From Sequences to Structures: A Computational Probability Approach Based on Percolation Theory

Alexey Nikolaev

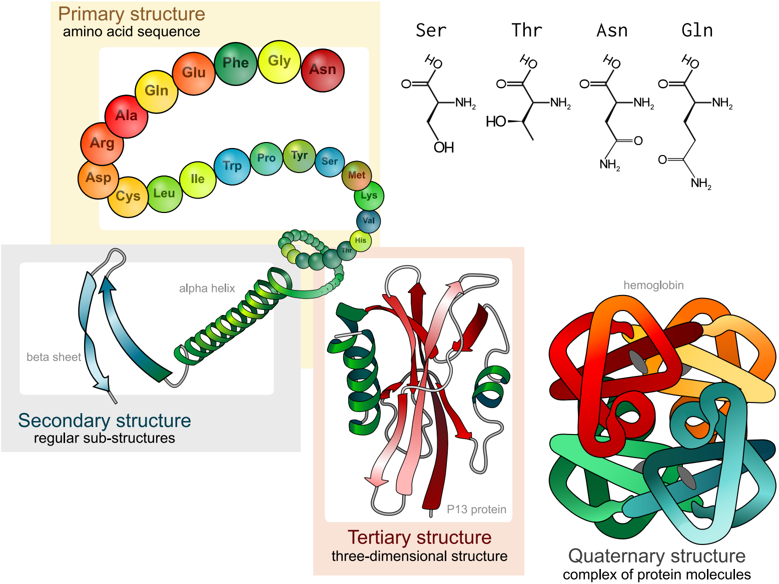
(The Graduate Center, CUNY)

Saad Mneimneh

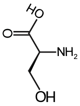
(Hunter College and the Graduate Center, CUNY)

