



Western Washington University
Western CEDAR

WWU Honors Program Senior Projects

WWU Graduate and Undergraduate Scholarship

Fall 12-5-2017

Constrained Sequence Alignment

Kyle Daling

Western Washington University

Follow this and additional works at: https://cedar.wvu.edu/wwu_honors



Part of the [Biology Commons](#), and the [Numerical Analysis and Scientific Computing Commons](#)

Recommended Citation

Daling, Kyle, "Constrained Sequence Alignment" (2017). *WWU Honors Program Senior Projects*. 58.
https://cedar.wvu.edu/wwu_honors/58

This Project is brought to you for free and open access by the WWU Graduate and Undergraduate Scholarship at Western CEDAR. It has been accepted for inclusion in WWU Honors Program Senior Projects by an authorized administrator of Western CEDAR. For more information, please contact westerncedar@wvu.edu.

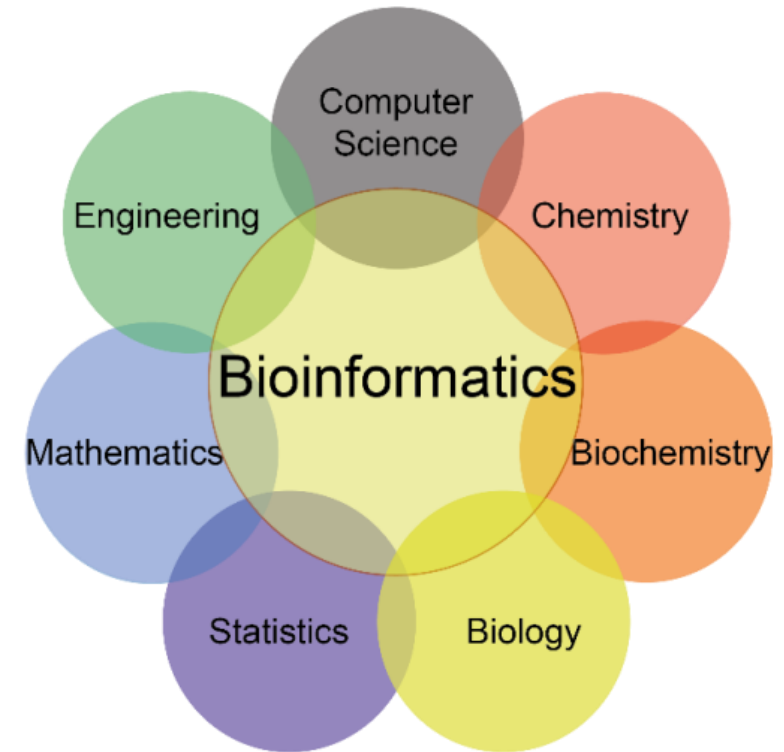
Constrained Sequence Alignment

Kyle Daling

Background

What is bioinformatics?

- Interdisciplinary field
 - Computer Science
 - Biology
 - Mathematics
 - Engineering
- Analyze and interpret biological data



Why bioinformatics?

