**Realistic visual explanations for individual predictions of the number of contributors made by any machine learning model**

**Abstract**

Using machine learning to determine the number of contributors in short tandem repeat profiles has been shown to obtain good accuracy. However, these predictions are not understandable to the biologist user who would normally determine the number of donors themselves. Therefore, we have created a visual aid that incorporates explanations from SHAP values and counterfactual examples for each prediction. Existing methods for generating counterfactuals have not attempted to handle correlated features, causing those methods to find examples that are impossible given the feature combinations. Since the features from STR data are highly corelated, we have implemented a new method that generates realistic counterfactuals on highly correlated data, and also cuts the number of feature changes in half as compared to presenting the closest counterfactual training data point.

1. **Introduction**

Deriving the number of contributors from Short Tandem Repeat (STR) profiles is a challenging task due to occluding factors such as allele sharing between donors, or allelic drop out. This becomes increasingly difficult when the number of contributors rises. Most software that is used for DNA interpretation requires the NOC to be entered by the user [1]. This could make the difference of including or excluding a person of interest, which is relevant for court.

There have been a multitude of different methods to estimate the NOC. From methods that use only allele counts in a qualitative approach such as the MAC [2], Maximum likelihood approach [3],

Probabilistic mixture algorithm [4]

NOCit tool [5] which is a likelihood-based approach taking into account both qualitative and quantitative data.

1-3 person mixtures. Synthetically created from 58 known donors. Modelling the types of peaks, peak ehights, drop out and stutter rates. posterior probability via a Monte Carlo-based approach. maximum probability method with NOCIt resulted in accuracies of approximately 83%. Computationally slow! Up to 9h for a 5 person mixture.

True allele up to ten contributors? probability model

nC-tool [6] uses both the MAC and Total Allele Count (TAC), as well as categories of drop-out to obtaining better results.

quantitative continuous model peak heights in the DNA profile and considers the

effect of artifacts and allelic drop-out. By using this software, the likelihoods of 1–4 persons’

contributions are calculated, and the most optimal number of contributors is automatically

determined; Kongoh was validated using 27 two-person mixtures, 27 three-person mixtures, and 18 fourperson mixtures. These mixtures were experimentally prepared using non-degraded DNA

from pristine blood samples.

Machine learning approach

PACE random forest [7]: 969 non-simulated DNA samples of 1 to 5 contributors generated from a combination of 120 individuals. They achieved about 90% accuracy on identifying 1, 2, 3, 4+ profiles.

Bayesian probability framework TrueAllele [8]. Closed-source. Minutes to longer for more complex

Correct estimation of the NOC is important.

The results that machine learning methods can obtain has been demonstrated to outperform standard methods [9]. This raised the question however of making these predictions understandable for experts. Without explanations, the experts cannot determine if they should trust the prediction or not. It is also difficult to defend why the expert picked one NOC over the other, if they only relied on the output of a machine learning model.

A decision tree was presented in a paper that made an attempt to make predictions more transparent [10]. However, this led to less accurate predictions (77%) and relied heavily on the filtering of artefacts, which are inherently part of STR data and the NOC estimation process. Moreover, from the explainable AI community, decision trees are not considered an explanation [source]. You are also forced to then use a decision tree, while more interesting predictors are being modelled that may perform even better in the future. The advantage of using an explanation per prediction, is that they are more accurate, and there is no need to filter through an entire decision tree.

A counterfactual is defined as

The **contribution** of this paper is as follows:

* Introduce the concept of eXplainable Artificial Intelligence (XAI) to the field of forensic science by demonstrating its value on a practical issue.
* Generate explanations for individual predictions of the Number of Contributors (NOC) by any machine learning model.
* Present the explanations consisting of SHAP values and counterfactual examples in a visualization.
* Implement a new method for finding realistic counterfactuals (ReCo) by deriving them from the training data. This produces examples that have fewer feature differences than using training examples, but are still plausible data points. To the best of our knowledge, this is the first method that handles correlated features automatically.
* Create a new realism metric that scores how plausible counterfactuals are in terms of their feature combinations, which is important with highly correlated features.

