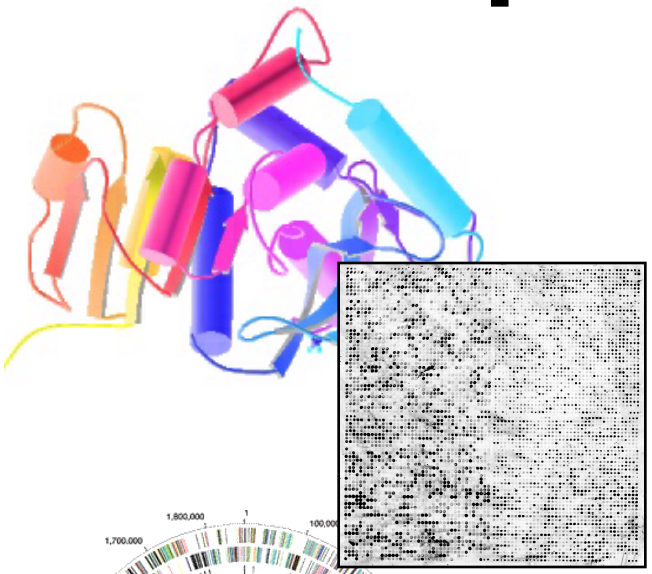
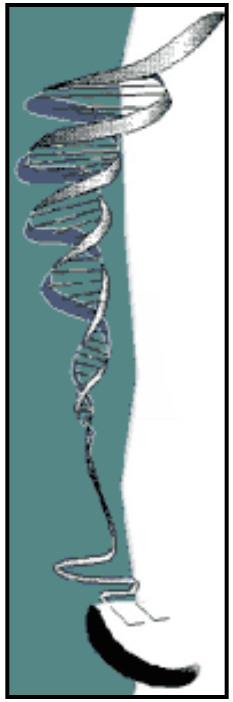
BIOINFORMATICS
Sequence to Structure





Mark Gerstein, Yale University
gersteinlab.org/courses/452
(Last edit in spring '11. Final version.)

Secondary Structure Prediction Overview

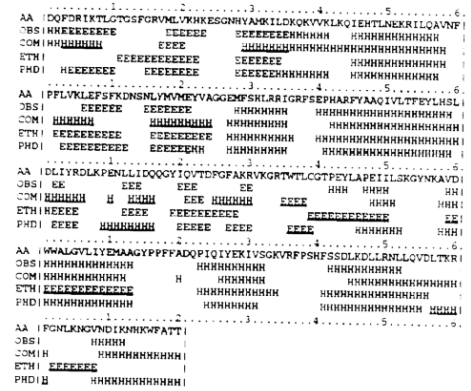
- Why interesting?
  - ◊ Not tremendous success, but many methods brought to bear.
  - ◊ What does difficulty tell about protein structure?

- Start with TM Prediction (Simpler)
- Basic GOR Sec. Struc. Prediction
- Better GOR
  - ◊ GOR III, IV, semi-parametric improvements, DSC
- Other Methods
  - ◊ NN, nearest nbr.

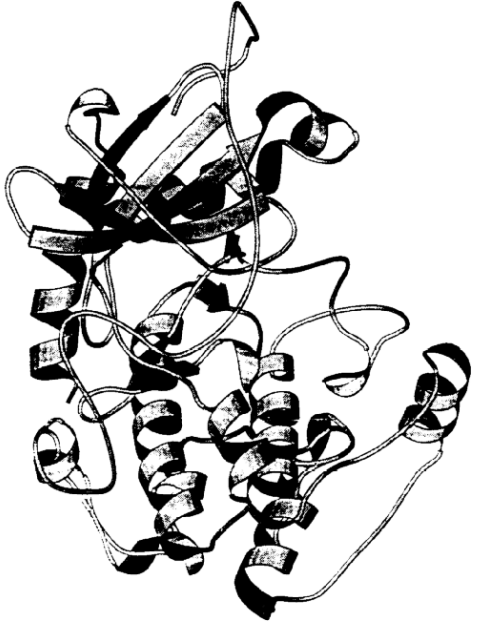
What secondary structure prediction tries to accomplish?

Credits: Rost et al. 1993;
Fasman & Gilbert, 1990

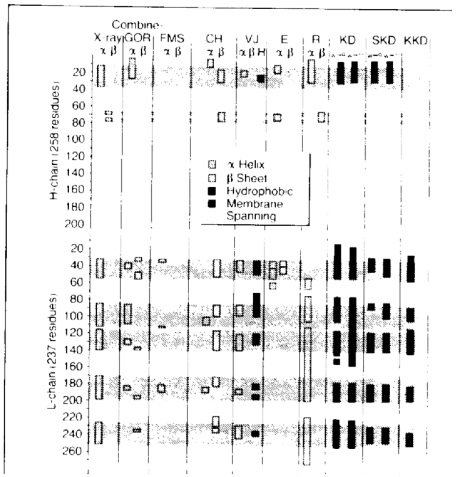
- Not Same as Tertiary Structure Prediction -- no coordinates
- Need torsion angles of terms + slight diff. in torsions of sec. str.



(a) Residue-by-residue comparison of experimentally observed (OBS) and predicted [COM[10], ETH[28], PHD (Ref. 35 and B. Rost and C. Sander, submitted)] structures of the catalytic subunit of the cAMP-dependent protein kinase (1cpk). 'AA' is the amino acid sequence taken from Protein Data Bank entry 1cpk (residues 27-287). Secondary structure: H = α-helix, E = β-sheet (extended), blank = loop. Predicted α-helices and β-strands that have insufficient overlap with an observed segment of the same type are underlined. Note the relatively good prediction of the location of segments for the ETH and PHD methods and overprediction of α-helices for the COM method.



(b) Ribbon view of the domain used in this blind test. The X-ray structure of catalytic subunit of the cAMP-dependent protein kinase. Drawn using Molscript[44].



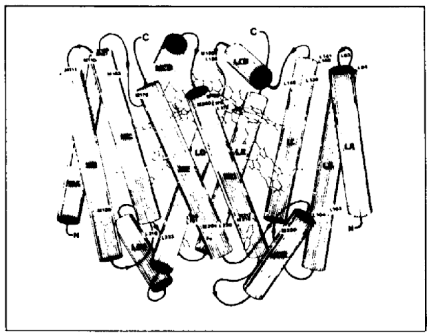


Figure 1
Column model for the core of the reaction center from *Rsp. viridis*. Reproduced, with permission, from Ref. 18.

```
Sequence   RPDFCLEPPYTGPCKARIIRYFYNAKAGLVQTFVYGGCRAKRNNFKSAEDAMRTCGGA
Structure  CCGGGGCCCCCCCCCCCCEEEEEEETTTTEEEEEEECCCCCTTTTBTTHHHHHHHHHCC
```

TM Helix Identification

The problem

Some TM scales:

GES

F	-3.7
M	-3.4
I	-3.1
L	-2.8
V	-2.6
C	-2.0
W	-1.9
A	-1.6
T	-1.2
G	-1.0
S	-0.6
P	+0.2
Y	+0.7
H	+3.0
Q	+4.1
N	+4.8
E	+8.2
K	+8.8
D	+9.2
R	+12.3

Goldman, Engleman, Steitz

KD

KD – Kyte Dolittle

I	4.5
V	4.2
L	3.8
F	2.8
C	2.5
M	1.9
A	1.8
G	-0.4
T	-0.7
W	-0.9
S	-0.8
Y	-1.3
P	-1.6
H	-3.2
E	-3.5
Q	-3.5
D	-3.5
N	-3.5
K	-3.9
R	-4.5

For instance, ΔG from transfer of a Phe amino acid from water to hexane

How to use GES to predict proteins

- Transmembrane segments can be identified by using the GES hydrophobicity scale (Engelman et al., 1986). The values from the scale for amino acids in a window of size 20 (the typical size of a transmembrane helix) were averaged and then compared against a cutoff of -1 kcal/mole. A value under this cutoff was taken to indicate the existence of a transmembrane helix.
- H-19(i) = [H(i-9)+H(i-8)+...+H(i) + H(i+1) + H(i+2) + . . . + H(i+9)] / 19

