Regression trees for analysis of mutational spectra in nucleotide sequences

Vladimir B. Berikov¹,* and Igor B. Rogozin²,³

¹Institute of Mathematics and ²Institute of Cytology and Genetics, Siberian Branch of the Russian Academy of Sciences, Novosibirsk 630090, Russia and ³Institute of Molecular Evolutionary Genetics and Department of Biology, Pennsylvania State University, 322 Mueller Laboratory, University Park, PA 16802-5301, USA

Received on November 30, 1998; revised and accepted on February 17, 1999

Abstract

Motivation: The study and comparison of mutational spectra is an important problem in molecular biology, because these spectra often reveal important features of the action of various mutagens and the functioning of repair/replication enzymes. As is known, mutability varies significantly along nucleotide sequences: mutations often concentrate at certain positions in a sequence, otherwise termed ‘hotspots’.

Results: Herein, we propose a regression analysis method based on the use of regression trees in order to analyse the influence of nucleotide context on the occurrence of such hotspots. The REGRT program developed has been tested on simulated and real mutational spectra. For the G:C→T:A mutational spectra induced by Sn1 alkylating agents (nine spectra), the prediction accuracy was 0.99.

Availability: The REGRT program is available upon request from V.Berikov.

Contact: berikov@math.nsc.ru

Introduction

With advancements in molecular biology, a large body of experimental data has been acquired on mutations in DNA. Data of this specific kind are otherwise known as ‘mutational spectra’. A mutational spectrum is often a quite informative feature of the functioning of various repair/replication enzymes. It is now believed that mutability varies significantly along nucleotide sequences: mutations, whether induced or spontaneous, occur at higher frequencies at certain positions of a nucleotide sequence (Benzer, 1961; Coulondre et al., 1978; Foster et al., 1982; Betz et al., 1993). Study of these positions (otherwise termed ‘hotspots’) suggests that there might be an association between the mutations observed and the features of DNA primary structure (otherwise called ‘context’) near the hotspots. Figure 1 exemplifies a mutational spectrum (Fowler and Schaaper, 1997). As can be seen, mutations occur much more often at some positions than at the others. In many cases, the reason for hotspots in a site is certain combinations of neighbouring bases (as reviewed by Horsfall et al., 1990; Boulikas, 1992). For example, it is the neighbourhood that accounts for a higher mutability of the Rg sites (hotspots G:C→A:T in g; hereinafter, a lower case letter stands for a mutable position, R = A or G) with respect to mutational spectra induced by some alkylating agents (Horsfall et al., 1990).

The problem of revealing hotspot context is normally addressed using two methods. Stormo et al. (1986) used multiple linear regression analysis to see how the context affects the mutability of different positions in the lacI gene in Escherichia coli cells treated with 2-aminopyrine. Data obtained indicate that the bases at positions −2 and −1 relative to the mutating base most strongly affect the frequency of mutations. However, it is assumed within the framework of the method that there is a linear correlation between the frequency of mutations in positions and the context factors, and that the factors are distributed normally. The problem is that in many, if not all, cases, neither assumption fits the context features of real mutational spectra.

Rogozin and Kolchanov (1992) employed the heuristics classification approach and the Monte-Carlo procedure to classify and build consensus of hotspots. The procedure of building a consensus (the rule that defines the hotspot context) is based on assessing the non-randomness of the bases in the neighbourhood of the hotspots. The statistical significance of the constructed set of consensus sequences is estimated using the Monte-Carlo procedure. A statistically significant description of the context of hotspots of somatic mutations (consensus sequences RgYW and TaA, R = A or G, Y = T or C, W = A or T) in immunoglobulin genes has become possible with this approach. It has been shown that these consensus sequences are characteristic of somatic mutations in various organisms (Rogozin et al., 1996).

*To whom correspondence should be addressed
For analysis of context influence on mutations, it seems very promising to apply regression trees (Breiman et al., 1984). Models such as these are well substantiated mathematically and do not place restraints on the variables being analysed, as the heuristic approaches do, which allows both quantitative and qualitative features of the context of mutational events to be analysed. The results presented allow regression analysis to be recommended as a tool for studying context features of the mutational spectra.

Materials and methods

Most generally, a mutational spectrum is described as follows: a large number of copies of a sequence are exposed to a mutagen (an environmental factor). Either one or none of mutation occurs in each copy. The mutants are revealed by a mutagen (an environmental factor). Either one or none of mutation occurs in each experiment. Instead of sequence of length 554 positions, the occurrence of more than one mutation in a copy is a complex case (because the mutational events are no longer independent), which requires special investigation.

Mathematical model

Consider the following mathematical model. Let \( g \) be a sequence of length \( K \): \( g = (g_1, g_2, \ldots, g^K) \), where \( g^m \in D_g \), \( m = 1, \ldots, K \); the set \( D_g \) is an unordered set of values of the sequence elements (‘symbols’). Analysing the genetic sequences \( D_g = \{A, C, T, G\} \), we assume that at a fixed set of conditions (‘experiment’) the elements can change their value (a mutation occurs at the position that matches an element). Here it is probable that \( g^m (g^m \in D_g, g^m \neq g^m) \) occurs instead of \( g^m \).

