

DATABASE SEARCHING

RESEARCH QUESTION

IDENTIFY HOMOLOGS
PREDICT FUNCTION

QUERY

DNA
PROTEIN (PREFERRED)

ALGORITHM

SPEED, SENSITIVITY
GLOBAL/LOCAL,
SCORING SCHEME

DATABASE

RESULTS
LIST OF SIMILAR SEQUENCES

DATABASE SEARCHING

- Interpret results
- Optionally change search strategy
 - Change algorithm
 - Change scoring scheme



BLAST®

Home Recent Results Saved Strategies Help

NCBI BLAST Home

BLAST finds regions of similarity between biological sequences

New DELTA-BLAST, a more sensitive p

BLAST Assembled Genomes

Find Genomic BLAST pages:

Enter organism name or id--completions will be suggested

GO

- | | | |
|--|---|--|
| <input type="checkbox"/> Human | <input type="checkbox"/> Rabbit | <input type="checkbox"/> Zebrafish |
| <input type="checkbox"/> Mouse | <input type="checkbox"/> Chimp | <input type="checkbox"/> Clawed frog |
| <input type="checkbox"/> Rat | <input type="checkbox"/> Guinea pig | <input type="checkbox"/> Arabidopsis |
| <input type="checkbox"/> Cow | <input type="checkbox"/> Fruit fly | <input type="checkbox"/> Rice |
| <input type="checkbox"/> Pig | <input type="checkbox"/> Honey bee | <input type="checkbox"/> Yeast |
| <input type="checkbox"/> Dog | <input type="checkbox"/> Chicken | <input type="checkbox"/> Microbes |

Basic BLAST

Choose a BLAST program to run.

- | | |
|----------------------------------|---|
| nucleotide blast | Search a nucleotide database using a nucleotide query.
<i>Algorithms:</i> blastn, megablast, discontiguous megablast |
| protein blast | Search protein database using a protein query.
<i>Algorithms:</i> blastp, psi-blast, phi-blast, diamond search |
| blastx | Search protein database using a translated nucleotide query. |
| tblastn | Search translated nucleotide database using a protein query. |
| tblastx | Search translated nucleotide database using a translated protein query. |

HEURISTIC SEQUENCE ALIGNMENT

- Optimal sequence alignment and statistics do not scale to database searches
 - Needleman-Wunch $O(nm)$
 - Smith-Waterman $O(nm)$

		c	a	c	g	t	a	t	
		0	-1	-2	-3	-4	-5	-6	-7
		-1	1	0	-1	-2	-3	-4	-5
		-2	0	1	0	0	-1	-2	-3
		-3	-1	0	2	1	0	-1	-2
		-4	-2	0	1	2	1	1	0

HEURISTIC SEQUENCE ALIGNMENT

- Gotoh (1982) simplified the dynamic programming algorithm
 - Introduced the affine gap penalty
 - He reasoned that two of the terms that are maximized in the dynamic programming algorithm depend only on the values in the **current and previous row and column**

NO NEED TO CALCULATE
THE ENTIRE MATRIX

Improved dynamic programming algorithm of Gotoh (1982)

The similarity score is written as

$$S_{i,j} = \max \{ S_{i-1,j-1} + s_{i,j}, P_{i,j}, Q_{i,j} \}, \text{ where}$$

$$P_{i,j} = \max_{1 \leq x \leq i} \{ S_{i-x,j} - w_x \}, \text{ and } Q_{i,j} = \max_{1 \leq x \leq j} \{ S_{i,j-x} - w_x \}$$

P may be obtained in a single step since,

$$P_{i,j} = \max_{2 \leq x \leq i} \{ S_{i-1,j} - w_1, \max_{1 \leq x \leq i} (S_{i-x,j} - w_x) \}$$

$$= \max_{1 \leq x \leq i-1} \{ S_{i-1,j} - w_1, \max_{1 \leq x \leq i-1} (S_{i-1-x,j} - w_{x+1}) \}$$

$$= \max_{1 \leq x \leq i-1} \{ S_{i-1,j} - w_1, \max_{1 \leq x \leq i-1} (S_{i-1-x,j} - w_x - r) \}$$

$$= \max \{ S_{i-1,j} - w_1, P_{i-1,j} - r \}$$

where the penalty of a gap of length x is given by,

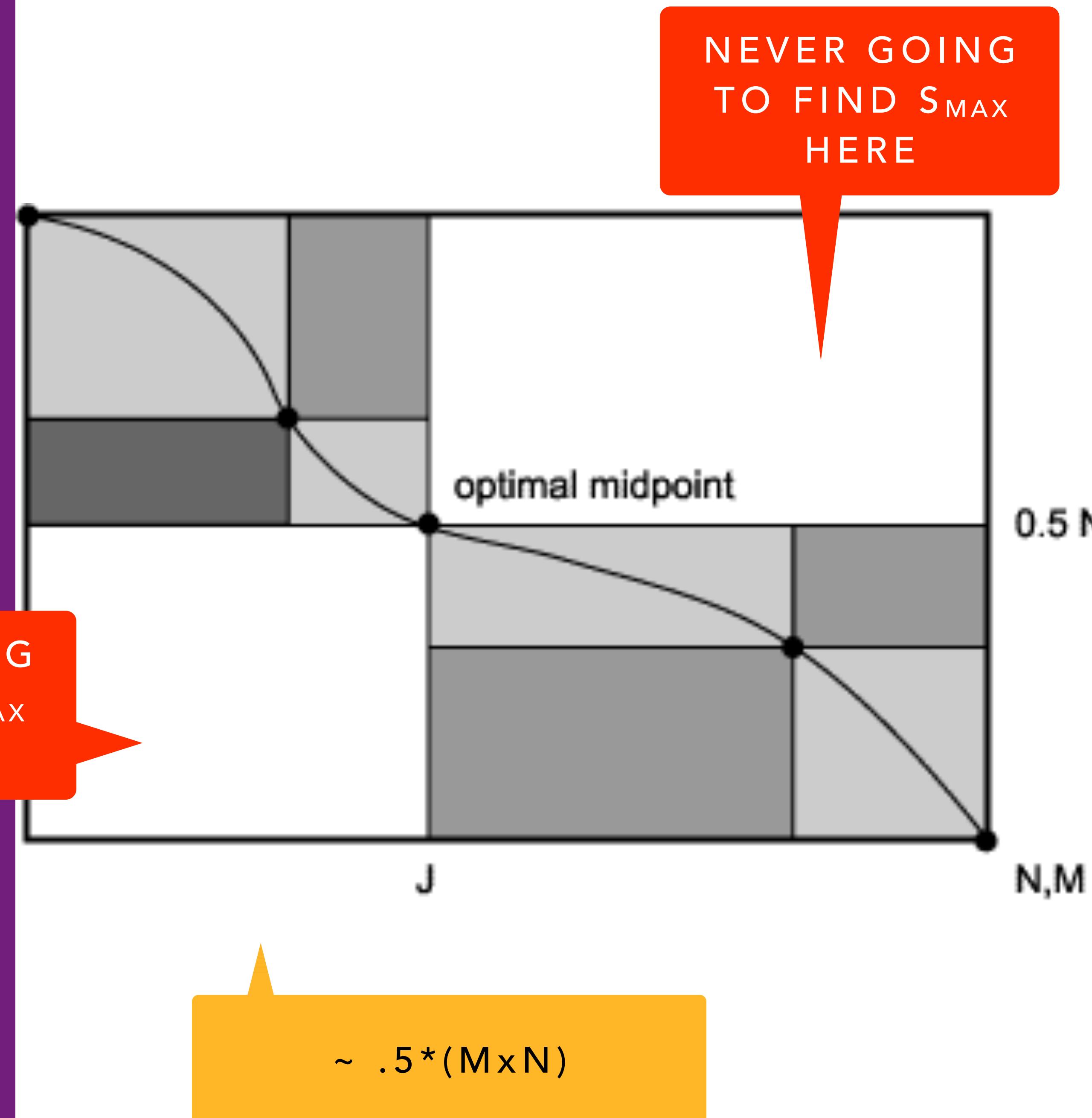
$$w_x = g + rx$$

and g is the gap opening penalty and r the gap extension penalty.

HEURISTIC SEQUENCE ALIGNMENT

- Myers and Miller (1988)

- Improved the algorithms so both global and local alignment require less time and space
- Start at beginning/end or midpoint
- Join alignments



HEURISTIC SEQUENCE ALIGNMENT

- Why are heuristic approaches necessary?
 - Dynamic programming requires order N^2L computations where
 - **N** is size of the query sequence
 - **L** is the size of the database
- Given size of databases, more efficient methods needed

HEURISTIC SEQUENCE ALIGNMENT

- How are heuristic approaches applied?
 - Feedback from current result guides future analytical direction
 - Search a small fraction of the cells in possible search space
 - Still maintain all the high scoring alignments

HEURISTIC SEQUENCE ALIGNMENT

PART OF ITERATIVE STRATEGY

- Heuristic methods are **not guaranteed** to find the optimal solution
- Can be much faster
 - >>50X improvement in speed/memory usage

NECESSARY TO SEARCH AGAINST THE
NCBI SEQUENCE DATABASE

HEURISTIC SEQUENCE ALIGNMENT

- 2 best known approaches
 - FASTA [Pearson & Lipman, 1988]
 - BLAST [Altschul et al., 1990]

TRADEOFFS OF
USING THE
HEURISTIC
METHODS?

ACCURACY

HEURISTIC SEQUENCE ALIGNMENT

- Global or Local?
 - Both local and global alignment methods may be applied to database searching
 - Local alignment methods are more useful since they do not make the assumption that the query protein and database sequence are of similar length

DNA VS PROTEIN SEARCHES

DNA VS PROTEIN SEARCHES

- Protein similarity infers homology
 - Conserved alphabet
- DNA similarity has less sensitivity
 - Reduced alphabet

16 DNA
SEQUENCES
THAT GIVE THE
SAME PROTEIN
SEQUENCE

Peptide	(1)	MET	LYS	PRO	HIS
DNA	(1)	ATG	AAA	CCT	CAT
	(2)	ATG	AAG	CCT	CAT
	(3)	ATG	AAA	CCC	CAT
	(4)	ATG	AAG	CCC	CAT
	(5)	ATG	AAA	CCA	CAT
	(6)	ATG	AAG	CCA	CAT
	(7)	ATG	AAA	CCG	CAT
	(8)	ATG	AAG	CCG	CAT
	(9)	ATG	AAA	CCT	CAC
	(10)	ATG	AAG	CCT	CAC
	(11)	ATG	AAA	CCC	CAC
	(12)	ATG	AAG	CCC	CAC
	(13)	ATG	AAA	CCA	CAC
	(14)	ATG	AAG	CCA	CAC
	(15)	ATG	AAA	CCG	CAC
	(16)	ATG	AAG	CCG	CAC

DNA VS PROTEIN SEARCHES

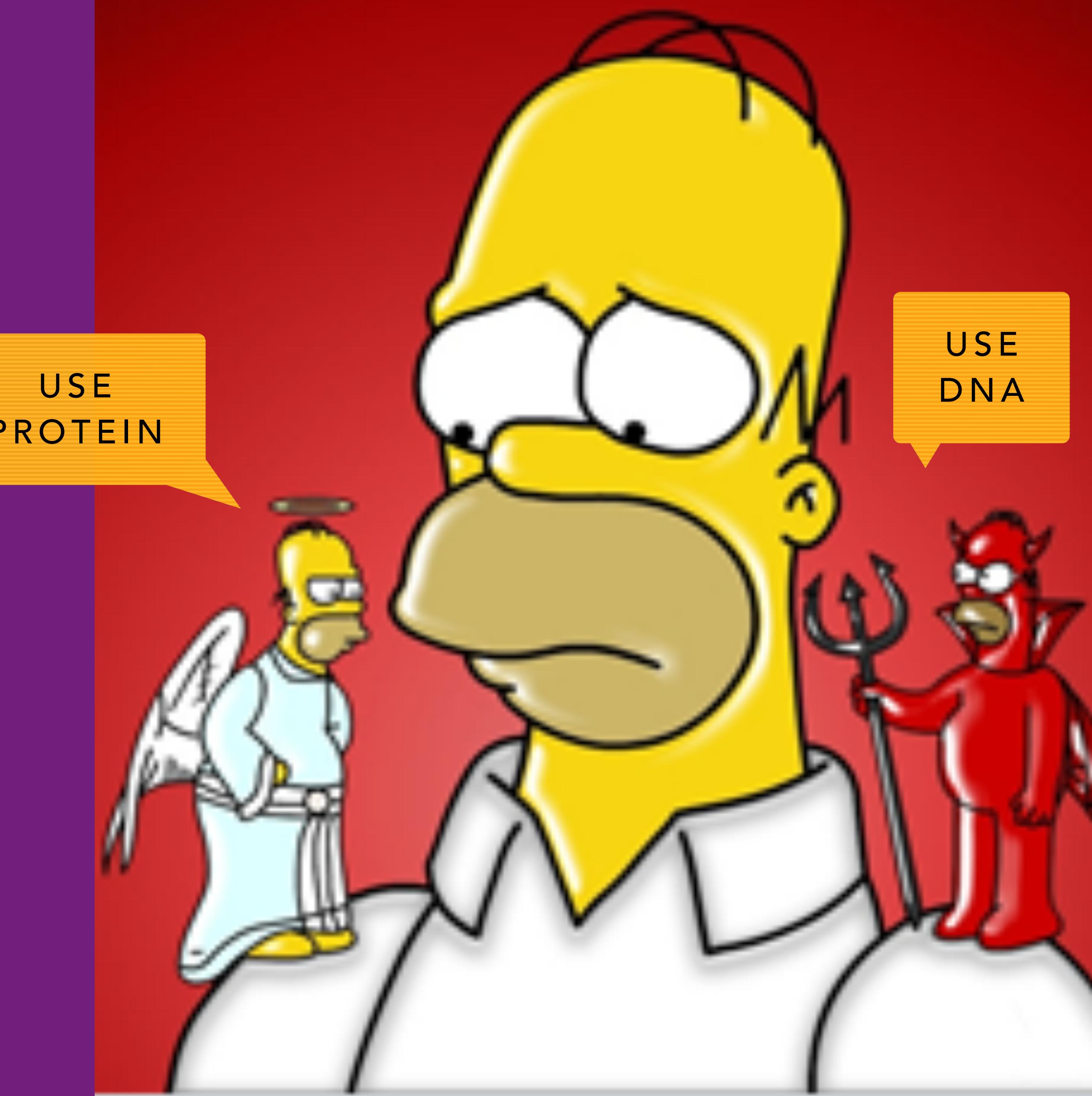
- Translate DNA to protein before search?
 - Inherent information loss from degenerate codons
 - Different DNA sequences can encode same protein

