Pedigree Reconstruction using Identity by Descent

Bonnie Kirkpatrick

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2010-43
http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-43.html

April 20, 2010

Acknowledgement

Thanks to Prof. Song for several lively discussions.

Pedigree Reconstruction using Identity by Descent

Bonnie Kirkpatrick

April 20, 2010

Currently, pedigrees are constructed by careful survey of the parent-offspring relationships between individuals in an extended family. The survey is usually conducted by interviewing potential subjects and by examining birth records. The manual labor involved in conducting these surveys is quite expensive and the resulting data can be incomplete or erroneous. In this paper, we present an alternative formulation of pedigree relationships that may be useful either for inferring pedigrees from micro-satellite data or for focusing a genealogical survey towards parts of the pedigree that are poorly resolved.

Much of the early work on pedigree reconstruction relied on a graphical model of inheritance in pedigrees, where the reconstruction algorithms choose pedigree graphs that maximized the likelihood of the observed data [1]. That formulation of the pedigree reconstruction problem is a typical example of parametric structured machine learning where the graphical model of interest is the pedigree model. The work presented here is a departure from parametric methods and develops combinatorial methods for estimating pedigree structures.

(Manuscript revised Apr 20, 2010. Manuscript drafted on May 16, 2008 as part of a class project for CS294-26/STAT260: Computational and Mathematical Population Genetics with Prof. Yun Song.)

1 Background

For diploid individuals, the traditional formulation of a pedigree is a directed graph where individuals are nodes and where every edge represent parentage, i.e. there is a directed edge $i \rightarrow j$ if and only if i is the parent of j (see Figure 1). Specifically, for each type of chromosome, this edge represents the transmission, from parent i to offspring j, of a single recombinant copy of that chromosome. In this formulation it is clear that the accuracy of the edges is of paramount importance and that the presence or absence of a single edge will determine whether many pairs of individuals are related to each other.

Whereas a pedigree represents all possible inheritance paths, identity by descent (IBD) can be thought of as the instantiation of particular inheritance paths for a single locus. For example, if individuals 14 and 15 inherit allele A from their mother, then those two individuals would be IBD for allele A (Fig. 1). If person 4 were to have allele A, it is possible for that allele to be IBD with the A allele found in persons 14 and 15. If it were IBD, then the A allele in person 14 would be

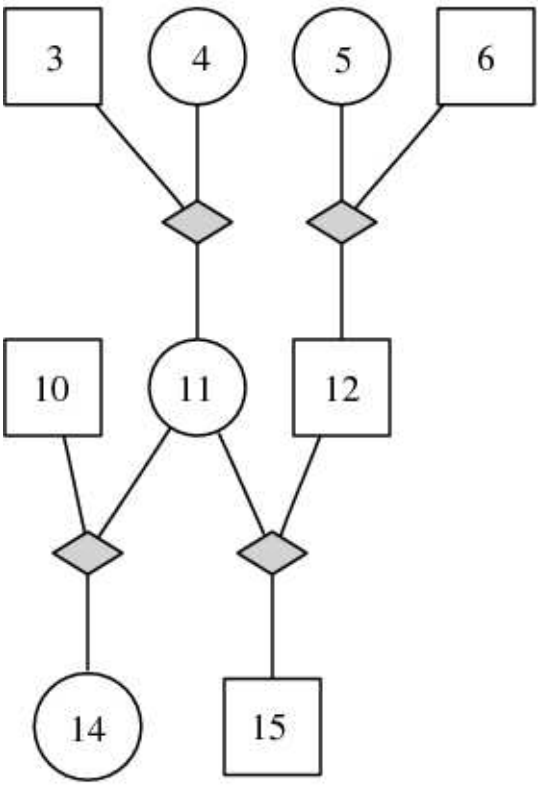

Figure 1: **An Example Pedigree.** Each node is an individual, with boxes representing males and circles representing females. The diamond nodes represent marriages, and the two individuals adjacent to and above the marriage node are the parents of the adjacent individual(s) below the marriage node. The marriage node is simply a slightly more compact way of representing edges from parents to children. Time proceeds in the downward direction, implicitly directing this graph. It is standard to discard the arrows on the edges and to use the top-to-bottom ordering of nodes in the graph to convey the directionality. Individuals without parents are called *founders*, and they are assumed to be unrelated.

an inherited copy of the A allele in person 4. This can be contrasted with identity by state (IBS) where an allele has the same type, but may or may not be inherited from the same ancestor. For example, if person 6 had an A allele, that allele clearly has the same state as the allele in person 4, but person 6 did not inherit that allele from 4 or from an ancestor of 4. We would say that the A-allele for 6 shares IBS and not IBD with an A allele in person 4.