Graph showing Peaks in scales

Illustrations Adapted From: von Heijne, 1992; Smith notes, 1997

Cor

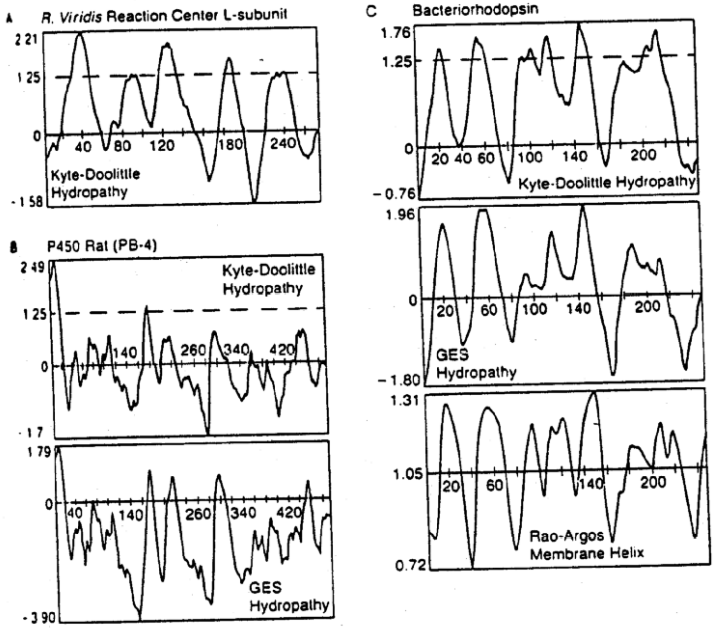
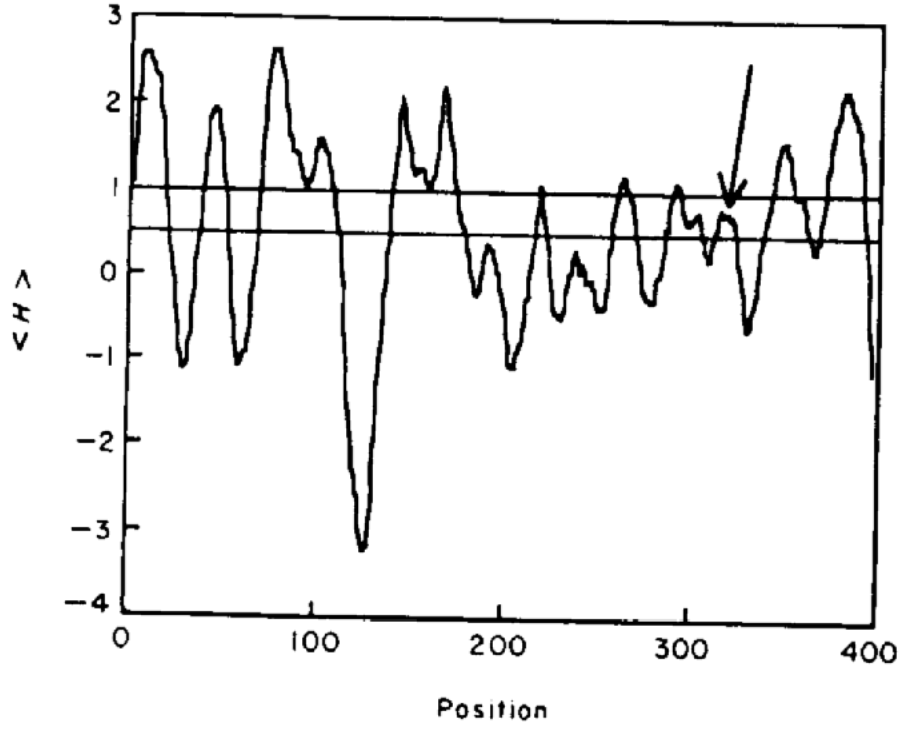
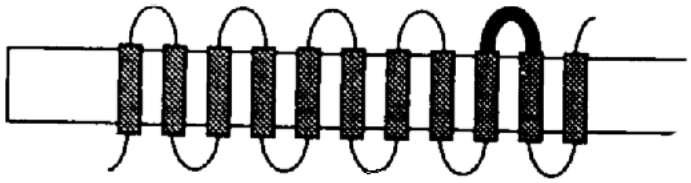


Figure 3.12. Representative profiles of three membrane proteins used to predict membrane-spanning helices. The amino acid scales of Kyte-Doolittle (804), Goldman–Engelman–Steitz (GES) (389), and Rao–Argos (1194) were used. A computer software package (SEQANAL) provided by Dr. A. Crofts (Univ. of Illinois) was used to generate these profiles. For comparative purposes, the Kyte–Doolittle and GES plots were obtained using a window of 19 residues and then smoothed using a second pass with a window of 7. The average value at each residue position is plotted as a function of residue number starting with the amino terminus on the left in each case. The values plotted for the Kyte–Doolittle and GES scales represent average hydropathy and transfer free energy per residue (kcal/mol). The Rao–Argos plot used a span of 7 residues and was smoothed used two additional passes with the same span of 7, as recommended by the authors. The scale values reflect the relative preference for being in a membrane-spanning helix. Note that the version of the GES algorithm which was used does not take into account possible ion pair formation. See text for details.





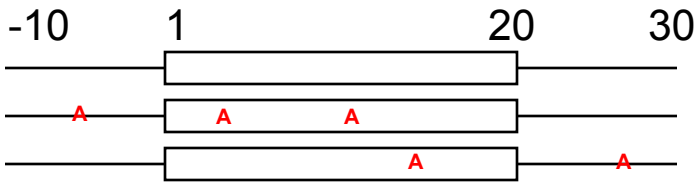
Ex. P(i,α) probability that residue i has secondary structure α

- Problem of DB Bias
- f(A) = frequency of residue A to have a TM-helical conf. in db
- f(A,i) = f(A) at position i in a particular sequence
- E(α)=statistical energy of helix over a window
- p(i, α) = probability that residue i is in a TM-helix

$$E_\alpha = \sum_i^N \ln f_\alpha^i$$

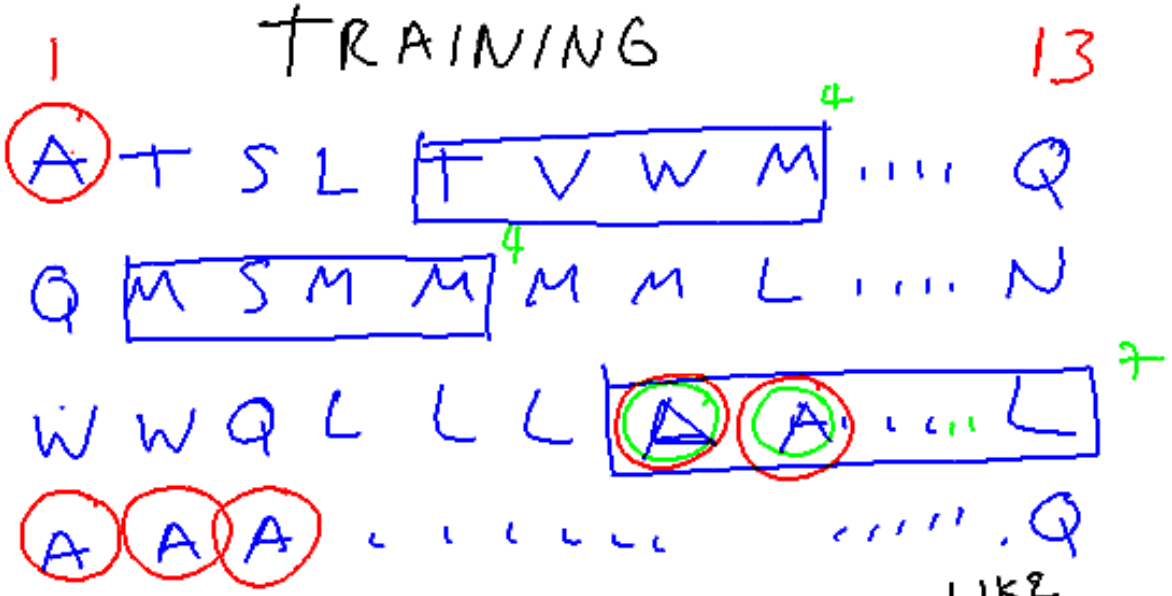
$$p_\alpha^i = \frac{e^{-Ea/RT}}{\sum_j^N e^{-E_j/RT}}$$

$F_{in\text{-}DB}(A) = 5/120$



$F_{in\text{-}TM}(A) = 3/60$

Example of Deriving a Scale from Frequencies



$$P(A) \text{ in } DB = \frac{6}{4 \times 13} = f_{DB}$$

$$P(A) \text{ in } HLX = \frac{2}{15} = f_{HLX}$$

LIKE GES

$$\sum_{\text{some}} \ln\left(\frac{f_{HLX}}{f_{DB}}\right)$$

$$\ln\left(\frac{f_{HLX}}{f_{\sim HLX}}\right)$$

Statistics Based Methods: Persson & Argos

- Propensity P(A) for amino acid A to be in the middle of a TM helix or near the edge of a TM helix

