BIOINFORMATICS

Vol. 16 no. 4 2000
Pages 367–371

On the convergence of a clustering algorithm for protein-coding regions in microbial genomes

Pierre Baldi [1]

Department of Information and Computer Science, University of California, Irvine, CA 92697-3425, USA

Received on October 3, 1999; revised and accepted on November 1, 1999

Abstract

Motivation: *As the number of fully sequenced prokaryotic genomes continues to grow rapidly, computational methods for reliably detecting protein-coding regions become even more important. Audic and Claverie (1998)* Proc. Natl Acad. Sci. USA, ***95****, 10026–10031, have proposed a clustering algorithm for protein-coding regions in microbial genomes. The algorithm is based on three Markov models of order k associated with subsequences extracted from a given genome. The parameters of the three Markov models are recursively updated by the algorithm which, in simulations, always appear to converge to a unique stable partition of the genome. The partition corresponds to three kinds of regions: (1) coding on the direct strand, (2) coding on the complementary strand, (3) non-coding.*

Results: *Here we provide an explanation for the convergence of the algorithm by observing that it is essentially a form of the expectation maximization (EM) algorithm applied to the corresponding mixture model. We also provide a partial justification for the uniqueness of the partition based on identifiability. Other possible variations and improvements are briefly discussed.*

Contact: *pfbaldi@ics.uci.edu*

Introduction

As the number of fully sequenced prokaryotic genomes continues to grow rapidly, computational methods for reliably detecting protein-coding regions become even more important. In Audic and Claverie (1998), a new method is presented for predicting protein-coding regions in microbial genomic DNA sequences. Unlike other methods (Borodovsky and McIninch, 1993; Borodovsky *et al.*, 1995; Salzberg *et al.*, 1998) that often require an annotated pre-existing training set, this method does not require a training set, or any prior knowledge of the statistical properties of the genome under study. In this sense, this method is also related to Hayes and Borodovsky (1998). It is essentially a clustering, or self-organizing approach, that uses all the available unannotated genomic data for its calibration and is not based on direct pairwise comparisons.

In a slightly simplified version, the method works essentially as follows. The genomic sequences under consideration are considered to result from n Markov models M_α of order k, each one responsible for a different kind of non-overlapping subsequences. In order to detect protein-coding regions, a natural number of Markov models is $n = 3$, corresponding to three different regions: (1) coding on the direct strand, (2) coding on the complementary strand, (3) non-coding. The available genomic sequences are then cut into non-overlapping fragments of length w. Typical values for k and w are $k = 5$ and $w = 100$. The resulting sequences are randomly partitioned amongst the three models and the three Markov models are initialized accordingly, in a semi-random fashion. The algorithm then proceeds iteratively by cycling through all the available fragments. At each cycle, a fragment W is assigned to one of the three classes depending on the highest posterior probability

$$P(M_\alpha|W) = \frac{P(W|M_\alpha)P(M_i)}{\sum_\beta P(W|M_\beta)P(M_\beta)}. \qquad (1)$$

The parameters of each Markov model are then updated using all the sequences assigned to the corresponding sub-model. The assignments of fragments to models could also be based on a threshold cut-off: if the posteriors are below a certain value, the corresponding fragment remains unassigned. The implementation described in Audic and Claverie (1998) is slightly different in order to handle length variability and to avoid setting up an arbitrary cut-off. These differences are discussed below. The windows also are not exactly contiguous but slightly spaced for convenience reasons that are irrelevant for the issues raised here—alternatively the spacing can be considered as part of the windows. Notice that k, w and n are the only parameters of the model that need to be fixed 'externally' in the algorithm. Obviously, the matrices

[1]Also at the Department of Biological Chemistry, College of Medicine, University of California, Irvine, USA. To whom all correspondence should be addressed.

of the Markov models are additional parameters of the model—but these are directly fit to the data.

It is clear that this method can easily be applied to unassembled genomes. In Audic and Claverie (1998), the method is validated on 10 complete bacterial genomes from four major phylogenetic lineages. It is empirically observed that this simple algorithm exhibits two essential features: (1) rapid convergence, typically within 50 iterations, (2) stability of the final Markov transition matrices and of the genomic partition under different random initializations. The resulting partition corresponds indeed to the three putative classes described above. The algorithm can identify protein-coding regions with an accuracy of up to 90% while tolerating simulated error rates of 1–2% [see also Borodovsky and Peresetsky (1994) and Mathe *et al.* (1999)].

For completeness, we now provide a very concise review of mixture models and the expectation maximization (EM) algorithm since these are essential to understand the convergence of the algorithm of Audic and Claverie (1998).

