exML: an Explainable Maximum Likelihood tool for proportion estimation in DNA data

Amit Bergman
Tel Aviv University

Viviane Slon
Tel Aviv University

Daniel Deutch
Tel Aviv University

ABSTRACT

We describe *exML*, an explainable analysis tool for DNA classification. The underlying problem is to estimate the proportion of different species that contribute to a given genomic dataset. While there are available solutions to this proportion estimation problem, they lack explainability. To this end, our proposed solution incorporates novel explanation methods that accompany a Maximum Likelihood analysis in this context. The explanations the tool generates include a variant of the commonly used attribution method that assigns influence scores to individual datums based on Shapley values, and a notion of counterfactuals that identify subsets of the input data that together are most influential. As a use case for this paper, we focus on the analysis of ancient DNA samples collected from archaeological sites, and show how the explanations generated by *exML* provide insights on otherwise ambiguous classification results.

1 INTRODUCTION

Consider the following problem: given a target dataset D, a set of classes C, and a reference set R in which each datum is associated with a single class $c \in C$, estimate the proportion of each class in D. This problem is common in the field of genomics, where D is typically a dataset of DNA sequences we are interested in the contribution of different groups to it (see [4], [6], [7]). As our use case, we will focus on this classification problem in the context of *ancient DNA of hominins*. Sequencing the DNA of ancient individuals can be used as a virtual time machine to explore our evolutionary history and that of our closest extinct relatives, the Neanderthals and the Denisovans. To date, over 6,000 ancient human genomes have been sequenced from fossilized skeletal remains [2], to assemble a database of labeled reference sequences. The retrieval of DNA from sediments deposited at archaeological sites has recently opened new avenues of research (e.g. [5]), allowing to generate ancient genetic data even in the absence of skeletal remains, by extracting and sequencing the DNA molecules that were deposited in the sediment for up to tens of thousands of years. A key property in our context is that a single DNA data sample can have DNA sequences *originate from multiple individuals*, and our goal is to estimate the proportion of each of the known hominin species (Neanderthals, Homo Sapiens and Denisovans) in the sample.

While the proportion estimation problem is commonly posed and studied (see [6], [5], [7]), note that there is an inherent ambiguity associated with its statement. For example, an estimation that the proportion in a certain dataset is 50% Neanderthals and 50% Homo sapiens, can be interpreted in different ways: **1)** Neanderthals and Homo sapiens lived jointly in a single location; **2)** All DNA reads are ambiguous, leading to the inability of the model to determine the species of origin; **3)** the data has originated from another species that is equally distant to both Neanderthals and Homo Sapiens or **4)** 10% of the data is from Homo Sapiens, 10% from Neanderthals, and the rest is unidentified. Indeed, state-of-the-art solutions do not allow to easily differentiate between the three (see [5], [8]).

To this end, we have devised an explanation-ready algorithm for proportion estimation. We start with a standard Maximum Likelihood Estimator, looking for a proportions vector $\alpha \in [0,1]^{|S|}$ that maximizes a likelihood function of the form $L(D,\alpha) := \prod_{d \in D} \sum_{s \in S} \alpha_s \cdot A_{s,d}$ where α_s corresponds to the proportion of species s. The parameters $A_{s,d}$ are computed based on the reference set R. In the context of ancient DNA analysis, this simple method already achieves similar to or superior accuracy when compared to the state-of-the-art [5]; yet, as illustrated above, its results are ambiguous. We thus introduce explainability capabilities of multiple flavors, which may be categorized based on the following two axes. The first is explanation type, namely (a) explanations that are based on quantifying the contribution of individual datums (e.g., DNA reads) or (b) explanations that are based on counterfactuals, which are dataset modifications that affect the classification in certain ways. The second axis is whether we explain a classification via data points from the given dataset D or via the effect of references data on parameters of the model. The latter is in a sense a “second order" explanation, as it does not directly explain the end result but rather explain the computation of $A_{s,d}$ based on R.

In designing the explanation, we are inspired by existing methods, but adapt them to our settings. First, we adapt existing definitions. For instance, in the context of attribution of individual contributions, a standard solution [3] is based on Shapley values, which intuitively quantify the marginal contribution of individual players to a cooperative game result, when these players are added to players subsets. For proportion estimation, the influence of adding a single player to a group decreases as the group size increases, an issue which we address by adding a scaling parameter to the Shapley formula. Another adaptation we applied is to adjust the explanation computation to our scenario: Both attribution-based and Counterfactual-based explanations involve executing the explained algorithm on multiple subsets of the data, which is typically inefficient. In our settings however, we were able to optimize this computation by leveraging the structure of the Maximum Likelihood method. We observed that the $A_{s,d}$ values can be calculated once during the preprocessing step, and to be reused in following executions.