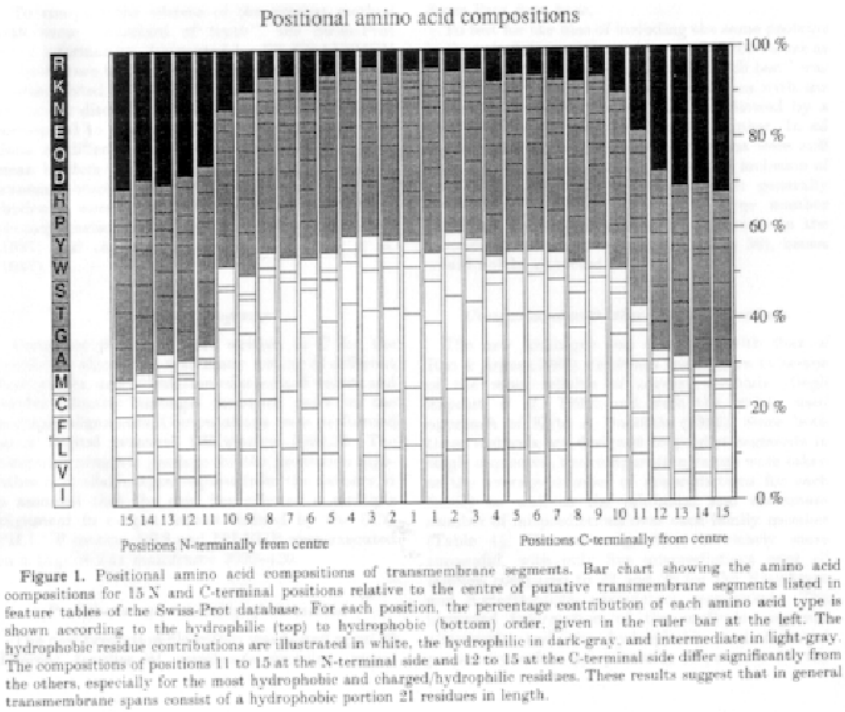
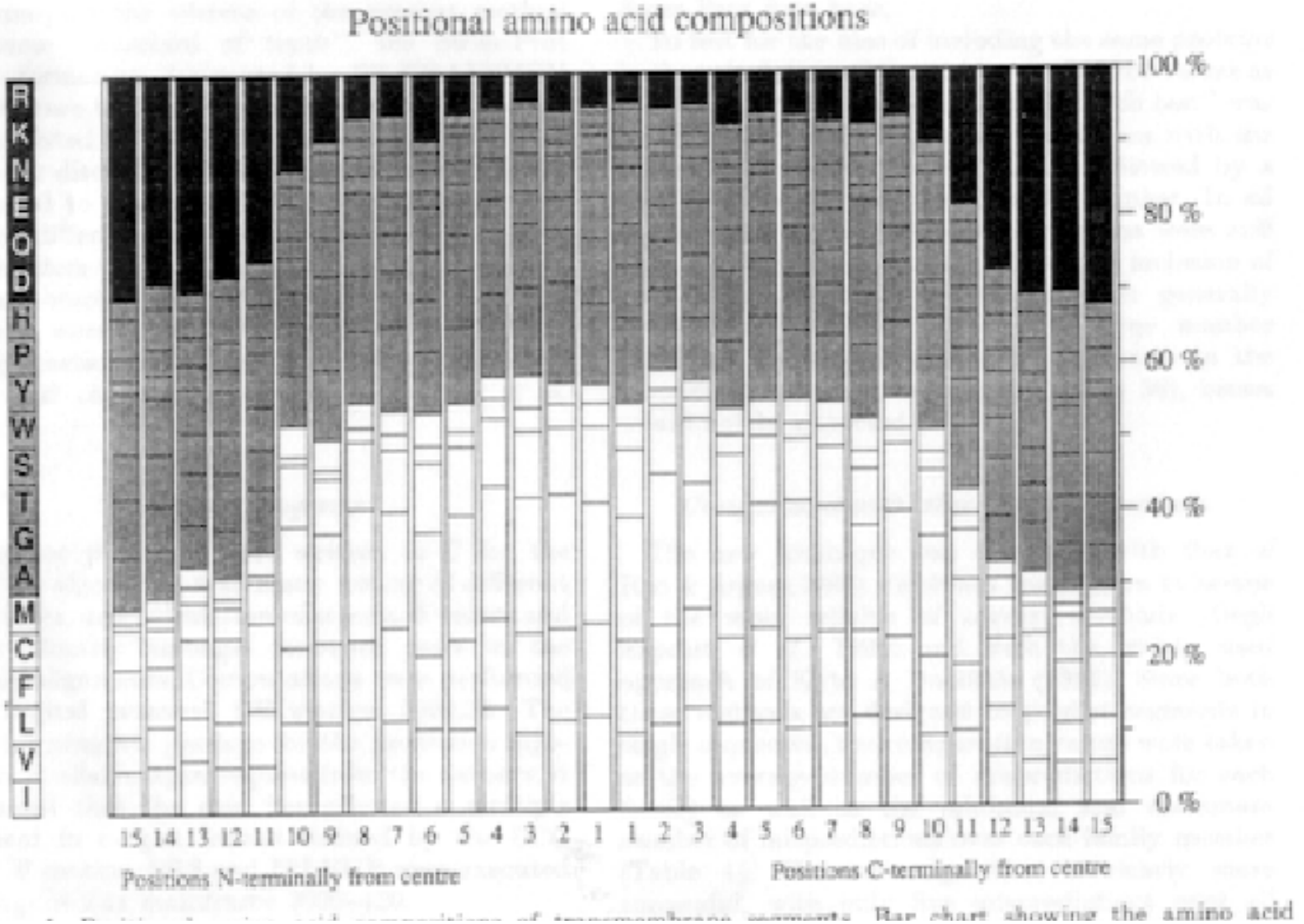


Figure 1. Positional amino acid compositions of transmembrane segments. Bar chart showing the amino acid compositions for 15 N- and C-terminal positions relative to the centre of putative transmembrane segments listed in feature tables of the Swiss-Prot database. For each position, the percentage contribution of each amino acid type is shown according to the hydrophilic (top) to hydrophobic (bottom) order, given in the ruler bar at the left. The hydrophobic residue contributions are illustrated in white, the hydrophilic in dark-gray, and intermediate in light-gray. The compositions of positions 11 to 15 at the N-terminal side and 12 to 15 at the C-terminal side differ significantly from the others, especially for the most hydrophobic and charged/hydrophilic residues. These results suggest that in general transmembrane spans consist of a hydrophobic portion 21 residues in length.

$$P(A) = \frac{\dfrac{n(A,\mathrm{TM})}{\sum_A n(A,\mathrm{TM})}}{\dfrac{n(A,\mathrm{everywhere})}{\sum_A n(A,\mathrm{everywhere})}}$$

Illustration Credits: Persson & Argos, 1994

$$P(A) = f_{TM}(A)/f_{SwissProt}(A)$$

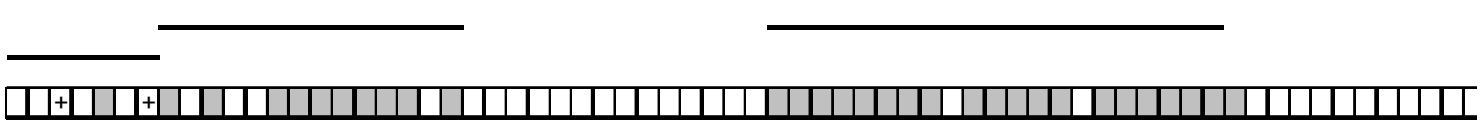


Extra

Scale Detail

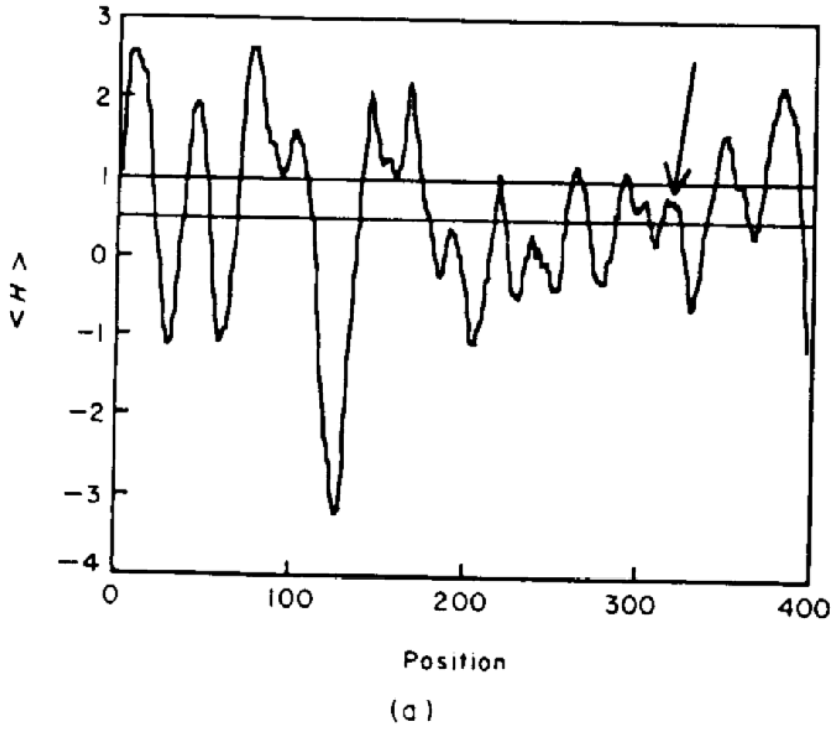
Add-ons ("hacks"): Removing Signal sequences

- Initial hydrophobic stretches corresponding to signal sequences for membrane insertion were excluded. (These have the pattern of a charged residue within the first 7, followed by a stretch of 14 with an average hydrophobicity under the cutoff).



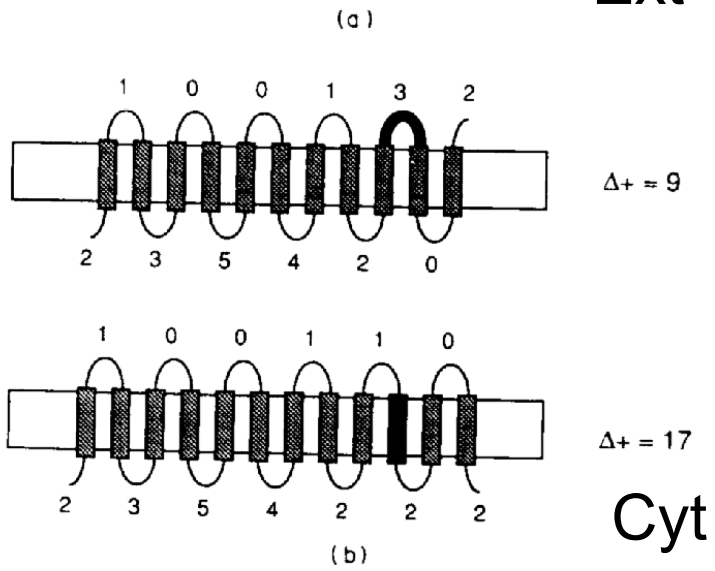
Add-ons: Charge on the Outside, Positive Inside Rule

- for marginal helices, decide on basis of R+K inside (cytoplasmic)



Credits: von Heijne, 1992

Ext



Cyt

Figure 4. (a) Hydrophobicity plot for the SecY protein. The upper and lower cutoffs are marked. A tentative transmembrane segment with a mean hydrophobicity falling between the 2 cutoffs is marked by an arrow. (b) Two possible topologies for the SecY protein based on the hydrophobicity plot. The putative transmembrane segment is shown in black. The number of Arg + Lys residues is shown next to each polar segment. Note that the correct alternative (bottom, including the putative transmembrane segment) has a much higher charge-bias than the incorrect one.