The following assumptions are made.

1. One mutation occurs in each experiment.
2. The probability of a mutation at the \( m \)th position is only dependent on \( 2r \) nearest left and right neighbouring elements (‘context’): \( a^m = (g^{m-r}, g^{m-r+1}, \ldots, g^{m-1}, g^m, g^m+1, \ldots, g^{m+r-1}, g^{m+r}) \), where \( r \) is a preset value, \( m = r+1, \ldots, K-r \). Thus, we are analysing mutable positions together with the neighbouring sequences (hereinafter referred to as mutation sites) (Figure 1).
3. The probability of a mutation at the \( m \)th position is not dependent on \( m \) (the index that matches the number of the position at which the mutation has occurred will sometimes be left out hereinafter).

Suppose that the observed frequency of mutations after \( T \) independent recurrent experiments on the given sequence is \( y_1, y_2, \ldots, y^K \). Let \( Y \) be a random variable standing for the number of mutations that have occurred; \( D_Y = R^+ \) is the domain of \( Y \). Let \( X_1, X_2, \ldots, X_n \) be a set of features that describes a site \( a^m \). The following features can be used as \( X_1, X_2, \ldots, X_n \): nucleotides in different positions, frequency of nucleotides in a site, frequency of dinucleotides in a site, etc. Note that each feature is either numerical or non-numerical (nominal, Boolean).

Let \( D_L \) be the domain of \( X_j, j = 1, \ldots, n \). Denote \( x^m \) as \( X(a^m) = (X_1(a^m), X_2(a^m), \ldots, X_n(a^m)) \), where \( X_j(a^m) \) is the value of \( X_j \) for the set \( a^m \). Assigning \( y^m \) to each \( x^m, m = r+1, \ldots, K-r \), yields a data table \( V = \{(x^m, y^m)\} \) with \( N = K - 2r \) observations. The index \( i \) is used throughout to enumerate these observations: \( V = \{(x_i, y_i)\}, i = 1, \ldots, N \). Suppose we need to derive a decision function for predicting how many mutations will occur depending on contextual features. The decision function is the mapping of the set \( D = D_1 \times \ldots \times D_k \) into the set \( D_Y \). The decision function allows prediction of the expected number of mutations on the basis of the description of any possible context.

Let the choice of the context be a random event; then the risk function is \( Q(f) = E(Y - f(X))^2 \). The decision function \( f \) that minimizes \( Q(f) \) is the optimum decision function. While distributions are unknown, an estimate of \( Q(f) \), \( d_k = (1/N)\sum_{i=1}^{N}(y_i - f(x_i))^2 \), is used.
Regression analysis of mutational spectra

\[-f(x_i)\] is used as the function of empirical risk. Denote the class of decision functions as \( \Phi \). Also, suppose there is a criterion for comparing various decision functions of the class. Now one can find the optimum decision function within \( \Phi \).

Let us consider the class of logical decision functions (Lbov, 1989). The logical decision function is defined by pair \( \langle \varphi, d_{\varphi} \rangle \), where \( \varphi = \{ E_1, \ldots, E_M \} \) is a partition of the set \( D \) into \( M \) subsets, \( E^S = E_1^S \times \ldots \times E_j^S \times \ldots \times E_M^S \). \( E_j^S \) is an interval if \( X_j \) is numeric, or an arbitrary subset of values if otherwise. Let us consider the simplest model of regression analysis: for any \( \varphi \) the set of decisions \( t_{\varphi} = \{ y^1, \ldots, y^S, \ldots, y^M \} \), where \( y^S \) is a certain constant, \( y^S \in D_y \). This class has been chosen for the following reasons:

1. Both numerical and non-numerical features fit a decision function.
2. Any decision function is a logical-and-probabilistic model, which describes cause-and-effect associations between the features of the context and the probability of mutation.
3. Easy to handle.

The partition \( \varphi \) is most conveniently described as a tree (Breiman et al., 1984; Lbov and Berikov, 1993). By a tree we mean a rooted (possibly non-dichotomous) tree for which every inner node is in correspondence with a certain feature \( X_j \); branches coming from the inner node are in correspondence with \( E_j \in E_j^S \) statement, where \( E_j^S \) has the above-mentioned type (i.e. an interval if \( X_j \) is numeric, or an arbitrary subset of values if otherwise); \( t = 1, \ldots, l_j \), where \( l_j \) is the number of branches coming from the node; additionally, \( E_j^1 \cup \ldots \cup E_j^{l_j} = D_j \). To each \( S \)-th terminal node a decision is assigned, namely \( y^S \in D_y \), \( S = 1, \ldots, M \), where \( M \) is the number of terminal nodes. The following logical statement matches a chain of branches connecting the root node and the \( S \)-th terminal node:

\[ f^S = \text{IF } x_j \in E_j^{l_1} \text{ AND } x_{j_2} \in E_{j_2}^{l_2} \text{ AND } \ldots \text{ AND } x_{j_m} \in E_{j_m}^{l_m} \text{ THEN } Y = y^S, \]

where \( m^S \) is the length of the chain.

