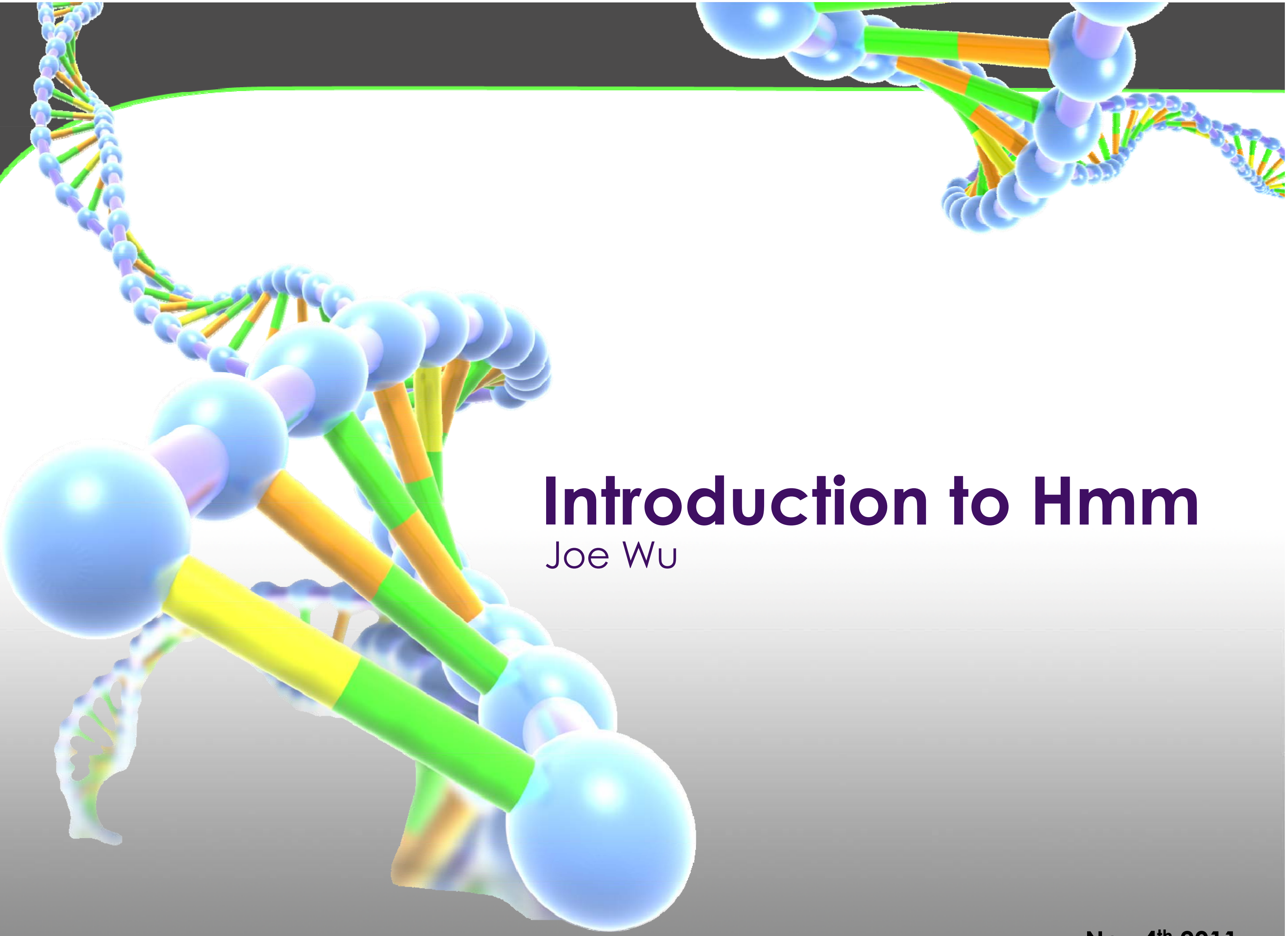


Introduction to Hmm

Joe Wu

Nov 4th 2011

•Agenda

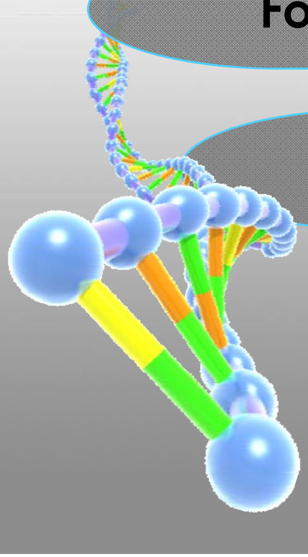
One — The applications of HMM.

Two — Standard Markov model (example: CG islands Discrimination)

Three — Hidden Markov model(example: CG islands Detection)

Four — Introduce Profile HMMs and PSSM

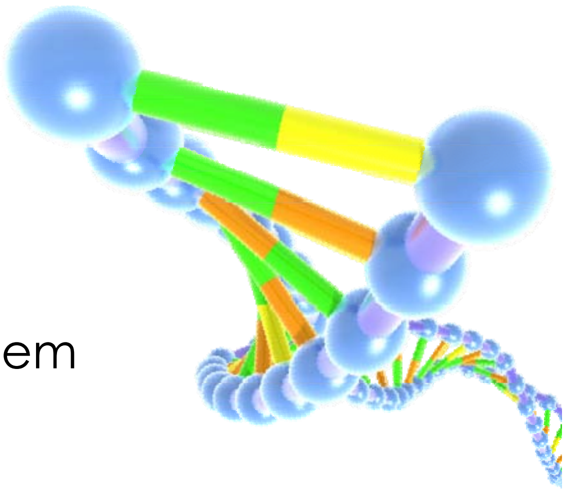
Five — Introduce Hmm databases and Hmmer3



•The applications of HMM

Speech Recognition

Phoneme



Iphone4s Siri :
a voice-controlled artificial intelligence system

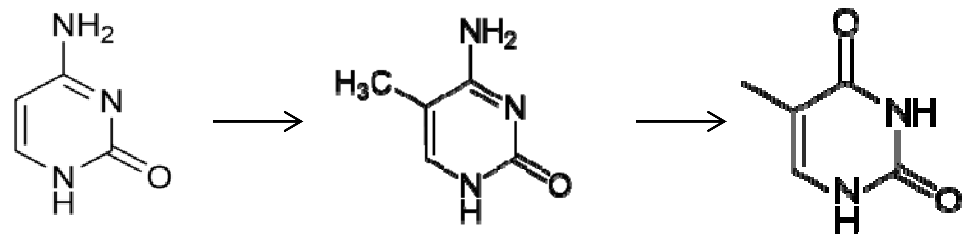
Android 4.0:
a real-time speech-to-text translation

Biological sequence searching and aligning

Nucleotides & AAs

Hmmer 3.0:
Used for searching sequence databases for homologs of protein sequences, and for making protein sequence alignments.

