

# 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®** Basic BLAST

Home Recent Results Saved Strategies Help

► NCBI/ BLAST Home

BLAST finds regions of similarity between biological sequences

**New** DELTA-BLAST, a more sensitive program

### BLAST Assembled Genomes

Find Genomic BLAST pages:

Enter organism name or id--completions will be suggested **GO**

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

### Basic BLAST

Choose a BLAST program to run.

<a href="#">nucleotide blast</a>	Search a <b>nucleotide</b> database using a <b>nucleotide</b> query <i>Algorithms: blastn, megablast, discontinuous megablast</i>
<a href="#">protein blast</a>	Search <b>protein</b> database using a <b>protein</b> query <i>Algorithms: blastp, psi-blast, phi-blast, de</i>
<a href="#">blastx</a>	Search <b>protein</b> database using a <b>translated nucleotide</b> query
<a href="#">tblastn</a>	Search <b>translated nucleotide</b> database using a <b>nucleotide</b> query
<a href="#">tblastx</a>	Search <b>translated nucleotide</b> database using a <b>translated nucleotide</b> query

### Specialized BLAST

# 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
c		-1	1	← 0	← -1	-2	-3	-4	-5
g		-2	0	1	0	0	-1	-2	-3
c		-3	-1	0	2	← 1	← 0	-1	-2
a		-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,

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

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

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

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

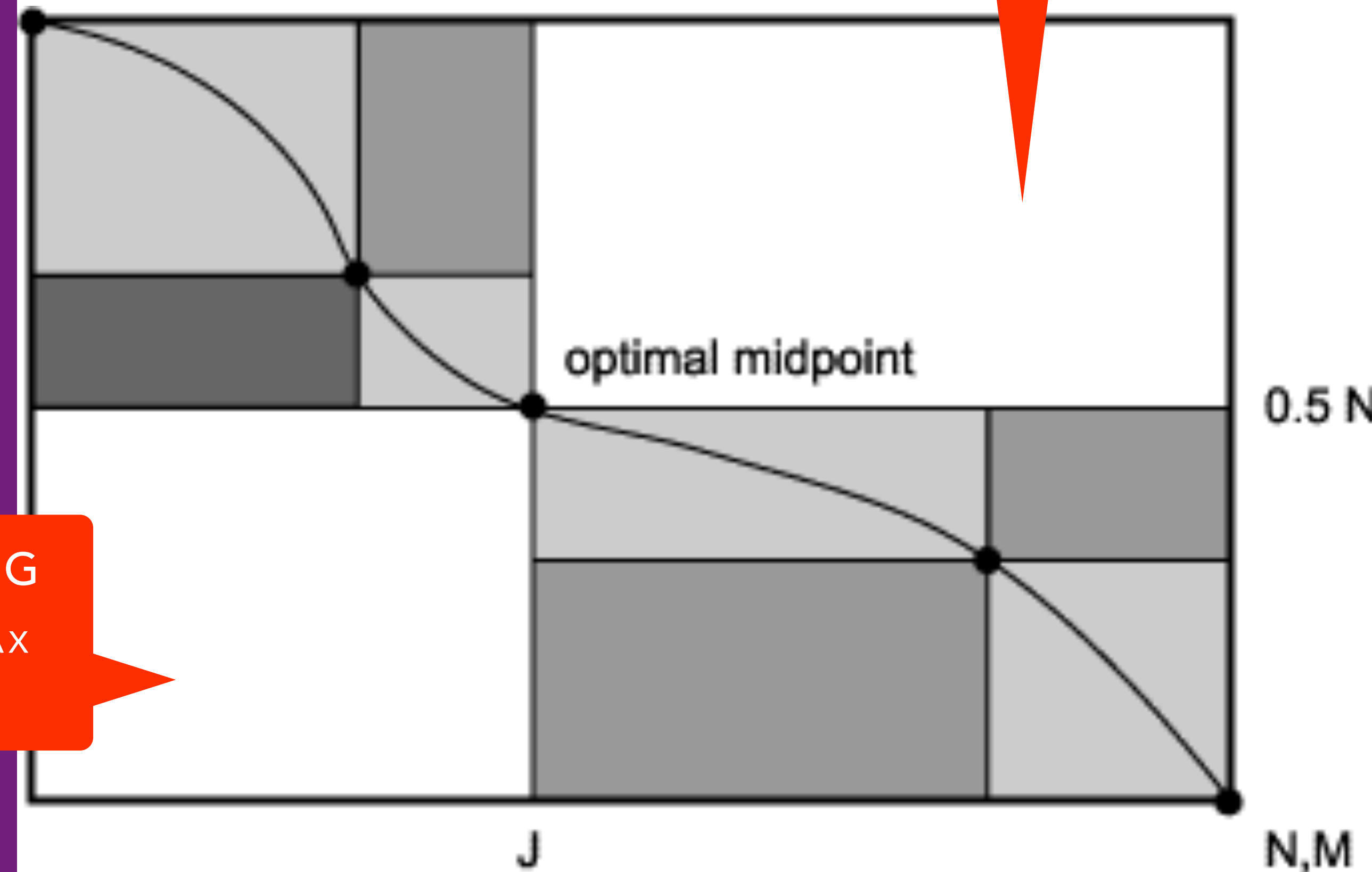
where the penalty of a gap of length  $x$  is given by,

$$w_x = g + rx$$

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 midpoint
  - Join alignments



$$\sim .5 * (M \times N)$$



# 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	AA <b>A</b>	CC <b>T</b>	CAT <b>T</b>
	(2)	ATG	AA <b>G</b>	CC <b>T</b>	CAT <b>T</b>
	(3)	ATG	AA <b>A</b>	CC <b>C</b>	CAT <b>T</b>
	(4)	ATG	AA <b>G</b>	CC <b>C</b>	CAT <b>T</b>
	(5)	ATG	AA <b>A</b>	CC <b>A</b>	CAT <b>T</b>
	(6)	ATG	AA <b>G</b>	CC <b>A</b>	CAT <b>T</b>
	(7)	ATG	AA <b>A</b>	CC <b>G</b>	CAT <b>T</b>
	(8)	ATG	AA <b>G</b>	CC <b>G</b>	CAT <b>T</b>
	(9)	ATG	AA <b>A</b>	CC <b>T</b>	CAC <b>T</b>
	(10)	ATG	AA <b>G</b>	CC <b>T</b>	CAC <b>T</b>
	(11)	ATG	AA <b>A</b>	CC <b>C</b>	CAC <b>T</b>
	(12)	ATG	AA <b>G</b>	CC <b>C</b>	CAC <b>T</b>
	(13)	ATG	AA <b>A</b>	CC <b>A</b>	CAC <b>T</b>
	(14)	ATG	AA <b>G</b>	CC <b>A</b>	CAC <b>T</b>
	(15)	ATG	AA <b>A</b>	CC <b>G</b>	CAC <b>T</b>
	(16)	ATG	AA <b>G</b>	CC <b>G</b>	CAC <b>T</b>

# DNA VS PROTEIN SEARCHES

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

"IT DEPENDS..."

Peptide	(1)	MET	LYS	PRO	HIS
DNA	(1)	ATG	AA <b>A</b>	CC <b>T</b>	CAT <b>T</b>
	(2)	ATG	AA <b>G</b>	CC <b>T</b>	CAT <b>T</b>
	(3)	ATG	AA <b>A</b>	CC <b>C</b>	CAT <b>T</b>
	(4)	ATG	AA <b>G</b>	CC <b>C</b>	CAT <b>T</b>
	(5)	ATG	AA <b>A</b>	CC <b>A</b>	CAT <b>T</b>
	(6)	ATG	AA <b>G</b>	CC <b>A</b>	CAT <b>T</b>
	(7)	ATG	AA <b>A</b>	CC <b>G</b>	CAT <b>T</b>
	(8)	ATG	AA <b>G</b>	CC <b>G</b>	CAT <b>T</b>
	(9)	ATG	AA <b>A</b>	CC <b>T</b>	CAC <b>T</b>
	(10)	ATG	AA <b>G</b>	CC <b>T</b>	CAC <b>T</b>
	(11)	ATG	AA <b>A</b>	CC <b>C</b>	CAC <b>T</b>
	(12)	ATG	AA <b>G</b>	CC <b>C</b>	CAC <b>T</b>
	(13)	ATG	AA <b>A</b>	CC <b>A</b>	CAC <b>T</b>
	(14)	ATG	AA <b>G</b>	CC <b>A</b>	CAC <b>T</b>
	(15)	ATG	AA <b>A</b>	CC <b>G</b>	CAC <b>T</b>
	(16)	ATG	AA <b>G</b>	CC <b>G</b>	CAC <b>T</b>

