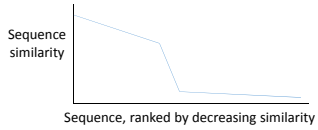
MCB 5472 Lecture #4:
Probabilistic models of homology:
Psi-BLAST and HMMs
February 17, 2014

From last week:

- BLASTp searches find homologs to a single sequence in a sequence database
  - Highest score to sequences best matching the query
  - Corollary: lower scores to distant sequences still matching the query

Finding all homologous sequences using BLASTp

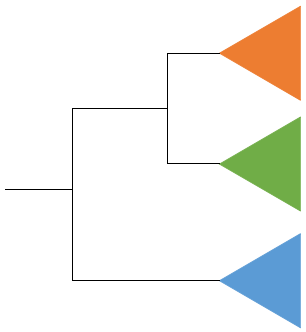
- In an ideal world (e.g., highly conserved sequences) a simple BLASTp search would recover all homologs of a single query sequence
  - Simply accept all sequences above some E-value cutoff
  - E-values can have a steep decline



Nice but…

- Assumes the query sequence perfectly represents all homologs
- False because:
  - Substitutions may be from lineage-specific biases and not conserved in homologs more generally, biasing search towards closer relatives
  - Conservation is always incomplete: the query may not contain conserved positions present in most other homologs
- Sequences close to a hard E-value cutoff can be easily excluded/included depending on search bias

Shown another way



Sensitivity vs specificity

- Trade off between minimizing false-negative detection (sensitivity) and false-positives (specificity)
  - A common trade-off in bioinformatics
- BLASTp is designed to maximize sensitivity
  - Specificity can be low – left to the user to cull out the false-positives

Net result:

- Divergent homologs are hard to detect
- Because they are close to typical E-value cutoffs, any bias can easily lead to them being excluded
- Discriminating between false- and true-positives can be problematic
  - Requires manual examination
- Finding deep homology is hard!

Better: use multiple queries representing all homologs

- Option 1: Run multiple individual BLASTps
  - Still easy to bias – "unknown unknowns"
- Option 2: Make a statistical model of sequence conservation amongst all homologs and use that to find different relatives
  - Diverse input sequences removes and averages out lineage-specific biases

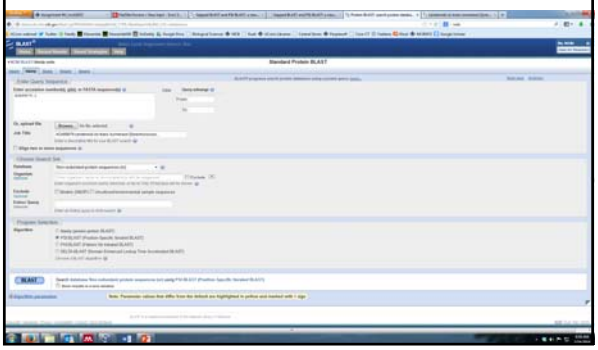
PSI-BLAST

- "Position-Specific Iterated BLAST"
- Works only for proteins
- Uses BLASTp to create a "position-specific score matrix" (PSSM)
  - Smith Waterman global alignments are an option for the command line but not the web (slower, more accurate)
- Uses matrix for subsequent database searches
- Matrix updated on each iteration
  - Bias reduced each time
  - Sensitivity increased towards distant homologs
  - False-positives reduced by model refinement

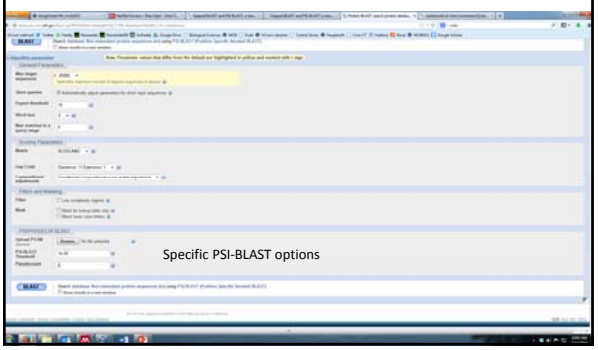
PSI-BLAST: step #1

- First iteration: standard BLASTp using a single sequence
- All homologs above a specified E-value threshold kept to make PSSM
  - Can be specified via parameters, manually edited on NCBI website implimentation