Figure 2 exemplifies a regression tree. In this example, features \( X_1, X_2, X_3 \) are non-numerical (nominal); \( D_j = \{ A, C, T, G \} \) for all \( j \).

Denote the total variance as \( d_0 = N^{-1} \sum (y^i - \bar{y})^2 \) and the residual variance which corresponds to the decision tree \( f \) as \( d_f = N^{-1} \sum \sum_i (y^i - y^S)^2 \). Here \( y = N^{-1} \sum_i y^i \), \( y^S \) is the subset of observations that matches the \( S \)-th terminal node \( B^S \) (i.e. satisfies \( f^S \)). Denote the number of observations from \( y^S \) as \( N^S \). The value \( d_f \) is an estimate of the risk function \( Q(f) \). The empirical risk should be minimized, and so too the complexity of the decision function (the more terminal nodes in the decision tree \( f \), the more complex the decision function). This demand may be due to the following reasons:

1. It is known (Breiman et al., 1984) that at a limited sample size and a large dimensionality of the space of features, a further growth in the complexity of the decision function class may adversely affect decision quality (the probability of a decision function being strongly different from the optimum is higher).
2. It might be desirable to simplify the logical model of a phenomenon.

Consider the expression: \( q_f = d_f/d_0 + \beta (M/N) \), where \( \beta > 0 \) is a parameter. When \( q_f \) reaches its minimum, a ‘compromise’ is attained between the empirical risk and the complexity of the decision. Herein, \( q_f \) is used as a criterion for comparing decision trees and choosing the best tree. The criterion in the given form allows efficient methods of dynamic programming to be applied in the search for the optimum decision tree.

An algorithm of constructing a decision tree for predicting the frequency of mutations

The available methods for decision tree construction in the problems of regression and discriminant analysis can be mainly classified as accurate and heuristic. The accurate methods (dynamic programming, branch-and-bound and exhaustive search) are inefficient on a large volume of data and a large number of features. Because genetic sequences are normally long enough, the heuristic methods appear to be a better choice. Herein, we will describe a modification of dynamic programming that allows a nearly optimal decision to be made (Lbov and Berikov, 1993). The common feature of most approximate methods is the so-called ‘top-down’ pro-
cess of successive independent branching of promising nodes.

Let \( f \) be the decision tree at a stage of construction (it was started from a single root node). The basic steps of the tree construction algorithm are as follows.

1. Ordering terminal nodes with respect to degree of promise. Are the available terminal nodes promising for further branching? To answer the question, a criterion (see the section below) is applied. Those found promising are ordered with respect to the degree of promise.

2. Decision making. To the nodes found unpromising (leaves), a decision is attributed (the anticipated number of mutation), and the nodes are omitted from consideration. The decision attributed to the \( S \)th node is \( y^{\delta} = (1/N^S) \sum_{i \in V^S} y_i \).

3. Check of tree completeness. If there are no more promising nodes, the tree is assumed complete.

4. Constructing a subtree. Given the most promising node, a subtree, which can be either optimum or nearly so, is constructed on its data set. The subtree then replaces the node (call this procedure ‘branching’) and the process restarts from step 1. The algorithm is described below (see the subsequent section Constructing a subtree).

**Is that node promising?**

Let \( B^S \) be the node for which it is required to assess the promise of further branching. The set of observed mutation frequencies \( \{y^{iS}, \ldots, y^{mS}, \ldots, y^{NS}\} \) corresponds to this node. Consider a hypothesis of homogeneity, which postulates that differences between mutation frequency among \( N^S \) sites are statistically insignificant. If the hypothesis fits the observations, the node in question will be considered as not needing further branching. The hypothesis is tested by Pearson’s \( \chi^2 \) statistic: let \( \chi^2 = \sum_{i \in V^S} (y^{im} - y^{\delta})^2/y^{\delta} \), where \( y^{\delta} \) is the expected mutation frequency. The distribution of \( \chi^2 \) can be regarded as nearly matching the distribution \( \chi^2 \) with \( N^S - 1 \) degrees of freedom, given the hypothesis is true and \( y^{\delta} \) is not too low (e.g. not less than 5). Given a significance level \( \alpha \) and the tables of distribution \( \chi^2 \), it is possible to see how well the hypothesis of homogeneity fits the observations.

If the expected frequency is low, homogeneity is tested using a Monte-Carlo procedure. Stable and quite powerful, this test is recommended if the number of zero rows is large (Piegorsch and Bailer, 1994), which (known as ‘sparseness’) is typical of real mutational spectra. Consider a random event in which a mutation has occurred has been selected. Repeat the simulation a set number of times (e.g. 100), estimate the probability \( p \) of the event \( \chi^2 \geq \chi^2 \). The observations corresponding to the \( S \)th node will be considered homogeneous, if \( p \geq 1 - \alpha \), where \( \alpha \) is the preset significance level.

If the inference is that the node requires further branching, \( d^S = (1/N^S) \sum_{m} (y^{im} - y^{\delta})^2/y^{\delta} \) will stand for the degree of promise. Thus, the degree of promise means the empirical variance of the subset in a node, which potentially can be ‘explained’ by the subtree constructed for the node.

