Administration

Sprache ?

- zu verhandeln

Topics

- Proteins / DNA / RNA
- alignments, conservation, ..

What if I failed in 2018 ?

Administration

Who are we ?

- Andrew Torda
- Timur Olzhabaev
- admin - Annette Schade (schade@zbh.uni-hamburg.de)

Where am I

- 42838 7331
- ZBH 1st floor (Bundesstr. 43)

Background

- numerical simulations

Who are you ?

- MLS 90 % ? (Pflicht)
- Biologen, Chemiker ?
- Others – vorsicht - Voraussetzungen
  - what is a protein ? what are amino acids ?
  - what is the DNA code ?
  - what does haemoglobin do ?

Übungen

- Freitage

- wenig Flexibilität mit den Gruppen
- Sie können sich untereinander austauschen

- komplizierte Fragen - Studienbüro

- man sollte die Übungen machen
  - klausurfragen

Prüfungen

Schriftliche Klausur

- 90 Minuten (vielleicht gibt es eine 2 stunde Buchung)
- 22 Juli / 30 Sep

Bücher

Nicht nötig

Folien

- In openolat (nicht Stine)

Overall Structure

- Sequences and alignments
- multiple sequence alignments, evolutionary aspects
- genomics
- applications, statistics

Goals, why are we here ?

Overall

- Given some sequence data, what can we find out about it ?
- Some references to structures

Situations – you have

- DNA sequence data (very common)
- corresponding protein sequence (rather common)
  - splice variants, pre/pro-versionen
- structure of a protein (much, much less often)

- If you have the structure of a protein, you know the sequence

My examples

Usually well-studied proteins

- myoglobin, haemoglobin, DNA gyrase, metabolic enzymes

In my examples

- you have sequence and structure
- pretend you are trying to work something out from the sequence

In real world

- you may have only sequence information

Data

Structures

- protein data bank (1.5×10^5 files)

Sequences

- genbank, NCBI (1.7×10^8 sequences)

Freely available

- downloadable
- searchable via web
- requirement for publication (often)

Not much private/secret data

Predictions

- what shape is this molecule ?
- will this small molecule inhibit some enzyme ?
- will this molecule be broken down in the body quickly ?

…

Predictions – different approaches

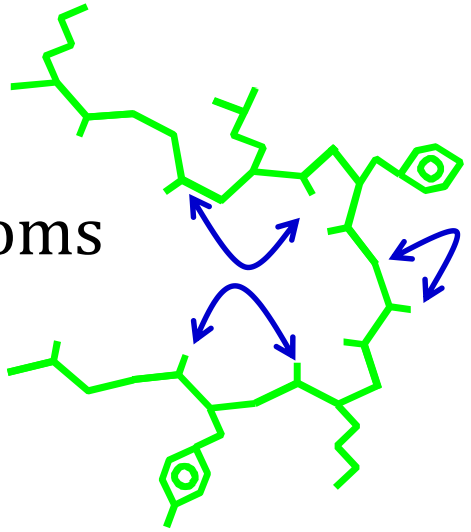
1. First principles (physics, chemistry)
2. Finding patterns (underlying principles not known)
3. Similarity

… explanation

1. First principles prediction

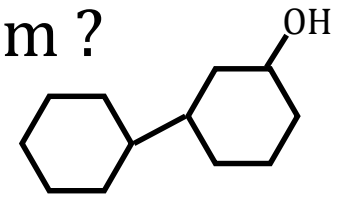
Protein structure example

- a protein molecule = set of atoms in space
- I know all the interactions between the atoms
- should be able to predict the 3D structure



Quantum chemistry

- I have a model for electron wave functions
- can I predict electron density around each atom ?
- predict pK_a for this molecule ?
- …



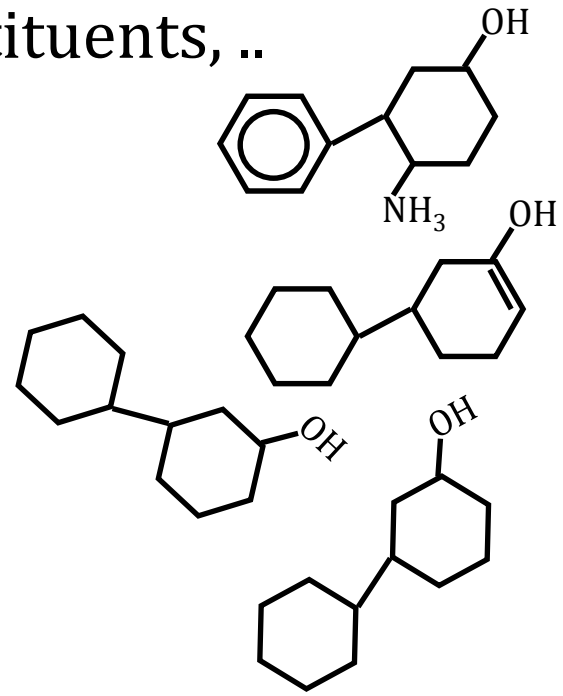
Elegant

- expensive, needs good models

2. Finding patterns

Take known data – collect properties, look for correlations

- look at mol wt, aromatic/aliphatic, substituents, ..
- for each molecule collect pK_a
- hope patterns can be found



Gene regulator recognition example

- take known examples
  - look at GC content
  - proximity to protein
  - sizes …

Field of data mining, machine learning

- little understanding of problem / chemistry
- often works

3. Similarity

Most important for this course

DNA

- is this region coding ?
- where does the reading frame start ?
- is this region involved in regulator binding ?

Protein sequence

- can one guess the structure ?
- is this membrane bound ?
- does it have a certain activity (kinase, transferase, ..) ?

Protein structure

- what is a likely function ?

Prediction by similarity

Need databases

- DNA sequences, protein sequences
- therapeutics with known properties

For some queries / your sequence.. Is your

- protein sequence similar to a known structure ?
- DNA sequence similar to a known regulatory region ?
- RNA similar to some RNAzyme ?

Experiments are slow, expensive, dangerous, smelly

Similarity in sequences

Protein / nucleotide

- same ideas, differences later

Questions

- are two sequences similar ?
- suspected similarity
  - how reliable is it ?
- detailed alignments (modelling, important residues, ..)

Plan

- generalities
- alignment methods
- DNA versions
- Protein versions
- differences

Alignments and Similarities

Problem

```
. . . A C A C T G A C T A . . .
. . . . . . A T T G A G T A . . .
. . . . . . 1 0 1 1 1 0 1 1 . . .
```

- 6 of 8 positions match

Implicitly

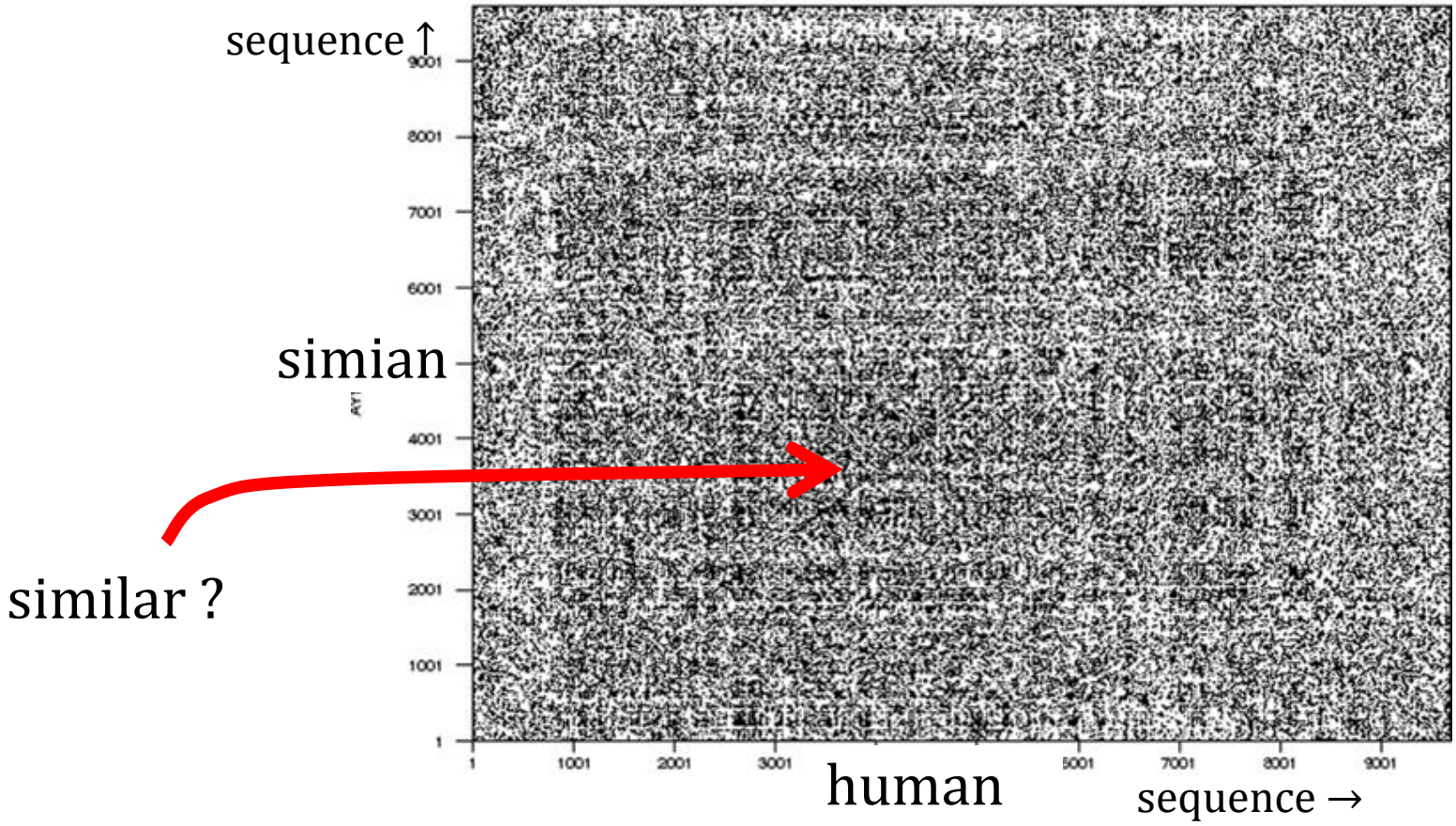
- I have already moved second sequence over the first
- gaps

```
. . . A C A C T T G A C T A . . .
. . . . . . A T T – G A G T A . . .
. . . . . . 1 0 1 0 1 1 0 1 . . .
```

- alignment not so obvious (gaps anywhere)
  - quick look

dot plot

Human and simian HIV



dot plot filtered

Similarity up to about 5200

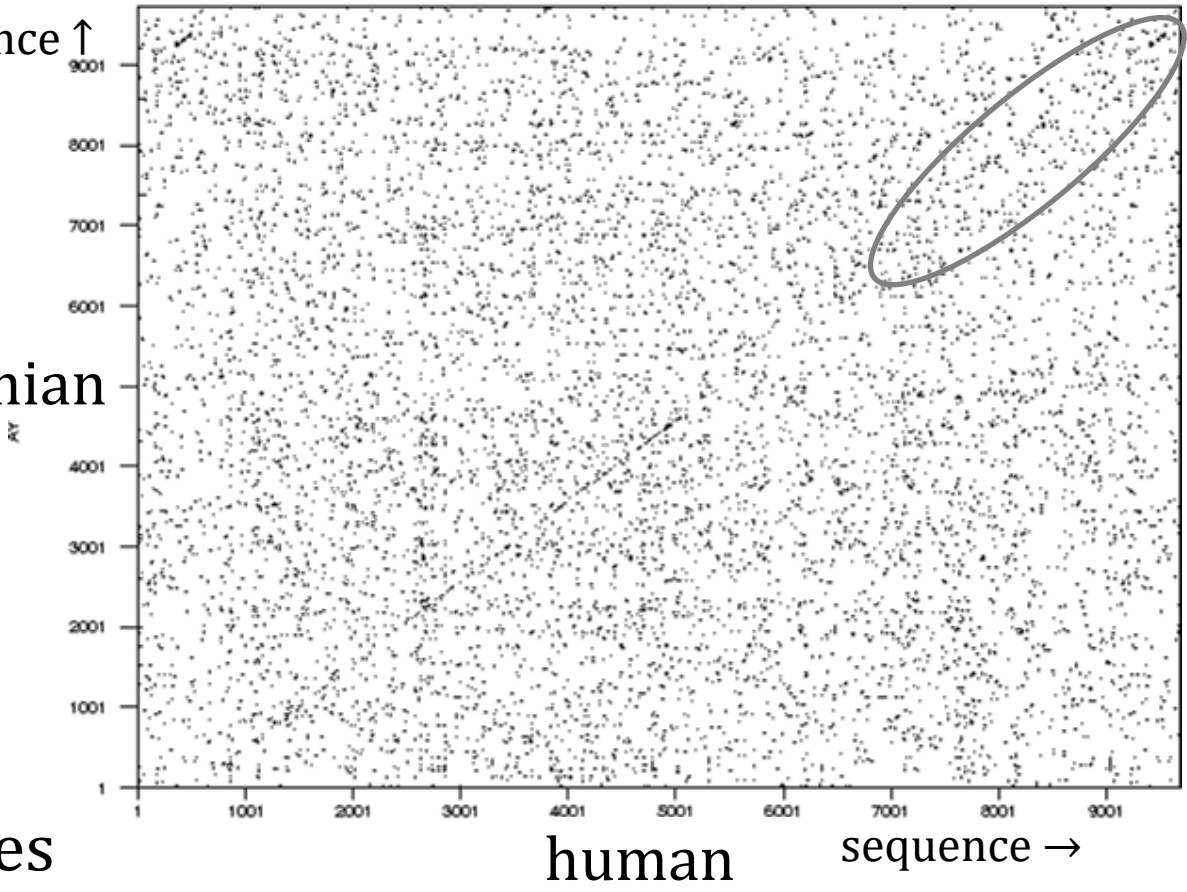
Circled region ?