•CG island Example



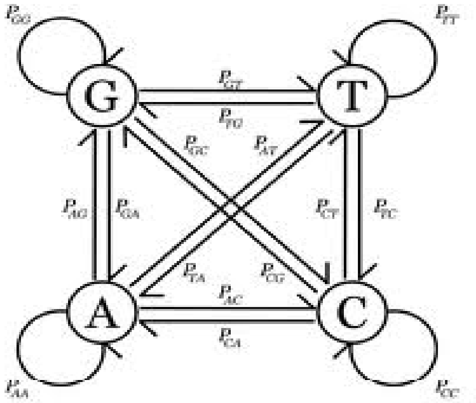
Cytosine (C) *5-Methylcytosine* *Thymine(T)*

In the human genome wherever the dinucleotide CG, the C is typically chemically modified by methylation. There is a relatively high chance of this methyl-C mutating into a T, with the consequence that in general CG dinucleotides are rarer in the genome. For biologically important reasons the methylation process is suppressed in short stretches of the genome, such as around the promoters or 'start' regions of many genes. In these regions we see many more CG dinucleotides than elsewhere, and in fact more C and G nucleotides in general. Such regions are called CG islands .

? Given a short stretch of genomic sequence,
how would we decide if it comes from a CG island or not?

? How do we find the CG islands in a long unannotated sequence?

•Standard Markov Model (introduction)



$$\underbrace{\text{ATCGCCGATGGTAATGCCTT}}_{\text{Length (L) = 20}}$$

- **States:**
  A, T ,C, G

- **Transition probability:**
  P_{AT} = The probability of A follow by T

- **Sequence probability:**

$$
\begin{aligned}
P(X) &= P(X_L, X_{L-1}, \ldots\ldots, X_1) \\
&= P(X_L|X_{L-1}, \ldots\ldots, X_1)\, P(X_{L-1}|X_{L-2}, \ldots\ldots, X_1) \ldots P(X_1) \\
&= P(X_L|X_{L-1})\, P(X_{L-1}|X_{L-2}) \ldots P(X_2|X_1) P(X_1)
\end{aligned}
$$

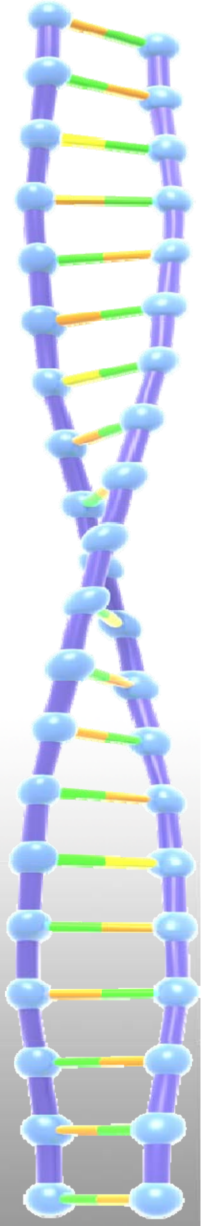
Bayes rule
$P(X,Y) = P(X|Y)P(Y)$

$P(X_1)$ = ???

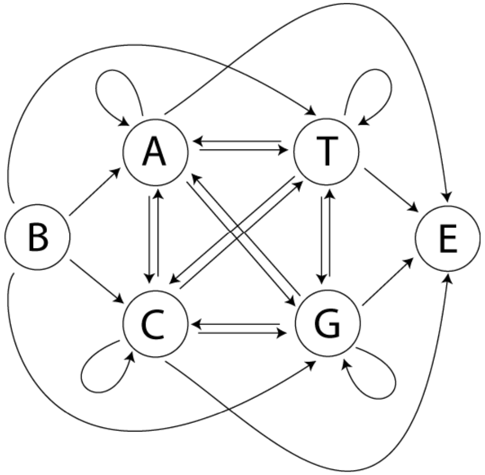
Markov property
$P(X_i|X_{i-1},\ldots X_1) = P(X_i|X_{i-1})$

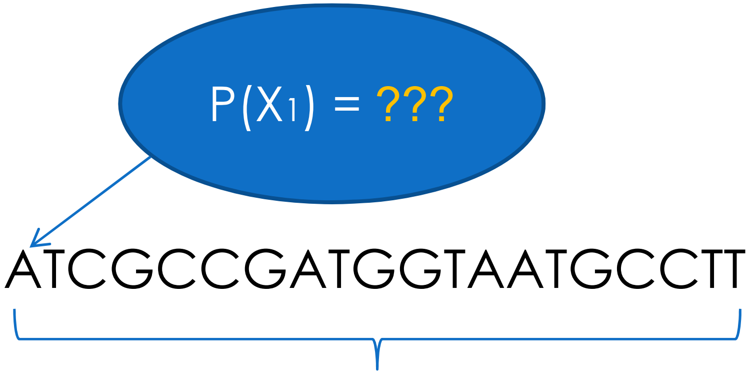
?

Exercise:
Based on the above markov chain, the sum of the probability of all possible sequences of length L is equal to 1.



•Standard Markov Model (with begin and end state)

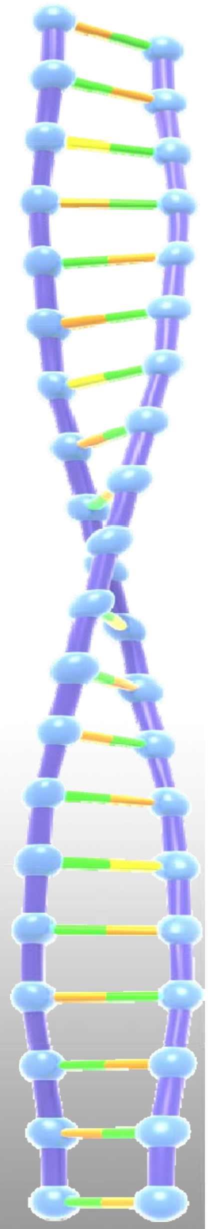




$P(X_1)$ = ???

ATCGCCGATGGTAATGCCTT

Length (L) = 20



- **States:**

  B,A,T,C,G,E

- **Transition probability:**

  $P(X_1 = A) = P_{BA}$: The probability the sequence begin with A.

  $P(E|X_L = T) = P_{TE}$: The probability the sequence end with T.

