Book review:
Sequence Analysis Primer
by Gribskov and Devereux*

John Kececioglu†

30 September 1992

Sequence Analysis Primer. Michael Gribskov and John Devereux, editors. Stockton Press, New York, 1991. xv + 279 pp. ISBN 1-56159-007-X (paper).

The last few years have seen a surge of interest in computational molecular biology, with massive efforts such as the Human Genome Project underway. *Sequence Analysis Primer*, edited by Michael Gribskov and John Devereux, is a welcome introduction to computer analysis of biological sequence data. Drs. Gribskov and Devereux are well-known for the Genetics Computer Group sequence analysis package; their volume is not a manual on the GCG package, though it naturally includes many examples from their system. Their stated goal is to explain what types of analysis are available, what their limitations are, and what can or cannot be concluded from the analysis. In this the book succeeds admirably. Contributors have written in a friendly style, and offer plenty of advice to the reader. Perhaps the most appealing aspect of the book is its emphasis on examples drawn from actual use of programs. The final chapter, for instance, is an extended example applying nearly all the preceding material to analysis of a particular sequence, the *Drosophila Notch* gene.

The book in large measure is directed towards biologists. Readers of this review are likely to be applied mathematicians, statisticians, and computer scientists, hence we try to indicate some of the research problems, and point to some of the more recent work.

The book is divided into four chapters, and the first three parallel the steps a biologist might take in analyzing a new sequence. Chapter 1, "DNA" by Peter Rice, Keith Elliston, and Michael Gribskov, covers sequence assembly (assembly of complete DNA sequences from experimentally determined fragments), identification of simple sites and transcriptional signals (patterns governing the translation of DNA into protein), coding region identification (identifying the regions of DNA that translate into protein), and RNA and DNA structure prediction (predicting the topological structure of folded RNA and DNA). The material on sequence assembly focuses on shotgun data, as it arguably "represents the most difficult problems in sequence assembly," but mainly discusses project management issues. The material on identifying sites and signals concentrates largely on patterns derived from alignments of

*This book review appeared in *Journal of Classification* 10, 144-148, 1993.

†Computer Science Department, University of California at Davis, Davis, California 95616. Electronic mail: `kece@cs.ucdavis.edu`.

instances of the site or signal; the alignment is reduced to a *weight matrix*, which is essentially a sequence whose positions are distributions over the alphabet. Material on coding region identification highlights both signal methods (methods that identify coding regions by finding delimiting signals), and statistical methods (methods that assess sequence content for statistical similarity to coding regions). The chapter contains many useful tables, including listings of sequence assembly programs by machine and vendor, bacterial and eukaryotic promoter patterns, and many signals described by weight matrices, such as polyadenylation signals, splice acceptor and donor sites, and ribosomal binding sites.

Chapter 2, "Protein" by Roland Lüthy and David Eisenberg, surveys protein secondary structure prediction (classifying the amino acids or *residues* of a protein into α-helix, β-strand, and turn structures), hydrophobicity plots for identifying globular, water-soluble, and membrane proteins (classifying proteins by the attraction or repulsion of residues to water), and patterns for identifying short protein sequences with known three-dimensional structure, such as zinc fingers and dinucleotide binding sites.

Chapter 3, "Similarity and Homology" by David States and Mark Boguski, covers the basics of sequence comparison: dot plots, alignment by dynamic programming, alignment scoring matrices, database searching, multiple sequence alignment, and statistical evaluation of alignments. *Dot plots* are an exploratory data analysis tool for visualizing matches between two sequences. A dot is placed at position (x, y) in the plane if the sequences match above a given threshold in a window around position x in the first sequence, and position y in the second sequence. Patterns of dots forming 45 degree lines indicate matching substrings. The chapter gives highly detailed case studies applying dot plots to finding internal repeats in haptoglobins, apolipoproteins, and cell division cycle gene products; certainly the case is made that dot plots are used extensively by biologists. The material on dynamic programming derives the basic recurrence for optimally aligning two sequences with insertion, deletion, and substitution costs, and discusses affine gap penalities and local versus global alignment. The chapter also explains how the standard Point Accepted Mutation scoring matrices (PAM matrices) were obtained, and does a good job of pointing out the limitations of their derivation. Many of the sequence comparison tools developed with NIH collaborators are described, including FASTA [16], BLAST [1], and BLAST3 [2] for database searching, and MSA [13] and MACAW [17] for multiple sequence alignment.