Mixture models and the EM algorithm

Markov models of order k

Consider an alphabet A, in our case $A = \{A, C, G, T\}$. An homogeneous Markov model M of order k for sequences over A is specified by an initial distribution $\pi(s)$ over all possible sequences of length k and an $|A|^k \times |A|$ transition matrix. The transition matrix specifies the probabilities $P(X|s)$ of producing the letter X given the prefix subsequence s of length k. The probability of a sequence W of length L is then described by

$$P(W|M) = \pi(s_0) \prod_{i=k}^{i=L-1} P(X_i|s_{i-k}). \qquad (2)$$

Mixture models

Mixture models (Everitt and Hand, 1981; Titterington *et al.*, 1985) are probabilistic models built using positive convex combinations of distributions taken from a given family. It is clear that the algorithm above corresponds to a mixture of three Markov models of order k with

$$P(W|M) = \sum_{\alpha} \lambda_\alpha P(W|M_\alpha) \qquad (3)$$

where the mixing coefficients λ_α satisfy: $\lambda_\alpha \geq 0$ and $\sum_\alpha \lambda_\alpha = 1$. The mixing coefficients represent, of course, the proportion of sequences in each class. Such a mixture model is also representable as a probabilistic graphical model or Bayesian network (Baldi and Brunak, 1998). Similar mixtures of hidden Markov models have in fact been used to model protein sub-families (Krogh *et al.*, 1994).

If the data D consists of N sequences $W_1, \ldots, W_N$ assumed to be independent, then the likelihood is given by

$$P(W_1, \ldots, W_N|M) = \prod_{i=1}^{N} \sum_{\alpha} \lambda_\alpha P(W_i|M_\alpha). \qquad (4)$$

By Bayes theorem, the posterior for an example W to belong to class α is given by

$$P(M_\alpha|W) = \frac{\lambda_\alpha P(W|M_\alpha)}{P(W|M)}. \qquad (5)$$

By differentiating the log-likelihood, augmented by the normalization constraints on the mixing coefficients and using equation (5), one obtains

$$\sum_{i=1}^{N} P(M_\alpha|W_i) - N\lambda_\alpha = 0 \qquad (6)$$

and

$$\sum_{i=1}^{N} P(M_\alpha|W_i) \frac{\partial \log P(W_i|M_\alpha)}{\partial \theta_\alpha} = 0 \qquad (7)$$

where θ_α is any free parameter of M_α. In reality the entries of the transition matrix of M_α are not entirely free since they must satisfy the normalization constraints. Such constraints can be added to the log-likelihood with the use of Lagrange multipliers. Alternatively the θ_αs can be reparametrized using, for instance, normalized exponentials as in Baldi and Chauvin (1994). But in general, this suggests a natural iterative algorithm to maximize the likelihood given by a mixture whereby the mixing coefficients are first set to the optimal empirical average

$$\lambda_\alpha^* = \sum_{i=1}^{N} P(M_\alpha|W_i)/N. \qquad (8)$$

The parameters θ_α are obtained by solving equation (7), which is a weighted average of the maximum likelihood equations for each individual component, weighted by the class membership posterior probabilities. This is in fact a special case of the EM algorithm.

Expectation maximization

The EM algorithm is useful in models and situations with hidden variables. Typical examples of hidden variables are missing or unobservable data, mixture parameters in a mixture model, and hidden states in graphical models, such as hidden states in HMMs (Hidden Markov Models). If D denotes the data, we assume that there is available a parametrized joint distribution on the hidden and observed variables $P(D, H|\theta)$, parametrized by θ. Let us assume that the objective is to maximize the likelihood $\log P(D|\theta)$. Since in general it is difficult to optimize

$\log P(D|\theta)$ directly, the basic idea is to try to optimize the expectation $E(\log P(D|\theta))$. The EM algorithm is an iterative algorithm that proceeds in two alternating steps, the E (expectation) step and the M (maximization) step. During the E step, the distribution of the hidden variables is computed, given the observed data and the current estimate of θ. During the M step, the parameters are updated to their best possible value given the presumed distribution on the hidden variables. The algorithm starts with an estimate θ^0 at time 0. At time t, the EM algorithm can be written as:

1. E step: Compute the distribution $Q^*(H)$ over H, such that $Q^*(H) = P(H|D, \theta^{t-1})$.
2. M step: Set $\theta^t = \arg_\theta \max E_{Q^*}[\log P(D, H|\theta)]$.

This can also be interpreted in terms of a double free energy optimization with respect to Q and θ (Baldi and Brunak, 1998).

In the case of a mixture, the hidden variables are the indicator variables corresponding to the choice of one of the components and θ represents the component parameters. The E step estimates the mixture coefficients using equation (8). The M step maximizes the likelihood associated with each component according to equation (7).