Peptide	(1)	MET	LYS	PRO	HIS
DNA	(1)	ATG	AAA	CCT	CAT
	(2)	ATG	AAG	CCT	CAT
	(3)	ATG	AAA	CCC	CAT
	(4)	ATG	AAG	CCC	CAT
	(5)	ATG	AAA	CCA	CAT
	(6)	ATG	AAG	CCA	CAT
	(7)	ATG	AAA	CCG	CAT
	(8)	ATG	AAG	CCG	CAT
	(9)	ATG	AAA	CCT	CAC
	(10)	ATG	AAG	CCT	CAC
	(11)	ATG	AAA	CCC	CAC
	(12)	ATG	AAG	CCC	CAC
	(13)	ATG	AAA	CCA	CAC
	(14)	ATG	AAG	CCA	CAC
	(15)	ATG	AAA	CCG	CAC
	(16)	ATG	AAG	CCG	CAC

"IT DEPENDS..."

DNA VS PROTEIN SEARCHES

- If given the option, use protein sequences for database similarity searches when possible

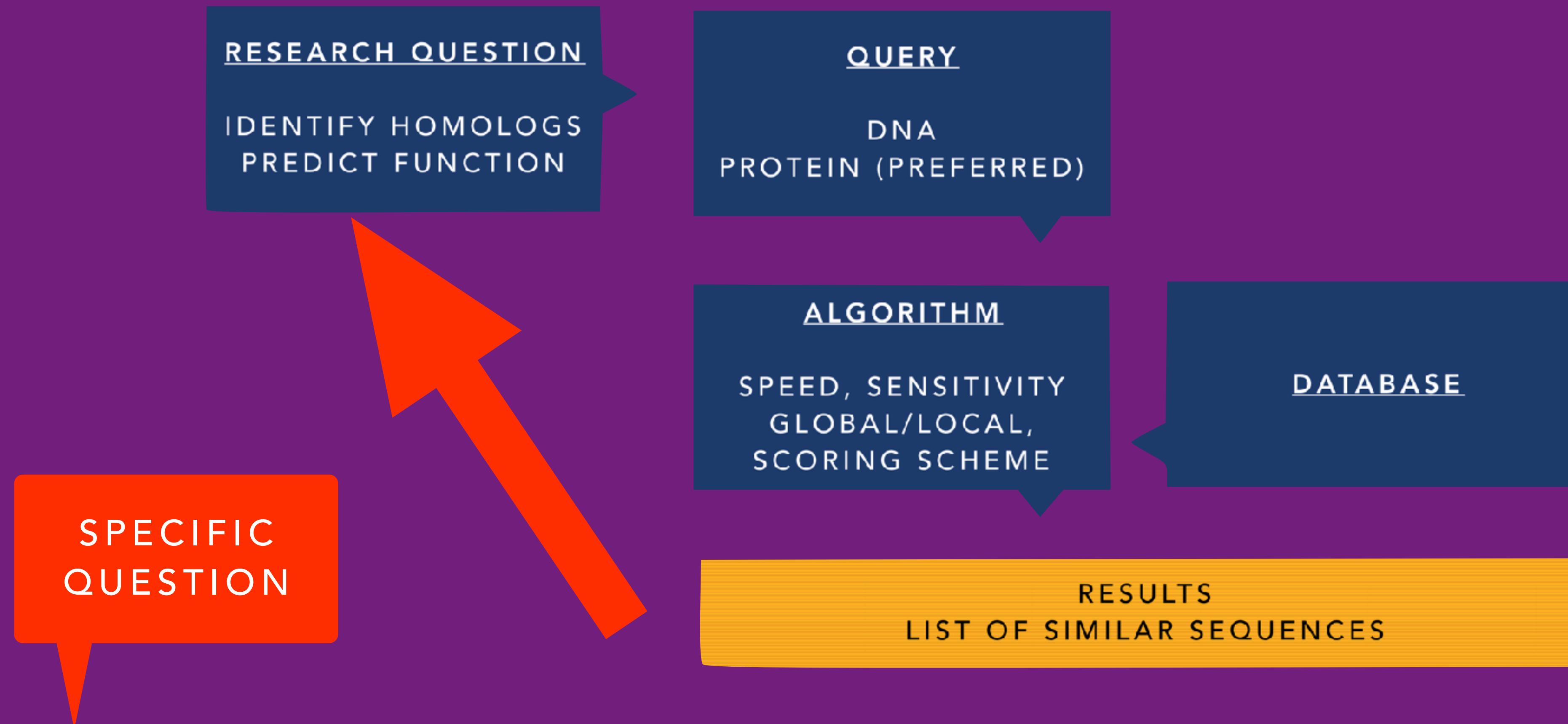


DNA VS PROTEIN SEARCHES

- Why?
 - DNA will have more random matches (bad matches; skew statistics)
 - DNA databases are larger and grow faster
 - DNA uses identity matrices; protein uses scoring matrices (more sensitivity)
 - Protein sequences diverge less than DNA encoding them

EVALUATING A DATABASE SEARCH

EVALUATING A DATABASE SEARCH



- How can we evaluate a database search?

EVALUATING A DATABASE SEARCH

- If we were to develop a new algorithm, how could we compare it to existing algorithms?