In this paper, we first describe the maximum likelihood algorithm we applied on the proportion estimation problem, and show the accuracy it attains on several datasets consisting of simulated and real-world ancient DNA reads of hominins (Homo sapiens, Neanderthals, or Denisovan). We then turn demonstrate the ambiguity of the output of the model, and show how the versatile explanations exML generates complement each other and allow users to resolve the ambiguity and obtain a holistic view of the analysis results and gain further insights on the data and model.

2 ALGORITHM FOR PROPORTION ESTIMATION

We first describe the general settings, and a maximum likelihood algorithm for the problem. In the problem of *Proportion Estimation*, we are given a dataset D (e.g. a set of unlabeled DNA reads) and a labeled reference set R (e.g. a set of labeled DNA reads). Labels are elements of a set S (e.g. a set of biological species). The goal is to output a distribution vector $v : S \mapsto [0, 1]$ capturing the estimated proportion of every label in D. In the case of ancient DNA, on which we will exemplify the algorithm we further have a substitution matrix M s.t $M_{i,j,k}$ is the likelihood of observing the letter j in index k of a DNA read, given that the DNA read is originated from an organism that is closely related to an organism that has the letter i in index k. This matrix is used to estimate the likelihood that an observed DNA read d is related to a reference r, and intuitively addresses the facts that genomes of closely related organisms are very similar, but not identical, and the DNA corruption that can affect the input dataset D. Table 1 summarizes the main notations.

Algorithm 1 is a Maximum Likelihood estimator for the task of proportion estimation. It consists of two steps (captured by methods invocation in lines 1 and 2): `PreProcessing`, namely estimating the likelihood of the observed data as a function of the model parameters, and `MaximizeLikelihood`, namely finding parameters to optimize this function. The `PreProcessing` is calculating the $A_{s,d}$ values for the likelihood function. To do that, it first aligns every DNA read d to every labeled DNA reference r, to find r_d - sub sequence of r that is most similar to d (line 6). Then, using r_d and M, it calculates $p[r_d]$ - the estimated likelihood to observe the DNA read d in a genome of an organism that is closely related to an organism with genome r (line 7). In lines 9-11, for each species $s \in S$ and read $d \in D$, it calculates $A_{s,d}$ - the average of the $P[r_d]$ values for references labeled with s. The `PreProcessing` method returns the $A_{s,d}$ values, and they are sent as parameters to `MaximizeLikelihood`, that uses them to return the proportions that maximizes the likelihood of the observed data D (lines 15-16), which is the output of the algorithm. Note that the algorithm is highly parallelizable, as the entire loop in lines 4-11 can be executed independently for every datum d. For the optimization in line 16 - we support two approaches. The first is heuristic but fast: it is a variant of gradient descent, where we initialize multiple arbitrary starting points, and advance in the direction that is most increasing the value of L. In this method, in every iteration we are finding the index that increasing its proportion is most improving the value of the likelihood function, and make small change to that direction. For the case where the label set S is small, we also support a brute-force solution: We divide the range $[0, 1]$ to B equal size bins (B is configurable), and evaluate L for all $B^{(|S|-1)}$ options for a representative value of each bin in each index in α (with the last value complementing the rest to 1, hence the exponent).

We implemented Algorithm 1 and tested its performance on 4 datasets of ancient human DNA reads collected from sediments, and on simulated datasets of ancient hominin DNA, in which the correct species proportion is known (data is simulated based on real genomes of hominins using the ancient DNA simulator tool published in [5]). For each dataset, we executed Algorithm 1 and calculated the KL divergence [1] between the correct proportions and the output of the algorithm. Figure 1 shows the results and compares our tool to Kallisto, a method based on k-mers and pseudo alignment that is the state-of-the-art method in classification of ancient DNA (see [5]). Note that when $|D|$ is small (as usually is the case in ancient DNA data), this simple algorithm outperforms state-of-the-art technique, and in larger $|D|$'s it is at least as accurate (though it runs slower, due to the computationally expensive alignment step).

