PLIGHT: A tool to assess privacy risk by inferring identifying characteristics from sparse, noisy genotypes

Prashant S. Emani[1,2], Gamze Gürsoy[1,2], Andrew Miranker[2], Mark B. Gerstein[1,2,3,4*]

[1]Program in Computational Biology and Bioinformatics, Yale University, New Haven, CT 06520, USA
[2]Department of Molecular Biophysics and Biochemistry, Yale University, New Haven, CT 06520, USA
[3]Department of Computer Science, Yale University, New Haven, CT 06520, USA
[4]Department of Statistics & Data Science, Yale University, New Haven, CT 06520, USA.

*Corresponding author: pi@gersteinlab.org

The leakage of identifying information in genetic and omics data has been established in many studies, with single nucleotide polymorphisms (SNPs) shown to carry a strong risk of reidentification for individuals and their genetic relatives. While the ability of thousands or hundreds of thousands of SNPs (especially rare ones) to identify individuals has been demonstrated, here we sought to measure the informativeness of even a sparse set of tens of noisy, common SNPs from an individual, by putting the genotype-based privacy leakage from an individual on quantitative footing. We present a computational tool, *PLIGHT* ("**P**rivacy **L**eakage by **I**nference across **G**enotypic **H**MM **T**rajectories"), that employs a population-genetics-based Hidden Markov Model of recombination and mutation to find piecewise matches of a sparse query set of SNPs to a reference genotype panel. Given the ready availability of auxiliary sources of noisy genotype data – such as acquiring small samples of environmental DNA or learning about someone's Mendelian diseases and physical characteristics – inference on sparse data becomes a genuine concern. We explore cases where query individuals are either known to be in databases or not, and consider both simulated "mosaics" of genotypes (i.e. genotypes stitched together from diploid segments sampled from two or more source individuals) and actual genotype data obtained from swabs of coffee cups used by a known individual. Our findings are as follows: (1) Even 10 common SNPs (minor allele frequency > 0.05) often are sufficient to identify individuals in conventional genomic databases. (2) We are able to identify first-order relatives (parents, children and siblings) of query individuals with 20-30 common SNPs. (3) We find some potential for leakage of phenotypic information, based on a simulated attack by combining polygenic risk scores (PRSs) of the piecewise genotypic matches. We also found, for simulated mosaics of two individuals, that 20 common SNPs were often sufficient to find the correct identities of both component individuals. Finally, applying PLIGHT to coffee-cup-derived SNPs, we find that our tool is able identify the individual (when present in the reference database) using as little as 30 SNPs; alternatively, when the individual is not present in the reference database, we reconstruct possible genomes for the individual based on just 30-90 query SNPs by piecewise matching to the reference haplotype database. In this way, we are able to perform a small degree of imputation of unobserved query SNPs. Overall, the tool could be used to determine the value of

selectively masking released SNPs, in a way that is agnostic to any explicit assumptions about underlying population membership or allele frequencies.

1. Introduction

Privacy concerns in the digital age are ubiquitous, with the three, ever-increasing spheres of individual data collection, access, and tools of attack colluding to render individuals vulnerable to a significant risk of compromising data exposure. Incursions upon individual privacy include the removal of personal control over the uses of such data, and the possibility of being subjected to discrimination on the basis of revealed information. Perhaps the most invasive forms of such attacks involve gaining access to information on the physical and mental constitution of an individual without their knowledge or consent. Such breaches are becoming increasingly likely in an era marked by massive health-based data collection and digitization efforts, whose ultimate goals include the provision of personalized medical interventions. Genetic data lie at the heart of these tailored medical approaches, as many phenotypes are believed to have an ultimate basis in our genetic makeup, and as such, are being collected as a part of large-scale projects such as the UK Biobank[1] and the NIH's AllofUs[2] program.

Based on essential work by Homer et al[3], genetic data in general, and single nucleotide polymorphisms (SNPs) in particular, were shown to enable the identification of individuals in DNA mixtures. Additional work reaffirmed the ability of SNPs to reveal whether an individual belonged to a study cohort or DNA mixture[4]. Gymrek et al[5] exploited the possibility of linking the genomes of individuals to surname data from genealogy websites, thereby clearly demonstrating the risk of exposure from the public release of genotype data. Nowadays, law enforcement agencies frequently employ SNP data in the identification of individuals, and their genetic relatives. While it has been suggested 20-30 independent SNPs are enough to re-identify individuals[6], this quantification needs to be updated based on available databases of human genomes and auxiliary biological data.

The increasing availability of cheap sources of SNP extraction are a cause for concern. SNPs can be extracted from genotyping and omics assays, say, as a part of genetic studies of disease vulnerability; forensics analyses of found objects; and can be inferred from established genotype-phenotype relationships. Functional genomics assays in particular allow both direct sequence-based extraction of variants[7], as well as indirect inference based on gene expression values and associated loci[8], and the ubiquity of large-scale omics projects make this source of variants especially concerning[7]. Several studies have demonstrated the risk of identification even in the case of partial privacy preservation measures such as SNP beacons[9–11], and the publication of only GWAS summary statistics[12]. Having established a strong case for the restriction of access to genotypic data, the field is now directing attention towards the notions of data sanitization[7] and comprehensive privacy preservation, say, through encryption-based analysis[13].

All such arguments in favor of protectionism are frequently confronted by the scientific community's desire for unrestricted access to datasets: undoubtedly, increased public access to data would democratize information, and enable biological analyses of greater statistical power in proportion to the size and quality of the available data. Striking a balance between the two requires a clear quantification of the risks of re-identification and inference, relative to the proposed benefit of releasing the data. In response to this challenge, we provide a computational

tool that assesses the degree to which a set of released SNPs could lead to genotypic and phenotypic inferences, using a Hidden Markov Model (HMM) approach (**Figure 1**). The tool is termed "**P**rivacy **L**eakage by **I**nference across **G**enotypic **H**MM **T**rajectories" or *PLIGHT*.

The premise of the tool is that even limited, noisy and sparsely distributed genotypic information carries with it a certain risk of identification and downstream inference. We frame concerns about noisy and sparse data, in particular, by envisioning means by which the presence of a few SNPs in an individual's genome may be inferred. Certain physical characteristics, and knowledge of Mendelian disorders in individuals, potentially leak information on mutations at particular loci[14]. Could sufficient information be gleaned from a photograph, or through a seemingly innocuous conversation, to expose an individual to risk of genetic identification? If a set of noisy SNPs could be assessed indirectly through inference, or through direct access to genetic material acquired from objects in contact with an individual[7], we show that it is often possible to expand upon this partial information using existing genetic databases. The assessment of identification risk is especially important in light of the availability of large-scale public genotypic databases of individuals stratified based on geographical and putative ethnic groupings, as well as SNP sets associated with (potentially socially compromising) disease risk.

Inspired by imputation methods such as IMPUTE2[15] and Eagle[16], the inference procedure in PLIGHT is based on the Li-Stephens model[17], where an HMM is used to explore the space of underlying pairs of haplotypes in a diploid genome with the possibility of *de novo* mutations and recombination between haplotypes (Figure 1A). A solution to the inference problem consists of a set of best-fit haplotype pairs at each observed locus, each pair being linked to another pair at the next locus, to form a set of piecewise matches to reference haplotypes. If multiple equally likely solutions exist, the method identifies all of them. Collectively, these form a set of genotypic *trajectories* through reference haplotype space, where a trajectory is defined as a sequence of reference haplotype pairs (for a diploid genome) at each locus that best fit the observations: that is, for observed query SNPs at genomic loci $l = \{1,2,\cdots,L\}$, the trajectory $\mathcal{T} = \{j(l), k(l)\}_{l=1}^{L}$, where j and k are the labels of best-fit reference haplotypes as a function of the observed genomic position (Figure S1). We label a single best-fit trajectory as a diploid *mosaic* genome if it exhibits a varying reference-haplotype composition across the observed loci (Figure 1B). We also note that the task of identifying all possible genotypic trajectories consistent with a query overlaps considerably with the construction of graph genomes[18,19] (see Discussion). PLIGHT is designed to be agnostic to any assumptions of membership of the query individual to a specific subpopulation, population homogeneity or to estimates of allele frequencies. It is, however, conditional on the choice of reference population.

The employment of HMMs in the modelling of linkage disequilibrium (LD) in genomes has substantial precedent, such as in the inference of local ancestry[20–22] and the determination of regions that are Identical by Descent (IBD)[23]. For example, Baran et al[22] use a latent-state-space reduction by running a two-level HMM: an inner model for each ancestral population within a genomic window of a certain length and a higher-level model for exchanges between the windows. We have chosen a more straightforward implementation of HMMs, so as to avoid any consequent assumptions either of the density of query SNPs (defining a genomic windows precludes recombination between SNPs in that window) or of ancestral membership. There is, however, a consequent price paid in terms of an increase in running time and memory usage. We partially ameliorate these costs using the methods described in the article. Finally, the Positional

Burrows-Wheeler Transform (PBWT)[24] has been used in tandem with the Li-Stephens model of recombination to improve the efficiency of haplotype matches across large databases, in a method termed *fastLS*[25]. The scaling of this method with respect to database size is far superior to the straightforward HMM implementation. However, while future developments may increase the scope of its applicability, *fastLS* does not allow position-specific variations in the recombination rate. For sparsely distributed SNPs, there will be large variability in the recombination rates between any pair of adjacent observation sites. PLIGHT allows for variation of the recombination model, and even include position-dependent recombination effects. Further, in contrast to the PBWT formalism, our implementation also includes the possibility for the user to include trajectories that are less-than-optimal, for robustness checks or exploration of noisy data.

In this work, we apply three variant algorithms within the PLIGHT framework to five broad categories of re-identification approaches. First, we determine the average number of SNPs required to uniquely identify an individual known to be in a database, in the presence of varying degrees of genotyping error/mutation. Second, for synthetic genotype (diploid) mosaics of 1000 Genomes phase 3[26] individuals, we attempt to identify the component individuals. These simulated examples serve as the basis for examining the performance of the PLIGHT algorithms, and represent cases of lateral variation in haplotype composition: that is, when the haplotype composition varies along the length of the chromosome. Third, we carry out a series of kinship attacks by trying to identify relatives of query individuals within databases. Kinship attacks require correct inference of the haplotype composition of both chromosomes in a diploid genome, as often only a part of one of the chromosomes is expected to be similar. Fourth, we consider an example of how information can be pooled across multiple genotypic trajectories by calculating polygenic risk scores (PRSs) for the best-fit genotypic segments, and then checking whether these values are similarly distributed as the ground-truth query genome's PRS, relative to PRSs from a background population of diploid genomes (Figure 1C). Finally, we run PLIGHT on a sparse set of noisy SNPs derived from coffee cups (as extracted in ref. [7]) to demonstrate a real-world attack scenario involving a coarse-grained imputation of unobserved SNPs.

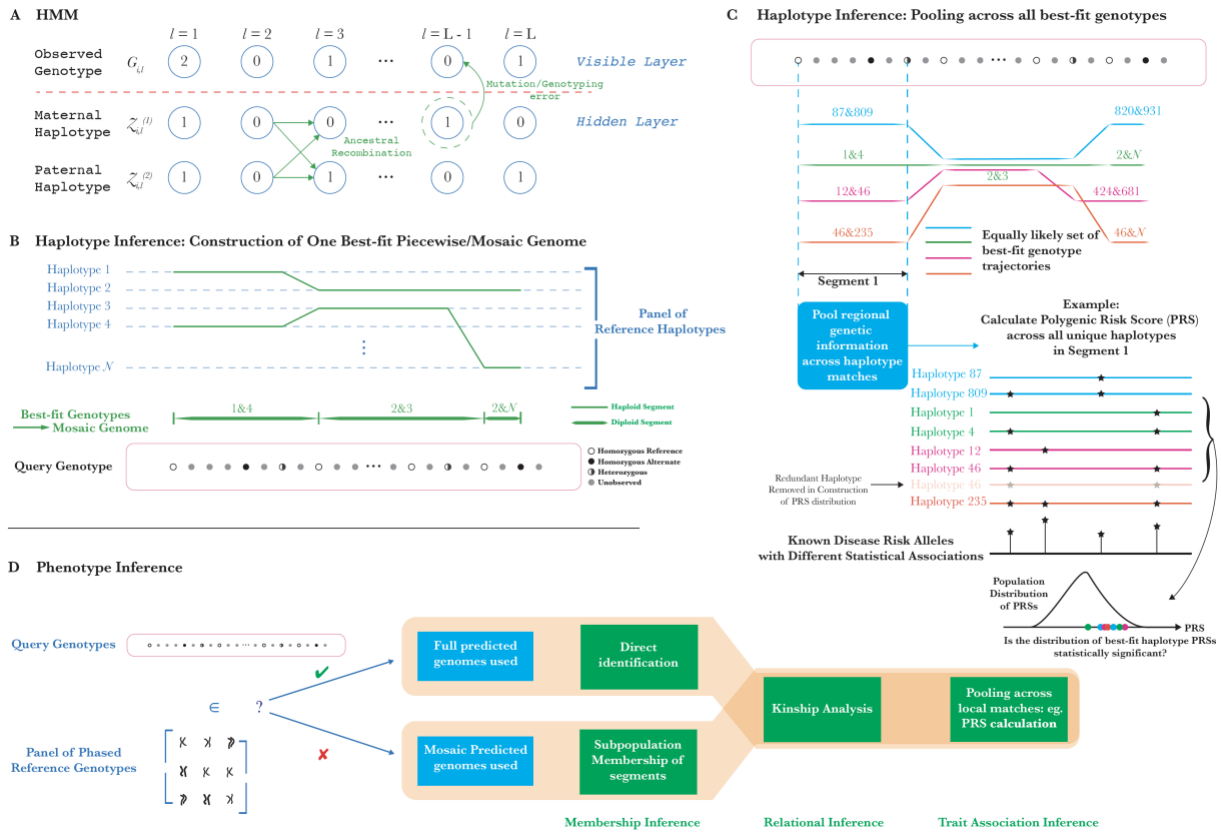


Figure 1. *Methodology of the HMM-based and inference analyses. (A) Schematic of the Li-Stephens HMM model for haplotypic recombination and mutation. (B) Inference of best-fit haplotype pairs in different genomic regions based on a chosen database of reference haplotypes, and the subsequent construction of the diploid mosaic genome. (C) An example of how the haplotype inference procedure described here could be used for novel attacks. By calculating PRSs for particular traits across all identified mosaic genome segments, we can assess whether there is a statistically significant clustering of the PRSs relative to background values. If this is found to be the case, the individual is more likely to be linked to such a trait. (D) Exploration of the types of phenotypic inference possible, conditional on whether the individual is known to be in the database of reference haplotypes or not.*