1. **Materials and methods**
   1. *Machine Learning model*

Originally, the estimation of the NOC was treated as a classification problem [1]. However, since the outputs of the model are ordinal, the problem could benefit from being tackled with a regression model. After a short benchmarking study (see Appendix 1), we concluded that a regression model can achieve better results. In this study, we used a Random Forest Regressor with default parameters.

* 1. *Data analysis and sampling*

The dataset initially consisted of 590 PowerPlex® Fusion 6C (PPF6C) profiles, either from a single donor, or a mixture up to 5 donors [10]. This NOC was based on ground-truth information about the profile. Each profile was then represented by 19 features such as allele counts, allele frequencies and peak heights. These features are almost all very highly correlated.

Tabular, numeric data. For the future, the features might change, but we expect them to remain numeric.

We noticed that this high-dimensional dataset was too sparse for generating counterfactual explanations using the training data; the most similar profile was still very different.

In sampling-based explanation approaches, a dense neighbourhood is created around the current profile to be explained. However, none of these methods can handle correlated features. Therefore, strange

* 1. ***Desiderata*** *explanations*

From meetings with the end users, we determined that there were two main questions of interest:

1. *What were the main reasons for the model to reach the current prediction?*
2. *How could the model have arrived at a different prediction?*

To answer question 1, we determined that the use of SHAP values would be sufficient to give an impression of feature importance. For correlated features, SHAP values will turn out lower for each individual feature

It is an issue that feature importance methods split the impact on the model over correlated features. The result of this issue is that the importance value for correlated contributing features is underestimated, in contrast to if their importance was left undivided. However, since main goal of these values is to give an impression of the contributing factors to a prediction, the exact values are not a priority. This part of the explanation is to observe a general sense of which features contributed to the prediction in which direction. For this purpose, we deem SHAP adequate.

To map out what the counterfactual explanations must accommodate, a list of desiderata was determined.

* Model-agnostic
* Interactive (target can be chosen by user)
* Valid (has desired output)
* Sparse (has not too many feature changes)
* Proximal (is close enough)
* *Robust (the same every time for the same profile)*
* Realistic (makes combinations of feature values that make sense)

Since determining the NOC with machine learning is still a novel approach, there is no consensus about which type of model is most fit. The NFI is looking to improve the model and used features in the future. Therefore, a model-agnostic method is preferable.

The level of interaction we determined was most valuable, was letting the user input the target prediction. Most existing methods assume a binary case, and thus do not have to concern themselves with which target to pick other than the opposite. In this problem, the range of possible values is 1-5. It is not always straightforward to pick the next-best option; different users determine different ranges of possibilities.

The next three desiderata are straightforward as they have been mentioned by most literature on counterfactual explanations. We need counterfactuals that have the desired output (are valid) [11, 12], have the fewest number of feature changes in comparison to the profile we want to explain (are sparse) [11, 12], and must not modify each feature value too much (are proximal) [12, 13].

Though diversity is often encouraged [14], we do not see its value for this problem currently. This is mainly because the explanations are new to the users and they do not want to be confused by seeing multiple, possibly contradicting examples.

The above constraints have mostly been covered quite well in the literature.

The notion of Multi-Objective Counterfactuals was first proposed with four objectives, solved by a genetic algorithm [12]. Besides distance between x and x’, and the number of feature differences, they also consider the distance to the target outcome, and plausibility based on distance to the training data.

To measure the **distance** between two data instances, we implemented an L1 norm function as shown in Equation 1.

|  |  |  |
| --- | --- | --- |
|  |  | () |

Where represents the range of the -th feature, the number of features, the profile to be explained, the counterfactual profile. distance is the measure of choice by the literature as it does not blow outlier distances out of proportion as L2 distance tends to do [12, 15, 16]. Though much of the literature scales each distance by the Median Absolute Deviation (MAD) [15, 16], this is not appropriate for the current dataset because not all features are normally distributed. If a feature with a value much larger than the MAD were to be scaled this way, the distance score would be dominated by that feature. Therefore, we scale with each feature’s range to minimize the influence of different ranges, variations, and distributions [12, 17]. This is quite robust even for unscaled and unnormalized features with lots of outliers, which is the case in this dataset. Another nice property is that . It can also be used alongside categorical variables by

replacing with .