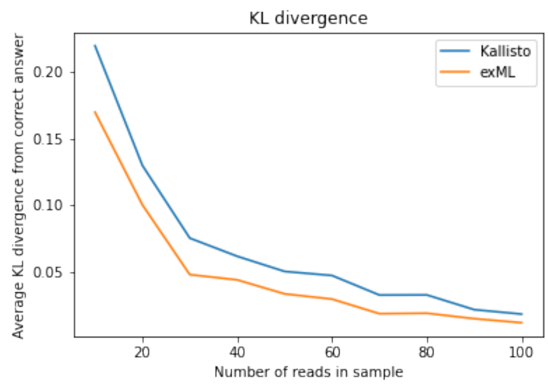


Figure 1: Comparison of exML and kallisto method, we generated 500 samples for each sample size, and calculated the average KL divergence between the output of the different algorithms and the correct answer (smaller KL divergence from correct answer means better accuracy).

Example 2.1. We generated 2 datasets on which we will exemplify the output of Algorithm 1: D_1 contains 20 DNA reads from Homo sapiens, 20 DNA reads from Denisovans, and 40 DNA reads from regions in the genome that are conserved across all hominin genomes (*i.e.,* non informative regions). D_2 contains 40 DNA reads from Homo sapiens and 40 DNA reads from Denisovans. The average length of a DNA read in both datasets is 75. When running Algorithm 1 on these datasets, using the label set $\{Homosapiens, Neanderthal, Denisovan\}$, and a database of labeled genomes of hominin species as references set, the output was $v(D_1) = v(D_2) = (0.5, 0, 0.5)$. This output means that for both datasets, the algorithm assigns 50% proportion to Homo sapiens, 0 to Neanderthal, and 50% proportion to Denisovans.

3 EXPLANATIONS

In the explanations our system produces, we employ state-of-the-art techniques for explanations, and adapt them to the particular settings in hand. As mentioned in the introduction, the system provides both attribution level explanations, and counterfactuals. We next detail detail both types, describe the algorithms our system utilizes to generate them, and show examples to their usefulness on real world scenarios in analysis of ancient DNA.

3.1 Attribution based explanations

3.1.1 Data-level attribution via Shapley values.

DNA read	A word $\in \{A, C, T, G\}^*$
D	Input dataset of DNA reads
S	Set of possible labels (species)
R	Set of DNA references (full genomes)
$Y \in S^{\vert R\vert}$	Labels for references, $Y_i \in S$ is the label of R_i
$\alpha^* \in [0,1]^{\vert S\vert}$	Algorithm's output - distribution vector ($\sum_i \alpha_i^* = 1$ and $\alpha_i^* \geq 0$)
R_s	$\{R_i \in R \vert Y_i = s\}$
M	Substitution matrix
$v(D) \in [0.1]^{\vert S\vert}$	Result of running Algorithm 1 on dataset D.

Table 1: Table of notations

Algorithm 1: Maximum likelihood algorithm for proportion estimation in genomic dataset

Data: input dataset D, substitution matrix M, label set S, labeled set of references R
Result: Distribution α^* - species proportion estimation

1 A = `PreProcessing`(D, R, M)
2 **return** `MaximizeLikelihood`(A, D)
3 **Procedure** `PreProcessing`(D, R, M)
4   **foreach** $d \in D$ **do**
5   **foreach** $r \in R$ **do**
6   Align d to r to get r_d:=sub sequence of r that is most similar to d
7       $P[r_d] = \prod_{t=1}^{|d|} M_{(r_d[t], d[t], t)}$
8   **end**
9   **foreach** $s \in S$ **do**
10       $A_{s,d} = \frac{\sum_{r \in R_s} P[r_d]}{|R_s|}$
11   **end**
12   **end**
13   **return** all $A_{s,d}$ values
14 **Procedure** `MaximizeLikelihood`(A,D)
15   $L(D; \alpha) := \prod_{d \in D} \sum_{s \in S} \alpha_s * A_{s,d}$
16   **return** $\alpha^* := argmax_\alpha(L(D; \alpha))$ with the constraints $\sum_i \alpha_i = 1$ and $\alpha_i \geq 0 \ \ \forall i$.

Definition 3.1. Given a dataset D, a set of labels S, and a proportion estimation algorithm A outputting $v : S \mapsto [0, 1]$, an *attribution-based data explanation* is a function $attr : D \mapsto \mathcal{R}^{|S|}$ where $attr(d)[i]$ quantifies the contribution of d to the proportion A assigns to label $i \in S$.

Example 3.2. As an example for definition 3.1, A can be Algorithm 1 described above, that gets as input a dataset D of DNA reads, S is the set $\{HomoSapiens, Neanderthal, Denisovan\}$ and the data-level attribution explanation of a single DNA read d is a vector $v^d \in R^3$ where v_0 quantifies how d changes the proportion Algorithm 1 assigns to Homo sapiens. For instance, if v_0^d is a positive number, it means that d makes the algorithm tend to increase the estimated proportion of Homo Sapiens.

