Interactive computer programs in sequence data analysis

Pudur Jagadeeswaran and Paul M. McGuire, Jr. *

Department of Human Genetics, Yale University School of Medicine, 333 Cedar Street, New Haven, CT 06510, USA

Received 6 October 1981

ABSTRACT

We present interactive computer programs for the analysis of nucleic acid sequences. In order to handle these programs, minimum computer experience is sufficient. The nucleotide sequence of the human gamma globin gene complex is used as an example to illustrate the data analysis.

INTRODUCTION

Recombinant DNA technology and rapid nucleic acid sequencing methods have resulted in an accumulation of large amounts of sequence data. We have been involved in determining the nucleotide sequences of the intergenic DNA of the human beta-globin gene cluster (1). During the course of this investigation it became necessary to develop computer programs to analyze the sequence data.

The sequence data analysis provides useful information in various ways. For example, the homology between the two sequences may give an idea of the degree of evolutionary divergence between those sequences. The direct repeats can show that a segment of a DNA or a gene is duplicated. The inverted repeats may be useful in generation of RNA secondary structure. The restriction endonuclease maps of the sequence may be useful in designing recombinant DNA molecules.

The distribution of bases in a given sequence over a fixed block length of this sequence (defined as block percentages) may illustrate A-T and G-C rich regions. The calculation of an autocorrelation function is a way to detect elements of nucleotide sequences (for example, dinucleotides) which tend to be periodically repeated. These periodicities may suggest important structural features such as the pitch of chromatin DNA (11). The random nucleotide sequences could be used as a control over the data sequence to detect significant

periodicity. The periodicities can be easily visualized by spectral analysis of the autocorrelation functions (14).

Several computer programs have been reported in the past for analyzing the nucleic acid sequences (2-10). In this paper we describe some of the programs used in our laboratory. They are divided into two groups. The first group of programs named MATCH searches for (a) homology between two DNA sequences; (b) direct and inverted repeats within a sequence. The underlying principle of this group is the matrix generation program described earlier (3). We have modified the base-pairing matrix into a dot matrix such that sequence homology and the repeats can be generated using the same principle.

The second group of programs named SEQSTAT provides (a) autocorrelation functions for n-mers (11); (b) random nucleotide sequence having the same n-mer composition as the data sequence; (c) periodicities of the autocorrelation functions; (d) block percentages of the sequences and (e) restriction endonuclease maps of the sequence.

These programs are designed to run on an IBM 370/158 computer with time-sharing options (TSO). They are interactive and written in Fortran IV, and TSO command language. In all examples, any typing done by the operator appears in lower case letters and is completed by a carriage-return character. The computer's response is in uppercase letters. The DNA sequences to be analyzed are stored in an independent user-created data file. The sample outputs are for dataset called GAMMA.DATA (sequence of the gamma-globin gene complex). The user has access to the central computer facilities via a terminal. The communication between the investigator and the computer for high-speed printing, plotting and output formats is controlled from the keyboard. A general description of these programs is given below with examples.

DESCRIPTION OF THE PROGRAMS

MATCH

This group of programs is designed to generate the dot matrix which provides information on the homology between two sequences and direct and inverted repeats within a sequence. The dot matrix described here is a modification of the base-pairing matrix (3). The dot matrix is generated by using one sequence as the X-axis (X-sequence) and another as the Y-axis (Y-sequence) as shown in Fig. 1. When a nucleotide in the X-sequence matches a nucleotide in the Y-sequence a dot is placed in the matrix at the coordinates corresponding to the locations of the

Fig. 1.

nucleotides in the sequences. The output of the program is a parallelogram matrix obtained by transforming the rectangular matrix as shown in Fig. 1. This results in horizontal continuous matches which are more comprehensible. The program enables the user to filter the noise which arises from short matches that obscure longer ones. Dots are put in the matrix when $N/L \times 100$ is greater than or equal to P where N is the the number of matches between any segment of length L from both sequences and P is the user-defined minimum match percentage. The "match" used in this program applies when the nucleotides are (a) identical or (b) complementary without G-T pairing or (c) complementary with G-T pairing. A match of type-(a) between two different sequences results in homology matrix. The horizontal contiguous dots show the homology between two different sequences. If the X-sequence and Y sequence are identical, the homology matrix will demonstrate direct repeats. Direct repeats are detected by two horizontal continuous lines if they are both parallel and vertical to each other (Figure 1b). A match of type-(b) produces a complementary strand homology matrix. This gives homology between opposite strands of two different sequences. When the two sequences are the same, the matrix shows inverted repeats. In Figure 1c each horizontal stretch of dots represents an inverted repeat. A match of type-(c) of the sequence with itself produces an inverted repeat matrix with GT pairing.