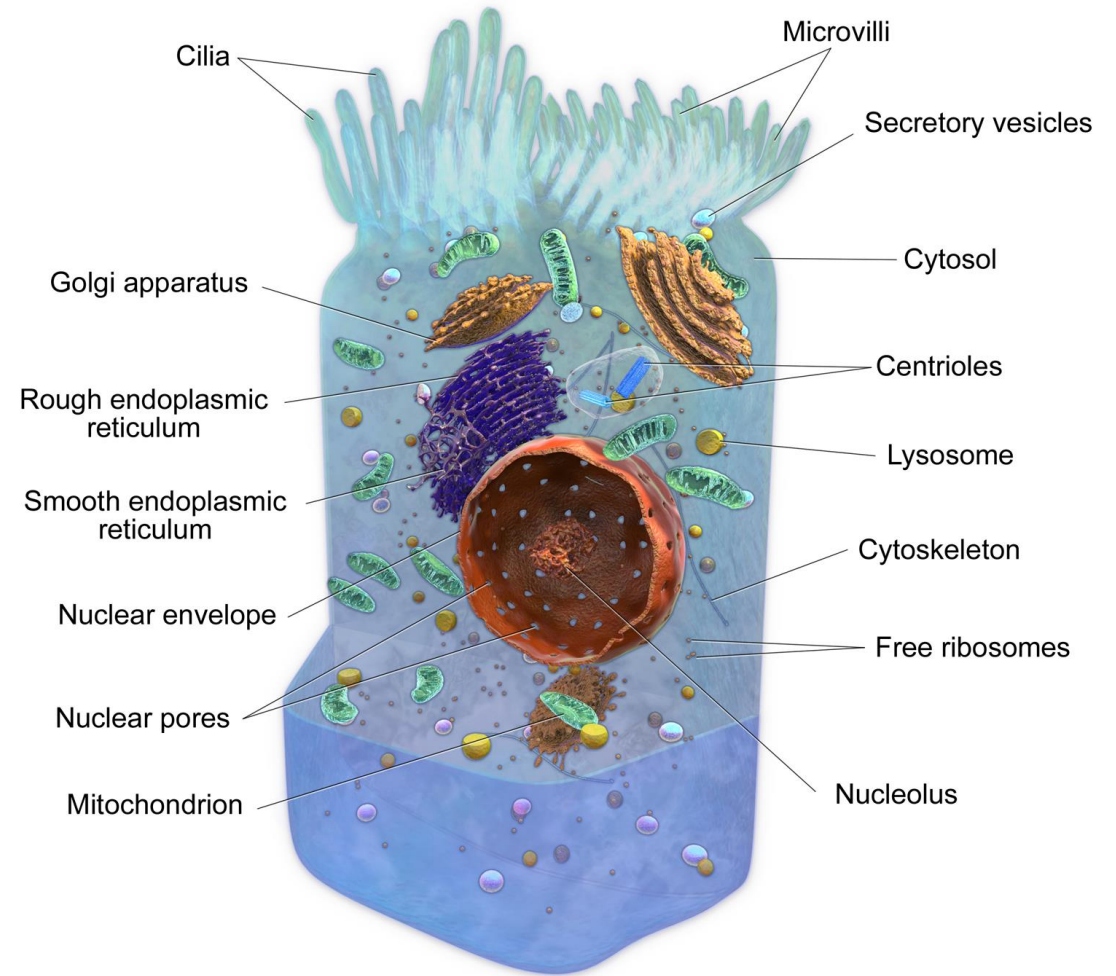
- Modern biological methods generate enormous amounts of data
- Need some way to analyze it

Biology Primer

- Cells
- Proteins
- Amino Acids
- Codons
- DNA

Cells

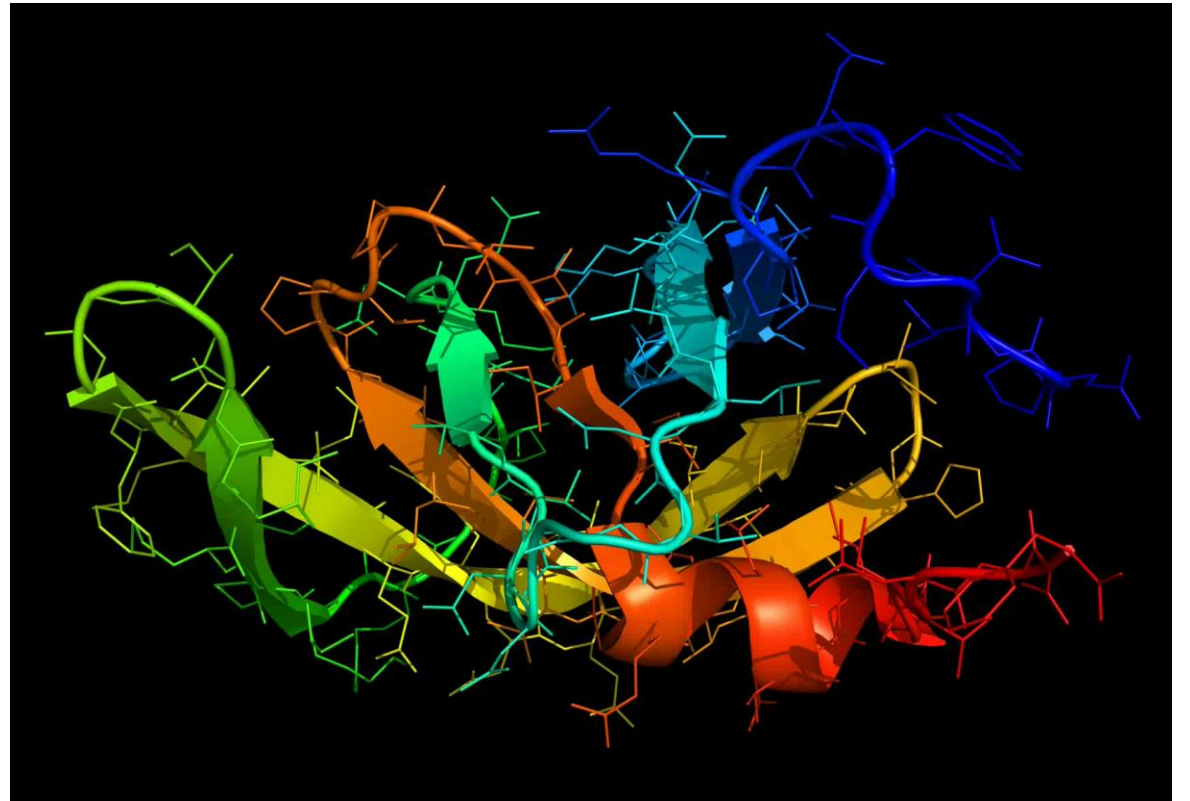
- Made of proteins
- Nucleolus contains DNA