January 8, 2013

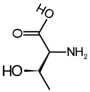
Protein Structure



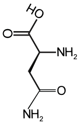
Ser



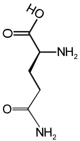
Thr



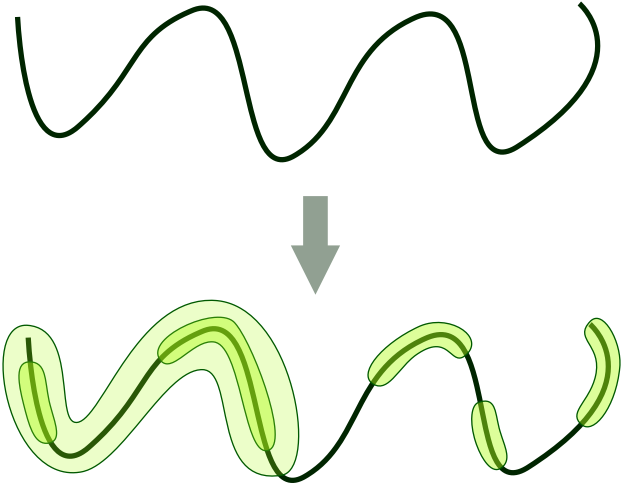
Asn



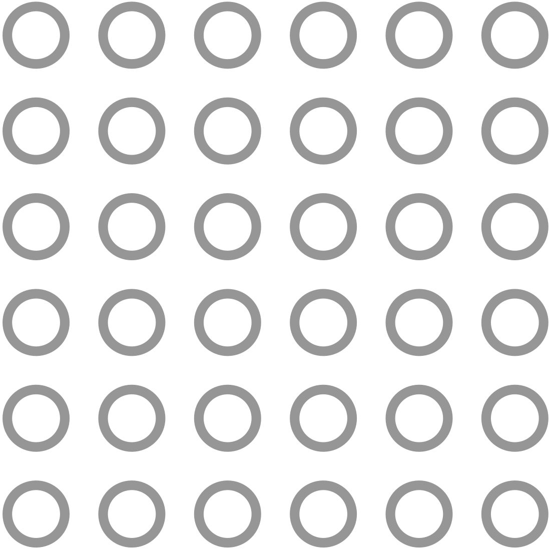
Gln



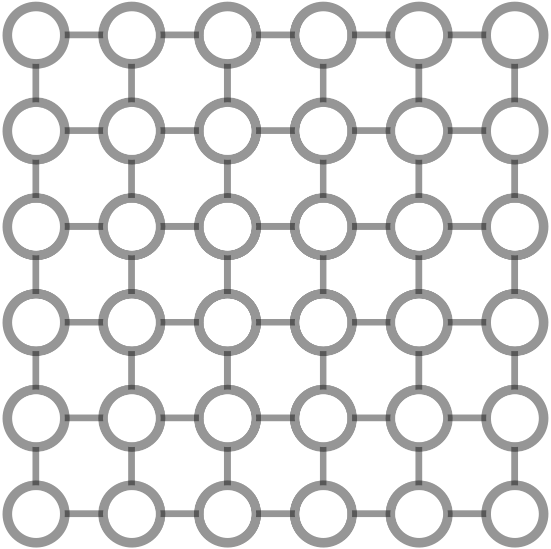
Can we find probably structurally important segments in a sequence?



