Some Physical Problems in Bioinformatics

Eric D. Siggia*
Center for Studies in Physics and Biology
Rockefeller University
1230 York Ave, New York, NY 10021

March 20, 2005

Abstract

Bioinformatics is a data driven field, in which a significant number of problems require statistical modeling. The flood of data emerging from genome centers uses sequence comparison to delimit and assign function to genes, and in very limited ways infers gene control from approximately repeated sequence motifs near to the genes themselves. Traditional topics in computer science such as coding theory, natural language processing, and old fashioned cryptography all impinge on the problem of deducing regulatory information from the genome, but are not probabalistic enough to cope with the fuzziness of biological patterns. The means by which living things encode information is a problem common to both neural biology and the regulation of gene expression by the genome. Physical analogies are employed to highlight some of the problems and opportunities in this area.

1 Introduction

Biological sequence data is growing at a faster rate than Moore's law. Sequencing is now an industrial enterprise carried out by robots and venture capitalists with not a graduate students in sight. Biologists flock to lectures

*The authors work has been supported for many years by the Division of Materials Research of the National Science Foundation.

with titles such as “Drowing in data, thirsty for knowledge” (S. Brenner Rockefeller 2001) hoping to learn what the genome teaches us about the large scale organization of life. While there is no question that an organism’s genome is of immediate utility to experiments targeting individual genes, and the comparison of genomes provides glimpses into the evolution of homologous genes, there is nothing immediately evident in the genome about how all the genes are coordinated. The various celebratory articles announcing a new genome give little more than lists of genes in this or that category by way of exegesis. It is as if those searching for extraterrestrial life obtained a telephone book (or to be charitable the yellow pages) of some remote civilization and tried to reconstruct the social system.

One route into the problem of how the genome defines the organism is through development and specifically how the genome dictates the expression of genes listed therein (E. Davidson) (All steps in the process by which a segment of a eukaryotic genome is transcribed into nuclear RNA; the introns spliced out; the mRNA is capped; exported to the cytoplasm; translated; and the nascent peptide chain chemically modified; are subject to regulation. The details fill several chapters of the major molecular biology texts and the student needs to become familiar with them). Among the commentaries surrounding the publication of the human genome (which will not be ’complete’ in the usual sense of this word for many years), was how few genes we have (30-35k) roughly twice the number of a model plant, nematode, and fly (16k). (Single celled organisms such as yeast have 6000 genes and bacteria have typically between 1000-4000 genes) Man is not the center of the genomic universe, anymore than he is the center of the celestial one. The realization of the commoness of man’s genomic endowment recalls an earlier ’paradox’ that a number of seemingly simpler organisms (salamander, tulips and water lilies) have larger genome sizes than we do by a factor of ~ 8. The resolution of this paradox was the category of ’junk DNA’, that with no obvious function; *they* may have more total DNA, but we have more genes. (Because of their genetic over endowment, the number of tulip genes may never be known directly, since they are expensive to sequence.)

The next line in defense of man’s uniqueness, is gene control as revealed most clearly in development. Here at least the numbers mark a big jump in the fraction of the genome available for regulatory purposes (80% vs 20%) when comparing model multicellular organisms (plant, nematode, and fly) with genome sizes in the range of 150 megabases (Mb) compared with 12 Mb for yeast and 1-4 Mb for bacteria (For humans, genes, narrowly defined as protein coding regions, make up less than 3% of the 3000Mb genome, and manifestly repetitive and perhaps parasitic DNA another 50%). Thus multicellularity, at least, calls for a big increase in regulatory depth. (How this regulation is achieved is also the subject of many texts and actively studied, but suffice it to say there is no Cartesian system in the genome giving coordinates for this or that body part, but rather a seemingly haphazard medley of space and time dependent signals that define one part relative to others.)

Everyone realizes but sometimes forgets to say, that cells make cells, genomes do not. There are no genes coding for lipids per se, but hundreds of different lipids, specific to particular locations in the cell, are built by a variety of enzymes. The cell is highly compartmentalized, traffic between compartments is regulated, and proteins with correlative activities are clustered. Particular subsystems can be reconstituted in vitro from purified components, but even the biochemists would not call this life.