Anatomy of a Cell

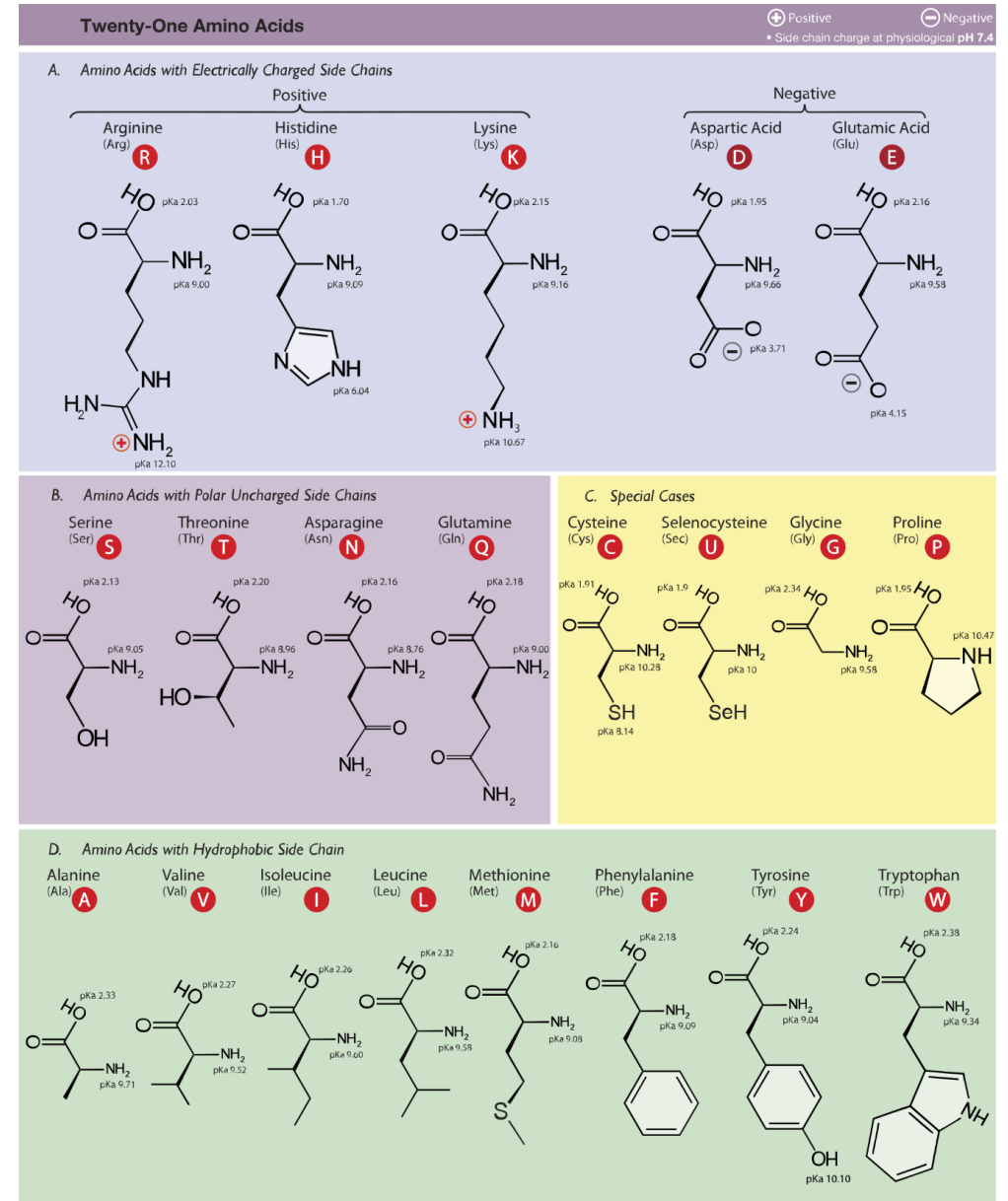
Proteins

- Chains of amino acids
- Fold into complex structures
- Alpha helices
- Beta sheets



Amino Acids

- 20 occur naturally
- Encoded by codons
- Can be treated as single letters



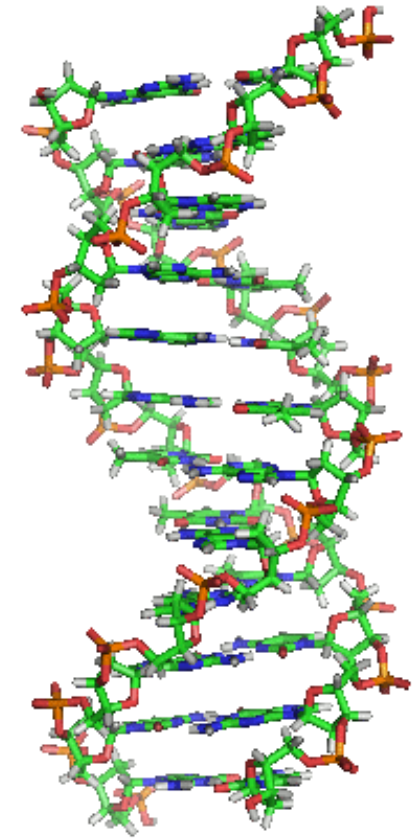
Codons

- Sequence of 3 base pairs
- Multiple codons exist for each amino acid
- TGT -> C
- GCC -> A

1st base	2nd base								3rd base
	T		C		A		G		
T	TTT	(Phe/F) Phenylalanine	TCT	(Ser/S) Serine	TAT	(Tyr/Y) Tyrosine	TGT	(Cys/C) Cysteine	T
	TTC		TCC		TAC		TGC		C
	TTA		TCA		TAA ^[B]	Stop (Ochre)	TGA ^[B]	Stop (Opal)	A
	TTG		TCG		TAG ^[B]	Stop (Amber)	TGG	(Trp/W) Tryptophan	G