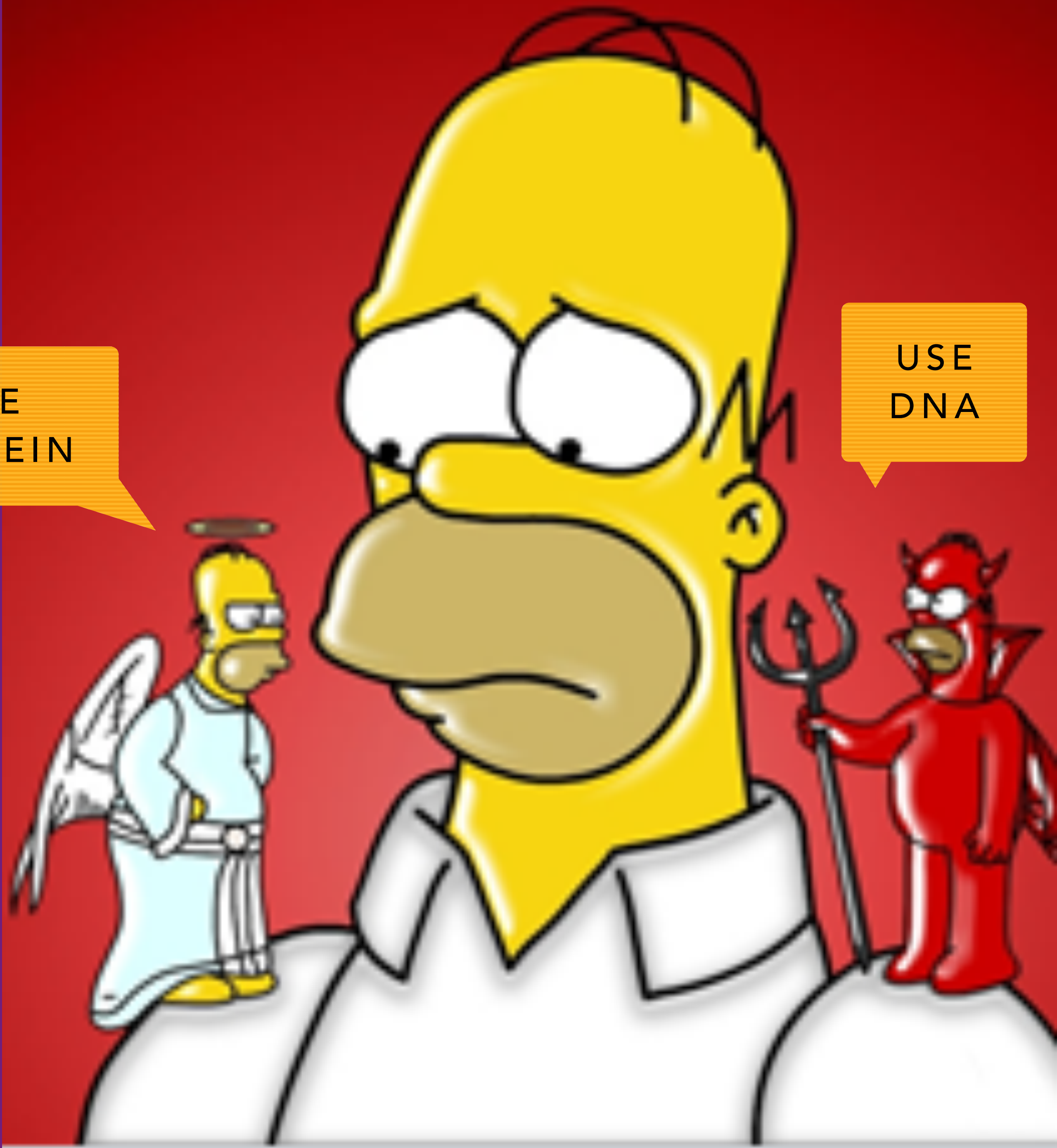


# DNA VS PROTEIN SEARCHES

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

USE  
PROTEIN

USE  
DNA



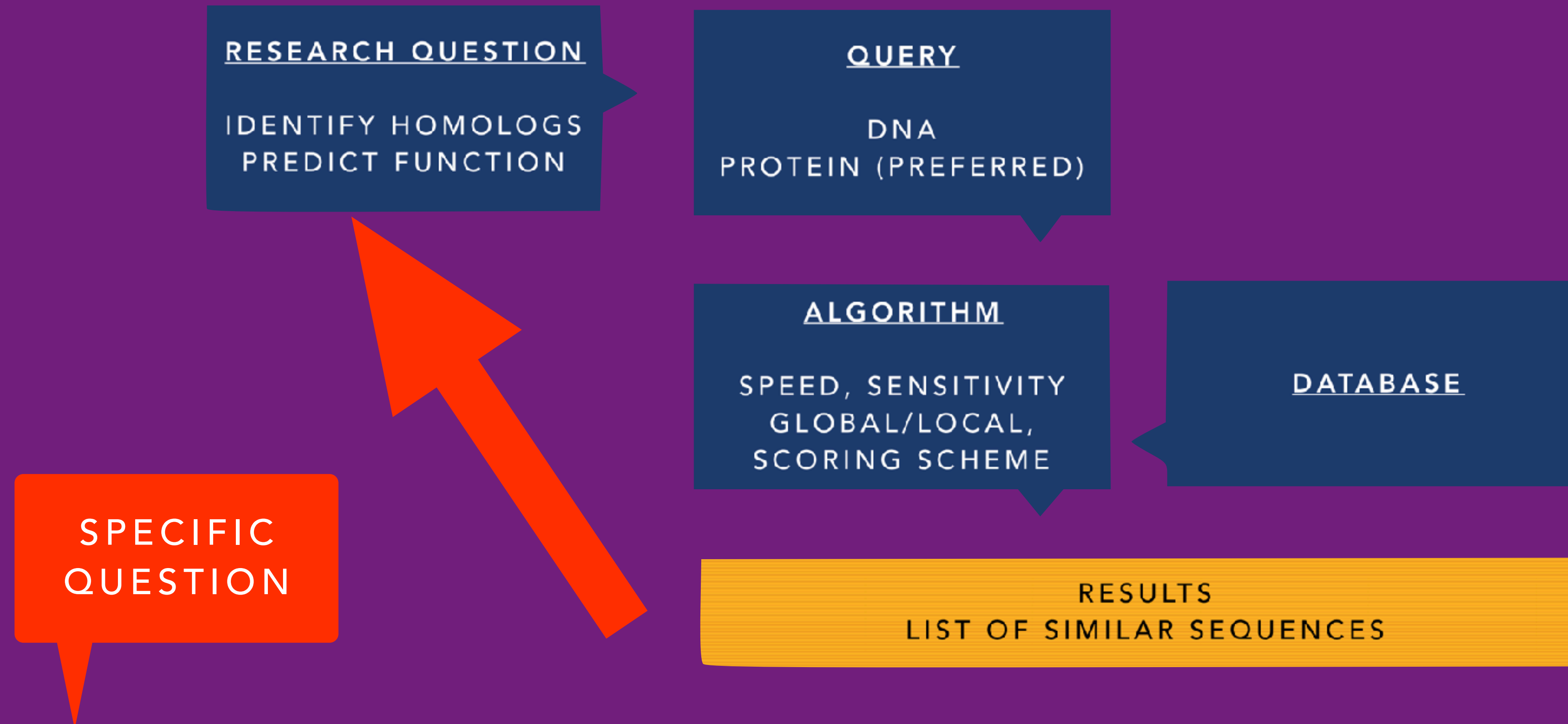


# DNA VS PROTEIN SEARCHES

- 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



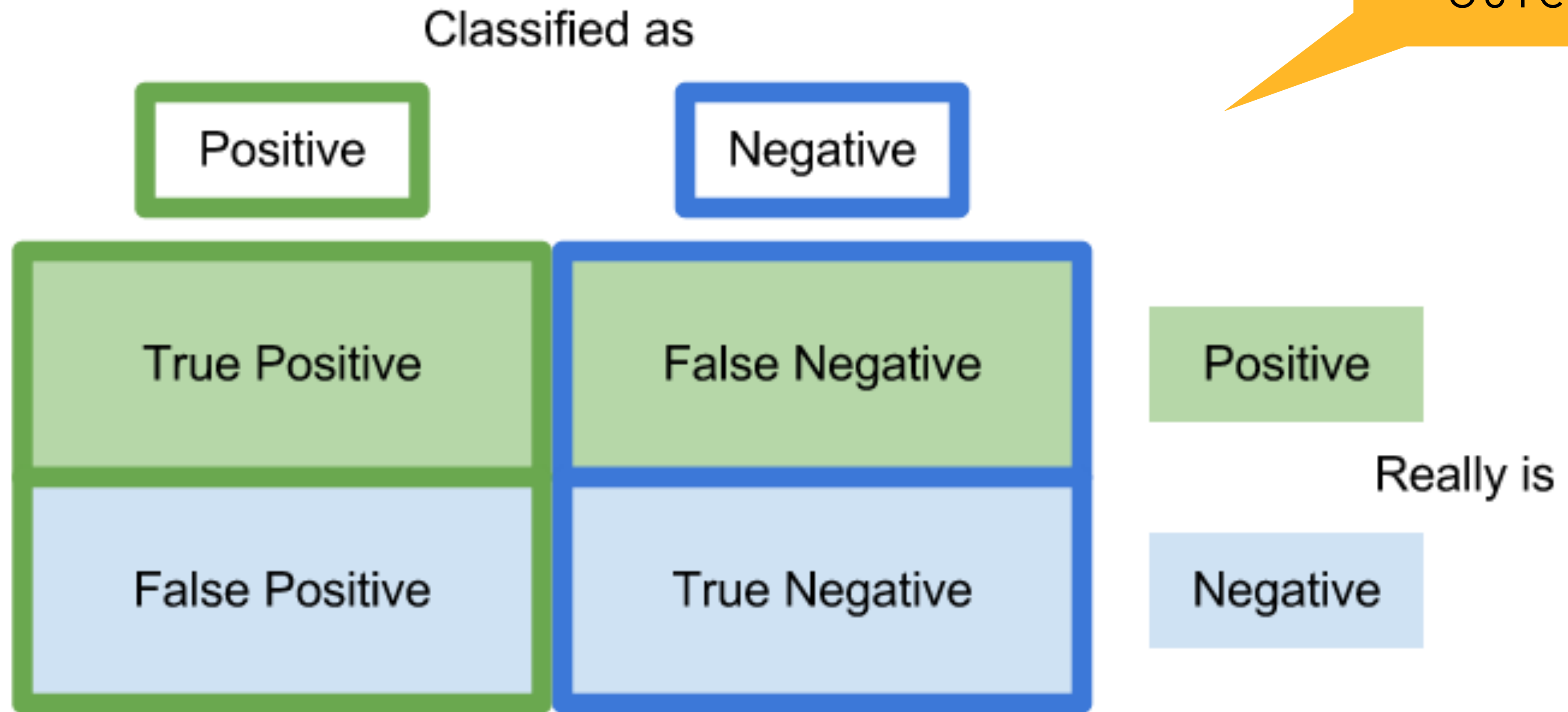
- How can we evaluate a database search?

# EVALUATING A DATABASE SEARCH

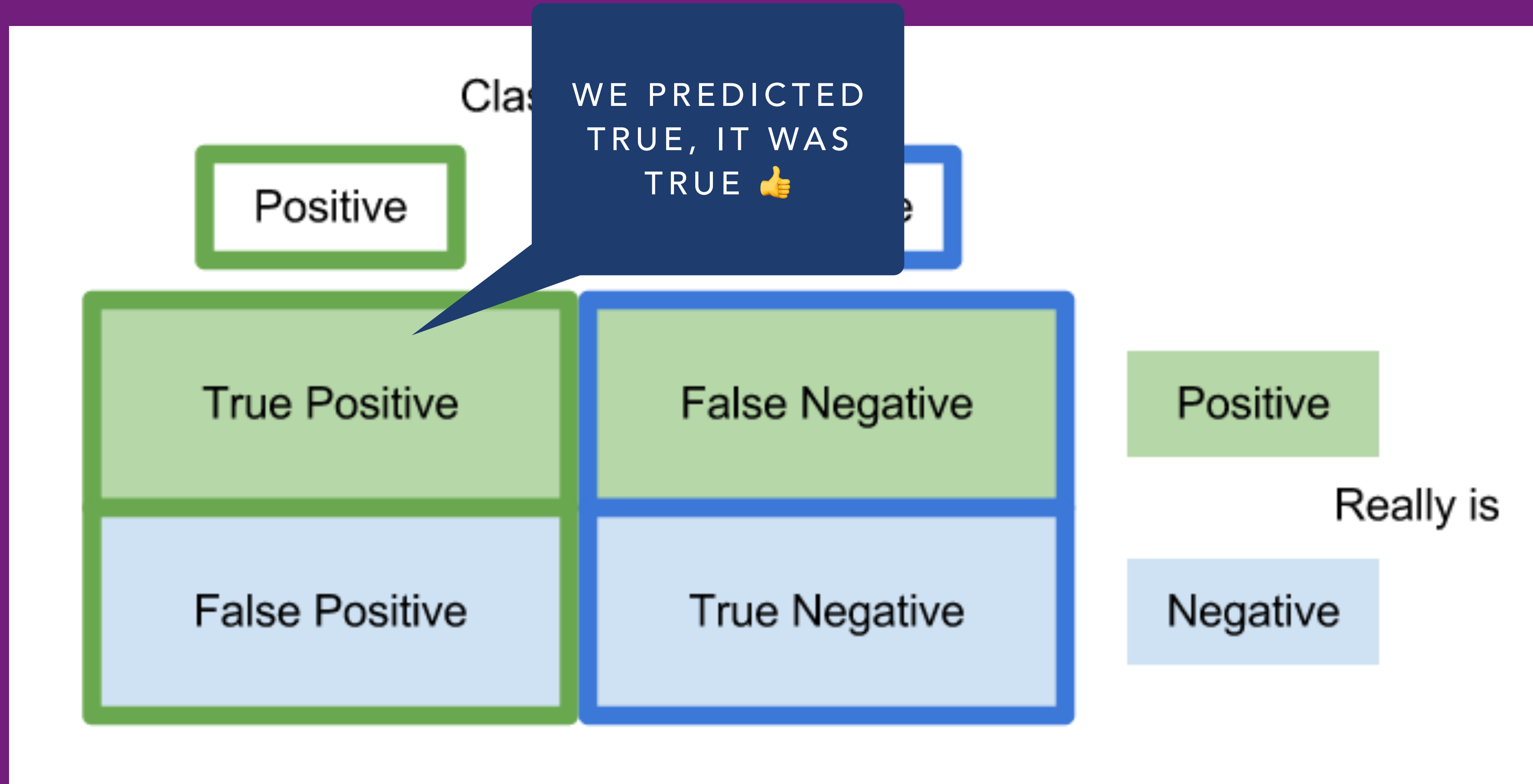
- If we were to develop a new algorithm, how could we compare it to existing algorithms?