EVALUATING A DATABASE SEARCH

WHEN WE LOOK
AT A RESULT
COMPARED TO
EXPECTED
OUTCOME

Classified as

Positive

Negative

True Positive

False Negative

False Positive

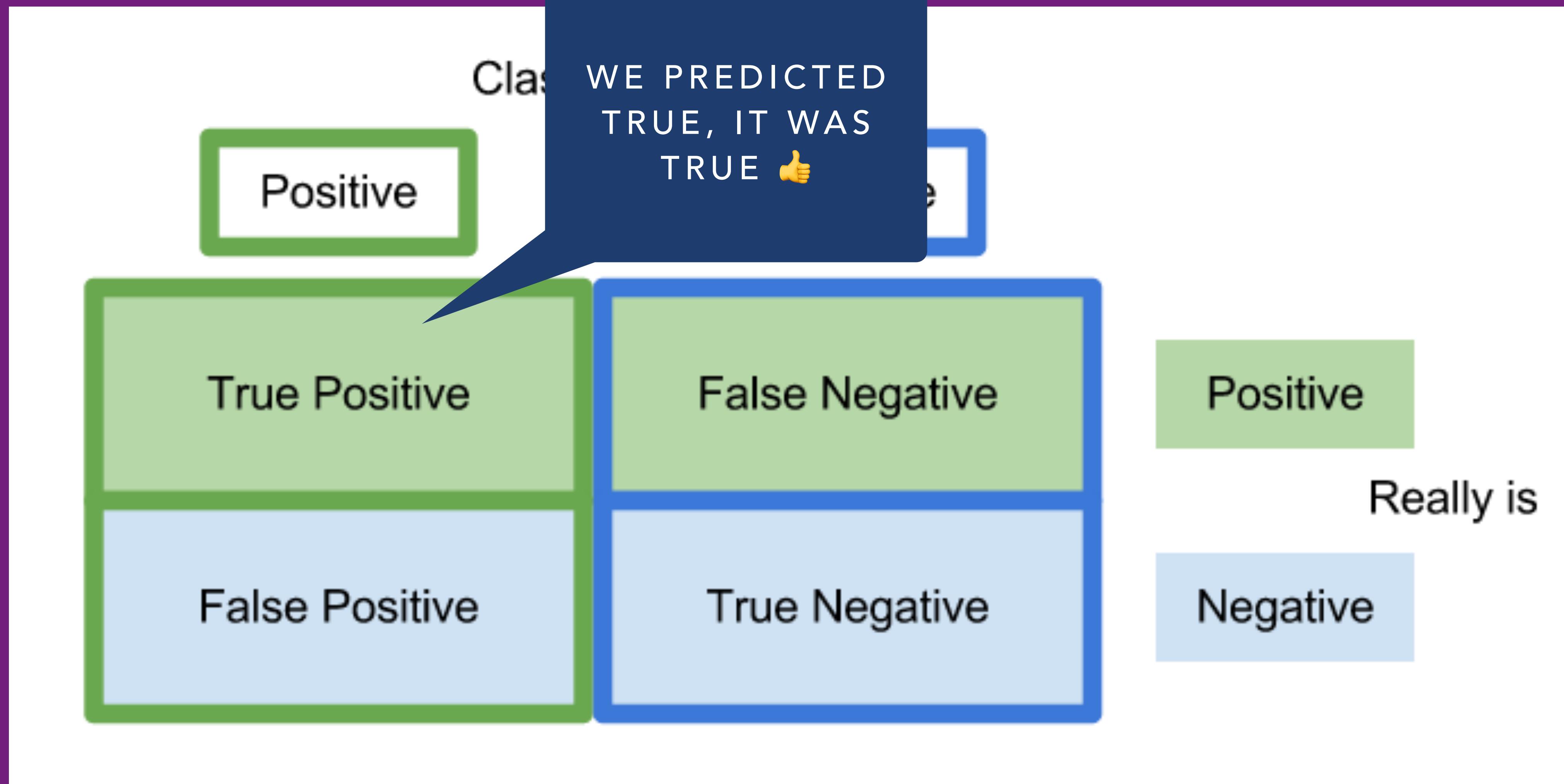
True Negative

Positive

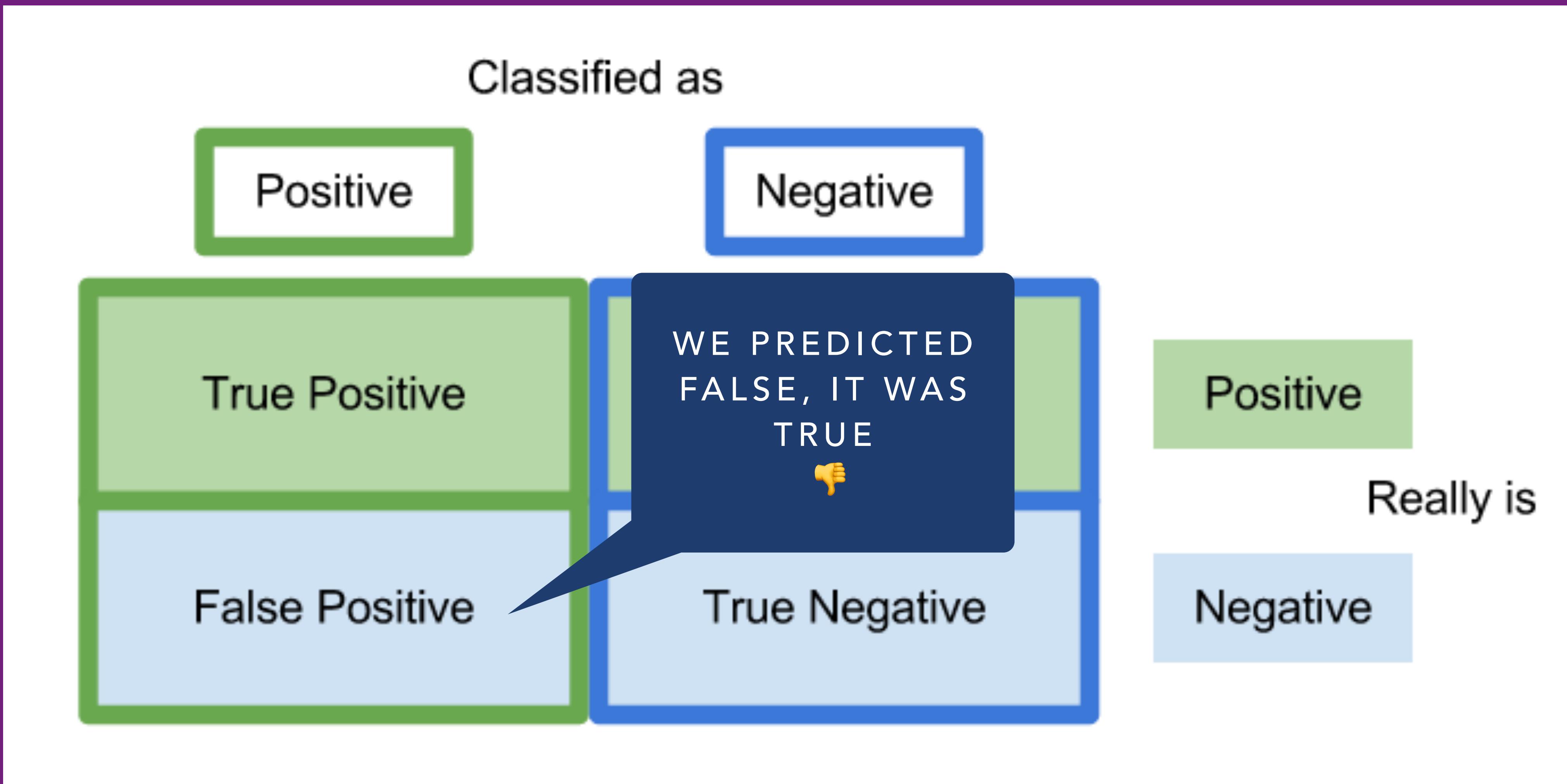
Really is

Negative

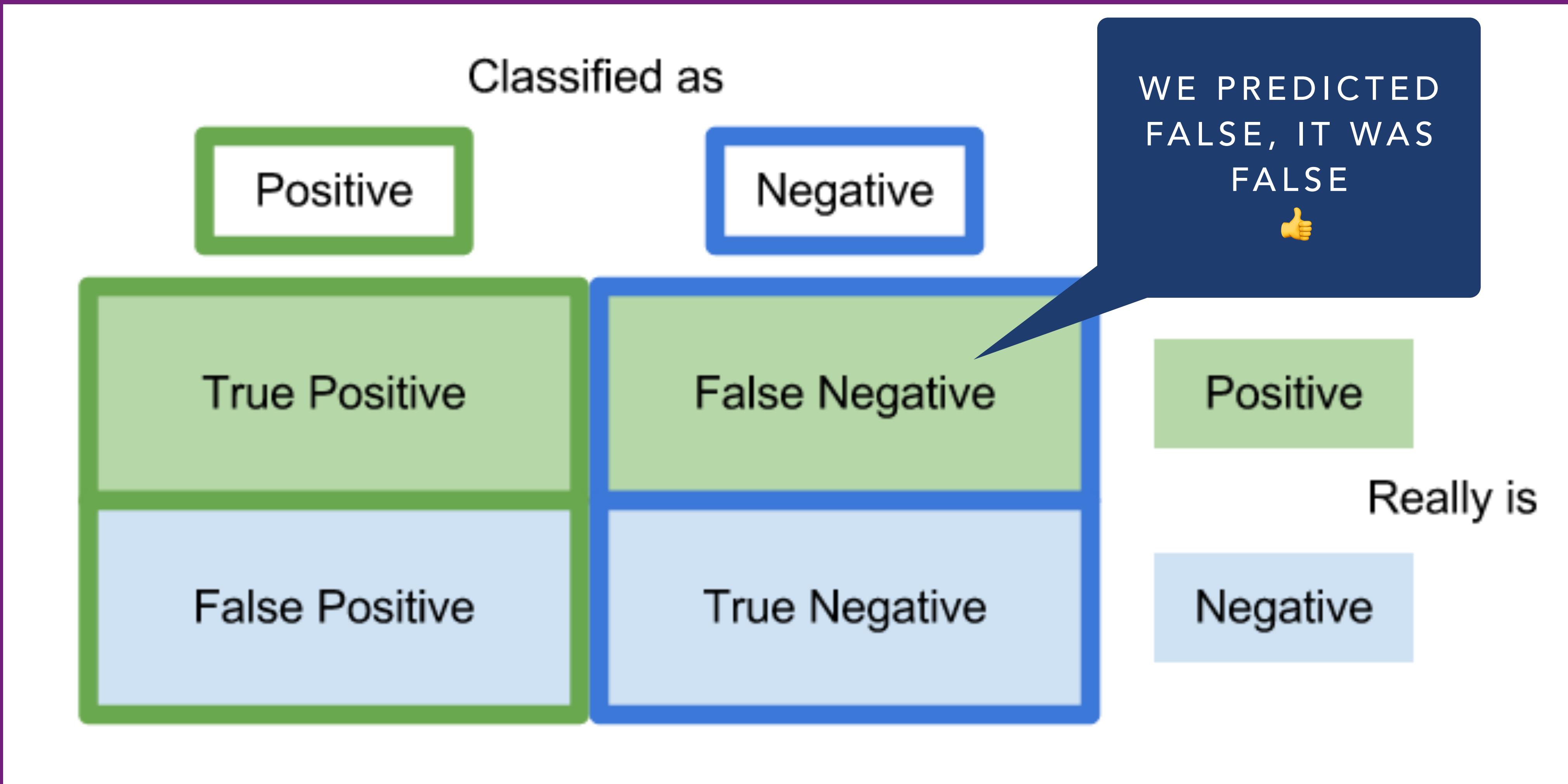
EVALUATING A DATABASE SEARCH



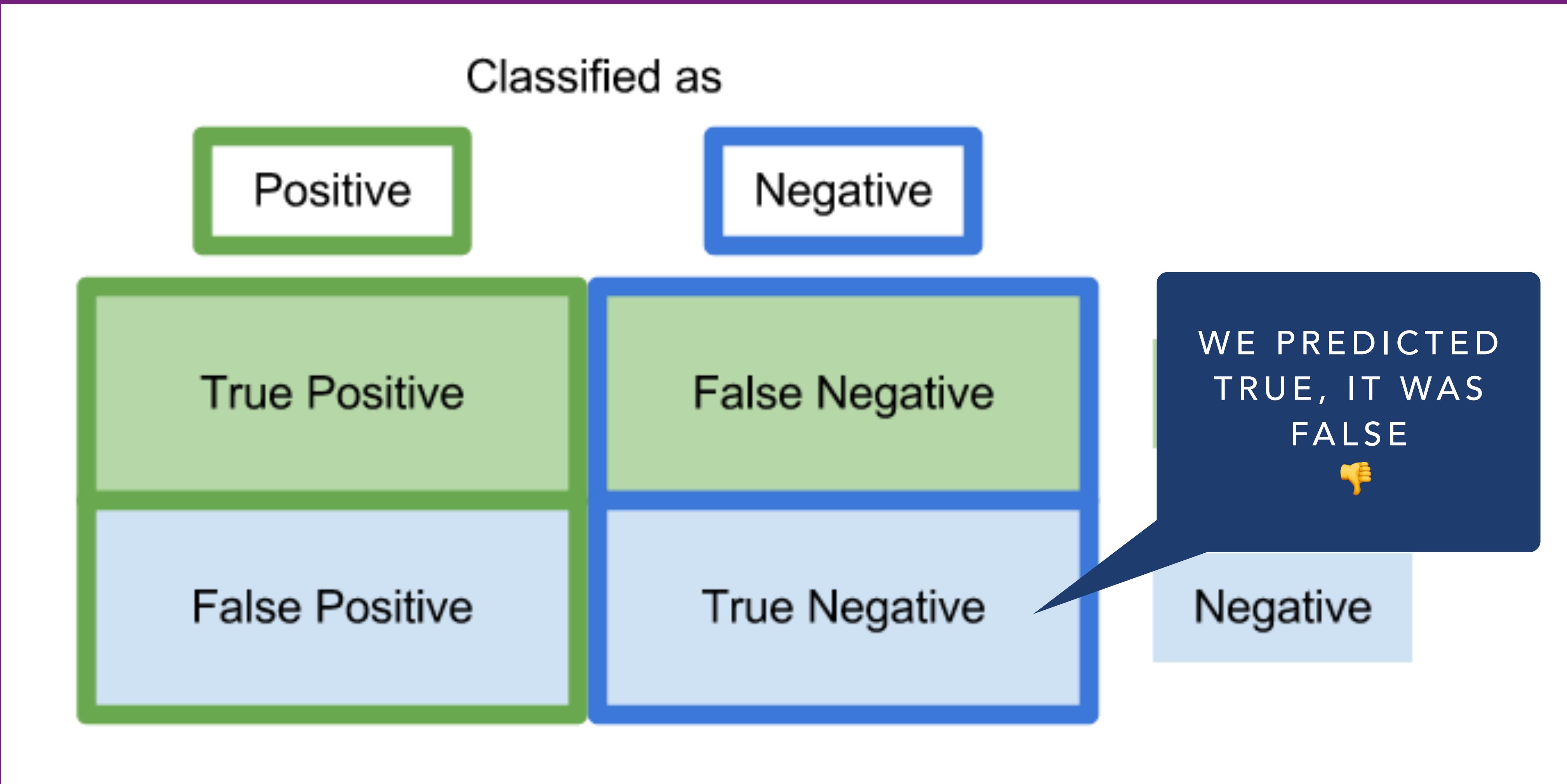
EVALUATING A DATABASE SEARCH



EVALUATING A DATABASE SEARCH



EVALUATING A DATABASE SEARCH



EVALUATING A DATABASE SEARCH

- Example database of food
- Query for “fruit”
- Results are sorted by score
 - > 90 predicted fruit
 - < 90 predicted not a fruit

> apple		100
> orange		99
> banana		93
> pumpkin		93
> grapes		92
> eggplant		91
<hr/>		
> kiwi		88
> lettuce		85
> tomato		79
> onion		77

THRESHOLD IS HEURISTIC

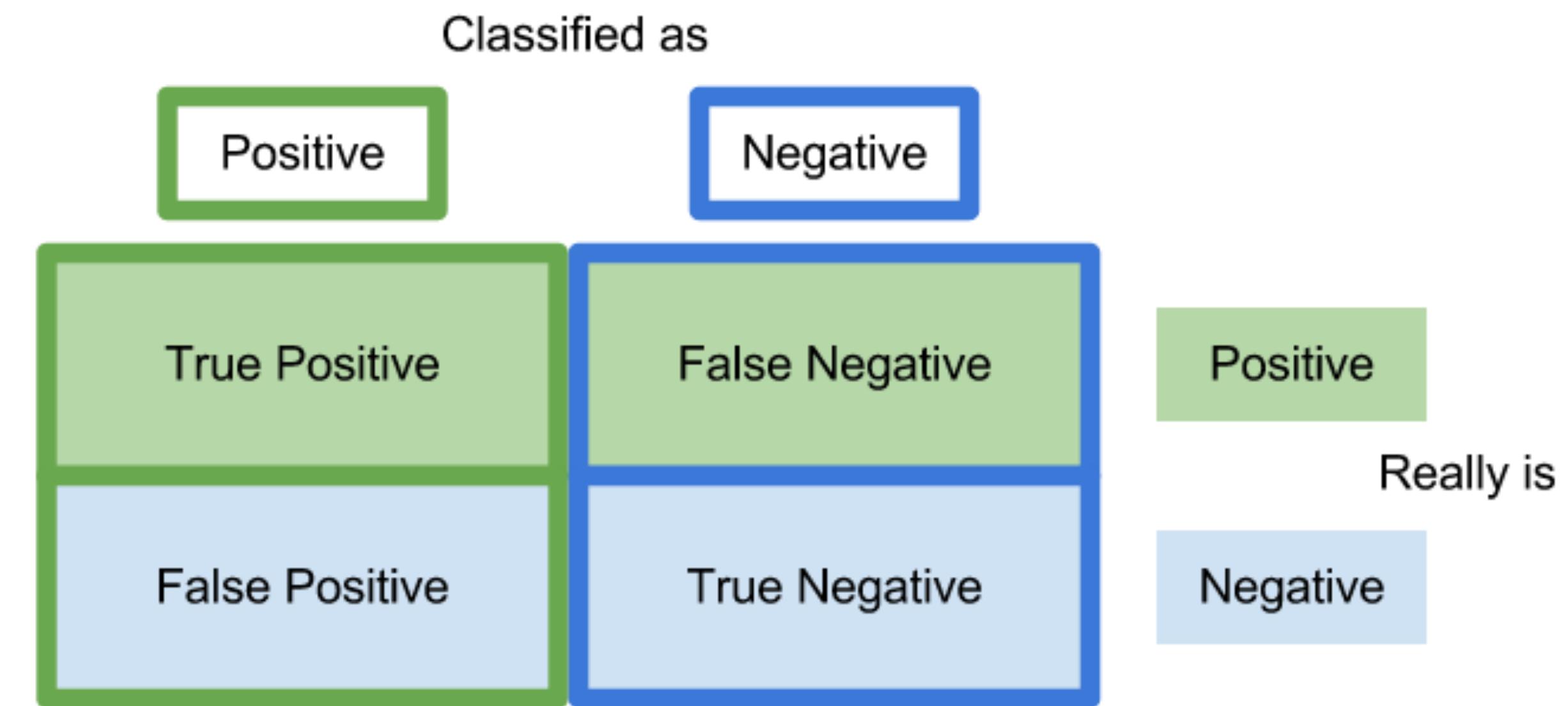
EVALUATING A DATABASE SEARCH

- True Positive - Predicted correctly
- False Positive - Predicted wrong
- True Negative - Correctly predicted not a member
- False Negative - Missed it

> apple	100	FP	TP
> orange	99	FN	
> banana	93	TN	
> pumpkin	93	FN	
> grapes	92	TN	
> eggplant	91		
> kiwi	88		
> lettuce	85		
> tomato	79		
> onion	77		

EVALUATING A DATABASE SEARCH

- Sensitivity
 - Ability to correctly classify as homologous
 - Sensitivity = $TP/TP+FN$
- Specificity
 - Ability to correctly classify as non-homologous
 - Specificity = $TN/(TN+FP)$



EVALUATING A DATABASE SEARCH

- Sensitivity = $TP/TP+FN$
 - $4/6 = 66\%$
- Specificity = $TN/(TN+FP)$
 - $2/4 = 50\%$