C	CTT	(Leu/L) Leucine	CCT	(Pro/P) Proline	CAT	(His/H) Histidine	CGT	(Arg/R) Arginine	T
	CTC		CCC		CAC		CGC		C
	CTA		CCA		CAA	(Gln/Q) Glutamine	CGA		A
	CTG		CCG		CAG		CGG		G
A	ATT	(Ile/I) Isoleucine	ACT	(Thr/T) Threonine	AAT	(Asn/N) Asparagine	AGT	(Ser/S) Serine	T
	ATC		ACC		AAC		AGC		C
	ATA		ACA		AAA	(Lys/K) Lysine	AGA	(Arg/R) Arginine	A
	ATG ^[A]	(Met/M) Methionine	ACG		AAG		AGG		G
G	GTT	(Val/V) Valine	GCT	(Ala/A) Alanine	GAT	(Asp/D) Aspartic acid	GGT	(Gly/G) Glycine	T
	GTC		GCC		GAC		GGC		C
	GTA		GCA		GAA	(Glu/E) Glutamic acid	GGA		A
	GTG		GCG		GAG		GGG		G

DNA

- Identified in 1869
 - James Watson
 - Francis Crick
- Double helix
- Genetic instructions
- Made up of 4 bases
 - Adenine (A)
 - Cytosine (C)
 - Guanine (G)
 - Thymine (T)



