

---

# Contents

<b>List of Code Challenges</b>	<b>xvii</b>
<b>About the Textbook</b>	<b>xix</b>
Meet the Authors . . . . .	xix
Meet the Development Team . . . . .	xx
Acknowledgments . . . . .	xxi
<b>7 Which Animal Gave Us SARS?</b>	<b>2</b>
The Fastest Outbreak . . . . .	3
Trouble at the Metropole Hotel . . . . .	3
The evolution of SARS . . . . .	3
Transforming Distance Matrices into Evolutionary Trees . . . . .	5
Constructing a distance matrix from coronavirus genomes . . . . .	5
Evolutionary trees as graphs . . . . .	7
Distance-based phylogeny construction . . . . .	10
Toward An Algorithm for Distance-Based Phylogeny Construction . . . . .	12
A quest for neighboring leaves . . . . .	12
Computing limb lengths . . . . .	14
Additive Phylogeny . . . . .	17
Trimming the tree . . . . .	17
Attaching a limb . . . . .	19
An algorithm for distance-based phylogeny construction . . . . .	19
Constructing an evolutionary tree of coronaviruses . . . . .	20
Using Least Squares to Construct Approximate Distance-Based Phylogenies . . . . .	22
Ultrametric Evolutionary Trees . . . . .	23
The Neighbor-Joining Algorithm . . . . .	27
Transforming a distance matrix into a neighbor-joining matrix . . . . .	27

---

Analyzing coronaviruses with the neighbor-joining algorithm . . . . .	31
Limitations of distance-based approaches to evolutionary tree construction	33
Character-Based Tree Reconstruction . . . . .	33
Character tables . . . . .	33
From anatomical to genetic characters . . . . .	34
How many times has evolution invented insect wings? . . . . .	35
The Small Parsimony Problem . . . . .	37
The Large Parsimony Problem . . . . .	43
Epilogue: Evolutionary Trees Fight Crime . . . . .	48
Detours . . . . .	51
When did HIV jump from primates to humans? . . . . .	51
Searching for a tree fitting a distance matrix . . . . .	51
The four point condition . . . . .	52
Did bats give us SARS? . . . . .	54
Why does the neighbor-joining algorithm find neighboring leaves? . . . .	56
Computing limb lengths in the neighbor-joining algorithm . . . . .	61
Giant panda: bear or raccoon? . . . . .	62
Where did humans come from? . . . . .	62
Bibliography Notes . . . . .	66
<b>8 How Did Yeast Become a Wine Maker? . . . . .</b>	<b>68</b>
An Evolutionary History of Wine Making . . . . .	69
How long have we been addicted to alcohol? . . . . .	69
The diauxic shift . . . . .	70
Identifying Genes Responsible for the Diauxic Shift . . . . .	70
Two evolutionary hypotheses with different fates . . . . .	70
Which yeast genes drive the diauxic shift? . . . . .	71
Introduction to Clustering . . . . .	72
Gene expression analysis . . . . .	72
Clustering yeast genes . . . . .	74
The Good Clustering Principle . . . . .	76
Clustering as an Optimization Problem . . . . .	78
Farthest First Traversal . . . . .	79
<i>k</i> -Means Clustering . . . . .	82
Squared error distortion . . . . .	82
<i>k</i> -means clustering and the center of gravity . . . . .	83
The Lloyd Algorithm . . . . .	85