The genetic data available for diploid individuals is genotype data. Very roughly, this data gives us the IBS states of every person at each locus. Continuing with the example above, a genotype of person 4 may be $\{A, B\}$ to indicate that this person has the A allele on one chromosome and the B allele on the other chromosome. If person 6 has genotype $\{A, C\}$ at the same locus, then our discussion of IBS versus IBD would hold. However, the ambiguity in the genotype is the lack of order, meaning that we do not know whether six's A allele was inherited from her father or her mother.

2 Methods

2.1 Reconstruction using Descent Splits

An alternative formulation of a pedigree would allow the hypothesis that a set of individuals is descended from a common ancestor (called a **descent split**), without specifying the number of generations between each of the individuals and their common ancestor(s). the presence or absence of a hypothesis may only change the closeness of the relationship between a pair of individuals (perhaps from cousins to 2nd-cousins), rather than removing the relationship entirely. This is in contrast to the traditional formulation of a pedigree as a collection of parent-offspring edges, where a missing edge entirely changes the nature of many relationships.

Definition. Let I be the set of individuals in a pedigree, and let X be the set of genotyped individuals in a pedigree. The **descent split** (or **d-split**) of an individual $i \in I$ is defined as an ordered bi-partition of X:

$$\begin{aligned} D_i(X) &= D_i^d \mid I \setminus D_i^d \\ &= \{j \in X |\ j \text{ is descended from } i\} \ \mid\ \{j \in X |\ j \text{ is not descended from } i\} \end{aligned}$$

where an individual is *not* a descendant of itself. For a particular set of interest, X, refer to the set of d-splits as $\mathcal{D}_X = \{D_i(X) \mid i \in I\}$.

The bi-partition of a d-split is ordered, because the left set in the partition specifies some relationship between all the individuals in D_i^d, whereas the right set in the partition is agnostic to relationships among the individuals in $X \setminus D_i^d$. For the example given in Figure 1, the full set of d-splits, $\mathcal{D}_I$, are: $D_{14} = \emptyset | I$, $D_{15} = \emptyset | I$, $D_{10} = \{14\} \mid I \setminus \{14\}$, $D_{12} = \{15\} \mid I \setminus \{15\}$, $D_{11} = \{14, 15\} \mid \{3, 4, 5, 6, 10, 11, 12\}$, $D_3 = \{11, 14, 15\} \mid \{3, 4, 5, 6, 10, 12\}$, $D_4 = \{11, 14, 15\} \mid \{3, 4, 5, 6, 10, 12\}$, $D_5 = \{12, 15\} \mid \{3, 4, 5, 6, 10, 11, 14\}$, and $D_6 = \{12, 15\} \mid \{3, 4, 5, 6, 10, 11, 14\}$. Similarly, if we restricted our attention to $X = \{14, 15\}$, then $\mathcal{D}_X$ would contain: $\emptyset | I$, $\{14\} \mid I \setminus \{14\}$, $\{15\} \mid I \setminus \{15\}$, and $\{14, 15\} \mid I \setminus \{14, 15\}$;

The term "descent split" is deliberately chosen to evoke the image of a split in a perfect phylogeny. Just as a set of splits determines a class of perfect phylogeny trees that are compatible with the splits, a set of descent splits specifies a class of pedigree graphs that are compatible with the splits. We will formalize this idea with several lemmas.

Lemma 2.1. *Let $\mathcal{D}_I = \{D_i(I) \mid i \in I\}$ be the d-splits defined by a pedigree P. This set can be used to reconstruct a unique pedigree which is identical to pedigree P.*

Lemma 2.2. *For pedigree P, let $\mathcal{D}_X = \{D_i(X) \mid i \in I\}$ be the set of d-splits that partition the genotyped individuals $X \subset I$. This set of d-splits specifies a class of pedigrees compatible with the splits. Pedigree P is one of the pedigrees compatible with the d-splits.*

First consider the d-splits in $\mathcal{D}_I$. Any trivial d-split, $D_i^d \in \mathcal{D}_I$ with $D_i^d = \emptyset$, clearly represents an individual that is childless. Therefore these d-splits represent individuals in the most recent generation of the pedigree. Now, find some ancestor i_1 and examine any directed path descending from that person, for example, $i_1 \to i_2 \to ... \to i_{k-1} \to i_k$, where the arrow indicates a directed parent-offspring relationship. We see that the d-splits along that descent path are ordered $D_{i_1}^d \supset D_{i_2}^d \supset ... \supset D_{i_k}^d$. Indeed, the cardinality of the d-split sets $D_{i_j}^d$ strictly decrease as we consider individuals lower in the path. These two ideas result in a simple algorithm for reconstructing the pedigree.