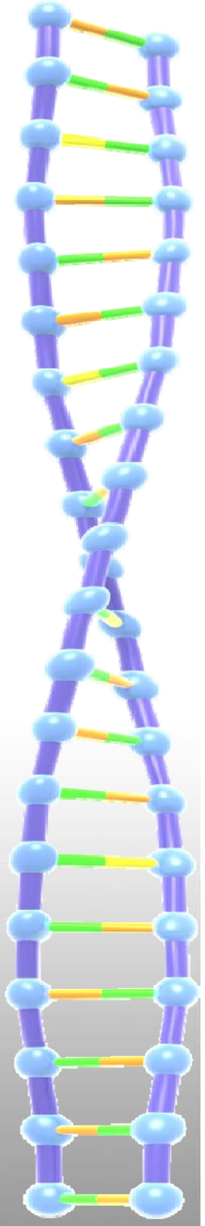
- **Sequence probability (with length L):**

$$P(X) = P(E|X_L)P(X_L|X_{L-1})\dots P(X_2|X_1)P(X_1)$$

Exercise:

Assume that the model has an end state, and that the transition from any state to the end state has probability **ε**. Show that the sum of the probability over all sequences of length L (and properly terminating by making a transition to the end state) is **$\varepsilon(1- \varepsilon)^{L-1}$**. Use this result to show that the sum of the probability over all possible sequences of any length is 1.

•Standard Markov Model (CG islands Discrimination)



Given a short stretch of genomic sequence,
how would we decide if it comes from a CG island or not?

From a set of human DNA sequences we extracted a total of 48 putative CG islands and derived two Markov chain models, one for the regions labelled as CG islands (the ”+” model) and the other from the remainder of the sequence (the ’-’ model).

Model +

+	A	C	G	T
A	0.180	0.274	0.426	0.120
C	0.171	0.368	**0.274**	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

P(x|Model +)

Model -

-	A	C	G	T
A	0.300	0.205	0.285	0.210
C	0.322	0.298	**0.078**	0.302
G	0.248	0.246	0.298	0.208
T	0.177	0.239	0.292	0.292

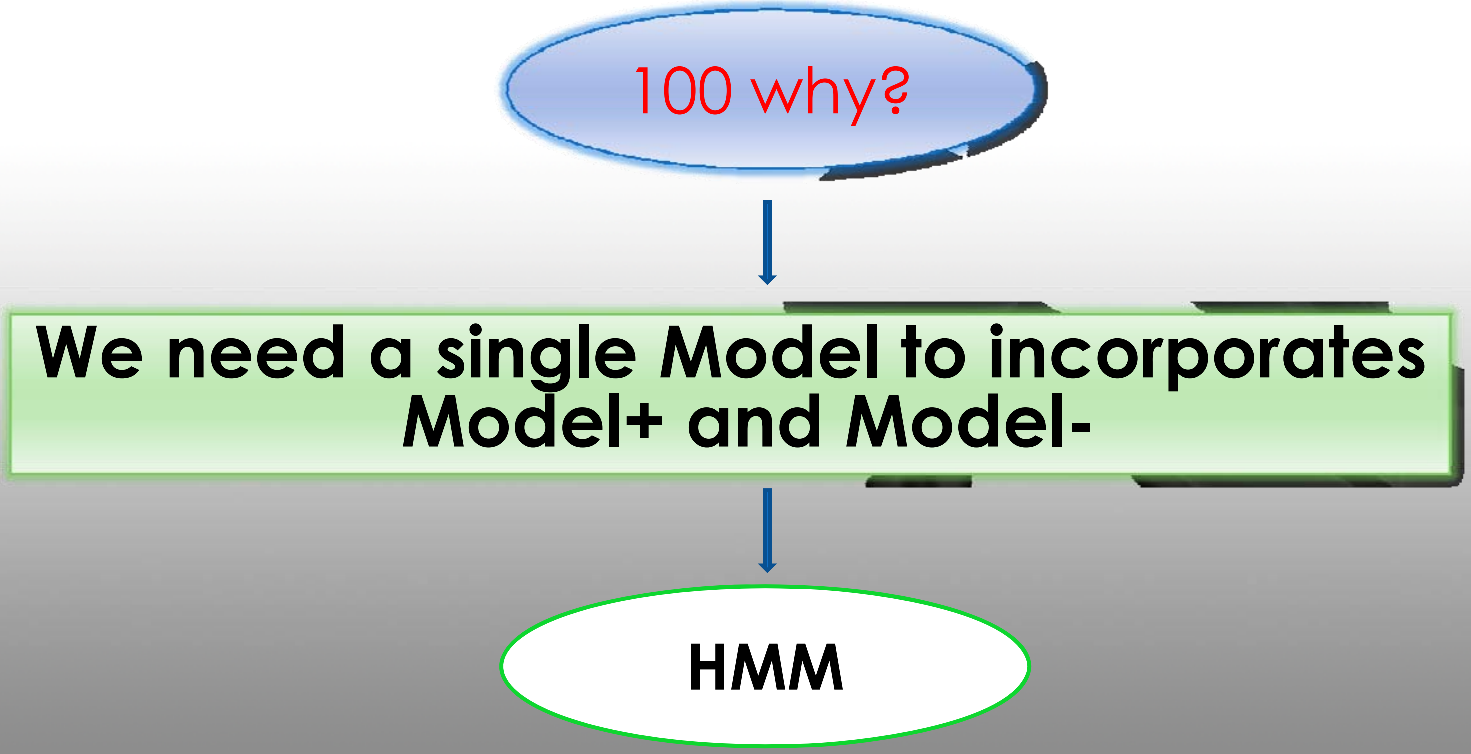
P(x|Model -)

Log-odds score: S(x) = log(P(x|Model +)/P(x|Model -))

•Hidden Markov Model (why HMM)

How do we find the CG islands in a long unannotated sequence?

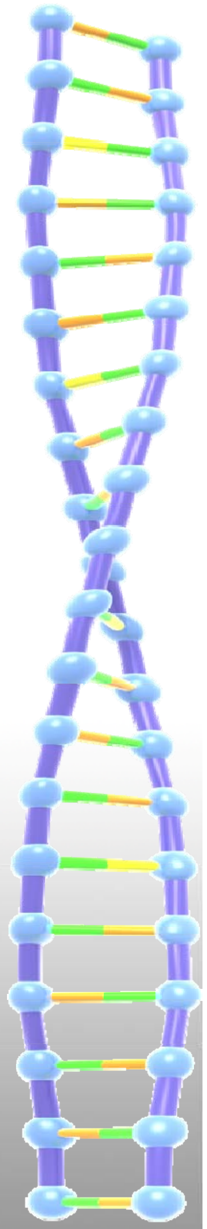
- We can use Standard Markov Model to calculate the log-odds score for a window of, say, 100 nucleotides around every nucleotide in the sequence and plotting it.