**Constructing a subtree**

Consider the following algorithm of constructing recursively a nearly optimum subtree. Define the following operations with decision trees.

1. Constructing the ‘initial’ tree for feature \( X_i \) and data set \( V \). Let some feature \( X_i \) \( (1 \leq j \leq n) \) be fixed. In the case of a numerical feature, the interval of its variation \( D_j \) is preliminarily divided into \( L_j \) disjoint intervals through placing boundaries (coordinates of the boundaries are equal to the arithmetic average of neighbouring non-identical values of feature \( X_i \) for the given data set). To each \( j \)th interval \( E_j \) (if the feature is non-numerical, it is its value) there corresponds node \( B_j \) of the tree with root node \( B \). The allowable level of the allowable pair of nodes. A new node, \( B_j^{p,q} \), replaces nodes \( B_j^p \) and \( B_j^q \) of the same layer of the tree \( (1 \leq p, q \leq L_j) \). The corresponding data sets are therefore united. If the feature is numerical, union is only applicable to the nodes that correspond to the adjoining intervals (this pair of nodes is called allowable); if otherwise, no restrictions are placed.

2. Union of the allowable pair of nodes. A new node, \( B_j^{p,q} \), replaces nodes \( B_j^p \) and \( B_j^q \) of the same layer of the tree \( (1 \leq p, q \leq L_j) \). The corresponding data sets are therefore united. If the feature is numerical, union is only applicable to the nodes that correspond to the adjoining intervals (this pair of nodes is called allowable); if otherwise, no restrictions are placed.

3. Recursive construction of the tree. For each terminal node of the initial tree and for each new node formed by the united nodes, an optimal (or nearly optimal) subtree is constructed recursively. The point is that the same algorithm is applied to the construction of a tree with root node \( B \) and to those subtrees whose root nodes are daughter ones to \( B \). The allowable level of recursive inclusion is determined by a preset value \( R \) (Figure 4). Recursive construction does not apply to the nodes that have proven unpromising.

The algorithm is therefore as follows. Construct an initial tree with root node \( B \), then, recursively, optimal (nearly optimal) subtrees. Unite in a pairwise manner the nodes that are recurrent independent tests, \( m = 1, \ldots, N^S \). Let \( \chi^2 = \sum_{m} (y^{im} - y^{\delta})^2/y^{\delta} \). Repeating the simulation a set number of times (e.g. 100), estimate the probability \( p \) of the event \( \chi^2 \geq \chi^2 \). The observations corresponding to the \( S \)th node will be considered homogeneous, if \( p \geq 1 - \alpha \), where \( \alpha \) is the preset significance level.

If the inference is that the node requires further branching, \( d^S = (1/N^S) \sum_{m} (y^{im} - y^{\delta})^2/y^{\delta} \) will stand for the degree of promise. Thus, the degree of promise means the empirical variance of the subset in a node, which potentially can be ‘explained’ by the subtree constructed for the node.

**Constructing a subtree**

Consider the following algorithm of constructing recursively a nearly optimum subtree. Define the following operations with decision trees.

1. Constructing the ‘initial’ tree for feature \( X_i \) and data set \( V \). Let some feature \( X_i \) \( (1 \leq j \leq n) \) be fixed. In the case of a numerical feature, the interval of its variation \( D_j \) is preliminarily divided into \( L_j \) disjoint intervals through placing boundaries (coordinates of the boundaries are equal to the arithmetic average of neighbouring non-identical values of feature \( X_i \) for the given data set). To each \( j \)th interval \( E_j \) (if the feature is non-numerical, it is its value) there corresponds node \( B_j \) of the tree with root node \( B \). The allowable level of the allowable pair of nodes. A new node, \( B_j^{p,q} \), replaces nodes \( B_j^p \) and \( B_j^q \) of the same layer of the tree \( (1 \leq p, q \leq L_j) \). The corresponding data sets are therefore united. If the feature is numerical, union is only applicable to the nodes that correspond to the adjoining intervals (this pair of nodes is called allowable); if otherwise, no restrictions are placed.

2. Union of the allowable pair of nodes. A new node, \( B_j^{p,q} \), replaces nodes \( B_j^p \) and \( B_j^q \) of the same layer of the tree \( (1 \leq p, q \leq L_j) \). The corresponding data sets are therefore united. If the feature is numerical, union is only applicable to the nodes that correspond to the adjoining intervals (this pair of nodes is called allowable); if otherwise, no restrictions are placed.

3. Recursive construction of the tree. For each terminal node of the initial tree and for each new node formed by the united nodes, an optimal (or nearly optimal) subtree is constructed recursively. The point is that the same algorithm is applied to the construction of a tree with root node \( B \) and to those subtrees whose root nodes are daughter ones to \( B \). The allowable level of recursive inclusion is determined by a preset value \( R \) (Figure 4). Recursive construction does not apply to the nodes that have proven unpromising.