2 New Technologies

Bioinformatics deals with such issues as efficient archival, retrieval and dissemination of information (eg gene ontologies); how to effectively compare sequence; automatically assign function to stretches of the genome (annotation); how to organize sequencing projects and assemble the $\sim$ 600 base pair (bp) fragments that are the immediate output of the sequencing machines into whole genomes. A snapshot of the field can be found on the web sites and proceedings of the major meetings (eg ISMB, and RECOMB), however the tone of these contributions is closer to a technology essential to biology rather than theoretical biology.

A number of bioinformatic problems such as locating genes in raw sequence, have a heavy statistical component [**?**]. Regulatory sequences pose different problems, since they occur in 100-500bp clumps of 2-20 sites each of 5-15bp. In the fly these so called 'modules' can be up to 100kb from the genes they control, though under 10kb is typical. Their discovery is akin to deciphering a language without knowing either the words or the grammar and in the presence of much variability. The algorithms are again statistical and involve difficult search problems to which physicists have much to contribute.

Biology just as physics, progresses through the application of new technologies. Sequencing is one such technology whose costs have decreased (to about \$0.10 per finished base) so much that a recent cover of Science pictured a Noah's ark of organisms at a cocktail party discussing whose genome would be sequenced next (Cornell is coordinating a canine project, so fido will be pleased). There is an interesting technical history to be written about the advances in instrumentation (capillary electrophoresis), chemistry (the end labeling dyes, and preparation of clones), computer science (the assembly algorithms), and process control (there were over 10^7 clones amplified and sequenced for the human project) that made all this possible. Success depended on engineering in the best sense of the term, since the costs and accuracies of all the technologies that intervened between organism and finished sequence had to be balanced against each other.

Another technology essential to my lectures is mRNA gene expression. Currently there are both artesanal small lab approaches and high tech industrial ones competing for acceptance and commercial success. The technology depends on an enzyme that copies RNA to DNA, used by certain viruses such as HIV; now productively harnessed to copy in one reaction all the mRNA produced by a population of cells to chemically labeled DNA.

The problem is then how to assay the level of all 6000 potential transcripts in yeast say, which are mixed in one tube. The key is, of course, to exploit the base complementarity of DNA. In the laboratory scale spotted array technology (http://cmgm.stanford.edu/pbrown/) as applied to yeast, pairs of gene specific primers are used in 6000 separate reactions to amplify genomic DNA (with the aid of another product of biotechnology, PCR, itself made possible by another hijacked enzyme, this time from hot spring bacteria). Then $\sim 50-100\mu$ spots of $\sim$ 500 bp double stranded (ds) DNA for each gene are arrayed on a specially surfaced glass slide by a robot, and anchored down. The fluorescently labeled single stranded (ss) cDNA (complementary to the mRNA) is then allowed to hybridize with the slide and the fluorescent level of each spot is a measure of that gene's expression. However the hybridization kinetics of ssDNA with surface bound dsDNA is unknowable, so the fluorescence is calibrated by processing a reference sample of mRNA identically to the real sample, but labeling it with another color. The color ratio is then the mRNA expression ratio. Clearly genes with similar sequences will cross hybridize and can not be distinguished.

The alternative high tech approach to measuring mRNA levels synthesises about 20 length 26bp tags for each gene directly on the chip by methods inspired by lithography in the semiconductor industry (http://www.affymetrix.com/index.shtml). The redundancy is necessary for controls and the company supplies black box software (that the mathematically literate would want to modify, M. Magnasco submitted) to reduce the multiple oligo readings to a single number. The kinetics of hybridization again has to be calibrated by color ratios, which now go on separate chips (each costing hundreds of dollars, so this is a technology aimed at medical applications). There are still many other technologies vying for attention (eg http://www.rii.com/home.htm); let a hundred flowers bloom! Because there is money to be made, lawsuits are common (eg the saga of Ed Southern vs Affymetrixs).

DNA chips are a versatile genome wide read out device. They have been utilized (P. Brown and M. Snyder) to measure where certain proteins bind on regulatory DNA, by crosslinking all proteins attached to DNA, fragmenting the DNA, extracting the protein (plus DNA) of interest with an antibody; undoing the crosslinks; and assaying the liberated DNA on an array spotted with all the intergenic regions of yeast.