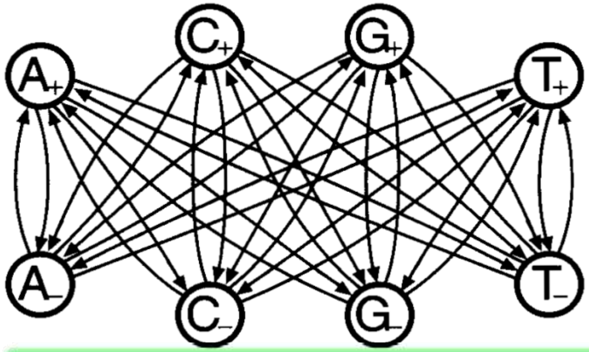
100 why?

We need a single Model to incorporates Model+ and Model-

HMM



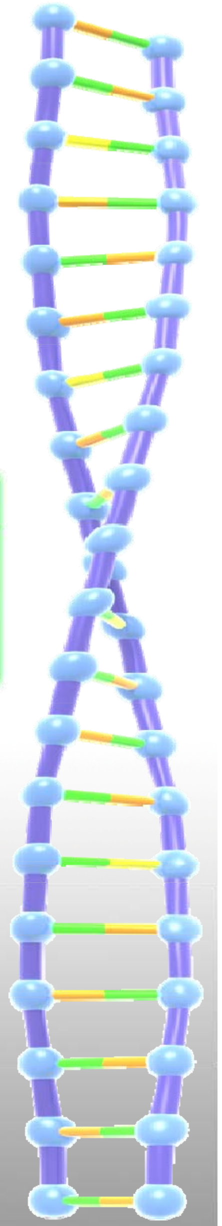
•Hidden Markov Model (introduction)



State path (π)	T-	C+	G+	C+	C-	…..
Sequence path (x)	T	C	G	C	C	…..

The essential difference between a Markov chain and a hidden Markov Model is that for a hidden Markov model there is not a one-to-one correspondence between the states and the symbols. (For example state C+ and C- both emit symbol C). Therefore we need to distinguish the sequence of states from the sequence of symbols

- **States:**
  A+,A-,T+,T-,C+,C-,G+,G-
- **Symbols**
  A,T,C,G
- **Transition probability**
  $a_{kl} = P(\pi_i = l \mid \pi_i = k)$
- **Emission probability**
  $e_k(b) = P(x_i = b \mid \pi_i = k)$
- **Sequence probability (with state path π):**
  $P(x, \pi) = a_{k\pi 1} \prod_{i=1 \mathrm{toL}} e_{\pi i}(x_i)\, a_{\pi i\ \pi i+1}$



•Hidden Markov Model (The most probable path)

Many state paths can generate the target sequence!!!

π1	T+	C+	G-	C+	C-	…..
π2	T-	C+	G+	C+	C+	…..
π3	T-	C-	G+	C-	C-	…..
x	**T**	**C**	**G**	**C**	**C**	**…..**

The most probable path : $\pi^* = \text{argmax } P(x, \pi)$

Viterbi Algorithm

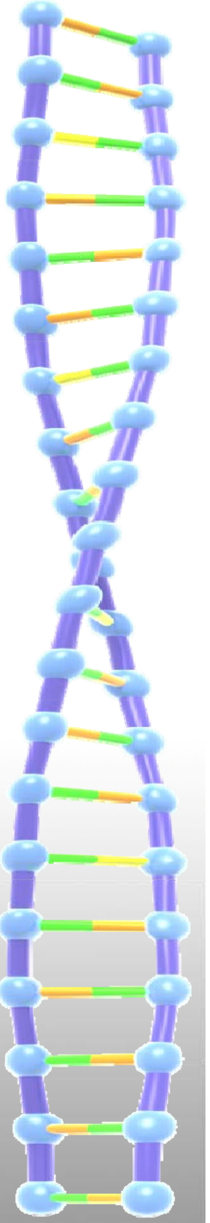
$$V_l(i+1) = e_l\,(x_{i+1})\max(V_k(i)a_{kl})$$

•$\mathbf{V_k(i)}$**:** The probability of most probable path up to x_i ending in state k.
•$\mathbf{V_l(i+1)}$ **:** The probability of most probable path up to x_{i+1} ending in state l.
•$\mathbf{a_{kl:}}$ Transition probability from state k to state l
•$\mathbf{e_l\ (x_{i+1})}$: Emission probability (x_{i+1} emits from state l)

Initialization: $i=0,\ V_0(0) = 1, V_k(0) = 0$

Termination: $P(x, \pi^*) = \text{Max}_k(V_k(L)\ a_{k0}\ ), \pi^* = \text{argmax}_\pi(V_k(L)\ a_{k0}\ ))$

Note: The start and end state both are 0



•Hidden Markov Model (the full sequence probability)

We must add the probabilities for all possible paths to obtain the full probability of x.

π1	T-	C+	G+	C+	C-	.....
π2	T+	C-	G+	C-	C-	.....
π3	T-	C-	G-+	C+	C-	.....
x	T	C	G	C	C	.....

The full probability of X : $P(x) = \sum_{\pi} P(x, \pi)$

Forward Algorithm

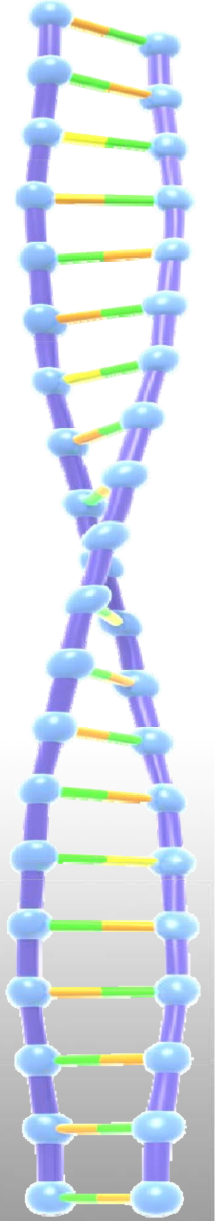
$$f_h(i+1) = e_h(x_{i+1}) \sum_k f_k(i)\, a_{kh}$$

•$f_k(i)$**:** The probability of the sequence up to x_i ending in state k. $P(x_{1\ldots}x_i, \pi_i=k)$
•$f_h(i+1)$ **:** The probability of the sequence up to x_{i+1} ending in state h.
•a_{kh} **:** Transition probability from state k to state h
•$e_h\ (x_{i+1})$ **:** Emission probability (x_{i+1} emits from state h)

