Lecture 4

Modeling Biological Sequences using Hidden Markov Models
<table>
<thead>
<tr>
<th>Project</th>
<th>Psets</th>
<th>Week</th>
<th>Date</th>
<th>Topic</th>
<th>Lec</th>
<th>Topic</th>
<th>Read*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Describe your previous research, areas of interest in computational biology, type of project that best fits your interests. Post in a profile that lets your classmates know you and find potential partners.</td>
<td>PS1 out on: L1-L5</td>
<td>1</td>
<td>Thu, Sep 7</td>
<td>Introduction</td>
<td>L1</td>
<td>Intro: Biology, Algorithms, Machine Learning, Course Overview</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Sep 8</td>
<td></td>
<td></td>
<td>R1</td>
<td>Recitation: 1: Biology and Probability Review</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Sep 12</td>
<td></td>
<td>Module I: Aligning and Modeling Genomes</td>
<td>L2</td>
<td>Alignment I: Dynamic Programming, Global and local alignment</td>
<td>2,3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Sep 14</td>
<td></td>
<td></td>
<td>L3</td>
<td>Alignment II: Database search, Rapid string matching, BLAST, BLOSUM</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Sep 15</td>
<td></td>
<td></td>
<td>R2</td>
<td>Recitation 2: Deriving Parameters of Alignment, Multiple Alignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Sep 19</td>
<td></td>
<td>Module II: Gene Expression and Epigenomics</td>
<td>L4</td>
<td>Hidden Markov Models Part 1: Evaluation/Parsing, Viterbi, Forward algorithms</td>
<td>7,8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Sep 21</td>
<td></td>
<td></td>
<td>L5</td>
<td>Hidden Markov Models Part 2: Posterior Decoding, Learning, Baum-Welch</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Sep 22</td>
<td></td>
<td></td>
<td></td>
<td>No classes - student holiday</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Mon, Sep 25</td>
<td></td>
<td></td>
<td></td>
<td>Mon, Sep 25</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Project profile due Tue 9/26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Project Profile: about the projects, self introductions, mentor intro, example projects, teamwork 32D-507</td>
<td></td>
</tr>
<tr>
<td>Identify previous project proposals, recent papers, and potential partners that match your areas of interest. List initial project ideas and partners.</td>
<td>PS2 out on: L6-R4</td>
<td>4</td>
<td>Thu, Sep 26</td>
<td>Expression Analysis: Clustering/Classification, K-means, Hierarchical, Bayesian</td>
<td>L6</td>
<td>Expression Analysis: Clustering/Classification, K-means, Hierarchical, Bayesian</td>
<td>15,16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Sep 28</td>
<td></td>
<td></td>
<td>L7</td>
<td>Transcript structure: GtScan, RNA-seq, Mapping, De novo Assembly, Diff Expr</td>
<td>14,15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Sep 29</td>
<td></td>
<td></td>
<td>R3</td>
<td>Recitation 3: Affinity Propagation Clustering and Random Forest Classification</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Oct 3</td>
<td></td>
<td>Module II: Gene Expression and Epigenomics</td>
<td>L8</td>
<td>Epigenomics: ChiP-Seq, Read mapping, Peak calling, IDR, Chromatin states</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Oct 4</td>
<td></td>
<td></td>
<td>L9</td>
<td>Three-dimensional chromatin interactions: 3C, 5C, HiC, ChIA-PET</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Oct 6</td>
<td></td>
<td></td>
<td>R4</td>
<td>Recitation 4: ENCODE, Epigenome Roadmap, ChromHMM, ChromImpute</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Project Planning: research areas, initial ideas, type of project, mentor matching, finding partners 32D-507</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Project Planning: research areas, initial ideas, type of project, mentor matching, finding partners 32D-507</td>
<td></td>
</tr>
<tr>
<td>Form teams of two, specify project goals, division of work, milestones, datasets, challenges Prepare slide presentation for the class and the mentors. Project proposal due Tue 10/19. Presented on Fri 10/20</td>
<td>PS3 out on: L10-R6</td>
<td>6</td>
<td>Thu, Oct 10</td>
<td>No Classes - Columbus Day Holiday</td>
<td>L10</td>
<td>Regulatory Motifs: Discovery, Representation, PBMs, Gibbs Sampling, EM</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Oct 12</td>
<td></td>
<td></td>
<td>R5</td>
<td>Recitation 5: Gapped Motif Discovery, DNAseP, PhyloBayes</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Oct 13</td>
<td></td>
<td>Module III: Regulatory Genomics and Networks</td>
<td>L11</td>
<td>Network structure, centrality, SVD, sparse PCA, L1/L2, modules, diffusion kernels</td>
<td>20,21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Oct 17</td>
<td></td>
<td></td>
<td>L12</td>
<td>Deep Learning, Neural Nets, Convolutional NNs, Recurrent NNs, Autoencoder</td>
<td>20,7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Oct 19</td>
<td></td>
<td></td>
<td>R6</td>