To generate attribution based data-level explanations, we apply the notion of Shapley values, with modifications to our settings. In the context of cooperative games, Shapley values estimate the contribution of a single player to the result of the game. The Shapley value of a player i is defined as (v is the value of a game and N is the number of players):

$$\frac{1}{N!} \sum_{G \subseteq [N] \setminus \{i\}} [v(G \cup i) - v(G)] * |G|! * (N - |G| - 1)! \qquad (1)$$

To use Shapley values in our context, we interpret every datum $d \in D$ as a player of a game, and $v(G)$ as the result vector of running Algorithm 1 on a subset $G \subseteq D$. Then, the output of the formula above is a vector that its j'th entry encodes d's contribution to the estimated proportion of species j. However, to apply this idea for generating data-level explanations in proportion estimation, two main adaptations are needed:

Execution on Samples. Shapley value formula requires executing an algorithm on a subset of its input set. When these values are used in Machine Learning models, usually this requires the output of a model on a subset of its features, to quantify the contribution of the different features to the result of the model. In general Machine Learning models, it is not feasible to calculate the output of the model only on part of the features, as this requires to train the entire model from scratch (a problem that SHAP solves by using the background distribution to sample values for features it integrates out [3]). However, in our Maximum Likelihood estimator, we actually can run the model on a subset of the input data D, by simply rewrite the formula in line 15 of Algorithm 1 so that we plug-in a subset G instead of the entire dataset D, and thus calculate what would be the output of the algorithm on the subset G. Furthermore, when we consider multiple subsets G, we only need to execute the expensive `PreProcessing` step once, and re-use the $A_{s,d}$ values it computes in multiple invocations of `MaximizeLikelihood` (A, G). This optimization allows our system to evaluate multiple outputs of the algorithm on different subsets of the input dataset, and to use that information to apply Shapley values formula and attribute the output of the model to the different datums in the dataset.

Scaling Shapley values. Since Algorithm 1 outputs proportions, and not absolute values, the Shapley formula needs to be refined, as illustrated in the following example:

Example 3.3. Consider a sample G for which the ground truth proportion is $(\frac{a}{|G|}, \frac{b}{|G|}, \frac{c}{|G|})$. Further consider a DNA read $i \notin G$ generated from a Homo sapiens. The ground truth for $G \cup \{i\}$ is $(\frac{a+1}{|G|+1}, \frac{b}{|G|+1}, \frac{c}{|G|+1})$.
Thus $v(G) - v(G \cup \{i\}) = (\frac{|G|-a}{|G|^2+|G|}, \frac{-b}{|G|^2+|G|}, \frac{-c}{|G|^2+|G|})$ which is equal to $(\frac{b+c}{|G|^2+|G|}, \frac{-b}{|G|^2+|G|}, \frac{-c}{|G|^2+|G|})$.
Note that $||v(G) - v(G \cup \{i\})||$ decreases approximately linearly as $|G|$ grows (see Figure 2 for empirical example that shows that the bigger the sample, the smaller the influence of adding a single read on the output).

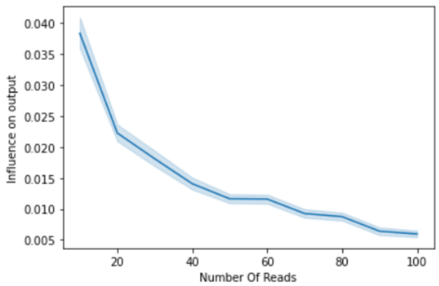


Figure 2: X axis is number of reads in a sample, Y axis is the average magnitude of change that is caused by adding a single read to a sample. The plot shows empirically that the more reads are in a sample, the smaller the influence of adding a single read is.

Thus, before applying Shapley values in our context, we are required to address the fact that the magnitude of the influence of adding a single data point to a set depends on the size of that set, otherwise the explanations are dominated by the smaller samples. We therefore added a scaling parameter to the original Shapley value formula, that takes into account the size of the subset. Specifically, to generate a data-level attribution explanation for datum i, exML samples M subsets of D, and outputs the following vector as explanation (This can be calculated efficiently, as described above, since the preprocessing step only needs to be executed once):

$$\frac{1}{M}\sum_{G\in Samples}[v(G\cup i)-v(G)] * |\boldsymbol{G}|. \tag{2}$$

Given a datum $d \in D$, the output of this process is an explanation vector v^d that corresponds to definition 3.1, and this is the vector that our system generates as an attribution based data-level explanation.

