	G	A	T	C	G	A	T	C	T	A	C	...

score/position	0.54	0.15	-4.27	-0.58	0.15	-0.58	-4.27	-4.27

score total	-13.13

	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	-4.27	-4.27	1.23	-4.27
C	-4.27	0.15	-4.27	-0.58	1.04	-0.58	-4.27	-0.58
G	-4.27	0.82	-4.27	-4.27	-0.58	1.23	-4.27	-4.27
T	0.82	-4.27	-4.27	1.04	0.15	-4.27	-0.58	1.23

...	A	C	G	A	T	C	G	A	T	C	T	A	C	...

score/position	-4.27	0.82	1.39	1.04	1.04	1.23	1.23	1.23

score total	3.71

How to score a sequence with a given position-weight matrix?

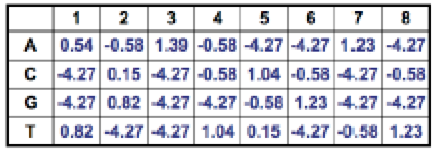
	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	-4.27	-4.27	1.23	-4.27
C	-4.27	0.15	-4.27	-0.58	1.04	-0.58	-4.27	-0.58
G	-4.27	0.82	-4.27	-4.27	-0.58	1.23	-4.27	-4.27
T	0.82	-4.27	-4.27	1.04	0.15	-4.27	-0.58	1.23

...	A	C	G	A	T	C	G	A	T	C	T	A	C	...

score/position	-4.27	-0.58	-4.27	-0.58	-4.27	-4.27	-0.58	-0.58

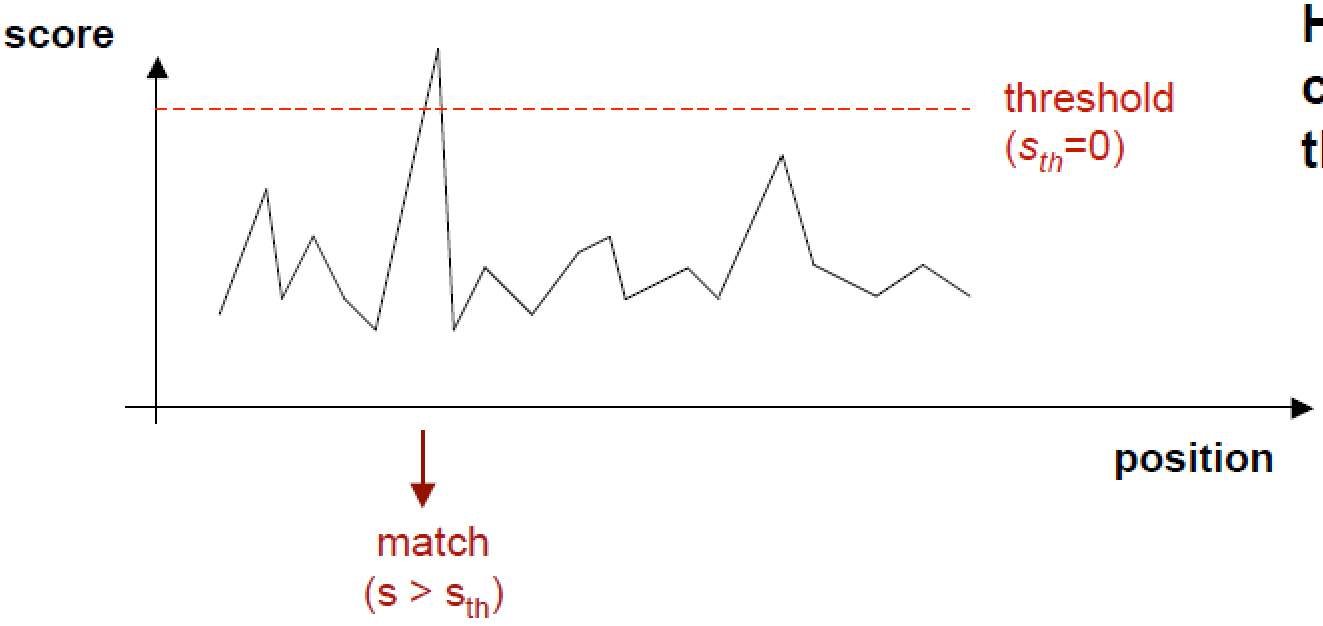
score total	-19.4

How to score a sequence with a given position-weight matrix?



	1	2	3	4	5	6	7	8
A	0.54	-0.58	1.39	-0.58	-4.27	-4.27	1.23	-4.27
C	-4.27	0.15	-4.27	-0.58	1.04	-0.58	-4.27	-0.58
G	-4.27	0.82	-4.27	-4.27	-0.58	1.23	-4.27	-4.27
T	0.82	-4.27	-4.27	1.04	0.15	-4.27	-0.58	1.23

AGATCCATTGACCGTTAGATTGAAGATTGATAGATTGATTTTGATCGACAAAGTG...