TRUE

ACCURATE
PREDICT
NOT TRUE

		TP	FP
> apple	100	FP	
> orange	99	FN	
> banana	93	TN	
> pumpkin	93	FN	
> grapes	92	TN	
> eggplant	91		
> kiwi	88		
> lettuce	85		
> tomato	79		
> onion	77		

DATABASE SEARCHING

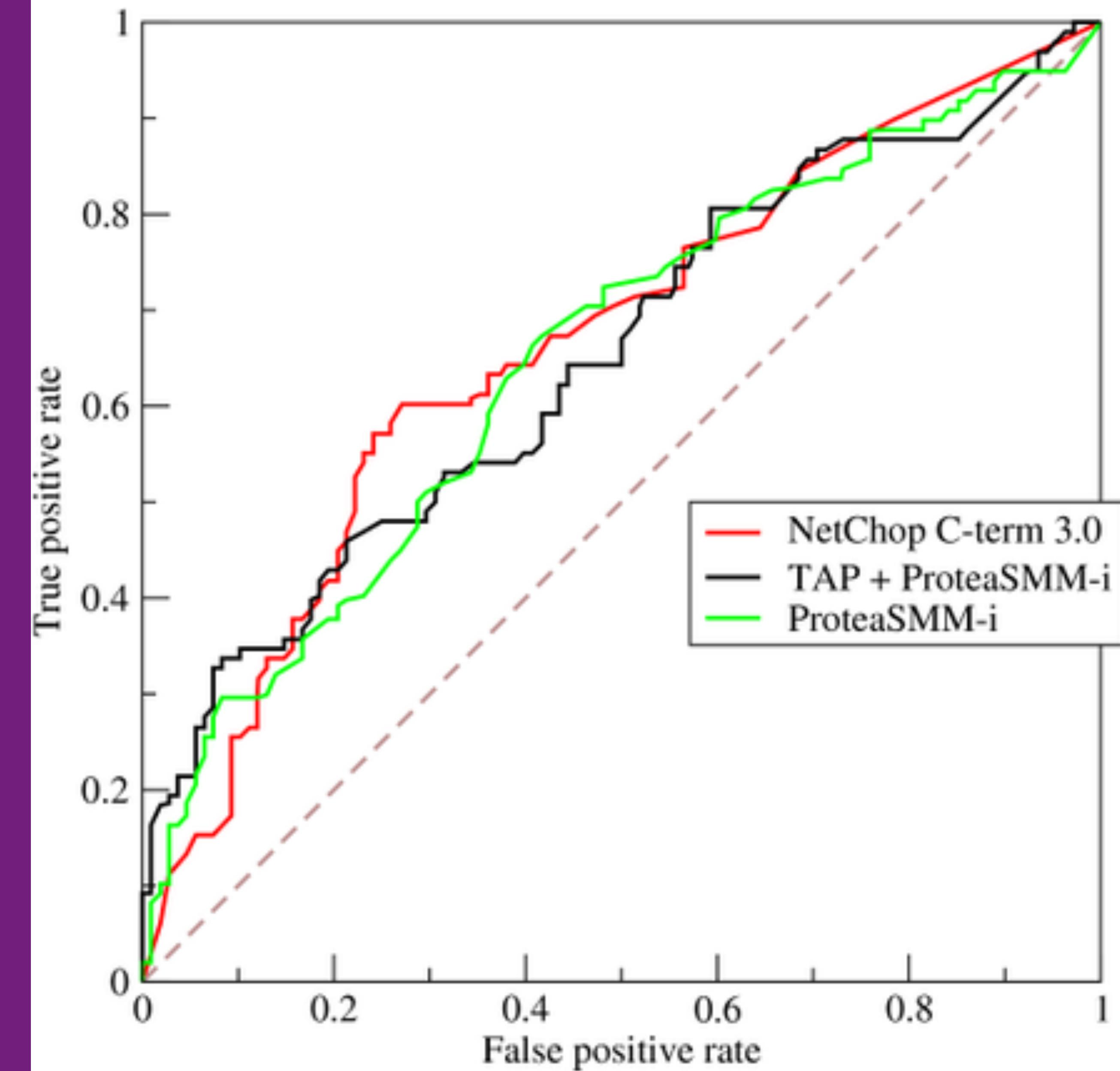
- BLAST results for query of “aspartokinase
 - Determine “true” by name
 - As precise as you want to be

HEURISTIC

Sequences producing significant alignments:				Score (Bits)	E value
sp P00561.2 AK1H_ECOLI	RecName: Full=Bifunctional aspartokina...			1875	0.0
sp P27725.1 AK1H_SERMA	RecName: Full=Bifunctional aspartokina...			1539	0.0
sp P44505.1 AKH_HAEIN	RecName: Full=Bifunctional aspartokinas...			1108	0.0
sp P57290.1 AKH_BUCAI	RecName: Full=Bifunctional aspartokinases...			1022	0.0
sp Q8K9U9.1 AKH_BUCAP	RecName: Full=Bifunctional aspartokinases...			1010	0.0
sp Q89AR4.1 AKH_BUCBP	RecName: Full=Bifunctional aspartokinases...			997	0.0
sp P49079.1 AKH1_MAIZE	RecName: Full=Bifunctional aspartokina...			610	0.0
sp O81852.1 AKH2_ARATH	RecName: Full=Bifunctional aspartokina...			608	0.0
sp Q9SA18.1 AKH1_ARATH	RecName: Full=Bifunctional aspartokina...			590	6e-180
sp P49080.1 AKH2_MAIZE	RecName: Full=Bifunctional aspartokina...			588	2e-179
sp P37142.1 AKH_DAUCA	RecName: Full=Bifunctional aspartokinases...			587	1e-178
sp P00562.3 AK2H_ECOLI	RecName: Full=Bifunctional aspartokina...			366	2e-103
sp Q57991.1 AK_METJA	RecName: Full=Probable aspartokinase; AltName:...			302	7e-85
sp Q5B998.1 DHOM_EMENI	RecName: Full=Homoserine dehydrogenase...			234	2e-63
sp O94671.1 DHOM_SCHPO	RecName: Full=Probable homoserine dehy...			219	2e-58
sp Q9S702.1 AK3_ARATH	RecName: Full=Aspartokinase 3, chloropl...			208	1e-53
sp Q9LYU8.1 AK1_ARATH	RecName: Full=Aspartokinase 1, chloropl...			201	2e-51
sp O23653.2 AK2_ARATH	RecName: Full=Aspartokinase 2, chloropl...			200	5e-51
sp P31116.1 DHOM_YEAST	RecName: Full=Homoserine dehydrogenase...			177	9e-45
sp Q9ZJZ7.1 AK_HELPJ	RecName: Full=Aspartokinase; AltName: Fu...			173	2e-43
sp O25827.1 AK_HELPY	RecName: Full=Aspartokinase; AltName: Fu...			173	3e-43
sp P10869.2 AK_YEAST	RecName: Full=Aspartokinase; AltName: Fu...			166	2e-40
sp P08660.2 AK3_ECOLI	RecName: Full=Lysine-sensitive aspartok...			164	4e-40
sp A4VJB4.1 AKLYS_PSEU5	RecName: Full=Aspartate kinase Ask_Ly...			159	8e-39
sp C3JXY0.1 AK_PSEFS	RecName: Full=Aspartate kinase; AltName:...			158	1e-38
sp P94417.1 AK3_BACSU	RecName: Full=Aspartokinase 3; AltName:...			157	4e-38
sp Q88EI9.1 AK_PSEPK	RecName: Full=Aspartate kinase; AltName:...			154	3e-37
sp O69077.2 AK_PSEAE	RecName: Full=Aspartokinase; AltName: Fu...			153	6e-37
sp P08495.2 AK2_BACSU	RecName: Full=Aspartokinase 2; AltName:...			152	1e-36
sp O67221.1 AK_AQUAE	RecName: Full=Aspartokinase; AltName: Fu...			142	3e-33
sp P02512.2 AK_CORGL	RecName: Full=Aspartokinase; AltName: Fu...			139	2e-32
sp P41993.1 AK_CORFL	RecName: Full=Aspartokinase; AltName: Fu...			137	9e-32
sp O8R0N1.1 AK_COREE	RecName: Full=Aspartokinase; AltName: Fu...			135	3e-31
sp Q59229.1 AK_BACSG	RecName: Full=Aspartokinase; AltName: Fu...			133	2e-30

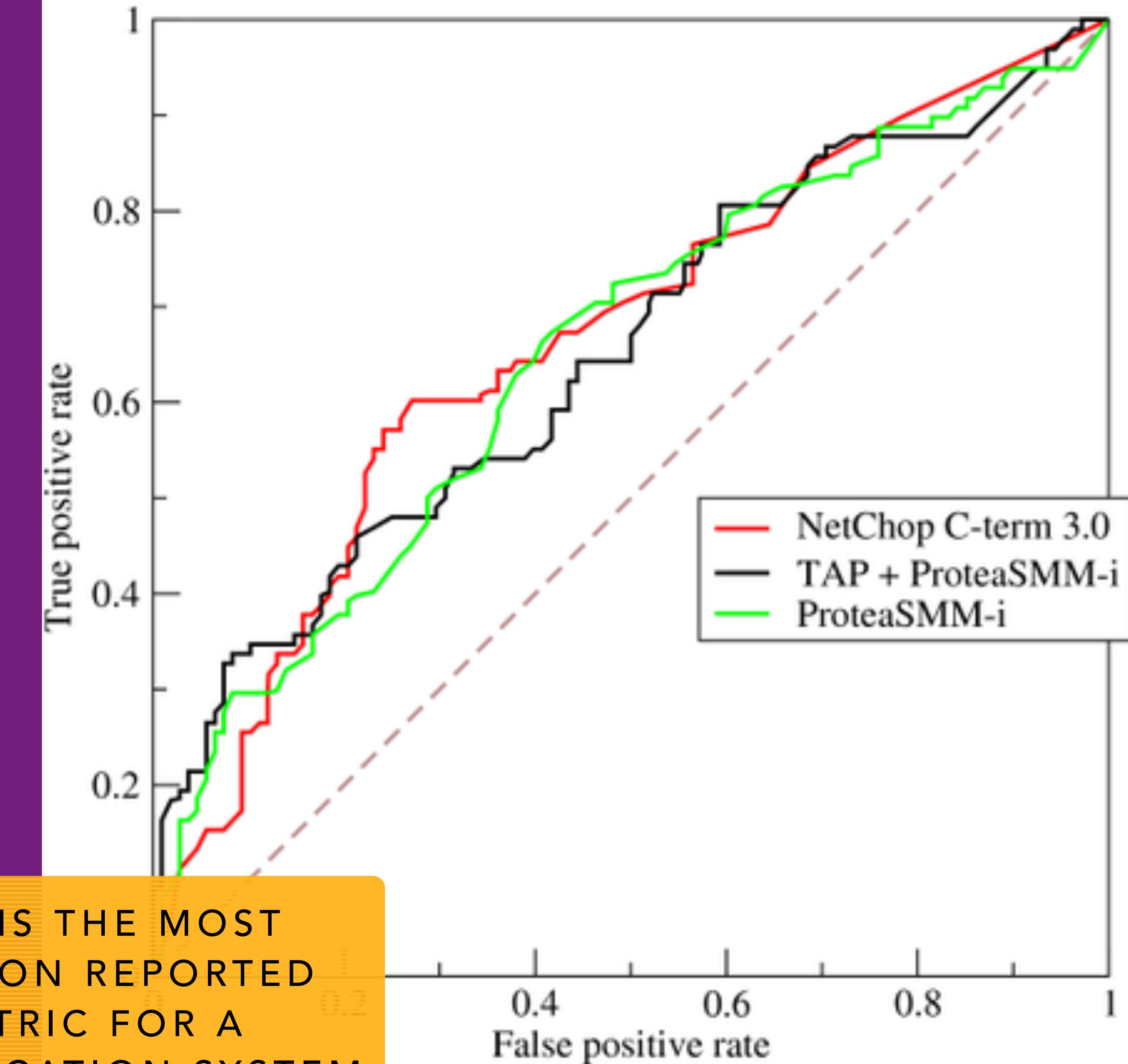
EVALUATING A DATABASE SEARCH

- Receiver operating characteristic (ROC) curve
 - Plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied
 - Sensitivity vs. Specificity
 - TPR/FPR



EVALUATING A DATABASE SEARCH

- Area under curve (AUC)
 - Probability that a randomly chooses positive instance over a chosen negative instance
 - Statistic for a classification method
 - <http://docs.eyesopen.com/toolkits/cookbook/python/plotting/roc.html>