<td>Recitation 6: Networks review, Recommendation systems, EHR, PhEWS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Oct 20</td>
<td></td>
<td>Project Feedback: Prepare 2-3 slide presentation of your term project for your mentor. 32D-507 at 4-5pm</td>
<td></td>
<td>Project Feedback: Prepare 2-3 slide presentation of your term project for your mentor. 32D-507 at 4-5pm</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Oct 20</td>
<td></td>
<td></td>
<td></td>
<td>Project Feedback: Prepare 2-3 slide presentation of your term project for your mentor. 32D-507 at 4-5pm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Evaluate/discuss three peer proposals, NIH review format. Review Panels Fri 10/27 Reviews back Tue 10/31</td>
<td>7</td>
<td>Thu, Oct 24</td>
<td>Evaluation/Discussion: reconciling strategies for improvement, feedback to author 32D-507</td>
<td></td>
<td>Evaluation/Discussion: reconciling strategies for improvement, feedback to author 32D-507</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Oct 26</td>
<td></td>
<td>Module IV: Population Genetics and Disease Genomics</td>
<td>L13</td>
<td>Population genetics: Linkage disequilibrium, pop struct, 1000genomes, allele freq</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Oct 27</td>
<td></td>
<td></td>
<td>L14</td>
<td>Disease Association Mapping, GWAS, organismal phenotypes</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Oct 27</td>
<td></td>
<td></td>
<td>R7</td>
<td>Recitation 7: Linkage Disequilibrium, Haplotype Phasing, Genotype Imputation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Oct 31</td>
<td></td>
<td>Module V: Comparative Genomics and Evolution</td>
<td>L15</td>
<td>Quantitative trait mapping, molecular traits, eQTLs, mediation analysis, imiwAS</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Nov 2</td>
<td></td>
<td></td>
<td>L16</td>
<td>Missing Heritability, Complex Traits, Interpret GWAS, Rank-based enrichment</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Nov 3</td>
<td></td>
<td></td>
<td>R8</td>
<td>Recitation 8: Rare Variants, EXAC</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Nov 7</td>
<td></td>
<td>Module V: Comparative Genomics and Evolution</td>
<td>L17</td>
<td>Comparative genomics and evolutionary signatures</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Nov 8</td>
<td></td>
<td></td>
<td>L18</td>
<td>Genome Scale Evolution, Genome Duplication</td>
<td>4.5,7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Nov 10</td>
<td></td>
<td>No Recitation, Veterans Day</td>
<td></td>
<td>No Recitation, Veterans Day</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Nov 14</td>
<td></td>
<td>Frontiers</td>
<td>L19</td>
<td>Phylogenetics: Molecular evolution, Tree building, Phylogenetic inference</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Nov 16</td>
<td></td>
<td></td>
<td>L20</td>
<td>Phylogenetics: Gene/species trees, reconciliation, coalescent, ARGS</td>
<td>28</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Nov 17</td>
<td></td>
<td>Frontiers</td>
<td>R9</td>
<td>Recitation 9: Phylogenetic distance metrics, Coalescent Process</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Nov 21</td>
<td></td>
<td>Quiz In Class Quiz (the only quiz - the class has no final exam)</td>
<td></td>
<td>Quiz In Class Quiz (the only quiz - the class has no final exam)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Nov 23</td>
<td></td>
<td>No Recitation, thanksgiving break</td>
<td></td>
<td>No Recitation, thanksgiving break</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Nov 24</td>
<td></td>
<td>Foundations</td>
<td>L21</td>
<td>Single-cell genomics: technology, analysis, microfluidics, applications, insights</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Nov 30</td>
<td></td>
<td>Frontiers</td>
<td>L22</td>
<td>Mining human phenotypes, PhEWS, UK Biobank, meta-phenotypes+imputation</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Dec 1</td>
<td></td>
<td>Frontiers</td>
<td>R10</td>
<td>Recitation 10: Project Feedback, results, interpretation, directions</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Dec 5</td>
<td></td>
<td>L23</td>
<td>Cancer Genomics, Single-cell Sequencing, Tumor-Immune Interface</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Thu, Dec 7</td>
<td></td>
<td>L24</td>
<td>Genome Engineering with CRISPR/Cas9 and related technologies</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fri, Dec 8</td>
<td></td>
<td>Frontiers</td>
<td>R11</td>
<td>Recitation 11: Presentation Tips - Intro, discussion, Slides, Presentation skills</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Dec 12</td>
<td></td>
<td>L25</td>
<td>Final Presentations - Part I (11am), 32-G8 reading room</td>
<td></td>
<td>L25</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tue, Dec 12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* readings refer to chapters in compiled 2016 scribe notes, available in the materials folder on Stellar
** recitation topics will be adjusted to respond to lecture and student needs
Module 1: Aligning and modeling genomes