The second objective is the number of different feature values between the two instances, measured with norm.

However, **realism** is often overlooked or handled quite poorly. One way that the realism of counterfactual is considered, is by its distance to the closest training data point [12]. Though this can give an impression of the general relation to the training data, it does not account for the correlation between features.

As an example, consider a profile with a Total Allele Count (TAC) of 98 and a Maximum Allele Count (MAC) of 6, that was predicted to have 4 contributors. To generate a counterfactual with a prediction of 3 contributors, the program might propose a profile with a TAC of 30. Though this would make the model predict a NOC of 3, the combination of the original MAC value of 6, with the new TAC value of 30 is impossible. There are a total of 23 loci in the profile; if there are 30 alleles in the entire profile (TAC), that would leave either 1 or 2 alleles per locus (30/23). It would thus be highly unlikely, or even impossible to have a locus with 6 alleles (MAC).

Though several studies have brought up the issue there should be a way to handle **correlated** features [17, 18], none have tackled this issue into their method. To the best of our knowledge, we are the first to develop a method that is suitable for datasets with correlated features.

The method was derived by starting with the training instances. These are the most realistic data points that we have. The biggest issues with these examples are that:

* They have a lot of different feature values as compared to the profile we want to explain. They are not sparse.
* Not all of these differences are relevant to arrive at their respective predictions. They are not informative.

ReCo tackles both of these issues by applying a filter to the found counterfactual instance.

We start from the original profile and its prediction:

Let the user define a target prediction that is different from the original prediction :

ReCo then finds all instances with the target prediction:

These candidates are scored by a weighted sum of two objectives; the distance score as defined in Equation 1, and the number of different features as defined in Equation 2:

We then calculate the SHAP values of both the original- and counterfactual instance:

Next, the differences in these SHAP values are measured and ordered according to misalignment with the difference in prediction. For example, if the prediction in the original profile is 3, and in the counterfactual is 4, the difference in prediction is positive. We then order the changes in SHAP values from most negative to most positive. Changes

Random **sampling**-based approaches [11, 15, 19-21], sampling through a genetic algorithm [12, 22]. Using the gradient of the loss with respect to the input [23], a method that is based on the data, not on the classifier like SHAP. None of these methods are suitable for datasets with correlated features, as they would produce unlikely feature combinations.

A similar piece of work uses **SHAP** values for the current instance to be explained, for both the predicted class A, and target class B [20]. Specific counterfactual instances are then generated by sampling nearest neighbours, changing only the features from the original instance that have negative SHAP values for class B. This approach suffers from the fact that only changing features with negative SHAP values, they limit the range of possible feature changes and therefore produce counterfactuals that are generally further away.

Similarly, a paper discusses using LIME and SHAP to generate counterfactuals from [11]. However, their method is based on highly-dimensional (1000+), behavioural or textual data. They produce counterfactuals by iteratively setting the top contributing features to 0, until the target class is reached. This is not viable in our dataset, since setting a value to 0 does not usually correspond to a realistic feature value.

Unlike studies about loan applications and similar situations, actionability is not a goal of this study. The DNA profiles cannot and will not be altered in the future.

Desiderata for explaining decision trees [13].

To present the information to the user, a visual approach was used. We incorporated information that answered both questions into one **visualization**.

For tabular data, there have been several approaches to present the information. For example, by a conversational statement [24],

Evaluation

We present a novel **realism** score which is calculated as follows

1. When the dataset is loaded, ReCo calculates each feature’s top correlated other features. . With being the feature of interest, and all other features.
2. Once a counterfactual is found, we define the differences in feature value between the counterfactual and original instance as:
3. Then for each element in , we perform the following steps:
   1. Look up its top correlated feature, and take that feature’s value
   2. Look up the combination of with of with the feature value in step a in the training data.
   3. If the combination exists, the realism score is incremented by 1, otherwise by 0.
   4. If the feature in step a was part of , we repeat step a with the next highly correlated feature.
4. **Results and discussion**
   1. *Initial features selection*