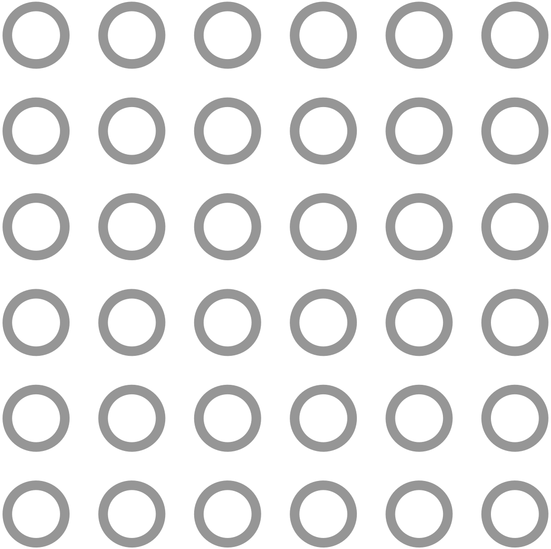
Percolation Example



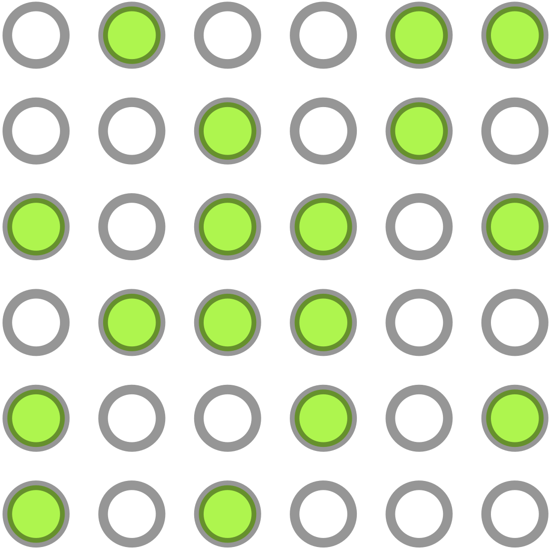
Percolation Example



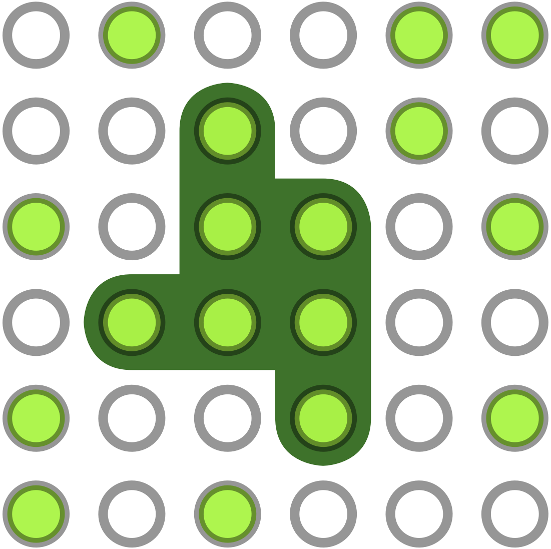
Percolation Example



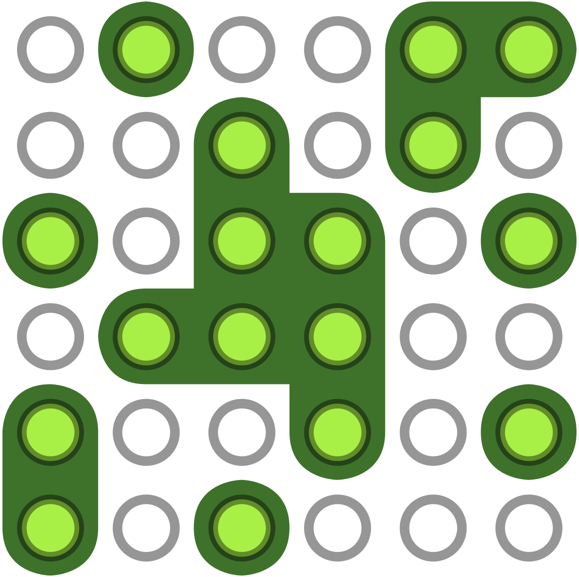
Percolation Example



Percolation Example



Percolation Example



Clusters in a sequence of nodes

We would like to develop a mathematical model that can identify structurally important clusters in sequences of nodes.

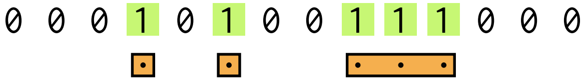
Clusters in a sequence of nodes

Assuming that some of the nodes in the sequence promote cluster formation, consider the following system:

There is a sequence of *1*s and *0*s.

```
0 0 0 1 0 1 0 0 1 1 1 0 0 0
```

*1*s form clusters, while *0*s do nothing.

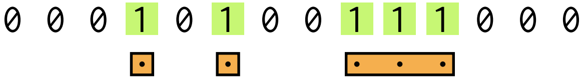


```
0 0 0 1 0 1 0 0 1 1 1 0 0 0
```

Clusters in a sequence of nodes

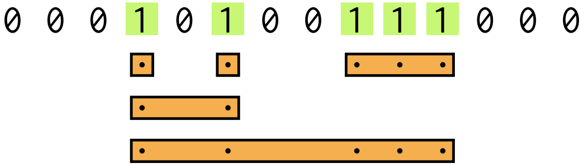
Intro
Model
Breakers
Proteins
Future Work

Connect only immediate neighbors.



0 0 0 1 0 1 0 0 1 1 1 0 0 0

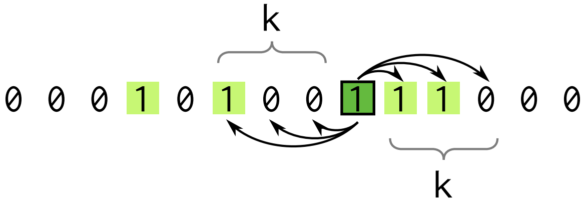
Is it good enough? Not really. We would like to capture clusters separated by *0*s.



0 0 0 1 0 1 0 0 1 1 1 0 0 0

Clusters in a sequence of nodes

Generalize: Each node is connected to k many nodes to the right, and k many to the left. $k \geq 0$.



The resulting clusers *may have gaps* of at most $k - 1$ consecutive *0*s.

Clusters for different k

0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0

k=1

k=2

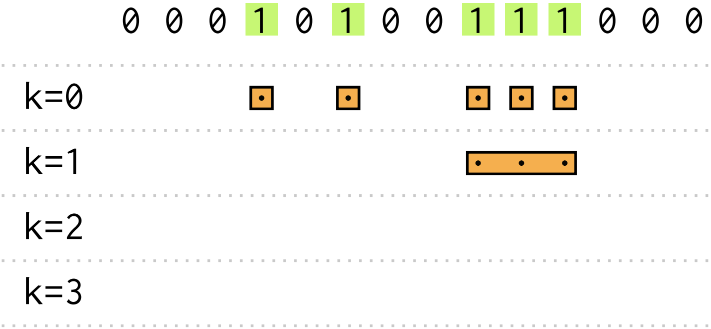
k=3

Intro
Model
Breakers
Proteins
Future Work

Clusters for different k

	0	0	0	1	0	1	0	0	1	1	1	0	0	0
k=0				⊡		⊡			⊡	⊡	⊡			
k=1														
k=2														
k=3														