The program also allows the investigator to choose whether the dot matrix will be written on the terminal, the magnetic disc or the pen plotter. The output written onto the disc is automatically reformatted, so that dot matrices larger than the line size of the terminal, or the line printer, can be printed in segments of 130 characters. The program can handle up to twenty data sets and compare any two of them. Other options control the size of the plot and the beginning and ending points of the sequences analyzed. The program can compare two sequences of up to 10,000 nucleotides each.

An example of the use of the program is illustrated in Fig. 2. The user types the command MATCH to start his program. The computer asks whether the user intends to write to the disc or to the plotter, issues a message and prompts for sequence data set names. When the operator finishes entering the dataset names he is asked to enter a name for the output datasets. The program then displays a table of five groups of options. Entering an integer from 1 to 5 results in a prompt for the

```

match

ENTER POSITIONAL PARAMETER DISK -
yes

ENTER POSITIONAL PARAMETER PLTT -
yes

FORMER REGION SIZE IS 350K
FORMER MARGIN SIZE IS 64K
NEW REGION SIZE IS 350K
NEW MARGIN SIZE IS 64K
SEQUENCE NAMES WILL BE ASSIGNED TO SEQUENTIAL UNIT NUMBERS
STARTING WITH FT01...ENTER NAME, HIT RETURN, ENTER A NULL
LINE TO FINISH
-->samee

-->

ENTER NAME FOR OUTPUT DATASETS...sameeout

MATCH: DIRECT & INVERTED HOMOLOGY, EXECUTED 14:35 THUR 1 OCT 81

(1)FTX      1 * FTY      2
(2)XST      1 * XEND    120 * YST      1 * YEND    120
(3)HLEN     10 * BLEN    10
    HPCT 70.0 * BPCT 70.0 * XPLT 25.0 * YPLT 25.0
(4) HOM      F * INV      F * GT      F
(5) PLTT      F * DISK      F * TERM      F

ENTER INTEGER 1-5 FOR PROMPT FOR VARIABLE VALUES
ENTER # TO SHOW VALUES... GO TO EXECUTE PROGRAM...END TO END
1
    FTX, FTY
?
1,1
5
    PLTT,DISK,TERM
?
t,t,f
4
    HOM, INV, GT
?
t,f,f
3
    HLEN,BLEN,HPCT, BPCT, XPLT, YPLT
?
10,10,60.0,70.0,25.0,25.0
so

PLOT1438 JOB CREATED
WEISHEA,GAMMAOUT,PLTTDATA                                OPENED FOR PLOTDATA OUTPUT

(1)FTX      1 * FTY      1
(2)XST      1 * XEND    120 * YST      1 * YEND    120
(3)HLEN     10 * BLEN    10
    HPCT 60.0 * BPCT 70.0 * XPLT 25.0 * YPLT 25.0
(4) HOM      T * INV      F * GT      F
(5) PLTT      T * DISK      T * TERM      F

ENTER INTEGER 1-5 FOR PROMPT FOR VARIABLE VALUES
ENTER # TO SHOW VALUES... GO TO EXECUTE PROGRAM...END TO END

end

00013437 POINTS GENERATED.
DO YOU WISH TO SUBMIT THE PLOT JOB? ENTER YES OR NO:
no

WEISHEA CHANGE WEISHEA,GAMMAOUT,DISKDAT
ENTRY (A) WEISHEA,GAMMAOUT,DISKDATA DELETED
SUBMIT REFORMATTED DOT MATRIX TO LINEPRINTER?(ANS. Y OR N)...n
LIST REFORMATTED DOT MATRIX ON TERMINAL?(ANS. Y OR N)...n

READY

```

Fig. 2.

new values for the respective group. The new values should be entered, separated by a space or comma, and must be of the same type as the initial values. For example, integers with decimal points should be used with HPCT, BPCT, etc. (see Fig. 2). When an integer, n_1 or n_2 , is assigned to FTX or FTY, the program uses the respective dataset from the created data list (Fig. 2) for the X-axis or Y-axis.