3 Sequence Comparison

The comparison between two sequences was probably the first 'killer application' that drew many computer scientists into molecular biology, and as a measure of their success, it would be impossible to imagine modern biology without it. This subject is well described in textbooks [**?**, **?**], so I will merely state the general ideas which recur in other problems and emphasize the shortcomings. General expositions have such a preemptory tone that the student might infer that it is a closed subject, whereas many obvious questions are not resolved and provide problems for the statistically inclined. The experts are well aware of these questions, but seldom write about them.

The first ingredient of sequence comparison for proteins is a scoring 'matrix' which quantifies, for pairs of amino acids, the differing penalities to be assigned to the replacement of one amino acid by another, (slight for similar residues and large when a hydrophobic residue is substituted for a charged one). Values are derived from collections of aligned homologous protein domains where there are no gaps or deletions. Treat positions in the alignment as independent and compute $P_{a,b} = \langle \rho_a \rho_b \rangle$ where the a, b run over the 20 amino acids and the normalizations are such that $\sum_b P_{a,b} = \langle \rho_a \rangle$, the fraction of a residues in the data set. There is an implicit time parameter τ induced by grouping with weight one, all sequences with percent identity over some value (this also prevents biases in the protein data base from overly influencing the scoring matrix). That τ indeed acts as a time within the correlation function $P_{a,b}$, can be seen from two limits; $P_{a,b}(\tau -\rangle 0) = \delta_{a,b}\langle\rho_a\rangle$ and $P_{a,b}(\tau -\rangle\infty) = \langle\rho_a\rangle\langle\rho_b\rangle$. One then defines the transition probability $T(a \rightarrow b) = P_{a,b}/\rho_a$ and the scoring or substitution matrix $s_{a,b} = \ln(P_{a,b}/\rho_a\rho_b)$. (Thus for short times there are no transitions, while for long times the probability for obtaining a given residue is independent of where it came from.) I have added these few details to make evident there is nothing very subtle in the construction of the scoring functions (eg BlosumXX) that everyone uses.

Two sequences (not necessarily of the same length) are brought into correspondence and thus scored, either by making point mutations or creating gaps (or intervals of deletions) which are penalized by one parameter to create and another to extend. Global alignment finds the optimal score accounting for the entire sequence. It is constructed in a time of order the product of the two sequence lengths by a recursive calculation. Place the two sequences along the x and y axis and map each alignment between them into a path through the rectangle thus defined. A diagonal bond means bases (or residues) i,j are paired, a horizontal bond means i (on the x-axis) is paired with a gap, and a vertical bond means the reverse. Starting from (0,0), find the best scoring path up to the perimeter of the sub rectangle defined by (i,j), and then fill in the next row and column from these values and proceed to the end of either sequence. Local alignment finds the highest scoring subsequences in a pair of sequences in comparable time. Various short cuts to complete pairwise comparison are essential to practical applications (nb there are over 10^{10} bases deposited in GENBANK) and go under names such as BLAST (http://www.ncbi.nlm.nih.gov:80/BLAST/) and FASTA (http://www.people.Virginia.EDU/ wrp/pearson.html).

The first thing a biologist does with a new sequence is compare it with the huge data base of known sequences. Thus it is important to know the probability of obtaining a certain score by chance from uncorrelated sequences, which is best done by first determining the functional form of $P(s \geq s_0)$, the probabality of a score larger than s_0. This is done by a stationary phase argument (Yu and Hwa) very analogous to the passage from a microcanonical ensemble to a canonical one. One finds for large s_0, $P \sim N_1 N_2 e^{-\lambda s_0}$, where $N_{1,2}$ are the lengths of the two sequences (or sequence times data base) being compared and for ungapped alignment $\lambda = 1$ because of the definition of the scoring function, while for gapped alignment λ has to be computed numerically as function of the gap penalties. Rapid ways of doing this akin to importance sampling in Monte Carlo have been developed by T. Hwa and coworkers. Some theory is necessary here, since a probabality has to be placed on events that are rare, but become possible when looking through a sample size of 10^{10}.

Now for the problems. The scoring function is designed for ease of computation, the iterative algorithm ignores history (prior resides on the optimal path) other than whether a gap is being created or extended. There are no block rearrangement moves for instance. The most widely used algorithms return the best local alignment (ie entropy is ignored) rather than the probability of transforming one sequence into another in all possible ways, which is more relevant biologically even within the impoverished move set of current algorithms. The scoring optimizes the contiguous interval with the highest total score without regard to length, but this does not mean that a shorter region of greater similarity might not give a more significant probability score under some other scheme.