DATABASE SEARCHING

Sequences producing significant alignments

OBSERVED TP/FP IN
CHUNKS OF 5 TO FIND
RATE

		Tp	Fp
	Score 5	5	0
(bits)	Value		
5-10	5	0	
10-15	3	0	2
15-20	1	0	4
20-25	3	0	2
25-30	0	0	
30-35	0	0	
35-40	0	0	
40-45	0	0	
45-50	0	0	
50-55	0	0	
55-60	0	0	
60-65	0	0	
65-70	0	0	
70-75	0	0	
75-80	0	0	
80-85	0	0	
85-90	0	0	
90-95	0	0	
95-100	0	0	
100-105	0	0	
105-110	0	0	
110-115	0	0	
115-120	0	0	
120-125	0	0	
125-130	0	0	
130-135	0	0	
135-140	0	0	
140-145	0	0	
145-150	0	0	
150-155	0	0	
155-160	0	0	
160-165	0	0	
165-170	0	0	
170-175	0	0	
175-180	0	0	
180-185	0	0	
185-190	0	0	
190-195	0	0	
195-200	0	0	
200-205	0	0	
205-210	0	0	
210-215	0	0	
215-220	0	0	
220-225	0	0	
225-230	0	0	
230-235	0	0	
235-240	0	0	
240-245	0	0	
245-250	0	0	
250-255	0	0	
255-260	0	0	
260-265	0	0	
265-270	0	0	
270-275	0	0	
275-280	0	0	
280-285	0	0	
285-290	0	0	
290-295	0	0	
295-300	0	0	
300-305	0	0	
305-310	0	0	
310-315	0	0	
315-320	0	0	
320-325	0	0	
325-330	0	0	
330-335	0	0	
335-340	0	0	
340-345	0	0	
345-350	0	0	
350-355	0	0	
355-360	0	0	
360-365	0	0	
365-370	0	0	
370-375	0	0	
375-380	0	0	
380-385	0	0	
385-390	0	0	
390-395	0	0	
395-400	0	0	
400-405	0	0	
405-410	0	0	
410-415	0	0	
415-420	0	0	
420-425	0	0	
425-430	0	0	
430-435	0	0	
435-440	0	0	
440-445	0	0	
445-450	0	0	
450-455	0	0	
455-460	0	0	
460-465	0	0	
465-470	0	0	
470-475	0	0	
475-480	0	0	
480-485	0	0	
485-490	0	0	
490-495	0	0	
495-500	0	0	
500-505	0	0	
505-510	0	0	
510-515	0	0	
515-520	0	0	
520-525	0	0	
525-530	0	0	
530-535	0	0	
535-540	0	0	
540-545	0	0	
545-550	0	0	
550-555	0	0	
555-560	0	0	
560-565	0	0	
565-570	0	0	
570-575	0	0	
575-580	0	0	
580-585	0	0	
585-590	0	0	
590-595	0	0	
595-600	0	0	
600-605	0	0	
605-610	0	0	
610-615	0	0	
615-620	0	0	
620-625	0	0	
625-630	0	0	
630-635	0	0	
635-640	0	0	
640-645	0	0	
645-650	0	0	
650-655	0	0	
655-660	0	0	
660-665	0	0	
665-670	0	0	
670-675	0	0	
675-680	0	0	
680-685	0	0	
685-690	0	0	
690-695	0	0	
695-700	0	0	
700-705	0	0	
705-710	0	0	
710-715	0	0	
715-720	0	0	
720-725	0	0	
725-730	0	0	
730-735	0	0	
735-740	0	0	
740-745	0	0	
745-750	0	0	
750-755	0	0	
755-760	0	0	
760-765	0	0	
765-770	0	0	
770-775	0	0	
775-780	0	0	
780-785	0	0	
785-790	0	0	
790-795	0	0	
795-800	0	0	
800-805	0	0	
805-810	0	0	
810-815	0	0	
815-820	0	0	
820-825	0	0	
825-830	0	0	
830-835	0	0	
835-840	0	0	
840-845	0	0	
845-850	0	0	
850-855	0	0	
855-860	0	0	
860-865	0	0	
865-870	0	0	
870-875	0	0	
875-880	0	0	
880-885	0	0	
885-890	0	0	
890-895	0	0	
895-900	0	0	
900-905	0	0	
905-910	0	0	
910-915	0	0	
915-920	0	0	
920-925	0	0	
925-930	0	0	
930-935	0	0	
935-940	0	0	
940-945	0	0	
945-950	0	0	
950-955	0	0	
955-960	0	0	
960-965	0	0	
965-970	0	0	
970-975	0	0	
975-980	0	0	
980-985	0	0	
985-990	0	0	
990-995	0	0	
995-1000	0	0	

DATABASE SEARCHING

OBSERVED TP/FP
IN CHUNKS OF 5

	TP	FP
5	5	0
5-10	5	0
10-15	3	2
15-20	1	4
20-25	3	2
	21	9

TP/TP_{TOT}

	Sensitivity	Specificity
10	0.24	1
20	0.48	1
30	0.67	0.88
40	0.81	0.67
50	1	0

$1-(FP/FP_{TOT})$

	TPR	FPR
10	0.24	0
20	0.48	0
30	0.67	0.22
40	0.81	0.33
50	1	1

DATABASE SEARCHING

sp P37142.1 AKH_DAUCA	RecName: Full=Bifunctional aspartokinase; Allosteric regulator	587	1e-178
sp P00562.3 AK2H_ECOLI	RecName: Full=Bifunctional aspartokinase; Allosteric regulator	366	2e-103
sp Q57991.1 AK_METJA	RecName: Full=Probable aspartokinase; Allosteric regulator	302	7e-85
sp Q5B998.1 DHOM_EMENI	RecName: Full=Homoserine dehydrogenase	234	2e-63
sp O94671.1 DHOM_SCHPO	RecName: Full=Probable homoserine dehydrogenase	219	2e-58
sp Q9S702.1 AK3_ARATH	RecName: Full=Aspartokinase 3, chloroplast	208	1e-53

```
grep "|" gistfile1.txt | awk '{print i++,"---->",$0}' | grep -v aspartoki
```

```
13 ----> sp|Q5B998.1|DHOM_EMENI RecName: Full=Homoserine dehydrogenase... 173 5e-53
14 ----> sp|O94671.1|DHOM_SCHPO RecName: Full=Probable homoserine dehy... 164 3e-49
18 ----> sp|P31116.1|DHOM_YEAST RecName: Full=Homoserine dehydrogenase... 138 4e-39
```

DATABASE SEARCHING

```
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline

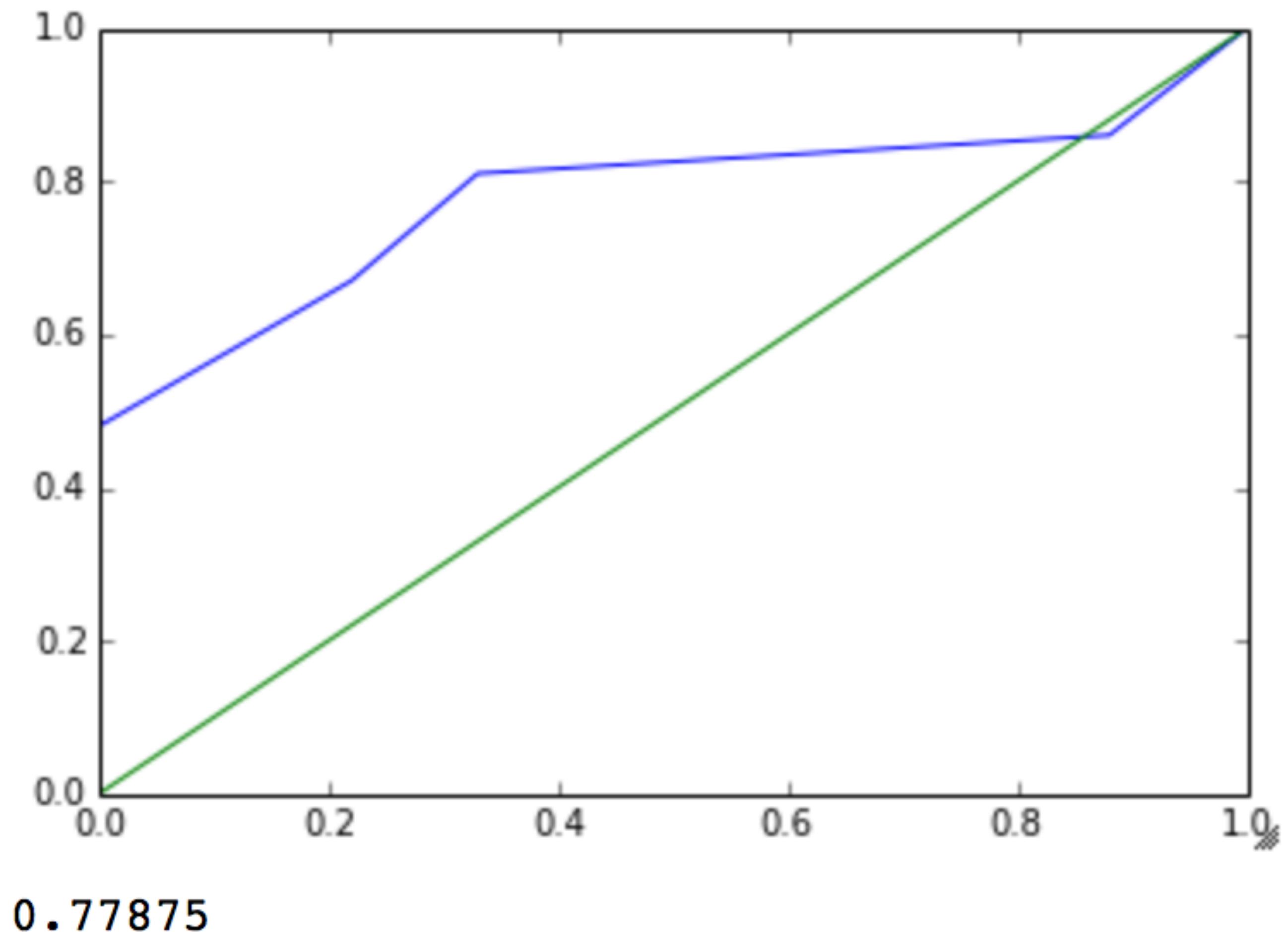
fpr = [0,0,.22,.33,.88,1]
tpr = [.24,.48,.67,.81,.86,1]

# This is the ROC curve
plt.plot(fpr,tpr)

# Plot the random line
plt.plot([0,1],[0,1])

plt.show()

# Plot the area under the curve (AUC)
auc = np.trapz(tpr,fpr)
print auc
```



BIOINFORMATICS

(FOR COMPUTER SCIENTISTS)

MPCS56420

AUTUMN 2018
SESSION 4



THE UNIVERSITY OF
CHICAGO



FASTA

FASTA

- Lipman and Pearson, 1985
- Derived from logic of the dot plot
 - Compute the best diagonals from all frames of alignment
- Compares query sequence pairwise with each sequence in the database

[Science](#). 1985 Mar 22;227(4693):1435-41.

Rapid and sensitive protein similarity searches.

[Lipman DJ](#), [Pearson WR](#).

Abstract

An algorithm was developed which facilitates the search for similarities between already available in databases. Because of the algorithm's efficiency on many now become a routine procedure for molecular biologists. The method efficiently aligned identical and differing residues in those regions by means of an amino acid giving high scores to those amino acid replacements which occur frequently in program designed to search protein databases very rapidly. For example, comparing the sequence of bovine insulin with all the sequences in the protein library in the National Biomedical Research Foundation library would take less than 2 minutes on a microcomputer (IBM PC).

PMID: 2983426 [PubMed - indexed for MEDLINE]



Publication Types, MeSH Terms, Substances, Grant Support

LinkOut - more resources

PubMed Commons

 0 comments

FASTA

Science. 1985 Mar 22;227(4693):1435-41.

Rapid and sensitive protein similarity searches.

Lipman DJ, Pearson WR.

Abstract

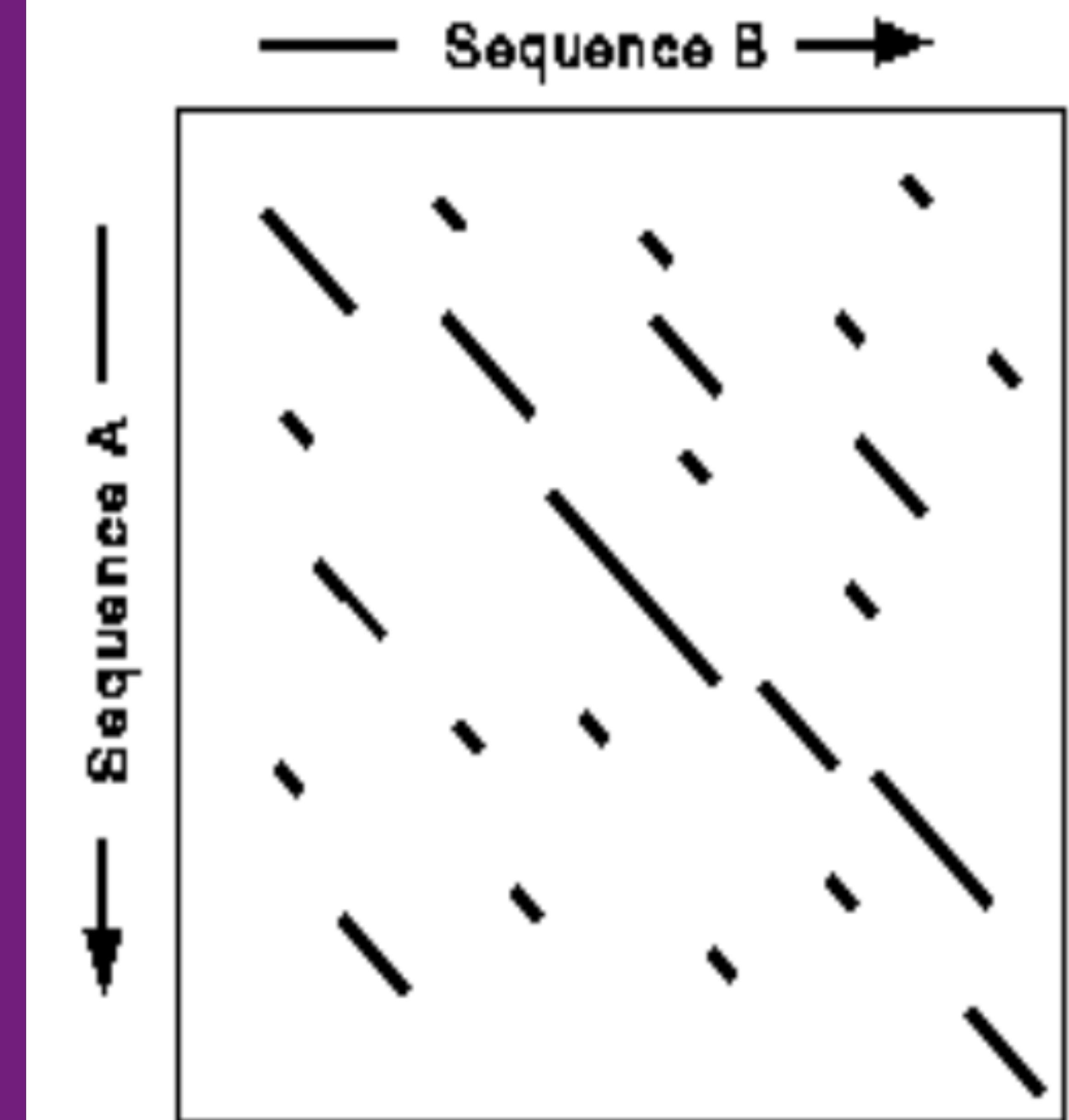
An algorithm was developed which facilitates the search for similarities already available in databases. Because of the algorithm's efficiency, it has now become a routine procedure for molecular biologists. The method aligns identical and differing residues in those regions by means of a scoring system giving high scores to those amino acid replacements which occur frequently. A program designed to search protein databases very rapidly. For example, a search of the National Biomedical Research Foundation library would take only a few minutes on a microcomputer (IBM PC).