The algorithm is therefore as follows. Construct an initial tree with root node \( B \), then, recursively, optimal (nearly optimal) subtrees. Unite in a pairwise manner the nodes that are recurrent independent tests, \( m = 1, \ldots, N^S \). Let \( \chi^2 = \sum_{m} (y^{im} - y^{\delta})^2/y^{\delta} \). Repeating the simulation a set number of times (e.g. 100), estimate the probability \( p \) of the event \( \chi^2 \geq \chi^2 \). The observations corresponding to the \( S \)th node will be considered homogeneous, if \( p \geq 1 - \alpha \), where \( \alpha \) is the preset significance level.

If the inference is that the node requires further branching, \( d^S = (1/N^S) \sum_{m} (y^{im} - y^{\delta})^2/y^{\delta} \) will stand for the degree of promise. Thus, the degree of promise means the empirical variance of the subset in a node, which potentially can be ‘explained’ by the subtree constructed for the node.
Fig. 3. Example of initial tree for the numerical feature $X_j$. Nodes $B_j^1$, ..., $B_j^{L_j}$ correspond to the intervals $E_j^1$, ..., $E_j^{L_j}$. Observations of feature $X_j$ are denoted as: $\times$.

Fig. 4. Recursive construction of the tree with root node $B$. The allowable level of recursive inclusion is $R = 3$. Subtree for $B_j^2$ is constructing; this subtree results from constructing the subtrees for nodes, which are daughter ones to $B_j^2$. daughter ones to $B$, which warrant the best values of the criterion at the union and recursive construction of the corresponding subtree. The algorithm of constructing a subtree with root node $B$ has the following steps to follow.

0. Set $j = 1$.
1. Fix the feature $X_j$.
2. Construct an initial subtree for feature $X_j$ and data set $V$. Let $L_j$ be a number of daughter nodes $B_j^t$ ($t = 1, \ldots, L_j$) of the initial tree, and $V_j^t$ be the subset of data which matches $B_j^t$.
3. From amongst nodes $B_j^t$, leaves are selected (see the previous section Is that node promising?).
4. For the nodes $B_j^t$, that are not leaves, optimal (or nearly optimal) subtrees are recursively constructed (the recursive procedure repeats steps 0–15; at this node $B_j^t$ stands for $B$, and $V_j^t$ stands for $V$).
5. Calculate the value of quality criterion for the constructed subtree with root node $B$.
6. Another pair of nodes, allowed for the union, $B_j^p$ and $B_j^q$ ($1 \leq p, q \leq L_j$) is united into another node, $B_j^{p,q}$.
7. Ensure that node $B_j^{p,q}$ is not a leaf.
8. If $B_j^{p,q}$ is not a leaf, a subtree with root node $B_j^{p,q}$ is recursively constructed. If $B_j^{p,q}$ is a leaf, then the subtree is not constructed.
9. Calculate the value of quality criterion for the subtree with root node $B$ and the new node $B_j^{p,q}$.
10. If there are no more nodes, allowed for union, then 11 else 6.
11. Save the best subtree.
12. The best subtree replaces the initial one; $L_j := L_j - 1$.
13. If $L_j = 1$, then go to 14, else go to 6, ignore the pairs of nodes that have been considered.
14. If $j = n$, then 15, else $j := j + 1$ go to 1.
15. The subtree is complete.

Again, the algorithm of subtree construction is applied in step 4 of the basic scheme of decision tree construction (see the previous section An algorithm of constructing a decision tree for predicting the frequency of mutations).

*Note 1.* If the root nodes of subtrees under construction are subordinate to node $B$, the feature $X_j$ as fixed at the previous stage may be excluded from further consideration (it is of no use because ‘smaller’ divisions with respect to this feature have been considered).

*Note 2.* When the features in consideration are non-numerical, observations of their values in node $B$ may not yield the given value of $D_j$. If so, such values will be considered to be united with the one for which the number of observations is maximal.

In contrast to other algorithms, ours allows:

1. Complexity of analysis (allowable level of recursive inclusion) to be increased so that not single-layered, but many-layered subtrees are being constructed. It allows one to handle more complex effects of the features on the value being predicted.
2. The level of branching is optimized in each node of the decision tree, which makes it possible to involve a wider class of trees (at the same complexity of analysis) and thus to improve decision quality.

3. A compromise to be found between the allowable level of recursive inclusion (and therefore decision quality) and available computer storage and time, and additional capacities (storage and time) to be involved by increasing the level.

Parameters of the tree construction algorithm

The proposed algorithm has several parameters. Selection of the parameters is a complex heuristic procedure. However, the experience of dealing with applied problems and computer simulation practice suggests that there are the most frequent ranges of values of the parameters at which decision quality is reasonable. Consider the parameters of the algorithm.

1. $\beta$ is the parameter of ‘the ratio of the empirical risk and complexity’: the higher $\beta$, the simpler laws and the higher the empirical risk; the lower $\beta$, the more complex laws and the lower the empirical risk. The recommended value for $\beta$ is 0.1–1.5 (1.0 in this study).

