J Hum Genet (2002) 47:377–379

SHORT COMMUNICATION

Masayuki Saito · Akira Saito · Naoyuki Kamatani

Web-based detection of genotype errors in pedigree data

Received: February 20, 2002 / Accepted: April 1, 2002

Abstract For linkage analysis using pedigree data, it is important to eliminate contradictions concerning Mendelian inheritance from genotypic data. Such contradictions may derive from either genotyping errors or pedigree errors. We implemented an error-checking algorithm in a World Wide Web-based program that can be used through the Internet even by computer nonspecialists. This program, named Checkfam, uses two error-checking algorithms to detect and report contradictions concerning Mendelian inheritance. With this program, users can guess the causes of genotype errors (technical problems in genotyping, pedigree misrecording, or errors in input data concerning familial relationships). The present program will be useful to researchers for checking genotypic data and for preparing correct input files for linkage analysis.

Key words Error · Genotype · Linkage analysis · Program · WWW

Introduction

For linkage analysis, contradictions concerning Mendelian inheritance should be eliminated from input pedigree files because linkage analysis programs do not accept input files with such errors. Contradictions concerning Mendelian inheritance can derive from two different causes: pedigree errors and genotyping errors. A pedigree error generally occurs when the relationship of pedigree members is misrecorded. Genotyping errors include all other typing errors such as misinterpretation of genotypes and data entry errors (Ehm et al. 1996). Identification of genotype errors can be achieved by checking whether data are in accord with Mendelian inheritance, but this is tedious work because the amount of data to be checked is usually quite large. Several computer programs that find Mendelian inconsistencies either in genotype data (Stringham and Boehnke 1996; O'Connell and Weeks 1998) or pedigree relationships (Boehnke and Cox 1997) have been developed to simplify this work. However, none of these programs can be used as World Wide Web (WWW)-based software. We have developed software named Checkfam that can be used through the Internet. This software checks Mendelian inheritance in genotype data entered in HTML form, and returns the inconsistencies, if any, in the data to end users.

M. Saito (✉) · A. Saito · N. Kamatani
Division of Statistical Genetics, Institute of Rheumatology, Tokyo Women's Medical University, 10-22 Kawada-cho, Shinjuku-ku, Tokyo 162-0054, Japan
Tel. +81-3-5269-1725; Fax +81-3-5269-1726
e-mail: msaito@ior.twmu.ac.jp

Algorithms and implementation

In Checkfam, two algorithms, a "nuclear-family algorithm" and a modified "genotype-elimination algorithm," are implemented that identify errors in genotypic data. The nuclear-family algorithm has been described previously (O'Connell and Weeks 1998). In this algorithm, each family is decomposed to nuclear families, each consisting of two parents and their children. Each nuclear family is then investigated for Mendelian inconsistencies. In the nuclear-family algorithm of Checkfam, the process of finding Mendelian inconsistencies in genotype data is as follows: (1) Different alleles (a_i, $i = 1, 2, \ldots$) are listed from the parents and the children, and a check is done to ensure that the number of alleles is not more than four. (2) If parental genotypes are (a_i, a_j) and (a_k, a_l), then the genotypes of their children must be (a_i, a_k), (a_k, a_i), (a_i, a_l), (a_l, a_i), (a_j, a_k), (a_k, a_j), (a_j, a_l), or (a_l, a_j). An error is reported if the data are not in accord with this rule.

The modified genotype-elimination algorithm is based on the extended version of the Lange-Goradia algorithm (Lange and Goradia 1987; Lange and Weeks 1989). In this algorithm, all possible genotypes of each member of the family are listed, and the data are recursively checked for Mendelian consistency. Genotypes that are incompatible

are eliminated from the genotype list until no more can be eliminated. The original Lange-Goradia algorithm was only for pedigrees without loops (Lange and Goradia 1987), but was then modified to include all types of pedigrees in the PEDCHECK program, although the method used was not described (O'Connell and Weeks 1998). The modified genotype-elimination algorithm in Checkfam is also able to check data from all pedigree structures, including loops. Before performing the modified genotype-elimination algorithm of Checkfam, each nuclear family in a pedigree is given an integral number termed the "ancestor number." The ancestor number of a nuclear family is defined as the number of ancestors described in the pedigree of a child in the nuclear family. Note that the number of ancestors is always the same for all the children of a nuclear family. Checking of the Mendelian inheritance is performed from one nuclear family to another in such a fashion that the nuclear family with a smaller ancestor number is examined later than one with a larger ancestor number. If the examination is performed in this order, all the children in a nuclear family should have already been examined for Mendelian inheritance if necessary when that nuclear family is checked.