PMID: 2983426 [PubMed - indexed for MEDLINE]



FASTA

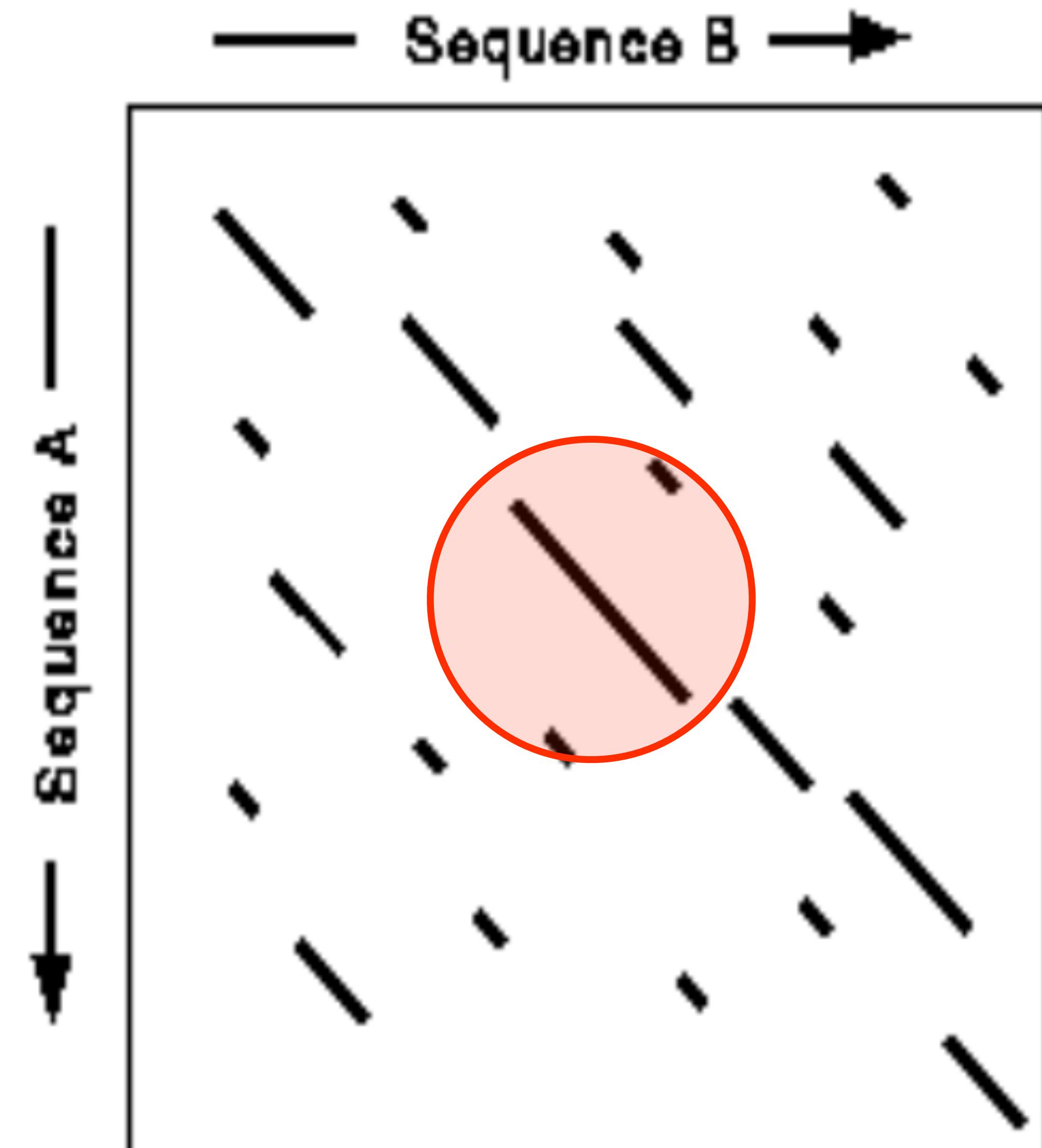
- Word method
 - Identify **exact** matches between words(k) in query and db sequence
 - DNA ~6 bases
 - Protein ~2 amino acids
- Only search in areas that contain word matches
 - Hash table lookup



Find runs of identical words

FASTA METHOD

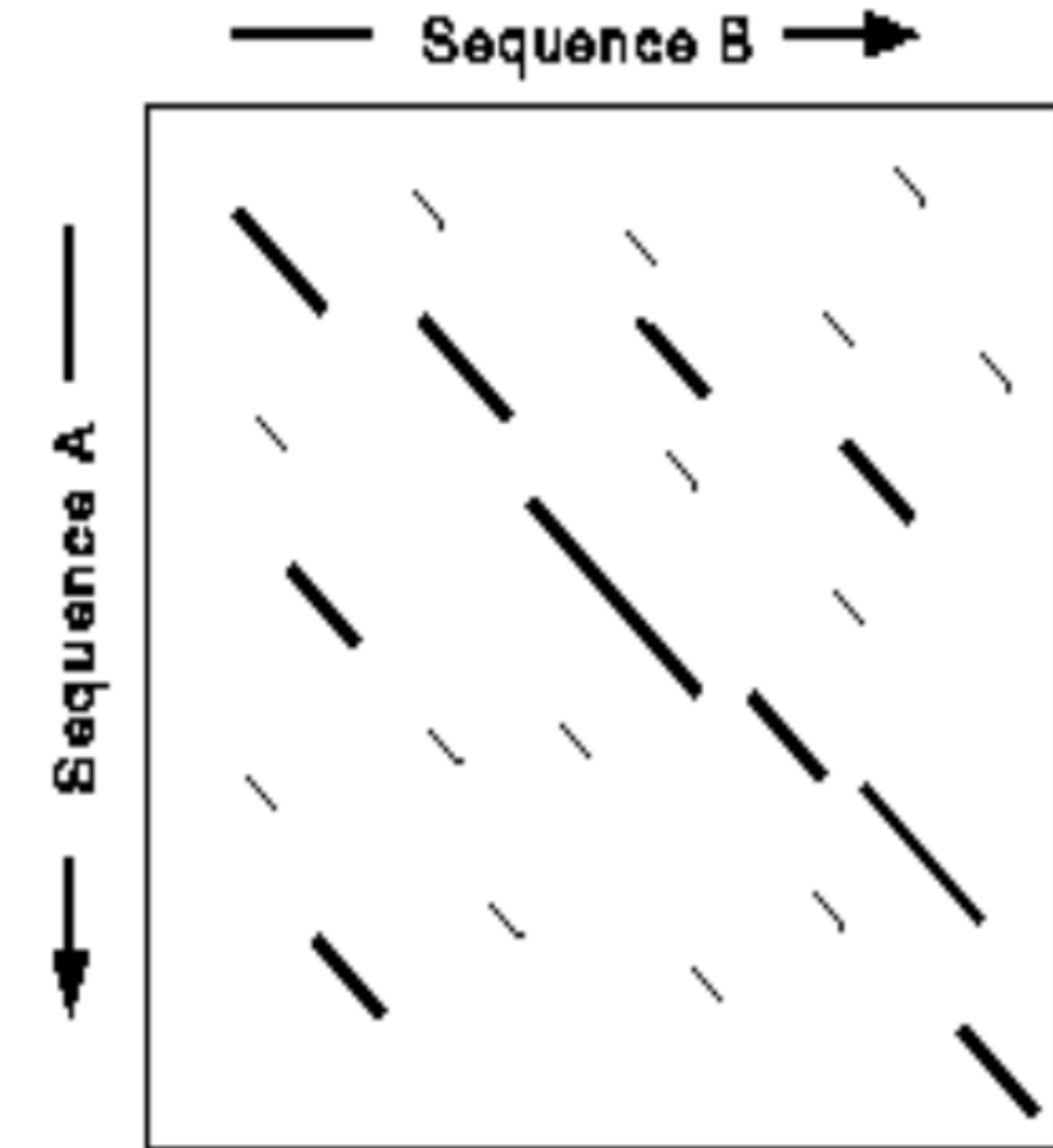
- Step 1: Find diagonals (paired pieces from each sequence without gaps) that have the highest density of common words



Find runs of identical words

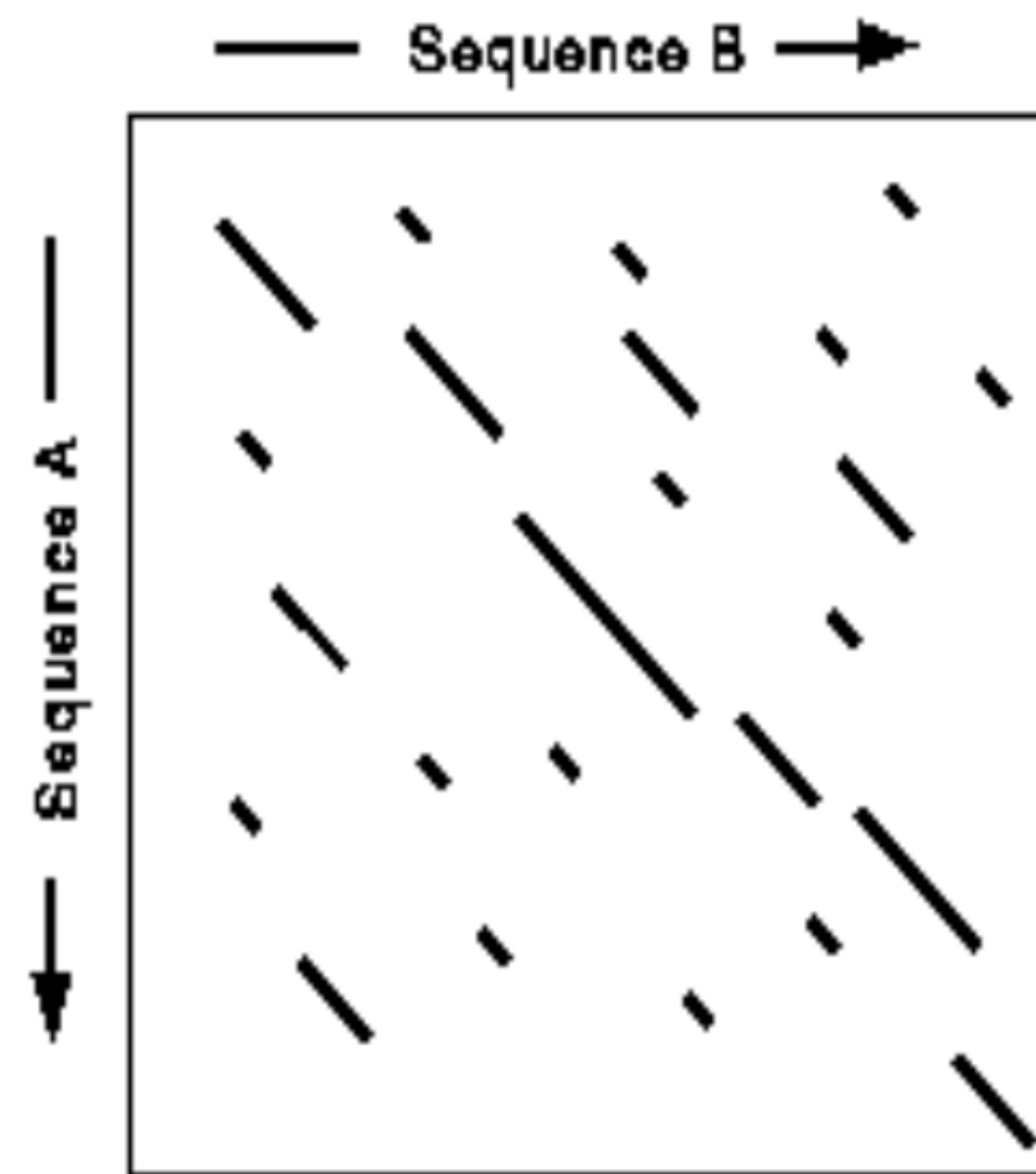
FASTA METHOD

- Step 2: Rescore these using a scoring (similarity) matrix and trim ends that do not contribute to the highest score
 - Result: partial alignments without gaps
 - Reported as the “init1” score