The scoring parameters are not contingent on the species being compared and more importantly not optimized for maximum discrimination. To make this clear by analogy, imagine a substrate with patches of material (the sequences) with different affinities for water. If the regions are to be distinguished based on their ability to adsorb water, what is the optimal point in the phase diagram at which to work. Clearly the condition where water wets one substrate and not the other will provide optimal discrimination, ie near a phase transition point, small inhomogeneities can have large effects. Of course in reality there are a continuum of substrate affinities and a cost to be paid for small domains. Nevertheless working at a random point in the phase diagram is not a recipe for optimal discrimination.

Two other issues are addressed in part by an extension of the BLAST algorithm, PSI-BLAST. The typical scoring function is position independent, yet certain regions of proteins are more constrained than others and they should be weighted differently (ie the catalytic region is more constrained than the loops which tether domains together). BLAST also looses information by using only pair scores in matching against a data base. A marginal match to several *unrelated* data base entries may be significant even if any pair is not. However separate data base entries for a human, mouse, and rat protein do not add much to comparisons against an unknown fly protein. So its not trivial to put a significance measure on the comparison of several species at once.

Given a genome the first question asked is what are the genes. Most attention has been focused on protein coding genes; those encoding functional RNA's (ie not messages) are very interesting but require different algorithms (S. Eddy). The primary modeling tool is Hidden Markov Models (HMM's) [**?**]. To illustrate just a simple Markov model, imagine one is presented with a long string σ_i of 0,1 which is not obviously periodic. One might model it by letting the ith bit occur with a probability that depends on several previous ones. So in the simplest case, where only memory of the previous bit matters, the model is entirely specified by a 2×2 matrix of transition rates $T(\sigma_1 \rightarrow \sigma_2)$ where $\sum_{\sigma_2} T_{1,2} = 1$ ie the sum of all probabilities for leaving a state, must be 1. The probability of observing 0,1 satisfies $\sum_1 p(\sigma_1) T_{1,2} = p(\sigma_2)$. Thus we have given the right and left eigenvectors of the matrix T with eigenvalue 1, which is in fact the largest eigenvalue because all the entries of T are positive. (Under these definitions, the usual nearest neighbor Ising model in one dimension would not be Markov since the correlations in spin are not strictly limited to a finite number of lattice sites).

Hidden Markov models originated in speech recognition where the computer was presented with sounds and needed to infer the phonemes that the speaker was uttering. So in our context, assume there were two hidden states coding (C) and non coding (N) with transitions between them as defined for our Markov model. For each hidden state there are separate probabilities for 'emitting' 0,1, and it is only 0,1 that one observes. The inversion problem has two levels; first of inferring the model parameters from data, and then partitioning the data into domains corresponding to hidden states. In the case of gene finding, one has a large training set where the hidden state is known and one can fit the emission probabilities and also the transitions between hidden states. Then real data can be scored and probabilities assigned to where the coding and non-coding regions lie. A HMM is well suited to gene finding since the biological structure can be built in. Promoters are followed by exons, exons by introns, successive exons must maintain a common codon phasing, and various splice signals must fall in the correct place etc. See C. Burge and S. Karlin for the current state of the art.

The task of determining parameters directly from data is done by iteration. The basic idea is to note that a transfer matrix like calculation (by summing over all paths through the hidden states starting from either end) will supply the total probability for the data given the model. Work from both the right and left ends and compute the probability for observing a certain base and hidden state (or transition between them) at a given point in the data. A suitable spacial average of this 'profile' value gives the next iterate for the parameters. (When parameters have converged or are directly fit, a profile calculation will reveal where the hidden states are.)

4 Clustering

Many problems in bioinformatics call for grouping similar things together - the task of clustering. These may be genes whose behavior is monitored in a series of chip experiments or a series of samples of cancer tissues for which the expression of a palette of genes is observed and one wants to group the cancers into types. Clustering can be effected along one dimension as in these examples, or in two when for instance one wants to find blocks in the array of *genes samples* which isolates subsets of genes that are most indicative of particular samples.