Chapter 4, "Practical Aspects" by Lisa Caballero, is an excellent illustration of the methods presented in the preceding three chapters. The goal is "to show the process of sequence analysis through examples ... reproducing the steps a researcher might take in analyzing a newly determined sequence." The treatment follows the organization of the book: sequencing project management, followed by coding region identification, and so on. The *Notch* sequence of *Drosophila* is analyzed, which encodes a large transmembrane protein believed to control cell development. Care was taken to make the chapter self-contained. While this means some material is necessarily duplicated, biologists who wish to quickly learn the basics can choose to read this chapter alone.

Ten appendices follow, including a partial listing of software suppliers (providing addresses, phone numbers, contact persons, and target machines), a listing of sequence and structural databases (describing their content, distributional media, and providing an electronic mail contact), and a glossary. One appendix gives considerations on what type of machine to buy (personal computer versus workstation versus mainframe computer), and minimum system configurations. Another appendix explains electronic mail and the network file transfer protocol `ftp`; it lists fifteen `ftp` sites and ten electronic mail information servers.

Overall a lot of work went into preparing the numerous examples that illustrate the methods. The authors clearly know the available software, and offer a wealth of advice for avoiding common pitfalls. In such a book it is difficult to achieve the right balance between describing methods and simply explaining the idiosyncracies of a program, and on the whole they achieve a fair balance. They are extremely thorough in documenting exactly which program, on what machine, with what parameters, produced the illustrations. And the book is up-to-date in describing newer software such as BLAST, MSA, and MACAW.

Reading the book, however, one is struck by the many quantitative analyses performed without any mathematical model or problem statement. Without a clear definition of what a program computes, one is unfortunately reduced to repeating, as are the authors, "results must be used with caution." (Indeed, part of the utility of the volume comes from the fact that experience is required to work around shortcomings of the programs and interpret their output.)

For example, in discussing DNA sequence assembly the book never states what the problem is. Current research models sequence assembly as the problem of finding a shortest sequence such that every fragment matches a substring within the experimental error rate—and we now have algorithms that deliver assemblies that in this sense are provably close to optimal [11].

In database searching, none of the programs described can guarantee that every sequence in the database within a given edit distance of the query sequence will be reported to the user. Subsequent algorithms satisfy this property while retaining fast search times [5, 14].

In identifying sites and signals, the classes of patterns discussed are consensus sequences and weight matrices. Both are derived from alignments of positive examples, and both essentially reduce the pattern to one sequence. Consensus sequences tend toward false negatives because a consensus does not capture the full range of instances, and weight matrices tend toward false positives because correlations between characters in the example sequences are lost. An approach that may avoid these difficulties is to (1) represent the examples by a compressed *regular expression*, which does not lose information, and (2) search for the pattern with efficient algorithms for comparing sequences to regular expressions [15].

In aligning sequences with affine gap penalties, the authors point out the difficulty of choosing gap penalty parameters. Chapter 4 demonstrates this with a comparison of the *Notch* and *Xotch* sequences, for which it appears no choice of parameters gives the correct alignment. Recent work on parametric sequence alignment has shown it is possible to efficiently compute a decomposition of the parameter space into convex regions such that any alignment optimal in a region is optimal for all choices of parameters within the region [8]. Computing such a decomposition for the *Notch* and *Xotch* sequences may resolve the gap penalty quandry.

Reading the enthusiastic discussion of dot plots, one cannot help but appreciate their appeal. What it seems the authors want, however, is a tool that can display all significant local similarities between very long sequences. The LAV local alignment viewer [18] coupled with a space efficient algorithm for local similarity [10] may provide such a tool.

Finally, the book does not cover phylogenetic trees, which may be the area of greatest interest to *Journal of Classification* readers.[1]

[1] The book however does give `ftp` and email addresses for PHYLIP, one of the most widely-used phylogeny packages.