How to choose the threshold?

How to choose the threshold?

There is a trade:

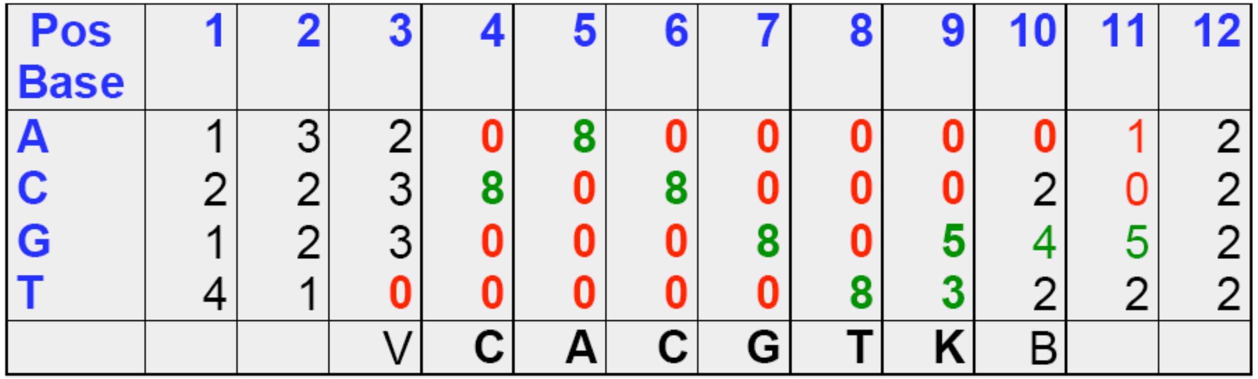
- Threshold too high => high selectivity, but low sensitivity
  High confidence in the predicted sites, but many real sites are missed

- Threshold too low => high sensitivity, but low selectivity
  The real sites are drowned in a lot of false positives.

One approach is to select the threshold on the basis of scores returned when the matrix is used to scan known binding sites for the factor.

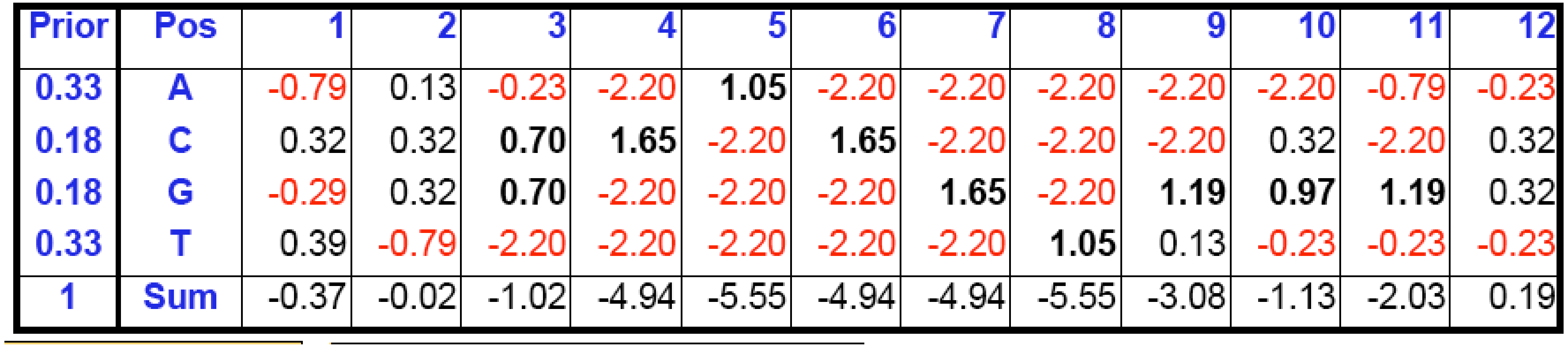
Another approach is based on the information content of the PSSM.

Example: Here is the matrix for the Pho4p binding site (*S. cerevisiae)*



Pos Base	1	2	3	4	5	6	7	8	9	10	11	12
A	1	3	2	0	8	0	0	0	0	0	1	2
C	2	2	3	8	0	8	0	0	0	2	0	2
G	1	2	3	0	0	0	8	0	5	4	5	2
T	4	1	0	0	0	0	0	8	3	2	2	2
			V	C	A	C	G	T	K	B		