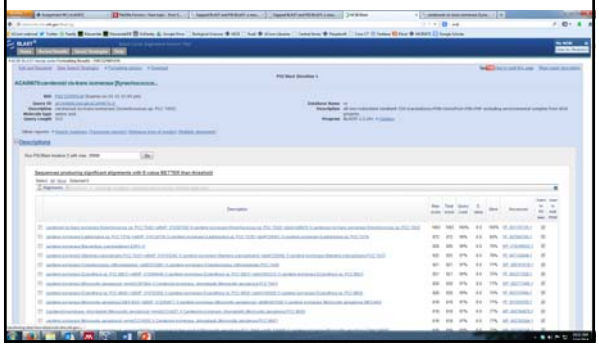
NCBI web implimentation



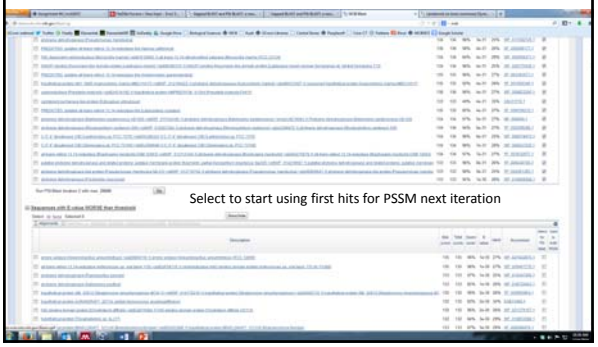
NCBI web implementation



NCBI web implementation



NCBI web implementation



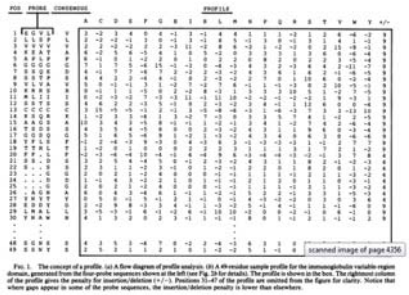
PSI-BLAST: step #2

- Makes (rough) multiple sequence alignment for the selected BLASTp results
- All hits aligned to the query
  - Not a true multiple sequence alignment
  - Possible to input an externally generated alignment via terminal version (but not web)
- Alternative at terminal: Smith Waterman global pairwise alignments
  - Not available for web
  - Slower but more accurate

PSI-BLAST: step #3

- Use sequence alignment to create Position-Specific Scoring Matrix (PSSM)
- PSSM:
  - Unique substitution matrix for each sequence alignment column
  - Extra column for gap penalty
    - Matrix is 21 x [query length] vs. 20 x 20 for normal matrix
  - Scores merge standard distance matrix with position-specific frequencies from 1st iteration, weighted by sequence similarity

Example PSSM



Gribskov et al. (1987) PNAS 84: 4355–4358

PSI-BLAST: step #4

- Query reference database using the PSSM
  - Recall: BLASTp looks for 2-3 amino acid words similar to the query sequence above some threshold score calculated from the distance matrix
  - An equivalent calculation can be performed using the PSSM; find possible words having a score > the same threshold
  - Subsequent BLAST steps are the same: extend matching words, recalculate with gaps, calculate statistics
  - E-values now reflect similarity to the query profile, not any individual sequence

PSI-BLAST: step #5

- Perform as many iterations as you like
- PSSM updated each time based on hits passing E-value threshold on the previous iteration
- Sequence-specific bias reduced each time as the PSSM is adjusted to reflect homolog in the entire input set

CrtR protein 1e-50 threshold

Iteration	Hits > 1e-50	Notes
1	151	
2	215	
3	258	Query not top hit, top E-value != 0.0
4	271	
5	271	
6	271	

Model corruption

- If a non-homologous sequence is included during model construction, can bias the model away from true homologs
- With subsequent iteration, model can be made completely useless
- Using a higher E-value cutoff can ameliorate
- On web can examine results and limit selection
  - Can't do this in high-throughput at terminal

