Statistical Models for Discrete Data of DNA Samples

Sanjay Man Tamrakar

December 4th, 2014

1 Introduction

In this paper we talk about different statistical models to predict the sequence of DNA. Deoxyribonucleic acid, or DNA, is the hereditary material in all living beings, which stores information as a code made up of four chemical bases: adenine (A), guanine (G), cytosine (C), and thymine (T). The order, or sequence, of these bases determines the information available for building and maintaining an organism. DNA bases pair up with each other, A with T and C with G, to form units called base pairs. DNA has a double-helical structure, somewhat like a ladder, with the base pairs forming the ladder's rungs. Biologists are interested in C and G rich DNA, meaning DNA with high C and G compared to A and T. Basically, DNA with high C and G are more stable in nature compared to A and T.

We can use certain statistical models of probability distributions to generate random sequences for discrete data of DNA samples. The mixture model with A, C, T and G are shown below. The first and the second die are given by biologists, where the first die is C and G rich and the second die is C and G poor. The third dice is fair and so that probabilities of getting an A, C, G, T are equal i.e. 0.25.

Let us assume that a process produces a DNA sequence of length N = 49 as follows:

CTCACGTGATGAGAGCATTCTCAGACCGTGACGCGTGTAGCAGCGGCTC

The process uses three tetrahedral dice. Two of the dice are loaded and one is fair. The probabilities of rolling the four letters are known to us, which are as follows:

	A	**C**	**G**	**T**
First die (θ_1)	0.15	0.33	0.36	0.16
Second die (θ_2)	0.27	0.24	0.23	0.26
Third die ($1-\theta_1-\theta_2$)	0.25	0.25	0.25	0.25

Let p_A, p_C, p_G, and p_T denote the probabilities that the process will generate any of the four letters. We calculate the probabilities as follows:

$$p_A = p(A|\theta_1) + p(A|\theta_2) + p(A|1-\theta_1-\theta_2)$$
$$= 0.15\theta_1 + 0.27\theta_2 + 0.25(1-\theta_1-\theta_2)$$
$$= 0.15\theta_1 + 0.27\theta_2 + 0.25 - 0.25\theta_1 - 0.25\theta_2$$
$$= -0.10\theta_1 + 0.02\theta_2 + 0.25$$

$$p_C = p(C|\theta_1) + p(C|\theta_2) + p(C|1-\theta_1-\theta_2)$$
$$= 0.33\theta_1 + 0.24\theta_2 + 0.25(1-\theta_1-\theta 2)$$
$$= 0.33\theta_1 + 0.24\theta_2 + 0.25 - 0.25\theta_1 - 0.25\theta_2$$
$$= 0.08\theta_1 - 0.01\theta_2 + 0.25$$

$$p_G = p(G|\theta_1) + p(G|\theta_2) + p(G|1-\theta_1-\theta_2)$$
$$= 0.36\theta_1 + 0.23\theta 2 + 0.25(1-\theta_1-\theta_2)$$
$$= 0.36\theta_1 + 0.23\theta 2 + 0.25 - 0.25\theta_1 - 0.25\theta_2$$
$$= 0.11\theta_1 - 0.02\theta_2 + 0.25$$

$$p_T = p(T|\theta_1) + p(T|\theta_2) + p(T|1-\theta_1-\theta_2)$$
$$= 0.16\theta_1 + 0.26\theta_2 + 0.25(1-\theta_1-\theta_2)$$
$$= 0.16\theta_1 + 0.26\theta_2 + 0.25 - 0.25\theta_1 - 0.25\theta_2$$
$$= -0.99\theta_1 + 0.01\theta_2 + 0.25$$

Then the probability of the whole sequence is given by:-

$$p(\text{seq.}) = p_C p_T p_C p_A p_C p_G \ldots\ldots p_C$$
$$= p_A^{10} p_C^{14} p_G^{15} p_T^{10}$$

To further move on, we need to define some of the terms and maximum likelihood; so that we get the better understanding of what it is, and use it more precisely to get the result. Now, we define maximum likelihood as given below:

2 Maximum Likelihood

R.A. Fisher developed the theory of maximum likelihood estimator. The likelihood function is the probability of the sample, considered as a function of an unknown parameter θ. If we have a large sample, the parameter value maximizing this function will often yield an excellent estimator of θ.