Initialization: i=0, $f_0(0) = 1$, $f_k(0) = 0$

Termination: $P(x) = \sum_k f_h(i+1)\ a_{k0}$

Note: The start and end state both are 0



•Hidden Markov Model (the posterior state probabilities)

What if different paths have almost the same probability as the most probable one? We need posterior decoding

The posterior probability: $P(\pi_i = k | x)$

$P(\pi_i = k | x) = P(x, \pi_i = k) / P(x)$ $P(x)$: by forward algorithm

⬇

$P(x, \pi_i = k) = P(x_1 \ldots x_i, \pi_i = k) P(x_{i+1} \ldots x_L | x_1 \ldots x_i, \pi_i = k) = f_k(i) P(x_{i+1} \ldots x_L | \pi_i = k) = f_k(i)\, b_k(i)$

$f_k(i)$: by forward algorithm $b_k(i)$: by backward algorithm

backward Algorithm

Recursion: $b_k(i) = \sum_h a_{kh} e_h(x_{i+1})\, b_h(i+1)$

•$b_k(i)$: $P(x_{i+1} \ldots\ldots x_L | \pi_i = k)$

•$b_h(i+1)$: $P(x_{i+2} \ldots\ldots x_L | \pi_{i+1} = h)$

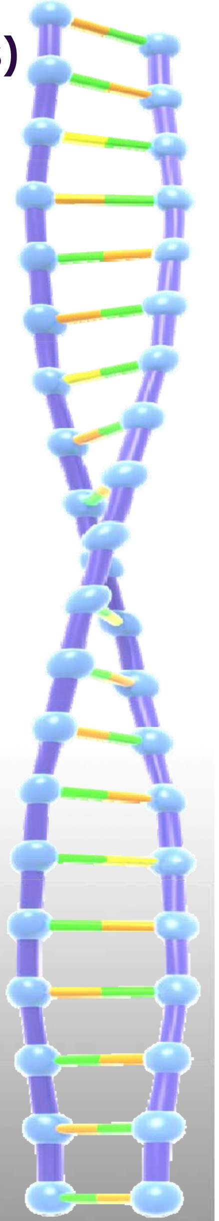
•$a_{kh:}$ Transition probability from state k to state h

•$e_h(x_{i+1})$: Emission probability (x_{i+1} emits from state h)

Initialization: $b_k(L) = a_{k0}$

Termination: $P(x) = \sum_h a_{0h} e_h (x_{i+1}) b_h(1)$

Note: The start and end state both are 0



•Hidden Markov Model (parameter estimation)

How we Specify the model in the first place?

Step1: Design the structure (states,connections)
Setp2: Estimate the transition a_{kh} and emission $e_k(b)$ probabilities.

Estimation when the state sequence is known

•$a_{kh} = A_{kh}/\sum_{h'} A_{kh'}$
•$e_k(b) = E_k(b) / \sum_{b'} E_k(b')$

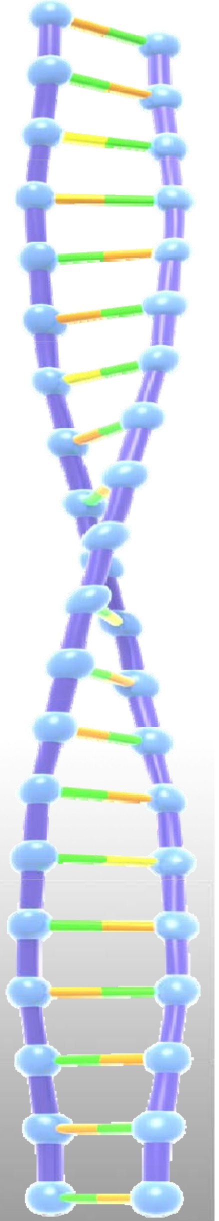
A_{kh} : number of transitions *k to h in training data* + r_{kh}
$E_k(b)$: number of emissions of *b from k in training data* + $r_k(b)$
Note: r_{kh} and $r_k(b)$ are pseudocounts.

Counting!

Estimation when the state sequence is unknown

•Baum-Welch algorithm
•Viterbi training

Training!



•Hidden Markov Model (parameter estimation)

Baum-Welch algorithm (EM)

Objective: Maximize $\sum_j \log P(x^j | \theta)$ (j training sequences)

θ : $a_{kh} = A_{kh} / \sum_{h'} A_{kh'}$ $e_k(b) = E_k(b) / \sum_{b'} E_k(b')$

A_{kh} and $E_k(b)$ are the *expected number of times each transition or emission is used, given the training sequences*.

$$A_{kh} = \sum_j \sum_i P(\pi_i = k, \pi_{i+1} = h | x^j, \theta)$$

$$E_k(b) = \sum_j \sum_i P(\pi_i = k, x_i = b | x^j, \theta)$$

$$P(\pi_i = k, \pi_{i+1} = h | x, \theta) = f_k(i) a_{kh} e_h(x_{i+1}) b_h(i+1) / P(x)$$

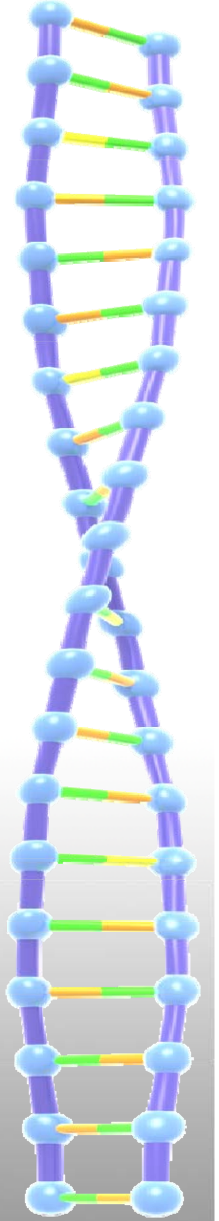
$$P(\pi_i = k, x_i = b | x, \theta) = f_k(i) b_k(i) / P(x) \text{ when } x_i = b, \text{ 0 otherwise.}$$

Recursion

```
For each sequence j = 1 ... n:
    Calculate $f_k(i)$ for sequence j using the forward algorithm
    Calculate $b_k(i)$ for sequence j using the backward algorithm
    Add the contribution of sequence j to $A_{kh}$ and $E_k(b)$.
    Calculate the new model parameters $a_{kh}$ and $e_k(b)$.
Calculate the new log likelihood of the model
```

Initialization: Pick arbitrary model parameters. θ [a_{kh}, $e_k(b)$]

Termination: Stop if the change in log likelihood is less than some predefined threshold or the maximum number of iterations is exceeded.