It should be clear now that the algorithm described in the introduction is an approximation to EM. The probabilities $P(M_\alpha|W_i)$ are implicitly approximated by the counts N_α/N where N_α is the total number of sequences assigned to class α. The parameters of each Markov model are updated by maximum likelihood using counts based on the corresponding set of sequences. This approximation to EM where probabilities are thresholded to 1 or 0 is routinely used in HMMs when emission or transition counts are based only on the most likely paths associated with each sequence in what is called Viterbi learning. In the clustering literature, this algorithm is also known as k-means.

Convergence

It can be shown (Dempster *et al.*, 1977) that each step of the EM algorithm tends to increase the likelihood. Thus in general the EM algorithm converges to a maximum of the likelihood function, albeit not necessarily a global one. In general, Viterbi-like approximations to EM are also convergent and are used as such in HMM applications. This provides an explanation for the convergence of the clustering algorithm described in Audic and Claverie (1998). It does not prove, however, convergence to a global optimum, nor the fact that such global optimum might be unique.

Identifiability

As far as the uniqueness of the global optimum is concerned, here we prove a slightly weaker result by restricting ourselves to the space of exact mixtures of Markov models of order k. In other words, if the data we are modeling is indeed produced by a mixture of Markov models of order k, then this mixture is unique or identifiable. Specifically, if for every W

$$\sum_\alpha \lambda_\alpha P(W|M_\alpha) = \sum_\beta \mu_\beta P(W|N_\beta) \qquad (9)$$

then there exists a permutation of the indices such that for each α, $\lambda_\alpha = \mu_\gamma$ and $M_\alpha = N_\gamma$ for some γ. Mixtures over a family F are identifiable if and only if the set F is linearly independent over the real numbers (Everitt and Hand, 1981; Titterington *et al.*, 1985). Thus we need only show that the set of Markov models of order k is linearly independent. The proof of this fact is given in the Appendix. One observation is that the identifiability result is valid in the case of large data sets—which is usually the case in genomic applications. It is of course possible for two completely different mixtures to coincide on a small data set, but not on all Ws.

Discussion

The model introduced in Audic and Claverie (1998) is a mixture of Markov models. The learning algorithm described in the introduction is a Viterbi approximation to the EM algorithm and as such is convergent—as observed in the original simulations. A simple mixture model does not capture the length of the different type of genomic regions, nor the transition events from one class to the next. This deficiency is addressed in Audic and Claverie (1998) by a slight modification of the training algorithm. Instead of using all the windows for training, only those corresponding to sufficiently *homogeneous* stretches of DNA are used—a stretch being homogeneous if and only if the windows it contains are classified in the same way. Such procedure does not alter the convergence qualities of the algorithm—it may even reinforce them since in a sense the training set gets cleaner and cleaner as the mixture model improves. When k is small with respect to w, it should not matter also whether fitting is done using the windows separately or the contiguous sequences themselves. Furthermore, in cases where only a relatively small fraction of the data is discarded the fitting operations are similar. An alternative approach of course is to incorporate a model of the region lengths and/or transitions into the probabilistic model itself, as in several current gene finders (Burge and Karlin, 1997). This can be achieved, for instance, by the use of hidden Markov models. The arrangement of hidden states and their transition probabilities can be used to model duration (Rabiner, 1989; Durbin *et al.*, 1998).

There are additional modeling possibilities that are suggested by the mixture framework, such as the use of

hierarchical modeling or the introduction of priors—in particular of Dirichlet priors. Markov models of order 5 seem to be optimal because of the well-recognized (Fickett and Tung, 1992) and important differences between DNA hexamer statistics in coding and non-coding regions. Yet another reason, as pointed out in Audic and Claverie (1998), is also because there is often not enough data to train Markov models of higher orders. Such problems could be addressed by using Dirichlet prior distributions equivalent to introducing pseudo-counts to handle *n*mers that are poorly represented in the available fitting data. The connection to EM suggests also a number of possible algorithmic variations such as smooth (non-Viterbi) and on-line training (Baldi and Chauvin, 1994) although it is unlikely that these alone could lead to substantial performance improvements. The experimentally observed—although not quantified—robust convergence of the algorithm to a single point suggests the presence of a strong attractor with a broad basin. Several elements of small stochasticity present in the algorithm may further help the convergence by escaping small local minima. We have shown that the mixtures considered here are identifiable. Thus the optimal mixtures have a unique representation and are likely to be non-degenerate.

The clustering method analysed here seem to work well with shotgun sequencing and with bacterial genomes, where coding regions often represent more than 90% of the total DNA. With more than 40 genomes already shotgun sequenced to date, such computational methods are useful for parsing the rapidly growing data. Their extension to eukaryotic genomes—where the fraction of coding sequences is often less than 10%—remains a challenge however.