Clusters for different k



Clusters for different k

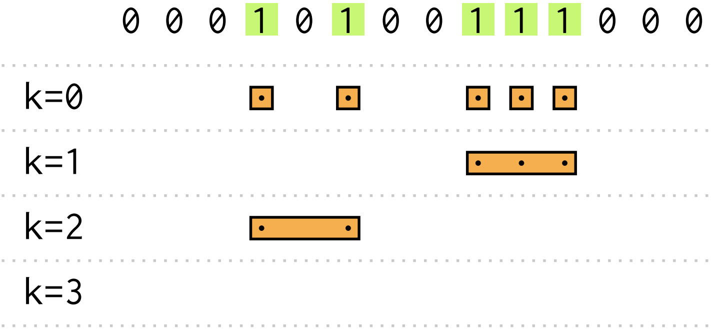
Intro

Model

Breakers

Proteins

Future Work



0 0 0 1 0 1 0 0 1 1 1 0 0 0

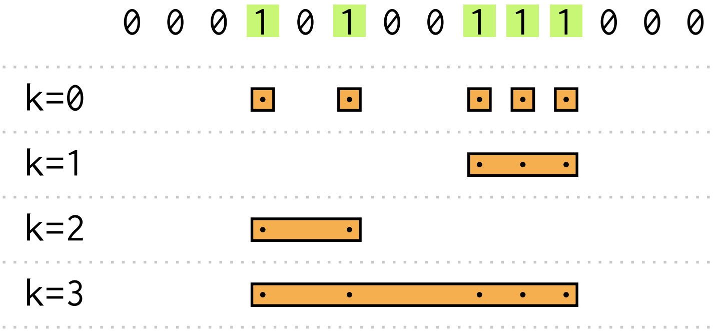
k=0

k=1

k=2

k=3

Clusters for different k



0 0 0 1 0 1 0 0 1 1 1 0 0 0

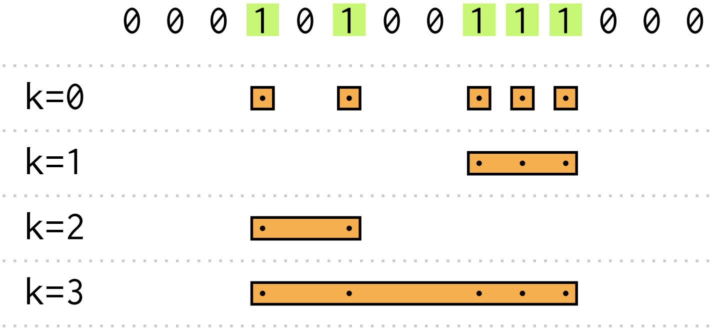
k=0

k=1

k=2

k=3

Clusters for different k



0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0

k=1

k=2

k=3

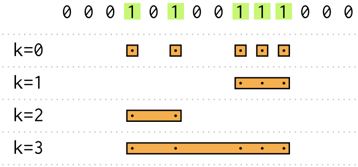
Too many clusters! Which are really important?

Probabilistic model

If it is observed that *1*s and *0*s are found in sequences with certain probabilities:

- p is the probability of *1*s, and
- $q = 1 - p$ is the probability of *0*s,

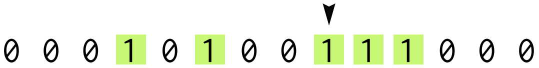
we can compute, how probable each of the clusters is.



Probabilistic model

Def. *Size of a cluster is the number of 1s in it.*

Def. *Given a 1, let $w_{k,s}$ be the probability to find that 1 in a cluster of size s at level k.*


0 0 0 1 0 1 0 0 1 1 1 0 0 0

$$w_{k,s} = (\beta_{k,s} - \beta_{k-1,s}) \cdot q^{2k},$$

$$\text{where } \beta_{k,s} = s(p\alpha_k)^{s-1}, \text{ and } \alpha_k = \frac{1-q^k}{1-q}.$$

Choosing the best cluster

Ok, if we found a cluster, how rare is it?

