

Contents

Preface	vii
List of contributors	xi
Other volumes in the series	xxv
 I – Introduction and Tutorial Background	
 <i>Chapter 1. Grand challenges in computational biology</i>	
<i>David B. Searls</i>	3
1. Introduction	3
2. Protein structure prediction	4
3. Homology search	5
4. Multiple alignment and phylogeny construction	6
5. Genomic sequence analysis and gene-finding	7
6. Conclusion	8
References	9
 <i>Chapter 2. A tutorial introduction to computation for biologists</i>	
<i>Steven L. Salzberg</i>	11
1. Who should read the tutorials?	11
2. Basic computational concepts	11
2.1. What is an algorithm?	12
2.2. How fast is a program?	13
2.3. Computing time is a function of input size	14
2.4. Space requirements also vary with input size	15
2.5. Really expensive computations	15
3. Machine learning concepts	16
3.1. Learning from data	16
3.2. Memory-based reasoning	17
4. Where to store learned knowledge	17
4.1. Decision trees	18
4.2. Neural networks	19
5. Search	19
5.1. Defining a search space	20
5.2. Search space size	20
5.3. Tree-based search	21
6. Dynamic programming	22
7. Basic statistics and Markov chains	24

8. Conclusion	26
9. Recommended introductory texts	26

Chapter 3. An introduction to biological sequence analysis

<i>Kenneth H. Fasman and Steven L. Salzberg</i>	29
1. Introduction	29
2. A little molecular biology	29
3. Frequency analysis	30
4. Measuring homology by pairwise alignment	31
5. Database searching	33
6. Multiple alignment	33
7. Finding regions of interest in nucleic acid sequence	34
8. Gene finding	35
9. Analyzing protein sequence	38
10. Whole genome analysis	39
Acknowledgments	40
References	40

II – Learning and Pattern Discovery in Sequence Databases

Chapter 4. An introduction to hidden Markov models for biological sequences

<i>Anders Krogh</i>	45
1. Introduction	45
2. From regular expressions to HMMs	46
3. Profile HMMs	49
3.1. Pseudocounts	50
3.2. Searching a database	54
3.3. Model estimation	55
4. HMMs for gene finding	56
4.1. Signal sensors	57
4.2. Coding regions	58
4.3. Combining the models	59
5. Further reading	60
Acknowledgments	62
References	62

Chapter 5. Case-based reasoning driven gene annotation

<i>G. Christian Overton and Juergen Haas</i>	65
1. Introduction	65
2. Case-based reasoning	66
3. CBR in biology	67
4. Annotating gene and promoter regions by CBR	68
5. Building the case-base	70
5.1. Data cleansing with the sequence structure parser	70
5.2. Structure and content	73
6. Classification of new cases	75

7. Case adaptation	76
7.1. Generating instance grammars	78
7.2. Predicting exon/intron structure	79
7.3. Predicting regulatory elements	80
7.4. Results	81
8. Database issues	83
9. Conclusions	84
Acknowledgments	85
References	85

Chapter 6. Classification-based molecular sequence analysis

<i>David J. States and William C. Reisdorf, Jr.</i>	87
1. Introduction	87
1.1. Molecular evolution	88
1.2. Sequence annotation	89
1.3. Artifacts due to sequence and database errors	91
1.4. Sequence classification	91
1.5. Sequence similarity search	92
2. A taxonomy of existing classifications	93
2.1. Domains versus molecules	93
2.2. Sequence versus structure	94
2.3. Clustering methods	94
2.4. Hierarchic versus atomic classifications	95
2.5. Manual, supervised and fully automated	95
3. Databases of protein sequence/structure classification	95
3.1. Nucleic acid sequence classifications	99
3.2. Tertiary-structure-based classifications	99
3.3. Assessing classifications	102
Acknowledgments	103
References	103

Chapter 7. Computational gene prediction using neural networks and similarity search

<i>Ying Xu and Edward C. Uberbacher</i>	109
1. Introduction	109
2. Exon prediction by neural networks	110
2.1. Coding region recognition	111
2.2. Splice junction recognition	113
2.3. Information fusion	114
3. Predicting coding regions in erroneous DNA	117
3.1. Localization of transition points	117
4. Reference-based gene structure prediction	120
4.1. EST-based reference model construction	120
4.2. Reference-based gene modeling	121
5. Conclusion	127
References	127

Chapter 8. Modeling dependencies in pre-mRNA splicing signals

<i>Christopher B. Burge</i>	129
List of abbreviations	129
1. Introduction	129
2. Review of pre-mRNA splicing	130
3. Signal prediction	132
4. Probabilistic models of signal sequences	133
4.1. The directionality of Markov models	135
5. Datasets	136
6. Weight matrix models and generalizations	136
6.1. Positional odds ratios	140
6.2. The branch point region	143
6.3. Parameter estimation error	144
6.4. A windowed weight array model	145
7. Measuring dependencies between positions	147
7.1. Dependencies in acceptor splice signals	149
7.2. Dependence structure of the donor splice signal	153
8. Modeling complex dependencies in signals	155
8.1. Maximal dependence decomposition	157
9. Conclusions and further reading	162
Acknowledgments	163
References	163