Algorithms can be categorized by a series of Levi-Straussian binaries; hard vs soft (is cluster membership binary or probabilistic); one pass or annealed; agglomerative vs devisive (do clusters grow from the primary elements by fusion, or do clusters derive by fission from larger sets). Phylogenetic (family tree) clustering is hard, one pass and agglomerative. The k-means algorithm is a descent scheme in which each point is assigned to the nearest center, and the centers repositioned to be the geometric centers of the points assigned to them. It becomes a divisive algorithm if new centers are added to eccentric clusters. Another scheme assigns a Potts variable to each element, and the coupling constant between two elements is a monotone function of their degree of similarity. As the temperature is lowered, groups within which the Potts degree of freedom is more correlated than some value define clusters (E. Domany).

There is no clustering algorithm optimal for all problems. Often essential to success is the choice of metric. For gene expression, frequently only a small percentage of the genome has a meaningful response. If one measures the correlation between genes by summing over all experiments (assuming many are available) the real signal from a few experiments is washed out by the noise from the others. Thus the metric should weight experimental values by their significance determined from the noise level.

Given a metric, cluster membership can be based on the average pair score of the new element with other cluster members, the best score with any single cluster member, or some other cluster wide score which is not a sum of pairs. Clustering is bedeviled, as are many other optimization problems, by multiple local optima and it is frequently unclear when, if ever, one has hit upon the best one. Another short coming of most schemes is the absence of a statistical model from which to assign significance to a particular clustering. Most algorithms will cluster random variables.

Some of these issues are illustrated by a clustering scheme for sequences developed at Rockefeller (van Nimwegen) which has an obvious bearing on motif finding and illustrates aspects of Bayesian statistics (named after an 18th century English cleric http://www-groups.dcs.st-andrews.ac.uk/ history/Mathematicians/Bayes.html now the object of cultic veneration). Assume an alphabet of size A and letter probabilities p_a. Then the probability of a particular string of letters (n_a of each, $\sum_1^A n_a = N$) is $P(data|model) == P(n_a|p_a) = \prod_1^A p_a^{n_a}$. This is properly normalized since the sum over all possible strings of data just reduces to $(\sum_1^A p_a)^N = 1$. To compute $P(p_a|n_a) = P(n_a|p_a)P(p_a)/P(n_a)$ (the definition of conditional probab.lity), we have to make an assumption about $P(p_a)$ namely that it be uniform ie $P(p_a) = \text{cst.}\delta(1 - \sum_1^A p_a) \prod_1^A dp_a$. To compute $P(n_a)$ we have to evaluate the integral $I(x) = \int_0^\infty \delta(x - \sum_1^A p_a) \prod_1^A dp_a$ for $x = 1$ (in which case the upper limit can be replaced by 1). By homogeneity, $I(x) = x^{N+A-1}I(1)$; multiply both sides by e^{-x} and integrate from zero to infinity on x, to find, $I(1) = \prod_1^A n_a!/(N + A - 1)!$. From this we can derive $P(p_a|n_a)$ and for instance show $\langle p_a \rangle = (n_a + 1)/(N + A)$, ie the average value of the model parameter p_a given a finite sample drawn from the distribution is not the most probable value n_a/N, which for instance can be 0.

To apply this to clustering, consider a large number, S, of sequences, each of length ℓ obtained by sampling M unknown frequency matrices, w_a^i, where $i = 1, 2..\ell$, $\sum_1^A w_a^i = 1$ (ie the entries in the matrix give for each column i the frequencies of the letters). The problem is then to group together the sequences from a common weight matrix and recover, within the errors imposed by the finite sample size, the set of w_a^i. Consider a subset of N sequences, then the probability $P(C)$ that they were drawn from single weight matrix is the product of $I(1)$ over all the ℓ columns. A probability distribution can be defined over the entire set of S sequences by allowing all possible partitions into clusters, each with a weight $\prod_i P(C_i)$. Thus there is a competition between all ways of partitioning S things into subsets and the 'energy' which favors, one can show, putting sequences from the same frequency matrix together. This weighting scheme can be used, either in a one pass phylogenetic clustering, or more correctly with Monte Carlo sampling which will generate soft clusters and allow an assignment of significance.