•Hidden Markov Model (parameter estimation)

Viterbi Training algorithm

Objective: Maximize $\sum_j \log P(x^j | \theta, \pi^*(x_j))$ (j training sequences)

This is not a true maximum likeihood objective function, but this alogrithm can coverges precisely , because the assignment of paths is a discrete process, and we can continue until none of the paths change.

θ : $a_{kh} = A_{kh} / \sum_{h'} A_{kh'}$ $e_k(b) = E_k(b) / \sum_{b'} E_k(b')$

A_{kh} and $E_k(b)$ are the *number of times each transition or emission is used, given the training* sequence and its most probable path.

Recursion

For each sequence *j = 1 … n:*

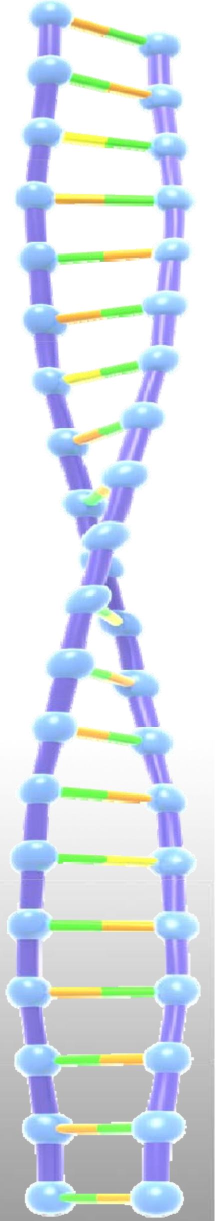
  Find the most probabel path $\pi^*(x_j)$ and its probability.

  Given the path calculate the new parameter by counting.

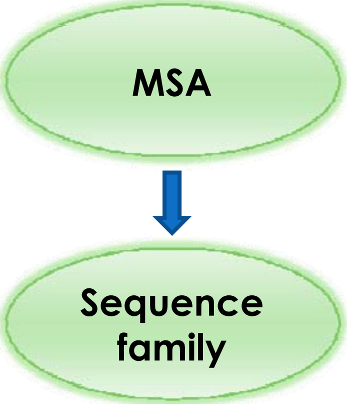
Calculate the new log likelihood of the model

Initialization: Pick arbitrary model parameters. θ $[a_{kh}, e_k(b)]$

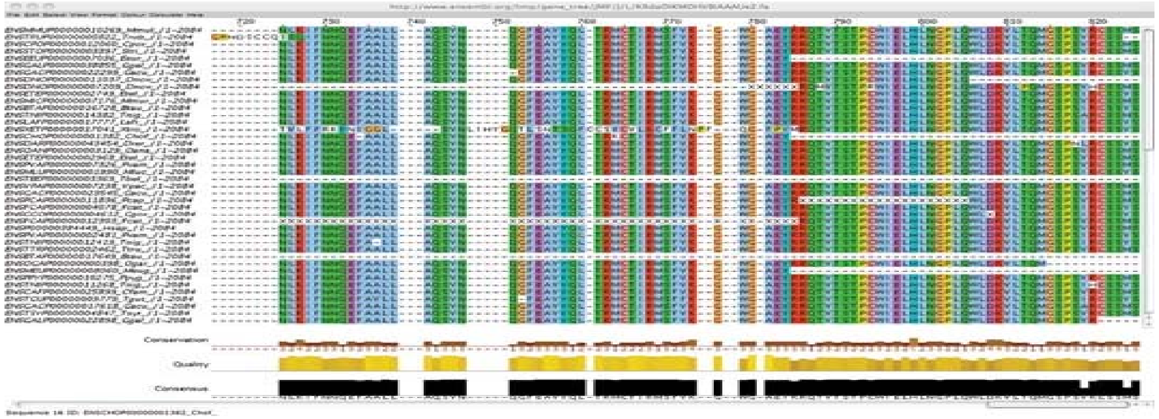
Termination: Stop when no paths changes.



•PSSM and Profile HMM


MSA

Sequence family



Target Sequence → … MSAVSCSTASSSGGRFRSKKKTSIHSP…

Are these conserved features present in the target sequence?

We need a statistic model!

PSSM

Poistion sepecific score matrix

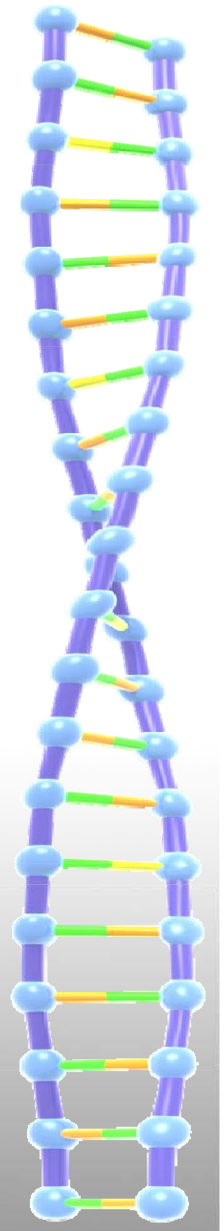
$$P(x|M) = \prod_{i=1toL} e_i(x_i)$$

$e_i(x_i)$: the probability of observing residue x_i in position i.

$$S\ (score) = \sum_{i=1toL} \log(ei(xi)/q(xi))$$

$q(x_i)$: the probability of x_i ***under a random*** model

Inadequate representation of the MSA (no gaps!)



•PSSM and Profile HMM

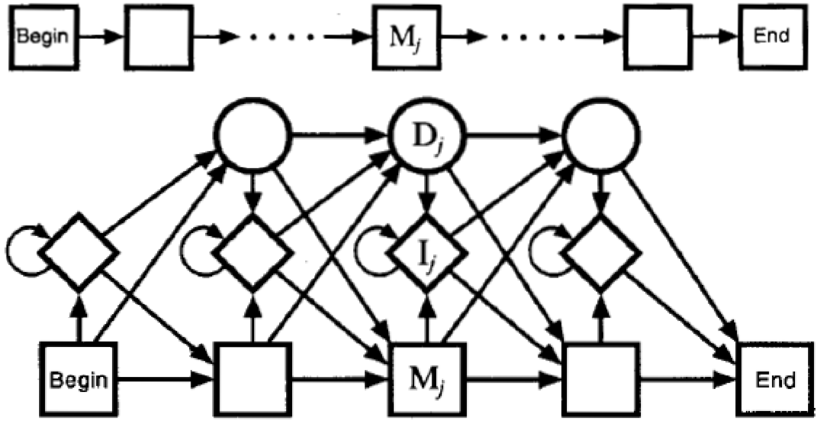
Are these conserved features present in the target sequence?
We need a statistic model!