• Module 1: Computational foundations
  – Dynamic programming: exploring exponential spaces in poly-time
  – Linear-time string matching, Hashing, Content-based indexing
  – Hidden Markov Models: decoding, evaluation, parsing, learning

• Last week: Sequence alignment / comparative genomics
  – Local/global alignment: infer nucleotide-level evolutionary events
  – Database search: scan for regions that may have common ancestry

• This week: Modeling genomes / exon / CpG island finding
  – Modeling class of elements, recognizing members of a class
  – Application to gene finding, conservation islands, CpG islands
We have learned how to align sequences to other sequences

- **L2: Sequence alignment**
  - Dynamic programming, duality path ↔ alignment
  - Global / local alignment, general gap penalties

- **L3: Rapid string search**
  - Exact string match, semi-numerical matching
  - Database search: Hashing, BLAST, variations

- **L17: Comparative genomics: evolutionary signatures**
  - Tell me how you evolve, I’ll tell you what you are
  - Identifying conserved elements through evolution

- **L18: Whole-genome assembly/alignment/duplication:**
  - Finding all common substrings within/across species
  - Contigs/scaffolds, string graphs, glocal alignmt paths

- Problem set 1 due next Tues, project planning
Today: apply these ideas to model DNA sequences

…GTACTCACC GGTTACAGGATTATGGGTTACAGGTAACCGTT…

• What to do with a completely new piece of DNA
  – Align it to things we know about (database search)
  – Align it to things we don’t know about (assembly)
• Stare at it
  – Non-standard nucleotide composition?
  – Interesting k-mer frequencies?
  – Recurrent patterns?
• Model it
  – Make some hypotheses about it
  – Build a ‘generative model’ to describe it
  – Find sequences of similar type

⇒ How do we model DNA sequences?
Modeling biological sequences with HMMs
(a.k.a. What to do with big unlabelled chunks of DNA)

- **Ability to emit** DNA sequences of a certain *type*
  - Not exact alignment to previously known gene
  - Preserving ‘properties’ of *type*, not identical sequence
- **Ability to recognize** DNA sequences of a certain type (state)
  - What (hidden) state is most likely to have generated observations
  - Find set of states and transitions that generated a long sequence
- **Ability to learn** distinguishing characteristics of each state
  - Training our generative models on large datasets
  - Learn to classify unlabelled data
Why Probabilistic Sequence Modeling?

• Biological data is noisy

• Probability provides a calculus for manipulating models

• Not limited to yes/no answers – can provide “degrees of belief”

• Many common computational tools based on probabilistic models

• Our tools:
  – Markov Chains and Hidden Markov Models (HMMs)
Markov Chains and Hidden Markov Models
Andrey Markov (1856-1922)
Predicting tomorrow’s weather

- Markov Chain

- What you see is what you get: next state only depends on current state (no memory)

- Hidden Markov Model

- Hidden state of the world (e.g. storm system) determines emission probabilities
- State transitions governed by a Markov chain

All observed
HMM nomenclature for this course

- **Vector** $\mathbf{x} = \text{Sequence of observations}$
- **Vector** $\mathbf{\pi} = \text{Hidden path (sequence of hidden states)}$
- **Transition matrix** $\mathbf{A} = a_{kl} = \text{probability of } k \rightarrow l \text{ state transition}$
- **Emission vector** $\mathbf{E} = e_k(x_i) = \text{prob. of observing } x_i \text{ from state } k$
- **Bayes’s rule**: Use $P(x_i|\mathbf{\pi}_i=k)$ to estimate $P(\mathbf{\pi}_i=k|x_i)$