- not so clear

- easy for a human to recognise
- not so easy to automate

Worse case …

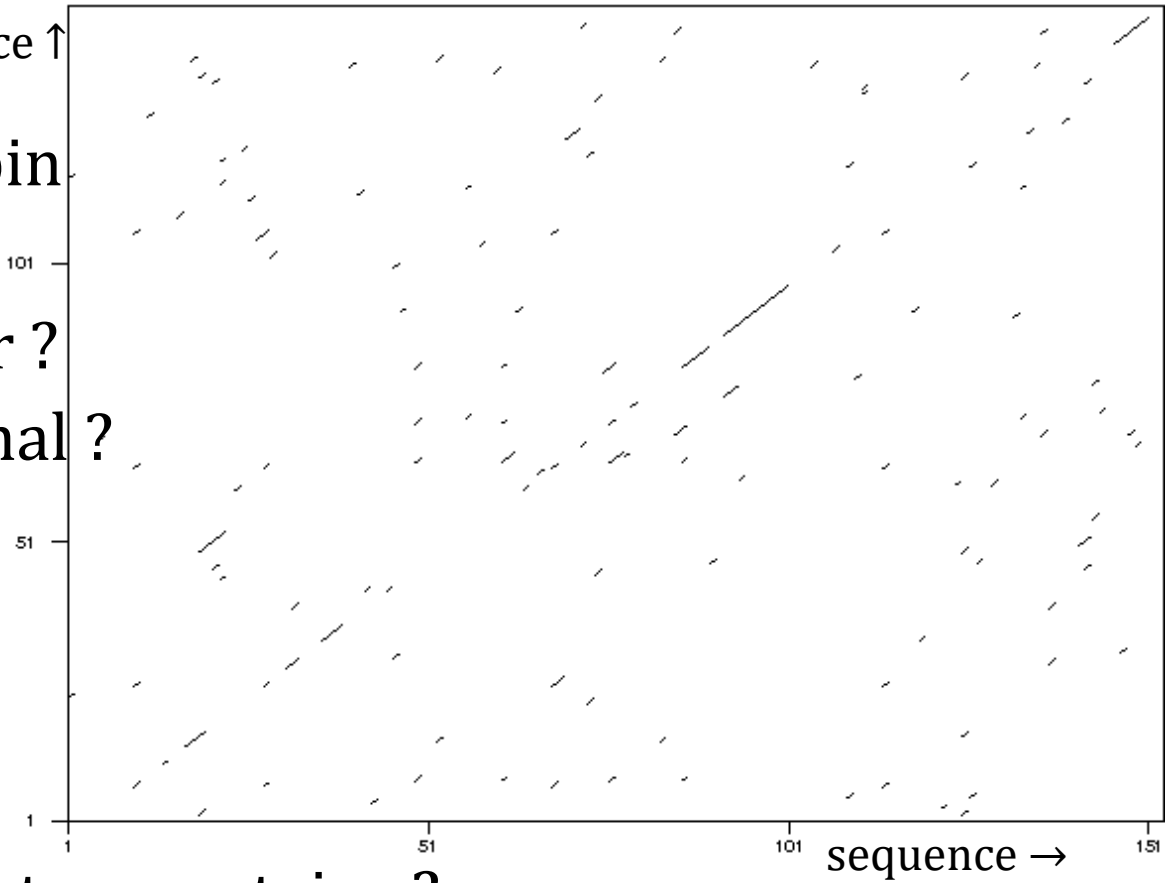
- two protein sequences



protein dot plot

2 proteins

- 2nrl, 2o58
- tuna / horse myoglobin

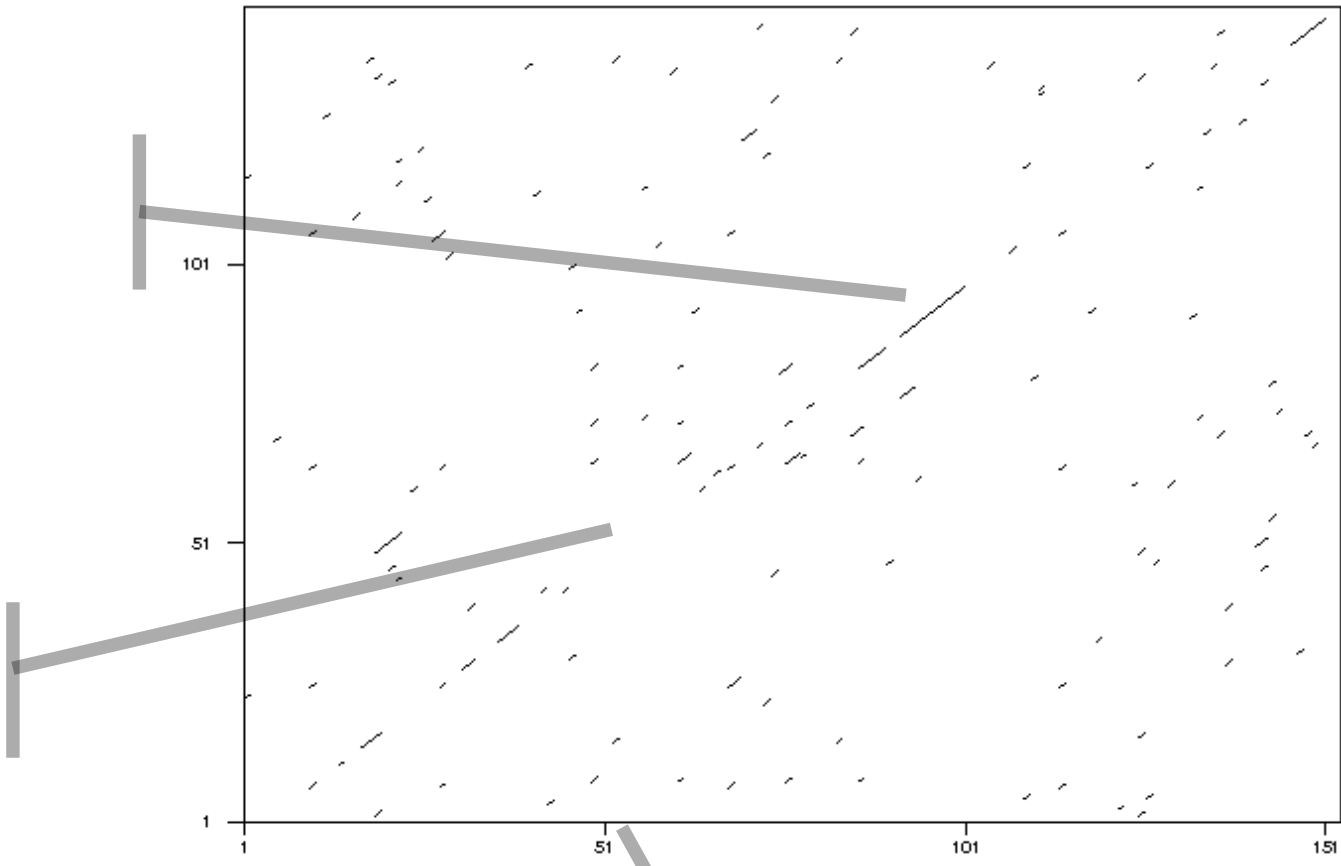


- are they really similar ?
- how real is the diagonal ?

- what is the identity ?
  - ≈ 45 %
- how similar are these two proteins ?
- is there a "correct alignment" ? Physical interpretation ?

Properties of alignment ?

Is the alignment above diagonal ?



What is happening here ?

What is aligned to residue 51 ?

Here we know the answer

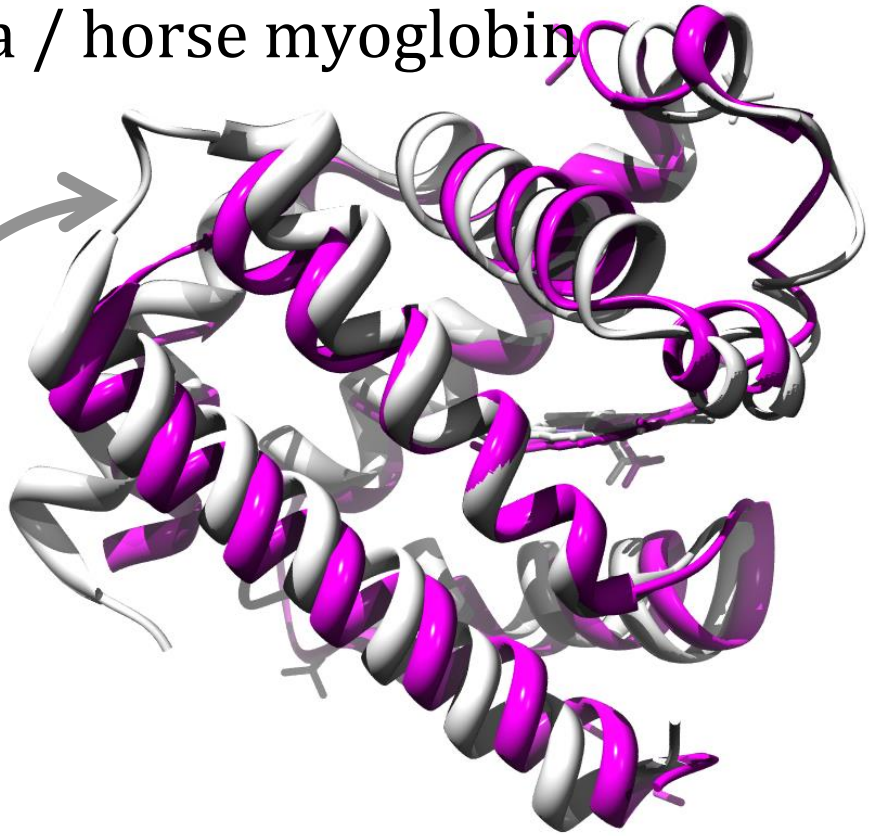
- look at structures

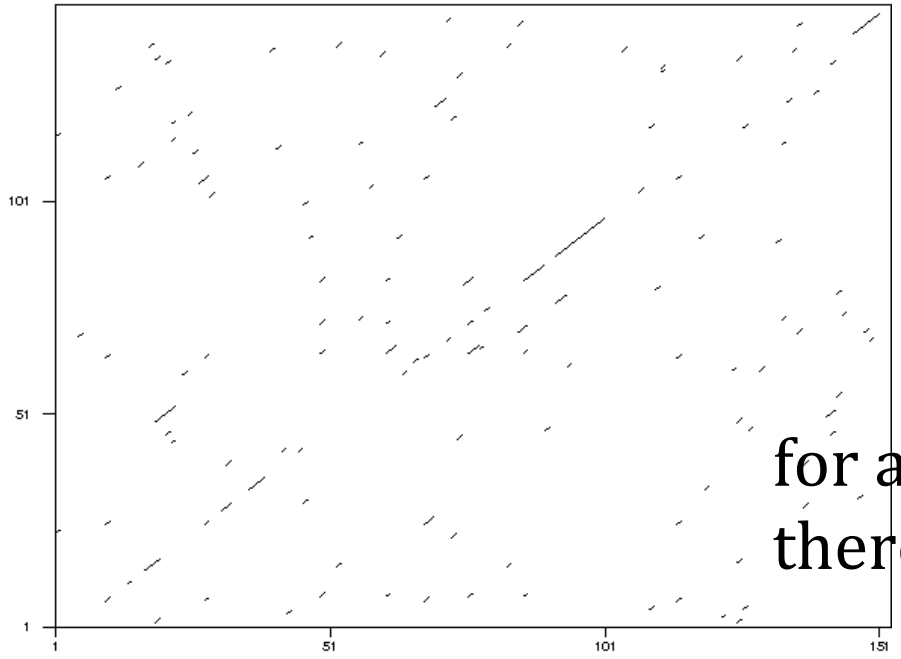
If one knew the structure

The same proteins as before: tuna / horse myoglobin

There are no holes ?

- there are some differences
  - some bits are longer

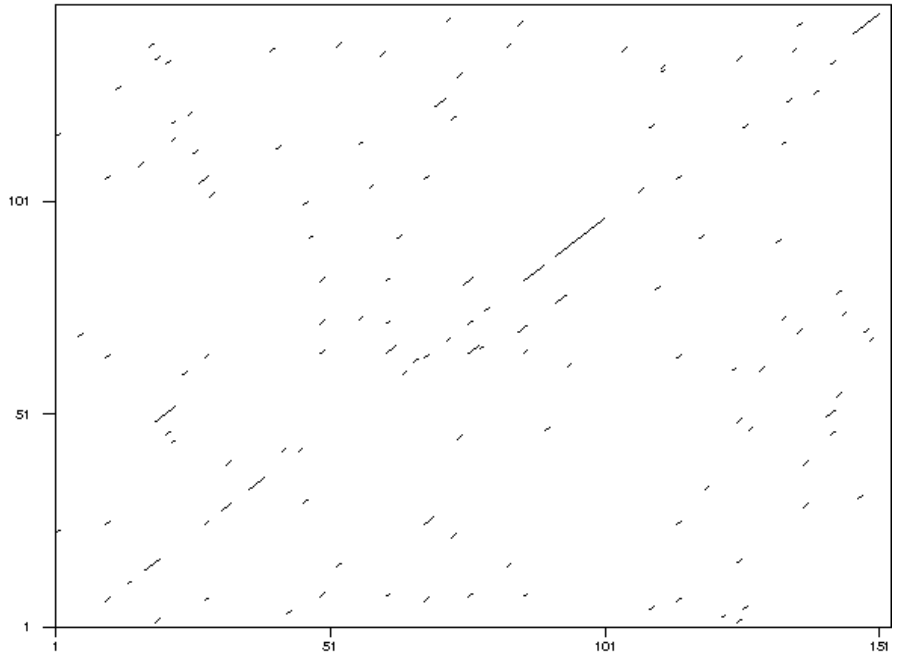


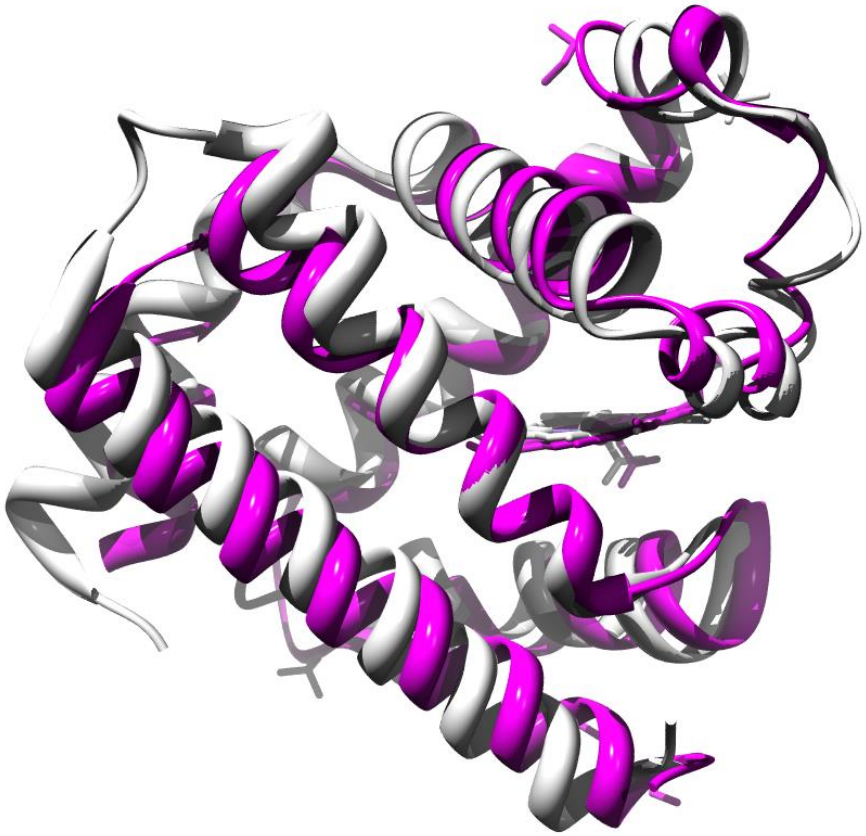


for almost every pink residue,
there is a corresponding grey residue

If one knew the structure..