String Matching

String Matching

- Given two sequences of amino acids
- Match: GADDLEEM
- To: GADDTVLEEM

String Matching

Match: GADDLEEM
To: GADDTVLEEM

- 4 matches
 - GADDLEEM
 - ||||
 - GADDTVLEEM
- 4 matches
 - GADDLEEM
 - ||||
 - GADDTVLEEM

- 2 matches
 - GADDLEEM
 - | |
 - GADDTVLEEM

String Matching

Match: GADDLEEM
To: GADDTVLEEM

- 4 matches

- GADDLEEM
- ||||
- GADDTVLEEM

- 4 matches

- GADDLEEM
- ||||
- GADDTVLEEM

- 2 matches

- GADDLEEM
- | |
- GADDTVLEEM

- 8 matches!

- GADD--LEEM
- |||| | |||
- GADDTVLEEM

Smith Waterman

- Dynamic programming algorithm
- Local alignment
- Not unique
- Configurable
 - Reward for match
 - Penalty for mismatch
 - Penalty for gap
- Slow

Smith Waterman

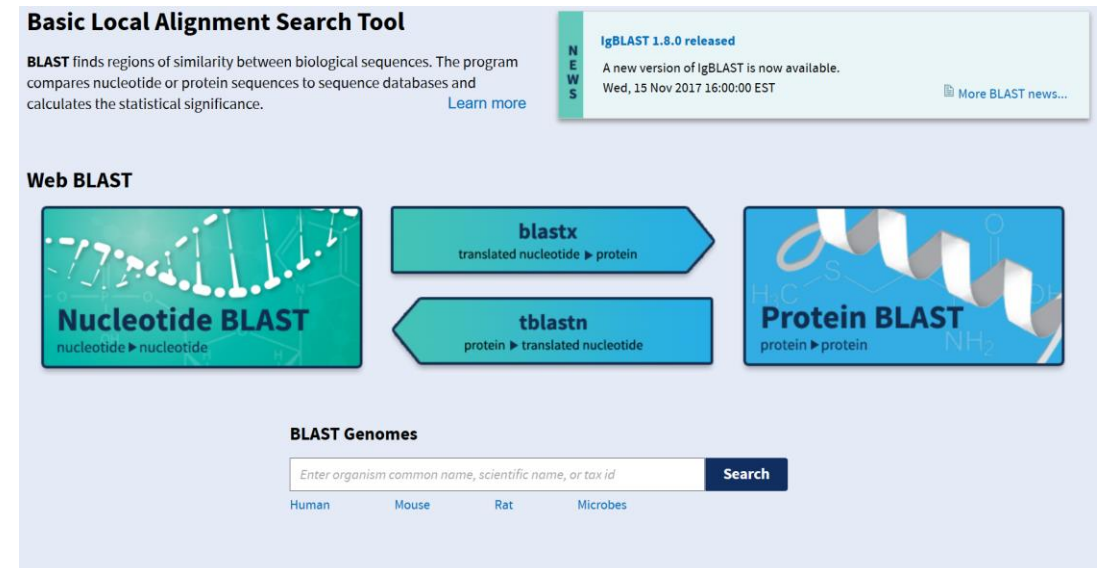
- Match reward: 3
- Mismatch penalty: -3
- Gap penalty: -2

		G	A	D	D	L	E	E	M
		0	0	0	0	0	0	0	0
G		0	3	1	0	0	0	0	0
A		0	1	6	4	2	0	0	0
D		0	0	4	9	7	5	3	1
D		0	0	2	7	12	10	8	6
T		0	0	0	5	10	9	7	5
V		0	0	0	3	8	7	6	4
L		0	0	0	1	6	11	9	7
E		0	0	0	0	4	9	14	12
E		0	0	0	0	2	7	12	17
M		0	0	0	0	0	5	10	15

- Generates alignment GADD - - LEEM

BLAST

- Basic Local Search Alignment Tool
- Fast
- Compromises accuracy for speed
- Much more complicated
 - Won't be going over an example
 - Can invoke as a “black box”
- Web based and local application



The screenshot displays the NCBI BLAST web interface. At the top, the title "Basic Local Alignment Search Tool" is followed by a brief description of BLAST's function and a "Learn more" link. A "NEWS" sidebar on the right highlights the release of "IgBLAST 1.8.0" with a date and a link to "More BLAST news...". The "Web BLAST" section features three main search options: "Nucleotide BLAST" (nucleotide to nucleotide), "blastx" (translated nucleotide to protein), and "tblastn" (protein to translated nucleotide). To the right of these is the "Protein BLAST" option (protein to protein). At the bottom, the "BLAST Genomes" section includes a search input field with placeholder text "Enter organism common name, scientific name, or tax id", a "Search" button, and a list of organism categories: Human, Mouse, Rat, and Microbes.