2. Results

2.1 Repurposing population-genetics models for privacy

The novelty of our approach lies in the application of well-known population-genetics approaches based on HMMs to questions associated with genomic privacy. Thus, the core of the method is an implementation of the Li-Stephens HMM[17], which matches an observed set of diploid genotypes with pairs of maternal and paternal haplotypes. The HMM models the probability of observing the set of input genotypes G_q, conditional on a reference database of haplotypes H: $P(G_q|H) = \sum_{Z_j^{(1)},Z_k^{(2)}} P\left(G_q \middle| Z_j^{(1)}, Z_k^{(2)}\right).P\left(Z_j^{(1)}, Z_k^{(2)} \middle| H\right)$. The paternal and maternal haplotypes, $Z_j^{(1)}$ and $Z_k^{(2)}$, are drawn from the reference database H. Potential

recombination between adjacent observed genomic positions is accounted for by transitions between HMM "states", i.e. haplotypes in a reference database, with *de novo* mutations or genotyping errors quantified by emission probabilities at each observed site (Figure 1A). The optimal set of matching haplotype pairs is found by maximizing $P(G_q|H)$. $P\left(Z_j^{(1)}, Z_k^{(2)}\middle|H\right)$ represents the probability of constructing the haplotype pairs $Z_j^{(\alpha)} = \left\{Z_{j(l),l}^{(\alpha)}\right\}_{l=1}^{L}$ as a composite of the reference haplotypes across all observed loci: $j(l)$ is the reference haplotype label as a function of the observed loci $l = 1, \cdots, L$, where transitions from other reference haplotypes may occur between adjacent loci depending on the recombination model. $P\left(G_q\middle|Z_j^{(1)}, Z_k^{(2)}\right)$ then quantifies how likely it is to observe the query genotypes given those composite haplotype pairs, allowing for mutations and/or genotyping errors.

PLIGHT establishes a state space of reference database haplotypes, with user-defined models of recombination (for example, non-linear rate of growth or hotspot models), and position-specific or non-position-specific mutation/error rates. The optimal path through state space is determined using the Viterbi algorithm[27]. With an input set of query SNPs, the three different algorithms within PLIGHT then proceed to find all the best-fit haplotype pair matches at each observed genomic position, the haplotype pairs being drawn from aforementioned state space. The task is essentially a form of coarse-grained imputation, the coarse-graining arising from the consideration of sparse query SNP sets, except that instead of filling in SNPs at non-observed location, the algorithms infer the labels of all reference haplotype pairs that are consistent with the query SNPs. The entire sequence of haplotype labels across all observations is termed a "trajectory", $\mathcal{T} = \{j(l), k(l)\}_{l=1}^{L}$. Multiple optimal trajectories can be found for a single query SNP set, and PLIGHT outputs all these paths. User-defined parameters also allow for the identification of increasingly sub-optimal trajectories.

The inference attack strategies employed by an adversarial agent are naturally separated into ones dependent on whether an individual is known or expected to be within the reference database being used, or not (Figure 1D). In the former, the goal is often to confirm the presence of the individual in the database, while in the latter, the goal is to infer potentially identifying characteristics using similarities to reference genotypes. In our demonstration of the PLIGHT framework, we consider both.

The other major consideration behind the design of our methods was contending with the scale of reference genotype databases: with the search space for matches to query genotypes ranging from thousands to (eventually) tens or hundreds of thousands of genotypes, HMMs quickly become computationally intractable in terms of the required memory and speed. Our approach to the problem was to construct three algorithms that negotiate the tradeoff between exactness of the calculation and the burden on memory resources: *PLIGHT_Exact* performs the exact HMM inference process using the Viterbi algorithm[27]; *PLIGHT_Truncated* phases in a process of truncating the set of all calculated trajectories to only those within a certain probability distance from the maximally optimal ones, resulting in a smaller memory footprint; and *PLIGHT_Iterative* iteratively partitions the reference search space into more manageable blocks of haplotypes and runs *PLIGHT_Exact* on each block, followed by pooling and repetition of the scheme on the resulting, smaller cohort of haplotypes. Thus, *PLIGHT_Truncated* and *PLIGHT_Iterative* are approximations to the full state

space exploration designed to reduce hard disk memory usage, and both hard disk and RAM usage, respectively. *PLIGHT_Truncated* was inspired by the state-space reduction methods in the Eagle2 imputation program[16], and involves slowly reducing the number of alternate trajectories beyond the highest probability trajectory stored in memory, until it reaches a user-defined fraction of the total number of possible trajectories. However, for most purposes *PLIGHT_Truncated* is superseded by *PLIGHT_Iterative* in performance. *PLIGHT_Truncated* mainly serves to determine the "compressibility" of the trajectories, i.e. the size of the trajectory subspace that is sufficient to match the results of the exact algorithm. *PLIGHT_Iterative* starts with a randomly shuffled set of partitions of the full reference haplotype space, runs the exact algorithm on each partition and pools the resulting best-fit haplotypes. This pooled set becomes the pruned reference haplotype database for the next iteration, with the procedures of (a) running the exact HMM on equally sized partitions and (b) pruning the reference database continuing until no change is observed in the size of the filtered set, or until the size of the pooled set of best-fit haplotypes falls below the pre-set partition size. The procedure is run with a user-defined number of initializations of randomly shuffled partitions. This algorithm has significantly better scaling properties than the others, with respect to the size of the reference database.

The results of all algorithms are visualized using a visualization module called *PLIGHT_Vis*, and processed using downstream inference modules. The overall computational framework is depicted in Figure 2.

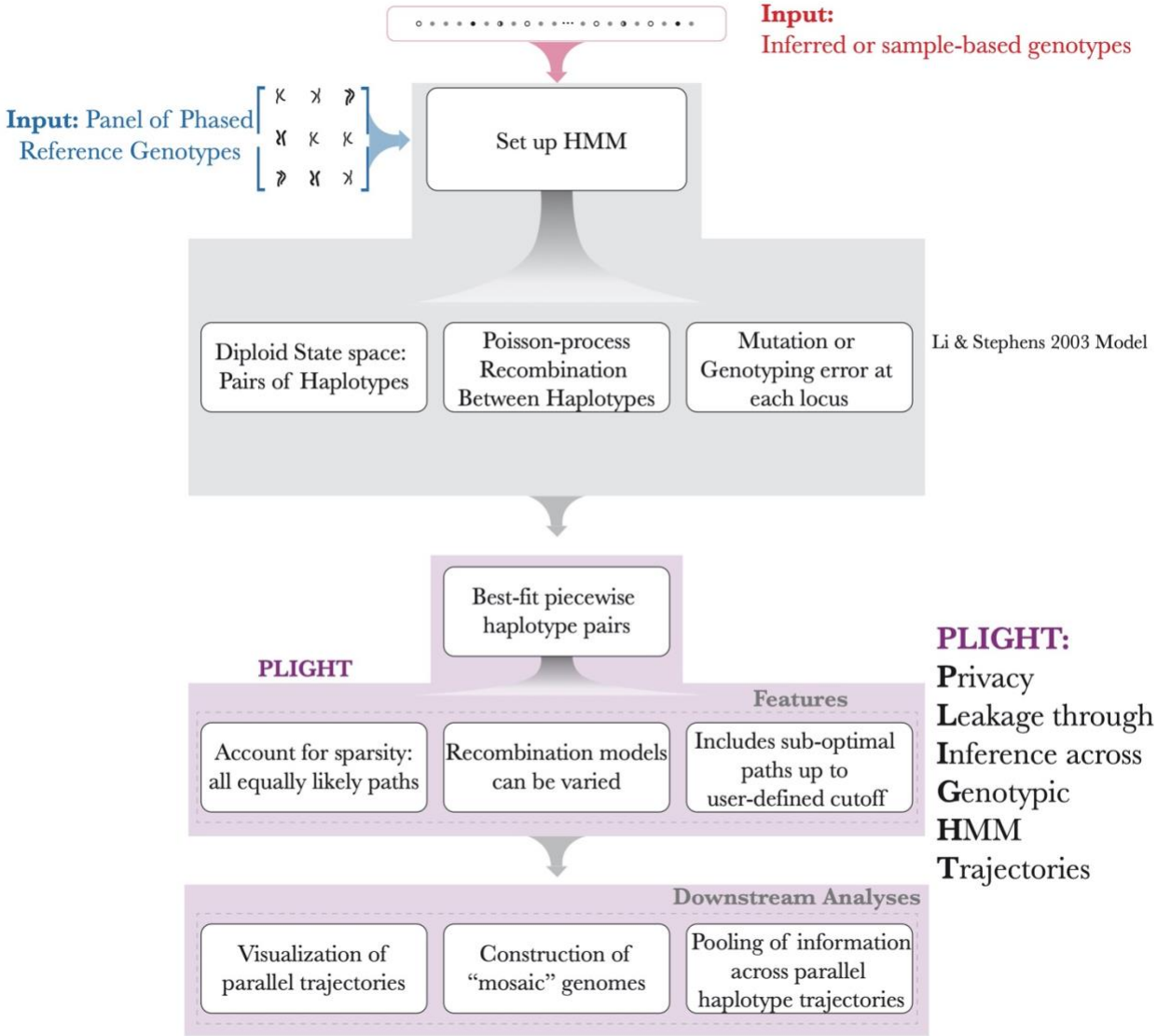


Figure 2. *Computational framework of PLIGHT. The inputs of a reference database and of the query SNPs are shown at the top. Next, the population-genetics model by Li and Stephens forms the core of the HMM-based framework. This includes the definition of a diploid state space, a Poisson-process based recombination model with rate of growth proportional to the genomic distance between two loci, and a mutation/error rate at each locus. This model is implemented within PLIGHT, which identifies the best-fit reference haplotype labels in the diploid state space. Special features include the identification of all equally likely trajectories, flexibility in the recombination models used, and an allowance for sub-optimal trajectories to be identified. Furthermore, PLIGHT includes visualization and mosaic genome reconstruction modules, allowing for other types of analyses involving the pooling of information across all identified mosaic trajectories.*

2.2 Datasets and shared parameters

In some of the following sections, we test our tools using a series of simulations. The primary reference database employed is the 1000 Genomes Phase 3 database[26] (based on the human genome reference build *GRCh37*, fasta file *human_g1k_v37.fasta.gz*[28]), with 2,504 phased genotypes in total from 26 different sampled populations. The methods rely on the availability of phased reference genotypes, and currently work by reading through the chromosome-separated *vcf* files. For all analyses, we filter out the low allele frequency SNPs, with minor allele frequency (MAF) restricted as: $0.05 \leq MAF \leq 0.5$. The lower cutoff is a user-defined parameter in our program with a default value of 0.05. We choose such a cutoff with the intention of quantifying the leakage associated with even relatively common SNPs, as the identifying information contained in low MAF SNPs will be significant.

2.3 Identification of individuals known to be within a database

We start with the simplest scenario, where an individual is known to be within a database. We tested out examples with different numbers of SNPs and varying values of the mutation rate/genotyping error (here we report the value of λ in the results, to simulate the effect of a particular genotyping error rate; see the *Methods* section for details on different mutation rate quantifications). For each of 5 mutation/error rates, we ran 10 iterations of the following simulation process. A single individual is selected at random from the full 1000 Genomes Phase 3 cohort of 2,504 individuals, and the genotype is simulated as describe in the *Methods*. Given the set of 2,504 individuals, the chosen individual is likely to be different for each of the 10 iterations (even though sampling was done with replacement).

The variant selection procedure is repeated until we have $N_{SNP} \in [1, \cdots, 40]$ for a particular simulation. To assess the HMM model, we consider for now only one chromosome (chromosome 17 was chosen at random for this analysis).

Therefore, for each error rate and iteration, we choose a single individual and simulate all 40 possibilities of $N_{SNP} \in [1, \cdots, 40]$ for that individual. The SNP selection is repeated from scratch for each of the 40 cases, so the SNP cohorts are not necessarily the same for the different values of N_{SNP}. Also, note that it is possible to run the simulations with SNPs that are homozygous for the reference allele. This would yield the same trajectory prioritization in the HMM model. However, we attempt here to simulate cases where the focus is on identified alternate alleles, reported from either SNP beacons or noisy functional genomics data.

Finally, we find the mean and standard deviation across the 10 iterations of the minimum value of N_{SNP} for which a unique identification of the individual was made. However, in the presence of a mutation rate, it is possible for these unique identifications to be different from the ground truth samples. Therefore, we also provide the mean and standard deviation of the minimum value of N_{SNP} for which a correct and unique identification is made. For reference, we also provide the mean and standard deviation of the pooled list of MAFs across all 10 iterations for each mutation rate value. We present these results in **Table 1**.

Table 1. *Results of the "individual known to be in the database" analysis, as a function of the scaled mutation rate λ, using $N_{SNP} \in [1, \cdots, 40]$ and 10 iterations (10 individuals chosen randomly with replacement) at each mutation rate. The three columns shown are: the minimum number of*

SNPs required to obtain a unique identification of an individual, the minimum number of SNPs required for that identification to match the true identity of the sample, and the pooled minor allele frequency of all sampled SNPs across the 10 iterations at each mutation rate.