Chapter 9. Evolutionary approaches to computational biology

<i>Rebecca J. Parsons</i>	165
1. Introduction	165
1.1. Optimization problems	166
2. Evolutionary computation	167
2.1. Landscape of evolutionary approaches	167
2.1.1. Genetic algorithms	168
2.1.2. Evolutionary strategies	170
2.1.3. Genetic programming	171
2.2. A primer on genetic algorithms	171
2.2.1. Fitness function	172
2.2.2. Representation	172
2.2.3. Genetic operators and selection mechanism	172
2.2.4. Control parameters	173
2.3. Putting the pieces of the genetic algorithm together	173
3. Assembling the DNA sequence jigsaw puzzle	174
4. Design for DNA sequence assembly	175
4.1. Choosing the right representation	176
4.2. Permutation-specific operators	178
5. Problem-specific operators	179
5.1. How to set the parameters	181
6. Good versus best answers	182
7. Future directions	183
8. Conclusions	183
References	184

<i>Chapter 10. Decision trees and Markov chains for gene finding</i>	
<i>Steven L. Salzberg</i>	187
Introduction	187
1. A tutorial introduction to decision trees	187
1.1. Induction of decision trees	188
1.2. Splitting rules	189
1.3. Pruning rules	190
1.4. Internet resources for decision tree software	191
2. Decision trees to classify sequences	192
2.1. Coding measures	192
3. Decision trees as probability estimators	193
4. MORGAN, a decision tree system for gene finding	194
4.1. The MORGAN framework	195
4.2. Markov chains to find splice sites	195
4.3. Parsing DNA with dynamic programming	196
4.4. Frame consistent dynamic programming	198
4.5. Downstream sequence	198
5. Data and experiments	199
6. Next steps: interpolated Markov models	201
7. Summary	202
Acknowledgments	202
References	202

III – Protein Structure Modeling and Prediction

Chapter 11. Statistical analysis of protein structures. Using environmental features for multiple purposes

<i>Liping Wei, Jeffrey T. Chang and Russ B. Altman</i>	207
List of Abbreviations	207
1. Protein structures in three dimensions	207
2. Statistics, probability, and machine learning concepts	209
2.1. Statistical inference from two sets of data	209
2.2. Classification problems and evaluation of classification algorithms	210
2.3. Conditional probability and Bayes' Rule	211
3. Statistical description of protein structures	212
4. Applications of statistical analysis of protein structures	213
4.1. Characterizing the microenvironment surrounding protein sites	214
4.2. Recognizing protein sites using environmental statistics	216
4.3. Comparing the environments of amino acids and constructing a substitution matrix	219
4.4. Threading a protein sequence onto a fold	221
4.4.1. Library of folds	222
4.4.2. Alignments	223
4.4.3. Generating and scoring actual and sample structures	223
5. Summary	224
Acknowledgments	224
References	225

Chapter 12. Analysis and algorithms for protein sequence-structure alignment
Richard H. Lathrop, Robert G. Rogers Jr., Jadwiga Bienkowska, Barbara K.M. Bryant, Ljubomir J. Buturović, Chrysanthé Gaitatzes, Raman Nambudripad, James V. White and Temple F. Smith

227

Chapter overview	227
1. Introduction	227
1.1. Why is it hard?	228
1.2. Why threading?	229
2. Protein threading – motivating intuitions	230
2.1. Basic ideas	230
2.2. Common assumptions of current threading work	231
2.3. Principal requirements for structure prediction	231
2.3.1. Library members (requirement i)	233
2.3.2. Objective function (requirement ii)	235
2.3.3. Alignment (requirement iii)	235
2.3.4. Core template selection or fold recognition (requirement iv)	236
2.4. Gapped block alignment	236
3. Formalization	237
3.1. Sequence	239
3.2. Core templates and library	239
3.3. Loops	240
3.4. Adjacency graph	240
3.5. A threading of a sequence into a core	240
3.6. Sets of threadings	241
3.7. Objective function	242
3.7.1. The fully general objective function	242
3.7.2. General pairwise interaction objective function	243
3.7.3. A typical pairwise score function	243
3.7.4. Computing g_1 and g_2 efficiently	244
3.7.5. Singleton-only objective function	244
3.7.6. Per-residue singleton-only objective function	245
4. Analysis – selection tasks	246
4.1. Bayesian analysis	247
4.1.1. Prior probabilities	247
4.1.2. Global sums	248
4.2. Selecting an alignment given a core template	248
4.3. Selecting a core template	249
4.4. Selecting structure and alignment jointly	249
4.5. Selecting individual core segment alignments	250
4.6. Super-secondary structures, or core template subsets	250
4.7. Secondary structure prediction	251
4.8. Variable $Z_{(n,C,t)}$	252
4.9. Recurrence equations for singleton-only objective functions	252
4.9.1. Equations for $Z_{(n,C,t)}^1$	253
4.9.2. Recurrence equations for $\mu_{(a,n,C)}^1$	253
4.9.3. Recurrence equations for $\mu_{(a,n,C,J,t)}^1$	253
4.9.4. Recurrence equations for $\mu_{(a,n,C,J,T)}^1$	254
4.9.5. Recurrence equation invariants	255
4.10. Recurrence equations for per-residue singleton-only objective functions	255
5. Analysis – search space tasks	256