**Transitions**: $a_{kl} = P(\mathbf{\pi}_i=l|\mathbf{\pi}_{i-1}=k)$
- Transition probability from state $k$ to state $l$

**Emissions**: $e_k(x_i) = P(x_i|p_i=k)$
- Emission probability of symbol $x_i$ from state $k$
**Components of a Markov Chain**

**Definition:** A *Markov chain* is a triplet \((Q, p, A)\), where:

- \(Q\) is a finite set of states. Each state corresponds to a symbol in the alphabet \(\Sigma\).
- \(p\) is the initial state probabilities.
- \(A\) is the state transition probabilities, denoted by \(a_{st}\) for each \(s, t\) in \(Q\).

For each \(s, t\) in \(Q\) the transition probability is: \(a_{st} \equiv P(x_i = t|x_{i-1} = s)\)

**Output:** The output of the model is the set of states at each instant time \(\Rightarrow\) the set of states are observable.

**Property:** The probability of each symbol \(x_i\) depends only on the value of the preceding symbol \(x_{i-1}:\ P(x_i | x_{i-1}, \ldots, x_1) = P(x_i | x_{i-1})\)

**Formula:** The probability of the sequence:

\[
P(x) = P(x_L, x_{L-1}, \ldots, x_1) = P(x_L | x_{L-1}) P(x_{L-1} | x_{L-2}) \ldots P(x_2 | x_1) P(x_1)
\]
Components of an HMM (Hidden Markov Model)

**Definition:** An **HMM** is a 5-tuple \((Q, V, p, A, E)\), where:

- \(Q\) is a finite set of states, \(|Q| = N\)
- \(V\) is a finite set of observation symbols per state, \(|V| = M\)
- \(p\) is the initial state probabilities.
- \(A\) is the state transition probabilities, denoted by \(a_{st}\) for each \(s, t \in Q\).
  - For each \(s, t \in Q\) the transition probability is: \(a_{st} = P(x_i = t | x_{i-1} = s)\)
- \(E\) is a probability emission matrix, \(e_{sk} = P(v_k \text{ at time } t | q_t = s)\)

**Output:** Only emitted symbols are observable by the system but not the underlying random walk between states \(-\) “hidden”

**Property:** Emissions and transitions are dependent on the current state only and not on the past.
<table>
<thead>
<tr>
<th>Scoring</th>
<th>Decoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Scoring ( x ), one path</td>
<td>3. Viterbi decoding</td>
</tr>
<tr>
<td>( P(x, \pi) )</td>
<td>( \pi^* = \arg\max_\pi P(x, \pi) )</td>
</tr>
<tr>
<td>Prob of a path, emissions</td>
<td>Most likely path</td>
</tr>
<tr>
<td>2. Scoring ( x ), all paths</td>
<td>4. Posterior decoding</td>
</tr>
<tr>
<td>( P(x) = \sum_\pi P(x, \pi) )</td>
<td>( \pi^\wedge = { \pi_i \mid \pi_i = \arg\max_k \sum_\pi P(\pi_i = k</td>
</tr>
<tr>
<td>Prob of emissions, over all paths</td>
<td>Path containing the most likely state at any time point.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Learning</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>5. Supervised learning, given ( \pi )</td>
<td>6. Unsupervised learning.</td>
</tr>
<tr>
<td>( \Lambda^* = \arg\max_\Lambda P(x, \pi</td>
<td>\Lambda) )</td>
</tr>
<tr>
<td>6. Unsupervised learning.</td>
<td>Viterbi training, best path</td>
</tr>
<tr>
<td>( \Lambda^* = \arg\max_\Lambda \max_\pi P(x, \pi</td>
<td>\Lambda) )</td>
</tr>
</tbody>
</table>
Examples of HMMs

The dishonest casino
The dishonest genome
... and many more
Example: The Dishonest Casino

A casino has two dice:

- **Fair die**
  \[ P(1) = P(2) = P(3) = P(5) = P(6) = \frac{1}{6} \]

- **Loaded die**
  \[ P(1) = P(2) = P(3) = P(4) = P(5) = \frac{1}{10} \]
  \[ P(6) = \frac{1}{2} \]

Casino player switches between fair and loaded die on average once every 20 turns

**Game:**

1. You bet $1
2. You roll (always with a fair die)
3. Casino player rolls (maybe with fair die, maybe with loaded die)
4. Highest number wins $2