Would you have recognised this from dotplot ?





Look at residue 51 in dot plot

- aligned residue not clear

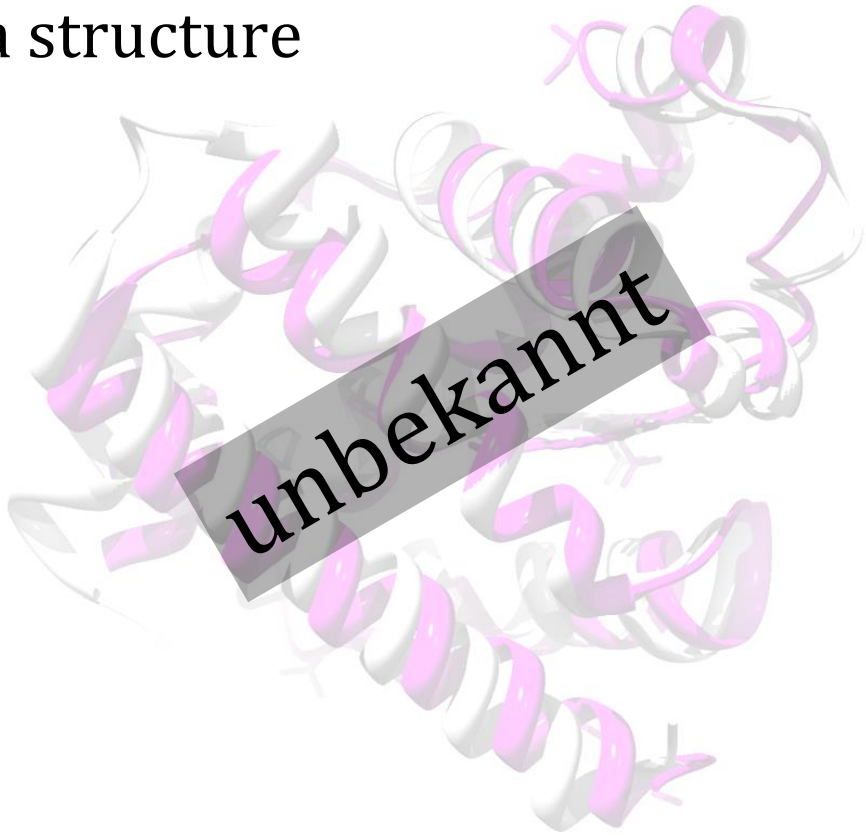
Look in structure

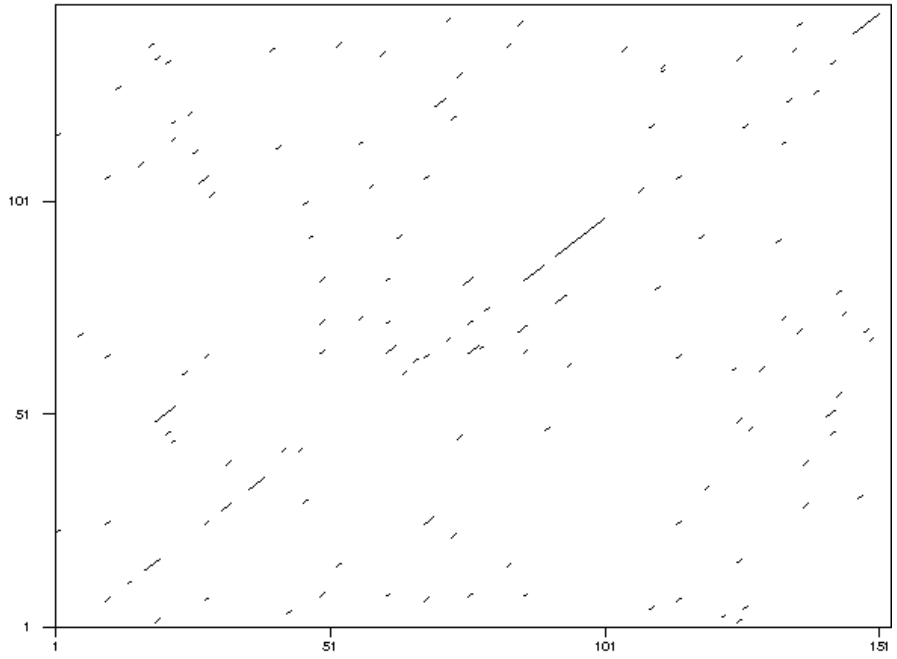
- aligned residues clear

If one knew the structure..

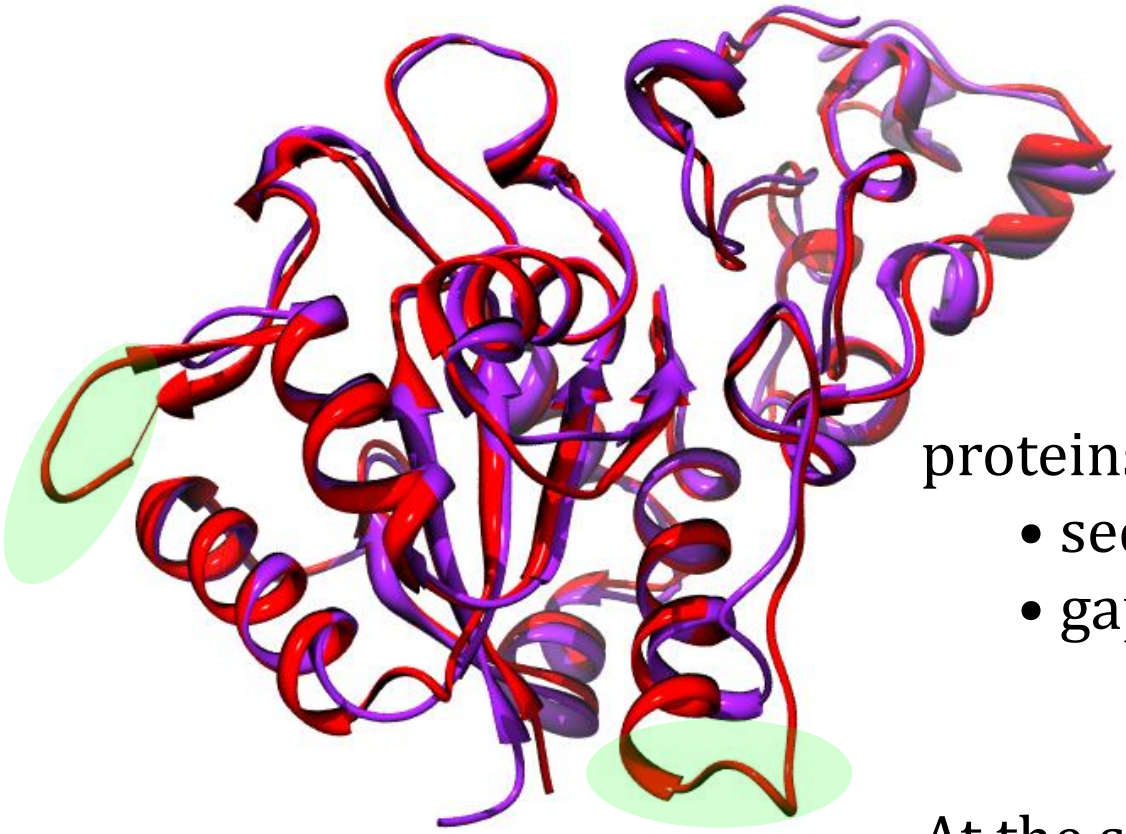
In these lectures, we do not have a structure

- we get as far as we can from sequence





Clearer Example

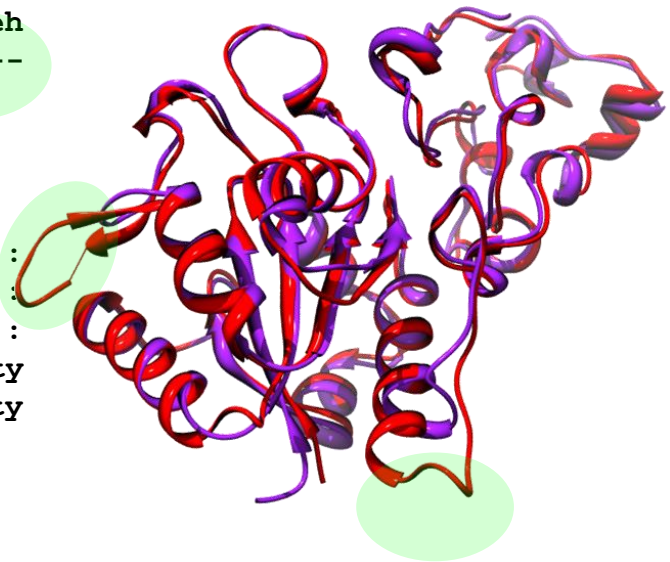


hydrogenases
- 40 % sequence identity
- 2frvG & 1cc1S

proteins – obviously similar
- sequence identity OK
- gaps and insertions

At the sequence level ?

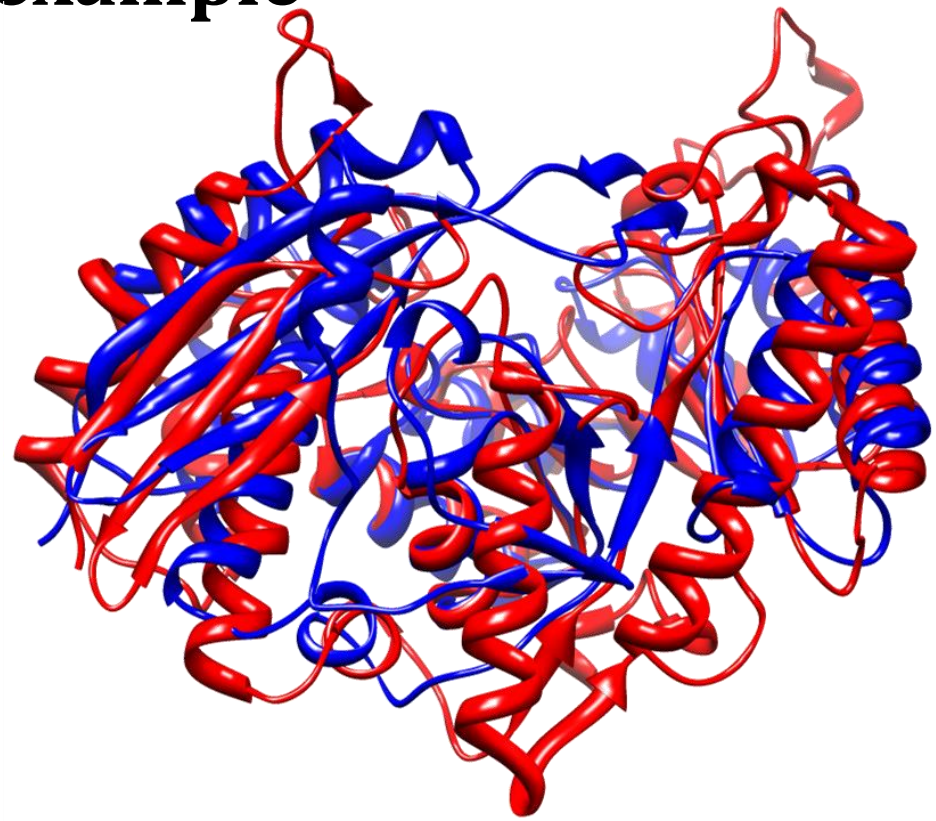
```
Seq ID 40.6 % (103 / 254) in 280 total including gaps
    :    1    :    2    :    3    :    4    :    5    :    6
    :    0    :    0    :    0    :    0    :    0    :    0
kkapviwvqgqgctgcsvsllnavhprikeilldvislefhptvmasegemalahmyeia
krpsvvylhnaectgcsesvlrtvdpyvdelildvismdyhetlmagaghaveea-l-he
   :    1    :    2    :    3    :    4    :    5    :
   :    0    :    0    :    0    :    0    :    0    :

    :    0    :    0    :    0    :    1    :    1    :    1
    :    7    :    8    :    9    :    0    :    1    :    2
    :    0    :    0    :    0    :    0    :    0    :    0
ekfngnfflvegaiptakegrycivgeakahhhevtmmelirdlapkslatvavgtcsa
aikg-dfvcvieggipmgdggywgk-------vggrnmydicaevapkakaviaigtcat
0      :    0    :    0              :    0    :    1    :    1
6      :    7    :    8              :    9    :    0    :    1
0      :    0    :    0              :    0    :    0    :    0

    :    1    :    1    :    1    :    1    :    1    :    1
    :    3    :    4    :    5    :    6    :    7    :    8
    :    0    :    0    :    0    :    0    :    0    :    0
yggipaaegnvtgsksvrdffadekiekllvnvpgcpphpdwmvgtlvaawshvlnpteh
yggvqaakpnptgtvgvnealgklgvkai--niagcppnpmnfvgtv--vhlltk-----
   :    1    :    1    :    1       :    1    :       1
   :    2    :    3    :    4       :    5    :       6
   :    0    :    0    :    0       :    0    :       0

    :    1    :    2    :    2    :    2         :    2    :
    :    9    :    0    :    1    :    2         :    3    :
    :    0    :    0    :    0    :    0         :    0    :
plpeldddgrpllffgdnihencpyldkydnsefaetftkpg-----ckaelgckgpsty
gmpeldkqgrpvmffgetvhdncprlkhfeagefatsfgspeakkgyclyelgckgpdty
  :    1    :    1    :    1    :    2    :    2    :    2
  :    7    :    8    :    9    :    0    :    1    :    2
  :    0    :    0    :    0    :    0    :    0    :    0
```