2. $R$ is the level of recursive inclusion, it sets the complexity of analysis (the number of features in the combination considered during subtree construction). The higher $R$, the higher decision quality, calculation time and storage. To save working time and computer storage, the algorithm has been modified so that $R$ cannot only be integer, but also fractional. The integer sets the warranted recursive inclusion ($R$). The fractional sets the share of the most informative features to be selected for testing different combinations of $R + 1$, $R + 2$, …, etc., features selected at the previous stage. Here the measure of informativeness of a feature is assumed the mean value of the quality criterion for the decision trees in which the feature is involved. The recommended value for $R$ is 1–2.5 (2.5 in this study).

3. $\alpha$ is the confidence level used when the homogeneity hypothesis is being tested. The recommended value for $\alpha$ is 0.01–0.05 (0.05 in this study).

The logic decision functions can be presented in the form of consensus sequences that are easy to describe and interpret. How a set of logic decision functions (consensus sequences) can be created is illustrated in Figure 5 (positions –2 to +2 are shown).

Mutational spectra were retrieved from the Database of Mutational Spectra (ftp.bionet.nsc.ru/pub/biology/dbms) (Rogozin et al., 1990). In this work, we omitted from consideration possible mutational asymmetry of two DNA strands, and all mutations at bases C and T were transformed into a complementary form (e.g. for position 120, site TTcTG was transformed into CAgAA; here and throughout the mutable position is designated by a lower case letter). In this work, we only consider base substitutions in a sequence, although the method is applicable to other classes of mutational events.

For one-letter designations of the nucleotide groups that occur at the positions of mutable sites, the commonly adopted nomenclature is used (A, T, G, C, W = A or T, S = G or C, R = A or G, Y = T or C, M = A or C, K = G or T, B = T or G, V = A or G or C, H = A or T or C, D = A or T or G, N = A or T or G or C) (Cornish-Bowden, 1985).

The REGRT program is available upon request from V. Berikov (berikov@math.nsc.ru) or I. Rogozin (rogozin@bionet.nsc.ru).

Results

Testing the algorithm

To test the approach, we analysed nine mutational spectra induced by the Sn1 alkylating agents in the lacI gene of E. coli: N-nitroso-N-methyl-N-α-acetoxyethylamine (NMAM) (Horsfall and Glickman, 1989), N-nitroso-N,N-dimethylamine (NDMA) (Horsfall et al., 1989), N-nitroso-N-methyl-1-N-α-acetoxybenzylamine (NMAB) (Horsfall and Glick- man, 1988) and N-methyl-N-nitro-N-nitrosoguanidine (MNNG) (Burns et al., 1987; Gordon et al., 1988, 1990). These mutagens are well studied, because they are classical
Regression analysis of mutational spectra

Most of the Sn1-induced mutations are G:C→A:T transitions. These spectra are briefly described in Table 1. It is noteworthy that in most of experiments (except ECLAC20 and ECLAC21), mutations were revealed in the first 180 bases of the coding sequence of the lacI gene (lacI^-d experimental test system). Only ‘available’ positions have been analysed, i.e. those leading to a mutant phenotype and then revealed by sequencing (Horsfall et al., 1990). As is known, the frequency of such mutations is affected by a local context. Mutations occur several times as frequently in the Rg site (g is the mutable position) as in Yg (Horsfall et al., 1990) (Table 1). Thus, these mutational spectra are a mix of at least two classes of sites. It is believed that this is a consequence of a similarity between the mechanisms of mutation induction (Sn1 mechanism).

Table 1. Description of mutational spectra induced by Sn1 alkylating agents in the lacI gene

<table>
<thead>
<tr>
<th>Spectrum name (mutagen, dose, strain)</th>
<th>Number of mutations (T)</th>
<th>Number of sites</th>
<th>Reference</th>
<th>Average number of mutations in Rg sites (y_Rg)</th>
<th>Average number of mutations in Yg sites (y_Yg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECLAC12A (NMAM)</td>
<td>76</td>
<td>22</td>
<td>Horsfall and Glickman, 1989</td>
<td>9.6</td>
<td>0.5</td>
</tr>
<tr>
<td>ECLAC16A (NMAB, 1 µM)</td>
<td>111</td>
<td>22</td>
<td>Horsfall and Glickman, 1988</td>
<td>10.6</td>
<td>1.9</td>
</tr>
<tr>
<td>ECLAC16B (NMAB, 0.1 µM)</td>
<td>58</td>
<td>22</td>
<td>Horsfall and Glickman, 1988</td>
<td>5.6</td>
<td>0.9</td>
</tr>
<tr>
<td>ECLAC17A (NDMA)</td>
<td>105</td>
<td>22</td>
<td>Horsfall et al., 1989b</td>
<td>10.4</td>
<td>1.6</td>
</tr>
<tr>
<td>ECLAC20 (MNNG, recA^-)</td>
<td>258</td>
<td>61</td>
<td>Gordon et al., 1988</td>
<td>10.1</td>
<td>1.4</td>
</tr>
<tr>
<td>ECLAC21 (MNNG)</td>
<td>168</td>
<td>37</td>
<td>Burns et al., 1987</td>
<td>11.7</td>
<td>1.6</td>
</tr>
<tr>
<td>ECLAC27A (MNNG, uvrB^-)</td>
<td>130</td>
<td>22</td>
<td>Gordon et al., 1990</td>
<td>13.4</td>
<td>1.4</td>
</tr>
<tr>
<td>ECLAC27B (MNNG, polA^-)</td>
<td>112</td>
<td>22</td>
<td>Gordon et al., 1990</td>
<td>11.8</td>
<td>1.3</td>
</tr>
<tr>
<td>ECLAC27C (MNNG, umuC)</td>
<td>125</td>
<td>22</td>
<td>Gordon et al., 1990</td>
<td>8.7</td>
<td>1.1</td>
</tr>
</tbody>
</table>