Slide credit: Serafim Batzoglou
The dishonest casino model

Hidden (model)

Fair

Loaded

Observed (world)

\[
\begin{align*}
P(1|\text{Fair}) &= 1/6 \\
P(2|\text{Fair}) &= 1/6 \\
P(3|\text{Fair}) &= 1/6 \\
P(4|\text{Fair}) &= 1/6 \\
P(5|\text{Fair}) &= 1/6 \\
P(6|\text{Fair}) &= 1/6 \\
P(1|\text{L}) &= 1/10 \\
P(2|\text{L}) &= 1/10 \\
P(3|\text{L}) &= 1/10 \\
P(4|\text{L}) &= 1/10 \\
P(5|\text{L}) &= 1/10 \\
P(6|\text{L}) &= 1/2
\end{align*}
\]
The dishonest genome model

Hidden (model)

Virus

“Self”

Observed (world)

\[
\begin{align*}
P(A|\text{Virus}) &= 1/6 \\
P(T|\text{Virus}) &= 1/6 \\
P(C|\text{Virus}) &= 1/3 \\
P(G|\text{Virus}) &= 1/3 \\
P(A|\text{Self}) &= 1/4 \\
P(T|\text{Self}) &= 1/4 \\
P(C|\text{Self}) &= 1/4 \\
P(G|\text{Self}) &= 1/4
\end{align*}
\]

Slide credit: Serafim Batzoglou
# Examples of HMMs for genome annotation

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology / Transitions</td>
<td>2 states, different nucleotide composition</td>
<td>2 states, different conservation levels</td>
<td>2 states, different trinucleotide composition</td>
<td>2 states, different evolutionary signatures</td>
<td>~20 states, different composition / conservation, specific structure</td>
<td>40 states, different chromatin mark combinations</td>
</tr>
<tr>
<td>Hidden States / Annotation</td>
<td>GC-rich / AT-rich</td>
<td>Conserved / non-conserved</td>
<td>Coding exon / non-coding (intron or intergenic)</td>
<td>Coding exon / non-coding (intron or intergenic)</td>
<td>First / last / middle coding exon, UTRs, intron1/2/3, intergenic, *(+/- strand)</td>
<td>Enhancer / promoter / transcribed / repressed / repetitive</td>
</tr>
<tr>
<td>Emissions / Observations</td>
<td>Nucleotides</td>
<td>Level of conservation</td>
<td>Triplets of nucleotides</td>
<td>Nucleotide triplets, conservation levels</td>
<td>Codons, nucleotides, splice sites, start / stop codons</td>
<td>Vector of chromatin mark frequencies</td>
</tr>
</tbody>
</table>
Running the model: Probability of a sequence

What is the joint probability of observing $x$ and a specific path $\pi$:

$\pi = \text{Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair}$

and rolls

$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

Joined probability $P(x, \pi) = P(x | \pi) P(\pi) = P(\text{emissions} | \text{path}) * P(\text{path})$

$P = \frac{1}{2} \times P(1 \mid \text{Fair}) \times P(\text{Fair}_{i+1} \mid \text{Fair}_i) \times P(2 \mid \text{Fair}) \times P(\text{Fair} \mid \text{Fair}) \cdots P(4 \mid \text{Fair})$

$= \frac{1}{2} \times (1/6)^{10} \times (0.95)^9$

$= 5.2 \times 10^{-9}$

Why is $p$ so small?
Running the model: Probability of a sequence

What is the likelihood of

$$\pi = \text{Load, Load, Load, Load, Load, Load, Load, Load, Load, Loaded}$$

and rolls

$$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$$

emission transition emission transition emission

$$p = \frac{1}{2} \times P(1 \mid \text{Load}) \ P(\text{Load}_{i+1} \mid \text{Load}_i) \ P(2 \mid \text{Load}) \ P(\text{Load} \mid \text{Load}) \ldots \ P(4 \mid \text{Fair})$$

$$= \frac{1}{2} \times (1/10)^8 \times (1/2)^2 \ (0.95)^9$$

$$= 7.9 \times 10^{-10}$$

Compare the two!
Comparing the two paths

Two sequence paths:

\[ P( x, \text{all-Fair} ) = 5.2 \times 10^{-9} \quad \text{(very small)} \]

\[ P( x, \text{all-Loaded} ) = 7.9 \times 10^{-10} \quad \text{(very very small)} \]