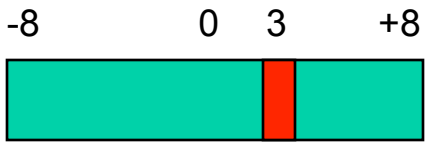
GOR

GOR: Simplifications

- For independent events just add up the information
- $I(S_j ; R_1, R_2, R_3,...R_{last})$ = Information that first through last residue of protein has on the conformation of residue j (S_j)
  - ◊ Could get this just from sequence sim. or if same struc. in DB (homology best way to predict sec. struc.!)
- Simplify using a 17 residue window: $I(S_j=H ; R[j-8], R[j-7], ..., R[j], ... R[j+8])$
- Difference of information for residue to be in helix relative to not: I(dSj;y) = I(Sj=H;y)-I(Sj=~H;y)
  - ◊ odds ratio: I(dSj;y)= ln P(Sj;y)/P(~Sj;y)
  - ◊ I determined by observing counts in the DB, essentially a lod value

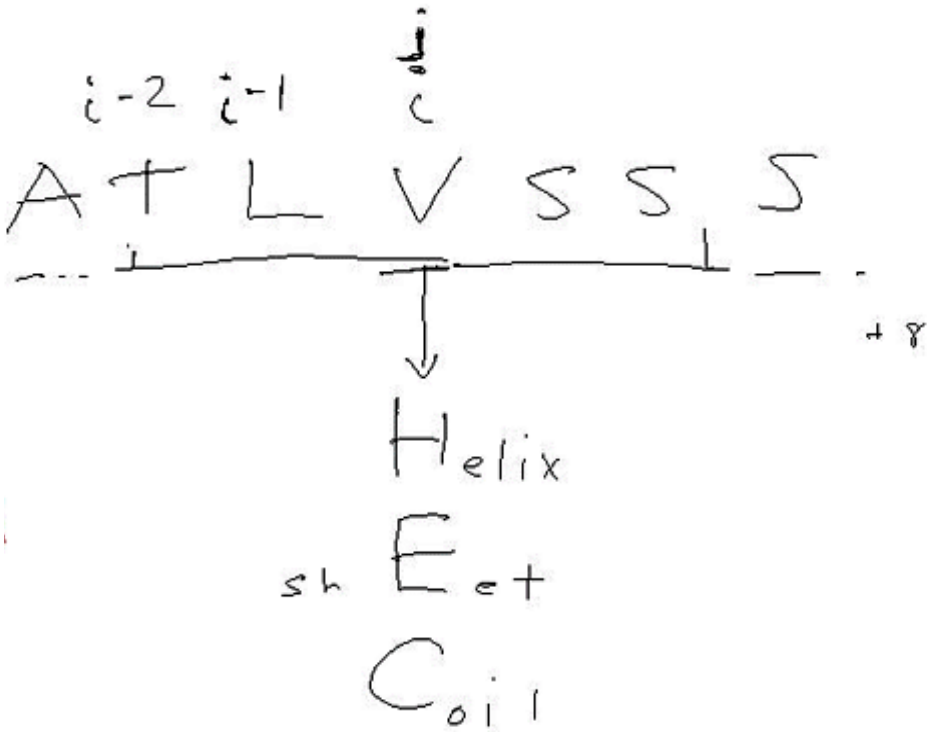
Basic GOR

- Pain & Robson, 1971;
  Garnier, Osguthorpe, Robson, 1978
- I ~ sum of I(Sj,R[j+m]) over 17 residue window centered on j and indexed by m
  - ◊ I(Sj,R[j+m]) = information that residue at position m in window has about conformation of protein at position j
  - ◊ 1020 bins=17*20*3
- In Words
  - ◊ Secondary structure prediction can be done using the GOR program (Garnier et al., 1996; Garnier et al., 1978; Gibrat et al., 1987). This is a well-established and commonly used method. It is statistically based so that the prediction for a particular residue (say Ala) to be in a given state (i.e. helix) is directly based on the frequency that this residue (and taking into account neighbors at +/- 1, +/- 2, and so forth) occurs in this state in a database of solved structures. Specifically, for version II of the GOR program (Garnier et al., 1978), the prediction for residue i is based on a window from i-8 to i+8 around i, and within this window, the 17 individual residue frequencies (singlets).



f(H,+3)/f(~H,+3)

The Secondary Structure Prediction Problem



INDEPENDENCE ASSUMPTI

"Grand Formula" $P(S_j = H \mid R_{-3} = A, R_{-2} = T, \ldots)$

GOR Simplification $P(S_j = H \mid R_{-3} = A)\ P(S_j = H \mid R_{-2} = T) \ldots$

GOR parameters

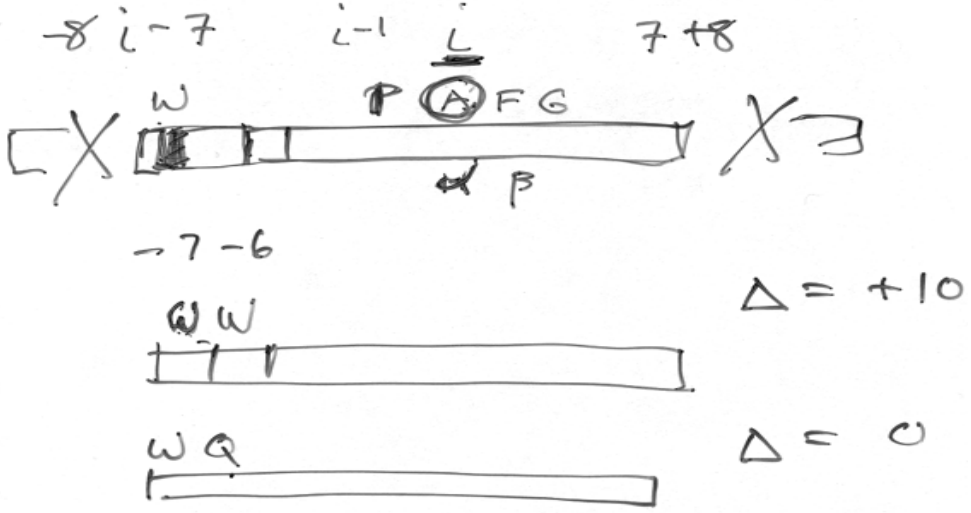
OBS =
F (residue "A" to be at window position j [e.g. =i-3] in a helix centered at position i)

EXP = F (residue "A" in the DB in general)

$$LOD = \ln \frac{OBS}{EXP}$$

$$\sum_{WINDOW} \varepsilon_{SCALE}(i)$$

$$\prod_{i} P_i(A)$$



Directional Information

helix

strand

coil

	i-8	i-7	i-6	i-5	i-4	i-3	i-2	i-1	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8
a	-12	-15	-12	-12	-17	-13	-25	-24	-32	-35	-32	-29	-24	-20	-12	-5	-6
c	36	26	41	50	45	31	29	19	7	5	27	29	38	48	41	45	59
d	-8	-10	-13	-8	-13	-10	12	25	50	43	39	27	7	-7	-4	-9	-5
e	-3	-11	-10	-11	-10	-7	-5	-23	-26	-23	-2	5	-1	-3	3	-5	-9
f	22	25	28	25	21	9	-23	-34	-49	-40	-29	-12	9	20	13	18	13
g	-3	-8	-18	-17	-7	2	26	68	97	58	19	-2	-18	-14	-18	-11	-11
h	15	9	-4	-7	8	-2	12	8	8	5	-4	1	-3	-5	-5	-10	-9
i	7	12	19	14	7	1	-21	-42	-66	-55	-26	-14	14	18	4	2	1
k	-12	-7	-10	-9	-1	5	11	5	0	9	5	-8	-20	-15	-7	-10	-12
l	2	8	11	11	11	2	-23	-42	-65	-63	-52	-39	-15	-11	-10	-6	0
m	11	14	4	3	-9	-16	-33	-52	-62	-77	-71	-54	-32	-7	3	9	9
n	-2	-8	-11	1	8	12	32	51	61	31	18	6	-6	-8	-4	2	2
p	4	8	4	-1	5	15	39	76	120	159	98	59	32	17	11	3	0
q	-1	-11	-12	-15	-17	-4	5	-5	-13	1	1	2	-2	-5	-1	-9	-20
r	-4	-9	-8	-10	-10	-13	-18	-16	-14	-9	-14	-16	-14	-11	-5	-3	-2
s	-3	-4	-4	-4	4	11	22	26	41	31	20	13	3	5	4	8	11
t	-5	-5	-5	-4	-7	-5	0	2	15	21	29	30	19	7	3	-4	-5
v	3	17	20	20	8	-2	-26	-46	-68	-51	-20	3	25	24	23	15	11
w	5	9	28	28	12	-16	-32	-46	-53	-38	-20	5	13	30	9	2	16
y	10	7	12	7	6	3	7	-1	-31	-14	-11	11	13	1	3	12	15

Credits: King & Sternberg, 1996

Table 3. *Directional informational parameters: I(Sj = x:x': Rj + m) for residue position versus residue type for α-helices*[a]