Another example

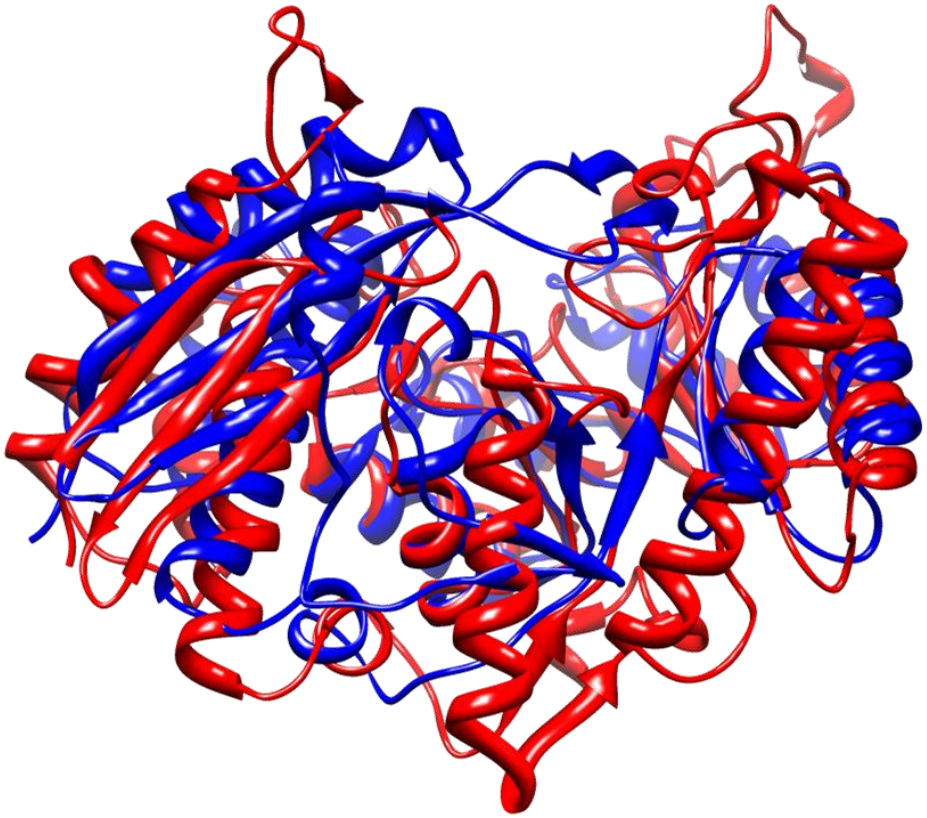
Structures known

- can easily get sequence alignment



2mnr mandelate racemase
4enl enolase

```
Seq ID 25.1 % (81 / 323) in 373 total including gaps
    :    1    :    2    :       3    :           4    :    5
    :    0    :    0    :       0    :           0    :    0
sktyavlglgngghafaaylalkgqsv--lawdidaqr------ikeiqdrgaiiaegpg
svehimrdv-nggwa-mryihangaslfflavyihifrglyygsykapreitwivgmviy
  0    :    0    :    1    :    1    :    1    :    1    :
  8    :    9    :    0    :    1    :    2    :    3    :
  0    :    0    :    0    :    0    :    0    :    0    :

    :    0    :    0    :    0    :    0    :    1        :
    :    6    :    7    :    8    :    9    :    0        :
    :    0    :    0    :    0    :    0    :    0        :
la--gtahpdlltsdiglavkdadvilivvpaihhasiaaniasyisegqli---ilnpg
llmmgtafmgyvlpwgqmsfwgatvitglfgaipg--igpsiqawllggpavdnatlnrf
    1    :    1    :    1    :    1        :    1    :    1
    4    :    5    :    6    :    7        :    8    :    9
    0    :    0    :    0    :    0        :    0    :    0

  1    :    1    :    1    :    1    :    1    :    1    :
  1    :    2    :    3    :    4    :    5    :    6    :
  0    :    0    :    0    :    0    :    0    :    0    :
atggalefrkilrengapevtigetssmlftcrserpgqvtvnaikgamdfaclpaakag
fslhyllpf-viaalvaihiwafhttgnnnptgvevrrtskadaekdtlpfwpyfvikdl
 :    2    :    2    :    2    :    2    :    2    :    2
 :    0    :    1    :    2    :    3    :    4    :    5
 :    0    :    0    :    0    :    0    :    0    :    0

  1    :    1    :    1    :       2    :    2    :         2
  7    :    8    :    9    :       0    :    1    :         2
  0    :    0    :    0    :       0    :    0    :         0
waleqigsvlpqyvavenvlhtsltnv-navm-hplptllnaarcesgtpf----qyyl-
fala-l--vllgffavvaympnylghpdnyvqanplstpahivpewyflpfyailrafaa
  :       2    :    2    :    2    :    2    :    3    :
  :       6    :    7    :    8    :    9    :    0    :
  :       0    :    0    :    0    :    0    :    0    :

     :    2    :    2    :    2    :    2    :    2    :
     :    3    :    4    :    5    :    6    :    7    :
     :    0    :    0    :    0    :    0    :    0    :
-egitpsv-gslaekvdaeriaiakafdlnvpsvcewypatiyeavqgnpayrgiagpin
dvwvvilvdgltfgivdakffgviamfga-i-avmalapw-ldtskvrsgayr----pkf
3    :    3    :    3    :       3    :    3    :         3
1    :    2    :    3    :       4    :    5    :         6
0    :    0    :    0    :       0    :    0    :         0

 2    :    2    :    3    :    3    :    3    :    3    :
 8    :    9    :    0    :    1    :    2    :    3    :
 0    :    0    :    0    :    0    :    0    :    0    :
lntryffedvstglvplselgravnvptplidavldlisslidtdfrkegrtleklglsg
---rmwfwflvldfvvltwvg-a--m--pt-eypydwis-liastywfay-flvilpllg
     :    3    :    3         :    3    :    4    :    4
     :    7    :    8         :    9    :    0    :    1
     :    0    :    0         :    0    :    0    :    0

 3       :
 4       :
 0       :
ltaag--irsave
atekpepipasie
   :    4
   :    2
   :    0
```



Getting alignments

Do we always know the structure ?

- if so, we would not do these lectures

- sequences are cheap
- structures are expensive

Usually one only has the sequence

Mission for today ?

- how does one find the best alignment based on sequence ?

Alignment methods

Best alignment not obvious

```
. . . . . . . . C C A T C C G C . .
. . . C G A T C C – T C C T C . . .
```

6 matches or

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

also 6 matches

Can we invent some rules to say which is best ?

How many alignments ?

For two sequences of length 10, how many alignments could I generate ?

```
. . . . . . . . A B C D E F G H I J . . . .
                Q R S T U V W X Y Z
. . . . . . Q R S T U V W X Y Z + more
```

with gaps

```
            Q R S T U V W X Y - Z
            Q R S T U V W X - Y Z  then with
```

gap 2

```
          Q R S T U V W X Y - - Z
              . . .
```

- then with multiple gaps … combinatorial explosion
- do not tackle the problem directly

Mission – Alignment algorithm

For DNA, protein, RNA

- develop some scoring scheme
- maximize matches and similarities

Algorithm

- allow some gaps, not too many
- must be much faster than brute force
- methods apply to proteins and nucleotides

What is coming

- simple scoring – DNA
- full alignment algorithm (Needleman and Wunsch)
- better scoring – proteins

Scoring for DNA

Sensible scheme

- matched pairs 2
- mismatch −3
- gaps −2

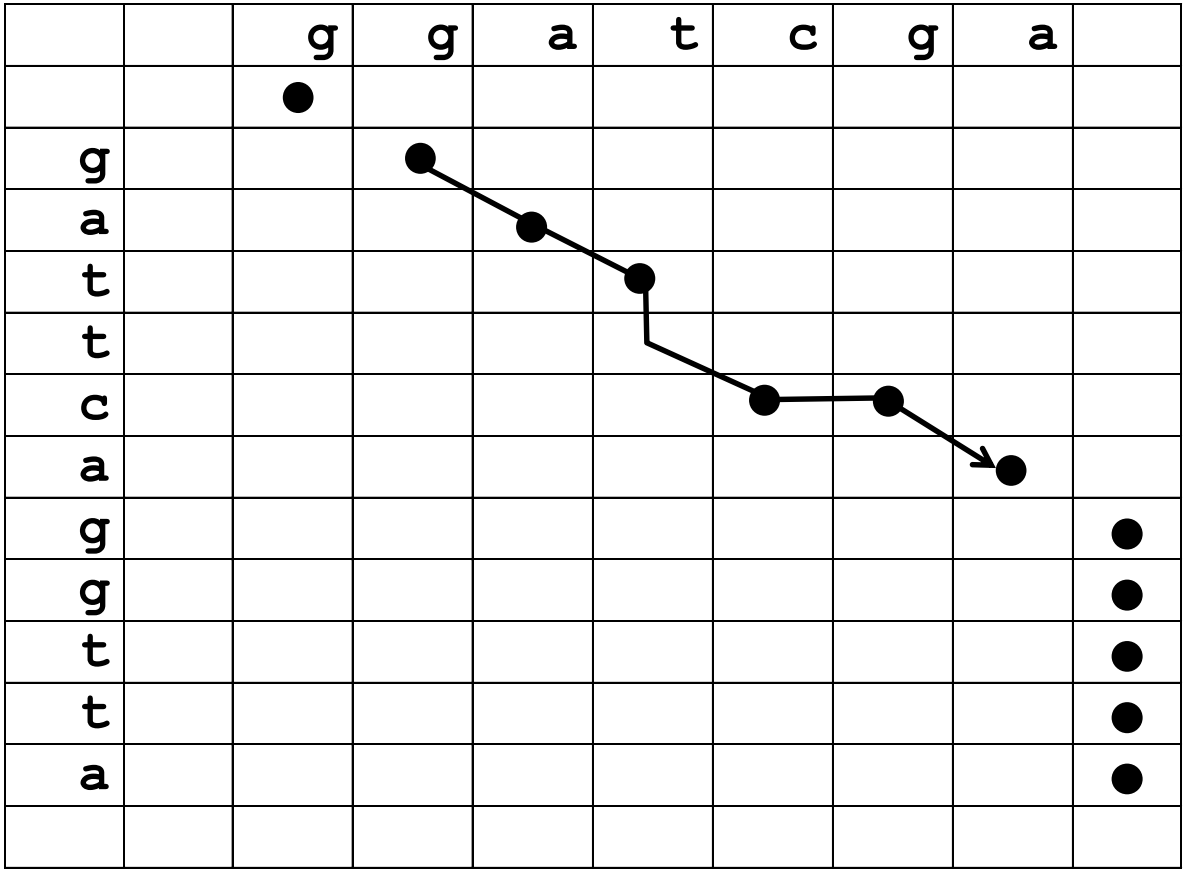
```
A  C  T  G  -  A  T  T  C  G  A
A  C  -  G  C  A  -  T  C  T  A
2  2 -2  2 -2  2 -2  2  2 -3  2
```

more sophisticated..

- gap opening costs − 2
- gap widening costs − 1
- so $cost = cost_{open} + (n_{gap} - 1)cost_{widen}$