XST or YST is the starting point of X or Y sequence, whereas XEND or YEND is the ending point of the X or Y sequence. The values for HLEN and BLEN will set the length "L" for type-(a) matrix and type-(b) or (c) matrix, respectively. Similarly, HPCT and BPCT will give the percentages "P". The size of the plot axes is determined by XPLT and YPLT. The true or false values (T or F) can be assigned to HOM, INV, GT, PLTT, DISK and TERM. HOM controls the type-(a) matrix and INV and GT control type-(b) and (c) matrices. PLTT creates datasets for the plot job. DISK writes the reformatted matrix onto the magnetic disc and TERM facilitates the output printing on the terminal. Typing \$ sign displays the table of options with the current values, so that the user can verify the entries. To execute the program operator types in GO and to terminate the program, he types in END. Then questions regarding the submission of the plot job and the printing of the reformatted matrix are supplied by the computer to receive the answers Yes (Y) or no (N). There is a provision to automatically name the reformatted matrix, for example, GAMMAOUT output dataset (Fig. 2) is named GAMMAOUT.LPTRLIST. An example of the plot output comparing Alu-family DNAs (1, 13) from the globin complex is shown in Fig. 3.

SEQSTAT:

This group of programs produces (a) autocorrelations, (b) random nucleotide sequences, (c) autocorrelation periodicities, (d) block percentages and (e) restriction endonuclease maps. It is initiated by the command SEQSTAT and is controlled by subcommands AUTOCORR, RANDOM, SASFREQ, BLKPER and SEQNAM to perform the above functions. After entering SEQSTAT, the user is prompted to supply the dataset name (GAMMA, as shown in Fig. 4), and to choose where the output will be listed and whether it will be retained on the disc. A dataset name must be supplied since the output is always written on the disc.

Therefore, a procedure was adopted that automatically names the output datasets based on the name of the sequence dataset and the subcommand. A number is also included in the output dataset name to

Fig. 3. Type (a) dot matrix showing the homology between Alu family DNA sequence preceding the delta globin gene and the Alu family DNA sequence preceding the gamma globin gene (1). Vertical coordinates represent the delta globin Alu family sequence and the diagonal coordinates correspond to the gamma globin Alu family sequence. The horizontal contiguous stretch of dots represents a significant match between these two Alu elements.

allow multiple use of a subcommand on the same dataset. A list of relevant datasets is displayed, showing the numbers already in use. Then the operator is prompted to enter a new number. Since all the subcommands require that the sequences be binary numbers, the dataset is recoded and saved as a compact dataset named GAMMA.CMPCTDAT. The program issues a prompt SEQSTAT and waits for a subcommand.

The prompt is reissued when each subcommand finishes executing.

```

seqstat
ENTER POSITIONAL PARAMETER NAME -
gamma
WRITE LISTINGS ON TERMINAL? (ANS. Y OR N)...n
WRITE LISTINGS ON LINE PRINTER? (ANS. Y OR N)...n
SAVE THE LISTINGS ON DISK? (ANS. Y OR N)...y
OUTPUT WILL BE LABELED WITH ##, WHERE # IS A DIGIT
NONVSAM ----- WEISHEA.GAMMA.DATA
      IN-CAT --- SYSCTLG.VUSER21
PLEASE ENTER NEXT ##, 00 IF IS FIRST...00
GAMMA.CMPCTDAT DOES NOT EXIST...CREATING IT
SEQSTAT
autocorr
N-MER LENGTH?...2
GAMMA.LINK2BIN DOES NOT EXIST...CREATING IT
N-MER LENGTH?...
?
2
MAX DIST. TO PLOT? (0 IF DON'T WANT PLOTS)...100
FIRST DISTANCE TO PLOT?...1
WILL PLOT EVERY NTH DISTANCE STARTING WITH FIRST DIST. TO PLOT
ENTER N?...1
WANT TO PLOT A SUBSET OF ALL POSSIBLE 2-MERS?(T OR F)...f
      WRITE INPUT DATASET FOR SASFREQ ON DISK? (ANS. Y OR N)
y
SEQSTAT
random
ENTER N OF LINK'N'BIN...2
ENTER Y OR N SEPARATED BY BLANKS BELOW N-MER LENGTH
5 4 3 2 1 0
n n n y n n
RESTART RANDOM NUMBER GENERATOR? (ANS. Y OR N)
y
ENTER 2 INTEGERS OF 9 OR LESS DIGITS SEPARATED BY A SPACE
?
3456295 897643
SEQSTAT
sasfreq
J0R WEISHEA@J0R00310) SUBMITTED
GAMMA.SAS00LST BEING CREATED NOW...
YOU WILL HAVE TO LIST IT WHEN IT IS DONE
SEQSTAT
blkper
BLOCK LENGTH?
?
50
PLOT BASES TOGETHER?(ANS T OR F)
?
f
PLOT BASES SEPARATELY UNSMOOTHED?(ANS T OR F)
?
t
PLOT BASES SEPARATELY SMOOTHED?(ANS T OR F)
?
f