Example 3.4 (Scaled Shapley values). Figure 3 shows data-level explanations generated by exML using the scaled Shapley values on D_1 and D_2 (as described in example 2.1). As stated above, the output of Algorithm 1 on both datasets is approximately the same, so using merely the output of Algorithm 1, one may not distinguish between them. However, as Figure 3 demonstrates, their different data-level explanations provide insights that direct the user to the correct interpretation of the results. These explanations reflect that in D_1, only the last 40 reads had substantial influence on the output, whereas in D_2, all reads were influential. This implies that the system has more evidence to its output on D_2. In addition, the non-influential reads in D_1 might also be attributed to being originated from a genomically distant organism that is not in the reference set and hence do not influence the output proportion of neither Homo sapiens nor Denisovans.

3.1.2 Reference-level attribution. In addition to explaining datums from the dataset, our solution supports attribution-based explanations on the reference level, aiming to quantify for every reference $r \in R$ its influence on the output of the algorithm.

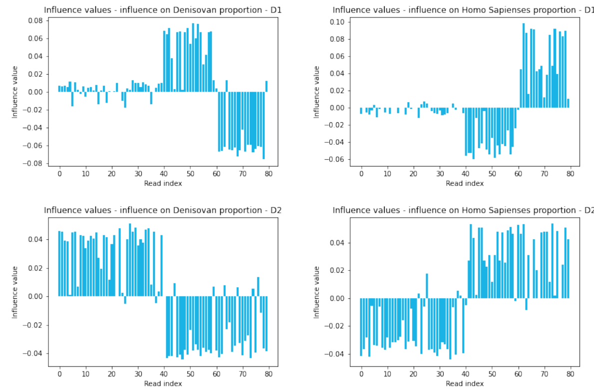


Figure 3: Data-level explanations on D_1 (top), and D_2 (bottom). x axis is the read index, and y axis is the explanation value. Right plots show the explanation values of Homo Sapiens proportions, and Left plots show explanation values on Denisovan proportions.

Definition 3.5. Given a dataset D, a set of labels S, a set of references R and a proportion estimation algorithm A outputting $v : S \mapsto [0, 1]$, an *attribution-based reference explanation* is a function $attr : R \mapsto \mathcal{R}^{|S|}$, where $attr(r)[i]$ quantifies the influence of r on the proportion A assigns to label i.

To calculate reference-level attribution, our system uses Algorithm 2, a sampling-based algorithm to calculate explanation vectors for references. For the context of Algorithm 2, we define $A^i_{s,d}$ as the value that Algorithm 1 would calculate as $A_{s,d}$, if the references set would not include r_i.

In line 1, we run the `PreProcessing` method of Algorithm 1, to get the initial $A_{s,d}$ values. In line 2 we loop over the references and for $A^i_{s,d}$ values where s is the label of the current reference r_i, we set $A^i_{s,d}$ value to be the average that would have been obtained if r_i was not in the references set (line 3); otherwise, we set $A^i_{s,d} = A_{s,d}$ (line 4) (the value of $A_{s,d}$ is not changed by removing a reference that its label is not s). In lines 5–9 we go over the samples, and calculate the change that is caused to the output vector of Algorithm 1 by removing r_i (comparing the maximum likelihood with the original values, to the maximum likelihood with the values $A^i_{s,d}$. In line 10 we average this influence value over all samples, to get an estimation of how removing reference i influences the output of Algorithm 1. In line 12, we return V, which is a list of all V_i values. V_i is the vector exML outputs as an explanation to reference i, according to Definition 3.5.

Example 3.6. Figure 4 shows the reference explanations generated by our system on two datasets, D_5 (top figure), and D_6 (bottom figure). Each dataset contains 50 reads generated from Neanderthals, 50 from Homo sapiens, and 50 from Denisovans. However, D_5's Neanderthals are from Neanderthal KX198082, and D_6's Neanderthals are from Neanderthal Altai KC879692. On both datasets, Algorithm 1 outputs the same estimation $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, but the reference explanations can help identify the specific sub group of Neanderthals

Algorithm 2: Calculating reference-level explanations

Data: References set R, input dataset D, Samples K, such that every $K_j \in K$ is a subset of D

Result: V, s.t V_i is the reference-level explanation of R_i

Run preProcessing() step of algorithm 1 to get $A_{s,d}$ and $p[i,d]$ values

foreach $r_i \in R$ **do**

  For each $A_{s,d}$ such that $s = Y_i$, $A^i_{s,d} := \frac{A_{s,d} * |R_s| - p[r_i, d]}{|R_s| - 1}$