We verified the REGTR program by using generated mutational spectra. For this purpose, a standard computer multinomial variates generator from the RANLIB compilation (Brown and Lovato, 1993) was used. We selected four possible values for the total number of mutations: N = 50, 100, 150, 200. Probability values for mutational spectra in computer simulations were taken from nine mutational spectra induced by Sn1 alkylating agents (Table 1). For each random spectra number of mutations y_i in a site, i was generated based on probabilities p_i(y_Rg/T) or p_i(y_Yg/T) (depending on the site context), where y_Rg and y_Yg are average numbers of mutations observed in the Rg and Yg sites (Table 1), and T is the total number of mutations for a real spectrum (Table 1). One hundred random mutational spectra with the same order of X_j as in real spectra and 100 random spectra with shuffled X_j were generated for each value of N and each real mutational spectrum. Results of generated spectra analysis are shown in Table 2. The probability of incorrect classification decreases with an increase of sample. The implication of testing the proposed algorithm is that the regression analysis can be recommended as a tool for studying mutational spectra.
Table 2. The error of classification for generated mutational spectra

<table>
<thead>
<tr>
<th>Spectrum name</th>
<th>Error of classification for N = 25</th>
<th>Error of classification for N = 50</th>
<th>Error of classification for N = 75</th>
<th>Error of classification for N = 100</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECLAC12A</td>
<td>0.33</td>
<td>0.04</td>
<td>0.00</td>
<td>0.01</td>
</tr>
<tr>
<td>ECLAC16A</td>
<td>0.62</td>
<td>0.47</td>
<td>0.08</td>
<td>0.04</td>
</tr>
<tr>
<td>ECLAC16B</td>
<td>0.52</td>
<td>0.50</td>
<td>0.06</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC17A</td>
<td>0.58</td>
<td>0.47</td>
<td>0.06</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC20</td>
<td>0.54</td>
<td>0.44</td>
<td>0.48</td>
<td>0.34</td>
</tr>
<tr>
<td>ECLAC21</td>
<td>0.45</td>
<td>0.31</td>
<td>0.06</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC27A</td>
<td>0.46</td>
<td>0.21</td>
<td>0.03</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC27B</td>
<td>0.50</td>
<td>0.17</td>
<td>0.05</td>
<td>0.02</td>
</tr>
<tr>
<td>ECLAC27C</td>
<td>0.65</td>
<td>0.48</td>
<td>0.23</td>
<td>0.10</td>
</tr>
<tr>
<td>Shuffled (\chi_j)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ECLAC12A</td>
<td>0.37</td>
<td>0.03</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC16A</td>
<td>0.58</td>
<td>0.43</td>
<td>0.04</td>
<td>0.02</td>
</tr>
<tr>
<td>ECLAC16B</td>
<td>0.50</td>
<td>0.45</td>
<td>0.04</td>
<td>0.04</td>
</tr>
<tr>
<td>ECLAC17A</td>
<td>0.55</td>
<td>0.48</td>
<td>0.09</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC20</td>
<td>0.50</td>
<td>0.44</td>
<td>0.40</td>
<td>0.28</td>
</tr>
<tr>
<td>ECLAC21</td>
<td>0.42</td>
<td>0.34</td>
<td>0.05</td>
<td>0.02</td>
</tr>
<tr>
<td>ECLAC27A</td>
<td>0.48</td>
<td>0.19</td>
<td>0.06</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC27B</td>
<td>0.52</td>
<td>0.18</td>
<td>0.03</td>
<td>0.00</td>
</tr>
<tr>
<td>ECLAC27C</td>
<td>0.69</td>
<td>0.45</td>
<td>0.27</td>
<td>0.14</td>
</tr>
</tbody>
</table>

The error of classification was calculated as the ratio of misclassified sites to the total number of sites studied.

Analysis of mutational spectra induced by N-ethyl-N-nitrosourea

It is assumed that N-ethyl-N-nitrosourea (ENU) and Sn1 alkylating agents have different specificity. Analysis of mutational spectra induced by ENU (Burns et al., 1988) (the entry name in the Database of Mutational Spectra is ECLAC13A) revealed two classes of mutable sites. The most mutable class is noted for the presence of base G at position –1 (average number of mutations is nine). Ag sites have a mutation frequency (average number of mutations is four) similar to those at the Tg and Cg sites (average number of mutations is three and one, respectively) and are aggregated into the second class. Thus, regression analysis provides further support for a difference in context specificity between ENU and Sn1 mutagens. Another ENU-induced spectrum was obtained for the gpt gene of E.coli (Richardson et al., 1987). Analysis of this spectrum reveals that 14 G:C→A:T mutations (total number of G:C→A:T mutations is 22) are observed in five Gg sites (total number of sites is 24). By Fisher’s (1935) hypergeometrical test, the frequencies of mutations in Gg sites and Hg sites differ significantly (\(P=0.996\)). These results support the view that the Gg sites are highly mutable in the gpt gene too.