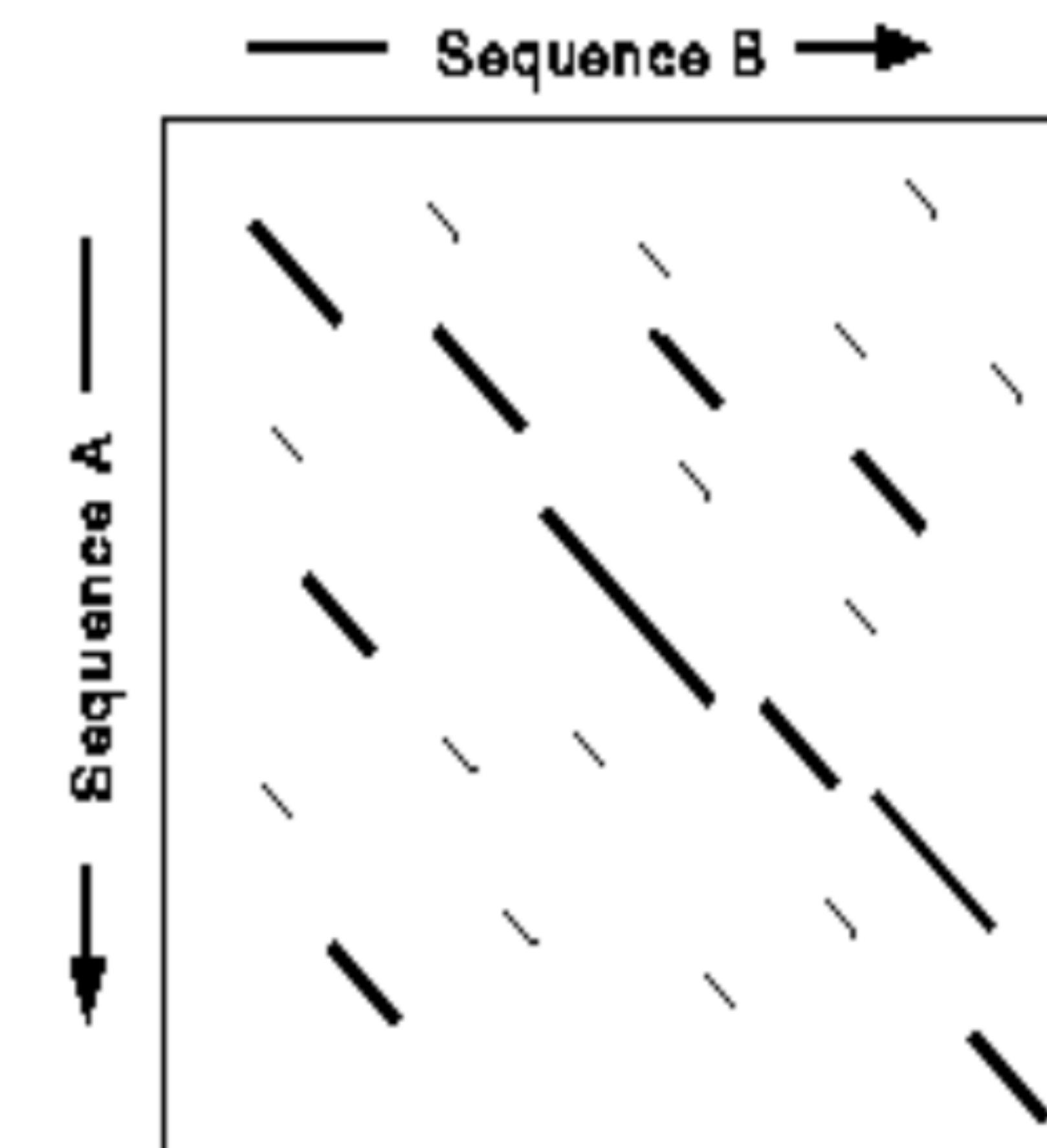


Re-score using PAM matrix
Keep top scoring segments

FASTA METHOD



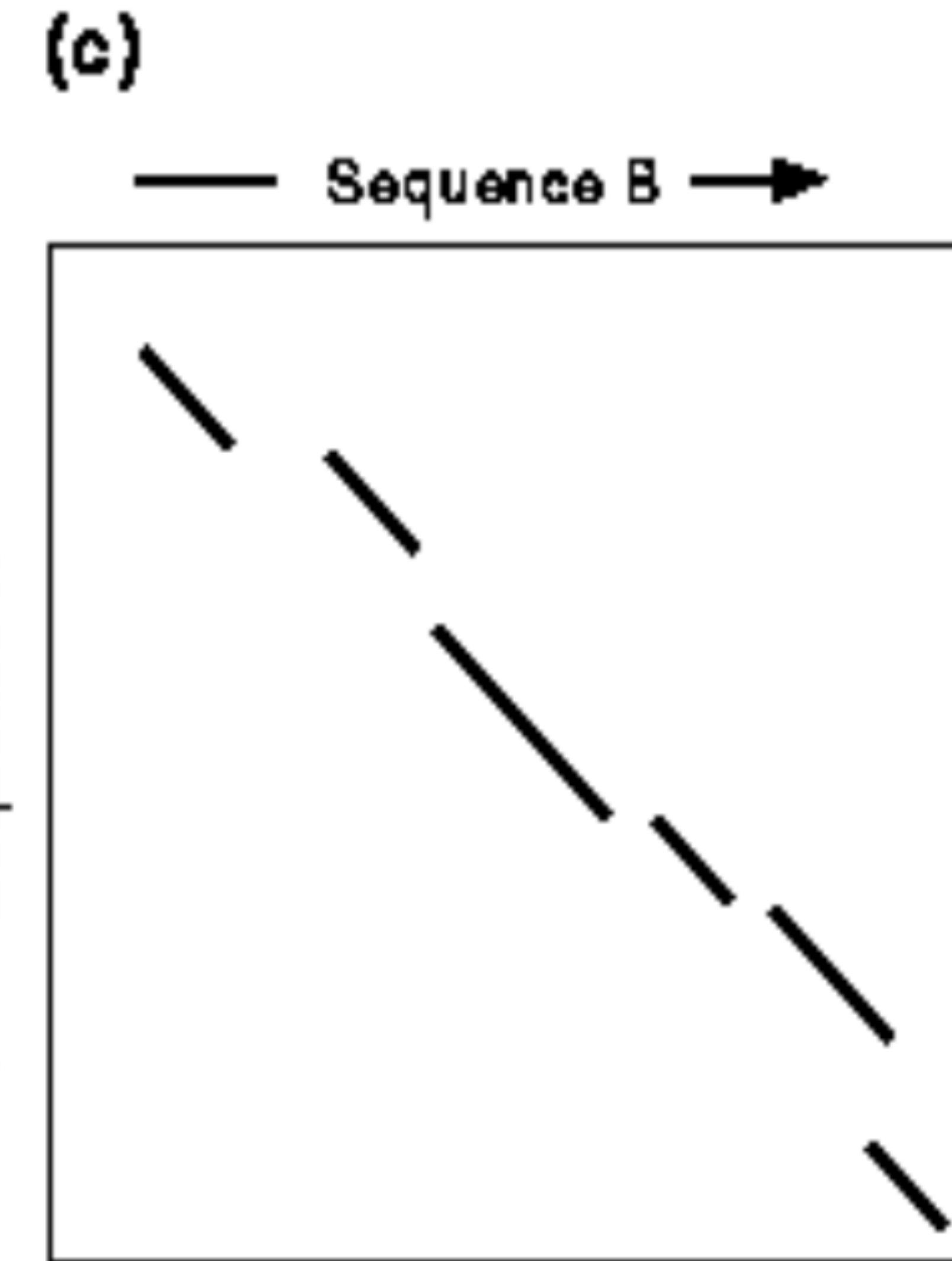
Find runs of identical words



Re-score using PAM matrix
Keep top scoring segments

FASTA METHOD

- Step 3. Join regions together, including penalties for gaps
- Result:
 - Unoptimized alignment with gaps
 - Reported as the “initn” score

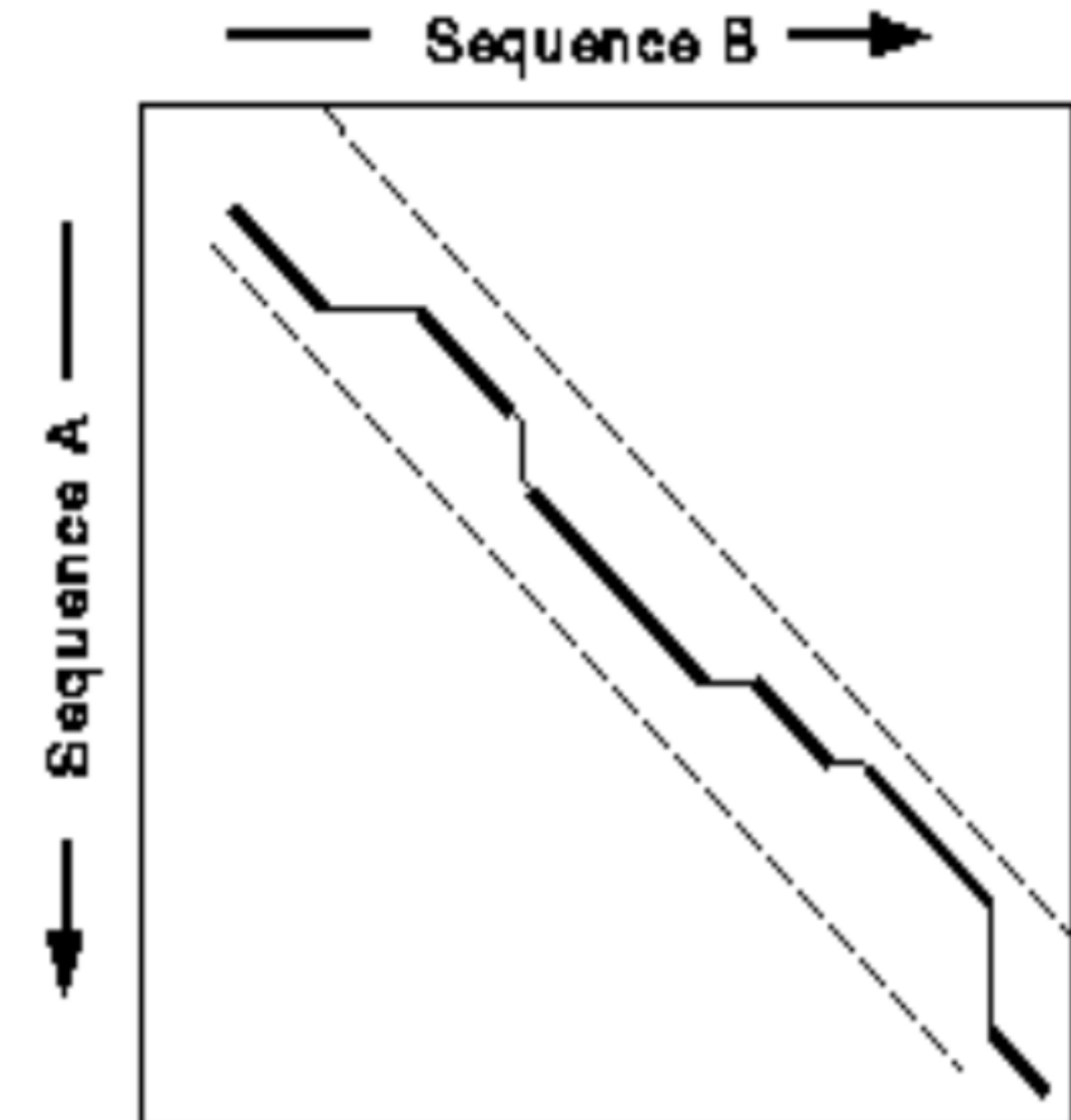


Join segments using gaps,
eliminate other segments

FASTA METHOD

- Step 4. Use dynamic programming in a band 32 residues wide around the best “initn” score
- Result:
 - Optimized alignment with gaps
 - Reported as the “opt” score