Likelihood ratio:

\[ P( x, \text{all-Fair} ) \text{ is 6.59 times more likely than } P( x, \text{all-Loaded} ) \]

It is 6.59 times more likely that the die is fair all the way, than loaded all the way.
What about partial runs and die switching

What is the likelihood of

\[ \pi = \text{Fair, Fair, Fair, Fair, Load, Load, Load, Load, Fair, Fair} \]

and rolls

\[ x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4 \]

\[ p = \frac{1}{2} \times P(1 \mid \text{Fair}) \times P(\text{Fair}_{i+1} \mid \text{Fair}_i) \times P(2 \mid \text{Fair}) \times P(\text{Fair} \mid \text{Fair}) \times \ldots \times P(4 \mid \text{Fair}) \]

\[ = \frac{1}{2} \times (1/10)^2 \times (1/2)^2 \times (1/6)^5 \times (0.95)^7 \times (0.05)^2 \]

\[ = 2.8 \times 10^{-10} \]

Much less likely, due to high cost of transitions
Let the sequence of rolls be:
\[ x = 1, 6, 6, 5, 6, 2, 6, 6, 3, 6 \]

Now, what is the likelihood \( \pi = F, F, \ldots, F \)?
\[
\frac{1}{2} \times (1/6)^{10} \times (0.95)^9 = 0.5 \times 10^{-9}, \text{ same as before}
\]

What is the likelihood \( \pi = L, L, \ldots, L \)?
\[
\frac{1}{2} \times (1/10)^4 \times (1/2)^6 \times (0.95)^9 = 0.5 \times 10^{-7}
\]

So, it is 100 times more likely the die is loaded
<table>
<thead>
<tr>
<th>Scoring</th>
<th>Learning</th>
<th>Decoding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Scoring $x$, one path $P(x, \pi)$</td>
<td>5. Supervised learning, given $\pi$ $\Lambda^* = \text{argmax}_\Lambda P(x, \pi</td>
<td>\Lambda)$</td>
</tr>
<tr>
<td>Prob of a path, emissions</td>
<td>6. Unsupervised learning. $\Lambda^* = \text{argmax}<em>\Lambda \max</em>\pi P(x, \pi</td>
<td>\Lambda)$</td>
</tr>
<tr>
<td>2. Scoring $x$, all paths $P(x) = \sum_\pi P(x, \pi)$</td>
<td>6. Unsupervised learning $\Lambda^* = \text{argmax}<em>\Lambda \sum</em>\pi P(x, \pi</td>
<td>\Lambda)$</td>
</tr>
<tr>
<td>Prob of emissions, over all paths</td>
<td>Viterbi training, best path</td>
<td>Path containing the most likely state at any time point.</td>
</tr>
</tbody>
</table>
3. DECODING:
What was the sequence of hidden states?

Given: Model parameters $e_i(\cdot), a_{ij}$
Given: Sequence of emissions $x$

Find: Sequence of hidden states $\pi$
Finding the optimal path

• We can now evaluate any path through hidden states, given the emitted sequences

• How do we find the best path?

• Optimal substructure! Best path through a given state is:
  – Best path to previous state
  – Best transition from previous state to this state
  – Best path to the end state

⇒ Viterbi algorithm
  – Define $V_k(i) =$ Probability of the most likely path through state $\pi_i=k$
  – Compute $V_k(i+1)$ as a function of $\max_{k'} \{ V_{k'}(i) \}$

  $V_k(i+1) = e_k(x_{i+1}) \times \max_j a_{jk} V_j(i)$

⇒ Dynamic Programming
Andrew J. Viterbi, Massachusetts Beta ’57, as a new teacher in 1963.
Finding the most likely path

- Find path $\pi^*$ that maximizes total joint probability $P[ x, \pi ]$

$$P(x, \pi) = a_{0\pi_1}^* \prod_i e_{\pi_i}(x_i) \times a_{\pi_i\pi_{i+1}}$$

Slide credit: Serafim Batzoglou
Calculate maximum $P(x, \pi)$ recursively

1. Assume we know $V_j$ for the previous time step $(i-1)$

2. Calculate $V_k(i) =$ current max $\times$ this emission $\times$ (max ending in state $j$ at step $i$)$ \times$ Transition from state $j$.