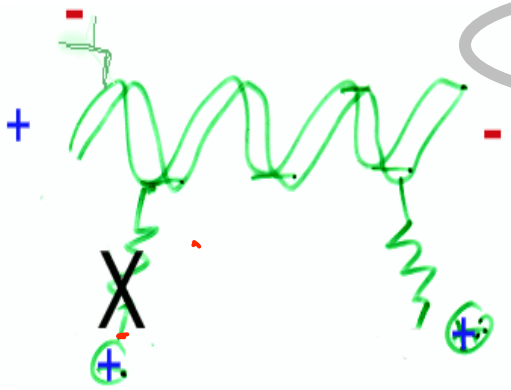
	i-8	i-7	i-6	i-5	i-4	i-3	i-2	i-1	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8
a	19	21	22	24	34	36	44	47	60	60	53	50	44	40	31	23	24
c	-47	-45	-44	-47	-44	-36	-44	-55	-56	-58	-54	-55	-58	-58	-59	-53	-66
d	14	15	14	15	17	21	15	17	-7	-11	-31	-42	-28	-12	-8	1	-5
e	14	16	15	20	26	27	34	52	62	57	32	15	19	12	6	7	9
f	-19	-14	-10	-4	-2	-1	6	-1	10	10	12	12	-4	-5	2	0	2
g	5	2	1	-5	-22	-30	-50	-70	-92	-52	-28	-21	-13	-17	-8	-6	-6
h	-22	-20	-9	-10	-19	-10	-14	-7	-11	-4	0	-3	-2	2	6	11	12
i	7	7	0	0	1	1	2	-5	1	2	1	7	-6	-3	10	8	6
k	-2	-1	-1	-1	-6	-9	-6	5	17	17	21	27	35	33	21	22	23
l	0	-1	0	6	9	16	30	33	45	47	51	53	37	32	30	25	18
m	4	3	15	23	30	30	39	36	45	54	57	53	44	29	30	14	1
n	2	3	2	-5	-9	-10	-16	-17	-31	-16	-17	-16	-9	-8	-9	-10	-5
p	-12	-15	-14	-19	-23	-25	-30	-48	-82	-195	-145	-104	-67	-49	-43	-33	-17
q	-4	3	7	4	13	8	10	24	35	32	31	21	18	18	9	8	6
r	5	3	6	13	7	13	19	27	34	32	36	41	33	29	23	21	18
s	-10	-7	-10	-10	-16	-17	-25	-21	-39	-35	-39	-41	-32	-35	-34	-35	-33
t	1	-1	-6	-8	-6	-11	-16	-25	-48	-47	-48	-46	-34	-31	-34	-26	-24
v	-5	-12	-13	-14	-13	-19	-17	-20	-15	-22	-22	-20	-26	-19	-15	-10	-5
w	0	-4	-12	-19	-7	14	16	12	18	17	12	8	1	-6	1	3	-13
y	-22	-19	-17	-20	-16	-21	-30	-32	-8	-10	-4	-12	-17	-9	-10	-14	-15

[a]Note that the convention used is the reverse of that adopted by (Garnier et al., 1978), for example the first entry for alanine at position j-8 is the amount of information that an alanine residue eight positions toward the N terminus has for predicting an α-helix

Table 4. *Directional informational parameters for residue position versus residue type for β-strands*

	i-8	i-7	i-6	i-5	i-4	i-3	i-2	i-1	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8
a	-8	-7	-13	-17	-23	-33	-26	-32	-43	-37	-30	-30	-26	-27	-26	-25	-25
c	3	13	-9	-20	-15	-3	9	33	47	51	21	19	9	-5	7	-5	-14
d	-7	-5	0	-9	-4	-14	-42	-73	-83	-59	-21	10	22	24	16	11	13
e	-14	-5	-5	-11	-21	-27	-45	-44	-57	-54	-46	-29	-25	-12	-12	-2	0
f	-9	-20	-32	-34	-30	-12	24	44	49	39	24	2	-9	-23	-24	-29	-23
g	-3	9	24	29	34	30	18	-23	-48	-27	6	27	39	38	33	23	23
h	6	11	17	22	12	16	0	-2	3	-2	5	3	8	4	-1	1	-3
i	-21	-30	-31	-21	-12	-3	26	58	76	64	33	11	-14	-24	-20	-14	-11
k	20	12	15	14	8	4	-8	-14	-25	-40	-39	-27	-20	-24	-20	-15	-15
l	-2	-10	-18	-27	-30	-27	-6	15	27	21	2	-19	-31	-29	-28	-26	-25
m	-22	-26	-29	-40	-31	-17	-7	23	24	28	17	2	-15	-31	-53	-36	-16
n	1	8	14	5	0	-6	-30	-65	-62	-28	-6	11	18	21	16	10	3
p	9	7	12	24	20	8	-22	-65	-108	-64	-8	17	25	30	32	31	21
q	6	12	8	16	8	-5	-22	-27	-30	-52	-49	-34	-22	-17	-9	2	20
r	0	8	3	-3	5	2	1	-14	-26	-32	-30	-35	-27	-26	-25	-25	-21
s	16	14	17	19	14	5	-3	-13	-15	-4	15	27	32	32	31	28	21
t	6	8	14	15	16	21	19	25	31	22	13	9	12	25	34	34	34
v	1	-11	-15	-11	4	25	51	75	91	81	49	19	-6	-12	-16	-11	-11
w	-8	-8	-28	-19	-9	5	23	44	45	30	13	-18	-22	-40	-15	-7	-9
y	13	13	4	14	12	20	24	37	48	31	20	-1	2	11	7	0	-4

Types of Residues



Credits: King & Sternberg, 1996

Table 3. *Directional informational parameters: I(Sj = x::x'; Rj + m) for residue position versus residue type for α-helices*[a]

	i-8	i-7	i-6	i-5	i-4	i-3	i-2	i-1	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8
a	19	21	22	24	34	36	44	47	60	60	53	50	44	40	31	23	24
c	-47	-45	-44	-47	-44	-36	-44	-56	-56	-58	-54	-55	-58	-58	-59	-53	-66
d	14	15	14	15	17	21	15	1	-7	-11	-31	-42	-28	-12	-8	1	-5
e	14	16	15	20	26	27	34	52	62	57	32	15	19	12	6	7	9
f	-19	-14	-10	-4	-2	-1	6	-1	10	10	12	12	-4	-5	2	0	2
g	5	2	1	-5	-22	-30	-50	-70	-92	-52	-28	-21	-13	-17	-8	-6	-6
h	-22	-20	-9	10	-19	-10	-14	-7	-11	-4	0	-3	-2	2	6	11	12
i	7	7	0	0	1	1	2	-5	1	2	1	7	8	-3	10	8	6
k	-2	-1	-1	-1	-6	-9	-6	5	17	17	21	27	35	33	21	22	23
l	0	-1	0	6	9	16	30	33	45	47	51	53	37	32	30	25	18
m	4	3	15	23	30	30	39	36	45	54	57	53	44	29	30	14	1
n	2	3	2	-5	-9	-10	-16	-17	-31	-15	-17	-16	-9	-8	-9	-10	-5
p	-12	-15	-14	-19	-23	-25	-3	-48	-82	-195	-145	-104	-67	-49	-43	-33	-17
q	-4	3	7	4	13	8	10	24	35	32	31	21	18	18	9	8	6
r	5	3	6	13	7	13	19	27	34	32	36	41	33	29	23	21	18
s	-10	-7	-10	-10	-16	-17	-25	-21	-39	-35	-39	-41	-32	-35	-34	-35	-33
t	1	-1	-6	-8	-6	-11	-16	-25	-48	-47	-48	-46	-34	-31	-34	-26	-24
v	-5	-12	-13	-14	-13	-19	-17	-20	-15	-22	-22	-20	-26	-19	-15	-10	-5
w	0	-4	-12	-19	-7	14	16	12	18	17	12	8	1	-6	1	3	-13
y	-22	-19	-17	-20	-16	-21	-30	-32	-8	-10	-4	-12	-17	-9	-10	-14	-15

[a]Note that the convention used is the reverse of that adopted by (Garnier et al., 1978), for example the first entry for alanine at position j-8 is the amount of information that an alanine residue eight positions toward the N terminus has for predicting an α-helix at position j.

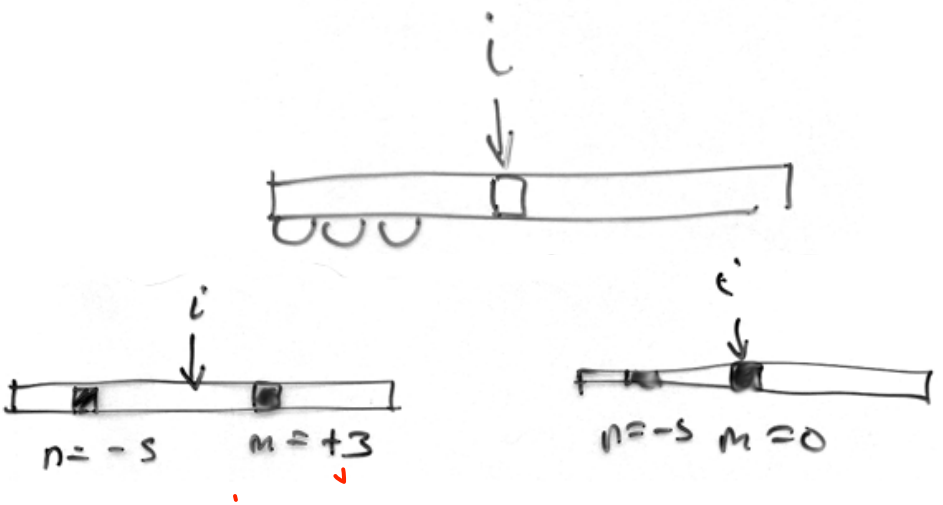
- Group I favorable residues and Group II unfavorable one:
- A, E, L -> H; V, I, Y, W, C -> E; G, N, D, S -> C
- P complex; largest effect on proceeding residue
- Some residues favorable at only one terminus (K)

Updated GOR ("IV")