Mutation/Error Rate $\boldsymbol{\lambda} = \frac{\boldsymbol{\theta}}{\mathbf{2(N+\theta)}}$	**Minimum number of SNPs for unique ID,** N_{SNP}^{Unique}**: Mean** $\pm$ **S.D. (10 iterations)**	**Minimum number of SNPs for correct ID,** $N_{SNP}^{Correct}$**: Mean** $\pm$ **S.D. (10 iterations)**	**Pooled MAF distribution across all 10 iterations: Mean** $\pm$ **S.D.**
0.0	6.7 $\pm$ 0.9	6.7 $\pm$ 0.9	0.25 $\pm$ 0.13
0.05	6.1 $\pm$ 2.0	9.9 $\pm$ 2.5	0.27 $\pm$ 0.13
0.1	7.5 $\pm$ 2.0	9.9 $\pm$ 2.3	0.25 $\pm$ 0.13
0.2	6.1 $\pm$ 2.2	21.6 $\pm$ 7.9	0.22 $\pm$ 0.12
0.3	6.2 $\pm$ 1.9	12.1 $\pm$ 15.5	0.22 $\pm$ 0.13

The results of the simulations indicate that, while an identification can be unique for nearly the same average N_{SNP}^{Unique} across all mutation rates, the ability to find the correct individual worsens with an increasing mutation rate. The mean value of $N_{SNP}^{Correct}$ does not increase monotonically, but the standard deviation does. It is still noteworthy that unique identification can be made with a very small number of common SNPs (~6-8), with MAF values that are distributed fairly evenly across the range $0.05 \leq MAF \leq 0.5$. Even in the presence of modest mutation rates ($\lesssim 0.1$), correct identification requires only about 10 common SNPs on average.

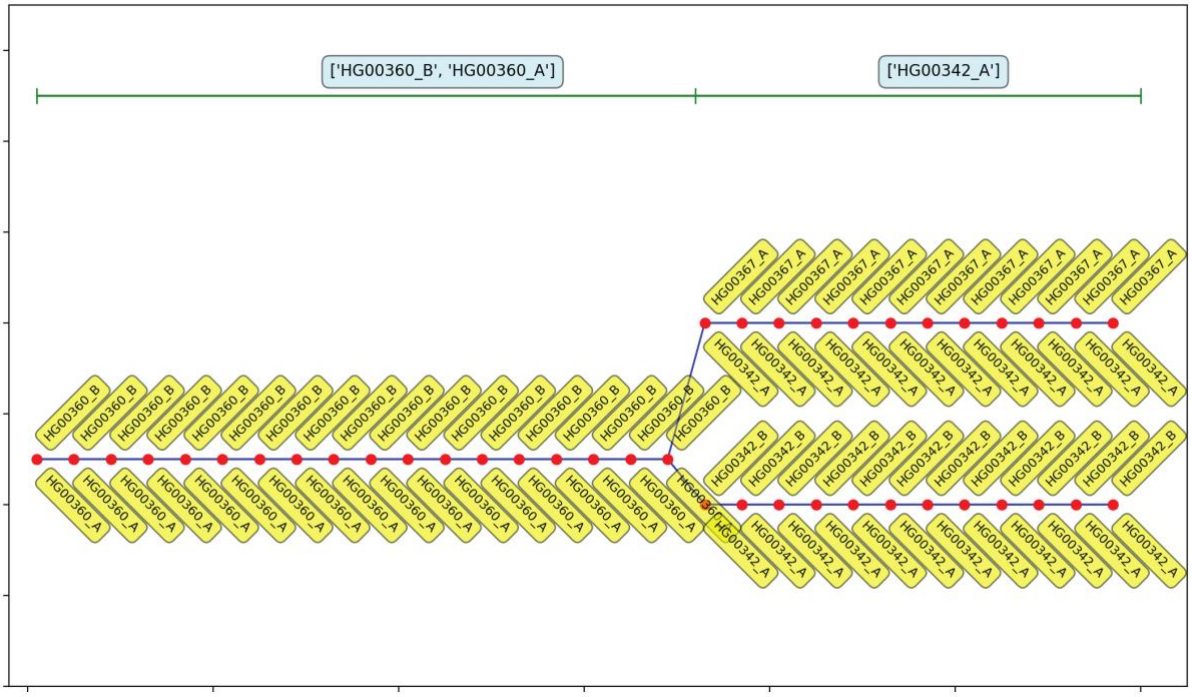
2.4 Mosaic overlap between query individuals and database individuals

We next evaluated the performance of the program on *mosaic* individuals, i.e. individuals whose genome is constructed by sampling the diploid genomes of two or more source individuals, using simulated analyses (the mutational process is the same as in the previous section). A pair of individuals is chosen at random from the 1000 Genomes set of individuals. The diploid mosaic genome is constructed for N_{Chr} sampled chromosomes using the simulation scheme described in the *Methods* section.

2.4.1 Exact search within a reference database of 400 haplotypes. For the first example, we use the *PLIGHT_Exact* module employing the full Li-Stephens model. Two individuals were selected, and the first half of the SNP genotypes were taken from one individual and the other half from the other. The mutation rate was set to 0, while the fixed per base recombination rate c_l was set at 0.5 cM/Mb and the default linear recombination model was used. The choice of 0.5 cM/Mb was used as, based on preliminary calculations, this value led to reasonable exploration of the haplotype space, without devolving into a uniform consideration of all haplotypes as equally probable. To understand this, consider the two extremes: a very low value of c_l will prefer to elongate the same haplotypes for the entire length of the observed SNP positions without crossing over, i.e. the emission probabilities will dominate in the overall likelihood; a very high value will lower the barrier to cross-overs, causing the probabilities to be equal across all choices of haplotypes, i.e. the transition probabilities will dominate in the overall likelihood. The value of c_l = 0.5 cM/Mb is also close to the value of 0.4 cM/Mb chosen in previous HMM studies[15,17].

To limit the dimensionality of the matrices used in the exact case, we only search for matches among 200 individuals (= 400 haplotypes) from the reference database. We sampled $N_{SNP} = 30$ SNPs from each of $N_{Chr} = 3$ chromosomes (chromosomes 1, 2 and 21). The two sampled individuals were HG00360 (first half) and HG00342 (second half). Note that, given the choice of random parameters in the SNP selection process for the simulated query genomes, the total length of the query genomes for each of the chromosomes was much smaller than the length of the chromosomes: in the following example, the SNP positions ranged from 1.03 Mb to 37.3 Mb for chromosome 1, from 0.4 Mb to 22.1 Mb for chromosome 2, and from 14.7 Mb to 38.4 Mb for chromosome 21. The results are provided in **Figure 3** and **Figure S2**.

A



B

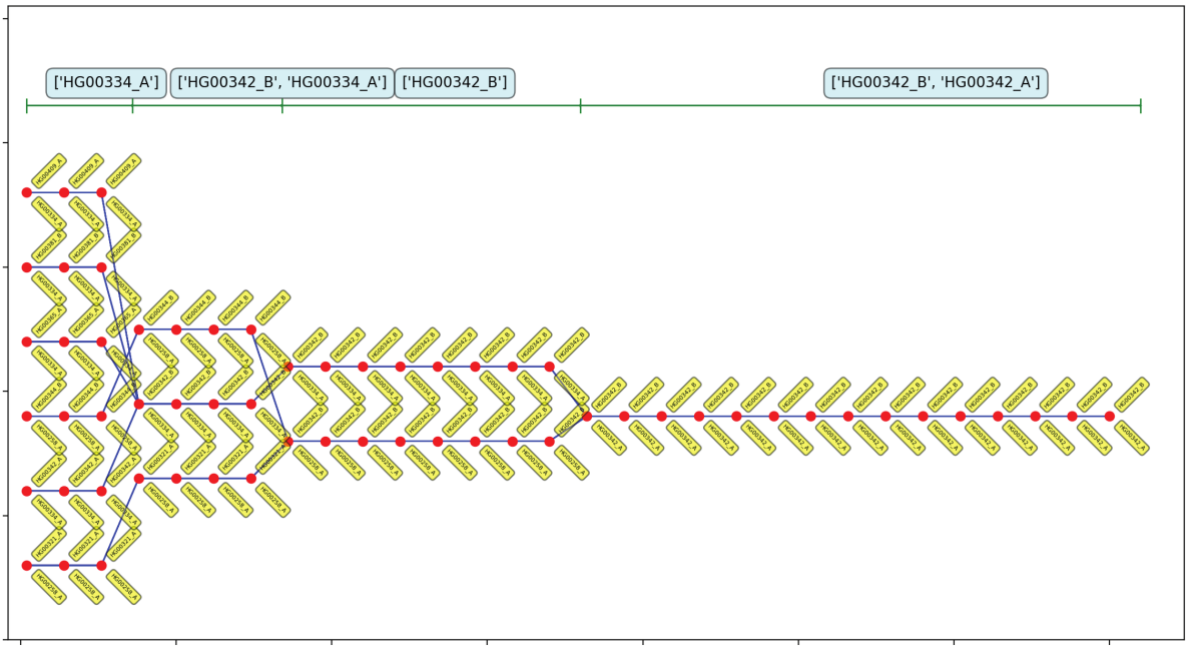


Figure 3. *Best-fit genotypic trajectories for the diploid mosaic genome of HG00360+HG00342 constructed across 30 SNPs each for chromosomes 1 and 2 (results for chromosome 21 shown in* ***Figure S2****). The composition of the best-fit pair of haplotypes at each locus is depicted by two yellow tags, one below and one above the red dots. (A) Trajectory for chromosome 1. (B) Trajectory for chromosome 2.*

Figure 3A shows the genotypic trajectories for chromosome 1, and **Figure 3B** shows the same for chromosome 2 (the results for chromosome 21 are provided as **Supplementary Figure S2**). The labeling scheme we use involves the splitting of the phased genotypes in the reference database into the two component parental haplotypes, with the (arbitrary) haplotype labels "A" and "B" appended to the name of the reference individual. The labels for the pair of haplotypes in the best-fit trajectories are depicted by one yellow tag below and another above the red dot marking each locus of each trajectory. The results for chromosome 1 indicate that the correct mosaic was identified as one of the two genotypic trajectories. The second trajectory consists of a mixture of one haplotype from the true individual (HG00342) and the other from a different individual (HG00367). The two trajectories branch out from HG00360 at the same SNP, indicating that for the last set of SNPs, HG00367_A and HG00342_B are likely identical (there are no mutations in this particular simulation). Indeed, the vcf files confirm this to be the case, at least for the last 13 SNPs in the simulation. The optimal haplotypes for certain stretches of the chromosome are identified at the top of the figures, with the boundaries between the stretches marked by green ticks. These optima are calculated by the simply maximizing the frequency of occurrence of the haplotypes in these regions. An important point to note here is that even though the simulation drew the last 14 SNPs from HG00342, the transition from HG00360 to HG00342 in the solution occurs only for the last 12 SNPs. This is because the trajectories require a certain number of additional genotypic steps to build enough probability to warrant a transition. If the inference process is seen as a combined approach to identify the stretches of best-fit genotypes, as well as the best-fit boundaries between these stretches, then this method does lead to a certain fuzziness in the identification of the boundaries. However, an uncertainty is likely to exist in any dataset, regardless of the method, if there is even a little error in the genotyping or any mutations.

The trajectories for chromosome 2 (**Fig. 3B**), on the other hand, include HG00342, but not HG00360. Several other trajectories and branch points occur in the region of the true HG00360 segment. The reason for the ground truth sample not arising as a part of the solution is likely that, at the current rate of recombination, it is more likely for one of the haplotypes (HG00342_B) to be maintained across a longer stretch of the chromosome, which then results in alternative haplotypes being selected over shorter segments than the true HG00360 segment. To test whether this may be the case, we ran the same sample through a calculation with double the previous recombination rate (= 1 cM/Mb). The results (**Figs. S3A, B and C**) indicate that increasing the recombination rate does indeed cause the true HG00360 segment to be included in the solution. The increase in the recombination rate is more favorable to the transition from one haplotype to another and thus allows for a greater exploration of the haplotype space. This is also evident from the increase in the average number of best-fit haplotypes per SNP evident from **Fig. S3**. The results for chromosome 21 (**Fig. S2**) do include the true individual HG00360, in addition to several alternative trajectories .

In summary, running the code with a few different, reasonable options of the recombination rate may help in robustly identifying the underlying genotypic trajectories.

2.4.2 Truncated algorithm. We subsequently ran the same SNP set through the truncated algorithm to assess the degree of compressibility of the trajectories and the potential for memory reduction. The results, which indicate a considerable degree of compressibility of the best-fit trajectories, are presented in the Supplementary Results and **Figure S4**.

2.4.3 Iterative and approximate search within a reference database of 5,008 haplotypes. The *PLIGHT_Iterative* algorithm contends with the memory requirements of a large reference database by subdividing the database into subgroups and running the HMM on the smaller haplotype cohorts. Each round of subdivision is run several times, with different haplotypes assigned to different subgroups each time. We test the same synthetic input dataset as for *PLIGHT_Exact*, but now search through the entire 1000 Genomes phase 3 database with 5,008 haplotypes. We ran two replicates for each of two sets of parameters: number of iterations (a) $n_{iter} = 20$; and (b) $n_{iter} = 30$; the subgroup size, S_{sg}, is chosen to be $S_{sg} = 300$, and the recombination rate set to c_l = 0.5 cM/Mb. The randomness of the algorithm is apparent from the fact that there is no unequivocal improvement in the identification of component individuals in going from $n_{iter} = 20$ to $n_{iter} = 30$. While it is possible that a significantly larger increase will consistently improve the mixing of the haplotypes in general, these runs are mostly able to identify the true component individuals in the mosaic sample, especially when information across the chromosomes was combined. The *consensus* trajectories, which are trajectories containing haplotypes frequently observed across chromosomes (see *Methods*), include both HG00342 and HG00360 in three out of four of the runs (though not necessarily within the same trajectory). Additionally, the chosen SNPs on each chromosome have different capacities for discerning the ground truth. The consensus results for chromosomes 1 (**Figure 4**) and 21 (data not shown) include the true HG00360+HG00342 combination within the same trajectory for one and two out of the four runs, respectively.