Here, $e_k(x_i)$ represents the emission of observation $x_i$, $a_{jk}$ is the transition probability from state $j$ to state $k$, and $V_j(i-1)$ represents the previously calculated maximum probability ending in state $j$. The diagram illustrates the process with hidden states and observations.
The Viterbi Algorithm

Input: \( x = x_1 \ldots x_N \)

**Initialization:**
\[ V_0(0) = 1, \ V_k(0) = 0, \ \text{for all } k > 0 \]

**Iteration:**
\[ V_k(i) = e_K(x_i) \times \max_j a_{jk} \ V_j(i-1) \]

**Termination:**
\[ P(x, \pi^*) = \max_k V_k(N) \]

**Traceback:**
Follow max pointers back
Similar to aligning states to seq

**In practice:**
Use log scores for computation

**Running time and space:**
Time: \( O(K^2N) \)
Space: \( O(KN) \)

Slide credit: Serafim Batzoglou
<table>
<thead>
<tr>
<th>Learning</th>
<th>One path</th>
<th>All paths</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td>5. Scoring $x$, one path $P(x, \pi)$</td>
<td>6. Scoring $x$, all paths $P(x) = \sum_{\pi} P(x, \pi)$</td>
</tr>
<tr>
<td></td>
<td>Prob of a path, emissions</td>
<td>Prob of emissions, over all paths</td>
</tr>
<tr>
<td></td>
<td>$\pi^* = \arg\max_{\pi} P(x, \pi)$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Most likely path</td>
<td>Path containing the most likely state at any time point.</td>
</tr>
<tr>
<td>Unsupervised</td>
<td>6. Scoring $x$, one path $P(x, \pi)$</td>
<td>6. Scoring $x$, all paths $P(x) = \sum_{\pi} P(x, \pi)$</td>
</tr>
<tr>
<td></td>
<td>$\Lambda^* = \arg\max_{\Lambda} P(x, \pi</td>
<td>\Lambda)$</td>
</tr>
<tr>
<td></td>
<td>$\Lambda^* = \arg\max_{\Lambda} \max_{\pi} P(x, \pi</td>
<td>\Lambda)$</td>
</tr>
<tr>
<td></td>
<td>Viterbi training, best path</td>
<td></td>
</tr>
</tbody>
</table>
2. EVALUATION
(how well does our model capture the world)

Given: Model parameters $e_i(.)$, $a_{ij}$
Given: Sequence of emissions $x$

Find: $P(x|M)$, summed over all possible paths $\pi$
Simple: Given the model, generate some sequence x

Given a HMM, we can generate a sequence of length n as follows:
1. Start at state $\pi_1$ according to prob $a_{0\pi_1}$
2. Emit letter $x_1$ according to prob $e_{\pi_1}(x_1)$
3. Go to state $\pi_2$ according to prob $a_{\pi_1\pi_2}$
4. ... until emitting $x_n$

We have some sequence x that can be emitted by p. Can calculate its likelihood. However, in general, many different paths may emit this same sequence x. How do we find the total probability of generating a given x, over any path?

Slide credit: Serafim Batzoglou
Given a sequence $x$,
What is the probability that $x$ was generated by the model (using any path)?

\[ P(x) = \sum_{\pi} P(x, \pi) = \sum_{\pi} P(x|\pi) P(\pi) \]

- (weighted average of conditional probability, summed over all paths, weighted by each path’s probability)

- Challenge: exponential number of paths
Calculate probability of emission over all paths

- Each path has associated probability
  - Some paths are likely, others unlikely: sum them all up
  - Return total probability that emissions are observed, summed over all paths
  - Viterbi path is the most likely one
    - How much ‘probability mass’ does it contain?
- (cheap) alternative:
  - Calculate probability over maximum (Viterbi) path $\pi^*$
  - Good approximation if Viterbi has highest density
  - BUT: incorrect
- (real) solution
  - Calculate the exact sum iteratively
    - $P(x) = \sum_{\pi} P(x, \pi)$
  - Can use dynamic programming

Slide credit: Serafim Batzoglou
The Forward Algorithm – derivation

Define the forward probability:

\[ f_l(i) = P(x_1 \ldots x_i, \pi_i = l) \]

\[ = \sum_{\pi_1 \ldots \pi_{i-1}} P(x_1 \ldots x_{i-1}, \pi_1, \ldots, \pi_{i-2}, \pi_{i-1}, \pi_i = l) e_l(x_i) \]

\[ = \sum_k \sum_{\pi_1 \ldots \pi_{i-2}} P(x_1 \ldots x_{i-1}, \pi_1, \ldots, \pi_{i-2}, \pi_{i-1} = k) a_{kl} e_l(x_i) \]

\[ = \sum_k f_k(i-1) a_{kl} e_l(x_i) \]

\[ = e_l(x_i) \sum_k f_k(i-1) a_{kl} \]
Calculate total probability $\sum_{\pi} P(x, \pi)$ recursively