The Rg consensus was revealed for hotspots of mutations induced by ENU in the uvrB+ strain of E.coli (the entry name in the Database of Mutational Spectra is ECLAC13B). Interestingly, the same consensus is typical of Sn1 alkylating agents. Ag sites have elevated mutation frequencies (average number of mutations is nine) as compared to Gg (average number of mutations is six). A similar observation is on MNNG-induced spectrum (ECLAC20): the average number of mutations is 12 at Ag sites and nine at Gg sites. To verify this conclusion, we performed a pairwise comparison of the spectra being studied in the lacI gene using the COMP12 and CORR12 programs (ftp.bionet.nsc.ru/pub/biology/dbmsms) (Babenko and Rogozin, 1999). On the whole, there is a significant correlation between the spectrum of mutations induced by ENU in the uvrB+ strain of E.coli and those induced by Sn1 alkylating agents. At present, it is difficult to say by what mechanism the mutations induced by ENU in the uvrB+ strain of E.coli occurred, but it can be suggested that the similarity in context specificity between this mutational spectrum and Sn1-induced spectra is non-random, and is due to common features in the molecular mechanisms of mutagenesis.

Discussion

Analysis of mutational spectra is quite an important problem, because the number of mutational spectra accumulated to date is high, and there are only a few for which information on the molecular mechanisms of mutagenesis is available and context features of hotspots are known. The method described allows these spectra to be analysed in detail. The algorithm of regression analysis has been tested on samples of real and generated biological data. Mutational spectra with known contextual features of hotspots have been analysed. A good agreement has been demonstrated between the predicted classes of available sites and real contextual features. This implies that the proposed technique is efficient in analysis of mutational spectra.

The results used in this work are optimal with respect to the classification quality criterion (Lbov and Berikov, 1993). It should be noted that a lot of mutational spectra were obtained for few available sites, and this case requires several variants of classification. The classification that is optimal with respect to quality may not be so from the researcher’s point of view. A set of suboptimal classifications is suggested in this case. The decision on the adequacy of a classification is made by the researcher based on other experimental data, on comparing with similar mutational spectra, and other approaches, which are difficult to formalize in terms of the proposed method. We are planning to continue treating the problem. Comparison of classification results for similar spectra can help to solve this problem. For example, such an approach
was applied for analysis of ENU-induced Gg mutable motif in this study. Another example is testing of the reliability of mutable RgYW and TaA motifs for different immunoglobulin V genes (Rogozin and Kolchanov, 1992; Rogozin et al., 1996).

It is possible that the order of the $X_i$ might have an effect on results of classification, since the subsets of the data at nodes further down the tree may be too small to assess the necessity of further subtree construction. We did not reveal any effect of the order of the $X_i$ during computer simulations (Table 2), although it cannot be excluded for some mutational spectra characterized by a complex hotspot context.

The effect of a nucleotide context on mutagenesis may result from (i) interaction of DNA and mutagen during the formation of pre-mutational damage, (ii) repair of pre-mutational damage, (iii) low DNA replication accuracy or (iv) the mutation-changed structure of the protein that is used for revealing mutations in experimental test systems. Involvement of stages (i)–(iii) is confirmed by significant differences in the distributions of hotspots in the sequence being studied and context specificity between different alkylating agents. A strong influence of pre-mutational damage repair on context specificity and the distribution of hotspots for the mutagen ENU is confirmed by the differences in mutational specificity and the distribution of hotspots for the mutagen ENU during the formation of pre-mutational damage, (ii) repair of pre-mutational damage, (iii) low DNA replication accuracy or (iv) the mutation-changed structure of the protein that is used for revealing mutations in experimental test systems. Involvement of stages (i)–(iii) is confirmed by significant differences in the distributions of hotspots in the sequence being studied and context specificity between different alkylating agents. A strong influence of pre-mutational damage repair on context specificity and the distribution of hotspots for the mutagen ENU is confirmed by the differences in mutational spectra between isogenic $uvrB^+$ and $uvrB^-$ strains of E.coli.

It is necessary to emphasize that quite a limited sample of contextual features was analysed in this work (bases at positions −5 to +5). It is also possible that more distant bases also affect mutations induced by alkylating agents, although our study indicates that the mutagenesis induced by such agents is most affected by the nearest-neighbour bases at positions −1 or +1.

Acknowledgements

This work was supported by the Russian Fund of Fundamental Research and the Russian State Program ‘Frontiers in Genetics’. IBR was partially supported by grants from the National Institutes of Health (GM-20293) and the National Science Foundation (DEB-9520832). We are thankful to V. Filonenko for translation of this manuscript into English and B. Tsygan for help in improving the translation. The helpful suggestions of two anonymous reviewers are gratefully acknowledged.

References