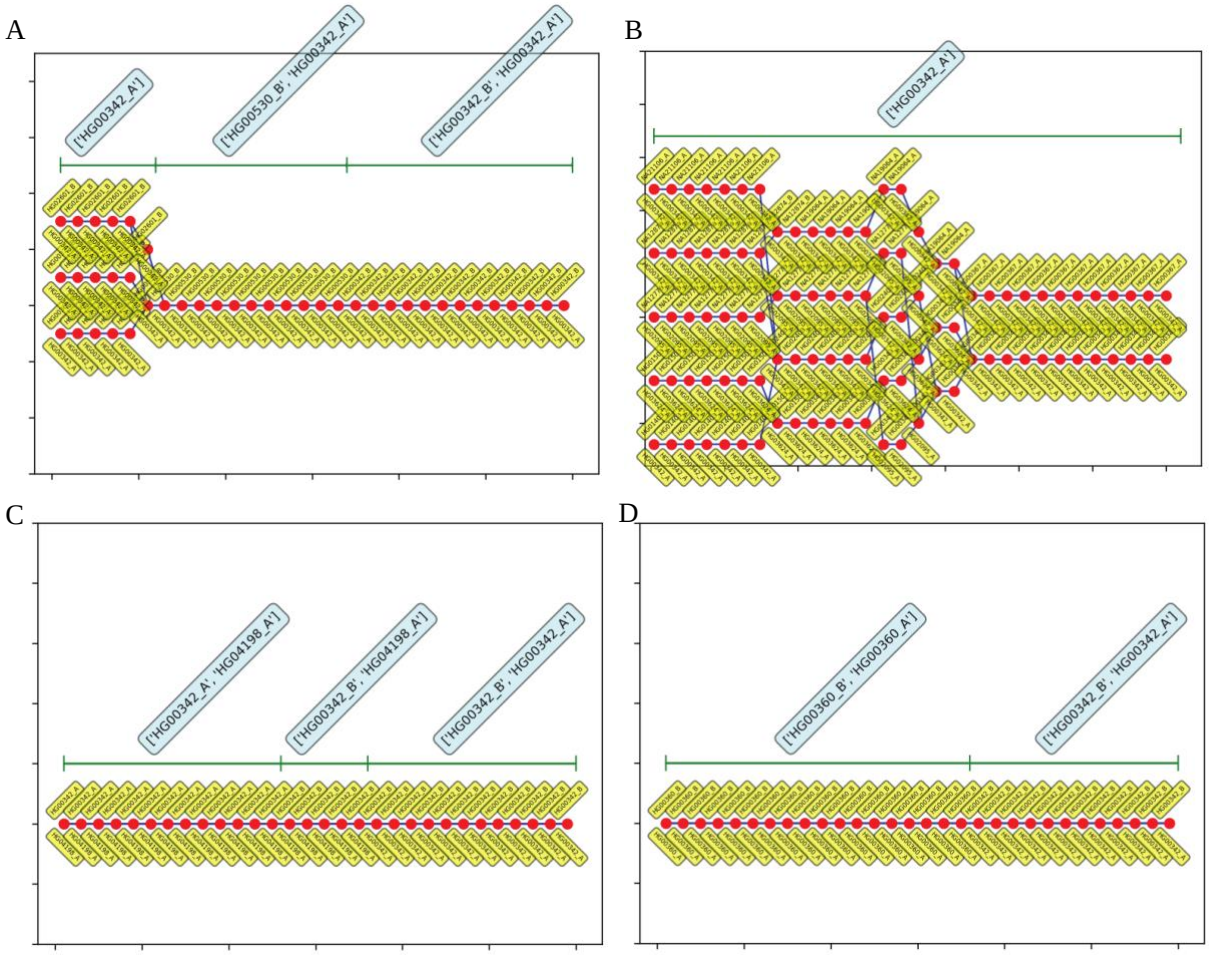


Figure 4. *Consensus genotypic trajectories for the diploid mosaic genome of HG00360+HG00342 constructed across 30 SNPs in chromosome 1, where the consensus score is evaluated by weighting the haplotypes in each trajectory in proportion to their occurrence across all three chromosomes. The composition of the best-fit pair of haplotypes at each locus is depicted by two yellow tags, one below and one above the red dots. (A)* $n_{iter} = 20$*, replicate 1; (B)* $n_{iter} = 20$*, replicate 2; (C)* $n_{iter} = 30$*, replicate 1; (D)* $n_{iter} = 30$*, replicate 2. Shown at the top of each panel are the most frequent haplotypes within each segment indicated.*

In summary, the iterative, mixing algorithm is able to pick out the ground-truth component individuals in the best-fit trajectories, albeit with a certain degree of randomness.

2.5 Kinship analysis

Having analyzed synthetic mosaics in the previous results, we seek to benchmark the methods on known, *natural* mosaics in the form of a kinship analysis. We use a set of 13 individuals taken from the related samples cohort of the 1000 Genomes Phase 3[29] to link them to those individuals among the Phase 3 main release cohort of 2,504 phased genotypes that are stated to be first- (parents, children or siblings), second- or third-order relatives (as provided in an associated pedigree file[30]). For this study, we choose a single chromosome for each individual with the

parameters: (I) $N_{SNP} = 20,\ n_{iter} = 20$; and (II) $N_{SNP} = 30$, $n_{iter} = 30$; the subgroup size was chosen to be $S_{sg} = 300$, and the recombination rate set to c_l = 0.5 cM/Mb. The chromosomes were chosen at random, and are different in general between cases (I) and (II). A successful identification is indicated as the inclusion of the related individual anywhere within the best-fit trajectories. The results are presented in **Table 2**.

Table 2. Results of the kinship analysis for the two sets of parameters. Case I: $N_{SNP} = 20,\ n_{iter} = 20$; and Case 2: $N_{SNP} = 30,\ n_{iter} = 30$. The degree of relatedness is indicated as "1st-, 2nd- and 3rd-order" and when a successful identification is made, the relationships of the identified individuals to the query individuals are explicitly specified, if available from the pedigree file[30].

Individual Sampled Chromosomes in case I and in case II	**Case I: Identified relatives (1st-, 2nd- and 3rd-order)** $N_{SNP} = 20,\ n_{iter} = 20$ If available, relation is specified.	**Case II: Identified relatives (1st-, 2nd- and 3rd-order)** $N_{SNP} = 30,\ n_{iter} = 30$ If available, relation is specified.
HG00124 Chr 19, Chr 21	2nd-order: **0/1**	2nd-order: **1/1**
HG00501 Chr 14, Chr 8	1st-order: **0/3**, 2nd-order: **0/2**	1st-order: **1/3 (sibling)**, 2nd-order: **0/2**
HG00635 Chr 7, Chr 16	1st-order: **0/3**, 2nd-order: **0/2**	1st-order: **1/3 (sibling)**, 2nd-order: **0/2**
HG00702 Chr 9, Chr 21	1st-order: **2/4 (mother and father)**	1st-order: **2/4 (mother and father)**
HG00733 Chr 10, Chr 11	1st-order: **1/2 (father)**	1st-order: **2/2 (mother and father)**
HG01983 Chr 16, Chr 10	1st-order: **0/1**, 2nd-order: **0/1**	1st-order: **0/1**, 2nd-order: **0/1**
HG02024 Chr 2, Chr 16	1st-order: **2/2 (mother and father)**	1st-order: **2/2 (mother and father)**
HG02046 Chr 10, Chr 10	1st-order: **0/1**	1st-order: **1/1 (sibling)**
HG02363 Chr , Chr 2	1st-order: **0/1**	1st-order: **0/1**
HG02372 Chr 19, Chr 18	1st-order: **1/1 (parent or child, unclear directionality)**	1st-order: **1/1 (parent or child, unclear directionality)**
HG02377 Chr 20, Chr 22	2nd-order: **0/1**	2nd-order: **0/1**
HG02381 Chr 14, Chr 18	1st-order: **0/1**	1st-order: **0/1**
HG02387 Chr 16, Chr 14	2nd-order: **0/1**	2nd-order: **0/1**

The analyses were designed to simultaneously explore the impact of several parametric factors in the identification of relatives: (a) n_{iter}, (b) N_{SNP} and (c) chromosome identity. From the results in preceding section, it is likely that changing n_{iter} had minimal impact. At the same time, previous results also indicated that the choice of SNPs and which chromosome they occur on will impact the best-fit trajectories. To parse these two effects, we note that irrespective of the query individual and the chromosomal choice, increasing the SNPs clearly improves the efficiency of the kinship discovery (comparing the 2nd and 3rd columns of Table 2). While for several queries no identification was made for either set of parameters, there is a clear and understandable increase in risk for 1st-order relatives compared to 2nd-order relatives. There was only a single instance in which a 2nd-order relative was discovered for a query individual, and in that instance there were no 1st-order kin in the 1000 Genomes main cohort to confound the identification process.

Therefore, we show that even a modest number of common SNPs have the capacity to reveal genetic relatives of query individuals in a cohort of size ~5,000 haplotypes.

2.6 Polygenic risk score (PRS) analysis

One of our primary goals in the use of the HMM is to show how information can be pooled across multiple trajectories, even in the absence of a direct identification of the query individual. To this end, we probe scenarios where either excessive mutations or the absence of the individual in the reference set may prevent the identification of the true individual, but the set of best-fit diploid mosaic genomes may collectively provide hints about the true individual. Another way of understanding this indirect inference is by considering the use of inferred genomes from a reference database as approximating proxies for the true individual, and the genotype reconstruction as coarse-grained imputation using the imperfect set of SNPs. Accordingly, we also carry out a rough calculation of the linear polygenic risk scores (PRSs) of these individuals for all the phenotypes in the GWAS catalog[31] (version 1.0.2) to see if we could infer certain phenotypes that are outliers for the set of alternate individuals. We carried out this analysis for one of the runs in the previous section, where the query individual was a mosaic of HG00360 (first half) and HG00342 (second half), and the best-fit trajectories were identified using *PLIGHT_Iterative* with $n_{iter} = 20$ and $S_{sg} = 300$. There were 8 parallel trajectories for chromosome 1, 15 for chromosome 2, and 2 for chromosome 21. The PRS calculation was carried out on all these trajectories independently and the PRSs across all trajectories (for each chromosome separately) for each trait were compared with the PRSs for a 50% HG00360 / 50% HG00342 mosaic as the true sample. Four different statistical measures were evaluated as described in the Methods section. We note that the scenario described here is slightly artificial, as some of the best-fit trajectories included the ground-truth individuals as well. Removal of these trajectories was not an option in this case, given the pervasiveness of the true individuals in the best-fit results. However, we discuss the calculations here as an illustrative example of the sort of aggregative attack that may be carried out. The results of the calculation are shown in **Table 3**.

Table 3: *PRS matching scores for the true query genome to the best-fit mosaic trajectories for the simulated mosaic SNPs across the regions of chromosomes ranging from the first observed SNP*

to the last observed SNP. Corresponding matching scores for the true sample to the background genomes are shown in parentheses. Four different cosine similarity scores are chosen with the aim of elucidating potentially subtle differences in the matching depending on the choice of phenotypes or method of averaging: (1) ***ALL*** *= Cosine similarity of the true sample score relative to the mean score, for all traits; (2)* ***> 1 SNP*** *= Cosine similarity of the true sample score relative to the mean score, for all non-zero traits with more than one GWAS SNP; (3)* ***PRS > 2*** *= Cosine similarity of the true sample score relative to the mean score, for all non-zero traits where the absolute value of the Z-score of the true PRS > 2; (4)* ***Compare, then average*** *= Cosine similarity of the true sample score* ***relative to the score of each trajectory, subsequently averaged****, for all traits. Non-zero traits are those for which the true sample has a non-zero PRS.*

Similarity score	**Chromosome 1**	**Chromosome 2**	**Chromosome 21**
ALL: *Best-fit mosaics (background individuals)*	0.9983 (0.9982) for 343 non-zero traits	0.9694 (0.8575) for 225 non-zero traits	0.9213 (0.9478) for 131 non-zero traits
> 1 SNP: *Best-fit mosaics (background individuals)*	0.9288 (0.7637) for 197 traits	0.9111 (0.7824) for 93 traits	0.9517 (0.9526) for 53 traits
PRS > 2: *Best-fit mosaics (the background individuals were used to calculate the Z-scores)*	0.8143 for 12 traits	0.7683 for 14 traits	0.9712 for 17 traits
Compare, then average: *Best-fit mosaics (background individuals)*	0.9978 (0.7113)	0.9276 (0.7811)	0.8961 (0.8402)

We carry out all these tests to determine where the leakage associated with PRSs was maximal. The correlation with the mean value in row (1), while high, was not consistently better in the mosaic haplotypes than the background. Focusing on non-zero traits with more than one GWAS SNP does improve the contrast, as seen in row (2). The results of row (3) indicate that there is no strong connection of the inference process with choosing outlier traits. Finally in row (4), it does seem that comparing each mosaic haplotype individually with the true sample yields good similarity scores relative to the background. Thus, a comparison of all the PRSs for the best-fit mosaic haplotypes for each trait could possibly leak information about the risk of a query individual.

2.7 Inference based on SNPs obtained from coffee cups

Moving away from simulated examples based on the reference database itself, we use a set of SNPs that were obtained from DNA samples acquired from swabs of used coffee cups in ref. [7] (more details in *Methods*). As described in ref. [7], surreptitiously acquired samples such as these pose significant identification risks to individuals. For our purposes, they additionally represent a source of noisy (i.e. error-prone) SNPs that can be used to test the methods herein.

We filter the SNPs obtained from the coffee cups by genotyping quality and read depth ('GQ > 99 & DP > 10' chosen as the vcf parameter filters, applied using *bcftools* version 1.10.2), to increase the confidence in the choice of genotypes. We subsequently select 30 SNPs each from chromosomes 3 (~position 4 Mb to 173 Mb) and 6 (~position 11 Mb to 166 Mb). We compare these against the blood-tissue-derived genotypes as the gold standard, as well as to

the 1000 Genomes phase 3 vcf files. All 30 chromosome 3 SNPs and 29 chromosome 6 SNPs (1 missing) were found in the blood-tissue and 1000 Genomes vcf files.

We checked the 60 SNPs obtained from the coffee cups against the true genotypes of the query individual: 8 genotypes out of 30 for chromosome 3 were incorrect, while 4 genotypes out of 29 were incorrect for chromosome 6; all incorrect cases involved calling homozygous alternate as opposed to the correct heterozygous genotypes. The incorrect calls are likely to arise due to possible contamination of the coffee cups by DNA from other individuals, as well as due to the inherent noisiness of genotypes obtained from the coffee cups. Accordingly, we include non-zero mutation rates in our inference to account for errors or contamination. Also, we note that both the coffee cup genotypes and the blood tissue gold standard genotypes are unphased: we distribute the SNP dosages in the blood tissue sample based on positional occurrence in the called genotype; thus, for a genotype call of "0/1", "0" is assigned to the first haplotype and "1" to the second.