Profile HMM

Deal with insertion and deletion

PSSM:

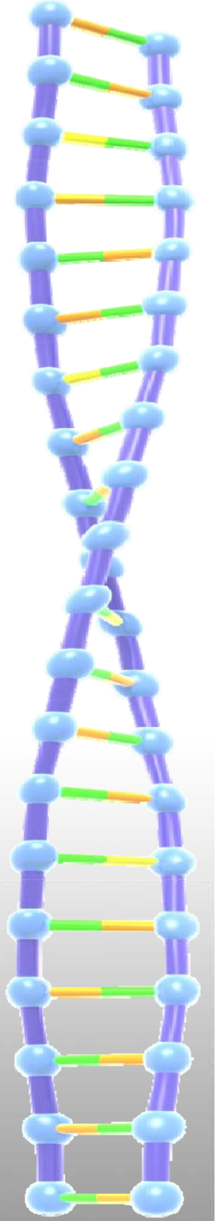
HMM:



M_j: Match state (emit resude with probability $e_{Mj}(b)$, b is one of 20 possible AAs)
I_j : Insertion state (allow multiple insertions, emit residues ramdomly)
D_j: Deletion state (dummy state, emit no residue to skip current position)

From multi-sqeuence alignment, we could determine the number of match states (design HMM) and the model parameters (train HMM).

Score : Log P(x|M)/P(x|R) M: HMM model R: Random model
The score is calucualted in bits, a high score means the target sequence is more likely belong to sequence family from which M is trained.



•HMMer 3 and HMM databases

HMMer3

- **Website:** http://hmmer.janelia.org/
- **User Guide:** ftp://selab.janelia.org/pub/software/hmmer3/3.0/Userguide.pdf

Main Functions

•**Hmmbuild:** Build a profile HMM from an input multiple alignment.
•**Hmmsearch:** Search a profile HMM against a sequence database.
•**Hmmscan:** Search a sequence against a profile HMM database.

Other utilities

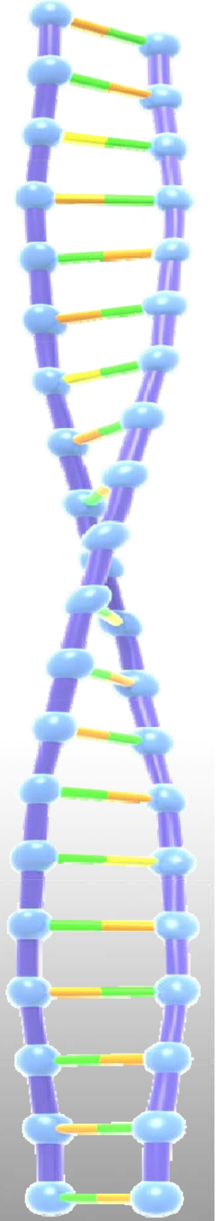
•**Hmmconvert:** Convert profile formats to/from HMMER3 format.
•**Hmmemit:** Generate (sample) sequences from a profile HMM.
•**Hmmfetch:** Get a profile HMM by name or accession from an HMM database.
•**Hmmpress:** Format an HMM database into a binary format for hmmscan.
•**Hmmstat:** Show summary statistics for each profile in an HMM database.

HMM databases

•**PFAM** (The wellcome trust sanger institue)
V25.0 (March 2011,12273 families) http://pfam.sanger.ac.uk/

•**TIGRFAM** (J. Craig Venter Institute)
V11.0 (August 2011) http://www.jcvi.org/cgi-bin/tigrfams/index.cgi

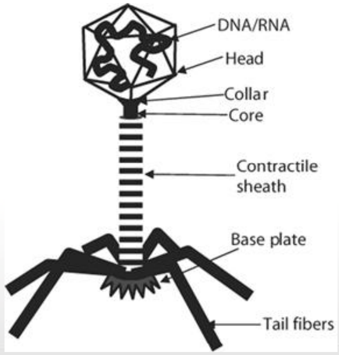
Note: PFAM and TiGRFAM both support HMMer3.



•Project (Phage/Prophage genome annotation)

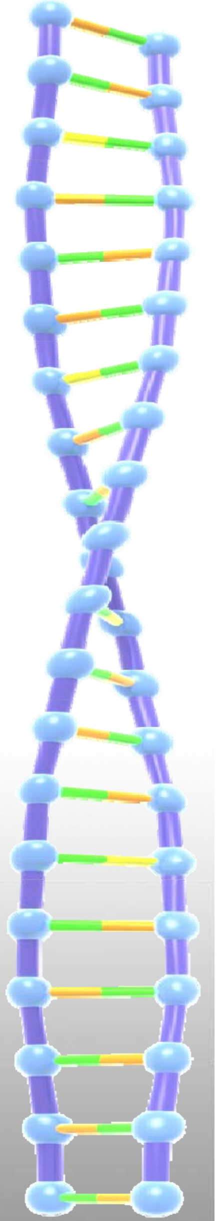
- **Website:** http://genedog.med.utoronto.ca:7777/joewu/war/Ppd1.html

- **Browser End:** Google web tool kit (Asynchronous javascript and XML-Ajax)
- **Server End:** Perl (CGI)
- **Database:** MySQL
- **Data Pineline:** JSON(Javascript Object Notation)



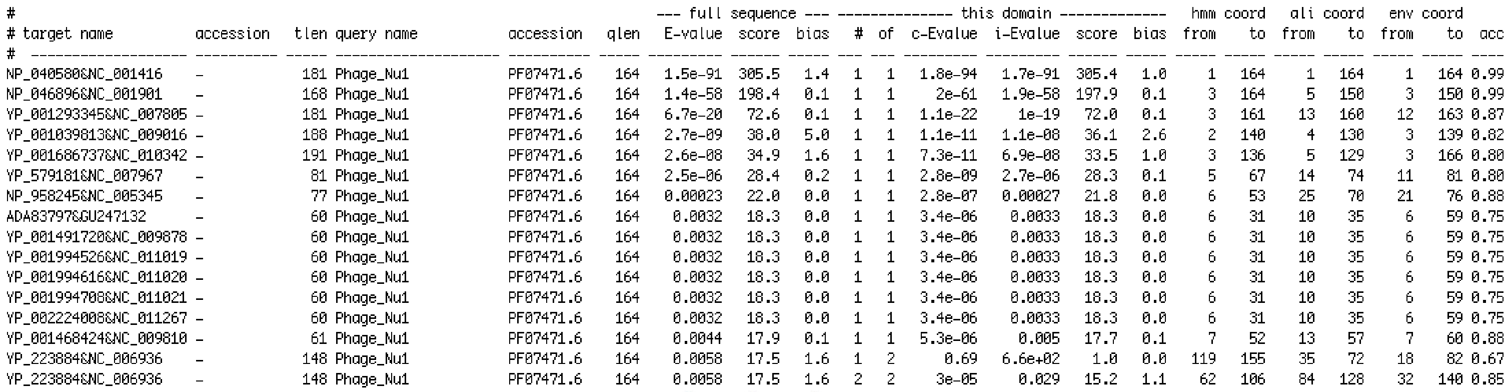
Bacteriophages (phages), the viruses infecting bacteria, are the most abundant biological entities on earth. Most bacterial genomes contain multiple integrated phage genomes, called prophages, many of which are capable of producing viable phage particles. These prophages often contain genes involved in bacterial pathogenesis, and they can also mediate significant changes in bacterial physiology.