Representing alignments

- sequences `GATTCAGGTTA` and `GGATCGA`

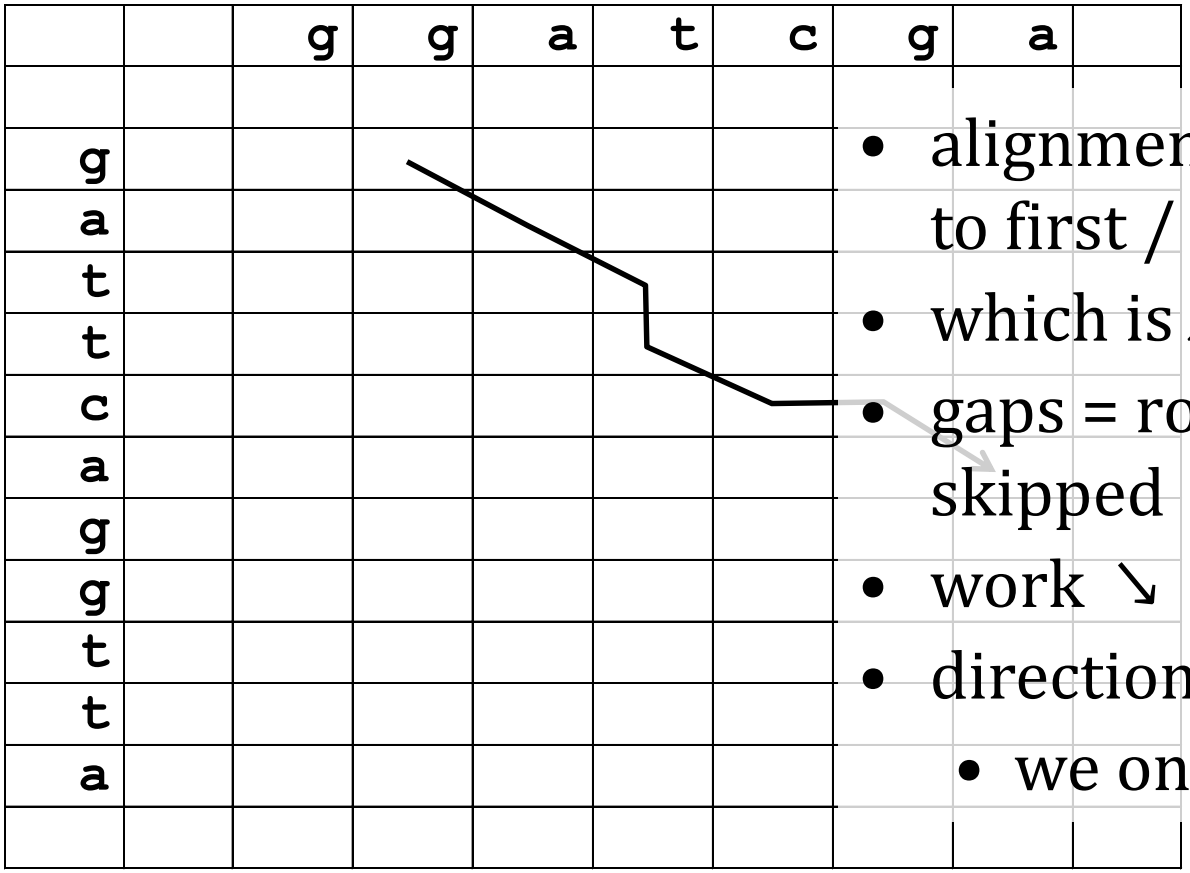


- would mean
  `GGAT-CGA-----`
  `-GATTC-AGGTTA`

- notes…

Representing alignments

```
GGAT-CGA-----
-GATTC-AGGTTA
```

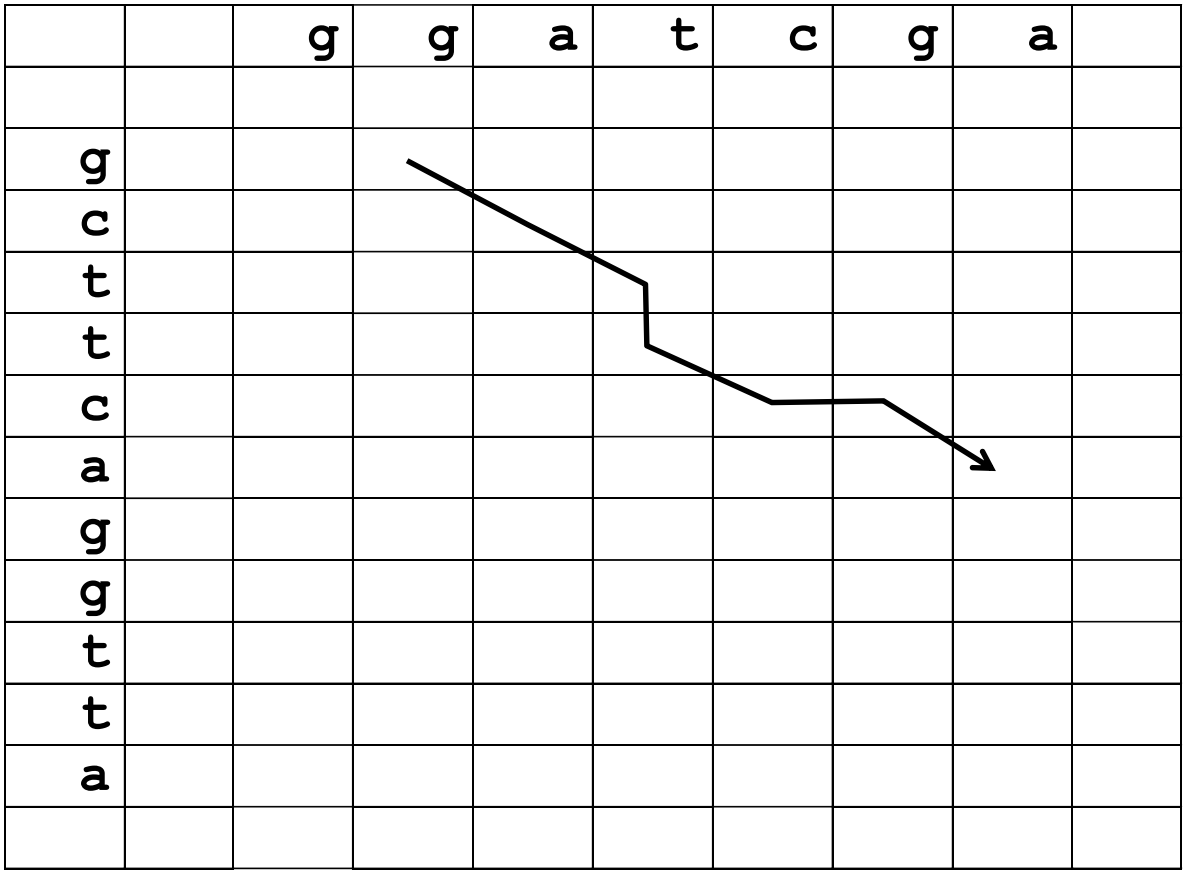


- alignment does not have to go to first / last row or column
- which is *x* and *y* is arbitrary
- gaps = row or column is skipped
- work ↘ or ↖ does not matter
- direction must be consistent
  - we only go → ↓

- make sure this is clear

Representing alignments with mismatches

- sequences GCTTCAGGTTA and GGATCGA



- would mean
  GGAT-CGA-----
  -GCTTC-AGGTTA

Calculating alignment - steps

Needleman and Wunsch algorithm

1. fill score matrix
2. find best score possible in each cell
3. traceback

fill score matrix

- For convenience, add some zeroes to the ends

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0								0
a	0								0
t	0								0
t	0								0
c	0								0
a	0								0
g	0								0
g	0								0
t	0								0
t	0								0
a	0								0
	0	0	0	0	0	0	0	0	0

Mission

- find path through this matrix with best score
- account for gaps

fill score matrix

- For convenience, add some zeroes to the ends
- Add in match, mismatch scores

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-3	2	-3	-3	-3	2	0
t	0	-3	-3	-3	2	-3	-3	-3	0
t	0	-3	-3	-3	2	-3	-3	-3	0
c	0	-3	-3	-3	-3	2	-3	-3	0
a	0	-3	-3	2	-3	-3	2	2	0
g	0	2	2	-3	-3	-3	2	-3	0
g	0	2	2	-3	-3	-3	2	-3	0
t	0	-3	-3	-3	2	-3	-3	2	0
t	0	-3	-3	-3	2	-3	-3	-3	0
a	0	-3	-3	2	-3	-3	-3	2	0
	0	0	0	0	0	0	0	0	0

Mission

- find path through this matrix with best score
- account for gaps

Summing the elements

- start at top left
- move right, then next line
- at each cell
  - find best score it could possibly have

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0	-3	-4	-4	4	3	1	0	2
c	0	-3	-5	-5	-2	6	0	-2	1
a	0	-3	-5	-6	-3	0	3	6	3
g	0	2	0	-6	-4	-1	6	0	6
g	0	2	4	-3	-4	-2	5	3	4
t	0	-3	-1	1	4	-2	-1	2	3
t	0	-3	-3	-1	3	1	-1	0	2
a	0	-3	-4	3	-4	0	-2	4	0
	0	0	-2	0	3	1	0	1	4

Diagonal (no gaps)

for each cell, 3 possible scores

1. **diagonal (no gap)**
2. best from preceding column
3. best from preceding row

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0	-3	-4	-4	4	3	1	0	2
c	0	-3	-5	-5	-2	6	0	-2	1
a	0	-3	-5	-6	-3	0	3	6	3
g	0	2	0	-6	-4	-1	6	0	6
g	0	2	4	-3	-4	-2	5	3	4
t	0	-3	-1	1	4	-2	-1	2	3
t	0	-3	-3	-1	3	1	-1	0	2
a	0	-3	-4	3	-4	0	-2	4	0
	0	0	-2	0	3	1	0	1	4

GAT
GAT

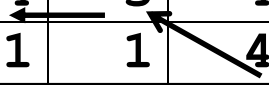
GG
GG

preceding row (gap)

for each cell, 3 possible scores

1. diagonal (no gap)
2. **best from preceding row**
3. best from preceding column

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0	-3	-4	-4	4	3	1	0	2
c	0	-3	-5	-5	-2	6	0	-2	1
a	0	-3	-5	-6	-3	0	3	6	3
g	0	2	0	-6	-4	-1	6	0	6
g	0	2	4	-3	-4	-2	5	3	4
t	0	-3	-1	1	4	-2	-1	2	3
t	0	-3	-3	-1	3	1	-1	0	2
a	0	-3	-4	3	-4	0	-2	4	0
	0	0	-2	0	3	1	0	1	4



GAT
G-T

preceding column (gap)

for each cell, 3 possible scores

1. diagonal (no gap)
2. best from preceding row
3. **best from preceding column**

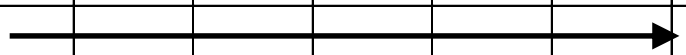
		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0	-3	-4	-4	4	3	1	0	2
c	0	-3	-5	-5	-2	6	0	-2	1
a	0	-3	-5	-6	-3	0	3	6	3
g	0	2	0	-6	-4	-1	6	0	6
g	0	2	4	-3	-4	-2	5	3	4
t	0	-3	-1	1	4	-2	-1	2	3
t	0	-3	-3	-1	3	1	-1	0	2
a	0	-3	-4	3	-4	0	-2	4	0
	0	0	-2	0	3	1	0	1	4

T-C
TTC

The order of cells

- start at top left
- every cell has best score considering all possible routes
- at end, highest score is best path

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0								
c	0								
a	0								
g	0								
g	0								
t	0								
t	0								
a	0								
	0								



- would also work if we went left and up

Reading the alignment

- find highest scoring cell (last row or column)
- how did we reach this cell ?
  - how did we reach preceding cell ?
  - …

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0	-3	-4	-4	4	3	1	0	2
c	0	-3	-5	-5	-2	6	0	-2	1
a	0	-3	-5	-6	-3	0	3	6	3
g	0	2	0	-6	-4	-1	6	0	6
g	0	2	4	-3	-4	-2	5	3	4
t	0	-3	-1	1	4	-2	-1	2	3
t	0	-3	-3	-1	3	1	-1	0	2
a	0	-3	-4	3	-4	0	-2	4	0
	0	0	-2	0	3	1	0	1	4

```
GGAT-CGA
-GATTC-AGGTTA
```

Trick with traceback

For each cell

- how did we reach it ? What was the preceding cell ?

		g	g	a	t	c	g	a	
	0	0	0	0	0	0	0	0	0
g	0	2	2	-3	-3	-3	2	-3	0
a	0	-3	-1	4	-3	-4	-5	4	0
t	0	-3	-3	-3	6	-1	-2	-3	4
t	0	-3	-4	-4	4	3	1	0	2
c	0	-3	-5	-5	-2	6	0	-2	1
a	0	-3	-5	-6	-3	0	3	6	3
g	0	2	0	-6	-4	-1	6	0	6
g	0	2	4	-3	-4	-2	5	3	4
t	0	-3	-1	1	4	-2	-1	2	3
t	0	-3	-3	-1	3	1	-1	0	2
a	0	-3	-4	3	-4	0	-2	4	0
	0	0	-2	0	3	1	0	1	4

```
GGAT-CGA
-GATTC-AGGTTA
```

Summary (Needleman and Wunsch)

- Alignments are paths through the matrix
- There is an astronomical number of possibilities (with gaps)
- This algorithm has visited all of them and found best
- allows for gap costs of form

$$cost = cost_{open} + (n_{gap} - 1)cost_{widen}$$

- best or only method ? wait..

Cost

- say both sequences are length n
- we have to visit n^2 cells in matrix
  - each time we have to look at a row or column of length $\approx n$
- total cost n^3 or worst cost $O(n^3)$
  - remember this for later

Smith and Waterman version

So far: global alignments

- best match, covers as much as possible

Imagine proteins with 3 domains

```
ABCDEABCDEABCDE
QRSTUVBCDEQRSTU
```

Want to see …

```
ABCDEABCDEABCDE
      ||||
QRSTUVBCDEQRSTU
```

not worth trying to align everything

Use “Smith and Waterman” method

- scoring scheme: matches positive, mismatches negative
- during traceback
  - do not just look for max score
  - start with positive score
  - stop if score goes negative
- result: “local alignments” – often most useful

Other alignment algorithms

Needleman and Wunsch / Smith Waterman

- for given problem – optimal results
- allow fancy gap penalties
- cost $O(n^3)$

Other methods

- $O(n^2)$ – very small limitation on gaps
  - this is the method used for accurate alignments

Faster

- …

Faster Seeded Methods (blast, fasta..)

Seeded

- idea: use seeds / fragments of length k
  - 11 - 28 for DNA
  - 2 - 3 for protein
- look for exact matches of query words in database
- extend if found
- time depends mainly on length $O(n)$
  most of the time no matches
- slow extension when a match is found

Seed size

- very small = lots of unimportant matches (slow)
- too big – may miss a match if there are too many changes

Fast versus slow

2 sequences (protein or DNA)

- time not an issue
- 1 000 alignments ? Time still not an issue
- $10^3 \times 10^3$ alignments ? Your decision

Databases

- non-redundant protein sequence database
  - $\approx 1.2 \times 10^8$ sequences
  - $\approx 3.6 \times 10^8$ residues
- must be fast
  - maybe occasionally miss a word
  - alignments may not be optimal

Problems so far – from DNA to proteins

We can align DNA sequences – maybe proteins

- how biological are the alignments, gaps and costs ?
  - coding versus non-coding DNA
    - 3 base pairs →1 residue

    `ACAG … … lots … CGA`…

    `AC-G … … lots … CGA` … one base deletion
      - 100's bases are shifted – amino acids in protein all wrong
  - non-coding region (binding / regulation / tRNA / rRNA..)
    - may not be so bad
- General problem – degeneracy ..

Degeneracy and Scoring

CCU, CCC, CCA, CCG are all proline (3rd position degenerate)

CCC→CCA	no problem
CCC→ACC	pro → ala (you die)

exactly the same mutation at DNA level (C→A)

DNA scoring scheme does not know about this

A rule..

- some mutations will have no effect
- some are drastic
- usually the third base in each codon is least important

Can one do better ?

Scoring protein alignments

Two aspects

- forget DNA
- account for amino acid similarity

Instead of DNA – work directly with protein sequences

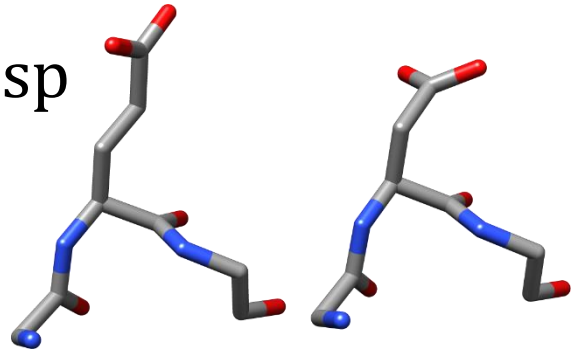
If our DNA is coding – easy to say

- CCUUCUUAU.. is pro-ser-tyr…
- immediate gain
  - CCC→CCA or similar will not be seen / affect alignment

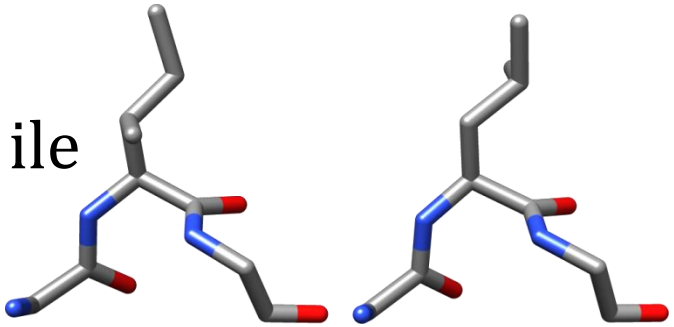
- more subtle gain…

Amino acid similarities

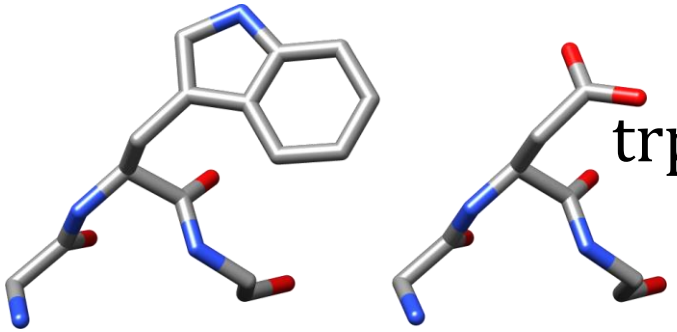
glu and asp



leu and ile



- many more similar examples
- glu →asp mutation, does it matter ? sometimes not
- trp →asp, big hydrophobic to small polar ? bad news
- relevance to alignments



trp and asp

Why we need better protein scoring

ANDREWANDRWANDRWW aligned to QNDRDW

```
ANDREWANDRWANDRWW
QNDRDW-----------

ANDREWANDR-WANDRWW
------QNDRDW------

ANDREWANDRWANDRWW
----------QNDRDW
```

- one of which is biologically more likely (E→D)
- how would we do it numerically ?