# EVALUATING A DATABASE SEARCH

WHEN WE LOOK  
AT A RESULT  
COMPARED TO  
EXPECTED  
OUTCOME

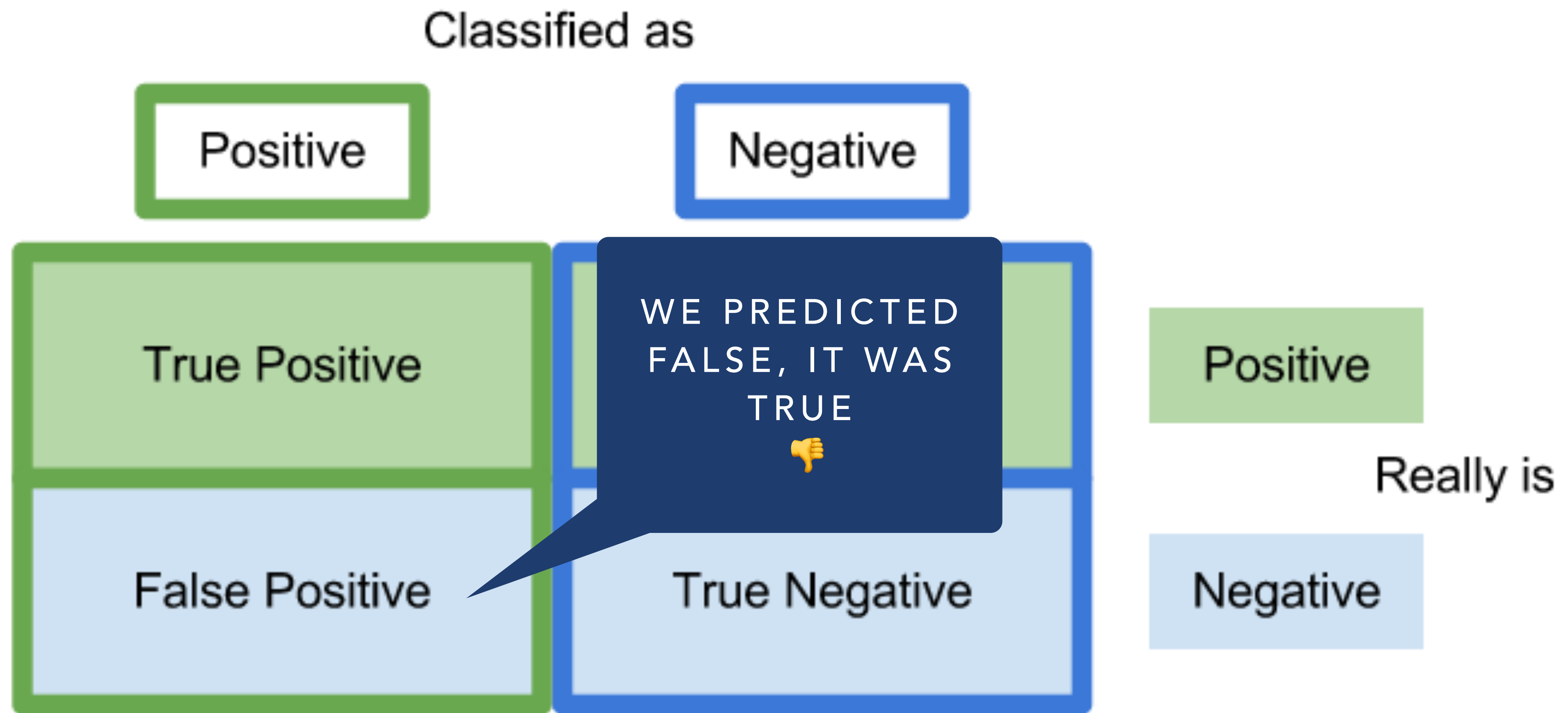


# EVALUATING A DATABASE SEARCH

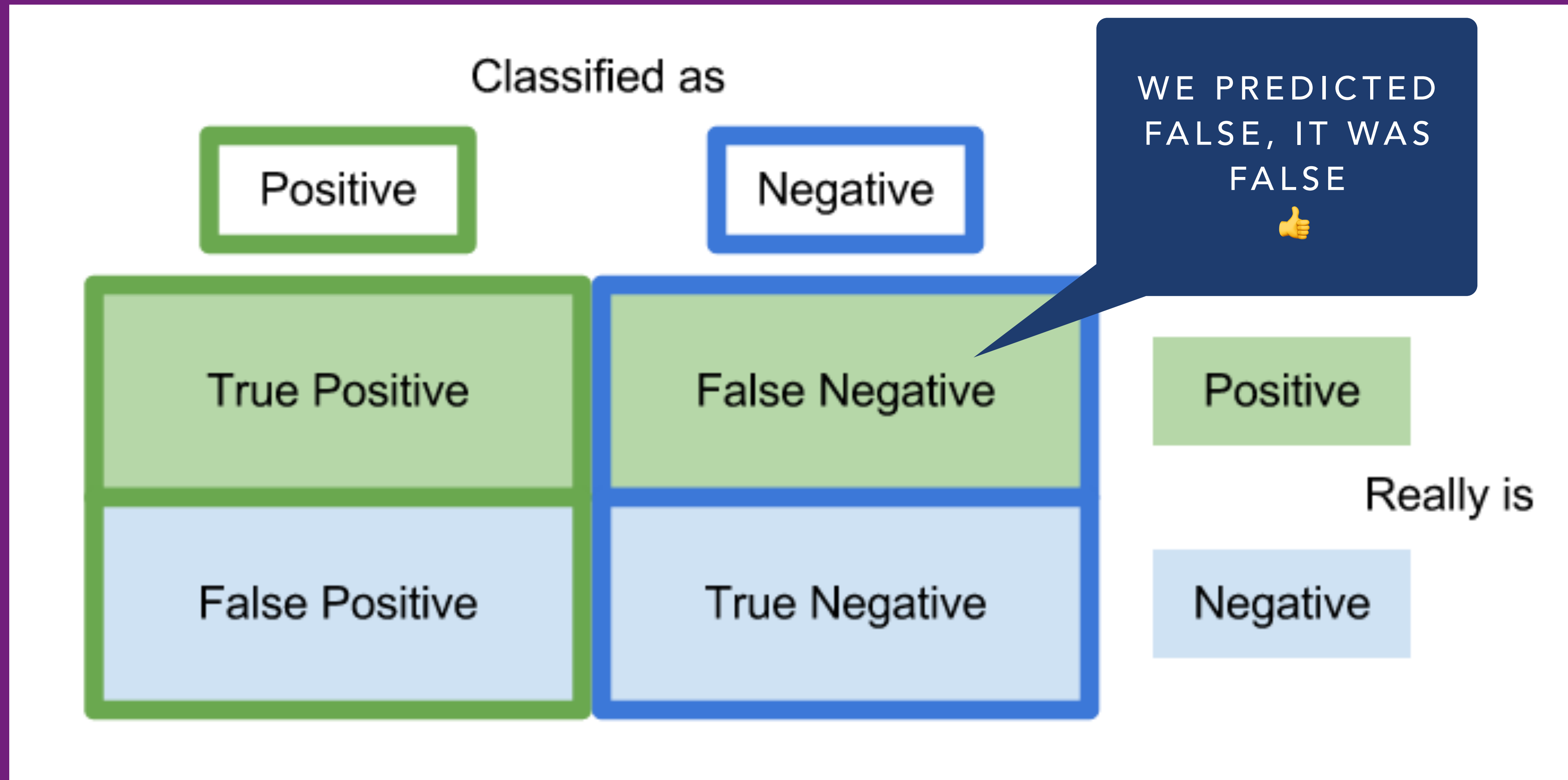




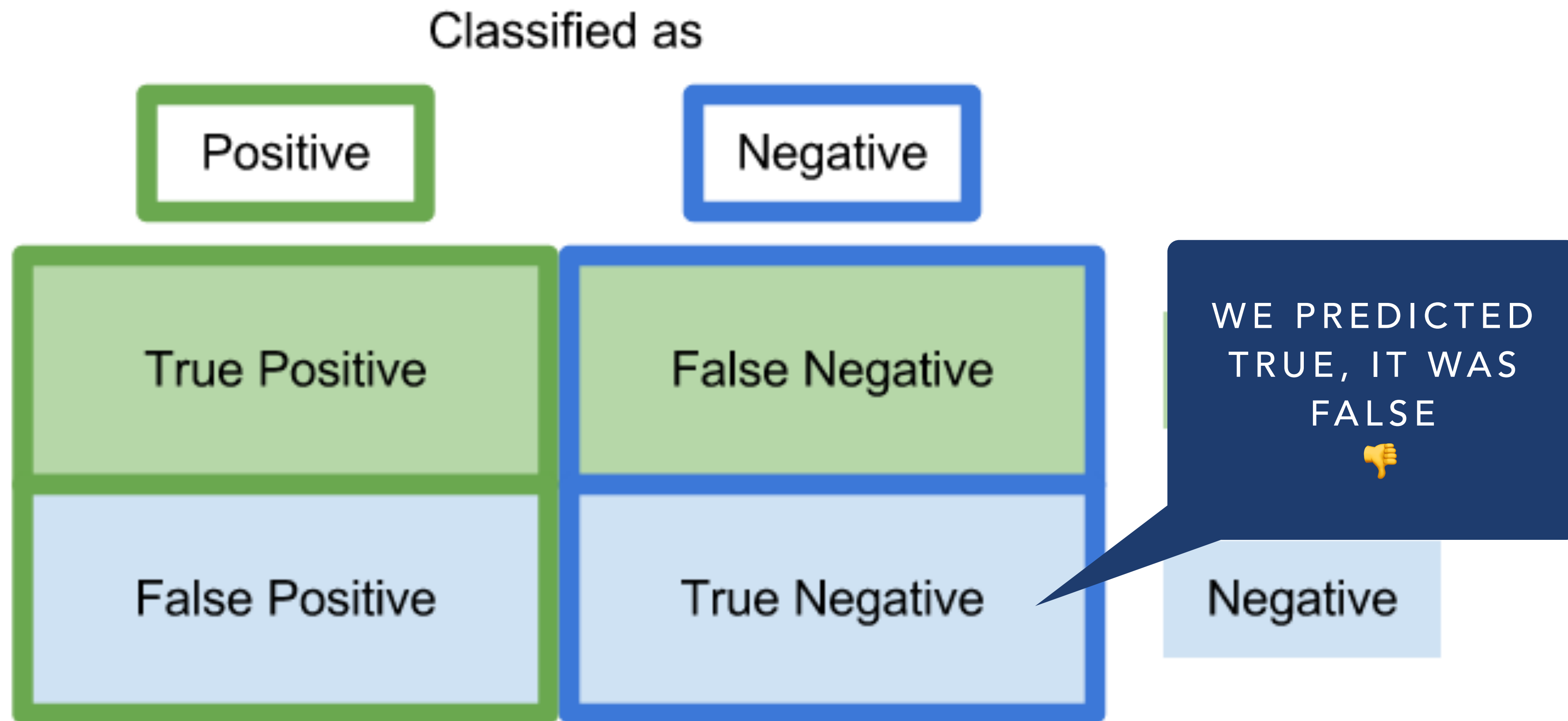
# EVALUATING A DATABASE SEARCH



# 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

THRESHOLD IS HEURISTIC

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

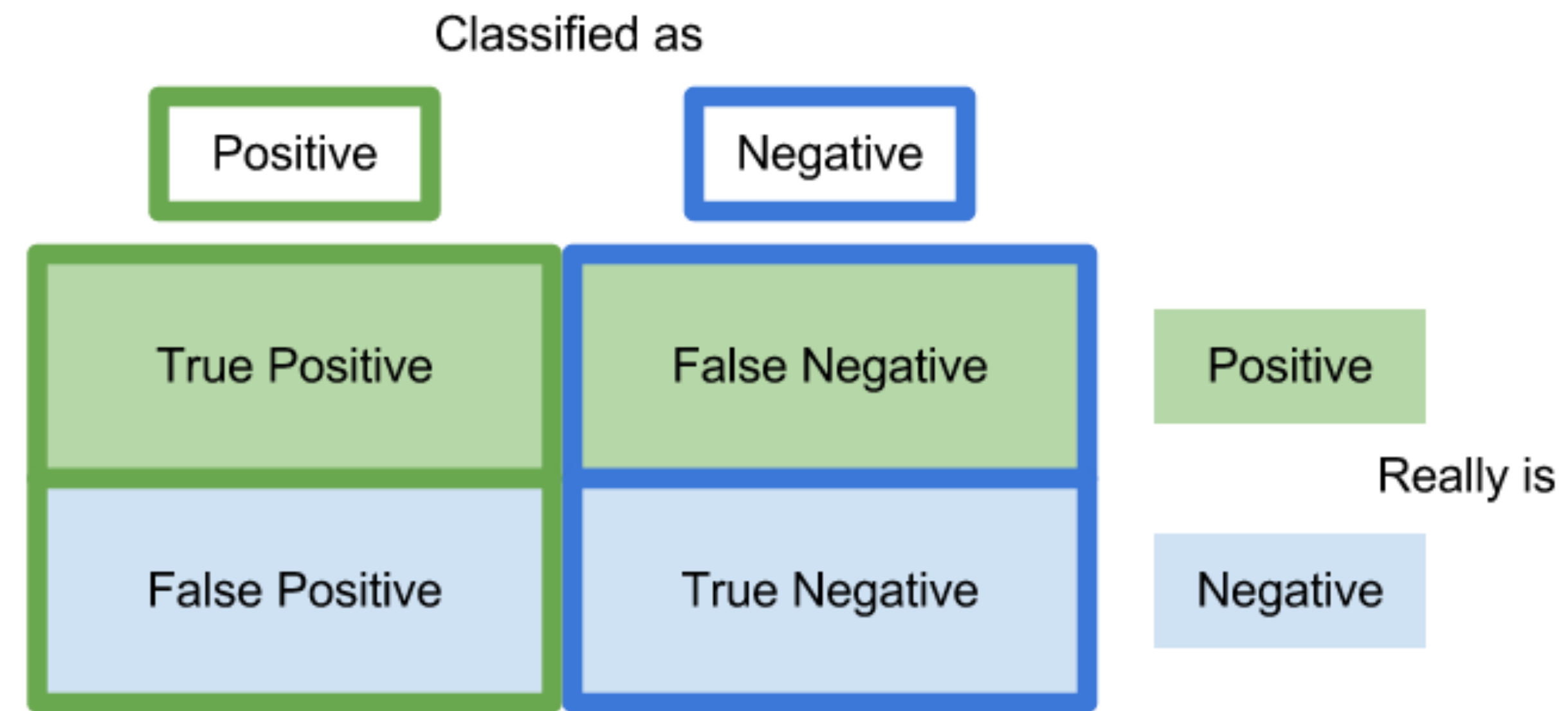
# EVALUATING A DATABASE SEARCH

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

		TP
		FP
		TP
> 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	

# EVALUATING A DATABASE SEARCH

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





# EVALUATING A DATABASE SEARCH

TRUE

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

ACCURATE  
PREDICT  
NOT TRUE

		TP
		FP
		TP
> 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

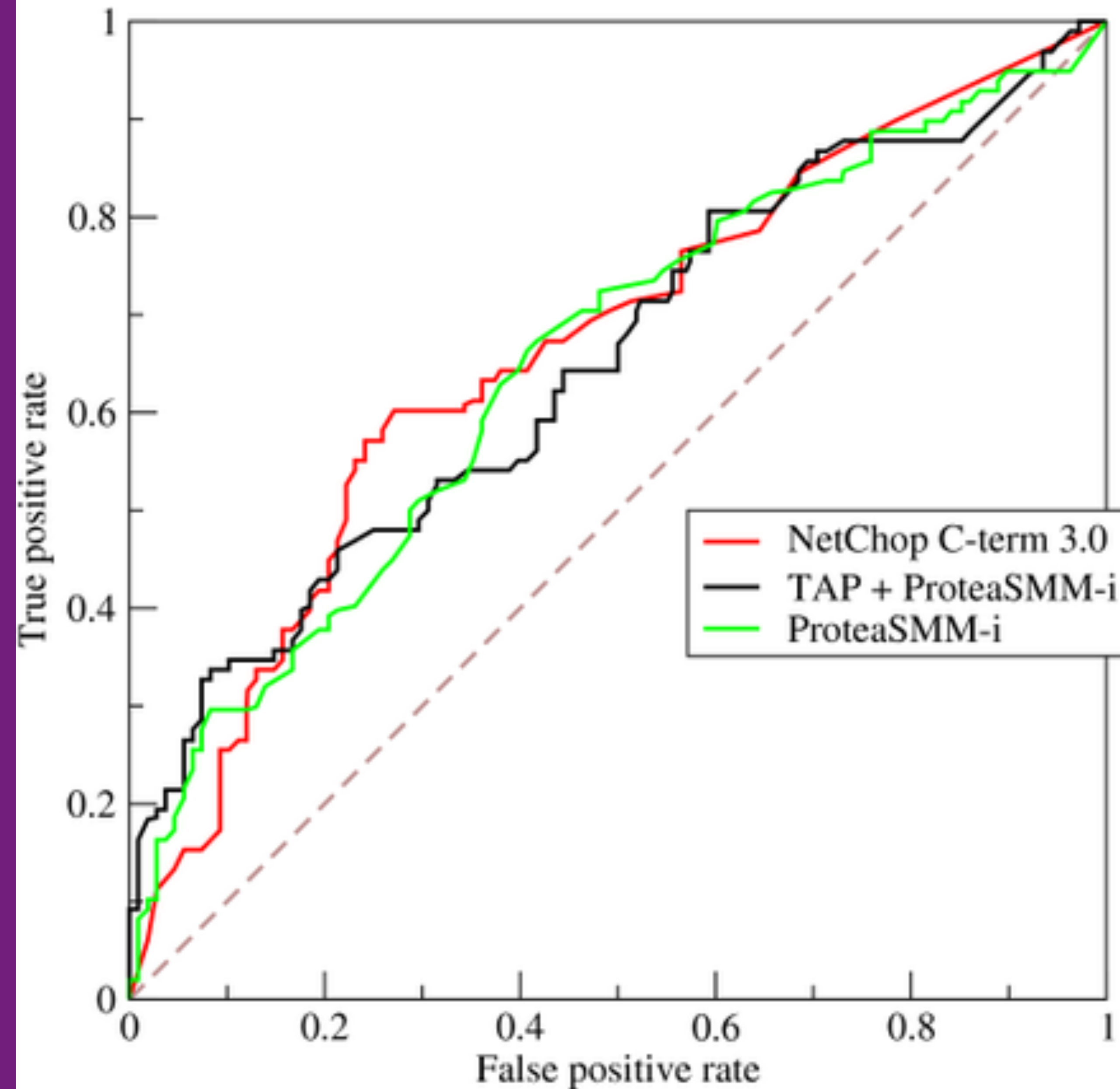
Sequences producing significant alignments:					Score (Bits)	E Value
sp	P00561.2	AK1H_ECOLI	RecName: Full=Bifunctional aspartokina...		1875	0.0
sp	P27725.1	AK1H_SERMA	RecName: Full=Bifunctional aspartokina...		1539	0.0