---

From centers to clusters and back again . . . . .	85
Initializing the Lloyd algorithm . . . . .	87
<i>k</i> -means++ Initializer . . . . .	88
Clustering Genes Implicated in the Diauxic Shift . . . . .	89
Limitations of <i>k</i> -Means Clustering . . . . .	90
From Coin Flipping to <i>k</i> -Means Clustering . . . . .	92
Flipping coins with unknown biases . . . . .	92
Where is the computational problem? . . . . .	95
From coin flipping to the Lloyd algorithm . . . . .	95
Return to clustering . . . . .	96
Making Soft Decisions in Coin Flipping . . . . .	97
Expectation maximization: the E-step . . . . .	97
Expectation maximization: the M-step . . . . .	99
The expectation maximization algorithm . . . . .	100
Soft <i>k</i> -Means Clustering . . . . .	100
Applying expectation maximization to clustering . . . . .	100
Centers to soft clusters . . . . .	101
Soft clusters to centers . . . . .	102
Hierarchical Clustering . . . . .	103
Introduction to distance-based clustering . . . . .	103
Inferring clusters from a tree . . . . .	106
Analyzing the diauxic shift with hierarchical clustering . . . . .	108
Epilogue: Clustering Tumor Samples . . . . .	109
Detours . . . . .	111
Whole genome duplication or a series of duplications? . . . . .	111
Measuring gene expression . . . . .	111
Microarrays . . . . .	112
Proof of the Center of Gravity Theorem . . . . .	113
Transforming an expression matrix into a distance/similarity matrix . . . . .	114
Clustering and corrupted cliques . . . . .	115
Bibliography Notes . . . . .	118
<b>9 How Do We Locate Disease-Causing Mutations? . . . . .</b>	<b>120</b>
What Causes Ohdo Syndrome? . . . . .	121
Introduction to Multiple Pattern Matching . . . . .	122
Herding Patterns into a Trie . . . . .	123
Constructing a trie . . . . .	123

---

Applying the trie to multiple pattern matching . . . . .	125
Preprocessing the Genome Instead . . . . .	127
Introduction to suffix tries . . . . .	127
Using suffix tries for pattern matching . . . . .	127
Suffix Trees . . . . .	131
Suffix Arrays . . . . .	133
Constructing a suffix array . . . . .	133
Pattern matching with the suffix array . . . . .	134
The Burrows-Wheeler Transform . . . . .	136
Genome compression . . . . .	136
Constructing the Burrows-Wheeler transform . . . . .	136
From repeats to runs . . . . .	138
Inverting the Burrows-Wheeler Transform . . . . .	139
A first attempt at inverting the Burrows-Wheeler transform . . . . .	139
The First-Last Property . . . . .	141
Using the First-Last property to invert the Burrows-Wheeler transform . . . . .	144
Pattern Matching with the Burrows-Wheeler Transform . . . . .	147
A first attempt at Burrows-Wheeler pattern matching . . . . .	147
Moving backward through a pattern . . . . .	148
The Last-to-First mapping . . . . .	150
Speeding Up Burrows-Wheeler Pattern Matching . . . . .	153
Substituting the Last-to-First mapping with count arrays . . . . .	153
Getting rid of the first column of the Burrows-Wheeler matrix . . . . .	154
Where are the Matched Patterns? . . . . .	156
Burrows and Wheeler Set Up Checkpoints . . . . .	157
Epilogue: Mismatch-Tolerant Read Mapping . . . . .	159
Reducing approximate pattern matching to exact pattern matching . . . . .	159
BLAST: Comparing a sequence against a database . . . . .	160
Approximate pattern matching with the Burrows-Wheeler transform . . . . .	162
Charging Stations . . . . .	164
Constructing a Suffix Tree . . . . .	164
Solving the Longest Shared Substring Problem . . . . .	167
Partial Suffix Array Construction . . . . .	169
Detours . . . . .	170
The reference human genome . . . . .	170
Rearrangements, insertions, and deletions in human genomes . . . . .	170
The Aho-Corasick algorithm . . . . .	170