Command line PSI-BLAST

- Part of BLAST+ package so same basic parameters apply
- Additional flags:

```
-num_iterations [number]
-out_pssm [filename]
-out_ascii_pssm [filename]
-comp_based_stats 0 # required
```

e.g., `[jlklassen@bbcsrv3 ~]$ psiblast –query test.faa –db all.faa –num_iterations 3 –out test_vs_all.psiblast –out_ascii_pssm pssm.out –comp_based_stats 0`

Starting PSI-BLAST with pre-computed PSSM

- Create PSSM using PSI-BLAST with `-out_pssm` flag (not `-out_ascii_psm`)
- Use `–in_pssm` flag instead of `–query`
- e.g., `[jlklassen@bbcsrv3 ~]$ psiblast –in_pssm pssm.out –db all.faa –num_iterations 3 –out test_vs_all.psiblast –comp_based_stats 0`

RPSBLAST

- PSI-BLAST queries a sequence database with an individual PSSM
- RPSBLAST does the opposite: queries an individual sequence with a database of PSSMs
  - e.g., From NCBI's Conserved Domain Database (CDD) to annotate sequences according to NCBI's ortholog family descriptsion
- In BLAST+, command is `rpsplast+` and works similarly to other BLAST+ commands except `–db` is now PSSMs, not sequences
- Possible to make your own PSSM database but complicated (most people use HMMer instead)

Beyond BLAST

- Recall that all BLAST programs are local alignments
  - Trade-off between speed and accuracy
- FASTA: alternative package for database
  - http://www.ebi.ac.uk/Tools/sss/fasta/
  - Heuristics like BLAST using word matching for initial sequence matching
  - Final alignments use Smith-Waterman global pairwise alignment method
- Advances in computer science and statistics on both fronts (i.e., better accuracy & better approximations)

HMMER

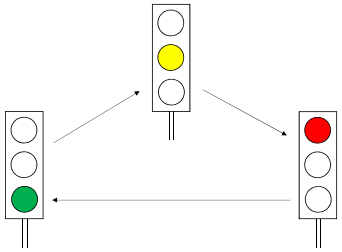
- HMMER is a software package similar to PSI-BLAST, i.e., searching databases with homology models
- Uses HMMs instead of PSSMs
- Advantages:
  - More statistically explicit models
  - HMMER3 as ~fast as BLAST
  - Easy to use at command line
  - Can make models for DNA, RNA, protein
- Disadvantages:
  - Initial alignment is always a second step
  - No NCBI interface (database-specific instead)

The purpose of HMMs

- To evaluate the probability of a sequence matching a model
  - Assumes preexisting model
- Essentially a classification problem
  - Given data, how well does it fit a model?
  - Given data and multiple models, which fits best?
  - e.g., Does a gene belong to a gene family?

Markov Chains

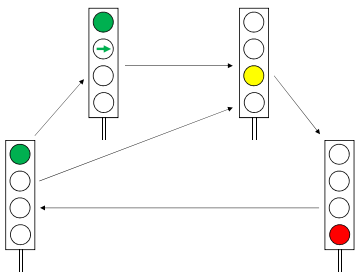
- "A finite number of states connected by transitions"



http://www.ch.embnet.org/CoursEMBnet/Basel03/slides/PSSM_HMMM.pdf

Markov Chains

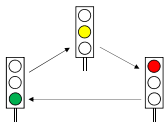
- "A finite number of states connected by transitions"



http://www.ch.embnet.org/CoursEMBnet/Basel03/slides/PSSM_HMMM.pdf

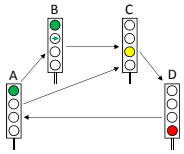
Transition probabilities

- The probability of moving from one state to another
- P(Yellow|Green) = 1
  - "The probability of transitioning to yellow given green"
- P(Red|Yellow) = 1
- P(Green|Red) = 1
- P(Green|Yellow) = 0
- P(Yellow|Red) = 0
- P(Red|Green) = 0



Transmission probabilities

- The probability of moving from one state to another
- P(C|B) = 1
- P(D|C) = 1
- P(A|D) = 1
- P(A|B) = P(vehicles turning left)
- P(A|B) = P(no vehicles turning left)
- P(B|A) = 0 etc.