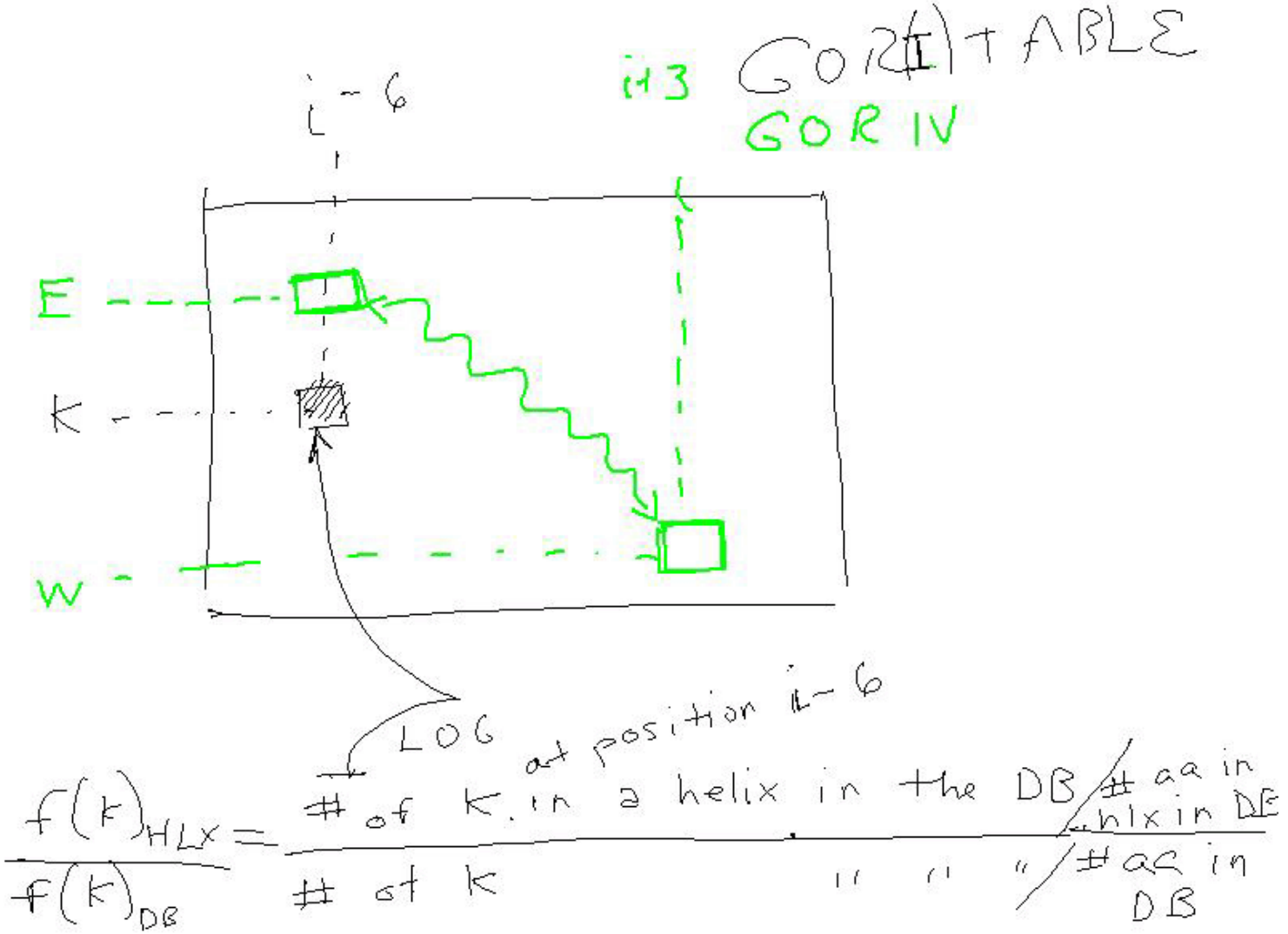
- I(Sj; R[j+m], R[j+n]) = the frequencies of all 136 (=16*17/2) possible di-residue pairs (doublets) in the window.
  - ◊ 20*20*3*16*17/2=163200 pairs
- Parameter Explosion Problem: 1000 dom. struc. * 100 res./dom. = 100k counts, over how many bins
- Dummy counts for low values (Bayes)

All Singletons in 17 residue window

All Pairs



How to calculate an entry in the simple GOR tables and a comparison to updated GOR (I vs IV)



Spectrum of calculations

Simple - 20 values at position i

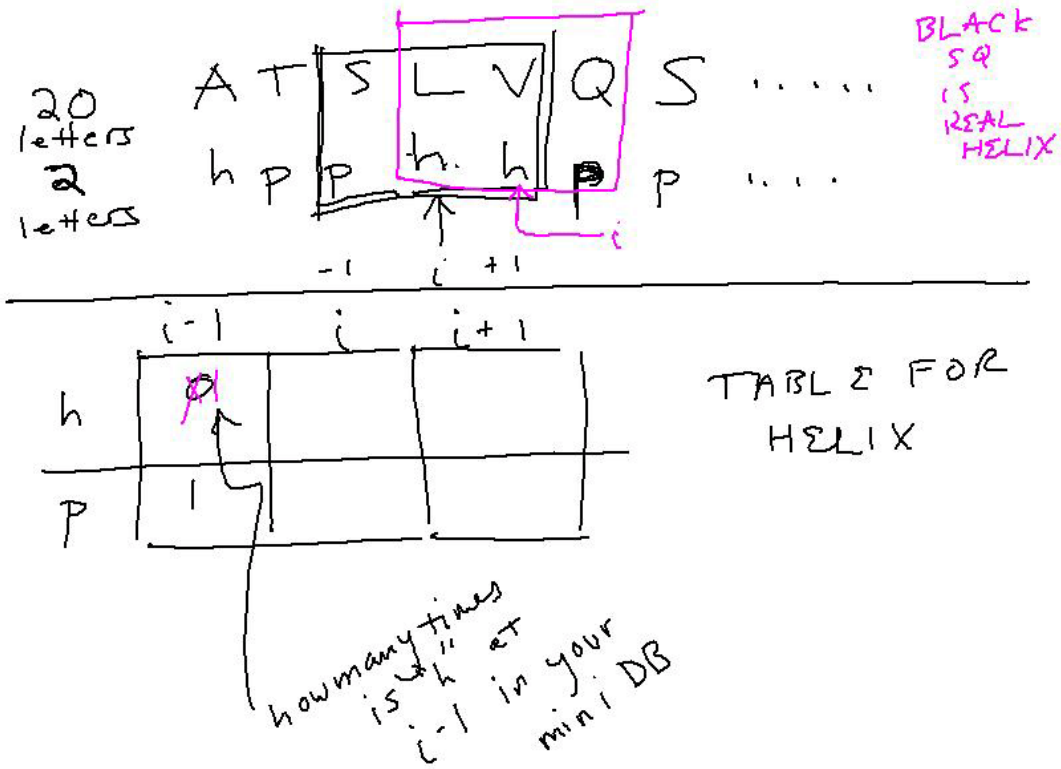
Simple GOR - ~1000 values within 17res window at i

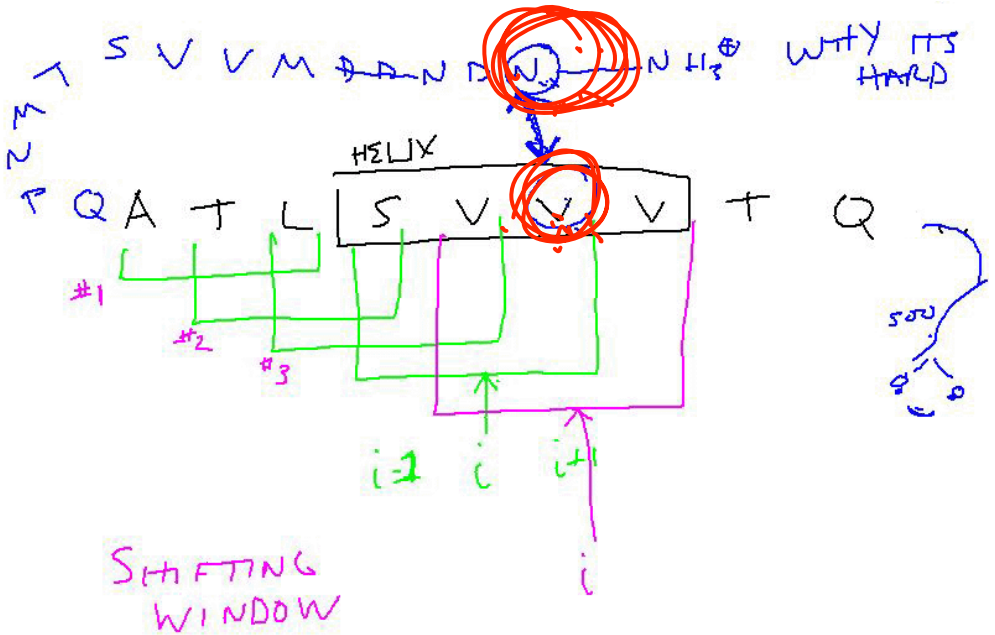
Updated GOR ~ 160K, all pairs within the window

(bin = how many times do I have a helix at i with A at position m=5 and V at position n=-4)

GOR-2010 - bigger window, triplets

GOR - 5000 -- all 15mer words, 20^15





An example of mini-GOR

Also, why secondary structure prediction is so hard

Assessment

- Q3 + other assess, 3x3
- Q3 = total number of residues predicted correctly over total number of residues (PPV)
- GOR gets 65%
  - ◊ sum of diagonal over total number of residue -- (14K+5K+21K)/ 64K
- Under predict strands & to a lesser degree, helices: 5.9 v 4.1, 10.9 v 10.6

TABLE II
GLOBAL RESULTS FOR DATABASE PREDICTION

	Observed			
	H	E	C	Total
Predicted				
H	14,460	3094	4790	22,344
E	1124	4965	2089	8178
C	6002	5546	21,496	33,044
Total	21,586	13,605	28,375	63,566
Q_{prd} [a]	64.7	60.7	65.1	
Q_{obs} [b]	67.0	36.5	75.8	
Q_3 [c] = 64.4%				

[a] Number of correctly predicted residues/number of predicted residues.
[b] Number of correctly predicted residues/number of observed residues
[c] Total number of correctly predicted residues/total number of residues.