Of the handful of books on biosequence analysis that have appeared in the last five years [3, 4, 6, 7, 9, 12], Gribskov and Devereux's primer is likely to be the one that will become essential for the biologist. For computer scientists and applied mathematicians, it gives a good idea of what biologists currently use, and a feel for what they still need. In tone it is similar to Doolittle's monograph [6], updated for the "current generation" of software. As pointed out above, many of the algorithms have already been superseded by new results. This simply means that it should be exciting to see the primer that emerges after the next five years. Perhaps in that time implementations of some of the newer results will have found their way into the standard packages, and become the "next generation" of sequence analysis tools.

References

[1] Altschul, S.F., W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology* 215, 403–410, 1990.

[2] Altschul, S.F. and D.J. Lipman. Protein database searches for multiple alignments. *Proceedings of the National Academy of Sciences USA* 87, 5509–5513, 1990.

[3] Bell, George I. and Thomas G. Marr, editors. *Computers and DNA*. Santa Fe Institute Studies in the Sciences of Complexity, Proceedings Volume 7. Addison-Wesley, Redwood City, California, 1990.

[4] Bishop, M.J. and C.J. Rawlings, editors. *Nucleic Acid and Protein Sequence Analysis: A Practical Approach*. IRL Press, Oxford, 1987.

[5] Chang, William I. and Eugene L. Lawler. Sublinear expected time approximate string matching and biological applications. Technical Report 91-654, Computer Science Division, University of California at Berkeley, Berkeley, California 94720, 1991. To appear in *Algorithmica*.

[6] Doolittle, Russell F. *Of URFs and ORFs: A Primer on How to Analyze Derived Amino Acid Sequences*. University Science Books, Mill Valley, California, 1986.

[7] Doolittle, Russell F., editor. *Molecular Evolution: Computer Analysis of Protein and Nucleic Acid Sequences*. Methods in Enzymology, Volume 183. Academic Press, New York, 1990.

[8] Gusfield, D., K. Balasubramanian, and D. Naor. Parametric optimization of sequence alignment. In Proceedings of the 3rd Annual ACM-SIAM *Symposium on Discrete Algorithms*, 432–439, 1991. To appear in *Algorithmica*.

[9] von Heijne, Gunnar. *Sequence Analysis in Molecular Biology: Treasure Trove or Trivial Pursuit*. Academic Press, San Diego, 1987.

[10] Huang, Xiaoqiu, Webb Miller, Scott Schwartz, and Ross Hardison. Parallelization of a local similarity algorithm. *Computer Applications in the Biosciences* 8, 155–165, 1992.

[11] Kececioglu, John. *Exact and Approximation Algorithms for DNA Sequence Reconstruction*. PhD dissertation, Technical Report 91-26, Department of Computer Science, The University of Arizona, Tucson, Arizona 85721, 1991.

[12] Lesk, Arthur M., editor. *Computational Molecular Biology: Sources and Methods for Sequence Analysis*. Oxford University Press, Oxford, 1988.

[13] Lipman, D.J., S.F. Altschul, and J.D. Kececioglu. A tool for multiple sequence alignment. *Proceedings of the National Academy of Sciences* 86, 4412–4415, 1989.

[14] Myers, Eugene. A sublinear algorithm for approximate keyword matching. Technical Report 90-25, Department of Computer Science, The University of Arizona, Tucson, Arizona 85721, 1990. To appear in *Algorithmica*.

[15] Myers, Eugene. Approximate matching of network expressions with spacers. Technical Report 92-5, Department of Computer Science, The University of Arizona, Tucson, Arizona 85721, 1992. Submitted to the *Bulletin of Mathematical Biology*.

[16] Pearson, W.R. and D.J. Lipman. Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences USA* 85, 2444–2448, 1988.

[17] Schuler, G.D., S.F. Altschul, and D.J. Lipman. A workbench for multiple alignment construction and analysis. *Proteins: Structure, Function, and Genetics* 9, 180–190, 1991.

[18] Schwartz, S., W. Miller, C.-M. Yang, and R.C. Hardison. Software tools for analyzing pairwise alignments of long sequences. *Nucleic Acids Research* 19, 4663–4667, 1991.