---

From suffix trees to suffix arrays . . . . .	171
From suffix arrays to suffix trees . . . . .	173
Binary search . . . . .	176
Bibliography Notes . . . . .	177
<b>10 Why Have Biologists Still Not Developed an HIV Vaccine?</b>	<b>178</b>
Classifying the HIV Phenotype . . . . .	179
How does HIV evade the human immune system? . . . . .	179
Limitations of sequence alignment . . . . .	181
Gambling with Yakuza . . . . .	182
Two Coins up the Dealer's Sleeve . . . . .	184
Finding CG-Islands . . . . .	185
Hidden Markov Models . . . . .	186
From coin flipping to a Hidden Markov Model . . . . .	186
The HMM diagram . . . . .	188
Reformulating the Casino Problem . . . . .	188
The Decoding Problem . . . . .	191
The Viterbi graph . . . . .	191
The Viterbi algorithm . . . . .	194
How fast is the Viterbi algorithm? . . . . .	195
Finding the Most Likely Outcome of an HMM . . . . .	196
Profile HMMs for Sequence Alignment . . . . .	198
How do HMMs relate to sequence alignment? . . . . .	198
Building a profile HMM . . . . .	201
Transition and emission probabilities of a profile HMM . . . . .	203
Classifying proteins with profile HMMs . . . . .	207
Aligning a protein against a profile HMM . . . . .	207
The return of pseudocounts . . . . .	208
The troublesome silent states . . . . .	209
Are profile HMMs really all that useful? . . . . .	216
Learning the Parameters of an HMM . . . . .	217
Estimating HMM parameters when the hidden path is known . . . . .	217
Viterbi learning . . . . .	219
Soft Decisions in Parameter Estimation . . . . .	221
The Soft Decoding Problem . . . . .	221
The forward-backward algorithm . . . . .	222
Baum-Welch Learning . . . . .	225

---

The Many Faces of HMMs . . . . .	227
Epilogue: Nature is a Tinkerer and not an Inventor . . . . .	227
Detours . . . . .	229
The Red Queen Effect . . . . .	229
Glycosylation . . . . .	229
DNA methylation . . . . .	229
Conditional probability . . . . .	230
Bibliography Notes . . . . .	232
<b>11 Was <i>T. rex</i> Just a Big Chicken?</b> . . . . .	<b>234</b>
Paleontology Meets Computing . . . . .	235
Which Proteins Are Present in This Sample? . . . . .	236
Decoding an Ideal Spectrum . . . . .	237
From Ideal to Real Spectra . . . . .	241
Peptide Sequencing . . . . .	244
Scoring peptides against spectra . . . . .	244
Where are the suffix peptides? . . . . .	246
Peptide sequencing algorithm . . . . .	248
Peptide Identification . . . . .	249
The Peptide Identification Problem . . . . .	249
Identifying peptides in the unknown <i>T. rex</i> proteome . . . . .	250
Searching for peptide-spectrum matches . . . . .	251
Peptide Identification and the Infinite Monkey Theorem . . . . .	252
False discovery rate . . . . .	252
The monkey and the typewriter . . . . .	254
Statistical significance of a peptide-spectrum match . . . . .	255
Spectral Dictionaries . . . . .	258
<i>T. rex</i> Peptides: Contaminants or Treasure Trove of Ancient Proteins? . . . . .	261
The hemoglobin riddle . . . . .	261
The dinosaur DNA controversy . . . . .	264
Epilogue: From Unmodified to Modified Peptides . . . . .	264
Post-translational modifications . . . . .	264
Searching for modifications as an alignment problem . . . . .	265
Building a Manhattan grid for spectral alignment . . . . .	267
Spectral alignment algorithm . . . . .	271
Detours . . . . .	273
Gene prediction . . . . .	273

---

Finding all paths in a graph . . . . .	275
The Anti-Symmetric Path Problem . . . . .	275
Transforming spectra into spectral vectors . . . . .	276
The infinite monkey theorem . . . . .	277
The probabilistic space of peptides in a spectral dictionary . . . . .	278
Are terrestrial dinosaurs really the ancestors of birds? . . . . .	279
Solving the Most Likely Peptide Vector Problem . . . . .	280
Selecting Parameters for Transforming Spectra into Spectral Vectors . . .	281
Bibliography Notes . . . . .	283
<b>Bibliography</b>	<b>285</b>
<b>Image Courtesies</b>	<b>291</b>