In Brief, the algorithm is as follows: (1) The ancestor number is counted for each nuclear family. (2) Mendelian inheritance is checked from one nuclear family to another so that the order of ancestor number is from largest to smallest. (3) When the Mendelian inheritance in a nuclear family is checked, all possible genotypes of the parents are listed and saved for examination of the other nuclear families that include them. (4) If there is at least one set of genotypes of all the family members that is in accord with Mendelian inheritance, the program reports "no errors detected"; otherwise, it reports the errors.

The format of the input data of Checkfam accords with the pedigree file format of the Linkage package (Linkage format, "prefile") (Terwilliger and Ott 1994). This format is essentially the same as that for Mapmaker/sibs and Genehunter. The report of Checkfam is displayed in "prefile" format. The data for each subject are displayed in a line and the data for a nuclear family are displayed in a block. When Mendelian inconsistencies are detected in a nuclear family at a marker locus, the data included in the inconsistencies are displayed in a colored block. If Mendelian inconsistencies are detected by the modified genoytpe-elimination algorithm at a marker locus, then all the data concerning the marker locus are displayed in a colored block.

The program was written in both C language and Perl. The source codes for C language were compiled by gcc. The software was implemented in a home page (http://www.genstat.net/checkfam/) so that it can be freely accessed through the Internet.

Results and discussions

The performance of Checkfam was checked on a Linux OS in an Intel Pentium III personal computer. Two types of test data were used: (a) 200 nuclear families with two children (800 individuals) and 100 markers (80,000 genotypes) and (b) 6 middle-sized families with 13 members (Kruglyak et al. 1996) and 400 markers (31,200 genotypes). Each data set, for either (a) or (b) type, was generated by the pedigree simulator, IVSIM (A Saito et al. 2001), which generates marker genotypes at multiple linked and unlinked loci for given pedigree structures. For the data of type (a), compatible with affected sib-pair analysis, the check was completed by the nuclear family algorithm within 1 min. Note that the genotype-elimination algorithm is not necessary for this type of data. For the data of type (b), compatible with parametric linkage analysis, Checkfam completed the checks both by the nuclear-family algorithm and by the modified genotype-elimination algorithm within 20 s. Most processing time was spent for the transmission of the data. The actual calculation time in the host machine was only a few seconds. This estimation was made with input data without missing genotypes. It is likely to take a little more time when there are genotype-uncertain data. The presence of missing data, however, had almost no influence on the calculation time when the nuclear-family algorithm was used. Giving genotypes to the founders had little influence on the calculation time when the modified genotype-elimination algorithm was used, because a genotype can be given without restriction to the parent who does not have genotyped ancestors. When the genotype of the member who had typed ancestors was unknown, giving a genotype to that subject had a significant effect on calculation time with the genotype-elimination algorithm.

Figure 1A shows an example of the input data for Checkfam. Figure 1B shows the Checkfam report when this example was applied to the program. There are at least four different causes that may generate errors in the input file. If there are technical problems in genotyping in a marker, then users will find multiple errors for that marker. In that case, two continuous columns corresponding to that marker will have colored blocks for many subjects (Fig. 1B,a). If there are misrecordings in the family relationships in a nuclear family, then marker genotype data for that nuclear family will have many colored blocks (Fig. 1B,b). On the other hand, when there are incompatibilities in such data as sex and familial relationships, the second to fifth columns corresponding to the nuclear family will be colored (Fig. 1B,c). If errors are limited to a nuclear family and a marker, there is likely to be a simple error. The error-checking program will be very useful if one can guess the cause of errors from the output data. Using this program, we found that we can often guess the cause of errors from the output data, as in Fig. 1B. Thus, given the output data in Fig. 1B, we can guess that there is a major technical problem in genotyping of the fifth marker. In addition, it is likely that there is a misrecording in the family relationships in the nuclear family involving subjects 5 to 7 in family 2. This figure also shows that there is a mistake in either the sex or the parent columns in the nuclear family involving 4 to 7 subjects in family 1. In addition, there are some other simple errors.