Hidden Markov Models

- HMMs are like Markov Chains in that they comprise states connected by transitions
- Difference: each state does not comprise a single symbol but rather a distribution of them
  - e.g., a column of a sequence alignment will contain some frequency of A, C, G and T
- Each state can "emit" a symbol with some probability

Hidden Markov Models

- Known:
  - The number of states
  - The transition probabilities
  - The emission probabilities
- Question: how well does a sequence match the model?
  - Evaluate the global probability by multiplying the probability of each step through the graph

A simplified model: identifying a 5' splice site

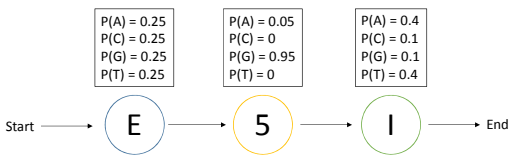
Eddy 2004 Nat. Biotechnol. 22: 1315-1316

Three possible states: exon, 5' splice site, intron

CTTCATGTGAAAGCAGACGTAAGTCA

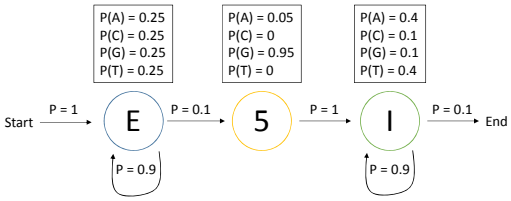
Emission probabilities

- 5' splice site nearly always G, occasionally A
- Exon sequence distributed uniformly
- Intron sequence is AT rich

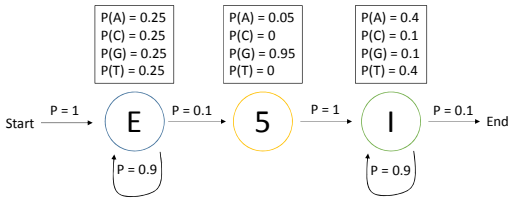


Transition probabilities

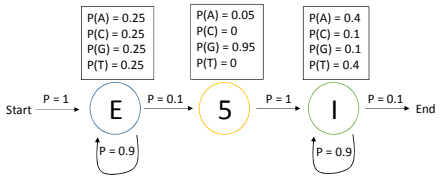
- Exon & intron can be multiple bases long
- 5' splice site only one base long
- Exact probabilities are flexible



Discuss: given this sequence, where is the splice site?

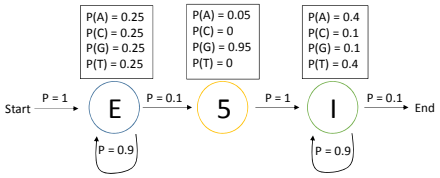


CTTCATGTGAAAGCAGACGTAAGTCA

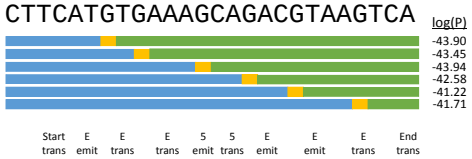


CTTCATGTGAAAGCAGACGTAAGTCA

- 14 possibilities (number of "A"s and "G"s)
- "A"s are sufficiently improbable that we will not work them out here



CTTCATGTGAAAGCAGACGTAAGTCA



$\log(P(1^{st})) = \log(1*(0.25^6*0.9^5)*(0.1)*0.95*1*[(0.4^{11})*(0.1^8)*(0.9^{18})]*0.1) = -43.90$

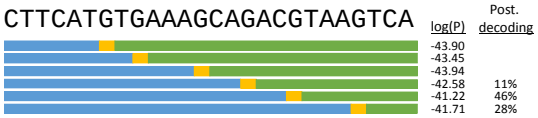
HMMs

- Given a model and input data, we can calculate the likelihood of any given classification
- Because model is fully parameterized, significance of each bath can be determined in a Baysian statistical framework
  - "Posterior decoding"

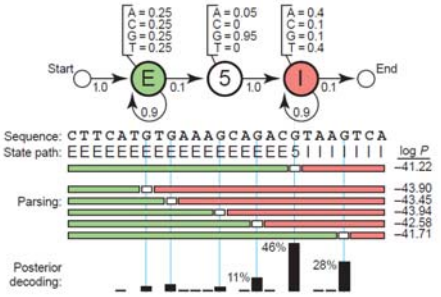
Posterior decoding

- Definition: probability of chosen path divided by sum of the probability of all other paths
- e.g., $-41.22/((-43.90) + (-43.45) + (-43.94) + \ldots)$