We carry out two analyses. The first involves merging the gold standard SNP set with the 1000 Genomes Phase 3 vcf file, while making sure that only overlapping SNPs were retained. We subsequently determine whether ~30 SNPs each on chromosomes 3 and 6 would be sufficient to identify the true query individual, using both *PLIGHT_InRef* (where a simpler algorithm is run with the recombination rate set to 0) and several runs of *PLIGHT_Iterative*.

The *PLIGHT_InRef* algorithm, with a mutation rate $\lambda = 0.2$, correctly and uniquely identifies the query individual out of the full set of 2,504 genotypes for both query chromosomes separately. We ran then four inference runs using *PLIGHT_Iterative*, where the full set of 5,008 haplotypes was used as the reference database: (a) $S_{sg} = 200, \text{mutation rate } \lambda = 0.1$; (b) $S_{sg} = 200, \text{mutation rate } \lambda = 0.2$; (c) $S_{sg} = 300, \text{mutation rate } \lambda = 0.1$; (d) $S_{sg} = 300, \text{mutation rate } \lambda = 0.2$. The variation of the subgroup size parameter was done to determine how stable the results were to different bootstrap sample sizes. In every single case, we detect one of the two haplotypes of the correct query individual, while the other inferred haplotype is drawn from the 1000 Genomes set. The lack of both query haplotypes showing up in this instance is likely due to a combination of the high error rate in the coffee cup SNP set and the approximation inherent in the iterative algorithm. We explored the reasons for the consistent appearance of the same gold-standard query haplotype in all the searches, and found that since all the errors involved incorrectly calling a genotype of "1" as a "2" , and since all unphased heterozygous genotypes in the blood tissue sample were called as "0/1", the errors in the coffee cup SNPs would be associated with the first haplotype in our arbitrary phasing scheme (see above). Thus, the algorithm finds the second gold-standard query haplotype consistently, as it correctly aligns with the coffee cup SNPs.

The second analysis on the coffee-cup-based SNPs involves running *PLIGHT_Iterative* on the query SNPs to search through the 1000 Genomes Phase 3 reference database without including the true query individual's genome. We ran the same four combinations of parameters as in the preceding analysis. For both chromosomes, there are haplotypes and haplotype combinations that are found robustly in multiple runs, indicating the algorithm is able to find close matches to the query SNPs. In subsequent analyses, we used some of these best-fit trajectories to explore other instances of inference.

We take trajectories from one of the runs (the results with $S_{sg} = 200, \text{mutation rate } \lambda = 0.2$) and reconstruct the mosaic genomes (see *Methods*). We use the overlapping SNPs

between the gold-standard query vcf and the 1000 Genomes phase 3 reference vcf to determine the total fraction of matching genotypes, and the correspondence score C, a weighted fraction that accounts for the rarity of each SNP in the reference database, which would influence the likelihood of observing a match by random chance (see *Methods*). We also calculate background values for 99 randomly chosen individual genomes from the 1000 Genomes dataset, ensuring that there was no overlap with the mosaic genome haplotypes. There are three matching trajectories for chromosome 3 and one trajectory for chromosome 6. The exact-matching-genotype fractions for the trajectories in chromosome 3 are 0.43, 0.43 and 0.43, compared to the background matching fractions ranging between 0.36-0.44 [Welch's two-sample t-test (for a null hypothesis of both distributions being the same) p-value = 2e-10]. For chromosome 6, the matching fraction is 0.40, with a background between 0.34-0.44. The correspondence scores are: chromosome 3, C = 0.32, 0.32, 0.32 [background 0.28-0.33, p-value = 0.0002]; chromosome 6, C = 0.31 [background 0.29-0.33]. Overall, while there is some evidence for correct imputation of the intervening genomic regions between observed SNPs, we sought to confirm the results, since the few samples drawn still fall within the background ranges.

Accordingly, we explore whether the imputation process improves with a larger number of SNPs. We select an additional 60 coffee cup SNPs from chromosomes 3 and 6 for further analysis: for chromosome 3, all 90 are found in the gold-standard blood-tissue genotypes, 89 in 1000 Genomes Phase 3; for chromosome 6, 88 are in the gold-standard blood-tissue genotypes and in 1000 Genomes. We conduct all the above tests for this extended set of ~90 query SNPs. Running *PLIGHT_Iterative* on the in-reference case allows correct identification of both haplotypes of the query individual for chromosome 6, while only one correct haplotype is found for chromosome 3 (we did not run the exact inference, as 30 SNPs was clearly sufficient to find the correct query individual). While some haplotypes occur in both the 30-SNP and 90-SNP runs, the resulting trajectories are different in general (i.e. different number and haplotype composition). We subsequently use one of the trajectories from the runs where the true query individual's genome is not in the reference database (the results with $S_{sg} = 300, \text{mutation rate } \lambda = 0.1$). There are 8 best-fit trajectories for chromosome 3 and 1 for chromosome 6. Comparisons to a background of 98 individuals are as follows: The exact-matching-genotype fractions for the trajectories in chromosome 3 are 0.45, 0.44, 0.44, 0.44, 0.43, 0.43, 0.43 and 0.44, compared to the background matching fractions ranging between 0.36-0.46 [Welch's two-sample t-test p-value = 2e-8]. For chromosome 6, the matching fraction is 0.43, with a background between 0.34-0.43. The correspondence scores are: chromosome 3, C = 0.33, 0.33, 0.33, 0.33, 0.31, 0.31, 0.32 and 0.33 [background 0.28-0.33, p-value = 0.0001]; chromosome 6, C = 0.33 [background 0.29-0.33]. Overall, while the degree of matching improves a little in going from the 30-SNP case to the 90-SNP case, the change is small. We note that the comparisons here reflect the results of matching of genotypes across entire chromosomes. However, we imagine that within any LD blocks associated with the query SNPs we would observe better matching on average than the global results demonstrated here.

We further consider the PRS calculations and comparisons of the previous section applied to both the 30-SNP and 90-SNP trajectories described above. The results are presented in **Table 4**.

Table 4: *PRS matching scores for the true query genome to the best-fit mosaic trajectories for the coffee cup SNPs across the regions of chromosomes ranging from the first observed SNP to the last observed SNP. Corresponding matching scores for the true sample to the background genomes are shown in parentheses. Four different cosine similarity scores are chosen with the aim of elucidating potentially subtle differences in the matching depending on the choice of phenotypes or method of averaging: (1)* ***ALL*** *= Cosine similarity of the true sample score relative to the mean score, for all traits; (2)* ***> 1 SNP*** *= Cosine similarity of the true sample score relative to the mean score, for all non-zero traits with more than one GWAS SNP; (3)* ***PRS > 2*** *= Cosine similarity of the true sample score relative to the mean score, for all non-zero traits where the absolute value of the Z-score of the true PRS > 2; (4)* ***Compare, then average*** *= Cosine similarity of the true sample score* ***relative to the score of each trajectory, subsequently averaged****, for all traits. Non-zero traits are those for which the true sample has a non-zero PRS.*

Cosine similarity metric	Chromosome 3	Chromosome 6
ALL: *Best-fit mosaics (background individuals)*	**30-SNP-case:** 0.9975 (0.9681) for 534 non-zero traits	**30-SNP-case:** 0.8230 (0.9611) for 663 non-zero traits
	90-SNP-case: 0.9954 (0.9840) for 587 non-zero traits	**90-SNP-case:** 0.9997 (0.9766) for 690 non-zero traits
> 1 SNP: *Best-fit mosaics (background individuals)*	**30-SNP-case:** 0.9995 (0.9995) for 294 traits	**30-SNP-case:** -0.013 (0.9971) for 374 traits
	90-SNP-case: 0.9997 (0.9997) for 317 traits	**90-SNP-case:** 0.9992 (0.9982) for 388 traits
PRS > 2: *Best-fit mosaics (the background individuals were used to calculate the Z-scores)*	**30-SNP-case:** 0.5182 for 53 traits	**30-SNP-case:** -0.1138 for 78 traits
	90-SNP-case: 0.5499 for 55 traits	**90-SNP-case:** -0.0925 for 65 traits
Compare, then average: *Best-fit mosaics (background individuals)*	**30-SNP-case:** 0.9975 (0.8693)	**30-SNP-case:** 0.8230 (0.7494)
	90-SNP-case: 0.9881 (0.9168)	**90-SNP-case:** 0.9997 (0.7711)

The performance seems to generally improve slightly in going from the 30-SNP to the 90-SNP case. However, overall, the performance relative to the background hints at only very subtle PRS leakage for the coarse-grained imputation. We reiterated the analysis with a 300-SNP query set as well, and found similar results: the significance of the genotype matching increases for the mosaic trajectories relative to the background, while the PRS analysis did not show any significant enhancement of PRSs inferred from the mosaic trajectories relative to those of the background individuals (results shown in the Supplementary Materials).

In summary, it was extremely easy to identify an individual from amongst 5,008 haplotypes, using just 30 SNPs on a single chromosome, even in the presence of significant errors. Searching against individuals in a reference database that does not include the query individual also yields best-fit trajectories that are robustly found by *PLIGHT_Iterative* across different parameter

settings. Expanding to a larger set of query SNPs indicates that the haplotype composition of the trajectories does change to reflect the new query SNPs, but with some haplotypes stably appearing in both sets of runs. The coarse-grained imputation at the level of 30 and 90 SNPs hints at only subtle differences relative to the background. We imagine that these differences would be amplified in the case of a much larger set of SNPs, and when focusing on local regions of the genome in the vicinity of the query SNPs.

3. Discussion

The analyses herein have reaffirmed the prevailing notion that very few SNPs, even common ones, have the ability to leak identifying information about a query individual. Furthermore, we have shown that this leakage extends beyond the direct discovery of an individual known to be a part of a database, but can also provide piecewise genetic matches within databases, either exactly or approximately, conditional on any chosen recombination rate model. This idea of mosaic genotypic matching naturally includes the ability to identify genetic relatives within databases, and we show that our algorithm has the ability to discover 1st-order relatives such as parents, children and siblings (and to a lesser degree, 2nd-order relatives) in cohorts of size ~5,000 haplotypes with as few as 30 common SNPs on a single chromosome. The upshot of these results is that investigators seeking to release genetic or omics datasets can employ our tool to assess the degree to which their to-be-publicized data could be used to compromise the identity of a study cohort or any related individuals.

It is our contention that this risk even extends beyond the direct identification of genetic identity or relatedness. To see how, we elucidate the capacity to extract further information from our trajectories. The process of identifying all genotypic trajectories that match a sparse set of data can be seen as a form of coarse-grained imputation: given a few SNPs, the genomic segments containing those SNPs are extended based on available reference haplotypes. However, instead of simply identifying one best-fit genotypic extension, we provide a larger list of all genotypic extensions that are consistent with the query SNP set. In other words, we provide a sense of the "entropy" of the query genotypes conditional on a chosen reference set, where the entropy is a measure of the size of the best-fit genotypic state space in different genomic regions. An attacker could use the auxiliary information of individuals pooled across these genomic regions to explore group phenotypic risk at particular loci, for example. We presented the results for one simplified case of such an attack, but a more sophisticated usage of the pooling attack could pick out risks for both Mendelian and complex genetic disorders.

The expansion of genetic reference databases has also highlighted the limitations of the current reliance on single, linear reference genomes. Capturing the full range of genetic variation across a species (including SNPs, indels, structural variants (SVs), tandem repeats, etc.) in a manner that avoids biases associated with the choice of reference, requires reference data structures that are able represent all possible sequence paths through available database genomes. Graph genomes[18,19] have been recently proposed as such frameworks for the generation of more inclusive reference databases, accounting for all know variants within a single traversable data structure. In one example[32], sequences are represented as edges and breakpoints between variants as nodes. The number of branches extending out or into any node depends on the number of variants associated with that genetic locus. A single haplotype is

constructed by starting at one end and traversing (in an acyclic manner) all branches that agree with the variants in the observed haplotype. If we now pose our query-matching problem in the context of graph genomes, we see that: (a) locating the individual query genotypes within the matching graph edges is easy, although we would have to include alternate branches weighted by the probability of de novo mutation or genotyping error; (b) identifying the consistent genotypic trajectories would amount to finding all possible paths that run through the branches determined in (a); and (c) it would be possible to account for linkage disequilibrium and the relative likelihood of each trajectory if the nodes of the graph genome were annotated with recombination rates calculated based on the reference database. In this way, we would be able to carry over the HMM approach into a graph genome context. If information on the number and identity of the reference haplotypes that map to each branch were also available, the pooling inference described above would also be possible.

Finally, we are also exploring the possibilities for using the underlying model for other problems. For example, we are applying a haploid version of the HMM model here to evaluate viral genotypic clustering and recombination patterns in SARS-CoV-2. It has been suggested to us that such models could also be applied to the assignment of single cells to their correct parent sample. Ultimately, the generality and simplicity of the models make them attractive for multiple applications, as is clear from the ubiquity of HMM-based genomic analyses in the literature.

Our intention is to provide a tool to assess the degree of leakage of a given set of SNPs, and to thereby determine the privacy leakage risk associated with a dataset. This intention is, of course, complementary to subsequent sanitization procedures, such as those outlined in our previous publication[7]. We envision many ways of coupling the information from PLIGHT with data sanitization methods such as pBAM[7] generation: for example, starting from a set of SNPs, we can run PLIGHT on several randomly chosen subsets; we can then determine the genotypic entropy (as described above) for particular SNPs across the PLIGHT runs for each random subset; SNPs that consistently have low entropy across the ensemble of runs might be flagged as highly informative SNPs, and would be prioritized for sanitization. Another possibility is to iteratively mask variants and run PLIGHT to determine the change in the degree of matching to database haplotypes. By judiciously applying one of these approaches, a balance can be struck between the utility of the released dataset and the associated privacy risk.