sp	P44505.1	AKH_HAEIN	RecName: Full=Bifunctional aspartokinas...		1108	0.0
sp	P57290.1	AKH_BUCAI	RecName: Full=Bifunctional aspartokinas...		1022	0.0
sp	Q8K9U9.1	AKH_BUCAP	RecName: Full=Bifunctional aspartokinas...		1010	0.0
sp	Q89AR4.1	AKH_BUCBP	RecName: Full=Bifunctional aspartokinas...		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 aspartokinas...		587	1e-178
sp	P00562.3	AK2H_ECOLI	RecName: Full=Bifunctional aspartokina...		366	2e-103
sp	Q57991.1	AK_METJA	RecName: Full=Probable aspartokinase; Al...		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	P41398.2	AK_CORGL	RecName: Full=Aspartokinase; AltName: Fu...		139	2e-32
sp	P41398.1	AK_CORPL	RecName: Full=Aspartokinase; AltName: Fu...		137	9e-32
sp	O8RON1.1	AK_COREF	RecName: Full=Aspartokinase; AltName: Fu...		135	3e-31
sp	Q59229.1	AK_BACSG	RecName: Full=Aspartokinase; AltName: Fu...		133	2e-30

HEURISTIC



# EVALUATING A DATABASE SEARCH

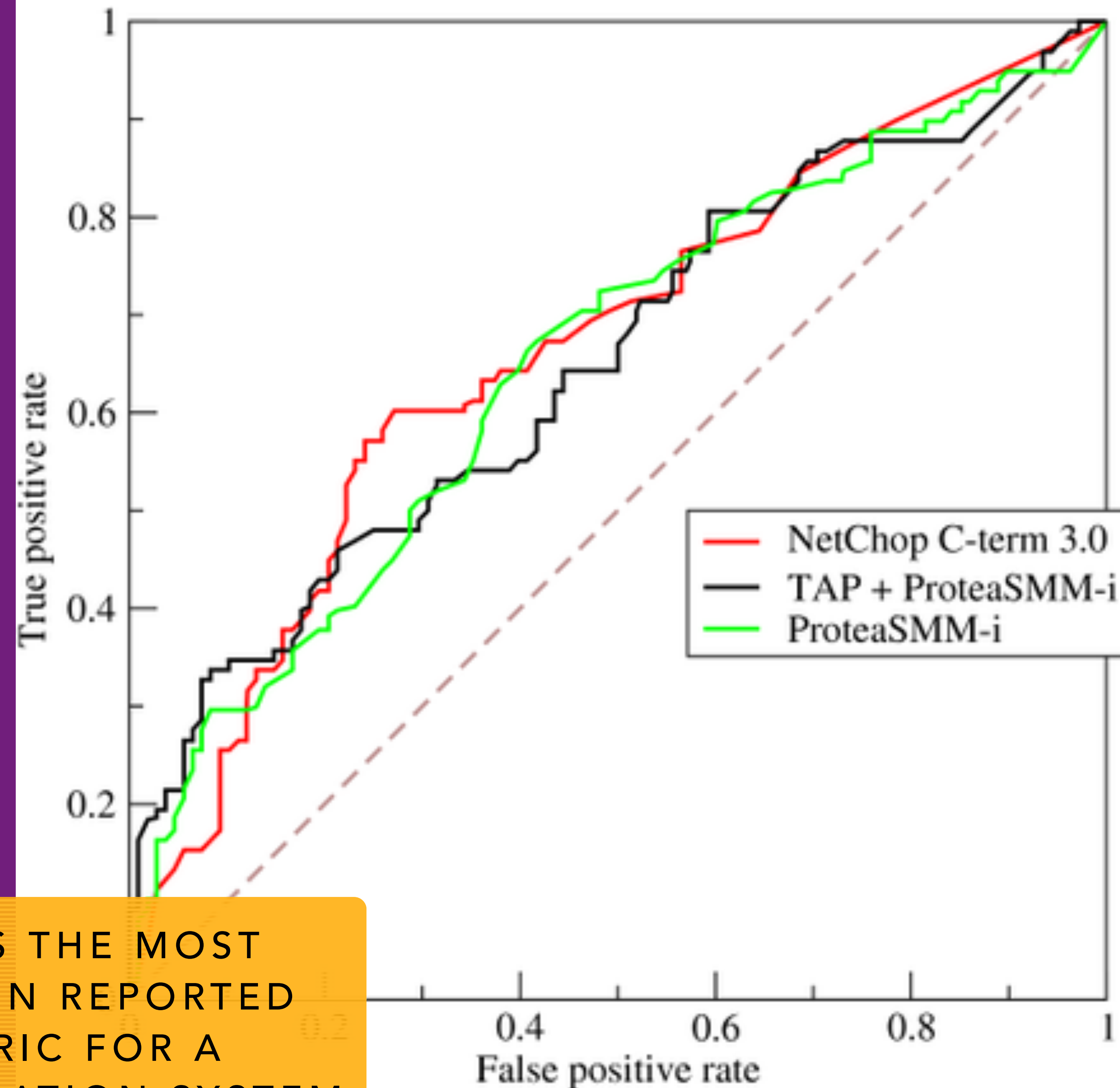
- Receiver operating characteristic (ROC) curve
  - Plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied
  - Sensitivity vs. Specificity
    - TPR/FPR