Let $X_1, X_2, X_3, \ldots\ldots X_n$ have joint density denoted by

$$f_\theta(x_1, x_2, \ldots\ldots x_n) = f(x_1, x_2, \ldots\ldots, x_n|\theta)$$

If we are given observed value $X_1 = x_1, X_2 = x_2, \ldots\ldots\ldots, X_n = x_n$, then the likelihood of θ is the function

$$lik(\theta) = f(x_1, x_2, \ldots\ldots, x_n|\theta),$$ considered as a function of θ.

The maximum likelihood estimate is the estimator of θ that maximizes the likelihood function of θ. The log-likelihood function is obtained by taking the logarithm on the likelihood function, which is also used because the logarithm is a monotonically increasing function. Since, the likelihood function and log-likelihood function are one-to-one, the likelihood function which generates the maximum estimator for θ_1 and θ_2, also yields the same value for the log-likelihood function.

Notice that the *p*(seq.) above is just a polynomial function of θ_1 and θ_2. We can thus apply polynomial algebra to our study of the statistical model. In order to apply algebra, we need to know some definitions, which are stated as follows:

Polynomial Ring

Let *R* be a commutative ring. The set of formal symbols

$R[x] = \{a_n x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0 \mid a_i \in R, n \text{ is nonnegative integer}\}$

is called the ring of polynomials over *R* in the indeterminate x.

Ideal

A subring A of a ring R is called a (two-sided) ideal of R if for every $r \in R$ and every $a \in A$ both ar and ra are in A.

Basis

Let $F = \{f_1, \ldots, f_s\}$ be a set of multivariate polynomials. Then the ideal generated by F_1 denoted by $I = \langle F \rangle$ is given by:

$\{\sum_{i=1}^{s} h_i f_i \quad h_1, \ldots hs \in k [x_1,x_2,\ldots x_n]\}$

The polynomials $f_1,\ldots, f_s$ are called a basis for the ideal they generate, and since F is finite, we say the ideal is finitely generated.

Variety

The variety of $\mathcal{F}$ is the set of all common zeros:

$$V(\mathcal{F}) = \{(z_1, \ldots.z_n) \in \mathbb{R}^n : f(z_1,\ldots, z_n) = 0 \text{ for all } f \in \mathcal{F}\}$$

Hilbert's Basis Theorem

Hilbert's Basis Theorem states that "Every ideal generates the finite basis"

S-polynomial

Given two polynomials $f, g \in k[x_1, x_2\ldots, x_n]$. Let J = l.c.m (leading monomial (f), leading monomial(g)). We define s-polynomial of f and g as the linear combination.

$$\text{s-poly}(f, g) = \frac{J}{\text{Leading Term (f)}} \cdot f - \frac{J}{\text{Leading Term (g)}} \cdot g$$

Monomial

Given a nonzero polynomial $f \in k[x_1, x_2,\ldots,x_n]$, we define:

• The multidegree of f as: $\text{multideg}(f) = \max(\alpha \in N^n : a_\alpha \neq 0)$.

• The leading monomial of f as: $LM(f) = x^{\text{multideg}}(f)$

.

• The leading coefficient of f as: $LC(f) = a_{\text{multideg}}(f)$.

• The leading term of f as: $LT(f) = LC(f)\cdot LM(f)$.

Lexicographic Order

Let α and β be in N^n. α >lex β if and only if the left-most nonzero entry in α − β is positive.

Graded Lex Order

α $>_{grlex}$ β if and only if |α| > |β| or (|α| = |β| and α $>l_{ex}$ β).

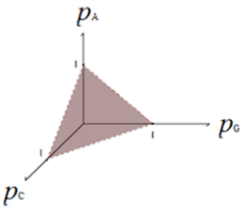
Graded Reverse Lex Order

α $>_{grevlex}$ β if and only if |α| > |β| or (|α| = |β| and the right-most nonzero entry in α − β is negative.