Substitution matrices

Earlier in DNA

- match = 2
- mismatch = −3

	A	C	G	T
A	2	−3	−3	−3
C	−3	2	−3	−3
G	−3	−3	2	−3
T	−3	−3	−3	2

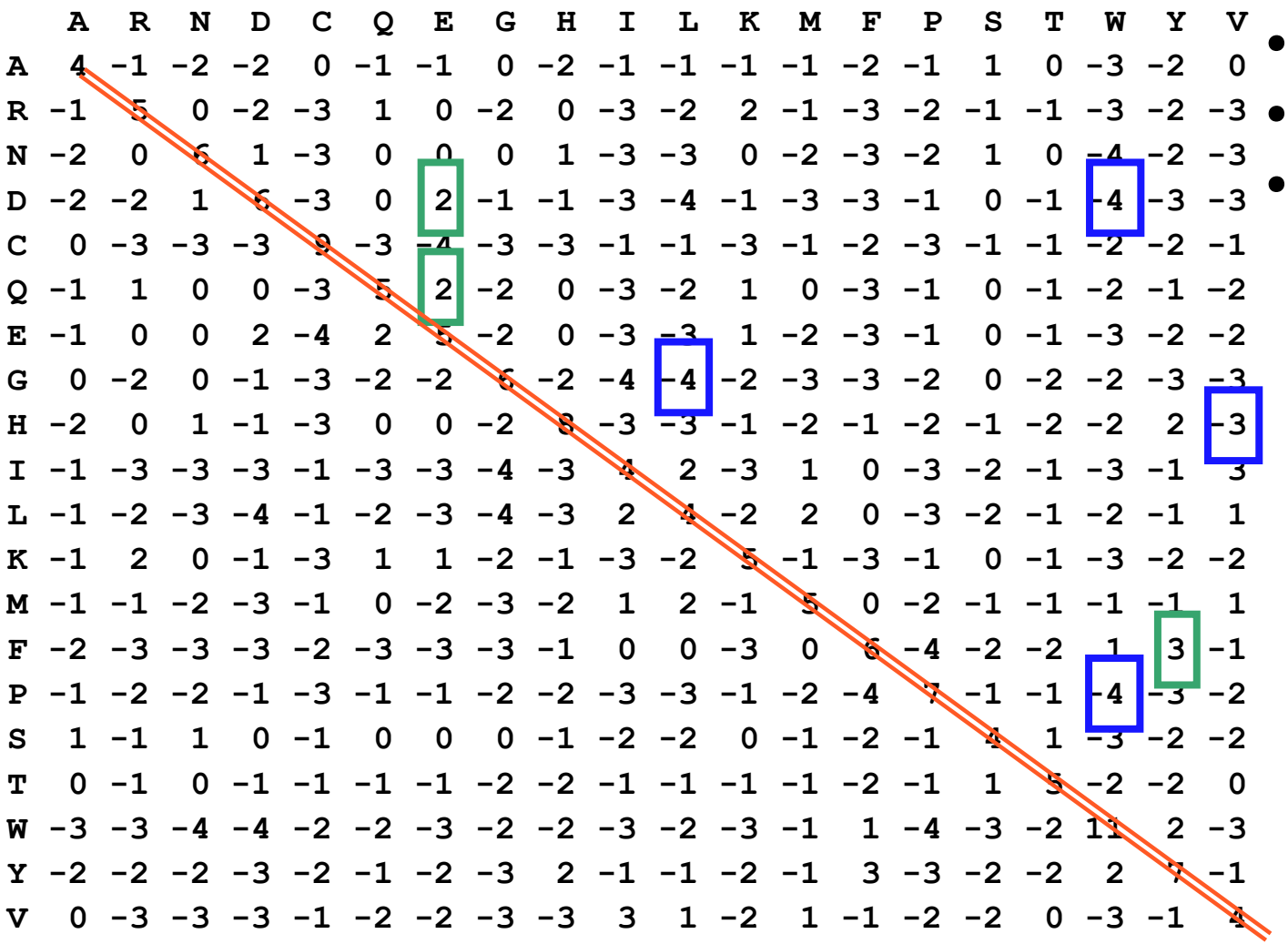
We want a matrix that says

	D	E	W	…
D	10	5	-5	
E	5	10	-5	
W	-5	-5	15	
…				

A full matrix..

A serious protein similarity matrix

blosum62:



	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

some features

- diagonal
- similar
- different

Using the score matrix

Algorithm (global alignment, local, fast, …)

- unchanged
- only scoring changes

If possible use the protein sequence rather than DNA

- not all DNA codes for proteins
- regulators, tRNA, catalytic RNA, sRNA, ..
- not possible for genomic comparisons

Automatically includes codons, amino acid similarity, ..

Where does this kind of matrix come from ?

Substitution Matrices

Many

- PAM point accepted mutations
- BLOSUM blocks substitution matrix

Philosophy

- if two amino acids are similar, we will see mutations often

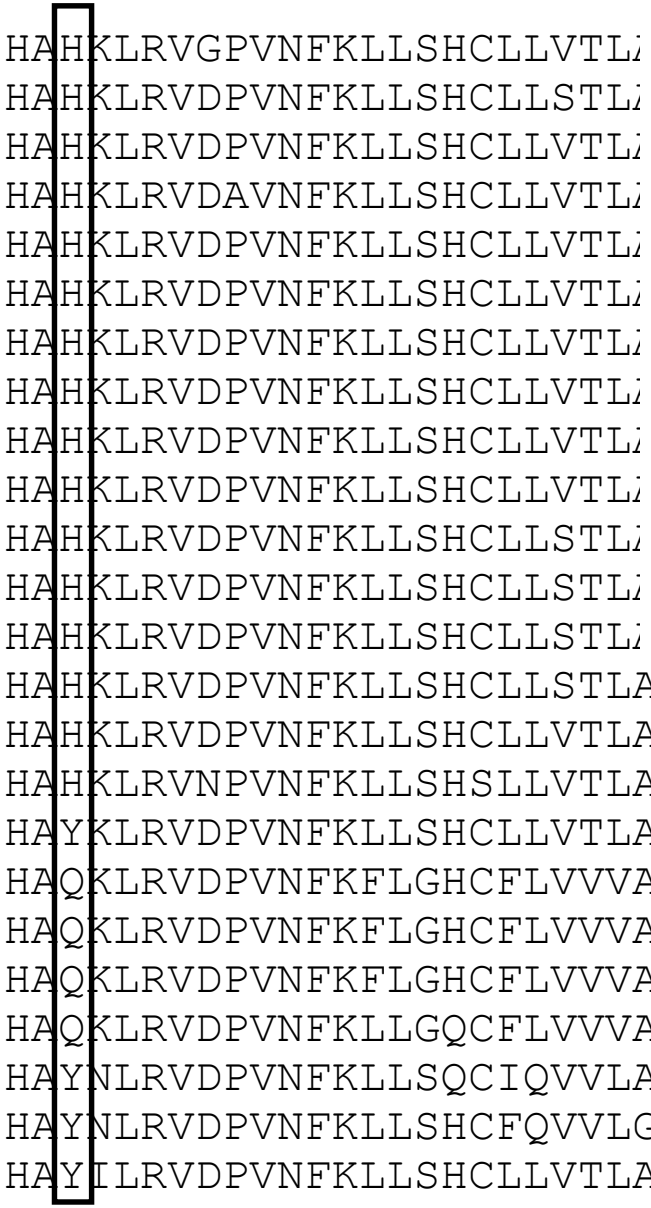
To quantify this..

- Take some very similar proteins (lots)

parts of some haemoglobins

```
HAHKLRVGPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLSTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDAVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSK
HAHKLRVDPVNFKLLSHCLLSTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSK
HAHKLRVDPVNFKLLSHCLLSTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSK
HAHKLRVDPVNFKLLSHCLLSTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSK
HAHKLRVDPVNFKLLSHCLLSTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHHPDDFNPSVHASLDKFLANVSTVLTSK
HAHKLRVNPVNFKLLSHSLLVTLASHLPTNFTPAVHANLNKFLANDSTVLTSK
HAYKLRVDPVNFKLLSHCLLVTLACHHPTEFTPAVHASLDKFFTAVSTVLTSK
HAQKLRVDPVNFKFLGHCFLVVVAIHHPSALTPEVHASLDKFLCAVGTVLTAK
HAQKLRVDPVNFKFLGHCFLVVVAIHHPSALTAEVHASLDKFLCAVGTVLTAK
HAQKLRVDPVNFKFLGHCFLVVVAIHHPSALTAEVHASLDKFLCAVGTVLTAK
HAQKLRVDPVNFKLLGQCFLVVVAIHNPSALTPEAHASLDKFLCAVGLVLTAK
HAYNLRVDPVNFKLLSQCIQVVLAVHMGKDYTPEVHAAFDKFLSAVSAVLAEK
HAYNLRVDPVNFKLLSHCFQVVLGAHLGREYTPQVQVAYDKFLAAVSAVLAEK
HAYILRVDPVNFKLLSHCLLVTLAARFPADFTAEAHAAWDKFLSVVSSVLTEK
```

parts of some haemoglobins



Consider an example column

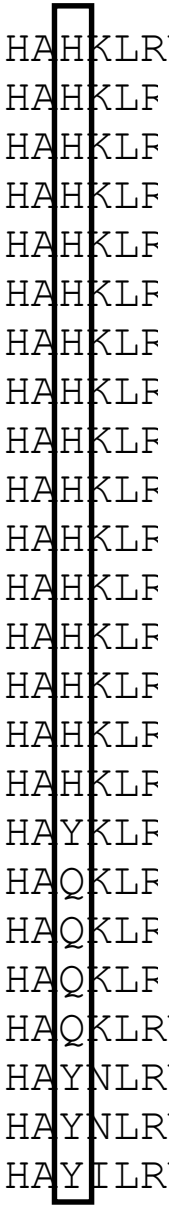
- how many pairs do we have ?
  1-2, 1-3, … 2-3, 2-4, …

$$n_{pairs} = \frac{n_{seq}(n_{seq}-1)}{2}$$

- how many columns ? n_c
- how many exchanges ? $n_c n_{pairs}$

HAHKLRVDPVNFKLLSHCLLSTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSK
HAHKLRVDPVNFKLLSHCLLVTLAAHHPDDFNPSVHASLDKFLANVSTVLTSK
HAHKLRVNPVNFKLLSHSLLVTLASHLPTNFTPAVHANLNKFLANDSTVLTSK
HAYKLRVDPVNFKLLSHCLLVTLACHHPTEFTPAVHASLDKFFTAVSTVLTSK
HAQKLRVDPVNFKFLGHCFLVVVAIHHPSALTPEVHASLDKFLCAVGTVLTAK
HAQKLRVDPVNFKFLGHCFLVVVAIHHPSALTAEVHASLDKFLCAVGTVLTAK
HAQKLRVDPVNFKFLGHCFLVVVAIHHPSALTAEVHASLDKFLCAVGTVLTAK
HAQKLRVDPVNFKLLGQCFLVVVAIHNPSALTPEAHASLDKFLCAVGLVLTAK
HAYNLRVDPVNFKLLSQCIQVVLAVHMGKDYTPEVHAAFDKFLSAVSAVLAEK
HAYNLRVDPVNFKLLSHCFQVVLGAHLGREYTPQVQVAYDKFLAAVSAVLAEK
HAYILRVDPVNFKLLSHCLLVTLAARFPADFTAEAHAAWDKFLSVVSSVLTEK

parts of some haemoglobins



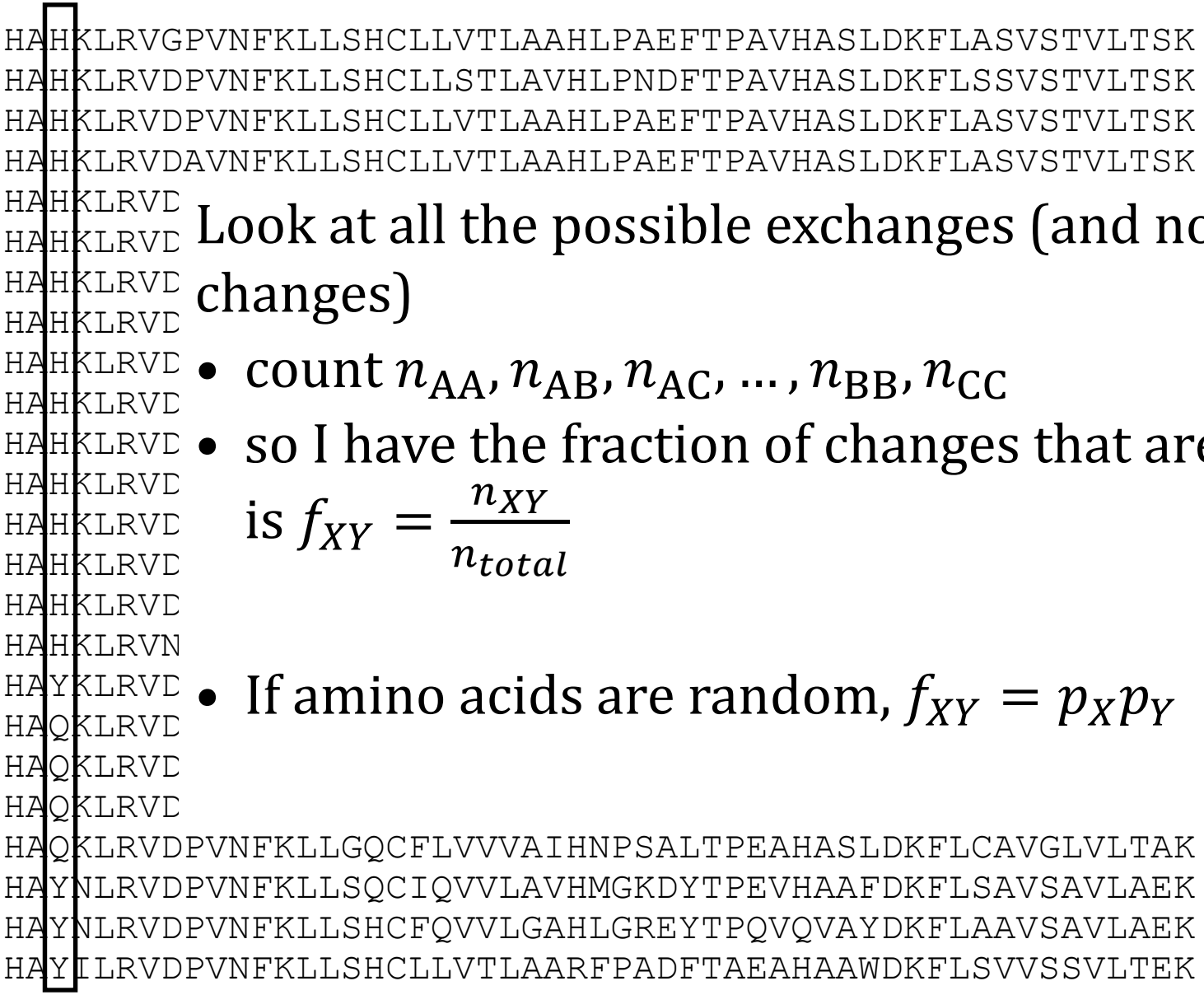
I have $n_c n_{pairs}$ exchanges

How many A↔B exchange by chance ?