4. Methods

4.1 Li-Stephens model and associated biological parameters

Let $G_q = \{G_{q,l}\}_{l=1}^{L}$ be the genotypes of a query individual q, observed at SNP loci $l = \{1,2,\cdots,L\}$. The probability of observing such an individual given a space of reference haplotypes $H = \{Z_{j,l}\}_{l=1;j=1}^{l=L_{Ref};j=N}$ (L_{Ref} = number of genotyped sites in the reference genomes, N = number of haplotypes in the reference database) is:

$$P(G_q|H) = \sum_{Z_j^{(1)},Z_k^{(2)}} P\left(G_q \middle| Z_j^{(1)}, Z_k^{(2)}\right). P\left(Z_j^{(1)}, Z_k^{(2)} \middle| H\right) \quad (1)$$

where the set of all possible haplotypes at the observed loci on the two chromosomes is given by $Z_j^{(\alpha)} = \left\{Z_{j(l),l}^{(\alpha)}\right\}_{l=1}^{L}$, with $Z_{j(l),l}^{(\alpha=1,2)}$ being the haplotype at position l, and j being the index of the sampled haplotype. The locus position is defined with respect to a linear reference genome, with the particular genome build assumed to be the same between the query and reference database genomes. We treat the haplotype index $j(l)$ as a function of l, as it is possible for the choice of reference haplotype to be different at each locus; that is, in the haplotype matching process, recombination between reference haplotypes may occur from one observed locus to the next (see **Fig. S1**). The second subscript explicitly indicates that, from reference haplotype $j(l)$, we select the genotype at locus l.

As previously described, we define a *trajectory* for a diploid genome query as $\mathcal{T} = \{j(l), k(l)\}_{l=1}^{L}$, which is the sequence of reference haplotype pairs that best match the query SNPs. In the case of a haploid query, the trajectory would be a sequence of single labels at each locus.

The assumption in the current iteration of the algorithm is that the observed genotypes and the reference haplotypes are registered with respect to the same, linear reference genome. This enables a simpler matching of reference haplotypes to observed genotypes. For data structures such as personal genomes and graph genomes additional genotype matching strategies would need to be incorporated, but the conceptual framework of searching through recombining haplotypes would be the same. In general, the set of genotyped sites does not have to perfectly overlap with the set of reference haplotype sites because of rare SNPs in an individual's genotype or differences in genotyping arrays. However, for the purposes of this study we only consider genotyped sites that overlap with those of the reference haplotypes, especially given our interest in determining the identification power of common SNPs. In case of the presence of structural variants overlapping SNP loci, we allow for missing genotypes in any of the reference haplotypes (except for *PLIGHT_Truncated* due to methodological conflicts by including missing genotypes). We thus consider the reference haplotypes as providing the complete search space, especially in light of the constantly growing genetic databases available for comparison. We avoid making explicit assumptions of population membership and statistics for the query individual with the belief that, beyond the implicit assumptions of the chosen reference set, this will enable more unbiased estimates of kinship and genotypic similarity.

The two terms on the right-hand side of Equation (1) are modeled as follows:

1. $P\left(Z_j^{(1)}, Z_k^{(2)} \middle| H\right)$ is the probability of obtaining a given set of haplotype observations at all the query loci[17,33]:

   $$P\left(Z_j^{(1)}, Z_k^{(2)} \middle| H\right) = P\left(Z_{j(1),1}^{(1)}, Z_{k(1),1}^{(2)} \middle| H\right) \prod_{l=1}^{L-1} P\left(\left\{Z_{j(l),l}^{(1)}, Z_{k(l),l}^{(2)}\right\} \rightarrow \left\{Z_{j(l+1),l+1}^{(1)}, Z_{k(l+1),l+1}^{(2)}\right\} \middle| H\right) \quad (2)$$

   All terms are written with the conditional dependence on the set of reference haplotypes, H, made explicit. $P\left(Z_{j(1),1}^{(1)}, Z_{k(1),1}^{(2)} \middle| H\right)$ is the probability of observing a given set of haplotypes at the first query locus. This is often drawn from a uniform distribution across all haplotype pairs, but could be modified if prior knowledge on the membership of the query individual in a particular subpopulation is available. Recombination is incorporated into the analysis in the

expressions for the transition probabilities from one query site to the next, $P\left(\left\{Z^{(1)}_{j(l),l}, Z^{(2)}_{k(l),l}\right\} \rightarrow \left\{Z^{(1)}_{j(l+1),l+1}, Z^{(2)}_{k(l+1),l+1}\right\} \middle| H\right)$. The specific expressions for the transition probabilities are provided in the Supplementary Materials. In our methods, a linear model of recombination (as in the Li-Stephens model) is included as the default. However, we allow for the inclusion of a user-defined model of recombination in its place. For example, if there is a known recombination hotspot between two adjacent query sites, it would alter the probability of transitioning between reference haplotypes in the search space, and thus impact the best-fit haplotypes calculated by the method. The user can explicitly include a vector of recombination values to be used for the $L-1$ intervals between query sites. Additionally, we make an implicit assumption of uniform transition probabilities, where transitions between any pair of haplotypes is the equally likely. If, on the other hand, a model is to be constructed where different subgroups have distinct recombination rates at particular locations and/or are assumed to impacted by assortative mating then transition probabilities would be conditional based on membership in these subgroups. In this iteration of our model, we do not provide a framework of this nature, but such an update would simply require the inclusion of appropriate bias terms conditional on the memberships of the initial and final haplotypes. However, we wish to emphasize that maintaining uniform transition probabilities helps prevent biased interpretations of ethnic group membership and isolation, and allows for the broad intermixing of haplotypes known to have occurred throughout human history[34].

2. $P\left(G_q \middle| Z^{(1)}_j, Z^{(2)}_k\right)$ quantifies the probability of observing the query genotypes given a particular set of underlying haplotypes:
$$P\left(G_q \middle| Z^{(1)}_j, Z^{(2)}_k\right) = \prod_{l=1}^{L} P\left(G_{q,l} \middle| Z^{(1)}_{j(l),l}, Z^{(2)}_{k(l),l}\right) = \prod_{l=1}^{L} P\left(Z^{(1)}_{j(l),l} + Z^{(2)}_{k(l),l} \rightarrow G_{q,l}\right) \qquad (3)$$
with $P\left(Z^{(1)}_{j(l),l} + Z^{(2)}_{k(l),l} \rightarrow G_{q,l}\right)$ determined by the number of sites that require mutation to match the observed genotypes shown in **Table 5**. This probability helps constrain the haplotypes that are possible given the observed genotypes, allowing for the case where mutations or genotyping errors occur (as considered in IMPUTE[15]). We follow the suggestion of the authors of IMPUTE to consider a background rate of base pair mutation θ (the ***thetamutationrate*** parameter in our code), in addition to a mutation rate per haplotype of $\lambda = \frac{\theta}{2(N+\theta)}$ (the ***lambdamutationrate*** parameter in our code) under the assumption of a neutral coalescent tree for N haplotypes[17,33]. However, it is possible for the user to explicitly augment the background rate θ with contributions from genotyping error, or to ignore the mutation rate altogether and set $\theta = \frac{2N\lambda}{1-2\lambda}$ such that λ is equal to the known genotyping error.

Table 5. *Table of conditional probabilities for the observed genotypes based on the sum of two reference haplotypes.*

	$G_{q,l}$		
	0	**1**	**2**
0	$(1-\lambda)^2$	$2\lambda(1-\lambda)$	λ^2

$Z^{(1)}_{j(l),l} + Z^{(2)}_{k(l),l}$				
	1	$\lambda(1-\lambda)$	$\lambda^2 + (1-\lambda)^2$	$\lambda(1-\lambda)$
	2	λ^2	$2\lambda(1-\lambda)$	$(1-\lambda)^2$

In summary, the aim is to figure out the contribution to the total probability of each of the haplotype combinations, by estimating $P\left(G_q \middle| Z_j^{(1)}, Z_k^{(2)}\right).P\left(Z_j^{(1)}, Z_k^{(2)} \middle| H\right)$ for all genotypic trajectories, and to maximize this probability.

4.2 Hidden Markov Model optimization

The problem of identifying the best-fit combination of haplotypes is well-suited to the framework of Hidden Markov models (HMMs) given the traditional treatment of the genome as a linear sequence of base pairs. In this understanding, meiotic recombination between loci does not occur between distant locations of a chromosome (as may occur, hypothetically, due to consistent 3D folding of the chromosomes within the nucleus), but has a certain probability of occurring at every intermediate site between any pair of loci. Usually, the greater the distance between the loci, the higher is the probability that recombination will have occurred in an ancestor of the query genome, though the probability is not necessarily uniform across every site. As seen in the previous section, it then becomes easy to associate HMM emission probabilities at genomic sites with mutation rates and HMM transition probabilities between latent haplotypes with recombination rates. Furthermore, in the above expressions first-order Markovian behavior is assumed, and the observed output genotype is seen to depend only on the underlying haplotypes at that site alone (so-called output independence). This constrains the type of HMMs considered here, but leaves open interesting future applications where such assumptions are relaxed.

The problem of identifying the best trajectory through haplotype space can be carried out using the Viterbi algorithm[27]. This method solves the problem of maximizing the probability of the trajectories through the latent space in time $O((N \times N)^2 L)$, where N is the number of possible haploid states, i.e. the number of reference haplotypes, $N \times N$ is the corresponding number of diploid states, and L is the number of observed loci:

$$Most\ likely\ trajectory = \underset{Z_j^{(1)}, Z_k^{(2)} \in \left\{Z_{j(l),l}^{(1,2)}\right\}_{l=1;j=1}^{l=L;j=N}}{\operatorname{argmax}} P\left(G_q \middle| Z_j^{(1)}, Z_k^{(2)}\right).P\left(Z_j^{(1)}, Z_k^{(2)} \middle| H\right)$$

$$= \underset{Z_j^{(1)}, Z_k^{(2)} \in \left\{Z_{j(l),l}^{(1,2)}\right\}_{l=1;j=1}^{l=L;j=N}}{\operatorname{argmax}} P\left(G_q, Z_j^{(1)}, Z_k^{(2)} \middle| H\right) \quad (4)$$

where expressions for the two probabilities are given in Equations 2 and 3. In the Supplementary Materials we provide further details on the equivalence between the optimization in Equation (4) and the Viterbi algorithm.

A calculation of the argument of maximum probability occurs separately at each pair of reference haplotypes and each locus, resulting in a set of best reference haplotypes at the previous locus. These sets of best reference haplotypes must be stored in a backtrace vector that is accessed at the last query site, when the best reference haplotypes at that site are traced back to the corresponding haplotypes from the previous site, and so on, resulting in a complete set of

best-fit reference haplotypes at all observed query sites. However, the fact that the time complexity scales with the square of the number of states places strain on computational resources due to the need to account for $N \times N$ reference haplotype combinations for diploid genomes. Additionally, the memory required to store the probabilities and the backtrace states scales as $O\big((N \times N)L\big)$.

We accordingly made modifications to the Viterbi algorithm to ameliorate these pressures on computing resources:

1) We utilize the commonly employed logarithmic form of the Viterbi algorithm to prevent the accumulation and subsequent round off of vanishingly small probability products.
2) *Matrix methods.* The probability vectors were encoded as Python *numpy* arrays. Under the assumption of an unbiased transition matrix (as given in Equation S3, with no assumption of subpopulation membership nor biased recombination), each *argmax* calculation in Equation 4 was calculated over an array whose elements were updated using the following simple rules:
   a. Let $\log v_l(j,k)$ be the log-probability vector (Equation S6).
   b. $\log v_l(j,k)$ is a matrix indexed by every pair of reference haplotypes. For memory purposes this matrix was flattened in 1D, keeping only the lower triangle of the matrix.
   c. At each observed genotype locus, initialize $\log v_l(j,k) = \log E_l^{G_l}(j,k)$, the vector of precalculated log-emission-probabilities for each pair of reference haplotypes and the observed genotype.
   d. For $l = 1$, set $\log v_1(j,k) = \log \frac{E_1^{G_1}(j,k)}{N^2}$, with the assumption of equal likelihood of all reference haplotypes at the first observed locus.
   e. Define $T_{on} = \log\left(e^{-\frac{\rho_l}{N}} + \frac{1-e^{-\frac{\rho_l}{N}}}{N}\right)$ and $T_{off} = \log\left(\frac{1-e^{-\frac{\rho_l}{N}}}{N}\right)$.
   f. Define the matrix $\Delta(j,k) = \log v_{l-1}(j,k) + 2T_{off}$
   g. For every pair of reference haplotypes (j,k), update all elements in the same row and column: $\Delta(j,.) \mathrel{+}= T_{on} - T_{off}$ and $\Delta(.,k) \mathrel{+}= T_{on} - T_{off}$.
   h. Find the maximum log-probability $M(j,k) = \max_{i.j} \Delta(i,j)$ and the corresponding arguments $BT(j,k) = \underset{\alpha,\beta}{\operatorname{argmax}}\, \Delta(\alpha,\beta)$, where the last term is the backtrace vector.
   i. Repeat steps (f)-(h) for all haplotype pairs. Note the matrix $\Delta(j,k)$ is reinitialized every time in step (f).
   j. Importantly, we modified the previous step to include all pairs of haplotypes that were within a certain range of the absolute maximum (by default, the cutoff was set as $|M(j,k)| - 0.01 * |M(j,k)|$. We did this as, given the assumed sparsity of the data, it was likely that multiple haplotypes would match exactly or nearly so. Additionally, this looser definition of maximization also compensates any rounding-off errors that may have caused two similar paths to deviate in log-probability. Changing this parameter could also allow the user to discover sub-optimal paths as well, if desired.
   k. Update $\log v_l(j,k) \mathrel{+}= M(j,k)$.
   l. Repeat steps (c)-(k) for every observed genotyped site.

m. When $l = L$, the process is terminated by finding $M = \max_{\alpha,\beta} \log v_L(\alpha, \beta)$ and $BT = All\ (j,k)\ such\ that\ |\log v_L(j,k)|\ \geq\ |M| - 0.01 * |M|$.
n. Using this terminal set of BT, the corresponding backtrace values for each selected pair of reference haplotypes are chosen and traced all the way back to the first observed site. This results in a set of trajectories that may fork and merge, and which form the basis of the subsequent phenotypic inference.