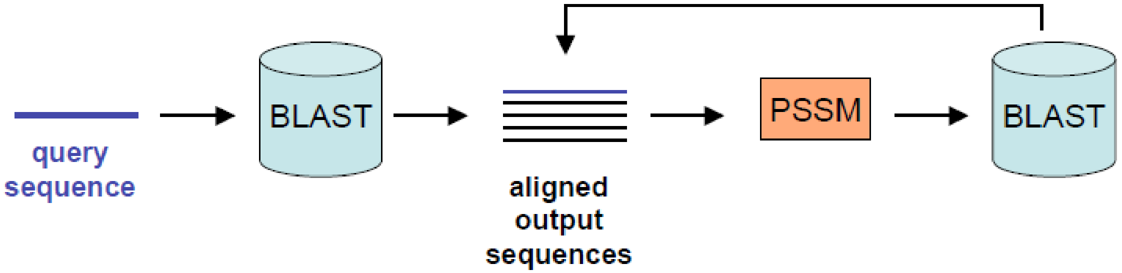
This PFM has been converted into a PSSM accounting for prior (background) probabilities (p_i) based on the GC content in *S. cerevisiae:*



Prior	Pos	1	2	3	4	5	6	7	8	9	10	11	12
0.33	A	-0.79	0.13	-0.23	-2.20	**1.05**	-2.20	-2.20	-2.20	-2.20	-2.20	-0.79	-0.23
0.18	C	0.32	0.32	**0.70**	**1.65**	-2.20	**1.65**	-2.20	-2.20	-2.20	0.32	-2.20	0.32
0.18	G	-0.29	0.32	**0.70**	-2.20	-2.20	-2.20	**1.65**	-2.20	**1.19**	**0.97**	**1.19**	0.32
0.33	T	0.39	-0.79	-2.20	-2.20	-2.20	-2.20	-2.20	**1.05**	0.13	-0.23	-0.23	-0.23
1	Sum	-0.37	-0.02	-1.02	-4.94	-5.55	-4.94	-4.94	-5.55	-3.08	-1.13	-2.03	0.19

PSI-BLAST : Position-Specific Iterated BLAST (Altschul et al, 1997)

- BLAST runs a first time in normal mode.
- Resulting sequences are aligned together (Multiple sequence alignment) and a PSSM is calculated.
- This PSSM is used to scan the database for new matches.
- Steps 2-3 can be iterated several times. This procedure typically converges after a few cycles.



Known problems:

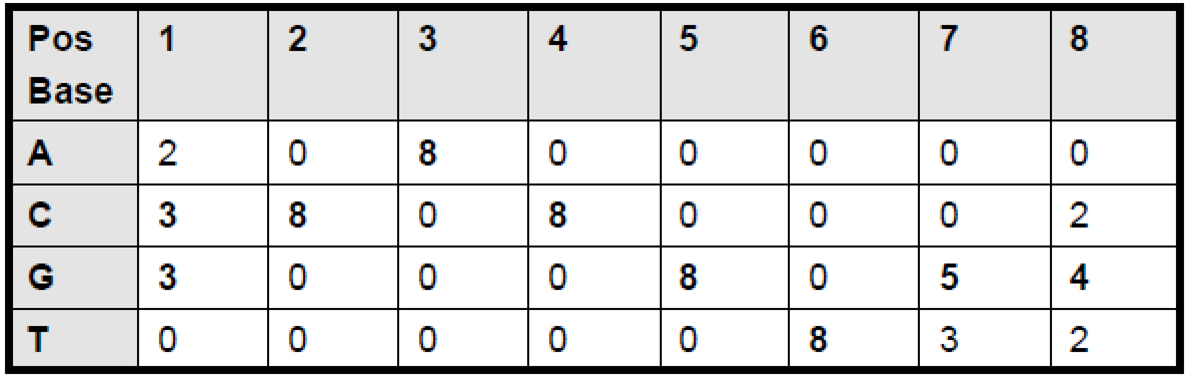
- Over-represented subfamilies may bias profile.
- Inappropriate E-value calculation may lead to the acceptance of false-positive matches
- Domain boundaries may not be properly identified during the first round.

Advice:

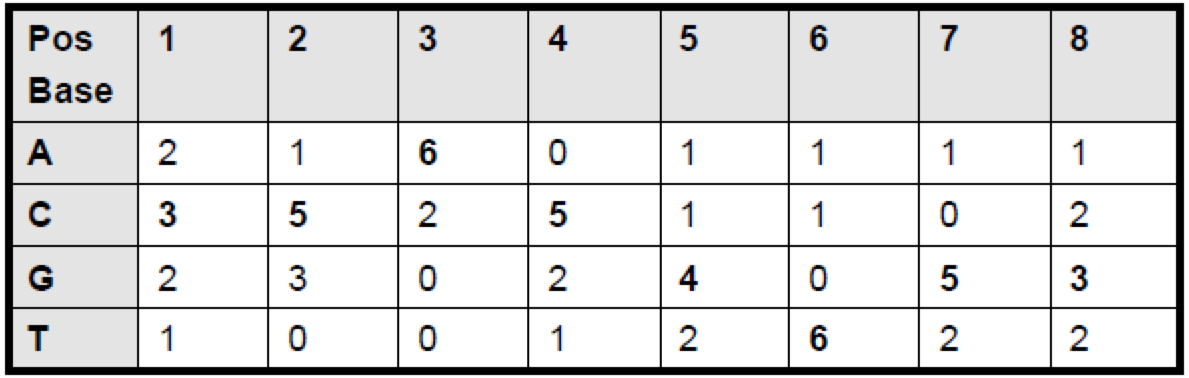
- The result of an iterative profile search may be verified by starting the procedure with a different seed.