# EVALUATING A DATABASE SEARCH

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

AUC IS THE MOST COMMON REPORTED METRIC FOR A CLASSIFICATION SYSTEM





# DATABASE SEARCHING

Sequences producing significant alignments

OBSERVED TP/FP IN  
CHUNKS OF 5 TO FIND  
RATE

sp	P00561.2	AK1H_ECOLI	RecName: Full=Bifunctional aspartokina...
sp	P27725.1	AK1H_SERMA	RecName: Full=Bifunctional aspartokina...
sp	P44505.1	AKH_HAEIN	RecName: Full=Bifunctional aspartokinas...
sp	P57290.1	AKH_BUCAI	RecName: Full=Bifunctional aspartokinas...
sp	Q8K9U9.1	AKH_BUCAP	RecName: Full=Bifunctional aspartokinas...
sp	Q89AR4.1	AKH_BUCBP	RecName: Full=Bifunctional aspartokinas...
sp	P49079.1	AKH1_MAIZE	RecName: Full=Bifunctional aspartokina...
sp	O81852.1	AKH2_ARATH	RecName: Full=Bifunctional aspartokina...
sp	Q9SA18.1	AKH1_ARATH	RecName: Full=Bifunctional aspartokina...
sp	P49080.1	AKH2_MAIZE	RecName: Full=Bifunctional aspartokina...
sp	P37142.1	AKH_DAUCA	RecName: Full=Bifunctional aspartokinas...
sp	P00562.3	AK2H_ECOLI	RecName: Full=Bifunctional aspartokina...
sp	Q57991.1	AK_METJA	RecName: Full=Probable aspartokinase; Al...
sp	Q5B998.1	DHOM_EMENI	RecName: Full=Homoserine dehydrogenase...
sp	O94671.1	DHOM_SCHPO	RecName: Full=Probable homoserine dehy...
sp	Q9S702.1	AK3_ARATH	RecName: Full=Aspartokinase 3, chloropl...
sp	Q9LYU8.1	AK1_ARATH	RecName: Full=Aspartokinase 1, chloropl...

	Tp	Fp
5	5	0
5-10	5	0
10-15	3	2
15-20	1	4
20-25	3	2
510	21	9

590	6e-180
588	2e-179
587	1e-178
366	2e-103
302	7e-85
234	2e-63
219	2e-58
208	1e-53
201	2e-51



# 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 aspartokinas...	587	1e-178
sp	P00562.3	AK2H_ECOLI	RecName: Full=Bifunctional aspartokina...	366	2e-103
sp	Q57991.1	AK_METJA	RecName: Full=Probable aspartokinase; Al...	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



```
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

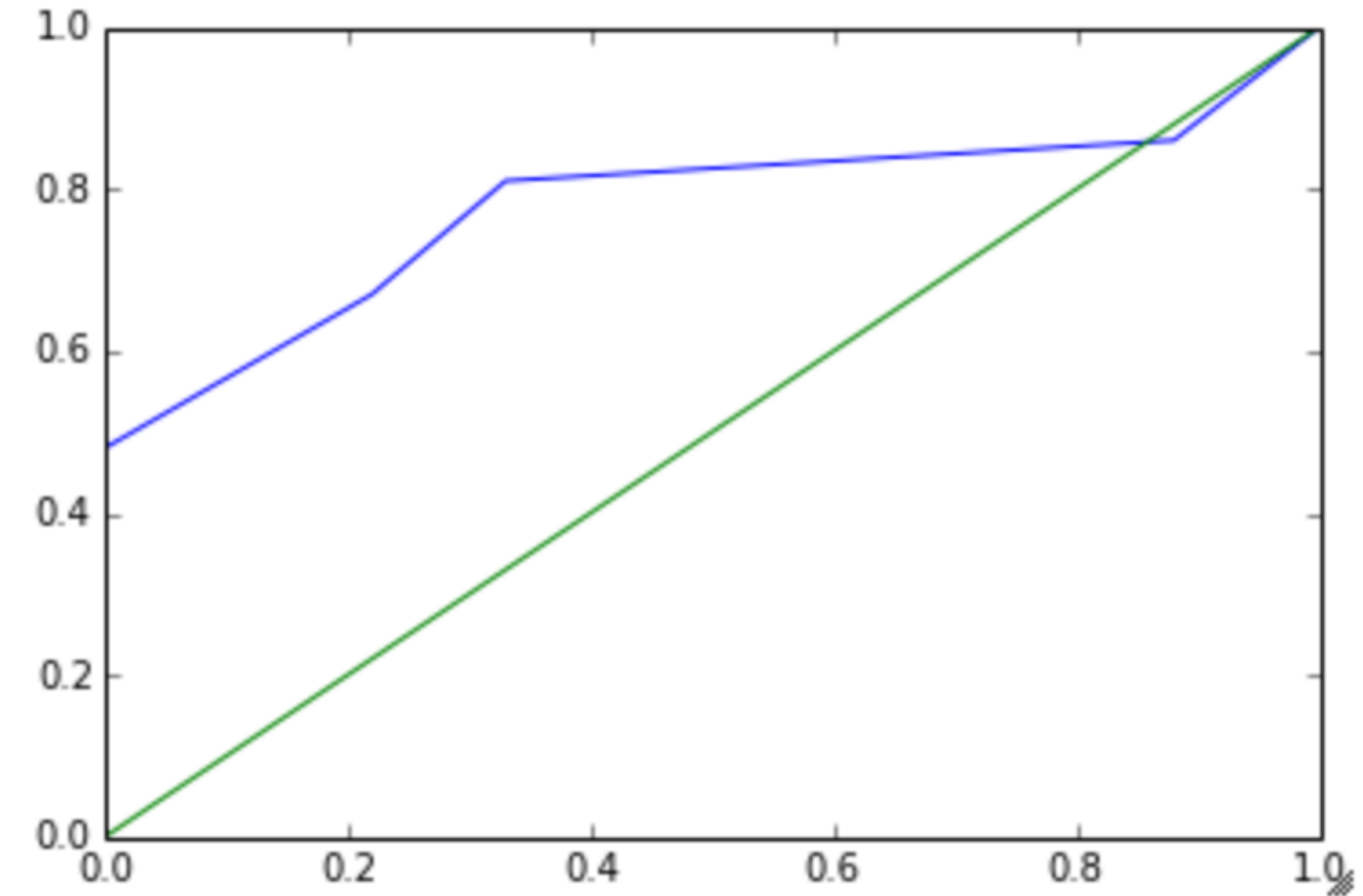
fpr = [0,0,.22,.33,.88,1]
tpr = [.24,.48,.67,.81,.86,1]

# 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
```



0.77875



# 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 sequences already available in databases. Because of the algorithm's efficiency on many computers, it now become a routine procedure for molecular biologists. The method efficiently aligns identical and differing residues in those regions by means of an amino acid substitution matrix giving high scores to those amino acid replacements which occur frequently in nature. The program designed to search protein databases very rapidly. For example, comparison of a sequence 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

- Heavily used for searching databases until advent of BLAST

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

## **Rapid and sensitive protein similarity searches.**

Lipman DJ, Pearson WR.

### **Abstract**

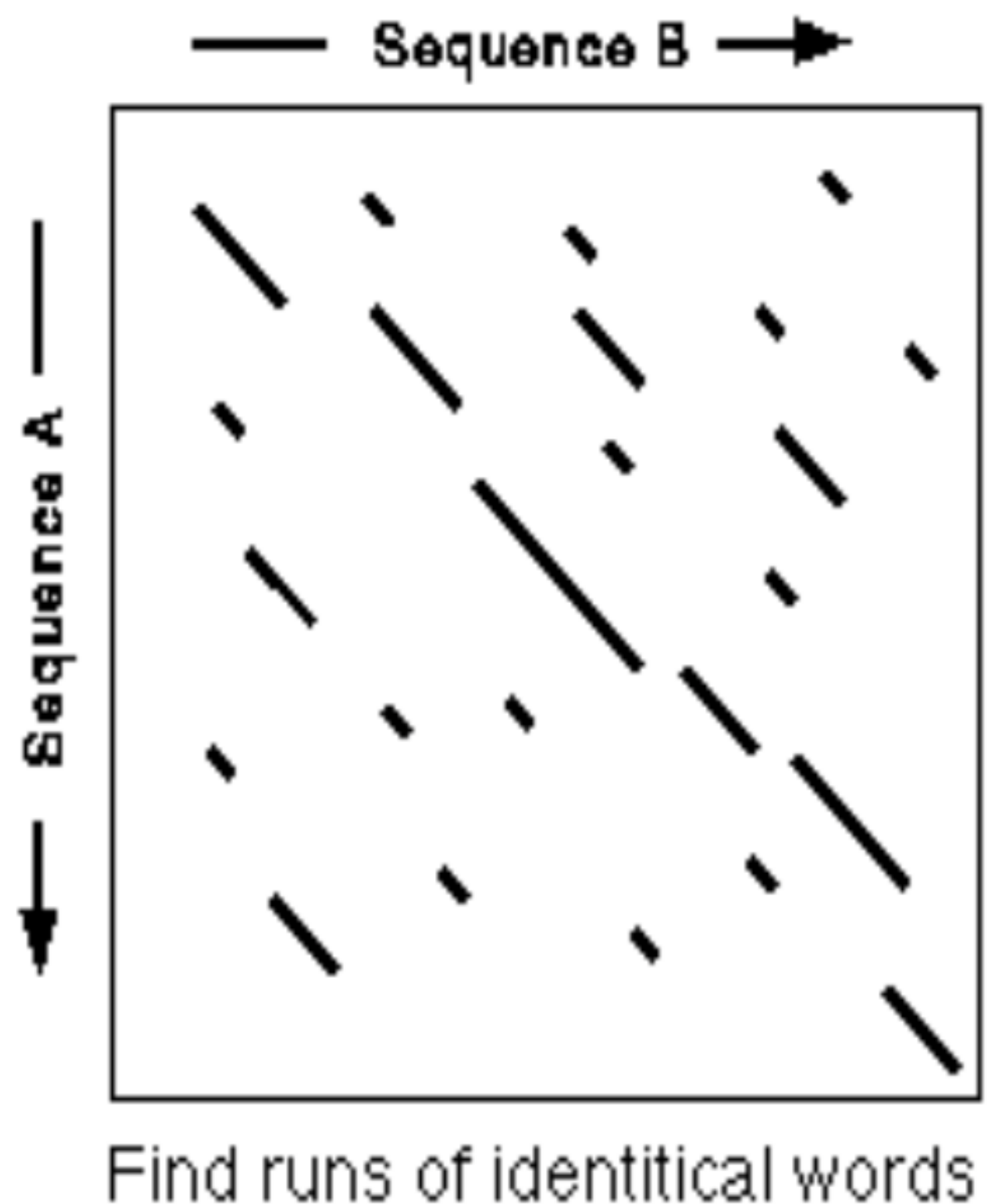
An algorithm was developed which facilitates the search for similar sequences already available in databases. Because of the algorithm's efficiency, it has now become a routine procedure for molecular biologists. The method aligns identical and differing residues in those regions by means of giving high scores to those amino acid replacements which occur frequently. This program designed to search protein databases very rapidly. For example, a search in the National Biomedical Research Foundation library would take less than 1 minute on a microcomputer (IBM PC).