  For each $A_{s,d}$ such that $s \neq Y_i$, $A^i_{s,d} := A_{s,d}$

  foreach $k_j \in K$ **do**

    $L^i(k_j; \alpha) := \prod_{d \in k_j} \sum_{s \in S} \alpha_s * A^i_{s,d}$

    $L(k_j; \alpha) := \prod_{d \in k_j} \sum_{s \in S} \alpha_s * A_{s,d}$

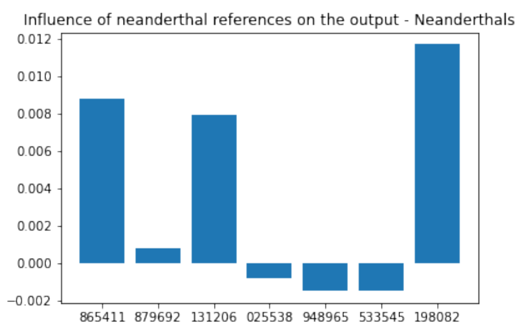
    $\delta_{i,j} \leftarrow argmax_\alpha(L^i(k_j; \alpha)) - argmax_\alpha(L(k_j; \alpha))$

  end

  $V_i = \frac{\sum_{j \in |K|} \delta_{i,j}}{|K|}$

end

return V



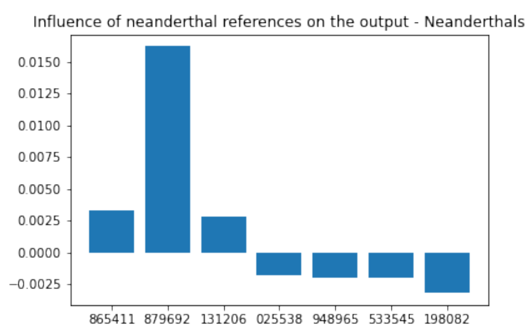


Figure 4: Reference explanations on D_5 (top) and D_6 (bottom). X axis is the reference index, and y axis is the influence of that reference on the proportion of Neanderthals in the output of algorithm 1.

that contributed to the dataset, as in both explanations, the reference that has the highest positive influence on the Neanderthal proportion is the one from which the data was actually generated.

3.2 Counterfactuals

Given a model M and an input x such that $M(x) = l$, a *counterfactual explanation* is a modified instance y, based on a typically small perturbation, for which $M(y) \neq l$. Intuitively, modifications to x that lead to a different label are indicative of the features that were responsible for the label l being chosen to begin with. We now define and elaborate on the 3 types of counterfactuals explanations our system produces. We start by defining few useful notations in the settings of Maximum Likelihood, and then use these notations to define the counterfactual notions.

Definition 3.7. Given a dataset D, a set of labels S and a likelihood function $L(\alpha, D)$, let $ML(D, L) := argmax_\alpha L(\alpha, D)$ be the vector α that maximizes L

For example, in our settings, if D is a set of DNA reads, α is a proportion vector, S is the set of known hominin species, and L is the likelihood function of Algorithm 1 on the dataset D, $ML(D, L)$ is the vector that maximizes the likelihood term $L(\alpha, D)$, for example, it can be the vector (0,0,1). This is typically the output of the maximum likelihood algorithm.

Definition 3.8. Let $H(D, L)$ be the index of the maximal value in $ML(D, L)$ (in case of multiple indexes that are maximizing, $H(D, L)$ can be either one of them)

Intuitively $H(D, L)$ is the label with the highest proportion assigned to it. In our example, if $ML(D, L) = (0, 0, 1)$, so $H(D, L) = 2$ as this is the index that maximizes $ML(D, L)$. In our example this corresponds to saying that the algorithm estimates that the Denisovans species is the most abundant in D.

Definition 3.9. Let *dominance-changing subset* be a subset $D^{'} \subseteq D$ such that $H(D, L) \neq H(D \backslash D', L)$.

This is a subset that removing it from the input dataset causes the vector that maximizes L to have a different maximizing index. In our example, if there are 2 DNA reads $D_1, D_2 \in D$ that removing them from D will cause the output of Algorithm 1 to be maximal when $\alpha = (1, 0, 0)$, then the set $\{D_1, D_2\}$ is a changing dominance subset with respect to L and D.

3.2.1 *counterfactual 1: minimum dominance changing subset.*

Definition 3.10. Define $CF1(L, D)$ as a subset $D^{'} \subseteq D$ that is a dominance-changing subset, and is also minimum subset with that property, meaning that for every subset $T \subseteq D$ that is dominance-changing with respect to L, we have that $|D^{'}| \leq |T|$.