For each of the DNA profiles, 278 features were engineered. A large number of features may result in over fitting, *i.e.* very good predictions for the training set but poor results when using the test set, while a small number may ignore vital information with predictive value [36]. To select those features that are most informative of the NOC, partial correlation calculations were performed. The features MAC and TAC were fixed in partial correlation as these are used in our laboratory and can be informative on the NOC [52]. For example, MAC and TAC had the highest correlation with NOC of all features (0.92 and 0.91). Supplementary Table 5 shows the top 50 features that included information regarding the number of alleles, allele frequencies and peak heights and were spread across the loci and dye channels. This top 50 did not include features regarding *e.g.* degradation slope and/or features counting loci with seven or more alleles. Apparently, these features did not add much to the information already obtained from the other features.

* 1. *Accuracy of the machine learning models*

The top 50 features that were obtained by partial correlation were used in the training and testing phase of each of the ten algorithms (see section 2.3). Supplementary Fig. 1 shows their accuracy as a function of the number of features. The number of features that resulted in the highest accuracy for the test set was selected and is presented in Table 3 (details on GridSearch parameters are presented in Appendix 2). Accuracy ranged between 0.792 and 0.833 which is lower than presented by Marciano et al., which ranged from 0.894 to 0.962 [36]. The lower accuracy in our study can be expected as our dataset included more complex profiles and more contributors (up to five instead of up to four contributors). However, our models are not strictly comparable as our dataset not only included more complex profiles, it also differed regarding the STR typing kit that was used and the features that were engineered.

Table 3. Train and test accuracy obtained per model (algorithm plus features) sorted from overall best to least performing model. For each algorithm, only the best performing number of features is shown.

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **Number of features** | **Accuracy**  **training set** | **Accuracy**  **test set** |
| LDA | 40 | 0.888 | 0.833 |
| RFC | 19 | 0.874 | 0.833 |
| LRC | 8 | 0.823 | 0.825 |
| GBC | 11 | 0.886 | 0.817 |
| SVC | 5 | 0.811 | 0.808 |
| LSVC | 4 | 0.803 | 0.808 |
| k-NN | 3 | 0.834 | 0.800 |
| DTC | 14 | 0.843 | 0.800 |
| MLPC | 8 | 0.842 | 0.798 |
| GaussianNB | 4 | 0.794 | 0.792 |

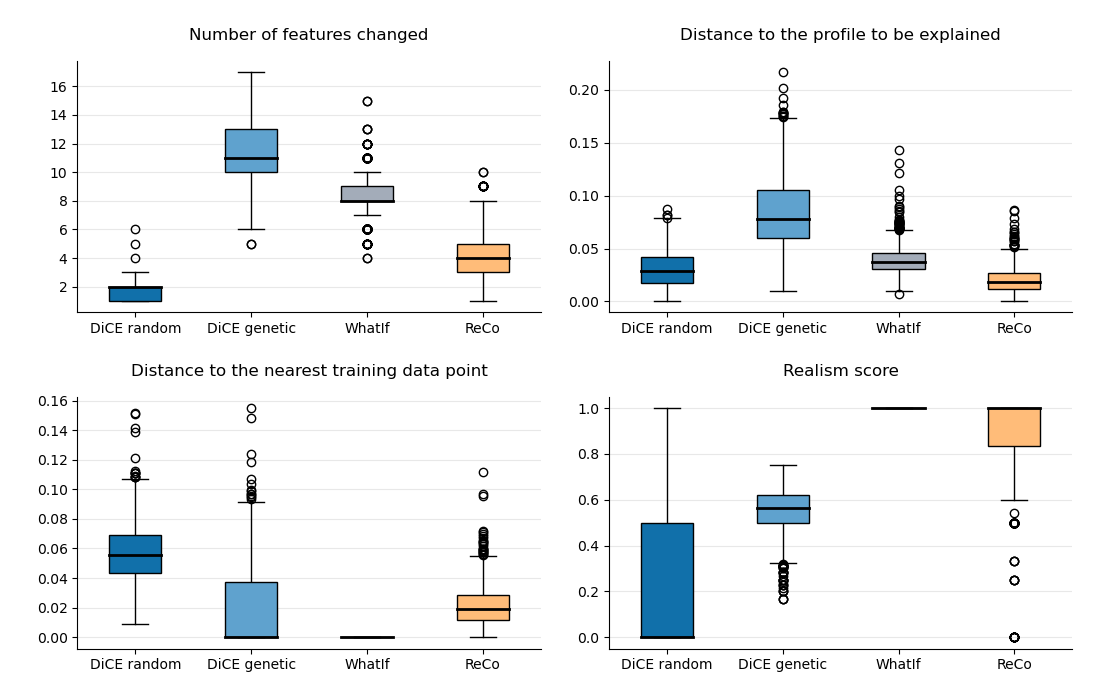
For all models but one (LSVC4), all incorrect predictions on the test set were one lower or one higher than the true NOC (Supplementary Fig. 2). Two models showed best train and test accuracy, *i.e.* Random Forest Classifier with 19 features and Linear Discriminant Analysis with 40 features, denoted as RFC19 and LDA40, respectively. Both models yielded 83.3% correctly predicted NOC for the test set and comparable accuracy for the training set (Table 3 and Supplementary Fig. 1).