Levenshtein distance

- Measure of edit distance
 - Number of edits to turn one sequence into the other
- Dynamic programming algorithm
- Example: Kitten -> Sitting
 - Distance of 3
 - Kitten
 - **S**itten
 - Sitt**i**n
 - Sitt**i**ng

		k	i	t	t	e	n
	0	1	2	3	4	5	6
s	1	<u>1</u>	2	3	4	5	6
i	2	2	<u>1</u>	2	3	4	5
t	3	3	2	<u>1</u>	2	3	4
t	4	4	3	2	<u>1</u>	2	3
i	5	5	4	3	2	<u>2</u>	3
n	6	6	5	4	3	3	<u>2</u>
g	7	7	6	5	4	4	<u>3</u>

RCSB PDB

- Database of 3D structures of molecules
 - Positions experimentally determined
- 135,519 structures
 - Protein
 - DNA
 - RNA



Previous Work

Prior implementation

- Created by two of Professor Jagodzinski's former students
 - Sasa Vukovic
 - Dane Lindell
- Implemented in Java
- Uses BLAST
- Uses Levenshtein distance



Prior algorithm

- Sort user entered sequences
- Score “first” sequence using BLAST
 - $\text{Scores} += \text{bitscore} * \text{weight}$
- Score remaining sequences
 - $\text{Scores} += (\text{seq length} - \text{minLevDistance}) * \text{weight}$
- Sort based on scores
- Give results

Prior algorithm—pros

- Configurable by command line arguments
- Uses BLAST to find alignments
- Attempts to implement Bitap algorithm
 - More complex, faster, Levenshtein distance

Prior algorithm—cons

- Dependent on “first” sequence
- BLAST “bitscore” is arbitrary
- No special handling for “short” sequences
- Output into “magic” file



My Work

Where to search?

- Entire PDB database
- 135519 structures
 - Amino acid sequences



Program inputs

```
usage: blast.py [-h] [-s SEQ [SEQ ...]] [--fasta-file FASTA_FILE] [-o OUT]
               [pdbid] [subsequences [subsequences ...]]

positional arguments:
  pdbid                pdbid_chain Protein and chain for constrained
                        alignment
  subsequences          X:X:X Specify the subsequence and weight for the
                        constrained alignment

optional arguments:
  -h, --help            show this help message and exit
  -s SEQ [SEQ ...], --seq SEQ [SEQ ...]
                        Specify a sequence string and weight (sequence:weight)
                        to search for.
  --fasta-file FASTA_FILE
                        Specify file that specifies all the sequences in the
                        PDB in FASTA format
  -o OUT, --out OUT     Specify output filename.
```

Program inputs

- Sequence of Amino Acids
- String or index
- Weights
- Example:
 - -s GADDTVLEEM:10 KVRQYDQILIE:20
 - 1hhp 26:36:10 54:65:20

Program inputs

- String or index
- Weights
- Example:
 - -s GADDTVLEEM:10 KVRQYDQILIE:20
 - 1hhp 26:36:10 54:65:20