Reconstruction Algorithm.
$Heap := (D_{i_0}, ..., D_{i_k})$ where $|D_{i_0}| \leq |D_{i_1}| \leq ... \leq |D_{i_k}|$
While $\exists D_{i_j} \in Heap$ with minimal $|D_{i_j}|$
$D_{i_j} := pop(Heap)$
Look for D_{i_f} and D_{i_m} such that $D_{i_j} \subseteq D_{i_f}$ and $D_{i_j} \subseteq D_{i_m}$ and for all other D with $D_{i_j} \subseteq D$, $|D_{i_f}| \leq |D|$ and $|D_{i_m}| \leq |D|$.
If D_{i_f} and D_{i_m} are found, add the nuclear trio to the pedigree graph, making nodes for i_m and i_f and adding parent edges $i_m \to i_j$ and $i_f \to i_j$.
Else i_j is a founder and has no parents.
End While

Example. If we take the d-splits $\mathcal{D}_I$ from the example in Figure 1, we can apply the algorithm to reconstruct the pedigree. Figure 2 shows the d-splits using a Venn diagram. The upper picture shows the reconstruction generated by the algorithm after the first three iterations. The bottom picture shows the full reconstruction in which the last two iterations of the algorithm construct the second generation of the pedigree.

Proof. of Lemma 2.1
Since we have a d-split for every individual, the algorithm will either assign founder status or parents to every individual. Now, if we look at a single step in the algorithm, each individual will be assigned the correct parents, due to the strictly increasing cardinality of d-splits as we consider d-splits for individuals in older generations. □

Interestingly, we can use the same algorithm when we consider d-splits on a subset of the individuals. As long as we have a separate d-split for each person in the pedigree, we will know the number of

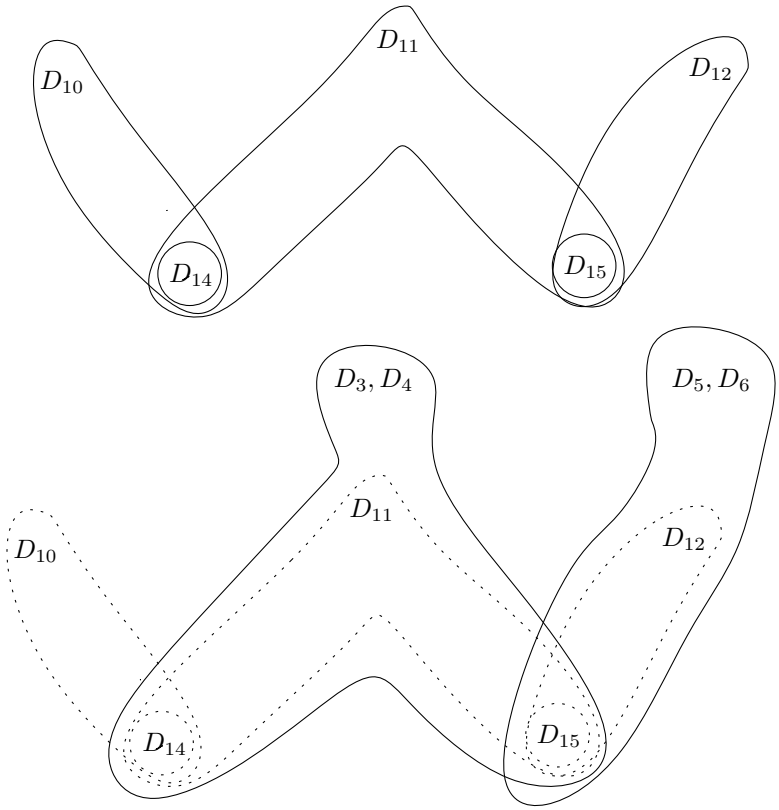

Figure 2: **Reconstructing a Pedigree from the Full D-Splits.** Given the d-splits in $\mathcal{D}_I$ for the set I of all the individuals in the pedigree in Fig. 1, we can use the reconstruction algorithm to recover the pedigree. These are the Venn diagrams of the reconstruction at two different steps in the algorithm. The upper panel shows the first three steps of the algorithm, while the bottom panel shows the complete reconstruction. Each d-split is drawn as a set containing the related individuals. Each set in the diagram is labeled with the name of its d-split, and the names of the d-splits are arbitrary as long as they are distinct.

generations in each lineage. The main difference is that each lineage has *non-decreasing* cardinality of d-splits as we more backwards in time. The missing information, now, is in not knowing which d-split was generated by the parent versus a more distant ancestor. For the example we gave above, if $X = \{14, 15\}$, then $D_{11}(X)$ and $D_3(X)$ indistinguishable.