```

Fig. 4.


```

SEQSTAT
seanma

OUTPUT LINE LENGTH?
?
120

ENTER NUMBER OF SPACING LINES
?
0

CONSIDER SEQ TO BE A LOOP?(ANS. Y OR N)
n

DO A SUBSEQUENCE?(ANS. Y OR N)
n

OUTPUT ENZYME NAMES?(ANS. Y OR N)
n

OUTPUT POSITION NUMBERS?(ANS. Y OR N)
y

OUTPUT COMPLIMENT SEQUENCE?(ANS. Y OR N)
y

OUTPUT FRAGMENT LENGTHS?(ANS. Y OR N)
n

SEQSTAT
change

FOR DATA SET CALLED NAME.DATA ENTER NAME
samaa

OUTPUT WILL BE LABELED WITH ##, WHERE # IS A DIGIT
NONVSAM ----- WEISHEA.GAMMA.AUTOOLST
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.BLK00PER
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.CMPCTDAT
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.DATA
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.LINK2BIN
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.PLT00BIN
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.SAS00LST
IN-CAT --- SYSCTLG.VUSER21
NONVSAM ----- WEISHEA.GAMMA.SEQ00LST
IN-CAT --- SYSCTLG.VUSER21
PLEASE ENTER NEXT ##, 00 IF IS FIRST...01

NONVSAM ----- WEISHEA.GAMMA.CMPCTDAT
IN-CAT --- SYSCTLG.VUSER21
SEQSTAT
end

READY

```

Fig. 4. Contd.

Each subcommand and its application to nucleotide sequence data analysis is briefly described below.

AUTOCORR:

Autocorrelation function for an n-mer in a sequence is given by plotting the number of occurrences of a distance between the locations of the n-mer versus the distance (11). The functions for some or all of the 4ⁿ n-mers can be combined into one plot. This subcommand performs its calculations in two steps. In the first step it prompts the user to select the length of the n-mer and automatically creates an intermediate

dataset named GAMMA.LINKnBIN (n=length of the n-mer). In the second step the operator is again prompted for the length of the n-mer and to select various options to control the nature of the plot. Another option controls the creation of an input dataset for the subcommand SASFREQ. This SASFREQ dataset is named as GAMMA.PLT##BIN where # is a digit. The output dataset is stored under the name GAMMA.AUT##LST. This subcommand also produces a catalog of all possible n-mers. An example of an autocorrelation plot is shown in Fig. 5. From these autocorrelation plots periodicities in a given sequence can be detected.

Fig. 5. Autocorrelation functions for the sum of all dinucleotides of the sequence preceding the gamma globin gene. The X axis corresponds to the distance between the dinucleotides and the Y axis represents the frequency of occurrence.

Fig. 6. Spectral density plot of the sum of all dinucleotide periodicities of the sequence (5.5 kb) preceding the gamma globin gene. The peak of spectral density counts is at a period 80 showing 80 base pair periodicity.

RANDOM:

This subcommand generates random sequences based on the percentages of the n-mers in the data sequence. The information on the percentages is contained in the LINKn.BIN dataset. However, LINKnBIN does not have information on oligomer lengths greater than n. The user is prompted to enter "n" of an existing LINKnBIN dataset and again asked to choose the length of the n-mers (see Fig. 4). The program produces random sequences by using pseudo-random number generator which will always produce the same series of random numbers unless it is restarted.

An option is provided to restart the random number generator. The

Fig. 7. Block percentages of A in the gamma globin gene. The X axis shows the block length and the Y axis represents the percentages.

output datasets are automatically named and stored as compact datasets. For example, a random sequence with the same dimeric composition as GAMMA.DATA will be named GAMMA.RAND2.CMPCTDAT. The random sequences produced by the program can be used as a control over the data sequence to detect significant periodicities.