PMID: 2983426 [PubMed - indexed for MEDLINE]



# FASTA

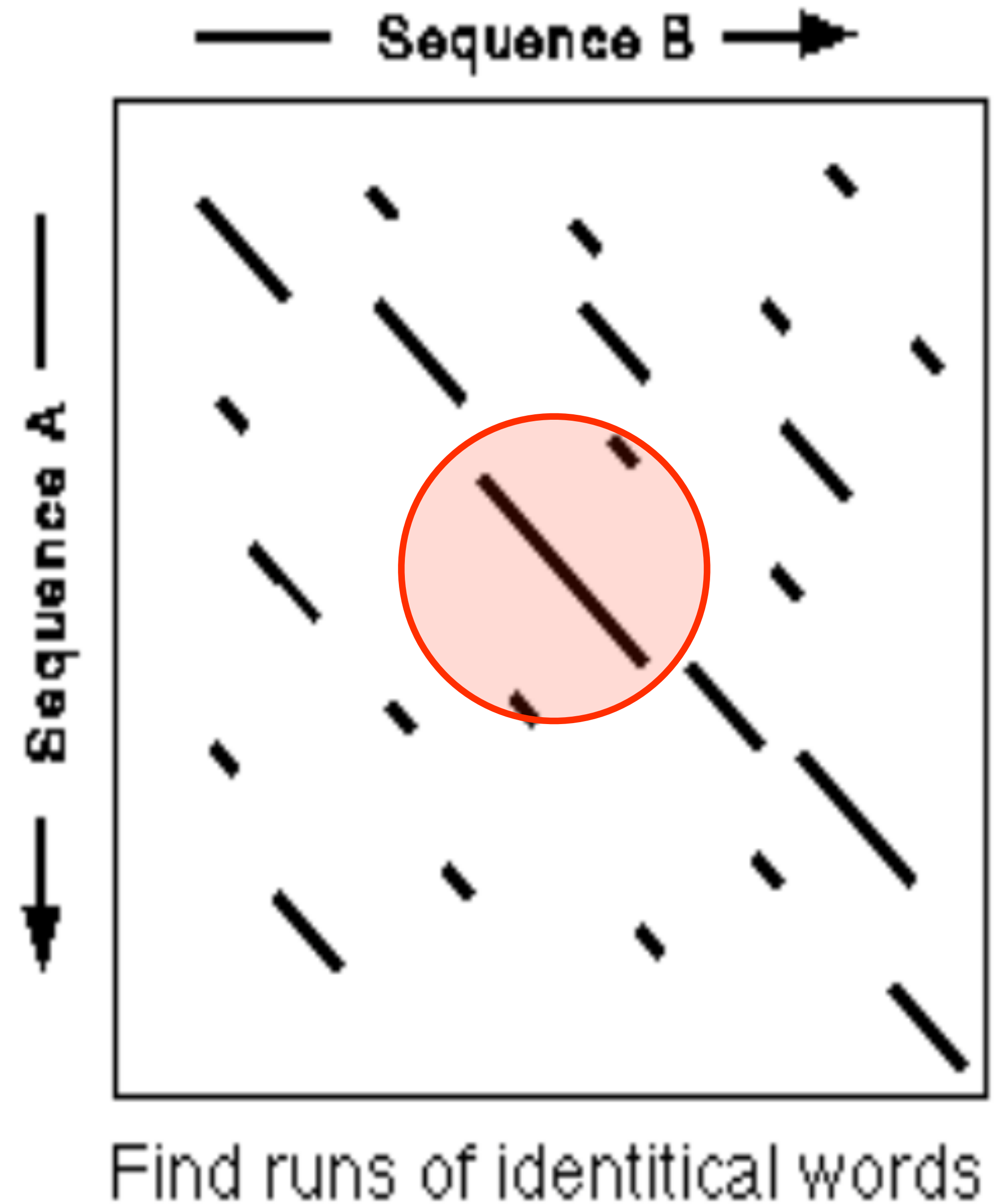
- Word method
  - Identify **exact** matches between words(k) in query and db sequence
    - DNA ~6 bases
    - Protein ~2 amino acids
- Only search in areas that contain word matches
  - Hash table lookup