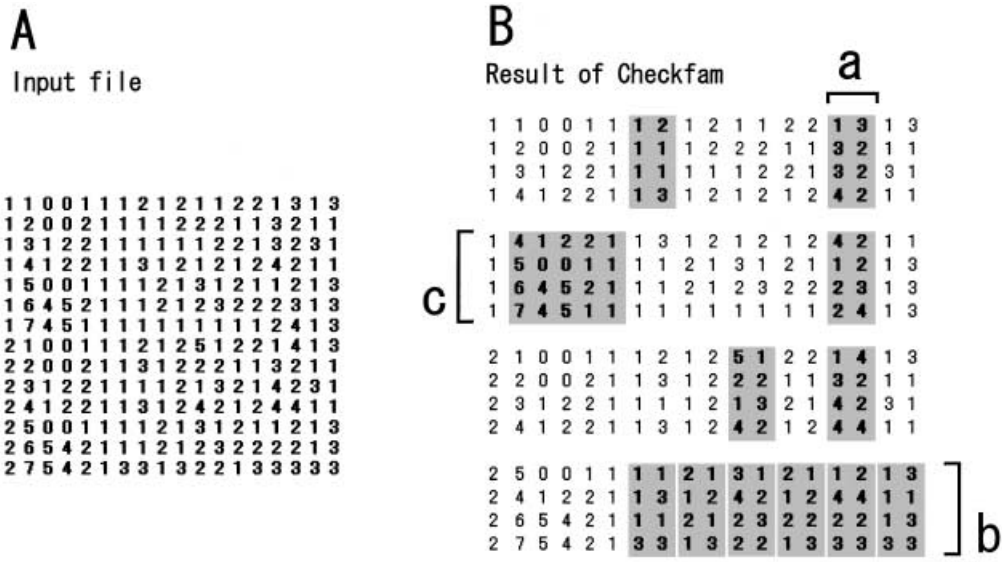

Fig. 1. **A** The form of data for input compatible with the Linkage format ("prefile"). For details, refer to Terwilliger and Ott (1994). **B** The output data reported by Checkfam when the data in **A** were applied to the program. Explanations for blocks *a–c* are in the text

When the parents of a nuclear family are untyped, the nuclear-family algorithm rarely detects Mendelian inconsistencies. Similarly, Checkfam rarely detects inconsistencies in families such as those for affected sib-pair analysis when the parents are untyped. In such a case, a more sophisticated approach is necessary (Boehnke and Cox 1997; Goring and Ott 1997; Broman and Weber 1998). Although Checkfam detects all Mendelian inconsistencies, it is usually difficult to decide which subject has the error from the program output data. When the nuclear-family algorithm detects a genotype error, the user may easily find the location of the error by searching the data for the nuclear family. When the modified genotype-elimination algorithm reports an error, however, it is difficult to determine which family member has an error. In such a case, the "critical-genotype algorithm" and "odds-ratio algorithm" in PEDCHECK may be helpful (O'Connell and Weeks, 1998).

For practical use, however, checking the marker data by the nuclear family algorithm is sufficient when the data at several marker loci are added. Researchers often wish to check inconsistencies in genotyping data before data collection is complete. In such a case, they can use Checkfam to check the data currently available. If the output suggests technical problems at a marker, they can perform additional genotyping at the marker for all the subjects. When error is suggested only for a nuclear family at a marker, then researchers can retype only for the nuclear family at that marker. When a nuclear family has inconsistencies at many marker loci, researchers may check relationships in that nuclear family.

Our program runs through a WWW browser, and can be used by researchers who are not familiar with computer software. The users can either type in the input data or can cut and paste the data to be examined. The software will report the results to users. Checkfam has been implemented on a home page (http://www.genstat.net/checkfam/) and can be used by computer nonspecialists. It is likely to be useful for researchers who perform genotyping for families.

Acknowledgments This study was supported by the Research for the Future Program of the Japan Society for the Promotion of Science.

References

Boehnke M, Cox NJ (1997) Accurate inference of relationships in sib-pair linkage studies. Am J Hum Genet 61:423–429

Broman KW, Weber JL (1998) Estimation of pairwise relationships in the presence of genotyping errors. Am J Hum Genet 63:1563–1564

Ehm MG, Kimmel M, Cottingham RW Jr (1996) Error detection for genetic data, using likelihood methods. Am J Hum Genet 58:225–234

Goring HH, Ott J (1997) Relationship estimation in affected sib pair analysis of late-onset diseases. Eur J Hum Genet 5:69–77

Kruglyak L, Daly MJ, Reeve-Daly MP, Lander ES (1996) Parametric and nonparametric linkage analysis: a unified multipoint approach. Am J Hum Genet 58:1347–1363

Lange K, Goradia TM (1987) An algorithm for automatic genotype elimination. Am J Hum Genet 40:250–256

Lange K, Weeks DE (1989) Efficient computation of lod scores: genotype elimination, genotype redefinition, and hybrid maximum likelihood algorithms. Ann Hum Genet 53 (Pt 1):67–83

O'Connell JR, Weeks DE (1998) PedCheck: a program for identification of genotype incompatibilities in linkage analysis. Am J Hum Genet 63:259–266

Saito A, Saito M, Kitamura Y, Akama H, Kamatani N (2001) Simulation of genotypes using inheritance vector. Eur J Hum Genet 9, suppl 1:342

Stringham HM, Boehnke M (1996) Identifying marker typing incompatibilities in linkage analysis. Am J Hum Genet 59:946–950

Terwilliger JD, Ott J (1994) Handbook of human genetic linkage. The Johns Hopkins University Press, Baltimore