How to measure the quality of a PSSM?

Which one of the following matrices is "the best"?



Pos Base	1	2	3	4	5	6	7	8
A	2	0	**8**	0	0	0	0	0
C	**3**	**8**	0	**8**	0	0	0	2
G	**3**	0	0	0	**8**	0	**5**	**4**
T	0	0	0	0	0	**8**	3	2



Pos Base	1	2	3	4	5	6	7	8
A	2	1	**6**	0	1	1	1	1
C	**3**	**5**	2	**5**	1	1	0	2
G	2	3	0	2	**4**	0	**5**	**3**
T	1	0	0	1	2	**6**	2	2

The **information content measures the conservation of the residues in** a given position. By definition the information content is:

$$\boxed{I_{i,j} = f'_{i,j} \log \frac{f'_{i,j}}{p_i}}$$

where $f'_{i,j}$ is the corrected frequency of residue i at position j.

$I_{ij} > 0$ when $f'_{ij} > p_i$ (i.e. when residue i is more frequent at position j than expected by chance)

$I_{ij} < 0$ when $f'_{ij} < p_i$

I_{ij} tends towards 0 when f'_{ij} -> 0. Indeed $\lim_{x \to 0} x \log x = 0$

$$\boxed{I_j = \sum_{i=1}^{A} I_{i,j}}$$

Information content of a column, where A = alphabet size

$$\boxed{I_{matrix} = \sum_{j=1}^{w} \sum_{i=1}^{A} I_{i,j}}$$

Information content of the matrix, where w = length of the matrix

4. Sequence logos

a

HEM13	CCCATTGTTCTC
HEM13	TTTCTGGTTCTC
HEM13	TCAATTGTTTAG
ANB1	CTCATTGTTGTC
ANB1	TCCATTGTTCTC
ANB1	CCTATTGTTCTC
ANB1	TCCATTGTTCGT
ROX1	CCAATTGTTTTG

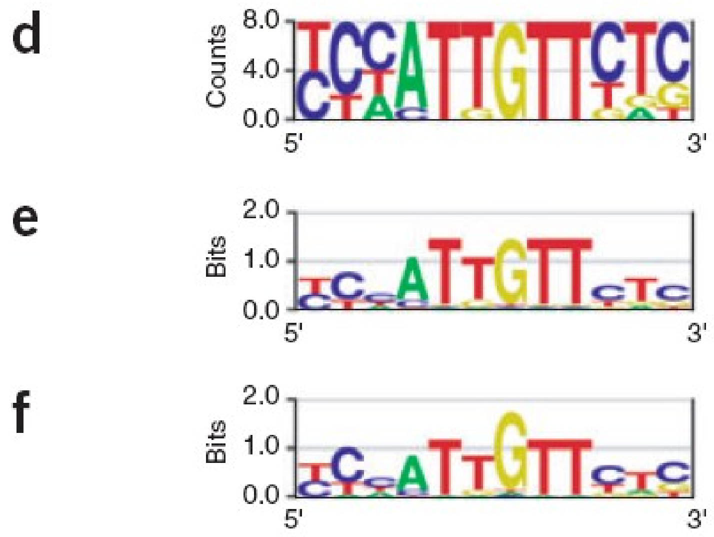
ROX1 binding sites

b YCHATTGTTCTC Consensus sequence

c

A	002700000010
C	464100000505
G	000001800112
T	422087088261

Position-specific frequency matrix



In (d), all the position have the same height. In (e) and (f), the size of the letters is scaled according to the information content. In (f), the information content is corrected to account for the background frequencies (observed in yeast). Because of the low GC content, the G in the middle is higher, reflecting the fact it carries more information than the *flanking A and T.*

Some limitations of the PSSM

It is difficult to recognize instances of the pattern that contain insertions or deletions. If a PSSM is designed to detect the motif GGCACGTGTA (and its variants), it will more likely fail to detect the GGCACCTGTGTA.

It cannot capture positional dependencies. Suppose, in a particular motif, we always see either RD or QH at positions j and j + 1, but never QD or RH. This is an example of a pattern that a PSSM can not represent.

PSSM are not well suited to represent variable length patterns.

This approach is not well suited for detecting sharp boundaries between two regions.

Thank You