- What is frequency of A ? in my alignment p_A
- What is frequency of B ? p_B
  - I would expect $p_A p_B$

How many exchanges of each type do I see in data?...

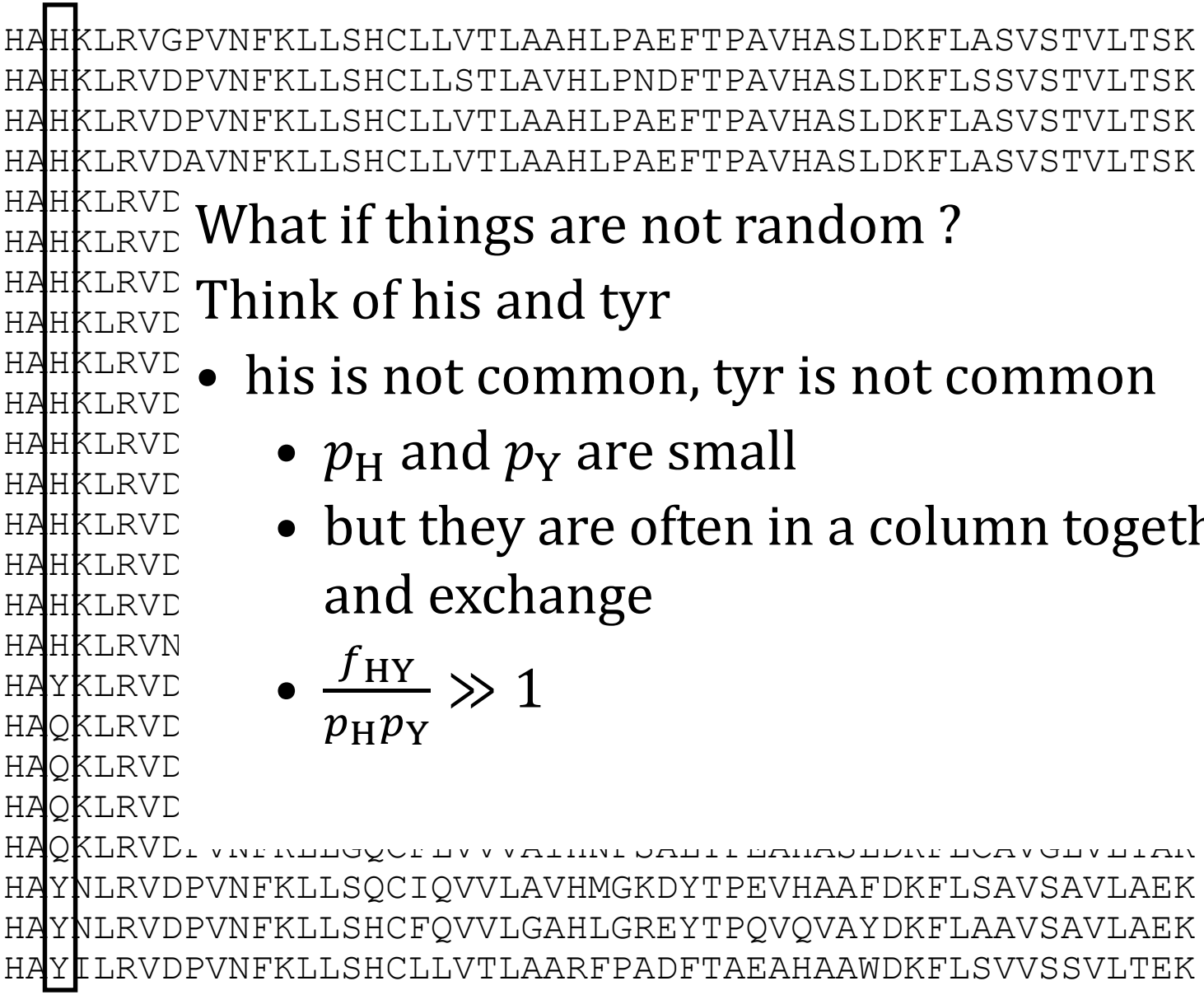
parts of some haemoglobins



Look at all the possible exchanges (and non-changes)

- count $n_{\mathrm{AA}}, n_{\mathrm{AB}}, n_{\mathrm{AC}}, \ldots, n_{\mathrm{BB}}, n_{\mathrm{CC}}$
- so I have the fraction of changes that are XY is $f_{XY} = \frac{n_{XY}}{n_{total}}$
- If amino acids are random, $f_{XY} = p_X p_Y$

parts of some haemoglobins



What if things are not random ?

Think of his and tyr

- his is not common, tyr is not common
  - p_{H} and p_{Y} are small
  - but they are often in a column together and exchange
  - $\dfrac{f_{\mathrm{HY}}}{p_{\mathrm{H}} p_{\mathrm{Y}}} \gg 1$

A rare exchange

What about Leu and Asp ? (big hydrophobic / small charged)

- both are common, but
  - you do not often see them in a column together
  - $\frac{f_{\mathrm{LD}}}{p_{\mathrm{L}} p_{\mathrm{D}}} \ll 1$
  - $\log_2\left(\frac{f_{\mathrm{LD}}}{p_{\mathrm{L}} p_{\mathrm{D}}}\right) < 0$ and previous example $\log_2\left(\frac{f_{\mathrm{HL}}}{p_{\mathrm{H}} p_{\mathrm{L}}}\right)$

A substitution matrix is just logarithms of what you see, divided by what you expect to see

- positive are likely exchanges
- negative are ..
- the biggest elements are diagonal (no exchange)

Calculating a substitution matrix

- We have all the probabilities p_{AB} and p_{AA}
- matrix element AB is $\log_2(p_{AB})$ why $\log_2$?
- is my example enough ?
  - needs much more data so as to get good probabilities
  - real calculation does not use Hb – uses 100s of proteins

Different matrices

Degree of homology

- if two sequences are very similar most residues not changed
- longer evolutionary time – many things change

Longer evolutionary times – matrix multiplication

So far: probability of one mutation A→B

- in longer evolutionary time
  - D→E→D→W→D…
    - multiple mutations
    - probability of conservation is lower (diagonal elements smaller)
    - all off-diagonal elements will be bigger
- more formally - long time p_{long} is $p \times p \times p \times \ldots$

How to account for this ?

- take matrix (like PAM) and do matrix multiplication

  M M M …
- result: a set of matrices
  - PAM10, PAM20, …

Longer evolutionary times - BLOSUM

BLOSUM

- not like PAM
  - works differently (Übung to the topic)
- less theory – more empiricism
  - uses different data for counting mutations

Nomenclature

short time	→	longer time
conserved		less conserved
PAM10		PAM20
BLOSUM80		BLOSUM62

Are these different matrices useful ?

In principle, yes

- looking for similar proteins – use blosum80
- more remote ? – use blosum62
- …

- in practice ?
- better way to find remote homologues (soon)

Now

- should you believe any of this ?

Are alignments correct ?
Is there a correct alignment ?

Are alignments correct ?

1. algorithmically
2. mathematically
3. is my model sufficient ?

Is there a correct alignment ?

1. structurally
2. biologically / evolutionarily

Details..

Are alignments correct ? (algorithm)

Yes

(Needleman and Wunsch, Smith-Waterman)

- We have a scoring scheme
  (identity, blosum similarity matrix)

- We find the alignment which maximises the score

In terms of maths / algorithm everything is correct

Are alignments correct ? (alternative answers)

not necessarily

- blosum matrix comes from counting mutations
  - we find the most likely substitutions

Align `ANDREW` and `ANDRWQANDRKWSANDRWWC`

```
ANDR-WQANDRKWSANDRWWC
ANDREW---------------  guess 1 [ includes gap ]
-------ANDREW--------  guess 2
--------------ANDREW-  guess 3
```

We can find the most likely

- often correct, sometimes wrong

Are alignments correct ? (alternative answers)

How good is "most likely" ?

- closely related sequences – very good
- distant relatives (tuna / horse myoglobin) – will have mistakes
- repetitive sequences **ACDEACQEACDDAQDE** (think **ACDE** × 4)

Are alignments correct ? (model)

My model for evolution

- mostly point mutations (DNA level)
- rarely a deletion/insertion

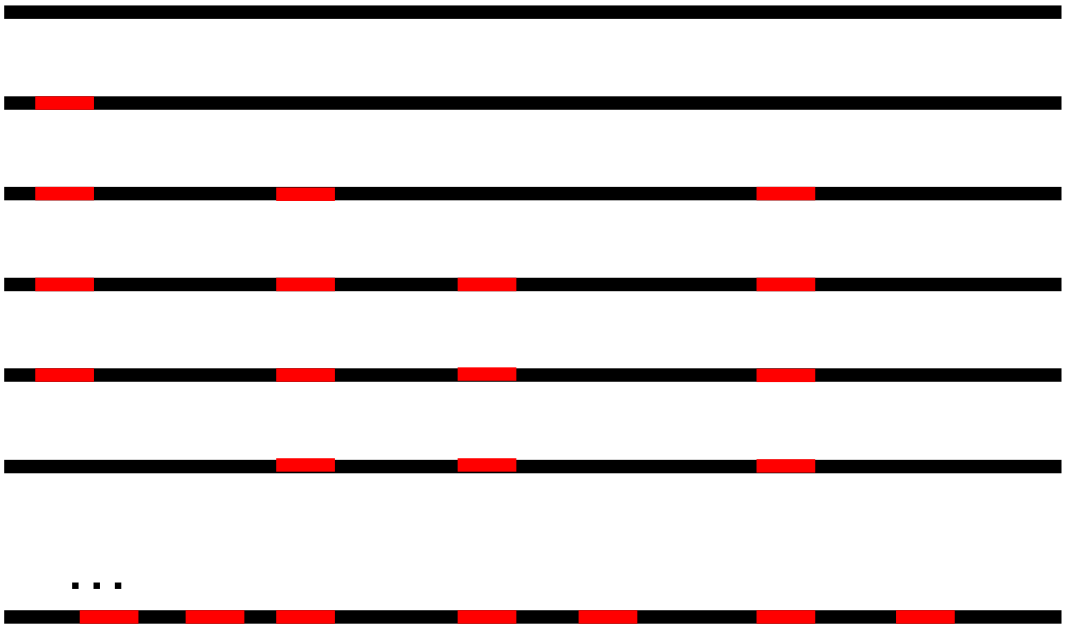
Is this enough ? Usually

- not always
  - fusion, insertion of random DNA, splicing, duplications..

Our model is good, but does not cover rare events

Is there a correct alignment ? evolutionary

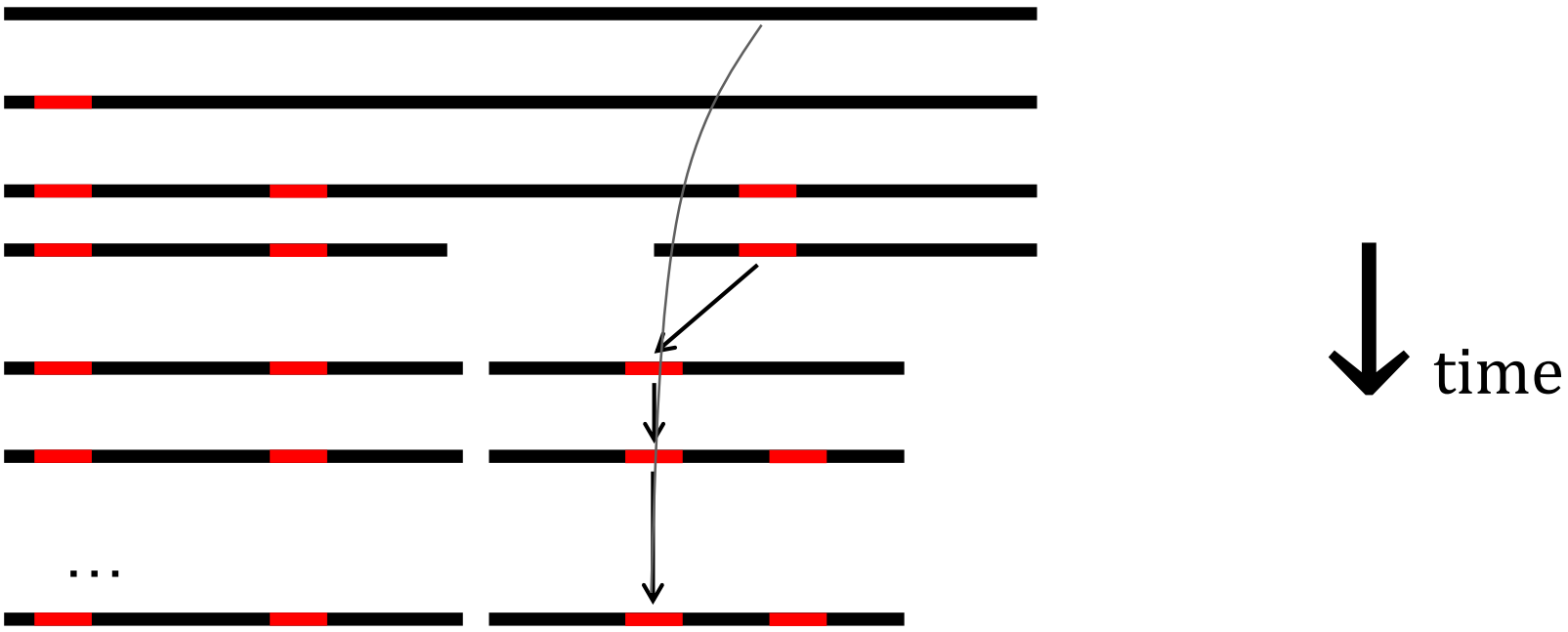
Yes



There is a correct sequence alignment

- with gaps ?