Proof. of Lemma 2.2
Again, since we have a d-split for every individual, the algorithm will either assign founder status or parents to every individual. Now, if we look at a single step in the algorithm, each individual will be assigned some parents, due to the non-decreasing cardinality of d-splits as we consider d-splits for individuals in older generations. However, the reconstruction will be different for re-orderings of the d-splits. This means that we cannot resolve the correct labels for individuals $I \setminus X$ in the interior of the pedigree. □

2.2 Reconstruction using Identity by Descent

In a practical sense, d-splits are not directly observable from genetic data. Furthermore, when we have individuals of interest, any information about the d-splits of their ancestors will be conflated, since we may not know how many individuals are in each lineage. In addition, d-splits are a feature of the relationships, and they are invariant to the specific inheritance paths at particular loci. How can we find d-splits in data, when our data are locus specific?

The closest theoretical construct to a d-split is IBD. Unfortunately, for a particular locus, IBD may yield fragments of some of the d-splits. We would need a certain amount of data to guarantee that we see IBD between every pair of related individuals. Worse still, multiple types of relationships may yield the same IBD states only with different frequencies.

For example, perhaps person 15 in Fig. 1 inherited an A from ancestor 4 while person 14 inherited a C from ancestor 3. In this case, we would have an IBD state which we could write as $\{4^p, 11^m, 15^m\} \mid\mid I^p \cup I^m \setminus \{4^p, 11^m, 15^m\}$, where 15^m is the maternal allele of individual 15, and the designation of the paternal allele 4^p as the A allele is arbitrary due to 4 being a founder. The double bar will serve to distinguish between our IBD notation and the d-split notation.

Of course, if we consider only individuals of interest, $X = \{14, 15\}$, then this IBD state becomes $\{15^m\} \mid\mid X^p \cup X^m \setminus 15^p$. Indeed, the list of possible IBD states for the alleles of X are: $\{15^m\} \mid\mid \{14^m, 14^p, 15^p\}$, $\{14^m\} \mid\mid \{14^p, 15^m, 15^p\}$, $\{14^m, 15^m\} \mid\mid \{14^p, 15^p\}$, and $\emptyset \mid\mid \{14^m, 14^p, 15^m, 15^p\}$. Unfortunately, if we consider a sibling pair, also named 14 and 15, the same IBD sets are possible for the maternal allele. However, sibling pairs can have IBD at both the maternal and paternal alleles simultaneously: $\{14^m, 15^m\} \mid\mid \{14^p, 15^p\}$ and $\{14^m, 15^m\} \mid\mid \{14^p, 15^p\}$.

The number of possible IBD states for a pedigree is far greater than the number of d-splits, so how would it help us to consider the IBD state space? IBS alone reveals little about relationships, because it cannot disambiguate between a common recent ancestor and a common state. IBD however allows us to exploit LD to learn which instances of common state are due to common ancestry. By considering the possible IBD states that are compatible with the observed IBS at neighboring loci, we can better estimate inheritance than by IBS alone. Of course, knowledge of inheritance is critical for learning the d-splits.

This suggests a Markov model with two levels of hidden states. The observations are the IBS states at each locus and the first level of hidden states is the IBD state at each locus. The second level of hidden states collates the IBD states into d-splits. In order to prevent the algorithm from producing an arbitrarily deep pedigree, it may be necessary to use a regularization parameter to limit the number of generations implied by the d-splits or perhaps to optimize the number of d-splits.

The model we choose is a simple model of IBD that disallows recombinations. The model forbids transitions in IBD state from locus to locus, meaning that there is zero probability of seeing IBD state s_m at locus m given a different state at locus $m-1$, $s_{m-1} \neq s_m$. The implementation involves dynamic programming over an exponential number of IBD states. The next level of the model will collapse the IBD states onto the compatible d-splits, and essentially score the d-splits. This last portion of the model is not yet fully formulated.

3 Results

The zero-recombination model for IBD was implemented in C++. Results were discouraging. Due to the exponential running time of these algorithms, the IBD prediction failed to work well for more than roughly 10 typed individuals.

4 Conclusions

Here we present a novel theoretical justification for pedigree reconstruction. The attractiveness of this approach is precisely that it may avoid maximum likelihood computations on a pedigree graphical model. In addition, this formulation allows for relationships to be established without establishing the exact nature of those relationships. This presents the option of using a pedigree reconstruction algorithm to direct a genealogical survey towards portions of the pedigree that are poorly resolved.

Clearly an exact algorithm for finding the d-splits using IBD will be exponential in the number of individuals. Indeed there is also an exponential running-time for reconstruction algorithms that use a maximum-likelihood approach. However, this formulation of d-splits and non-recombinant IBD may be amenable to clever approximations or data structures that facilitate the computations and make the method practical for larger numbers of individuals. The d-split formulation may also aid the development of approximation algorithms that are fast enough for practical applications.

Future work involves finding a frequentist method of scoring a proposed pedigree against a null model of unrelatedness. Another area of interest is to establish connections between this model and the maximum-likelihood approach.

References

[1] E. A. Thompson. *Pedigree Analysis in Human Genetics.* Johns Hopkins University Press, Baltimore, 1985.