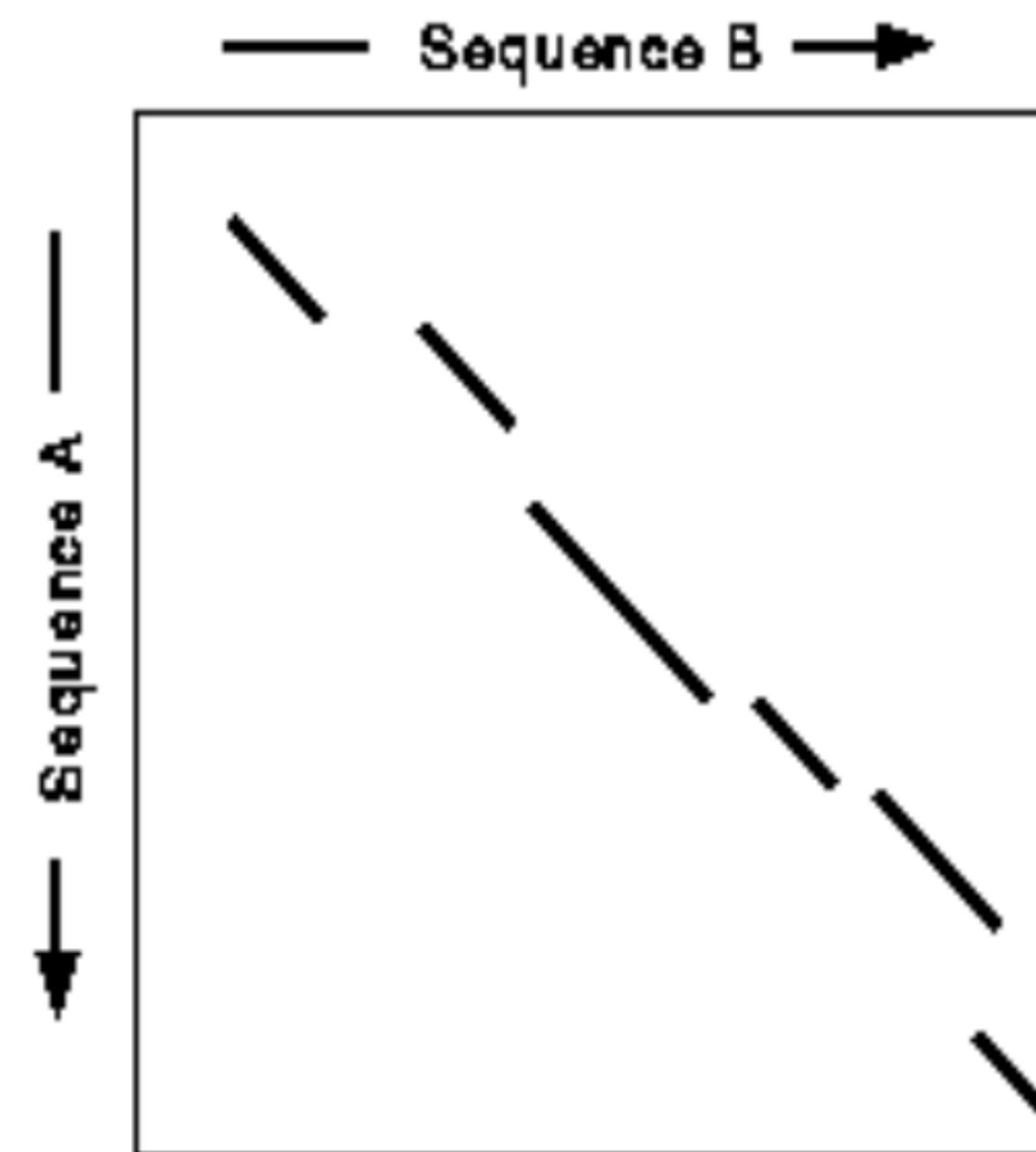
(d)



Use dynamic programming to create an optimal alignment

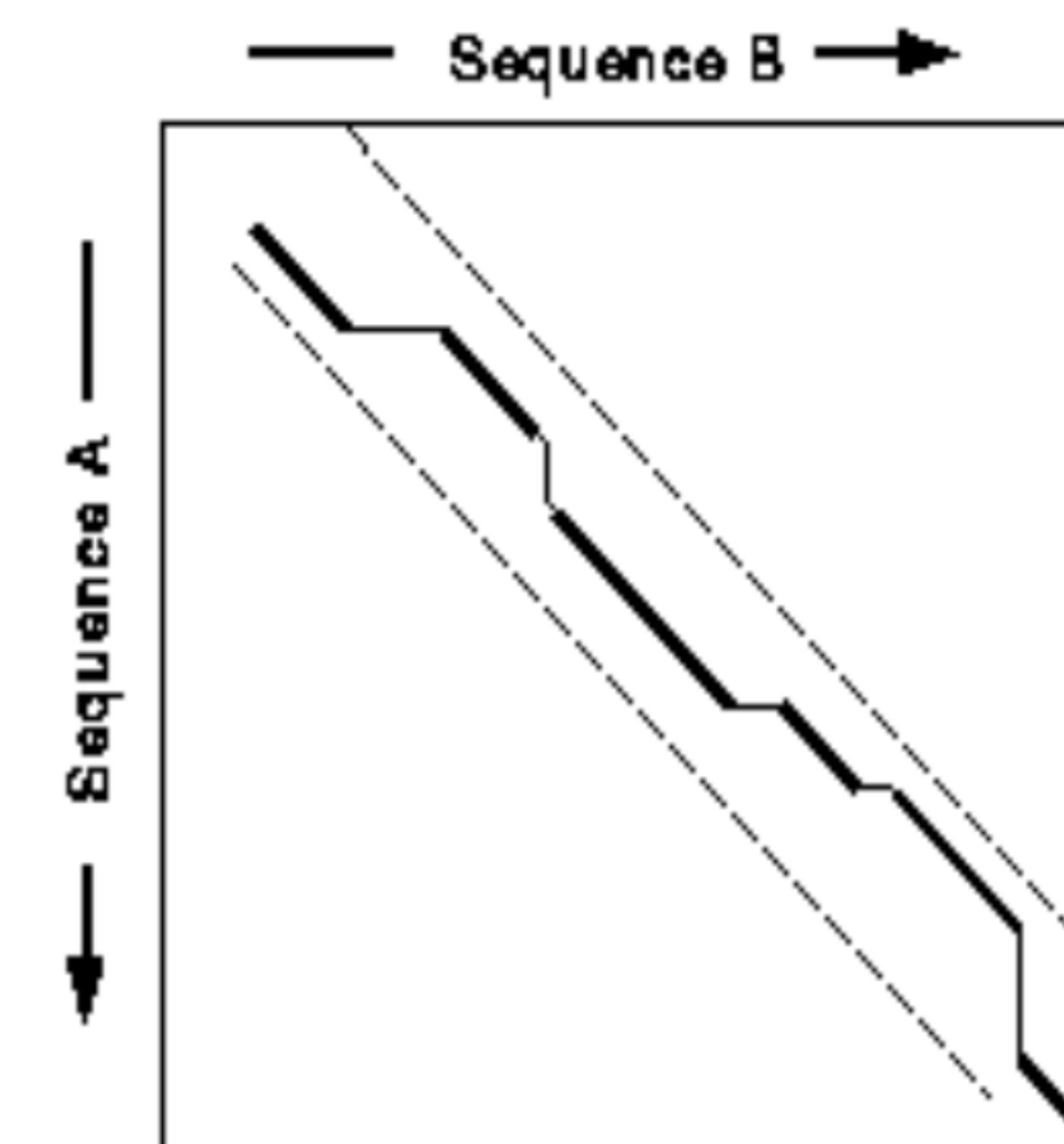
FASTA METHOD

(c)



Join segments using gaps,
eliminate other segments

(d)

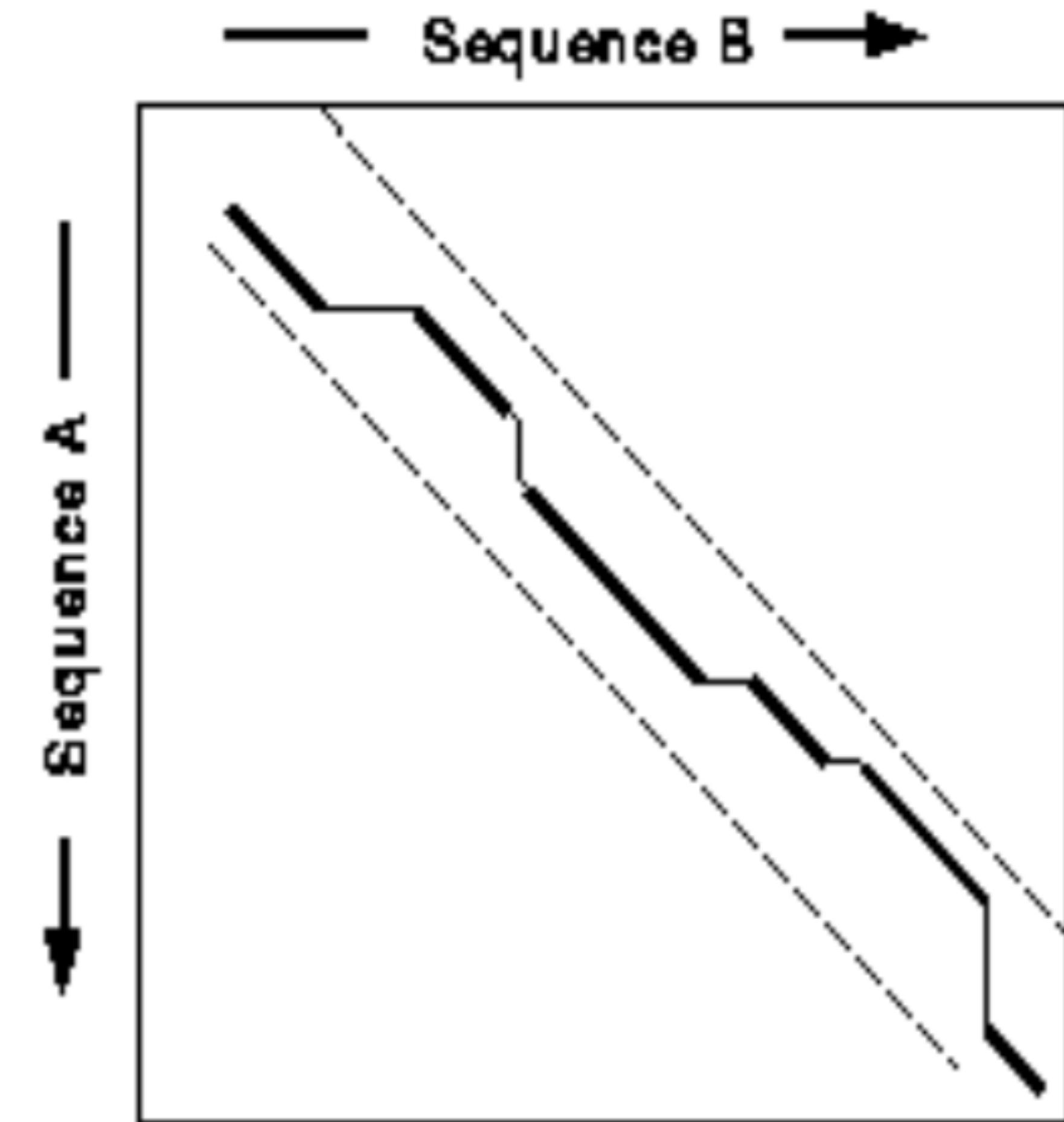


Use dynamic programming to
create an optimal alignment

FASTA LIMITATIONS

- Larger words (k) increases speed since fewer “hits” are found
 - Decreases sensitivity for finding similar (but not identical) sequences since exact matches of this length are required
 - Words have to be exact match

(d)



Use dynamic programming to create an optimal alignment

FASTA FAILS

FASTA LIMITATIONS

- FASTA can miss significant similarity since homologous sequences **do not** need to share identical residues
 - Example:
 - Asp-Lys-Val is homologous to Glu-Arg-Ile; missed even with word size of 1 since no amino acid matches
 - Example:
 - Gly-Asp-Gly-Lys-Gly is homologous to Gly-Glu-Gly-Arg-Gly but there is only match with word size of 1 (no consecutive matches)

DLV
ERI

GDGKG
GEGRG

FASTA LIMITATIONS

- Example
 - Codon “wobble”, DNA sequences may look like
XXyXXyXXy
 - X's are conserved and y's are not
 - **GGuUCuACgAAg** and **GGcUCcACaAAA** both code for the same peptide sequence (Gly-Ser-Thr-Lys)
 - Don't match with k size of 3 or higher

MET	LYS	PRO	HIS
ATG	AAA	CCT	CAT
ATG	AAG	CCT	CAT
ATG	AAA	CCC	CAT
ATG	AAG	CCC	CAT
ATG	AAA	CCA	CAT
ATG	AAG	CCA	CAT
ATG	AAA	CCG	CAT
ATG	AAG	CCG	CAT
ATG	AAA	CCT	CAC
ATG	AAG	CCT	CAC
ATG	AAA	CCC	CAC
ATG	AAG	CCC	CAC
ATG	AAA	CCA	CAC
ATG	AAG	CCA	CAC
ATG	AAA	CCG	CAC
ATG	AAG	CCG	CAC

FASTA SUITE

- Suite of programs
 - FASTX and FASTY translate a nucleotide query for searching a protein database
 - TFASTX and TFASTY translate a nucleotide database to be searched with a protein query
 - SSEARCH (local)
 - GGSEARCH (global)
 - GLSEARCH (global query against local database)
- <http://www.ebi.ac.uk/Tools/ssss/fasta/>

The screenshot shows the EMBL-EBI FASTA Protein Similarity Search interface. At the top, there is a navigation bar with tabs for Protein, Nucleotide, Genomes, Proteomes, and Whole Genomes. Below the navigation bar, there is a "Help & Documentation" link and a breadcrumb navigation path: Tools > Sequence Similarity Searching > FASTA. The main section is titled "Protein Similarity Search" and contains a brief description: "This tool provides sequence similarity searching against protein heuristic search with a protein query. FASTX and FASTY translate (local), GGSEARCH (global) and GLSEARCH (global query, local". Below this, there are two main steps: "STEP 1 - Select your databases" and "STEP 2 - Enter your input sequence". In "STEP 1", under "PROTEIN DATABASES", the "Databank Selected" section shows "UniProt Knowledgebase" checked. Other options include "UniProtKB/Swiss-Prot", "UniProtKB/Swiss-Prot isoforms", "UniProtKB/TrEMBL", "UniProtKB Taxonomic Subsets", "UniProt Clusters", and "Patents". In "STEP 2", there is a text input field with placeholder text "Enter or paste a PROTEIN sequence in any supported format:" and a file upload section with the message "no file selected".

BIOINFORMATICS

(FOR COMPUTER SCIENTISTS)

MPCS56420

AUTUMN 2018
SESSION 4



THE UNIVERSITY OF
CHICAGO