Credits: Garnier et al., 1996

```
AASDTLVVIPWERE Input Seq
HHHHHEEEECCCHH Pred.
hhhheeeeeeeech Gold Std.
```

More Types of Secondary Structure Prediction Methods

- Parametric Statistical
  - ◊ struc. = explicit numerical func. of the data (GOR)
- Non-parametric
  - ◊ struc. = NON- explicit numerical func. of the data
  - ◊ generalize Neural Net, seq patterns, nearest nbr, &c.
- Semi-parametric: combine both
- single sequence
- multi sequence
  - ◊ with or without multiple-alignment

GOR Semi-parametric Improvements

- Filtering GOR t regularize

$[\neg a, c, \neg c, \mathbf{a}, a, c, \neg a] \rightarrow \mathbf{c}$

$[\neg a, \neg a, c, \mathbf{b}, *, \neg b] \rightarrow \mathbf{c}$

$[\neg a, c, c, \mathbf{a}, a, \neg b, \neg a] \rightarrow \mathbf{c}$

$[\neg a, *, *, \mathbf{a}, b] \rightarrow \mathbf{b}$

$[a, c, *, \mathbf{a}, a, a, \neg a] \rightarrow \mathbf{c}$

$[\neg a, *, *, \mathbf{a}, c] \rightarrow \mathbf{c}$

$[*, c, *, \mathbf{a}, a, b, \neg a] \rightarrow \mathbf{c}$

$[a, *, *, \mathbf{a}, c, *, \neg c] \rightarrow \mathbf{c}$

$[c, b, t\ \mathbf{a}, a, *, a] \rightarrow \mathbf{b}$

$[\neg a, \neg a, \mathbf{a}, a, c, \neg a] \rightarrow \mathbf{c}$

$[c, *\ \mathbf{a}, a, \neg a, a] \rightarrow \mathbf{c}$

a = α-helix, b β-strand, c = coil, * = vildcard (α-helix or β-strand or coil) $\neg$ = not.

If the pattern (ı the left is met in a predict on, then the secondary structure in bold on the left is rewritten as the secondary structure on the right of the rule. For example:

$[b, b, b, \mathbf{a}, c] \rightarrow [b, b, b, \mathbf{c}, c]$

$[b, b, c, \mathbf{a}, c] \rightarrow [b, b, c, \mathbf{c}, c]$

$[b, b, b, \mathbf{a}, b, b, b] \rightarrow [b, b, b, \mathbf{b}, b, b, b].$

Illustration Credits: King & Sternberg, 1996

DSC -- an improvement on GOR

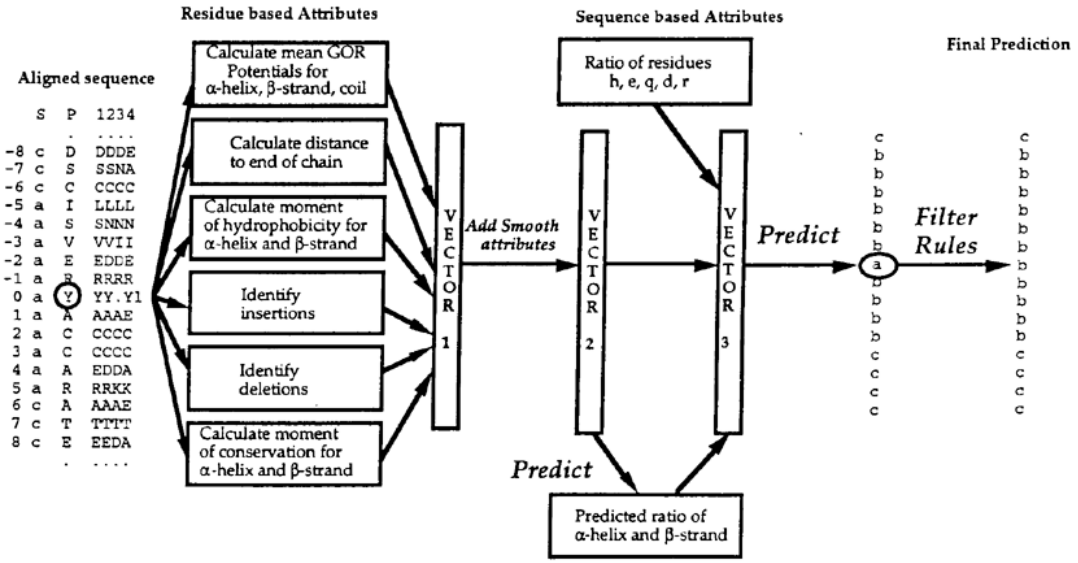
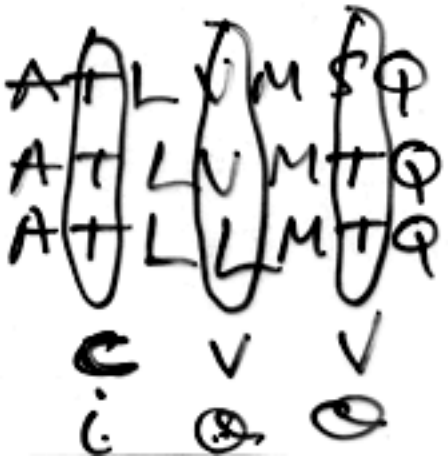


Fig. 1. DSC prediction method. For the aligned sequence: S is the observed secondary structure of the primary sequence, P. The residue at position 0 is predicted (circled).

- GOR parms
- + simple linear discriminant analysis on:
  - ◊ dist from C-term, N-term
  - ◊ insertions/deletes
  - ◊ overall composition
  - ◊ hydrophobic moments
  - ◊ autocorrelate: helices
  - ◊ conservation moment

Illustration Credits: King & Sternberg, 1996

Conservation, k-nn



Patterns of Conservation

k-nearest neighbors

Neural Networks

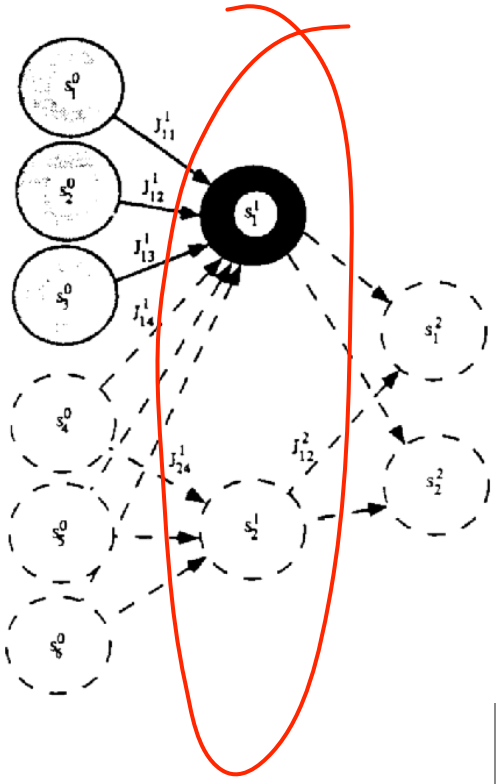
Figure 1. Function of a perceptron, the simplest neural network. A simple perceptron has only 1 output unit (black). Each of the left nodes receives a certain input signal (e.g. binary, i.e. =0 or 1). All units are connected to the output node by the junctions J^1, with e.g. J^1_{1j} connecting input unit j with output unit 1. The contribution of each left node (e.g. the jth) to the signal arriving at the right one is a product of the strength of the junction connecting the 2 units, and the input: e.g. $J^1_{1j}s^0_j$. All products (here 3) are summed by the right node (here s^1_1). This sum is then evaluated by a non-linear trigger function. The resulting map of the sum onto an interval between 0 and 1 is the actual output of the network. The broken-line nodes show a potential extension of the perceptron to a 2-layered feed-forward network. Stippled circles, input units, signal = 1 or 0. Black circle, output unit. Step 1, the input to this unit is summed according to:

$$h^1_i = \sum_{j=1}^{N^0+1} J_{ij}s^0_j \quad \text{(here, } i=1\text{)}.$$

Step 2, the output from this unit is computed by a sigmoid trigger function:

$$s^1_i = \frac{1}{1+\exp(-h^1_i)}.$$

Broken-line circles, the potential extension to a 2-layered feed-forward network.