- Assume we know $f_j$ for the previous time step (i-1)

- Calculate $f_k(i) =$ \begin{align*}
\text{updated sum} & \quad e_k(x_i) * \sum_j (f_j(i-1) \times a_{jk}) \\
\text{this emission} & \quad \text{sum ending in state } j \text{ at step } i \times \text{transition from state } j \\
& \quad \text{every possible previous state } j
\end{align*}

Slide credit: Serafim Batzoglou
The Forward Algorithm

Input: $x = x_1 \ldots x_N$

**Initialization:**

$f_0(0) = 1$, $f_k(0) = 0$, for all $k > 0$

**Iteration:**

$f_k(i) = e_k(x_i) \times \sum_j a_{jk} f_j(i-1)$

**Termination:**

$P(x, \pi^*) = \sum_k f_k(N)$

**Running time and space:**

Time: $O(K^2N)$

Space: $O(KN)$

---

Slide credit: Serafim Batzoglou
<table>
<thead>
<tr>
<th>Scoring</th>
<th>Decoding</th>
<th>Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Scoring $x$, one path</td>
<td>3. Viterbi decoding</td>
<td>5. Supervised learning, given $\pi$</td>
</tr>
<tr>
<td>$P(x, \pi)$</td>
<td>$\pi^* = \arg\max_{\pi} P(x, \pi)$</td>
<td>$\Lambda^* = \arg\max_\Lambda P(x, \pi</td>
</tr>
<tr>
<td>Prob of a path, emissions</td>
<td>Most likely path</td>
<td>Viterbi training, best path</td>
</tr>
<tr>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>$P(x) = \sum_{\pi} P(x, \pi)$</td>
<td>$\pi^\Lambda = {\pi_i</td>
<td>\pi_i = \arg\max_k \sum_{\pi} P(\pi_i = k</td>
</tr>
<tr>
<td>Prob of emissions, over all paths</td>
<td>Path containing the most likely state at any time point.</td>
<td>Baum-Welch training, over all paths</td>
</tr>
</tbody>
</table>
## Examples of HMMs for genome annotation

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology / Transitions</td>
<td>2 states, different nucleotide composition</td>
<td>2 states, different conservation levels</td>
<td>2 states, different trinucleotide composition</td>
<td>2 states, different evolutionary signatures</td>
<td>~20 states, different composition/conservation, specific structure</td>
<td>40 states, different chromatin mark combinations</td>
</tr>
<tr>
<td>Hidden States / Annotation</td>
<td>GC-rich / AT-rich</td>
<td>Conserved / non-conserved</td>
<td>Coding exon / non-coding (intron or intergenic)</td>
<td>Coding exon / non-coding (intron or intergenic)</td>
<td>First/last/middle coding exon, UTRs, intron1/2/3, intergenic, *(+/- strand)</td>
<td>Enhancer / promoter / transcribed / repressed / repetitive</td>
</tr>
<tr>
<td>Emissions / Observations</td>
<td>Nucleotides</td>
<td>Level of conservation</td>
<td>Triplets of nucleotides</td>
<td>64x64 matrix of codon substitution frequencies</td>
<td>Codons, nucleotides, splice sites, start/stop codons</td>
<td>Vector of chromatin mark frequencies</td>
</tr>
</tbody>
</table>
What have we learned?

- **Modeling sequential data**
  - Recognize a *type* of sequence, genomic, oral, verbal, visual, etc…
- **Definitions**
  - Markov Chains
  - Hidden Markov Models (HMMs)
- **Examples of HMMs**
  - Recognizing GC-rich regions, preferentially-conserved elements, coding exons, protein-coding gene structures, chromatin states
- **Our first computations**
  - Running the model: know model $\rightarrow$ generate sequence of a ‘type’
  - Evaluation: know model, emissions, states $\rightarrow$ $p$?
  - Viterbi: know model, emissions $\rightarrow$ find optimal path
  - Forward: know model, emissions $\rightarrow$ total $p$ over all paths
- **Next time:**
  - Posterior decoding
  - Supervised learning
  - Unsupervised learning: Baum-Welch, Viterbi training