Def. Weight *of a cluster with size s at level k is*

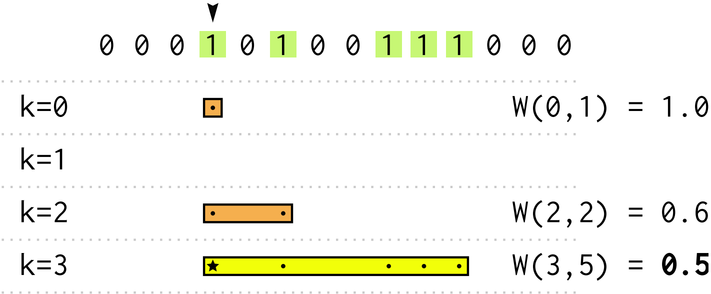
$$W(k,s) = \frac{1}{\zeta_k} \min\left(\sum_{t=1}^{s} w_{k,t}, \sum_{t=s}^{\infty} w_{k,t} \right)$$

The normalizing constant $\zeta_k = \sum_{s=1}^{\infty} w_{k,s}$.

If a cluster has very small weight, it is not very likely to occure at random. Thus we can expect that it is important.

If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Choosing the best cluster



0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0 W(0,1) = 1.0

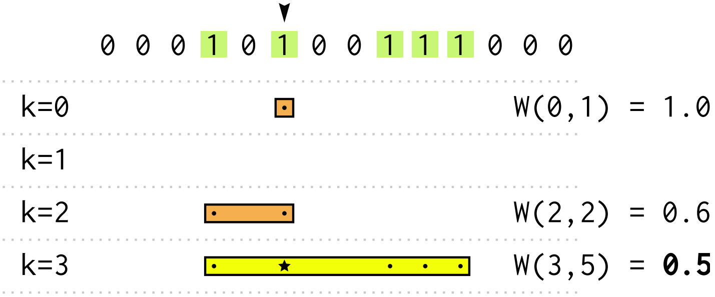
k=1

k=2 W(2,2) = 0.6

k=3 W(3,5) = **0.5**

If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Choosing the best cluster



0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0 W(0,1) = 1.0

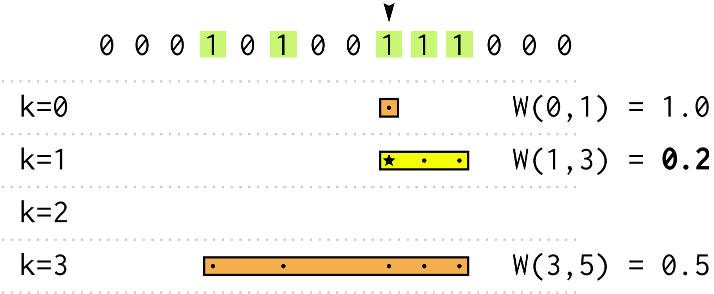
k=1

k=2 W(2,2) = 0.6

k=3 W(3,5) = **0.5**

If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Choosing the best cluster



0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0 W(0,1) = 1.0

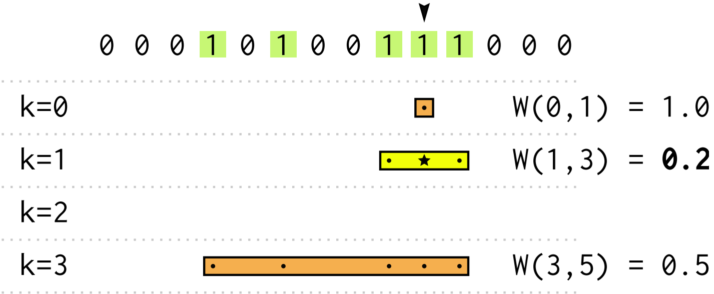
k=1 W(1,3) = **0.2**

k=2

k=3 W(3,5) = 0.5

If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Choosing the best cluster



0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0 W(0,1) = 1.0

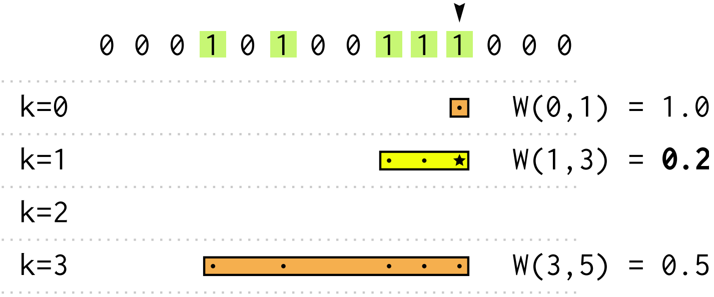
k=1 W(1,3) = **0.2**

k=2

k=3 W(3,5) = 0.5

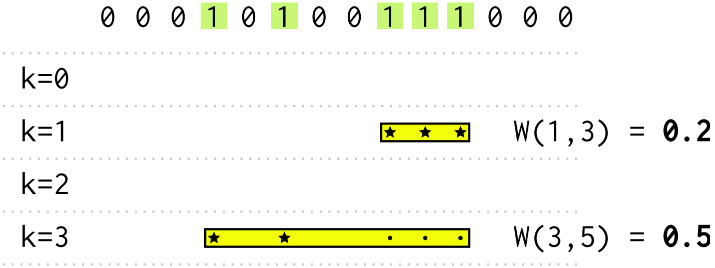
If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Choosing the best cluster



If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Chosen best clusters can be nested



0 0 0 1 0 1 0 0 1 1 1 0 0 0

k=0

k=1 ★ ★ ★ W(1,3) = **0.2**

k=2