This project involves the construction of a web platform that will use to accurately annotate proteins required for phage morphogenesis. The project involves using a set of sequence profiles (Profile HMM) derived from alignments of sequences that are clearly homologous as determined from sequence similarity and genome position. The web platform will be designed to efficiently assess the usefulness of these HMMs in accurately identifying proteins of known function. The database will incorporate genomic position data to validate the accuracy of annotations provided by the HMMs.



•Typical Hmmer3 output

> hmmsearch <xxxxx.hmm> <sequence database> xxxx.out



```
#                                                                           --- full sequence --- -------------- this domain -------------   hmm coord   ali coord   env coord
# target name        accession   tlen query name           accession   qlen   E-value  score  bias   #  of  c-Evalue  i-Evalue  score  bias  from    to  from    to  from    to  acc
# ------------------- ---------- ----- -------------------- ---------- ----- --------- ------ ----- --- --- --------- --------- ------ ----- ----- ----- ----- ----- ----- ----- ----
NP_040580&NC_001416  -            181 Phage_Nu1            PF07471.6    164   1.5e-91  305.5   1.4   1   1   1.8e-94   1.7e-91  305.4   1.0     1   164     1   164     1   164 0.99
NP_046896&NC_001901  -            168 Phage_Nu1            PF07471.6    164   1.4e-58  198.4   0.1   1   1     2e-61   1.9e-58  197.9   0.1     3   164     5   150     3   150 0.99
YP_001293345&NC_007805 -          181 Phage_Nu1            PF07471.6    164   6.7e-20   72.6   0.1   1   1   1.1e-22     1e-19   72.0   0.1     3   161    13   160    12   163 0.87
YP_001039813&NC_009016 -          188 Phage_Nu1            PF07471.6    164   2.7e-09   38.0   5.0   1   1   1.1e-11   1.1e-08   36.1   2.6     2   140     4   130     3   139 0.82
YP_001686737&NC_010342 -          191 Phage_Nu1            PF07471.6    164   2.6e-08   34.9   1.6   1   1   7.3e-11   6.9e-08   33.5   1.0     3   136     5   129     3   166 0.80
YP_579181&NC_007967  -             81 Phage_Nu1            PF07471.6    164   2.5e-06   28.4   0.2   1   1   2.8e-09   2.7e-06   28.3   0.1     5    67    14    74    11    81 0.80
NP_958245&NC_005345  -             77 Phage_Nu1            PF07471.6    164   0.00023   22.0   0.0   1   1   2.8e-07   0.00027   21.8   0.0     6    53    25    70    21    76 0.88
ADA83797&GU247132    -             60 Phage_Nu1            PF07471.6    164    0.0032   18.3   0.0   1   1   3.4e-06    0.0033   18.3   0.0     6    31    10    35     6    59 0.75
YP_001491720&NC_009878 -           60 Phage_Nu1            PF07471.6    164    0.0032   18.3   0.0   1   1   3.4e-06    0.0033   18.3   0.0     6    31    10    35     6    59 0.75
YP_001994526&NC_011019 -           60 Phage_Nu1            PF07471.6    164    0.0032   18.3   0.0   1   1   3.4e-06    0.0033   18.3   0.0     6    31    10    35     6    59 0.75
YP_001994616&NC_011020 -           60 Phage_Nu1            PF07471.6    164    0.0032   18.3   0.0   1   1   3.4e-06    0.0033   18.3   0.0     6    31    10    35     6    59 0.75
YP_001994708&NC_011021 -           60 Phage_Nu1            PF07471.6    164    0.0032   18.3   0.0   1   1   3.4e-06    0.0033   18.3   0.0     6    31    10    35     6    59 0.75
YP_002224008&NC_011267 -           60 Phage_Nu1            PF07471.6    164    0.0032   18.3   0.0   1   1   3.4e-06    0.0033   18.3   0.0     6    31    10    35     6    59 0.75
YP_001468424&NC_009810 -           61 Phage_Nu1            PF07471.6    164    0.0044   17.9   0.1   1   1   5.3e-06     0.005   17.7   0.1     7    52    13    57     7    60 0.88
YP_223884&NC_006936  -            148 Phage_Nu1            PF07471.6    164    0.0058   17.5   1.6   1   2      0.69   6.6e+02    1.0   0.0   119   155    35    72    18    82 0.67
YP_223884&NC_006936  -            148 Phage_Nu1            PF07471.6    164    0.0058   17.5   1.6   2   2     3e-05     0.029   15.2   1.1    62   106    84   128    32   140 0.85
```

- **E-value:**

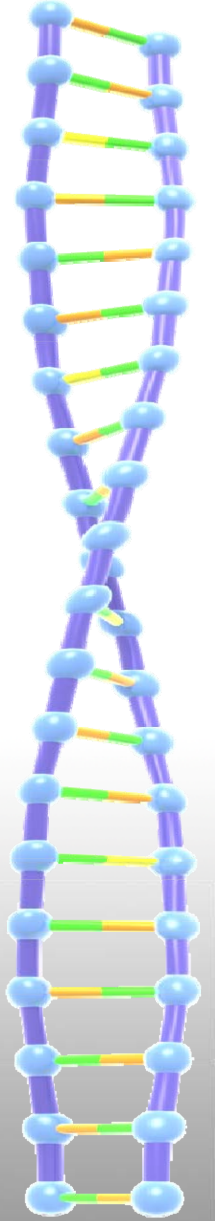
  The statistical significance of the match to this sequence: the number of hits we'd expect to score this highly in a database of this size if the database contained only random sequences. The lower the E-value, the more significant the hit. **[Extreme value (Gumbel) distribution]**

- **Score:**

  The log-odds score for the complete sequence.

- **Bias:**

  A correction term for biased sequence composition that's been applied to the sequence bit score.



The End