---

## List of Code Challenges

<b>Chapter 7</b>	<b>2</b>
(7A) Compute Distances Between Leaves . . . . .	11
(7B) Compute Limb Lengths in a Tree . . . . .	17
(7C) Implement <b>ADDITIVEPHYLOGENY</b> . . . . .	20
(7D) Implement <b>UPGMA</b> . . . . .	25
(7E) Implement <b>NEIGHBORJOINING</b> . . . . .	30
(7F) Implement <b>SMALLPARSIMONY</b> . . . . .	40
(7G) Adapt <b>SMALLPARSIMONY</b> to Unrooted Trees . . . . .	41
(7H) Find the Nearest Neighbors of a Tree . . . . .	45
(7I) Implement <b>NEARESTNEIGHBORINTERCHANGE</b> . . . . .	47
<b>Chapter 8</b>	<b>68</b>
(8A) Implement <b>FARTHESTFIRSTTRAVERSAL</b> . . . . .	80
(8B) Compute the Squared Error Distortion . . . . .	82
(8C) Implement the Lloyd Algorithm for $k$ -Means Clustering . . . . .	85
(8D) Implement the Soft $k$ -Means Clustering Algorithm. . . . .	103
(8E) Implement <b>HIERARCHICALCLUSTERING</b> . . . . .	106
<b>Chapter 9</b>	<b>120</b>
(9A) Construct a Trie from a Collection of Patterns . . . . .	124
(9B) Implement <b>TRIEMATCHING</b> . . . . .	126
(9C) Construct the Suffix Tree of a String . . . . .	132
(9D) Find the Longest Repeat in a String . . . . .	132
(9E) Find the Longest Substring Shared by Two Strings . . . . .	133
(9F) Find the Shortest Non-Shared Substring of Two Strings . . . . .	133
(9G) Construct the Suffix Array of a String . . . . .	133
(9H) Implement <b>PATTERNMATCHINGWITHSUFFIXARRAY</b> . . . . .	135



---

(9I) Construct the Burrows-Wheeler Transform of a String . . . . .	138
(9J) Reconstruct a String from its Burrows-Wheeler Transform . . . . .	147
(9K) Generate the Last-to-First Mapping of a String . . . . .	151
(9L) Implement <b>BWMATCHING</b> . . . . .	151
(9M) Implement <b>BETTERBWMATCHING</b> . . . . .	156
(9N) Find All Occurrences of a Collection of Patterns in a String . . . . .	158
(9O) Find All Approximate Occurrences of a Collection of Patterns in a String . . . . .	162
(9P) Implement <b>TREECOLORING</b> . . . . .	168
(9Q) Construct the Partial Suffix Array of a String . . . . .	169
(9R) Construct a Suffix Tree from a Suffix Array . . . . .	172
<b>Chapter 10</b>	<b>178</b>
(10A) Compute the Probability of a Hidden Path . . . . .	190
(10B) Compute the Probability of an Outcome Given a Hidden Path . . . . .	191
(10C) Implement the Viterbi Algorithm . . . . .	195
(10D) Compute the Probability of a String Emitted by an HMM . . . . .	197
(10E) Construct a Profile HMM . . . . .	206
(10F) Construct a Profile HMM with Pseudocounts . . . . .	209
(10G) Perform a Multiple Sequence Alignment with a Profile HMM . . . . .	212
(10H) Estimate the Parameters of an HMM . . . . .	219
(10I) Implement Viterbi Learning . . . . .	220
(10J) Solve the Soft Decoding Problem . . . . .	223
(10K) Implement Baum-Welch Learning . . . . .	226
<b>Chapter 11</b>	<b>234</b>
(11A) Construct the Graph of a Spectrum . . . . .	239
(11B) Implement <b>DECODINGIDEALSPECTRUM</b> . . . . .	240
(11C) Convert a Peptide into a Peptide Vector . . . . .	245
(11D) Convert a Peptide Vector into a Peptide . . . . .	245
(11E) Sequence a Peptide . . . . .	249
(11F) Find a Highest-Scoring Peptide in a Proteome against a Spectrum . . . . .	250
(11G) Implement <b>PSMSEARCH</b> . . . . .	252
(11H) Compute the Size of a Spectral Dictionary . . . . .	258
(11I) Compute the Probability of a Spectral Dictionary . . . . .	260
(11J) Find a Highest-Scoring Modified Peptide against a Spectrum . . . . .	273