CTTCATGTGAAAGCAGACGTAAGTCA



Note: many non-zero posterior decodings



(From actual Nat. Biotechnol. paper)

Profile HMMs

- HMMs applied to multiple sequence alignments
- Each alignment column is a state
  - Emission probabilities based on sequence conservation in the alignment weighted by sequence similarity
- A separate series of states exists for gaps
  - Emission probabilities encompass gap extension
- Significance of sequence matching model calculated similarly to our much simpler splicing example

HMMER3

- Software package for making and using profile HMMs
- Like BLAST+, can use own models or download from others
- Approximately as fast as BLAST+ (previous versions were not)
- http://hmmer.janelia.org/

`hmmbuild` & `hmmpress`

- Input: multiple sequence alignment
- 2 steps: (i) make HMM; (ii) compress for database searching (like `makeblastdb`)

`hmmbuild <output file> <input MSA>`

e.g., `hmmbuild test.hmm test.muscle.faa`

`hmmpress <input HMM>`

e.g., `hmmpress test.hmm`

Creates: `test.hmm.h3m test.hmm.h3i test.hmm.h3f test.hmm.h3p`

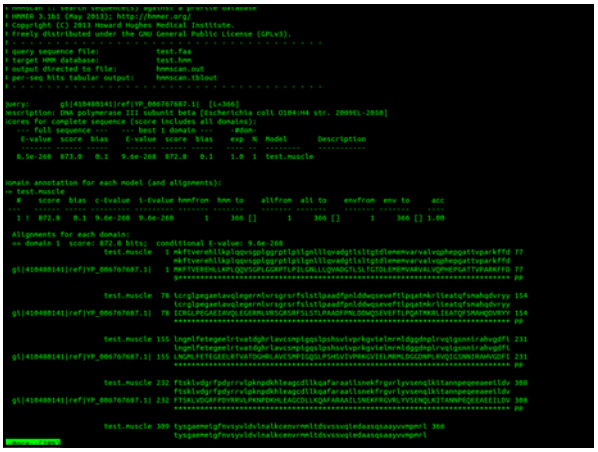
- Requires predefined input set
- Requires sequence alignment
- No simple iterative model updating option like PSI-BLAST
  - Have to rerun alignment and `hmmbuild` instead

`hmmscan`

- Actual search function (cf. `rpsblast`) comparing sequences to profiles

`hmmscan –o <output> <HMM name> <query seq>`

e.g., `hmmscan –o hmmscan.out test.hmm test.faa`



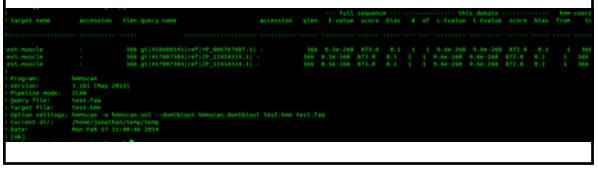
HMMER long output

- Two outputs: one for complete sequence, one for domains
- E-value and score analogous to BLAST output
- "bias": score adjustment based on compositional bias in the database
- Domains:
  - i-Evalue: likelihood of hit in entire database
  - c-Evalue: likelihood of hit in hits
  - Domain boundaries
  - Domain alignments

`hmmscan` domain table output format

`hmmscan –domtblout <domain table output> <HMM name> <query seq>`

e.g., `hmmscan –domtblout hmmscan.domtblout test.hmm test.faa`



Other useful HMMER functions (working similarly)

- `hmmsearch`: search models vs. sequences (cf. `PSI-BLAST`)
- `hmmalign`: align sequences to HMM
- `nhmmscan`: align nucleotide sequences to nucleotide HMM
  - `hmmbuild` autodetects input format or can be specified

Database sources

- Many different databases supply HMMs for various purposes
  - Pfam: protein domains in sequences
  - Rfam: RNA annotation in genomes
  - Interpro: integration of different orthology methods
    - Uses HMMs and simpler motif matching cf. regular expressions
- Each has its own website, not integrated like NCBI