Appendix: Independence of Markov models

Here we prove that Markov models of order k are independent. For simplicity, we prove it in the case where the alphabet has only two symbols $A = \{0, 1\}$ and when $k = 0$. The general case can be studied along the same lines. When $k = 0$, a Markov model M is entirely described by a single number $P(0|M) = p$. Assume for contradiction that there are n different Markov models $M_1, \ldots, M_n$ described by the probabilities $p_1, \ldots, p_n$ which are dependent. Then there exists a vector of non-zero real numbers $a_1, \ldots, a_n$ such that for every W: $\sum_i a_i P(W|M_i) = 0$. For each word containing r 0s and s 1s, this translates into

$$\sum_{i=1}^{n} a_i p_i^r (1 - p_i)^s = 0. \qquad (10)$$

Notice that if $p_1 = 1$ for instance, then for every s: $a_1 + \sum_{i>1}^{s} a_i p_i^s = 0$. Since all the models are different by assumption, all the other p_i are strictly less than one. By letting $s \to \infty$, we see that $a_1 = 0$ which contradicts our starting assumption. Thus, without any loss of generality, we can assume that all the p_is are strictly between 0 and 1. The set of equations obtained when $s + r = n$ is an homogeneous linear system in the a_is with a classical Van der Monde matrix. The only cases where it can have additional solutions involves equations of the form $p_i = 0$ or $p_i = 1$, which is impossible by the remark above, or $p_i = p_j$ which is equivalent to $M_i = M_j$ for some i and some j. The latter is also impossible since all M_is must be different. Therefore, the models M_i must be independent.

Acknowledgements

The work of P.B. has been initially supported by an NIH SBIR grant to Net-ID, Inc., and currently by a Laurel Wilkening Faculty Innovation award at UCI.

References

Audic,S. and Claverie,J. (1998) Self-identification of protein-coding regions in microbial genomes. *Proc. Natl Acad. Sci. USA*, **95**, 10026–10031.

Baldi,P. and Brunak,S. (1998) *Bioinformatics: The Machine Learning Approach*. MIT Press, Cambridge, MA.

Baldi,P. and Chauvin,Y. (1994) Smooth on-line learning algorithms for hidden Markov models. *Neural Computation*, **6**, 305–316.

Borodovsky,M. and McIninch,J.D. (1993) Genmark: parallel gene recognition for both DNA strands. *Computers Chem.*, **17**, 123–133.

Borodovsky,M., McIninch,J.D., Koonin,E.V., Rudd,K.E., Medigue,C. and Danchin,A. (1995) Detection of new genes in a bacterial genome using Markov models for three gene classes. *Nucleic Acids Res.*, **23**, 3554–3562.

Borodovsky,M. and Peresetsky,A. (1994) Deriving non-homogeneous Markov chain models by cluster analysis algorithm minimizing multiple alignment entropy. *Computers Chem.*, **18**, 259–268.

Burge,C. and Karlin,S. (1997) Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.*, **268**, 78–94.

Dempster,A.P., Laird,N.M. and Rubin,D.B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Stat. Soc.*, **B39**, 1–22.

Durbin,R., Eddy,S.R., Krogh,A. and Mitchinson,G. (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK.

Everitt,B.S. and Hand,D.J. (1981) *Finite Mixture Distributions*. Chapman and Hall, London and New York.

Fickett,J.W. and Tung,C.S. (1992) Assessment of protein coding measures. *Nucleic Acid Res.*, **20**, 6441–6450.

Hayes,W.S. and Borodovsky,M. (1998) How to interpret an anonymous bacterial genome: machine learning approach to gene identification. *Genome Res.*, **8**, 1154–1171.

Krogh,A., Brown,M., Mian,I.S., Sjölander,K. and Haussler,D. (1994) Hidden Markov models in computational biology: applications to protein modeling. *J. Mol. Biol.*, **235**, 1501–1531.

Mathe,C., Peresetsky,A., Dehais,P., Montagu,V. and Rouze,P. (1999)

Classificatiion of *A.thaliana* gene sequences: clustering of coding sequences into two groups according to codon usage improves gene prediction. *J. Mol. Biol.*, **285**, 1977–1991.

Rabiner,L.R. (1989) A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE*, **77**, 257–286.

Salzberg,S.L., Delcher,A.L., Kasif,S. and White,O. (1998) Microbial gene identification using interpolated Markov models. *Nucleic Acids Res.*, **26**, 544–548.

Titterington,D.M., Smith,A.F.M. and Makov,U.E. (1985) *Statistical Analysis of Finite Mixture Distributions*. John Wiley and Sons, New York.