Weight = 10

Weight = 20

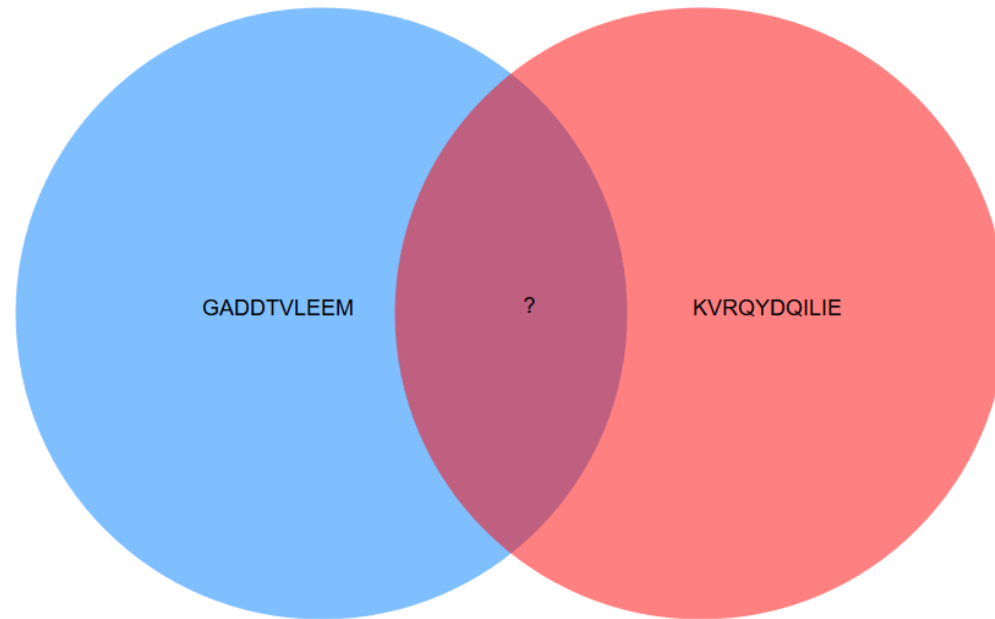
PQITLWQRPLVTIKIGGQLKEALLDT**GADDTVLEEM**SLPGRWKPKMIGGIGGGFI**KVRQYDQILIE**ICGHKAIGTVLVGPTPVNIIG...

Algorithm design

- BLAST
 - Is fast
 - Produces good alignments

Algorithm design

- BLAST
 - Is fast
 - Produces good alignments
- Use BLAST more!



Algorithm

- Find all matches for queries from blast
- Score using algorithm
- Return sorted and formatted results

Algorithm—Example

GADDTVLEEM:10 KVRQYDQILIE:20

Algorithm—Example

- Run BLAST on GADDTVLEEM

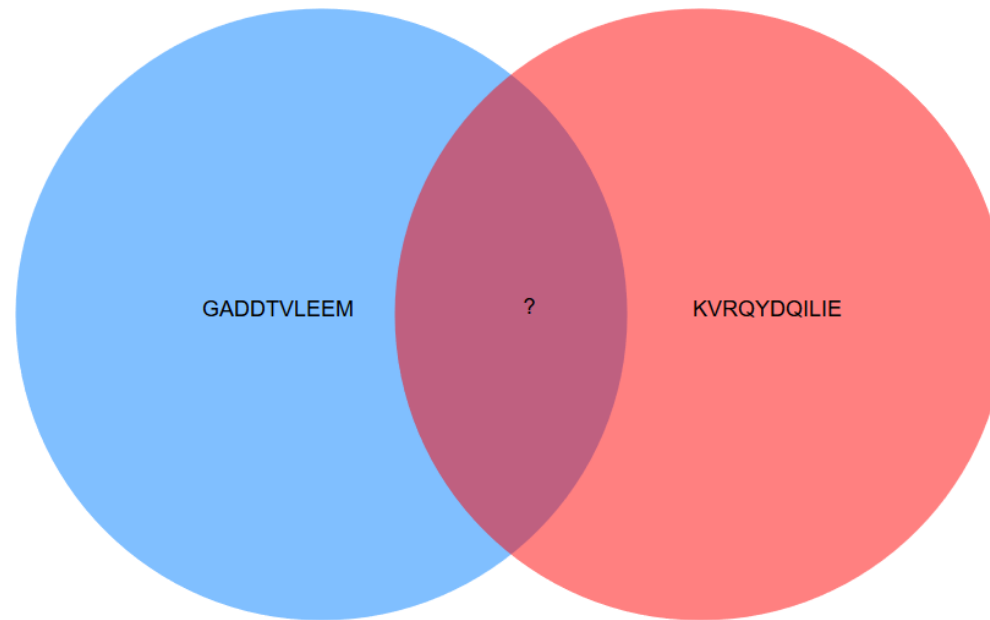
- 2hc0_B
- 2hc0_A
- 2hb2_A
- 2azc_B
- 2azc_A
- 4obk_B
- 4obk_A
- 4obj_B
- 4obj_A
- 4obh_D
- ...