Is there a correct alignment ? evolutionary



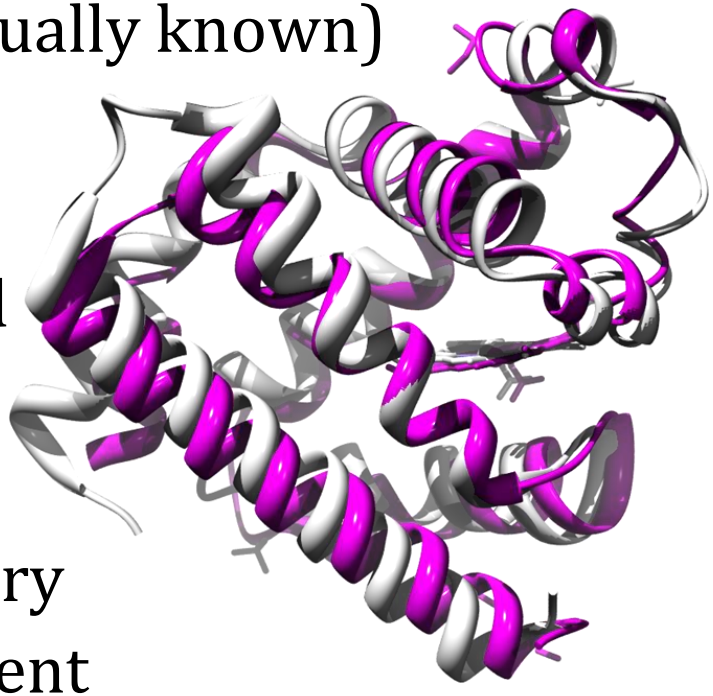
There is a correct alignment

- the evolutionary history of each residue / base

Is there a correct alignment ? (structure)

Yes

- every protein has a structure (not usually known)



- structures can be aligned
- corresponding residues can be found

This alignment may

- not correspond to evolutionary history
- not agree with best sequence alignment

  (imagine deleting some residues on solvent exposed helix)

Correct alignments - summary

There is something of a correct alignment

- evolutionary history or structure (both are good definitions)

The algorithm maximises a score

- it is numerically correct
- it will not always correspond to the correct alignment (evolution or structure)

Alignments and searches

The practical goals - different problems

1. find the best alignment (modelling, predictions of important residues)
   - we care about alignment methods

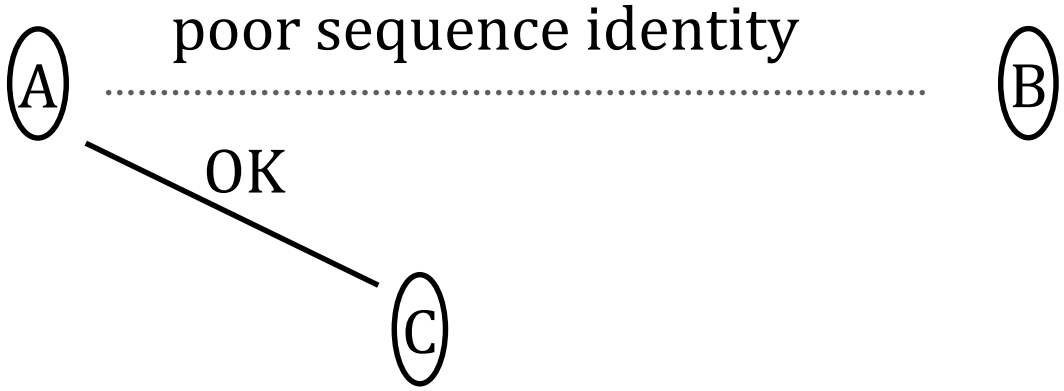
2. find related known proteins (what kind of protein do I have ?)

- we need sensitive search methods

iterated searches (psi-blast)

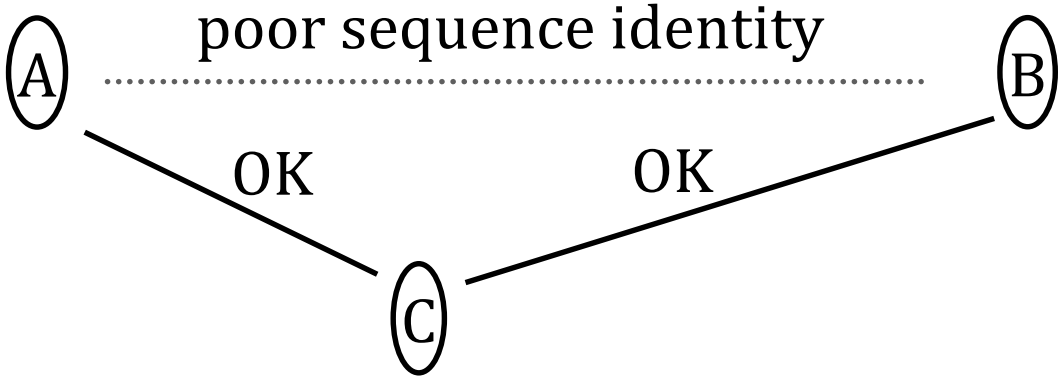
Search with protein A and find a very remote protein B

but there is another protein C



search with C finds B

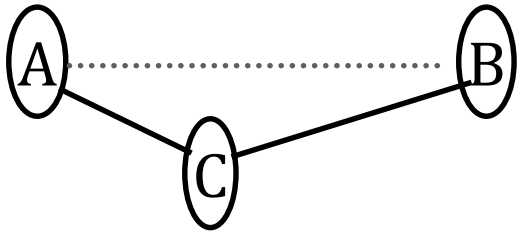
- the original AB relation is believable
- how to automate this ?



iterated searches (psi-blast)

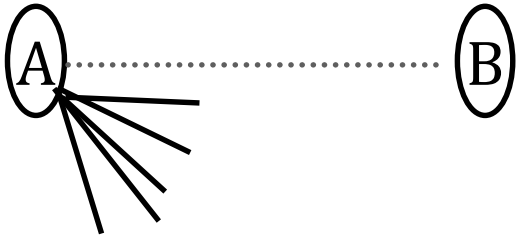
Searching with "A" finds lots of homologues

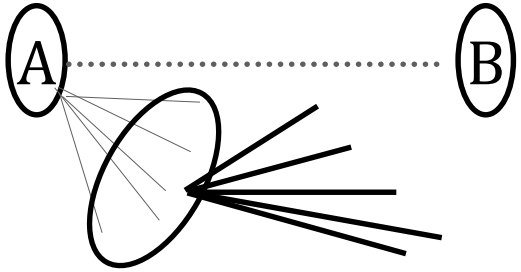
- cannot start a search with each



Alternative

- find all the homologues to A
- build an average sequence (profile)
- from this profile – repeat search
- build new average / repeat





Result

- at each step
- include reliable homologues
- eventually A→ B may be found

iterated searches (psi-blast)

In practice

- really only one program (+ web page) NCBI blast / psiblast
- most significant advance in finding remote homologues in a decade

sequence identity / similarity / significance

Significance

- I find a homologue – is it evolutionarily related or just noise ?
  - probability estimations later
- how important is 10% sequence identity ? 90 % ?
- is 25 % identity in DNA as useful as in a protein ?

First principles DNA

- what would you expect by chance ?

  GGATCGA

  GATTCAGGTTA

- At each position ¼ chance of a match
  - average 25 % sequence identity with random DNA

    … very wrong

Naïve identity expectation – base usage

Two problems

1. uneven character frequency
2. gaps

Character frequency – what if I have a two letter alphabet ?

- a world with two bases
  `GCGGCGCGCCGCGCGCGCGC`
  average sequence identity 50 %

- a world with usually two bases - sometimes `A` or `T`
  `GCGACGCGTCGCGCGTTCGCGC`
  average sequence identity : a bit less than 50 %

- a four letter alphabet
  `GCGACACGTCGTGAGTTCTTGC` nearly 25 %

Naïve identity expectation – base usage

One extreme

- two types of base – 50 % sequence identity expected

Other extreme

- Random sequences – 25 % identity

Our world – somewhere between

How significant ?

- malaria is about ⅓ GC (not ½)
- *Streptomyces coelicolor* is 72 % GC
- GC differs between organisms, coding/non-coding regions

Consequence

- even randomly sampled sequences, will have > 25% sequence identity

Naïve identity expectation - gaps

- ungapped: 2 matches from 9 aligned (22 %)
  ```
  GGATCGCAC
  GACTGAGGTTA
  ```
- one gap: 3 matches 8 aligned (38 %)
  ```
  GGATCGCAC
  GACT-GAGGTTA
  ```
- more gaps: 4 matches from 6 positions (50 %)
  ```
  GGATCGCAC
  GACT-G-AGGTTA
  ```
- more gaps: 5 matches from 6 positions (83 %)
  ```
  GGATC-GCAC
  G-A-CTG-AGGTTA
  ```

The more gaps one allows - the higher the identity

- one can make score arbitrarily good

Protein – random matches

20 amino acids

- naïve expectation – 5 %

	%
ala	8.4
leu	8.3
gly	7.8
trp	1.5
cys	1.7

Proteins are not like a 20 character alphabet:

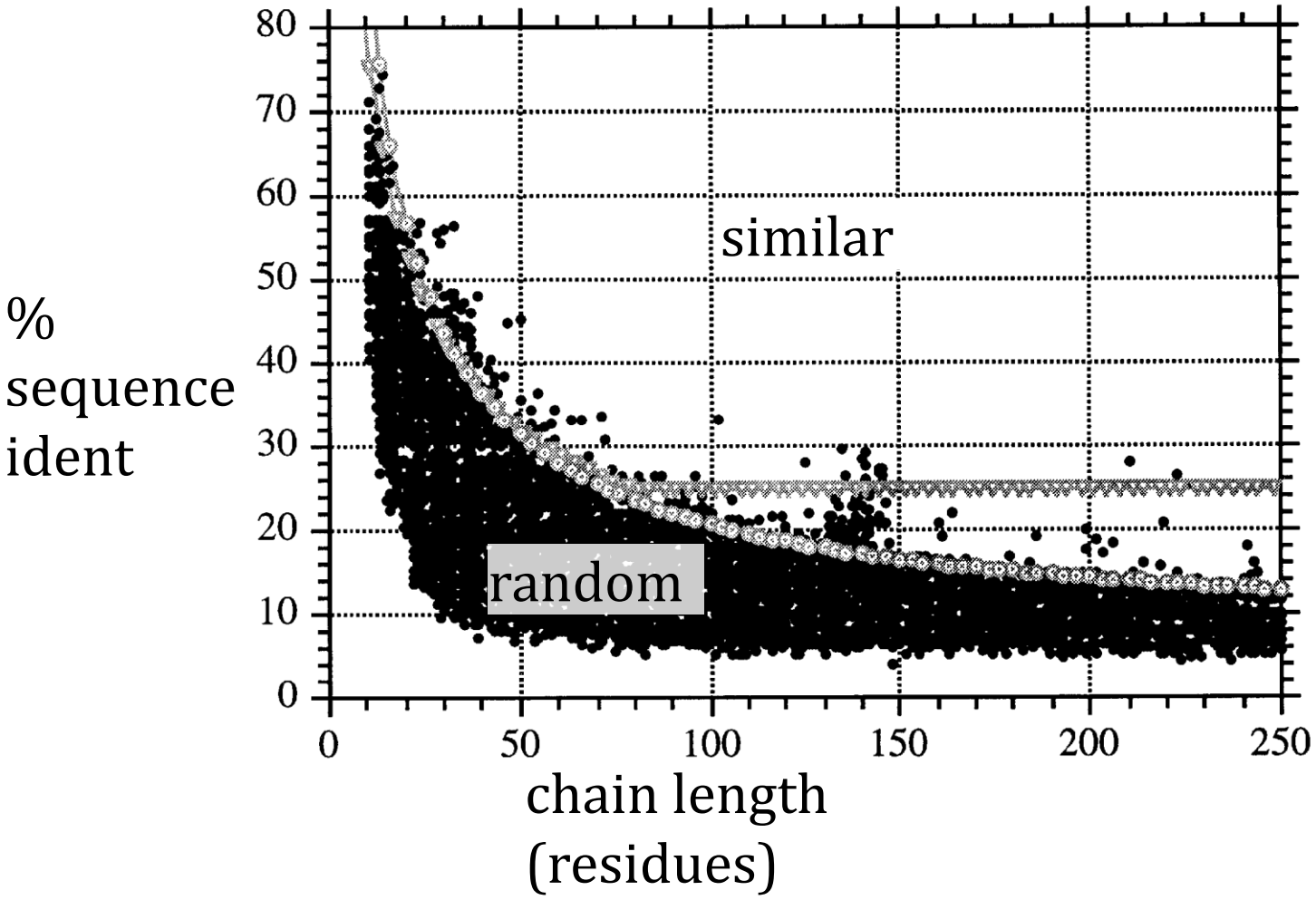
- varies between organisms
- varies between cell compartments, soluble, membrane bound…

Practical result - random sequences, realistic gaps

- 20 to 25 % identity by chance
- depends on length..

protein size and identity

- small proteins – need 30 % to believe they are related
- big proteins < 20 % , almost certainly related



Rost, B.Prot Eng, 12,85–94, 1999

Searching for significance – homologues

Try easy steps first

- simple searches first
- see if enough information is found

- gradually go to more sensitive methods (slightly more error prone)

Use the “least speculative” methods first

- simple blast searches before iterated ones

What can one do ?

- Are two proteins related ? Can I transfer information ?
- For protein A, find the corresponding residue in protein B
  - mutation susceptibility, transferring information on active sites
- Protein modelling (winter semester)
- Basis of multiple sequence alignments (next topic)