Besides accuracy, precision and recall are measures of relevance and can be used to compare the performance of the various models. When generating precision-recall plots and comparing the ten models it becomes clear that not one of the models outperformed all others as precision and recall differed per NOC (Supplementary Fig. 3). It was expected that precision and recall decreases with an increasing NOC except for the precision of 5p mixtures as these cannot be over-estimated using the machine learning models in this study. This trend was observed for most of the models, including LDA40 and RFC19 (Supplementary Fig. 3).

RFC19 and LDA40 were selected to further assess their performance.

* 1. ReCo versus the state of the art

To determine the quality of ReCo, we have compared it against the current counterfactual methods. WhatIf is a method based on training data. While DiCE implements sampling approaches. DiCE genetic is actually a Python adaptation of .



While DiCE random performs best in terms of the number of differences, and quite well on distance, it performs poorly on realism and is the furthest away from the training data. This is because DiCE random starts from the original instance, and perturbs a random feature until the target prediction is reached. This strategy helps keep the number of feature differences and the overall distance score low, but does not in any way account for the relations between the features.

An improvement can be seen when the genetic version is used; the median realism score almost hits a ‘sufficient’ 0.6, and the distance to the training data is practically zero. We can attribute the higher realism score to the ‘mating’ of the DNA profiles.

* 1. Future work

The R

1. **Conclusion**

This study describes

**Acknowledgements**

We are thankful to Corina Benschop for insightful discussions, Jim for sampling 5000 DNA mixtures with EuroForMix, BiS for participating in the user studies,

**References**

1. P.M. Schneider, R. Fimmers, W. Keil, G. Molsberger, D. Patzelt, W. Pflug, T. Rothamel, H. Schmitter, H. Schneider, B. Brinkmann, The German Stain Commission: recommendations for the Interpretation of mixed stains, Int. J. Legal Med. 123 (2009) 1–5.

**Supplementary Material**

**G:\BI\BI_ALG\R&D\Research\Corina\Tables & figures\NOC tool\Train test accuracies 10 algorithms copy.tif**

Supplementary Figure 1. Train and test accuracies per algorithm and per number of features. A) LDA, B) RFC, C) LR, D) GBC, E) SVC, F) LSVC, G) k-NN, H) DTC, I) MLPC, J) GaussianNB. The selected number of features per algorithm is presented in the green circle.

**G:\BI\BI_ALG\R&D\Research\Corina\Tables & figures\NOC tool\Train test per NOC per model copy.tif**

Supplementary Figure 2. Confusion matrices for the train and test set per model. A) LDA40, B) RFC19, C) LR8, D) GBC11, E) SVC5, F) LSVC4, G) k-NN3, H) DTC14, I) MLPC8, J) GaussianNB4.



Supplementary Figure 3. Precision-recall plots for the test set per true NOC. A) LDA40, B) RFC19, C) LR8, D) GBC11, E) SVC5, F) LSVC4, G) k-NN3, H) DTC14, I) MLPC8, J) GaussianNB4.