SASFT3Q:

This subcommand submits the SAS job that uses PLT##BIN as an input dataset for spectral analysis (14). Since the program runs independently of the terminal this subcommand cannot list the output. Instead, the dataset is named and written onto the disc. The output dataset is called, GAMMA.SAS##LST which the user will have to list independently. The output has information on spectral densities and periodograms. An example of the spectral analysis is shown in Fig. 6. In the spectral density plots each peak of spectral density of counts (Fig. 6) represents a periodicity. This analysis is an easier way of visualizing periodicities in a given sequence. The results of the SASFREQ analysis will be presented in future (Jagadeeswaran, P. and Weissman, S. M., manuscript in preparation).

BLKPER:

This subcommand plots the percentages of individual bases as well as any two bases put together in a block of a sequence of a given length

Fig. 8.

on the abscissa and the order of blocks on the ordinate. In this program the user can choose to produce the smoother or unsmoothed plots. There is also an option in this program to plot the percentages of the

bases adjacent to one another in a block: The output dataset is called GAMMA.BLK##PER. An example of this is shown in Fig. 7, using the sequence of gamma globin gene (12, P. Jagadeeswaran, unpublished). The block percentages are useful for example in detecting pyrimidine clusters as found in globin genes (1).

SEQNAM:

This subcommand generates the restriction endonuclease names at their locations on a sequence which can be numbered. The program lists the names of commercially available restriction endonucleases which have the same initial recognition point and their locations. In such cases, one of the enzymes' names is listed with a star in the main body of the sequence. The sequence can be listed with a specified line length (in multiples of 10) and the number of spaces between the lines in either single or double stranded form. The program has a provision to treat the nucleotide sequence in a circular (loop) or a linear fashion. The user has an option to list the numbers of a sequence at which the restriction enzyme sites occur followed by the list of lengths of the fragments produced. These numbers are given in the same order of the cleavage sites from the starting point in a sequence as well as in the ascending order of the length of the fragments. The program is capable of doing any of the above options for a segment of the sequence with a given beginning and end point. Fig. 8 illustrates the application of SEQNAM to Pvu II - Hpa II sequence preceding the gamma globin gene.

There are two auxiliary subcommands named CHANGE and END. CHANGE helps the operator to perform any of the above subcommands on a different dataset. END terminates the program.

ACKNOWLEDGMENTS

We are grateful to Dr. Sherman M. Weissman, in whose laboratory this work was carried out, for his suggestions. The research was supported by Grant Number P01 GM 20124, awarded by the National Institutes of Health.

*Present address: 514 Elm Street, New Haven, CT 06511, USA

REFERENCES

1. Jagadeeswaran, P., Pan, J., Spritz, R., Duncan, C. D., Biro, P. A., Tuan, D., Forget, B. G. and Weissman, S. M. Proc. ICN-UCLA Symp. Dev. Biol. (in press).

2. Jagadeeswaran, P., Cherayil, J. D., Pattabiraman, N. and Sasisekharan, V. (1977) *Current Science* 46, 691-694.
- 3a. Jagadeeswaran, P. (1977) Studies on the primary and secondary structures of 5S RNA from Mycobacterium smegmatis. Ph. D. thesis, Indian Institute of Science, Bangalore, India.
- 3b. Tinoco, I.,jun., Uhlenbeck, O. C. and Levine, M. D. (1971) *Nature* 230, 362-367.
4. Staden, R. (1977) *Nucl. Acids Res.* 4, 4037-4051.
5. Staden, R. (1978) *Nucl. Acids Res.* 5, 1013-1015.
6. Staden, R. (1980) *Nucl. Acids Res.* 8, 817-825.
7. Queen, C. L. and Korn, L. J. (1980) *Methods in Enzymol.* 65, 595-609.
8. Korn, L. J., Queen, C. L. and Wegman, M. N. (1977) *Proc. Natl. Acad. Sci.* 74, 4401-4405.
9. Sege, R., Soll, D., Ruddie, F. H. and Queen, C. (1981) *Nucl. Acids Res.* 9, 437-444.
10. Gingeras, T. K. and Roberts, R. J. (1980) *Science* 209, 1322-1325.
11. Trifonov, E. N. and Sussman, J. L. (1980) *Proc. Natl. Acad. Sci.* 77, 3816-3820.
12. Slightom, J. L., Blechl, A. E. and Smithies, O. (1980) *Cell* 21, 627-638.
13. Jagadeeswaran, P., Forget, B. G., and Weissman, S. M. *Cell* (in press).
14. SAS USER'S GUIDE, 1979 Edition, Chapter on Spectra Procedure, p. 381-388.