k=3 ★ ★ • • • W(3,5) = **0.5**

If a given *1* belongs to several clusters, we choose the one with the *least* weight.

Choosing the best cluster

It can be nice to know the distribution of the best clusters. At what level k they are usually found?

Let $P(k)$ be the probability that, for a given 1, the best cluster is at the level k.

Theorem. $P(k) = 0$ *for all* k.

That is, for any cluster, you can always find a better one, if the sequence is long enough.

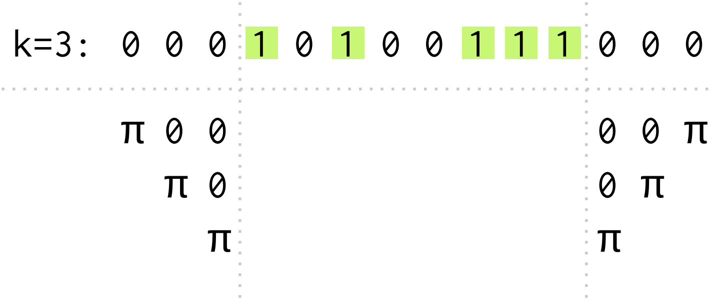
We have to stop clusters growing infinitely large!

Need for breakers

Some nodes that were previously zeroes now become *breakers*. Once reached, they stop cluster growth completely. Call them π in our single-character notation.

Let π also denote the probability of breakers.

$$p + q + \pi = 1$$



```
k=3:  0 0 0 | 1 0 1 0 0 1 1 1 | 0 0 0
      ------+-----------------+------
      π 0 0 |                 | 0 0 π
        π 0 |                 | 0 π
          π |                 | π
```

Intro
Model
Breakers
Proteins
Future Work

Probability $w_{k,s}$ for the breakers case

With the introduction of breakers, we actually can get three types of clusters:

Open on both sides:

$$w_{k,s}^{(0\pi)} = (\beta_{k,s} - \beta_{k-1,s}) \cdot q^{2k}$$

With a breaker on one side:

$$w_{k,s}^{(1\pi)} = (\beta_{k,s} - \beta_{k-1,s}) \cdot 2q^{k} \alpha_k \pi$$

With breakers on both sides:

$$w_{k,s}^{(2\pi)} = (\beta_{k,s} - \beta_{k-1,s}) \cdot (\alpha_k \pi)^2$$

Weight $W(k,s)$ for the breakers case

Intro
Model
Breakers
Proteins
Future Work

With the introduction of breakers, we actually can get three types of clusters:

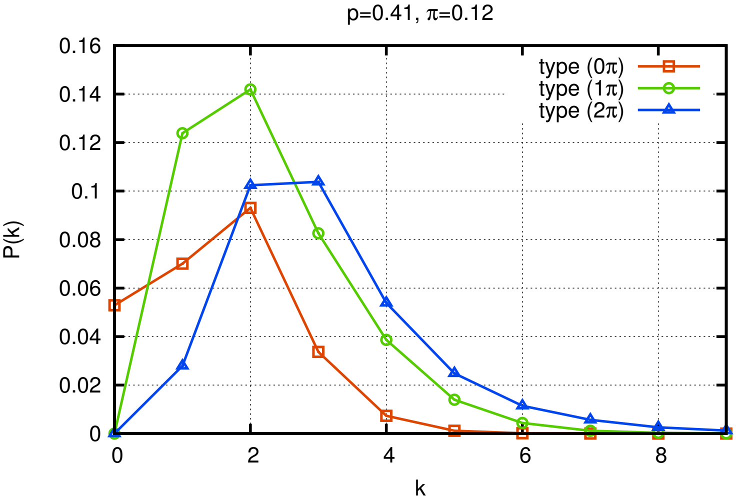
$$W^{(X\pi)}(k,s) = \frac{\min\left(\sum_{t=1}^{s} w_{k,t}^{(X\pi)},\ \sum_{t=s}^{\infty} w_{k,t}^{(X\pi)}\right)}{\sum_{t=1}^{\infty}\left(w_{k,t}^{(0\pi)} + w_{k,t}^{(1\pi)} + w_{k,t}^{(2\pi)}\right)}$$

where $X \in \{0,\ 1,\ 2\}$.

$$\sum_{t=1}^{\infty} w_{k,t}^{(X\pi)} = \sum_{t=1}^{\infty} (\beta_{k,t} - \beta_{k-1,t}) \cdot C_k^{(X\pi)} = (B_k - B_{k-1}) \cdot C^{(X\pi)},$$

where $B_k = \dfrac{1}{(p\alpha_k - 1)^2}$, $C_k^{(0\pi)} = q^2$, $C^{(1\pi)} = 2q^k\alpha_k\pi$, and $C_k^{(2\pi)} = (\alpha_k\pi)^2$. Also, $\alpha_k = (1-q^k)/(1-q)$ (the same as before).

$P(k)$. The probability to choose a cluster at level k.



Experiments with pretein databases

Can we make our method find secondary structures (helices and strands)?

How amino acids map to $\{1,\ 0,\ \pi\}$? Use genetic algorithm.

We simply say that if a residue is covered by any of our clusters, we predict that it belongs to a helix or a strand. Then, check, how good the prediction is.

$$\text{Fitness} = \frac{\text{number of correctly predicted residues}}{\text{total number of residues}}$$

Experiments with pretein databases

We get with fitness 67%:

$$\begin{aligned} \{V,\ I,\ L,\ F,\ M,\ Y,\ W,\ A\} &\longrightarrow 1 \\ \{P,\ G\} &\longrightarrow \pi \\ \text{others} &\longrightarrow 0 \end{aligned}$$

Hydrophobic amino acids are responsible for cluster formation.

Can we really predict secondary structures?

Secondary structure prediction?

There are “Helix”, “Strand”, and “Coil” regions.

1) Drop clusters that have size $s = 1$.

2) We predict that residues in clusters formed at levels $k = 1$ and $k = 2$ are *Strands*.

3) We predict that the remaining residues in other clusters are *Helices*.

4) The rest residues are *Coils*.

$$\text{Q3} = \frac{\text{number of correctly predicted residues}}{\text{total number of residues}}$$

Secondary structure prediction?

Genetic algorithm on randomly selected records from DSSP produced the following map:

$$\begin{aligned} \{V,\ I,\ L,\ F,\ M,\ Y\} &\rightarrow 1 \\ \{P,\ G\} &\rightarrow \pi \\ \text{others} &\rightarrow 0 \end{aligned}$$

With this map, on a standard protein dataset CB-513, we get

$$Q3 = 55\%.$$

This is not 70-80%, but still it is better than, e.g. Chou-Fasman method that has $Q3 = 46 - 48\%$.

Future work

1. To go beyond secondary structures:
   - How to make breakers weaker?
   - Probabilistic assignment of the map residue $\longrightarrow \{1, 0, \pi\}$.
   - Get rid of breakers, and insert strings of zeroes instead, e.g. $P \longmapsto 00000$, and $G \longmapsto 00$.
2. How far can we get in predicting sec. structures?
   - Map pairs or triples of residues to $\{1, 0, \pi\}$.
   - Search for helices and strands separately.
3. Use clusters to guide protein folding simulation.