- Run BLAST on KVRQYDQILIE

- 3mim_B
- 4q5m_A
- 4q1h_A
- 3kt5_A
- 3dox_A
- 2whh_A
- 1lv1_A
- 5v4y_A
- 5kao_B
- 5kao_A
- ...

Algorithm—Example

- Gather proteins from both BLAST runs



Algorithm—Example

- Score using Levenshtein distance
- For each sequence
 - `Score += weight * (seq length - edit distance)`

Algorithm—Example

- 2hc0_b: 300
 - PQITLWKRPLVTIKIGGQLKEALIDT**GADDTVLEEM**NLPGRWKPkiGGIGGLI**KVRQYDQI****P**IEICGHKAIGT
VLIGPTPANIIGRNLLTQIGCTLNF
- 3mim_b: 320
 - PQVTLWQRPLVTIKIGGQLKEALLDT**GADDTVLEEM**SLPGRWKPkmIGGIGGGFI**KVRQYDQIL**IEICGHKAI
GTVLVGPTPVNIIGRNLLTQIGATLNFGGSSG
- 2hb2_a: 300
 - PQITLWKRPLVTIKIGGQLKEALIDT**GADDTVLEEM**NLPGRWKPkiGGIGGLI**KVRQYDQI****P**IEICGHKAIGT
VLIGPTPANIIGRNLLTQIGCTLNF
- 4q5m_a: 310
 - PQVTLWQRPLVTIKIGGQLKEALLDT**GAD****N****T****VLEEM**SLPGRWKPkmIGGIGGGFI**KVRQYDQIL**IEICGHKAI
GTVLVGPTPVNIIGRNLLTQIGMTLNFGGSSGPQVTLWQRPLVTIKIGGQLKEALLDTGADNTVLEEMSLP
GRWKPkmIGGIGGGFIKVRQYDQILIEICGHKAIGTGLVGPTPVNIIGRNLLTQIGATLNF
- ...

Algorithm—Example

- Sort based on score

Algorithm—Results

```
2bbb_b | Score: 320, Gaps: [18]
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNF
-----aaaaaaaa-----bbbbbbbbbb-----

1gnm_a | Score: 320, Gaps: [18]
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPDNIIGRNLLTQIGCTLNF
-----aaaaaaaa-----bbbbbbbbbb-----

1iiq_a | Score: 320, Gaps: [18]
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNF
-----aaaaaaaa-----bbbbbbbbbb-----

2r3w_a | Score: 320, Gaps: [18]
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPVNVIGRNLLTQIGCTLNF
-----aaaaaaaa-----bbbbbbbbbb-----

1z1f_b | Score: 320, Gaps: [18]
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKVIGTVLVGPTPTNVIGRNLLTQIGCTLNF
-----aaaaaaaa-----bbbbbbbbbb-----

2uy0_b | Score: 320, Gaps: [18]
PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNF
-----aaaaaaaa-----bbbbbbbbbb-----
```


Demo

Future Work

Future work

- Better indication of alignments
- Implement Smith-Waterman algorithm to improve alignments
- Continue improving performance
- More options for formatting output
- Additional metrics to improve output
- [Website](#) for improved usability and ease of access

Acknowledgements

Acknowledgements

- Filip Jagodzinski
 - Advisor
- Logan Bridges
 - Research partner

References

References

- <https://commons.wikimedia.org/wiki/File:Myoglobin.png>
- <http://www.nust.edu.pk/INSTITUTIONS/Centers/RCMS/ap/pg/MSBioinformatics/Pages/default.aspx>
- [https://commons.wikimedia.org/wiki/File:DNA orbit animated static thumbnail.png](https://commons.wikimedia.org/wiki/File:DNA_orbit_animated_static_thumbnail.png)
- https://en.wikipedia.org/wiki/DNA_codon_table
- [https://commons.wikimedia.org/wiki/File:Amino Acids.svg](https://commons.wikimedia.org/wiki/File:Amino_Acids.svg)
- [https://commons.wikimedia.org/wiki/File:Blausen 0208 CellAnatomy.png](https://commons.wikimedia.org/wiki/File:Blausen_0208_CellAnatomy.png)
- <https://blast.ncbi.nlm.nih.gov/Blast.cgi>
- <https://www.rcsb.org/pdb/home/home.do>
- <https://www.draw.io/>