5.1. Lower bound on scores in threading sets	256
5.1.1. Lower bound invariants	257
5.2. Splitting threading sets	257
6. Analysis – search space formulae	259
6.1. Fast approximate formulae	259
6.1.1. Computing P_1 and P_2 efficiently	260
6.2. Exact search space size, probabilities, and uniform sampling	260
6.2.1. Search space size	260
6.2.2. Exact segment placement probabilities	261
6.2.3. Uniform random sampling	261
6.3. Exact analytic search space mean and standard deviation	261
6.4. Computing the exact analytic mean and standard deviation efficiently	262
7. Computational complexity	262
7.1. Computational complexity and NP-completeness	263
7.2. Alignment – informal sketch of proof	265
7.3. Selection tasks and Bayes' constants	267
7.3.1. Complexity of $\mu_{(a,n,C)}$	267
7.3.2. Complexity of $Z_{(n,C,k)}$	267
7.3.3. Complexity of $\mu_{(a,n,C,J,I)}$ and $\mu_{(a,n,C,J,T)}$	267
8. Search algorithm	267
8.1. Branch and bound algorithm	268
8.2. Algorithm pseudo-code	269
8.3. Lower bound implementation	269
8.3.1. Implementation	269
8.3.2. Computing the lower bound efficiently	271
9. Computational experiments	272
9.1. Core template library	272
9.2. Problem size and computation time	273
10. Discussion	277
10.1. Scoring schemes	278
11. Conclusions	279
Acknowledgments	280
References	280

Chapter 13. THREADER: protein sequence threading by double dynamic programming

David Jones	285
1. Introduction	285
2. A limited number of folds	287
3. The fold library	288
4. The modelling process	289
5. Evaluating the models	289
6. Statistically derived pairwise potentials	290
6.1. Ponder and Richards (1987)	291
6.2. Bowie et al. (1990)	291
6.3. Bowie et al. (1991)	292
6.4. Finkelstein and Reva (1991)	292
7. Optimal sequence threading	293
8. Formulating a model evaluation function	295
9. Calculation of potentials	301

10. Searching for the optimal threading	302
11. Methods for combinatorial optimization	303
11.1. Exhaustive search	303
11.2. Monte Carlo methods	303
11.3. Dynamic programming	304
11.4. Double dynamic programming	305
12. Residue selection for sequence-structure alignments	308
13. Evaluating the method	309
14. Software availability	310
References	310

Chapter 14. From computer vision to protein structure and association

<i>Haim J. Wolfson and Ruth Nussinov</i>	313
--	-----

1. Introduction	313
2. Problem formulation	315
3. Molecular surface representation and interest feature extraction	315
3.1. Caps, pits, and belts	316
3.2. Knobs and holes	317
4. Interest point correspondence	317
5. Large protein-protein docking	319
5.1. The algorithm	319
5.1.1. Knob and hole extraction	320
5.1.2. Interest point matching	320
5.1.3. Clustering of transformations	320
5.1.4. Verification of penetration and scoring	321
5.2. Experimental results	321
6. The Geometric-Hashing-based docking method	324
6.1. The algorithm	324
6.1.1. Reference frame definition	325
6.1.2. Ligand preprocessing	326
6.1.3. Recognition	326
6.1.4. Implementation details	328
6.1.5. Choice of reference sets	328
6.1.5.1. Distance constraint	328
6.1.5.2. Directional constraints	328
6.1.6. Choice of points belonging to a reference set	328
6.1.7. Voting constraints	329
6.1.8. Aligning the receptor with the ligand	329
6.1.9. Improvement of the transformation	329
6.1.10. Verification of penetration and contact scoring	329
6.2. Experimental results	330
7. Further developments and future tasks	332
Acknowledgments	333
References	333

Chapter 15. Modeling biological data and structure with probabilistic networks

<i>Simon Kasif and Arthur L. Delcher</i>	335
--	-----

1. Introduction	335
-----------------	-----

2. Probabilistic networks	336
3. Using chain graphs in biology	337
4. Modeling hidden state systems with probabilistic networks	339
5. Tree networks	341
6. Probabilistic networks: learning and inference	342
7. Application of probabilistic networks for protein secondary structure prediction	343
8. A probabilistic framework for protein analysis	344
9. Protein modeling using causal networks	344
10. Protein modeling using the Viterbi algorithm	346
11. Towards automated site-specific mutagenesis	348
12. Using the EM algorithm	348
13. Discussion	350
References	352

IV – Reference Materials

<i>Appendix A. Software and databases for computational biology on the Internet</i>	355
1. Sequence Analysis Programs	355
2. Databases	358
3. Other software and information sources	360
<i>Appendix B. Suggestions for further reading in computational biology</i>	361
Index	367