In the example of DNA proportion estimation, this counter factual is an interesting research insight, as it points out subsets of the data that make the algorithm substantially change its decision, which provides information both on these subsets (maybe they originate from different species then the others) and on the stability of the output (if a relatively small subset is sufficient to make a substantial change to the output, the algorithm is less stable and less confident in its output).

we will now show a reduction that shows that the exact calculation of CF1 given a function and a dataset is NP-hard, and following we will describe a greedy algorithm we devised and that our system uses to approximate CF1.

***reduction from SAT*.** Given a boolean formula ψ in variables $x_1...x_n$, we will build an instance of the counterfactual problem and show that finding the counterfactual subset is equivalent to finding a satisfying assignment to ψ.

Let D be the set $\{1...n+1\}$.
$L(\alpha, D)$ will be 1 if $\alpha = (1,0,0)$ and 0 otherwise, so the dominant species on all the dataset is species 0 ($H(D,L) = -0$).
$L(\alpha, D^{'})$ will be equal to $L(\alpha, D)$ if $D^{'}$ corresponds to not satisfying assignment of ψ, and will be the opposite if it is satisfying. An assignment $a_1...a_n$ is converted to a group simply by taking into the group the indexes that correspond to variables that are getting true in the assignment (for example, if in the assignment x_7 gets the value "true", then D_7 will be in the subset, and vice versa). The reduction is polynomial time since it is just checking the assignment (given a formula it builds the function L that only needs to convert the assignment to a subset of D and to check the assignment on the formula ψ).. If someone solves the counter factual problem and finds a subset that is changing dominance, it means that we can take the changing dominance subset and convert it to a satisfying assignment, and solve SAT. If there is no satisfying assignment, then there will be no changing dominance subset since on every $D^{'}$ the function L will be the same. Hence, if there is an algorithm that solves CF1, if it is returning a dominance changing subset, we solved SAT since we have an assignment, and if it returns that there is no dominance changing subset, we know that the formula is not satisfiable, which concludes the reduction.

algorithm 3 is a greedy algorithm that aims to approximate cf1. It is based on removing datums from the dataset that have the Shapley value that is most in favor of the dominant species, hoping to quickly converge to a dataset in which this is no longer the dominant species.

Algorithm 3: Calculating counterfactual 1

Data: dataset D, and the output of Algorithm 1 on that dataset
Result: subset that is an estimation of cf1
Assume w.l.o.g that the dominant species in D is s
Calculate scaled shapley values for every $d \in D$, define v^d as the vector of Scaled shapely value of read d, and v^d_s is the influence of this read on s proportion.
$D^{'} = \emptyset$
while $H(D \backslash D^{'}, L) == s$ **do**
    $D^{'} = D^{'} \cup d^{'}$, when $d^{'} := argmax_{d \notin D'}(v^d_s)$
end
return $D^{'}$ as an estimation to the minimal changing dominance subset

---

It is easy to see that the set that is returned is changing dominance, and that if there exists a changing dominance subset, this algorithm will return a changing dominance subset. But as stated above, this is only an approximation to the *minimum* changing dominance subset.

3.2.2 counterfactual 2: ϵ-changing subset.

Change dominating species from **'Homo Sapiens'** to **'Neanderthal'** would require removing **11** reads:
[42, 57, 54, 52, 58, 56, 43, 46, 45, 44, 59]

Change dominating species from **'Homo Sapiens'** to **'Neanderthal'** would require removing **30** reads:
[32, 34, 38, 49, 37, 57, 22, 46, 51, 42, 50, 58, 23, 55, 53, 59, 41, 36, 25, 39, 26, 44, 40, 29, 15, 2, 47, 56, 5, 24]

Figure 5: CF1 output on D_3 (top) and D_4 (bottom)

Definition 3.11. Define $CF2(D, L)$ as a minimum subset $D' \subseteq D$ that changes the model's output by at least ϵ, i.e.
$|| \ ML(D, L) - ML(D \backslash D'.L) \ || > \epsilon$.

This counterfactual is targeting subsets that cause a substantial numeric difference in the output vector of the proportion estimation algorithm. To estimate this counterfactual, we employ algorithm 4. We initialize $D' = \emptyset$, and greedily add to D' a datum d that maximizes the term $|| \sum_{i \in D'} S(i) + S(d) ||$, until $|| \ v(D) - v(D \backslash D') \ || > \epsilon$. Note that since we are comparing vector norms, d is not necessarily the datum with the highest norm of the Shapley vector, but rather one that shifts the decision in the right direction w.r.t. the existing set D'.