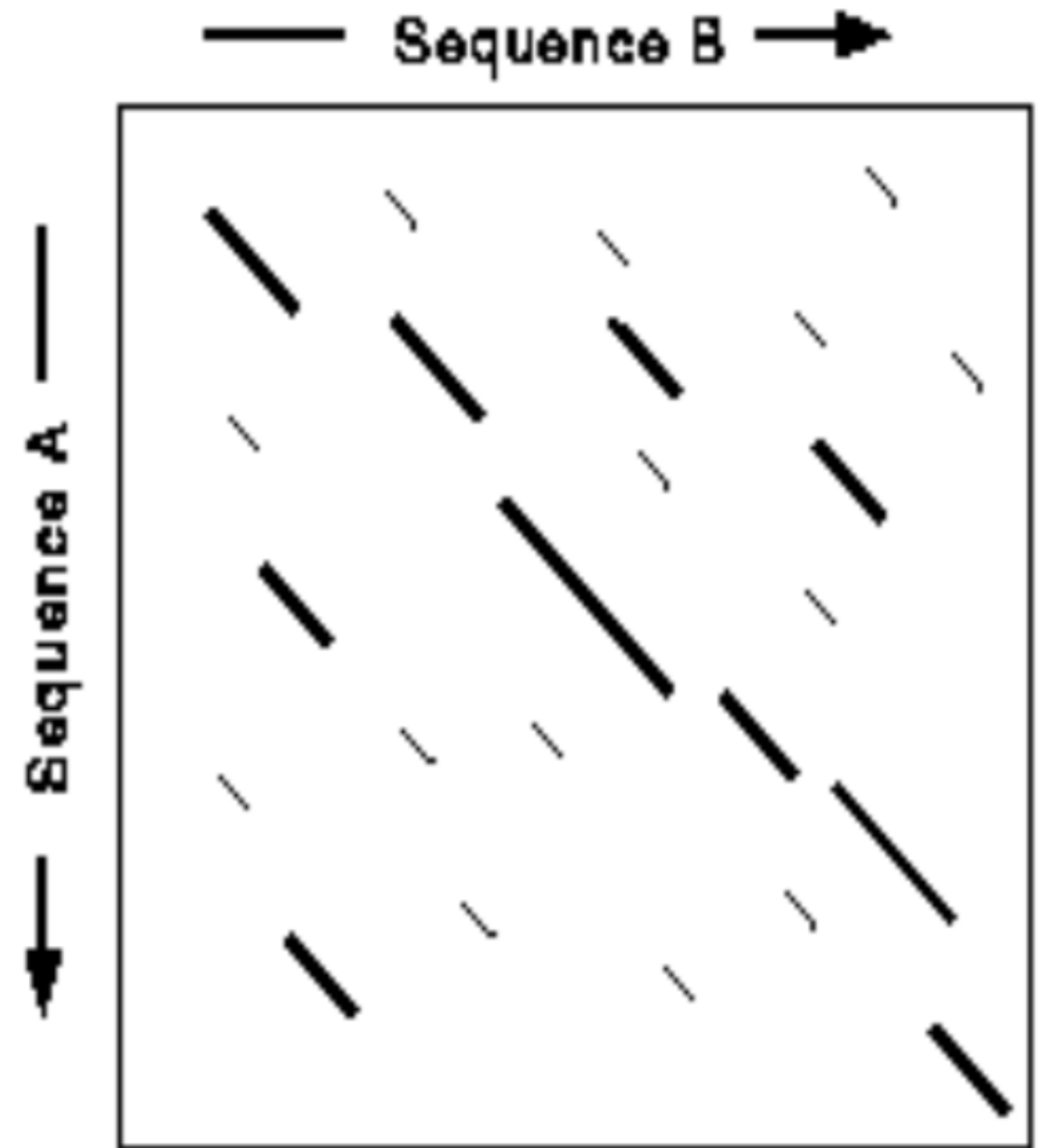
# FASTA METHOD

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



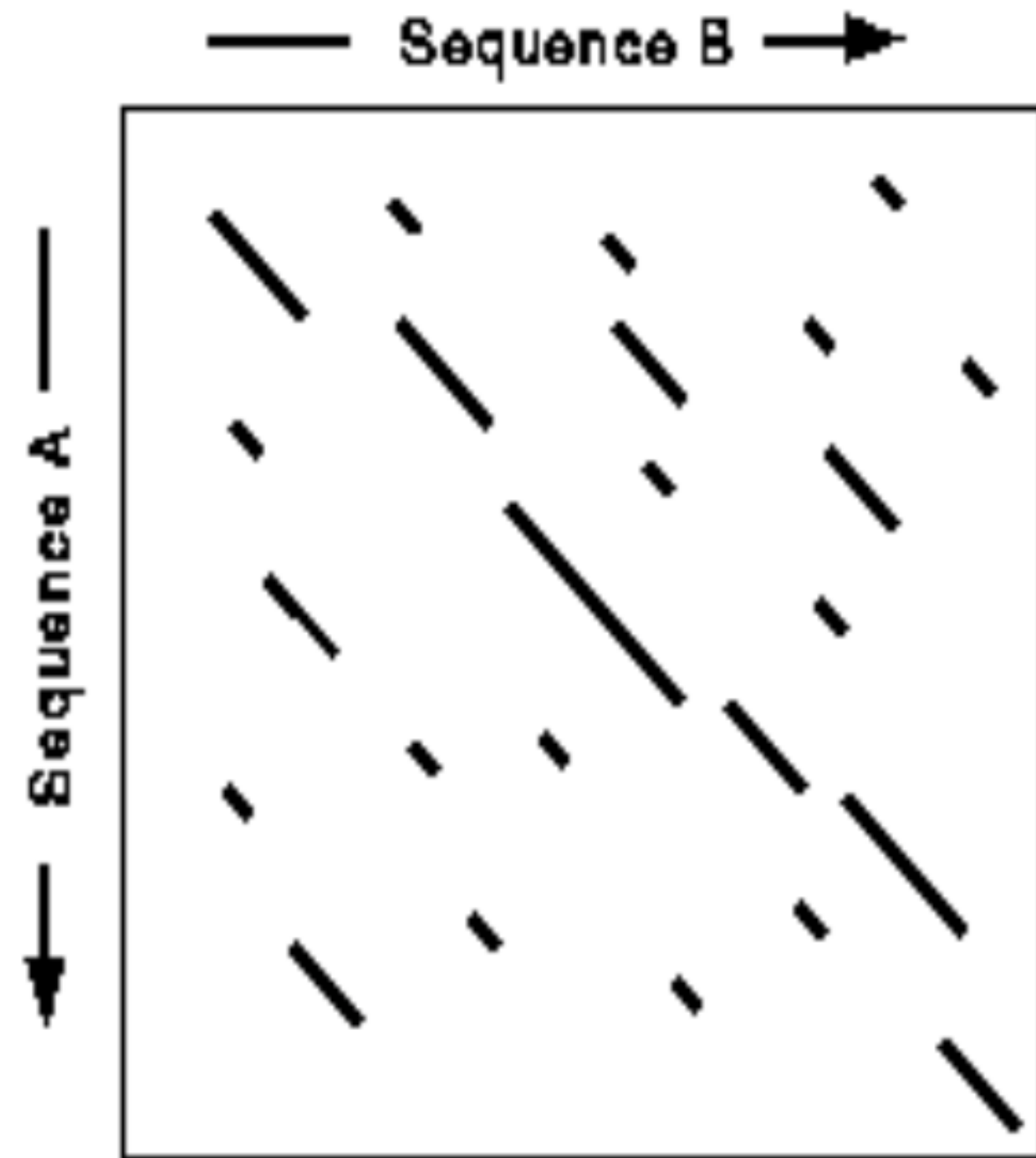
# FASTA METHOD

- 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

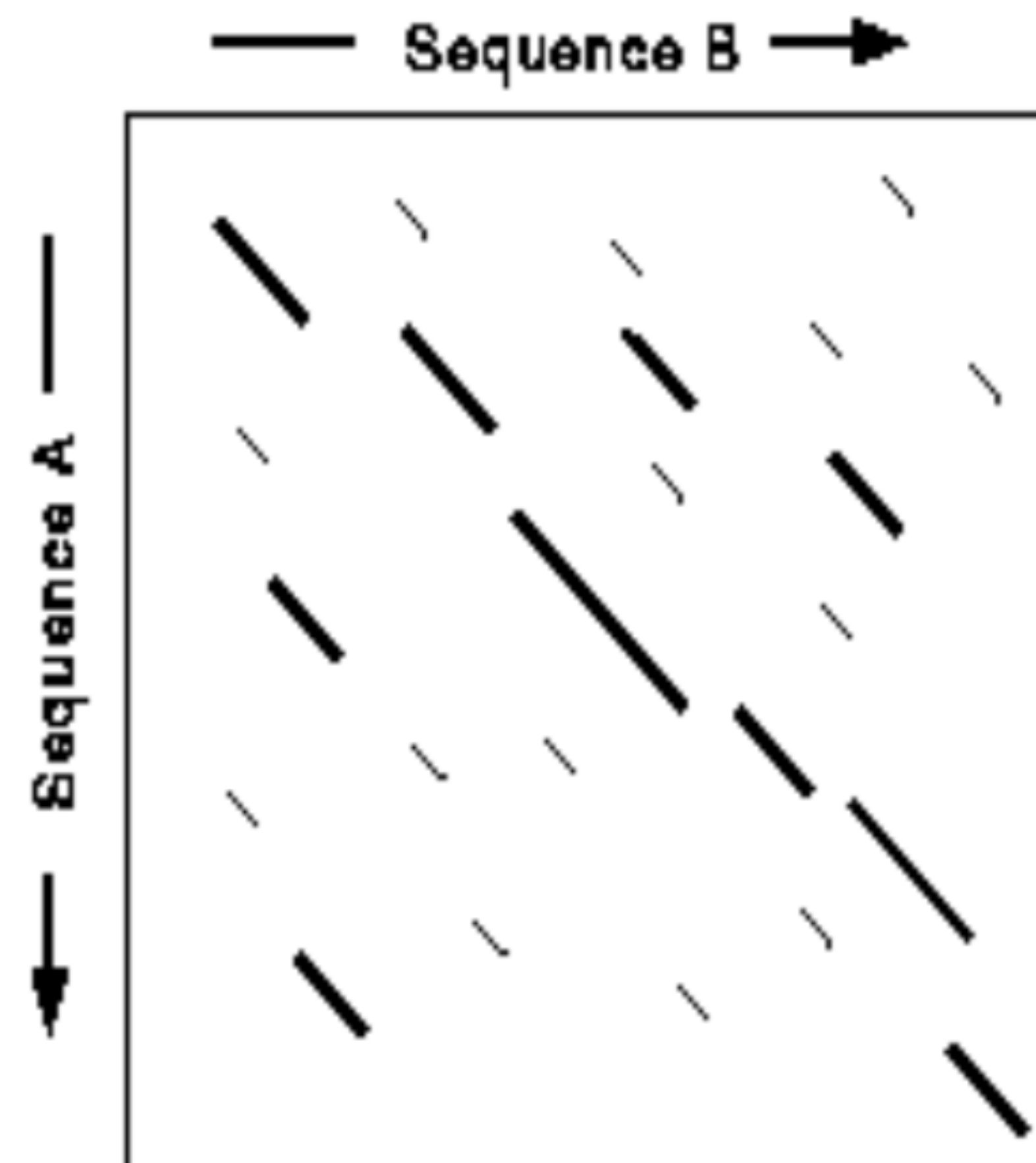


Re-score using PAM matrix  
Keep top scoring segments

# FASTA METHOD



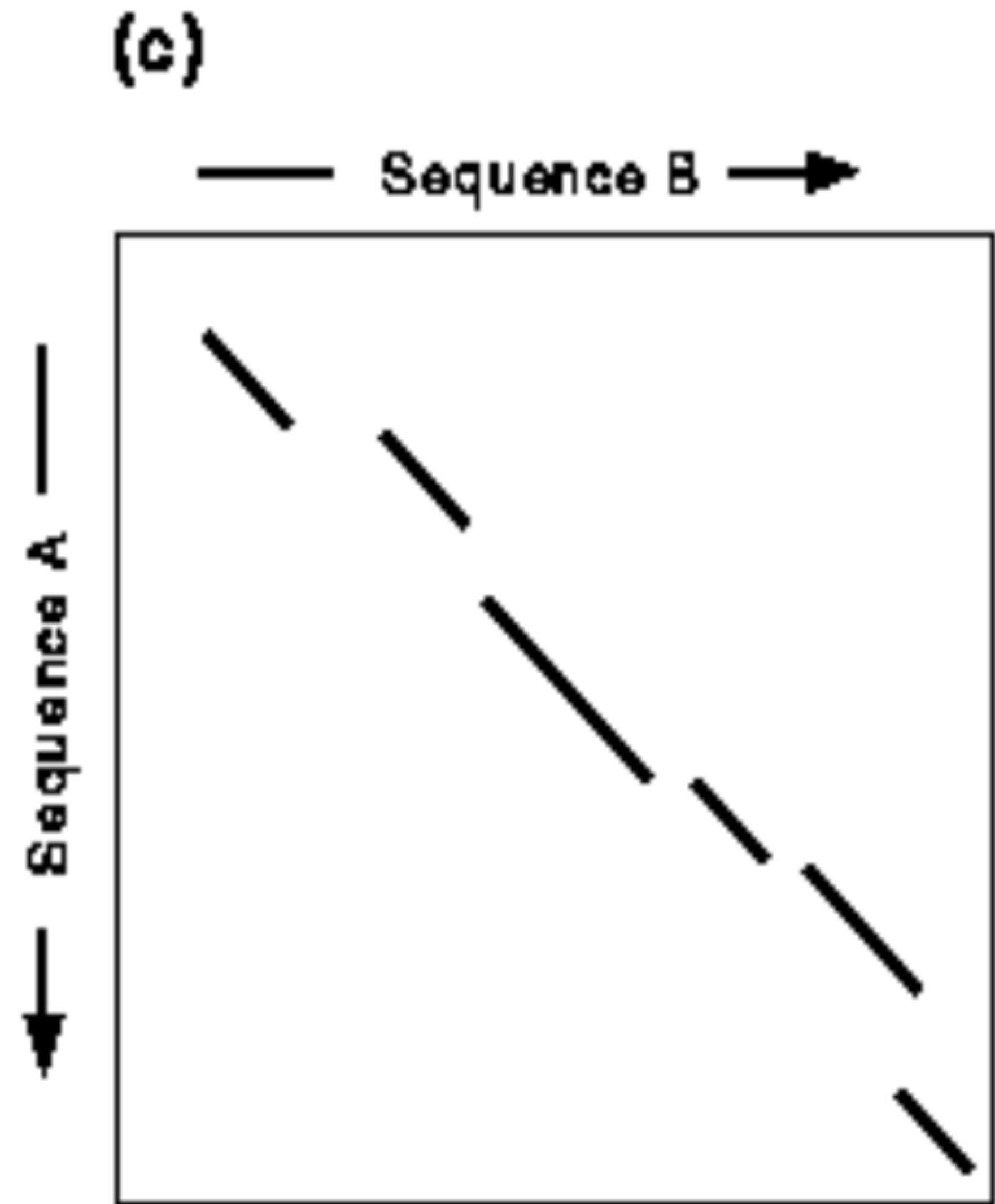
Find runs of identical words



Re-score using PAM matrix  
Keep top scoring segments

# FASTA METHOD

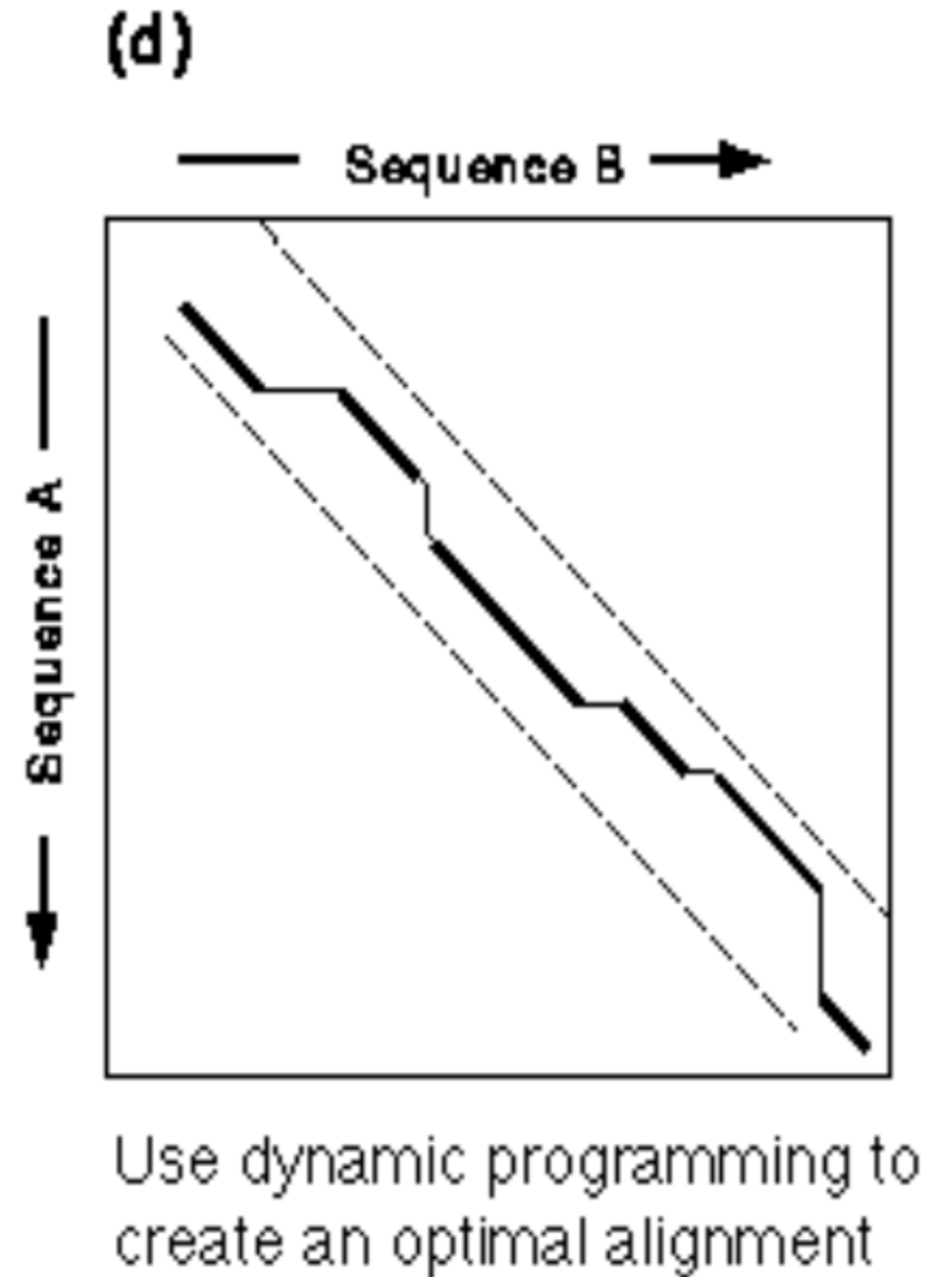
- Step 3. Join regions together, including penalties for gaps
- Result:
  - Unoptimized alignment with gaps
  - Reported as the "initn" score



