NEWPATXL, a general paternity program. XL macro version

by Bill Amos, Dept. of Zoology, Cambridge, CB2 3EJ, UK

email: w.amos@zoo.cam.ac.uk, tel/fax +44 (0)1223 336616

Introduction

NEWPAT is a generalised paternity program which calculates allele frequencies, checks for the presence of non-amplifying alleles, assays each input file for duplicate entries, searches for parent-offspring relationships according to user-inputted criteria and then uses a randomisation approach to assess the significance of any matches found. This version is run by opening the Excel file ‘NewPat.xls’, pasting appropriate input data into the Input sheet(s), then selecting Tools>Macro>Macros . >’Main’ and then hit ‘Run’. Outputs go to two separate output sheets, one for the allele frequencies and one for the paternity / identity matching.

Input file

The program accepts an input file in the following format:

a	a	a	a	a	a
Albert	M	234	236	122	128
Mary26	F	232	234	124	124
PoMary26	O	232	236	124	122

First line: **Must** contain non-empty cells extending to the end of the data block (the program counts along row 1 and calculated the number of loci assuming two initial columns of descriptive data followed by genotypes stored as two columns per locus.

Subsequent lines: Each line of data must contain a name in column 1, sex/status in column 2 and genotype data in the remaining columns.

sample name: can be any, though short is preferable!

status: must be either ‘M’ (male), ‘F’ (female) or ‘O’ (offspring). Status of offspring can be omitted if the name of the individual is prefixed with ‘Po’ (for historical reasons!)

genotype: one allele per column, missing data left as either empty cells or, preferably, scored as zeros. All alleles must be numeric data.

Data for paternity analysis should be placed in sheet ‘Input1’ and be arranged mother 1 followed by all her offspring, mother 2 folled by her offspring . . and ending with all the males together. Allele frequencies are calculated either from these data or from separate data pasted into sheet ‘Input2’ using the same format (though with no sample order restrictions). Separate allele frequency data are advisable when many offspring are likely to share the same parent(s).

Program options

NewPatXL has somewhat fewer options than NewPat, but has a greater range of output information. The main omission is that NewPatXL does not allow sex-linked loci (though it can be used to detect them). When run, the user is presented with a paramter entry form. Current options are:

1) *Mother-offspring arrangement*

Two arrangements of females and their offspring are catered for. If maternal information is available, the data must be arranged Mother 1, Offspring of mother 1, [other offspring of mother 1], Mother 2, offspring of mother 2, [other offspring of mother 2], etc., and ending with all males together at the end. Alternatively, if no maternal information is available, the file should consist of all offspring together followed by all candidate males. Note, there is no cross-checking for name compatibility.

2) *Sex-specific allele frequencies*

In many scenarios, males will include immigrants and may be genetically slightly different from the females. To allow for this, I prefer to use sex-specific allele frequencies based on mothers (and offspring) for the females and males for the candidate fathers. However, the program also allows the user to deploy a single set of allele frequencies based on all data in the appropriate input sheet.

3) *Father for offspring or vice versa*

It is often useful to generate both sets of comparisons, looking for offspring which can be assigned to each male and then for potential fathers for each offspring. These alternative analyses can be selected by the toggle given.

4) *Randomization number*

When a paternity match is found, the program then simulates large numbers of offspring / males to determine how frequently such a match will arise by chance. The match criteria in the randomisations are based on the nature of the match in question. Thus, if a match is made which involves one unscored locus and one mismatch, the randomisation tests will allow one unscored locus and one mismatch . Randomisation matches must have an equal or greater relatedness score compared with the match being tested (see below). The default randomisation number is set at 100 times the size of the data set being used. Consequently, the resulting number of matches represents the percent probability that a similarly sized data set will yield a match by chance. On slow computers, this figure could be set lower, e.g. to 10, to increase speed of running.

5) *Mismatch criteria*

A prime feature of NewPat is that it allows flexibility of matching criteria, both in terms of the different matching processes involved (mother offspring, genotype identity and father-offspring) and in terms of the form of mismatches which can be allowed. Up to 4 criteria can be set for each category.

a) Number of unscored loci

The user can allow matches to include any number of unscored loci

b) Number of mismatches

This sets the number of loci which are allowed to be genetically incompatible with a match, excluding null allele mismatches (see below).

c) Maximum repeat unit difference

Since many allele mis-scores will involve only a single repeat unit difference (genotype 4/5 scored as 4/4 or 4/6 due to stutter bands), some level of scoring inaccuracy can be allowed without too much loss of resolution by setting the total number of repeat units which would need to change to achieve a match. For example, with a parameter value set to 1, allele 15 would be allowed to match alleles 14 and 16, but not alleles 13 or 17. Note, these ‘sloppy’ mismatches are still treated as mismatches, such that if the allowable mismatches are set to zero the repeat unit parameter is ignored.

d) Probability of nullmatch

Some loci carry null alleles. In addition, allele non-amplification and other problems can lead to similar effects. Consequently, NewPat estimates null allele frequencies at each locus in turn and then will allows homozygote mismatches between parent and offspring at loci where null alleles may be present. The user is prompted for the minimum total probability of a null mismatch. When a potential null-match arises, the program calculates the probability that the homozygote mismatch is due to a null allele. The default value of 0.05 is set to allow null mismatches at one or maybe two loci where there are problems. Higher values (maximum=1) are more stringent, and will only allow matches at loci where the problem is severe. Some indication of the problem is given locus by locus on sheet Output1. Note, since the nature of the problem is similar, the null allele parameter can be used to

Data rescaling and allele frequency calculation

For programming convenience, all allele numbers are transformed onto a scale such that 1 is the shortest alleles, 2 the next shortest etc. To do this, the program assumes that allele lengths below 50bp in length do not occur, and flags alleles which contravene this assumption.