3 Gröbner Basis

Now, let us think about the correspondence between the algebra and the geometry.
We can take these probability distribution and embed into a geometric figure as follows:

$\{(p_A, p_C, p_G, p_T) | \sum = 1\}$ ⟷



The above figure lives in $\mathbb{R}^3$. Any point on this figure (inside or in the boundary) corresponds to the probability distribution for (p_A, p_C, p_G, p_T). Varying the parameters θ_1 and θ_2 gives a subspace of the tetrahedron.

"**Every ideal corresponds to a variery and every variety corresponds to an ideal**"

A Gröbner basis is a set of multivariate polynomials that has enticing algorithmic and algebraic properties allowing us to analyze situations like the above. Every set of polynomials can be converted into a Gröbner basis.

The theory of Gröbner is based upon the concept of ideals generated by finite sets of multiple polynomials. So, we need to know some definitions, which are listed below:

Let F be a finite set of polynomials. Then F is a Gröbner basis if and only if every S-polynomial has remainder zero when divided by F. When F is a Gröbner basis division always produces unique remainders which we can use as a normal form for a polynomial's equivalence class modulo the ideal generated by F. Gröbner bases also allow us to solve systems of equations and to analyze varieties in an algebraic setting.

To calculate the Gröbner basis, Buchberger invented an algorithm called Algorithm Buchberger which is as follows:

4 Algorithm Buchberger

The algorithm is used to calculate Gröbner's basis. The simplest algorithm to describe, and the basis for many optimizations, is "Algorithm Buchberger" which is as follows:

Input: A polynomial set $F = (f_1,...,f_n)$ that generates an ideal I.
Output: A Gröbner basis $G = (g_1,...,g_t)$ that generates the same ideal I with $F \subset G$.

$G := F$
$M := \{\{f_i, f_j\} \mid f_i, f_j \in G \text{ and } f_i \neq f_j\}$
Repeat
$\{p, q\}$:= a pair in M
$M := M - \{\{p, q\}\}$
$S := Spoly(p,q)$
$h := NormalForm(S,G)$
IF $h \neq 0$ THEN
$M := M \cup \{\{g, h\} \forall g \in G\}$
$G := G \cup \{h\}$
Until $M = \emptyset$

In this algorithm, the set of polynomials F, the s-polynomial gets added till the remainder of the set of function is equal to zero. Then, all the functions of G is the equivalent Gröbner basis *G* of the given function F.

Since, the algorithm takes a long time to process, we can modify into a different algorithm with variable ideas and make it process much faster, which is given below:

5 Algorithm Buchberger (modified)

To modify the algorithm, we need to put some of the ideas of Buchberger's Criterion. The three criterion of Buchberger are as follows:

(i) **Buchberger's Criterion I** : In the process of picking a pair $\{f_i, f_j\}$, choose a pair $\{f_i, f_j\}$ such that LCM(LM(fi),LM(fj)) is minimal among all the pairs.

(ii) **Buchberger's Criterion I**I: There are s-polynomials that may be ignored and we do not need to compute their normal form because they are guaranteed to reduce to zero modulo F. If the LM(fi) and LM(fj) are relatively prime, then s-poly(fi, fj) reduces to 0 modulo F. Thus, pick a pair {fi, fj} such that LM(fi) and LM(fj) are not relatively prime.

(iii) **Buchberger's Criterion III** : If there is an element f_k of the basis such that the $LM(f_k)$ divides $LCM(LM(f_i),LM(f_j))$ and if the S-poly(f_i, f_k) and the S-poly(fj , fk) have already been considered, then S-poly(f_i, f_j) reduces to zero and hence could be ignored.

Input: A polynomial set $F = (f_1,...,f_n)$ that generates an ideal I.
Output: A Gröbner basis $G = (g_1,...,g_t)$ that generates the same ideal I with $F \subset G$.