3) *Memory constraints.* The scaling of the matrices with the number of pairs of reference haplotypes puts significant constraints on the reference database that can be considered at any point in the analysis. We discuss two of the modules constructed to address memory issues in later subsections, and focus here on the treatment of the backtrace vector used in all the modules. Instead of explicitly storing the backtrace vector in RAM, we use the Python package *gzip* to write the backtrace vector determined for each observed site directly to a gzipped file. We used the same strategy to read through the gzipped file during the final stage of recovering the best-fit trajectories.

4.3 Parallelization, Truncation and Iterative Schemes

We created 3 separate modules in our package: first, the module *PLIGHT_Exact* encodes the full Li-Stephens HMM as described above; second, we created a module *PLIGHT_Truncated* that truncates the possible trajectory extensions at each observed site to a fraction of the total possible state space to reduce the memory requirements; third, we created a module *PLIGHT_Iterative* that slices up the total reference space into randomly chosen, more tractable subsets, runs *PLIGHT_Exact* on the subsets and pools the best states for a rerun. *PLIGHT_Truncated* was created to consider means of reducing the memory footprint on hard drives, as the backtrace vectors in the HMM model can occupy a significant amount of space. However, the amount of RAM consumed by *PLIGHT_Truncated* remains essentially the same as *PLIGHT_Exact*. The *PLIGHT_Iterative* algorithm was designed to ameliorate both the hard disk and RAM costs of the model, and is thus the preferred method for reference databases beyond ~500 reference haplotypes. In the following, we describe both the parallelization procedure for speed-up, and the additional steps taken in the last two modules, which are, in general, approximations of the full model. However, we show that the approximations often approach the full model when judicious choices of the parameters are made. Additionally, for the first module, we explicitly design a function for the case where the genotyped individual is known to lie within the reference database, as this problem involves a significantly more restricted search (being one-dimensional in the search for genotypes versus the two-dimensional case of pairs of haplotypes), with the recombination rate set to 0. This version of the *PLIGHT_Exact* algorithm is referred to as the *PLIGHT_InRef* algorithm.

4.3.1 Parallelization scheme. In all three modules, we enable the usage of Python's *multiprocessing* scheme to run the calculations over every set of haplotype pairs in parallel. This greatly improves the speed of the analysis, as the matrix manipulations are some of the more time-consuming steps.

4.3.2 Truncation scheme. In the *PLIGHT_Truncated* module, at every observed site we truncate the possible haplotype states by choosing the top T sets of (α, β) pairs. This scheme was inspired by similar techniques employed in the Eagle2 imputation program[16]. The main premise of this approximation is after a certain number of observed loci, only a fraction of a the total number of trajectories will meaningfully contribute to the best-fit states, and allow the retention of only a fraction of the total number of states in memory. We discuss the details of this method and its associated results in the Supplementary Material, as it mainly serves to demonstrate the compressibility of many trajectories.

4.3.3 Iterative scheme. In the *PLIGHT_Iterative* module, the following sequence of steps are run:

a. Set a tractable subgroup size S_{sg} for a single run of the full HMM model
b. Randomly shuffle the identities of the reference haplotypes, and chunk the full reference haplotype into subgroups of size S_{sg}.
c. Run the full HMM model as described in the section *Hidden Markov Model optimization* for each subgroup.
d. Repeat steps (b) and (c) a user-defined n_{iter} times.
e. Pool together all the best-fit haplotypes from each of the subgroups and each of the iterations; best-fit pairs are separated into their constituent haplotypes at this stage.
f. This pooled set is fed back into step (b) and the process is repeated until:
    i. The length of the pooled list is smaller than S_{sg}; or
    ii. The current pooled list is identical to the list derived during the previous pooling step; or
    iii. The current pooled list is larger in size than the previous pooled list.
g. Once the outermost loop over pooling steps is exited, the full module is run on the final best-fit list of haplotypes, with the output being a file with the best-fit pairs of haplotypes chosen from the final pooled list.
h. The parallelization is run over the calculations in step (c) as before.

The trade-off here is, of course, between speed and memory usage. Searching through large databases could take significantly longer even when distributed in this manner, but the memory burden is substantially alleviated.

An issue related to the subdivision process is that the globally optimal combinations of haplotypes, as obtainable in an exact Li-Stephens HMM algorithm, may not co-occur in the subgroups defined in this approximate algorithm. This "mixing problem" is dealt with in two ways: for the inner loop, we run the (subdivision + Exact HMM) process n_{iter} times, with each iteration involving a random subdivision of the input haplotypes and a subsequent running of the HMM for each subdivision in each iteration (total HMMs run: $n_{iter} \times n_{subgroups}$, with n_{groups} = number of subgroups required to include all input haplotypes at this stage); for the outer loop, we obtain the union of the best-fit haplotypes across all $n_{iter} \times n_{subgroups}$ runs and use this set of haplotypes as the input set for the next round of random subdivisions. Note that for the union set in the outer loop, we only include the identities of the haplotypes and none of the haplotype combinations from the inner loop HMM runs. The idea is that if certain haplotypes have significant contributions to any segments of the genotype match, they will be retained through

the different stages and allowed to combine with many other haplotypes. The loops are exited if there is no change in the set of best-fit haplotypes, or if the best-fit list from one iteration of the outer loop is larger than that from the previous iteration (to prevent infinite loops of iteration), or if the set of best-fit haplotypes can be determined by a single run of *PLIGHT_Exact*. If the loop exits with a large number of best-fit haplotypes, we recommend rerunning the code (to allow the randomness to explore different trajectories) or modifying the parameters (such as the recombination rates or n_{iter}).

4.4 Visualization and analysis module

To help visualize the full set of trajectories through the reference haplotype space, we constructed a processing and visualization module, termed *PLIGHT_Vis*. For the trajectory representation step, the most efficient way of representing the multiple possible trajectories with potential overlap was to create a series of multiply linked lists. This involved reading through the output list of best-fit nodes from the HMM modules, with each set of nodes being laid down in the graph in reverse order from the final observed genotype site to the first. At every observed site, the union set of nodes (i.e. with no repeated nodes, even if the same node appears in several parallel trajectories) are laid out and their connectivity with the soon-to-be-added layer at the previous site is established. This continues until the whole graph is constructed. The Python package *matplotlib* is then used to generate a visual representation of the graph with the help of the linked lists. Arrows connect nodes (represented as dots) at one observed site to their corresponding antecedents at the next observed site. The reference sample identities of the two inferred haplotypes at each node are printed above and below the dots.

Additionally, we provide two simple analysis tools as well:

1. For each chromosome, we identify the maximally represented inferred haplotype at each observed site and provide this information at the top of the visualization.
2. For cross-chromosome quantification of the most representative trajectories, we score each trajectory in each chromosome by a weighted sum,
   $Score(tr) = \sum_{h=1}^{H(tr)} \sum_{chr=1}^{N_{chr}} P(h|chr)P(chr)$
   where $H(tr) =$ Number of unique haplotypes within trajectory tr, $P(h|chr) =$ Probability of instances of haplotype i occurring in the predicted trajectory set for chromosome chr, $P(chr)$ = Probability of each chromosome, taken to be a uniform distribution $\Rightarrow P(chr) = \frac{1}{N_{chr}}$. $P(h|chr)$ is calculated as the fraction of trajectories for a given chromosome within which the haplotype h occurs. The score of each trajectory is therefore the sum of the cross-chromosome probability of a haplotype being found in the prediction, taken over all unique haplotypes in that trajectory. The cross-chromosome *consensus* trajectories with the maximum score in each chromosome are stored in a file. This weighting heuristic was chosen with the intention of identifying trajectories in each chromosome that share significant information with trajectories in all other chromosomes.

4.5 Simulation of samples

Our evaluation of the performance of the code was often conducted on simulated data. For the analysis of individuals known to be within the reference database, we selected a single individual at random from the reference database. For the analysis of mosaic genotypes, we selected two or more individuals at random. For all the individuals in a given simulation we created a genotype sample using the following methods. To spread out the selected SNPs across the genome, we only choose SNPs ordered along the chromosome with a probability $p \sim Bernoulli(1,0.003)$. Each SNP thus chosen is randomly mutated at the rate $\lambda = \frac{\theta}{2(N+\theta)}$ at each site (i.e. each SNP has a probability of being mutated to a new value given by Table 5). If the final genotype at the SNP is heterozygous or homozygous in the alternate allele, we select that SNP in our final list. Note that this selection process is meant to mimic the case where only alternative alleles are obtained, and the reference alleles are left out. However, this can also lead to an apparent inflation of the mutation rate, as unmutated, homozygous reference alleles are left out. In the results shown below, this inflation of the apparent mutation rate is implicitly assumed.

For the individual-in-the-reference case, this sampling method directly provides the input dataset. For the mosaic case, we divide each chromosome into a set of segments (for example, two segments for a mosaic of two reference individuals) and assign the genotypes of different individuals, mutated in the above fashion, to each of the segments.

4.6 Mosaic genome reconstruction

To validate the inferred trajectories against the query genomes, we need to reconstruct the mosaic genomes based on the labels inferred. To do so, we essentially grab segments from the reference vcf file corresponding to the reference haplotypes inferred in each trajectory: for $\mathcal{T} = \{j(l), k(l)\}_{l=1}^{L}$, we have $L-1$ segments to reconstruct; for each segment (defined in the following), we extract all the SNP haplotypes for reference haplotypes j and k, combining their values at each position to get the inferred genotype. The genomic segments of identified individuals at each SNP were constructed according to: $Start\ of\ segment\ i = \frac{SNP_{i-1}+SNP_i}{2}; End\ of\ segment\ i = \frac{SNP_i+SNP_{i+1}}{2}$; where $SNP_i = Genomic\ position\ of\ SNP\ i$, and the first segment started at SNP_1 and the last segment ended at SNP_L. We store the results of this reconstruction in a vcf file.

4.7 Polygenic risk score calculation

To quantify some of the risks associated with the pooling of information across multiple genotypic trajectories, we performed approximate calculations of the linear polygenic risk scores (PRSs) based on all the SNP associations in the GWAS catalog[31] (version 1.0.2). We first identified all the individuals in each of the trajectories of a single HMM run. We then constructed the diploid mosaic genome of the query individual based on each HMM trajectory as described in the previous subsection on mosaic genome reconstruction. The resulting genotypes are then used to calculate the PRSs for each phenotype y and each individual n as:

$$PRS(y,n) = \sum_{i=1}^{R(y)} \beta_i x_i^n, \quad (5)$$

where $\beta_i = Signed\ effect\ size\ of\ the\ risk\ allele\ at\ SNP\ i$,

$x_i^n \in \{0,1,2\} = Genotype\ of\ individual\ n\ at\ SNP\ i,$
and $R(y) = Total\ number\ of\ risk\ alleles\ for\ phenotype\ y.$

The PRS is very approximate in the sense that no SNP filtering was conducted beyond those presented in the GWAS catalog, either by p-value or by LD with other associated SNPs. However, the aim here is merely to determine whether there are aggregate properties across the genome that can be inferred using our approach. We calculated the Pearson's correlation between the PRSs of the true samples and the best-fit mosaic genomes within the regions and chromosomes sampled. All traits for which the PRS of the true sample was non-zero ('non-zero traits') were included. To assess whether the PRS correlations between the true individual and inferred mosaic genomes were statistically significant, we sampled a background set of ~100 individuals from the 1000 Genomes dataset that did not occur in any of the test sets we ran, and calculated the PRSs for the non-zero traits.

We ran several statistical tests to assess the correspondence in PRSs between the true samples and the best-fit mosaic genomes, relative to the background scores: (1) evaluated the cosine similarity between the true and mean values of the best-fit scores, and compared to the mean value of the background scores; (2) carried out the same analysis as in (1) but only for those traits that had more than one GWAS SNP in the regions sampled (so as to remove traits that trivially had a single SNP); (3) carried out the same analysis as in (1) but only for those traits for which the true sample had an absolute Z-score > 2 (i.e. traits for which the true sample itself is an outlier relative to the background); and (4) evaluated the cosine similarity between the true sample and each of the best-fit mosaic genomes, and then found the mean of the cosine similarity values (same for the background).

4.8 Analysis of SNPs derived from coffee cups

The query SNPs sets derived from a single genotyped individual were obtained from samples collected as part of our previously published analysis[7]. These include Illumina Whole Genome Sequencing (WGS) results from swabs of coffee cups, as well as gold standard blood tissue samples from the same individuals. In brief, a QIAmp DNA Investigator Kit was used to purify DNA from the coffee cup swabs; the DNA was PCR-amplified using a REPLI-g Single Cell Kit; for the tissue samples, purified PCR-free DNA was directly used; all samples subsequently underwent Illumina WGS. The resulting fastq files were mapped to the hg19 reference genome (b37 assembly) using bwa[35], with the BAM outputs being de-duplicated using Picard tools. Finally, the de-duplicated BAM files were processed utilizing GATK[36,37] to produce variant call sets in vcf format from both genotyping assays. For further details on sample and data processing, see ref. [7]. All genotypes were unphased, and so when the blood tissue sample is used as a reference, we arbitrarily phase the data; for eg., a genotype call of "0/1" gets distributed with "0" to the first haplotype and "1" to the second.

The coffee-cup-based genotypes serve as a noisy, sparse dataset, while the blood-tissue-based results serve as a higher quality baseline for comparison. The vcf coordinates are defined according human genome reference assembly *GRCh37*, using the reference fasta file *human_g1k_v37.fasta.gz*[28], the same as for the 1000 Genomes Phase 3 database. This registers

the vcf files from the coffee cups and from the reference database of the 1000 Genomes project in the same coordinate system, an essential requirement for *PLIGHT* to identify matches between the query and reference datasets.

We carried out two analyses on the coffee-cup-based SNP sets. First, we included the query individual in the 1000 Genomes reference database and assessed whether we could identify the correct haplotype matches for the query SNPs using both the algorithm for cases where the individual is known to be in the database (the *PLIGHT_InRef* algorithm), as well as *PLIGHT_Iterative* algorithm.