Intuitively for given S, there is a limit to how many frequency matrices can be resolved (which depends also on their degree of polarization). Discrimination obviously improves if more samples from the same matrix are supplied. Finally there is a very interesting regime where it possible to classify most sequences if the set of M frequency matrices is known yet it is impossible to cluster these sequences knowing nothing about the matrices. The former problem is the one faced by the cell, since it 'knows' the proteins which do the site recognition, whereas sequence clustering is only a problem for the bioinformatician.

5 Gene Regulation

The extraction of the sites active in transcription control from the genome is a more daunting task than gene identification since the individual protein binding sites are much smaller than typical exons and their arrangement is not so correographed as the promoter-exon-intron pattern of genes. Three types of data can be brought to bear on the problem and all appear necessary. For a single genome, one can search for repetition between the regulatory regions of different genes. The repeats can be at the level of specific strings (perhaps with a few spelling errors) or groups of similar strings that occur in clusters. In all cases it is assumed that improbability under some model implies function and for the calculations to be tractable there needs to be some vestige of the signal on scales short enough to be searched exhaustively. (The hard cases are those where the motif is long and mutated and where there is no statistically significant signal in just a few copies.) The issue raised above, that the cell can function by merely classifing sites while there may not be enough copies to allow clustering, is clearly relevant here. The application of one genome wide algorithm to yeast was discussed by H. Bussemaker.

The second source of data is comparative genomics, namely we exploit the fact that what is functional is more constrained and evolves less rapidly than what is not. The protein coding regions serve as landmarks for the regulatory regions to compare since they are much larger and evolve more slowly than the regulatory sites. In reality there are merely degrees of constraint and the scale in bp on which compensatory mutations (preserving fitness) can occur is also unknown. The ideal case is individual protein binding sites immersed in a sea of random sequence. In bacteria where the total regulatory region of a gene is a few hundred bases, the conserved domains are typically larger than a single binding site (N. Rajewsky submitted). The current state of the art (McCue and C. Lawrence) in this area is to examine the regulatory regions for one gene from several species. One is then faced with the task of clustering sites for individual genes into families recognized one hopes by a single protein.

Finally there remains mRNA expression data. If the question being asked is how expression follows from sequence, there is little reason to first cluster genes based on similarity of expression and then look for common sequence motifs. The clustering should follow from the sequence. Following the idea that the polymerase which makes mRNA is recruited to the promoter by equilibrium binding to certain sites (or other proteins attached to these sites), we have fit the log of the expression ratio, R_g for gene g, to the sum of contributions F_m for motif m by minimizing $\sum_g (R_g - C - \sum_m (F_m N_{g,m}))^2$ with respect to F_m and C, where the integer $N_{g,m}$ is the number of copies of motif m upstream of gene g. (H. Bussemaker). This scheme is sensitive to combinatorial control. Genes which do not respond, but carry a functional site, are informative about potential compensatory factors. All genes are fit and when the residuals are Gaussian it is easy to assign significance to the sequence motifs that correlate with expression.

The intent of this very condensed summary is to stimulate the curiosity of students in the physical sciences for a nascent field where a medley of techniques are required for success. Bioinformatics is most fruitfully situated as a branch of natural science, merely publishing a clever algorithm is not enough, it has to be used on real data to solve a real problem. The most significant problems will probably emerge by looking at genome wide data rather than reading biology texts, though they are essential. Their authors, are in most cases not quantitatively trained and do not know what can be done computationally. The flood of quantitative information in the form of genomic sequences, gene expression, and protein interactions, provides for the first time in molecular biology a realm where the primary discoveries could emerge from analysis of public data. It remains to be seen whether new data available for gene regulation will support a level of interpretation that would merit the term theoretical biology.

Other than the books cited below, a number of other authors have been mentioned in the text, their past and future contributions along with abstracts can be found by searching in the medline data base (http://www4.ncbi.nlm.nih.gov/PubMed which any student must be familiar with. The references are restricted to a few common texts.

References

[1] Durbin, R., Eddy S., Kroch A., and Mitchison, G., “Biological Sequence Analysis”, Cambridge Univ. Press 1998.

[2] Waterman, M.S., “Introduction to Computational Biology”, Chapman & Hall N.Y., 1995.

[3] Gusfield, D., “Algorithms on Strings Trees, and Sequences”, Cambridge Univ Press, N.Y., 1997.