G := F
M := {{f_i, f_j}|1 ≤ i<j ≤ s}
t := s
Repeat
{f_i, f_j} := a pair in M
IF (LCM(LM(f_i),LM(f_j)) 6= LM(f_i)·LM(f_j)) AND
NOT(Criterion(f_i, f_j ,M) then
S := S-polynomial(f_i, f_j)
h := NormalForm(G,S)
IF h ≠ 0 THEN
t := t + 1
f_t := h
M := M S{{i, t}∀1 ≤< i ≤ t − 1}
G := G U S{f_t}
M := M − {{f_i, f_j}}
Until M = ∅

where Criterion(f_i, f_j ,M) is true provided the conditions in above are met.

This is the advanced form of the algorithm, where we consider all of the Buchberger's criterion and reduce our time to calculate the Gröbner basis. This is the algorithm used nowadays, to calculate the Gröbner basis as it is much more efficient and widely accepted. This algorithm is also known as Francis's Algorithm.

5 Application

Since each characters are generated independently, likelihood function is given by :-

$$\mathcal{L} = p_C p_T p_C p_A p_C p_G \ldots\ldots p_A = p_A^{10} \cdot p_C^{14} \cdot p_G^{15} \cdot p_T^{10}$$

Now, log-likelihood is given by:-

$$\ell\ (\theta_1, \theta_2) = \log(\mathcal{L}(\theta_1, \theta_2))$$

$$= 10 \cdot \log(p_A(\theta_1, \theta_2)) + 14 \cdot \log(p_C(\theta_1, \theta_2)) + 15 \cdot \log(p_G(\theta_1, \theta_2)) + 10 \cdot \log(p_T(\theta_1, \theta_2))$$

The solution to this optimization problem can be computed in closed form, by equating the two partial derivatives of the log-likelihood function to zero:

$$\frac{\partial l}{\partial \theta 1} = \frac{10}{p} \cdot \frac{\partial p}{\partial \theta 1} + \frac{14}{p} \cdot \frac{\partial pC}{\partial \theta 1} + \frac{15}{p} \cdot \frac{\partial pG}{\partial \theta 1} + \frac{10}{p} \cdot \frac{\partial pT}{\partial \theta 1} = 0,$$

$$\frac{\partial l}{\partial \theta 2} = \frac{14}{p} \cdot \frac{\partial pA}{\partial \theta 2} + \frac{14}{p} \cdot \frac{\partial pC}{\partial \theta 2} + \frac{15}{p} \cdot \frac{\partial pG}{\partial \theta 2} + \frac{10}{p} \cdot \frac{\partial pT}{\partial \theta 2} = 0.$$

Using Gröbner bases, we obtain

$13003050\ \theta_1 + 2744\ \theta_2^2 - 21161250\theta_2 - 6290625 = 0$ .....(i)

$134456\ \theta_2^3 - 10852275\ \theta_2^2 - 43047281250\theta_2 + 935718750 = 0$ .....(ii)

Solving equation (i) and (ii), we get

$$(\hat{\theta}_1, \hat{\theta}_2) = (0.5191263945, 0.2172513326).$$

The log-likelihood function attains its maximum value at this point:

$$\ell(\theta_1, \theta_2) = -67.08253037.$$

The corresponding probability distribution

$$(\hat{p}_A, \hat{p}_C, \hat{p}_G, \hat{p}_T) = (0.202432, 0.289358, 0.302759, 0.205451)$$

is very close to the empirical distribution

$\frac{1}{49}(10, 14, 15, 10) = (0.204082, 0.285714, 0.306122, 0.204082)$.

Thus, we can conclude that the the maximum value estimator for our θ_1 and θ_2 are close to the values we got from the empirical distribution.

6 References

[1] B. Buchberger. A Criterion for Detecting Unnecessary Reductions in the Construction of Grobner Bases. Lecture Notes in Computer Science, vol. 72. Springer-Verlag, 1979.

[2] Pachter, Lior, and Bernd Strumfels. *Algebraic Statistics for Computational Biology*. Cambridge: Cambridge UP, 2005. Print.

[3] T. Becker and V. Weispfenning. Grobner Bases: A Computational Approach to Commutative Algebra, Springer-Verlag, 1993.

[4] Watkins, David S. "Francis's Algorithm." *The American Mathematical*

*Monthly*118.5 (2011): 387-403. Web.