Second, we ran *PLIGHT_Iterative* on the same query SNPs, but without including the true query individual in the reference database. The output was a set mosaic trajectories, serving as approximate reconstructions of the query genome. Using the resulting trajectories we assessed the accuracy to which we could impute all SNPs across the full range of observed SNP loci. The accuracy metrics included a straightforward calculation of the fraction of SNPs correctly identified (only exact matches of the genotypes), as well as measure of the degree to which the inferred trajectory matched the query genome, with the contribution from each SNP weighted by a function of the genotype frequency:

$$Correspondence\ score\ C = \sum_{s=1}^{N_{SNP}} \left(1 - p_s(G_s^{\mathcal{T}})\right) . \frac{\left(2 - \left|G_s^{\mathcal{T}} - G_s^{Q}\right|\right)}{2} \qquad (6)$$

The score C finds the total degree of correspondence between the set of query individual genotypes, $\{G_s^Q\}_{s=1}^{N_{SNP}}$, and the set of genotypes for trajectory $\mathcal{T}$, $\{G_s^{\mathcal{T}}\}_{s=1}^{N_{SNP}}$, where N_{SNP} is the total number of overlapping SNPs defined in the vcf files of the reference database and the query individual between the first observed SNP and the last observed SNP. For example, if a set of 50 query SNPs is chosen on chromosome 17 of the query individual's genome, with the first SNP at position 137,000 and the 50th SNP at position 70,200,000. The inferred trajectory will also be defined between the same first and last positions. If the vcf of chromosome 17 for the query individual has 210,000 SNPs between position 137,000 and position 70,200,000 that overlap with the SNP set of the reference database vcf between the same positions, then $N_{SNP} = 210{,}000$. Next, $\left|G_s^{\mathcal{T}} - G_s^{Q}\right|$ quantifies the deviation of the genotype dosage of $\mathcal{T}$ from that of the query Q at SNP position s, and is subtracted from 2 and divided by 2 to set a score scale where 0 corresponds to maximal deviation of $\mathcal{T}$ from Q and 1 corresponds to a perfect match between the two. Finally, $p_s(G_s^{\mathcal{T}})$ is the genotype frequency (as opposed to the allele frequency) of the SNP dosage $G_s^{\mathcal{T}}$, which is the probability that trajectory $\mathcal{T}$ could have a given dosage at random based on population occurrence frequencies. The heuristic $\left(1 - p_s(G_s^{\mathcal{T}})\right)$ is therefore a measure of the non-randomness of the trajectory SNP dosage. In summation, $C = 0$ would occur if no SNPs matched between $\mathcal{T}$ and Q and/or the SNPs occurred in the reference population at 100% frequency, while $C = 1$ would occur if $\mathcal{T}$ and Q agreed at every SNP position and the SNPs were extremely rare (and so the matching of the two is very likely to be a non-random occurrence).

We compared the fraction of correct SNPs and the correspondence scores for our trajectories to the equivalent scores calculated on a set of 99 randomly selected genomes from the 1000 Genomes database to assess if these scores were significant.

4.9 Software availability

We have made our software available for download at https://github.com/gersteinlab/PLIGHT. We provide information on the software requirements, parameter options and examples on how to run the code.

Acknowledgments

The authors would like to thank Hussein Mohsen for his helpful comments on the manuscript.

References

(1) UK Biobank https://www.ukbiobank.ac.uk/.

(2) NIH All of Us Program https://allofus.nih.gov/.

(3) Homer, N.; Szelinger, S.; Redman, M.; Duggan, D.; Tembe, W.; Muehling, J.; Pearson, J. V.; Stephan, D. A.; Nelson, S. F.; Craig, D. W. Resolving Individuals Contributing Trace Amounts of DNA to Highly Complex Mixtures Using High-Density SNP Genotyping Microarrays. *PLoS Genet.* **2008**, *4* (8). https://doi.org/10.1371/journal.pgen.1000167.

(4) Visscher, P. M.; Hill, W. G. The Limits of Individual Identification from Sample Allele Frequencies: Theory and Statistical Analysis. *PLoS Genet.* **2009**, *5* (10), 1–6. https://doi.org/10.1371/journal.pgen.1000628.

(5) Gymrek, M.; McGuire, A. L.; Golan, D.; Halperin, E.; Erlich, Y. Identifying Personal Genomes by Surname Inference. *Science* **2013**, *339* (6117), 321 LP – 324. https://doi.org/10.1126/science.1229566.

(6) Lin, Z.; Owen, A. B.; Altman, R. B. Genomic Research and Human Subject Privacy. *Science* **2004**, *305* (September), 183.

(7) Gürsoy, G.; Emani, P.; Brannon, C. M.; Jolanki, O. A.; Harmanci, A.; Strattan, J. S.; Cherry, J. M.; Miranker, A. D.; Gerstein, M. Data Sanitization to Reduce Private Information Leakage from Functional Genomics. *Cell* **2020**, *183* (4), 905-917.e16. https://doi.org/10.1016/j.cell.2020.09.036.

(8) Harmanci, A.; Gerstein, M. Quantification of Private Information Leakage from Phenotype-Genotype Data: Linking Attacks. *Nat. Methods* **2016**, *13* (3), 251–256. https://doi.org/10.1038/nmeth.3746.

(9) Shringarpure, S. S.; Bustamante, C. D. Privacy Risks from Genomic Data-Sharing Beacons. *Am. J. Hum. Genet.* **2015**, *97* (5), 631–646. https://doi.org/10.1016/j.ajhg.2015.09.010.

(10) Raisaro, J. L.; Tramér, F.; Ji, Z.; Bu, D.; Zhao, Y.; Carey, K.; Lloyd, D.; Sofia, H.; Baker, D.; Flicek, P.; et al. Addressing Beacon Re-Identification Attacks: Quantification and Mitigation of Privacy Risks. *J. Am. Med. Informatics Assoc.* **2017**, *24* (4), 799–805. https://doi.org/10.1093/jamia/ocw167.

(11) Von Thenen, N.; Ayday, E.; Cicek, A. E. Re-Identification of Individuals in Genomic Data-Sharing Beacons via Allele Inference. *Bioinformatics* **2019**, *35* (3), 365–371. https://doi.org/10.1093/bioinformatics/bty643.

(12) Im, H. K.; Gamazon, E. R.; Nicolae, D. L.; Cox, N. J. On Sharing Quantitative Trait GWAS Results in an Era of Multiple-Omics Data and the Limits of Genomic Privacy. *Am. J. Hum. Genet.* **2012**, *90* (4), 591–598. https://doi.org/10.1016/j.ajhg.2012.02.008.

(13) Erlich, Y.; Narayanan, A. Routes for Breaching and Protecting Genetic Privacy. *Nat. Rev.*

Genet. **2014**, *15* (6), 409–421. https://doi.org/10.1038/nrg3723.
(14) Posey, J. E.; O'Donnell-Luria, A. H.; Chong, J. X.; Harel, T.; Jhangiani, S. N.; Coban Akdemir, Z. H.; Buyske, S.; Pehlivan, D.; Carvalho, C. M. B.; Baxter, S.; et al. Insights into Genetics, Human Biology and Disease Gleaned from Family Based Genomic Studies. *Genet. Med.* **2019**, *21* (4), 798–812. https://doi.org/10.1038/s41436-018-0408-7.
(15) Howie, B. N.; Donnelly, P.; Marchini, J. A Flexible and Accurate Genotype Imputation Method for the next Generation of Genome-Wide Association Studies. *PLoS Genet.* **2009**, *5* (6). https://doi.org/10.1371/journal.pgen.1000529.
(16) Loh, P. R.; Danecek, P.; Palamara, P. F.; Fuchsberger, C.; Reshef, Y. A.; Finucane, H. K.; Schoenherr, S.; Forer, L.; McCarthy, S.; Abecasis, G. R.; et al. Reference-Based Phasing Using the Haplotype Reference Consortium Panel. *Nat. Genet.* **2016**, *48* (11), 1443–1448. https://doi.org/10.1038/ng.3679.
(17) Li, N.; Stephens, M. Modeling Linkage Disequilibrium and Identifying Recombination Hotspots Using Single-Nucleotide Polymorphism Data. *Genetics* **2003**, *165*, 2213–2233.
(18) Novak, A. M.; Hickey, G.; Garrison, E.; Blum, S.; Connelly, A.; Dilthey, A.; Eizenga, J.; Elmohamed, M. A. S.; Guthrie, S.; Kahles, A.; et al. Genome Graphs. *bioRxiv* **2017**. https://doi.org/10.1101/101378.
(19) Paten, B.; Novak, A. M.; Eizenga, J. M.; Garrison, E. Genome Graphs and the Evolution of Genome Inference. *Genome Res.* **2017**, *27* (5), 665–676. https://doi.org/10.1101/gr.214155.116.
(20) Pasaniuc, B.; Kennedy, J.; Mandoiu, I. Imputation-Based Local Ancestry Inference in Admixed Populations. In *Proceedings of the 5th International Symposium on Bioinformatics Research and Applications*; ISBRA '09; Springer-Verlag: Berlin, Heidelberg, 2009; pp 221–233. https://doi.org/10.1007/978-3-642-01551-9_22.
(21) Price, A. L.; Tandon, A.; Patterson, N.; Barnes, K. C.; Rafaels, N.; Ruczinski, I.; Beaty, T. H.; Mathias, R.; Reich, D.; Myers, S. Sensitive Detection of Chromosomal Segments of Distinct Ancestry in Admixed Populations. *PLoS Genet.* **2009**, *5* (6). https://doi.org/10.1371/journal.pgen.1000519.
(22) Baran, Y.; Pasaniuc, B.; Sankararaman, S.; Torgerson, D. G.; Gignoux, C.; Eng, C.; Rodriguez-Cintron, W.; Chapela, R.; Ford, J. G.; Avila, P. C.; et al. Fast and Accurate Inference of Local Ancestry in Latino Populations. *Bioinformatics* **2012**, *28* (10), 1359–1367. https://doi.org/10.1093/bioinformatics/bts144.
(23) Bercovici, S.; Meek, C.; Wexler, Y.; Geiger, D. Estimating Genome-Wide IBD Sharing from SNP Data via an Efficient Hidden Markov Model of LD with Application to Gene Mapping. *Bioinformatics* **2010**, *26* (12), 175–182. https://doi.org/10.1093/bioinformatics/btq204.
(24) Durbin, R. Efficient Haplotype Matching and Storage Using the Positional Burrows-Wheeler Transform (PBWT). *Bioinformatics* **2014**, *30* (9), 1266–1272. https://doi.org/10.1093/bioinformatics/btu014.
(25) Lunter, G. Haplotype Matching in Large Cohorts Using the Li and Stephens Model. *Bioinformatics* **2019**, *35* (5), 798–806. https://doi.org/10.1093/bioinformatics/bty735.
(26) Auton, A.; Abecasis, G. R.; Altshuler, D. M.; Durbin, R. M.; Bentley, D. R.; Chakravarti, A.; Clark, A. G.; Donnelly, P.; Eichler, E. E.; Flicek, P.; et al. A Global Reference for Human Genetic Variation. *Nature* **2015**, *526* (7571), 68–74. https://doi.org/10.1038/nature15393.

(27) Viterbi, A. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Trans. Inf. Theory* **1967**, *13* (2), 260–269. https://doi.org/10.1109/TIT.1967.1054010.
(28) 1000 Genomes Consortium, ftp site: http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/.
(29) 1000 Genomes Phase 3 Related Samples Panel ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/release/20130502/supporting/related_samples_vcf/related_samples_panel.20140910.ALL.panel.
(30) 1000 Genomes Phase 3 related individuals pedigree file ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/working/20130606_sample_info/20130606_g1k.ped.
(31) Buniello, A.; Macarthur, J. A. L.; Cerezo, M.; Harris, L. W.; Hayhurst, J.; Malangone, C.; McMahon, A.; Morales, J.; Mountjoy, E.; Sollis, E.; et al. The NHGRI-EBI GWAS Catalog of Published Genome-Wide Association Studies, Targeted Arrays and Summary Statistics 2019. *Nucleic Acids Res.* **2019**, *47* (D1), D1005–D1012. https://doi.org/10.1093/nar/gky1120.
(32) Rakocevic, G.; Semenyuk, V.; Lee, W.-P.; Spencer, J.; Browning, J.; Johnson, I. J.; Arsenijevic, V.; Nadj, J.; Ghose, K.; Suciu, M. C.; et al. Fast and Accurate Genomic Analyses Using Genome Graphs. *Nat. Genet.* **2019**, *51* (2), 354–362. https://doi.org/10.1038/s41588-018-0316-4.
(33) Marchini, J.; Howie, B.; Myers, S.; McVean, G.; Donnelly, P. A New Multipoint Method for Genome-Wide Association Studies by Imputation of Genotypes. *Nat. Genet.* **2007**, *39* (7), 906–913. https://doi.org/10.1038/ng2088.
(34) Narasimhan, V. M.; Patterson, N.; Moorjani, P.; Rohland, N.; Bernardos, R.; Mallick, S.; Lazaridis, I.; Nakatsuka, N.; Olalde, I.; Lipson, M.; et al. The Formation of Human Populations in South and Central Asia. *Science* **2019**, *365* (6457). https://doi.org/10.1126/science.aat7487.
(35) Li, H.; Durbin, R. Fast and Accurate Short Read Alignment with Burrows–Wheeler Transform. *Bioinformatics* **2009**, *25* (14), 1754–1760. https://doi.org/10.1093/bioinformatics/btp324.
(36) DePristo, M. A.; Banks, E.; Poplin, R.; Garimella, K. V; Maguire, J. R.; Hartl, C.; Philippakis, A. A.; del Angel, G.; Rivas, M. A.; Hanna, M.; et al. A Framework for Variation Discovery and Genotyping Using Next-Generation DNA Sequencing Data. *Nat. Genet.* **2011**, *43* (5), 491–498. https://doi.org/10.1038/ng.806.
(37) Van der Auwera, G. A.; Carneiro, M. O.; Hartl, C.; Poplin, R.; Del Angel, G.; Levy-Moonshine, A.; Jordan, T.; Shakir, K.; Roazen, D.; Thibault, J.; et al. From FastQ Data to High Confidence Variant Calls: The Genome Analysis Toolkit Best Practices Pipeline. *Curr. Protoc. Bioinforma.* **2013**, *43* (1110), 11.10.1-11.10.33. https://doi.org/10.1002/0471250953.bi1110s43.