Allele frequencies are calculated using the NULLTEST algorithm I developed for estimating the frequency of putative null alleles. The program uses an iterative approach to find the best fit frequency of null (non-amplifying) alleles which best fit the data, assigning any homozygote excess to the null allele category. It does not matter whether non-amplification is due to sporadic PCR failure or to a genuine null allele. The result is a list of optimal allele frequencies (negative null allele frequencies rectified to zero), which is output to both screen and the results file, and which is used in subsequent calculations. Also given are the raw null allele frequencies estimates. These are included because some potential problems (e.g. mis-scoring homozygotes as adjacent allele heterozygotes) can be identified through finding a heterozygote excess (= negative null allele frequency). There is an option for estimating the null allele frequency confidence limits which operates by resampling.

Mother-offspring analysis

The program constructs a paternal allele file with one entry for each offspring. Every offspring's genotype is compared with its putative mother and the paternal bands deduced. It is important to note that the algorithm distinguishes between possible null alleles and definite paternal alleles. To illustrate, mother = 1,3 offspring = 3,3 the paternal band could be either a '3' or a null, and the paternal bands are stored as 3,0. However, if mother = 1,3 and offspring = 3,4 the paternal band must be a '4', and hence the paternal bands are registered as 4,4. Where either the mother is untyped or the offspring does not match its mother, the offspring's genotype is entered in full, equivalent to assuming no maternal information. Where the pup's genotype is missing at a locus, the paternal bands are entered as 0,0 (which matches anything).

All instances where a mother and offspring do not match are outputted to both screen and the output results file. The program attempts to distinguish between mistypings, where typically the mismatch will involve only one locus and mother and offspring genotypes will show high relatedness, and mis-samplings, where mother and pup are unrelated. To do this, both the number of mismatches are recorded and the loci at which mismatches occur. In addition, a coefficient of relatedness (following Queller and Goodnight, values for first degree relatives are distributed about 0.5, values for non-relatives are distributed about 0) is calculated between mother and offspring. Pups with high relatedness (>0.2) and fewer than 2 mismatching loci are included in the paternity analysis.

Paternity analysis

In each comparison, the pup's paternal alleles are compared with a male's genotype, locus by locus, to assess:

1) whether both individuals have been scored at this locus
2) whether an allele is shared (a match),
3) whether a match could exist by invoking a null allele (a nullmatch),
4) whether the genotypes are incompatible with being from a parent-offspring pair

If option 3 is found, the program calculates the probability that a null allele would be involved, using the estimated null allele frequency calculated above. If option 4 is found, the program calculates the size of the difference between the pup's paternal and the male's alleles which are closest in size. Clearly, in most cases, a mismatch of just one repeat could potentially be explained by misreading a gel, whereas larger differences are progressively less likely to arise.

The user is prompted for match criteria in terms of three parameters, the number of unscored loci allowed, the number of mismatches allowed and the combined probability of nullmatches occurring. In essence, the number of unscored loci should be set as low as possible whilst still retaining enough power to eliminate the vast majority of false paternities. The number of mismatches allowed is a luxury which becomes effective when the probability of a chance match is very low. Finally, the setting of the nullmatch probability should depend on the allele frequency output. If null alleles in general appear rare or absent, a low value should be selected (though bear in mind that most large data sets include some ‘dodgy’ samples where there is a danger of amplifying or scoring only one of the two alleles, and these can be allowed to match via the null route).

Each match which fulfils the match criteria is then assessed by a randomisation subroutine. First the relatedness value is calculated between the male and the pup (NB, **not** between the male and pup’s paternal bands). Then, a large number of randomisations are performed, drawing alleles randomly to create pseudo-genotypes which may then be tested for fit to either the male or the pup. When pups are being assigned to males, pseudo-pups are generated, whereas when fathers are being found for pups, pseudo-males are generated. Pseudo-offspring are created by drawing both a mother and a pup and then generating a paternal allele profile just as the program does for a real mother-offspring pair. Note, it is important to draw the mother as well as the pup, since the match probabilities for a pup and for its paternal alleles are radically different.

Finally, when the full analysis is complete, the program then examines the level of background paternities expected by chance. To do this, it creates a complete dummy data file for both mother-offspring and males, and conducts a full paternity test. This process is replicated 10 times to obtain the mean numbers of males achieving 0, 1, 2, . . n paternities. For maximum reality, all properties of the original data set such as the location of untyped loci and presence of sex linkage are retained in the dummy data sets.

Bugs and improvements. The following improvements have been added:

- optional joint or sex-specific allele frequencies
- maximum number of loci now set at 30
- maximum number of alleles increased to 100 per locus (happy now Karen!?)
- more efficient design now allows larger file sizes (maximum of 32,000 alleles in a file)
- minimum relatedness criteria, a filter for low resolution datasets which can be swamped by false positives
- an improved final simulation based on ten replicate data sets.
- toggle to allow alternative suffices

- a minor bug in the identity checking routine has been fixed

Version 5 (release Aug 2000) eliminates a bug introduced in the previous version which causes background paternity assignment rates to be too low by a factor of approximately 10 and outputs raw null allele frequency estimates.

Improvements: forthcoming improvements include:

- translation into Visual Basic for running directly in Excel as a macro
- speed enhancement by automatically re-ordering loci such that the most informative are dealt with first.
- automatic checking for characteristic patterns of mis-scoring such as alleles outside the expected range and confusion between homozygotes and adjacent allele heterozygotes

Please let me know of any problems or suggested improvements!!