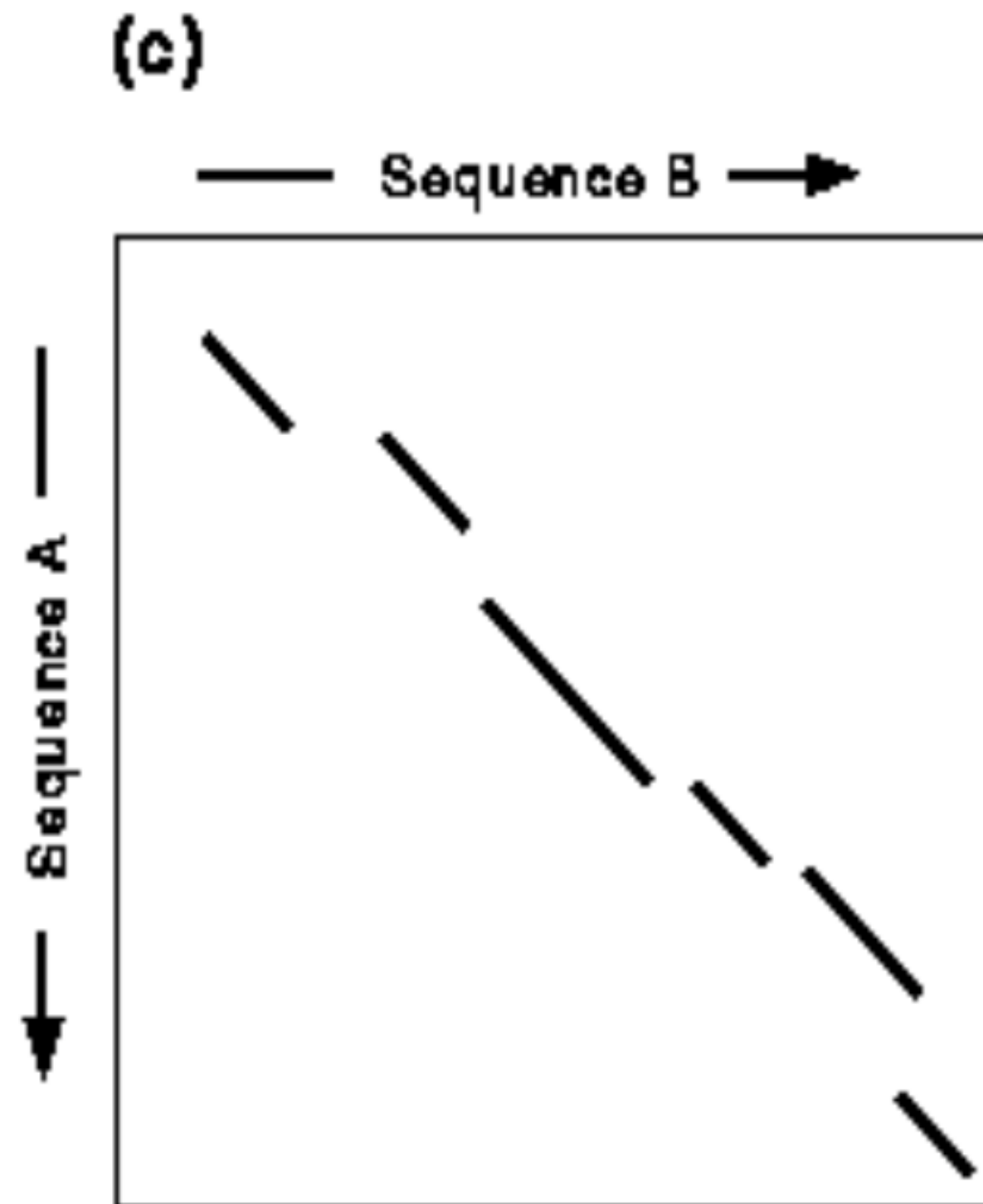
Join segments using gaps,  
eliminate other segments

# FASTA METHOD

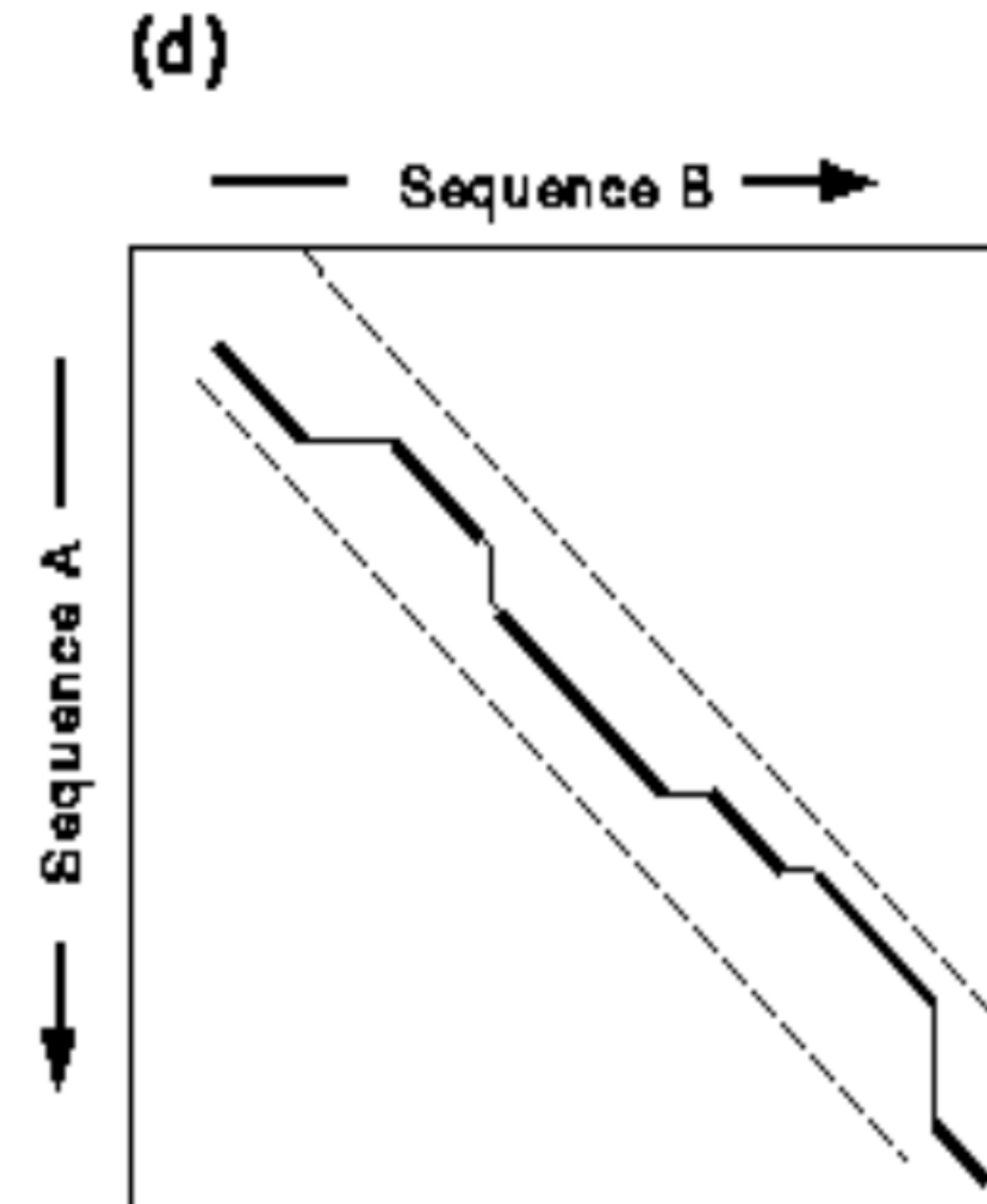
- Step 4. Use dynamic programming in a band 32 residues wide around the best "initn" score
- Result:
  - Optimized alignment with gaps
  - Reported as the "opt" score



# FASTA METHOD



Join segments using gaps,  
eliminate other segments

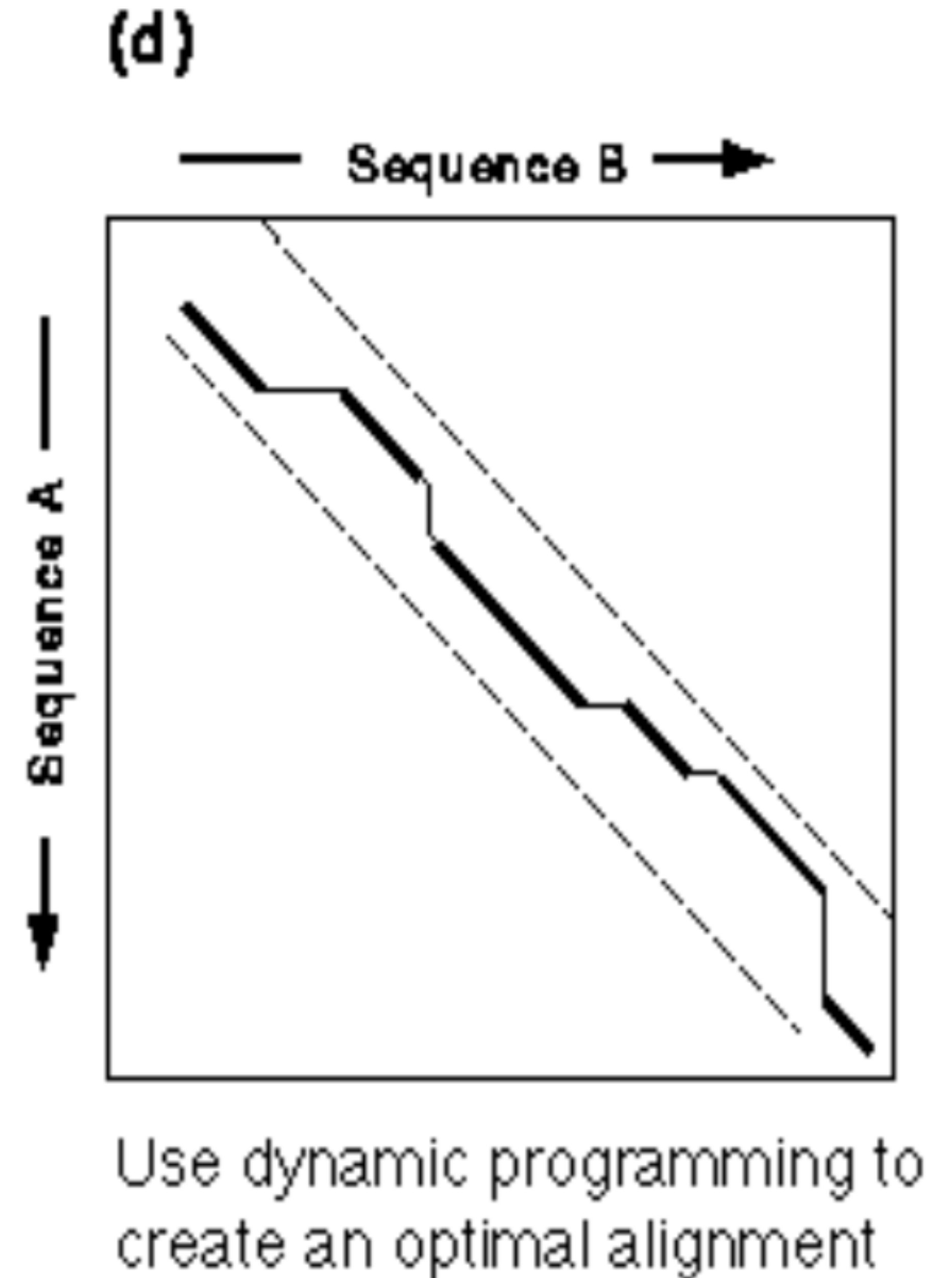


Use dynamic programming to  
create an optimal alignment



# 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



# FASTA LIMITATIONS

FASTA FAILS

- 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/sss/fasta/>

EMBL-EBI

# FASTA

Protein | Nucleotide | Genomes | Proteomes | Whole Genomes | Ser

Help & Documentation

Tools > Sequence Similarity Searching > FASTA

## Protein Similarity Search

This tool provides sequence similarity searching against protein databases. It offers a heuristic search with a protein query. FASTX and FASTY translate a nucleotide query (local), GGSEARCH (global) and GLSEARCH (global query, local).

STEP 1 - Select your databases

### PROTEIN DATABASES

1 Databank Selected

- ☒ UniProt Knowledgebase
- ☐ UniProtKB/Swiss-Prot
- ☐ UniProtKB/Swiss-Prot isoforms
- ☐ UniProtKB/TrEMBL
- UniProtKB Taxonomic Subsets
- UniProt Clusters
- Databases

STEP 2 - Enter your input sequence

Enter or paste a **PROTEIN** sequence in any supported format:

or Upload a file: **Choose File** no file selected



# BIOINFORMATICS

(FOR COMPUTER SCIENTISTS)

MPCS56420  
AUTUMN 2018  
SESSION 4



THE UNIVERSITY OF  
CHICAGO