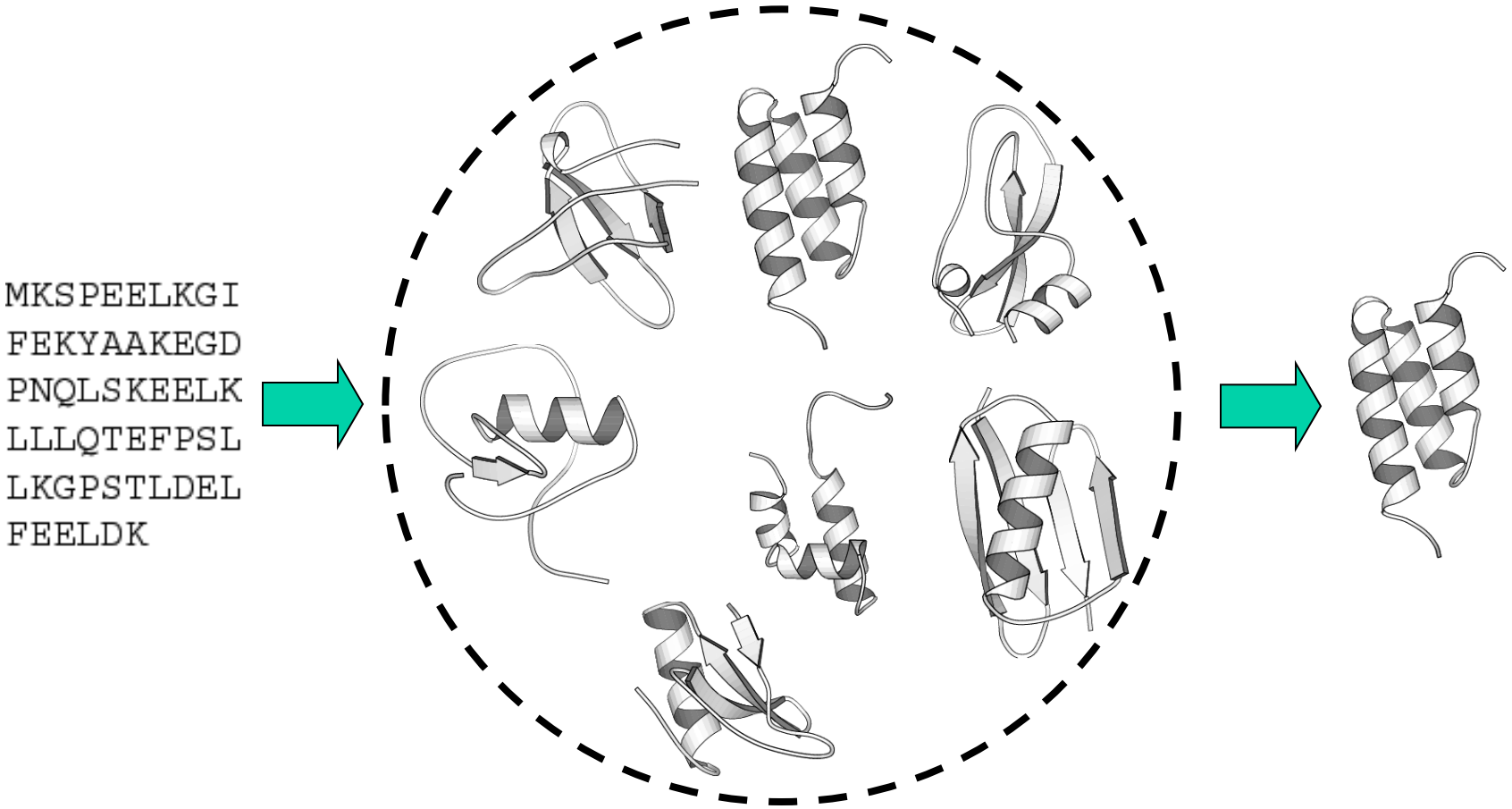
- Somehow generalize and learn patterns
- Black Box
- Perceptron (above) is Simplest network
  - ◊ Multiply junction * input, sum, and threshold

Illustration Credits: Rost & Sander, 1993

Yet more methods….

- struc class predict
  - ◊ Vect dist. between composition vectors
- threading via pair pot
- Distant seq comparison
- ab initio from md
- ab initio from pair pot.

Fold recognition



Query sequence Library of known folds Best-fit fold

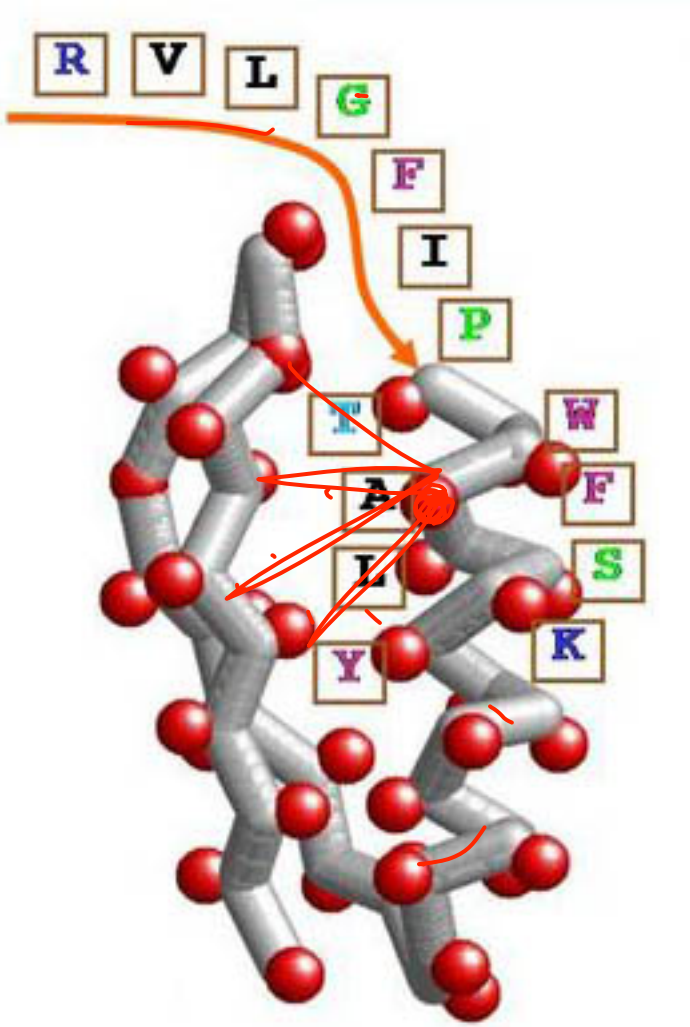
Why fold recognition?

- Structure prediction made easier by sampling 1,000~10,000 folds, rather than $>4^{100}$ possible conformations
- Practical importance: fold assignment in genomes
- Fold recognition can be done using sequence-based (BLAST, HMM, profile alignment) or structure-based methods (threading)

Fold recognition by threading

- Input: A query sequence, a fold library
- For each fold template in the library:
  - ◊ Generate alignments between the query sequence and the fold template
  - ◊ Evaluate alignments; choose the best one
- Do this for all folds, choose the best fold

What is threading



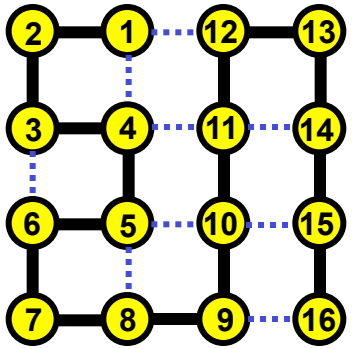
- Query sequence:

  RVLGFIPTWFALSKY

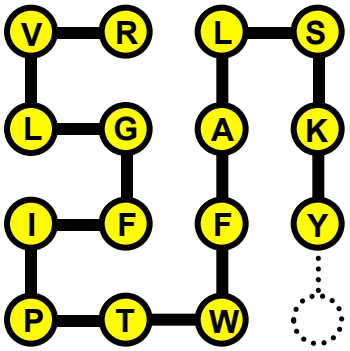
- Thread the sequence onto the fold template
- Use structural properties to evaluate the fit
  - ◊ Environment
  - ◊ Pairwise interactions

Align sequence to fold: an example

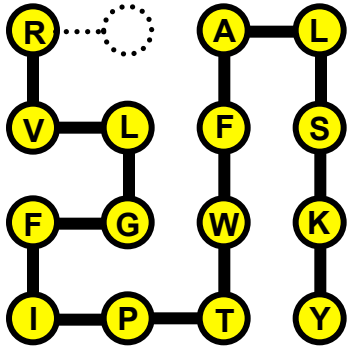
- Align: `RVLGFIPTWFALSKY` to:



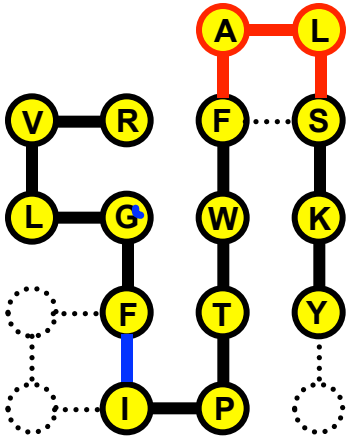
Many possible alignments:




```
1234567890123456
RVLGFIPTWFALSKY-
```




```
1234567890123456
-RVLGFIPTWFALSKY
```




```
123456789012--3456
RVLGF--IPTWFALSKY-
```

Deletion Insertion

Evaluate alignments using threading energy function

- $E_{total} = E_{env} + E_{pair} + E_{gap}$
- E_{env}: Total environment energies. Measures compatibility of a residue and its corresponding 3D environment (secondary structure, solvent accessibility)
- E_{pair}: Total pairwise energies. Measures interaction between spatially close residues
- E_{gap}: Gap opening and extension penalities

Relationship to Generalized Similarity Matrix

- PAM(A,V) = 0.5
  - ◊ Applies at every position
- S(aa @ i, aa @ J)
  - ◊ Specific Matrix for each pair of residues
    i in protein 1 and
    J in protein 2
  - ◊ Example is Y near N-term. matches any C-term. residue (Y at J=2)
- S(i,J)
  - ◊ Doesn't need to depend on a.a. identities at all!
  - ◊ Just need to make up a score for matching residue i in protein 1 with residue J in protein 2

i ⇨

		1	2	3	4	5	6	7	8	9	10	11	12	13
		A	B	C	N	Y	R	Q	C	L	C	R	P	M
1	A	1												
2	**Y**					1			5	5	5	5	5	5
3	C			1					1		1			
4	Y					1								
5	N				1									
6	R						1					1		
7	C			1					1		1			
8	K													
9	C			1					1		1			
10	R						1					1		
11	B		1											
12	P												1	

J ⇩

Find the best alignment

- NP-hard problem; needs approximation
- Dynamic programming and the "frozen approximation"
  - ◊ Approximately calculate amino acid preferences for each residue position by fixing the interaction partners at that position
  - ◊ Find best alignment using dynamic programming
  - ◊ Update interaction partners for each position; repeat till convergence
- Other optimization techniques
  - ◊ Simulated annealing
  - ◊ Branch-and-bound, etc.

Using Dynamic Programming in Threading