Algorithm 4: Calculating counterfactual 2

Data: dataset D, and likelihood function L
Result: subset that is an estimation of cf2
Calculate scaled shapley values for every $d \in D$, define v^d as the vector of Scaled shapely value of read d, and v^d_s is the influence of this read on s proportion.
$D^{'} = \emptyset$
while $|| \ ML(D, L) - ML(D \backslash D', L) \ || \leq \epsilon$ **do**
    $D^{'} = D^{'} \cup d^{'}$, when
    $d^{'} := argmax_{d \notin D'}(|| \sum_{i \in D'} S(i) + S(d) \ ||)$
end
return $D^{'}$ as an estimation for cf2

3.2.3 counterfactual 3: θ-labeled subset.

Definition 3.12. Define $CF3(D, L, s)$ as a maximum subset $D'_s \subseteq D$ such that $ML(D'_s, L)[s] \geq \theta$ for a specific $s \in S$.

This counterfactual intuitively corresponds to a subset of the data that has a high proportion of a given label. To estimate CF3 for a given species s, we employ algorithm 5.We initialize $D'_s = \emptyset$, and greedily add reads d that has the highest $S(d)$ w.r.t. s, until there are no reads for which $v(D'_s \cup d)[s] \geq \theta$.

Example 3.13 (counterfactuals). Figure 5 shows the output of *CF1* on two datasets: D_3 (top figure) has 20 reads from Homo sapiens, and 40 non-informative reads, and D_4 (bottom) has 40 reads from Homo Sapiens and 20 non-informative reads. Though Algorithm 1's output is the same for both datasets ($(1,0,0)$), Figure 5 shows that on D_3, CF1 contains 11 reads, whereas in D_4, CF1 contains 30 reads. This conveys to the user that the output on D_4 is more stable, as a larger modification is required for it to change.

Algorithm 5: Calculating counterfactual 3

Data: dataset D, likelihood function L, and label $s \in S$
Result: subset that is an estimation of cf3

Calculate scaled shapley values for every $d \in D$, define v^d as the vector of Scaled shapely value of read d, and v_s^d is the influence of this read on s proportion.
$D_s^{'} = \emptyset$
while *True* **do**
  $d^{'} := argmax_{d \notin D'}(v_s^d)$
  if $ML(D^{'} \cup d^{'}, L)[s] \geq \theta$ **then**
    continue
  else
    Break
  end
end
return $D^{'}$ as an estimation for cf3

4 IMPLEMENTATION

We implemented *exML* using python, while utilizing the following tools: *biopython* for alignments and parsing of DNA data, *joblib* for parallel computation, and *SHAP* for generating visualizations. Reference genomes were downloaded from the *NCBI database*, and simulated ancient DNA data was generated using a python script published in [5].

The source code can be found here, together with instructions on how to run the code, or to load and run a GUI we created that interacts with the system.

REFERENCES

[1] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.
[2] Yichen Liu, Xiaowei Mao, Johannes Krause, and Qiaomei Fu. 2021. Insights into human history from the first decade of ancient human genomics. *Science* 373, 6562 (2021), 1479–1484.
[3] Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc.
[4] Caitlin M Stewart, Prachi D Kothari, Florent Mouliere, Richard Mair, Saira Somnay, Ryma Benayed, Ahmet Zehir, Britta Weigelt, Sarah-Jane Dawson, Maria E Arcila, et al. 2018. The value of cell-free DNA for molecular pathology. *The Journal of pathology* 244, 5 (2018), 616–627.
[5] Benjamin Vernot, Elena I Zavala, Asier Gómez-Olivencia, Zenobia Jacobs, Viviane Slon, Fabrizio Mafessoni, Frédéric Romagné, Alice Pearson, Martin Petr, Nohemi Sala, et al. 2021. Unearthing Neanderthal population history using nuclear and mitochondrial DNA from cave sediments. *Science* 372, 6542 (2021), eabf1667.
[6] Samuel H Vohr, Rachel Gordon, Jordan M Eizenga, Henry A Erlich, Cassandra D Calloway, and Richard E Green. 2017. A phylogenetic approach for haplotype analysis of sequence data from complex mitochondrial mixtures. *Forensic Science International: Genetics* 30 (2017), 93–105.
[7] Jinliang Wang. 2003. Maximum-Likelihood Estimation of Admixture Proportions From Genetic Data. *Genetics* 164, 2 (06 2003), 747–765. https://doi.org/10.1093/genetics/164.2.747 arXiv:https://academic.oup.com/genetics/article-pdf/164/2/747/42052529/genetics0747.pdf
[8] David S Watson. 2021. Interpretable machine learning for genomics. *Human genetics* (2021), 1–15.