G:\BI\BI_ALG\R&D\Research\Corina\Tables & figures\NOC tool\NOC probabilities RFC LDA incorrect sample comparison copy.tif

Supplementary Figure 4. Probabilities per NOC for DNA profiles in the test set that received an incorrect NOC prediction when using RFC19 (A) or LDA40 (B). Empty bars indicate that the NOC for this sample was correctly predicted by the particular model.

Supplementary Table 1. Overview of PowerPlex® Fusion 6C DNA profiles in the ‘extremes’ dataset.

|  |  |  |
| --- | --- | --- |
| **Extreme because:** | **Number of contributors** | **Number of samples** |
| More donors than the model was trained with | 6 | 16 |
| Relatives (brothers) within the mixtures in combination with a large degree of allelic drop-out | 2 | 5 |
| 3 | 3 |
| 4 | 2 |
| Extreme level of degradation (at least three locus drop-outs) | 3 | 3 |
| 4 | 3 |
| 5 | 3 |

Supplementary Table 2. Overview of the numbers of DNA-profiles used to examine the effect of replicates.

|  |  |  |
| --- | --- | --- |
| **Number of Contributors** | **Number of different DNA extracts** | **Total number of DNA-profiles**  **(including replicates)** |
| 2 | 30 | 90 |
| 3 | 28 | 84 |
| 4 | 29 | 87 |
| 5 | 27 | 81 |
| Total | 114 | 342 |

Supplementary Table 3. Overview of the 25 sample features. Sample features take account of all 23 autosomal loci within the PowerPlex® Fusion 6C profiles. Amelogenin and the Y-chromosomal markers were excluded from the analyses.

|  |  |  |
| --- | --- | --- |
| **Number** | **Feature** | **Details** |
| 1 | MAC | Maximum allele count (MAC); Maximum number of alleles observed on a locus |
| 2 | TAC | Total allele count (TAC); Total number of alleles per profile |
| 3 | Mean Allele count | Mean, median or standard deviation of the number of alleles per locus |
| 4 | Median Allele Count |
| 5 | Standard Deviation Allele Count |
| 6 | Minimum Allele Count | Minimum number of alleles observed per locus |
| 7 | Minimum NOC | Minimum number of contributors (NOC); Maximum Allele Count / 2, rounded up to 0 decimals |
| 8 | AC 0 | Number of loci with an allele count of 0 (*i.e.* empty loci/ locus drop-outs), 1 or 2, 3 or 4, 5 or 6, 7 or 8, or 9 alleles or more. |
| 9 | AC 1-2 |
| 10 | AC 3-4 |
| 11 | AC 5-6 |
| 12 | AC 7-8 |
| 13 | AC ≥9 |
| 14 | Maximum PH | Largest, lowest, mean, median or standard deviation of the peak height (PH) of an allele (RFUs) observed across the profile |
| 15 | Minimum PH |
| 16 | Mean PH |
| 17 | Median PH |
| 18 | Standard deviation PH |
| 19 | Degradation Slope1 | Slope of line where  Y= sum of peak heights per locus and X= average fragment length per locus |
| 20 | Degradation Slope2 | Slope of line where  Y= average of peak heights per locus and X= average fragment length per locus |
| 21 | Number of Peaks above ST | Number of peaks above or below, or the ratio between the number of peaks above/below or below/above the stochastic threshold (ST) of 800 RFUs |
| 22 | Number of Peaks below ST |
| 23 | Ratio peaks below/above ST |
| 24 | Ratio peaks above/below ST |
| 25 | Match probability | The probability of a random, unrelated person matching to this DNA profile. The probability is calculated using the allele frequencies of 2085 male Dutch individuals database [40]. |

Supplementary Table 4. Overview of the 11 locus features which were calculated for each of the 23 autosomal loci within the PowerPlex® Fusion 6C profiles.

|  |  |  |
| --- | --- | --- |
| **Number** | **Feature** | **Details** |
| 1 | Allele count | Number of alleles |
| 2 | Minimum NOC | Allele count / 2, rounded up to 0 decimals |
| 3 | Maximum PH | The largest, smallest, mean, median or standard deviation of the peak height (PH, in RFUs) of alleles at the particular locus |
| 4 | Minimum PH |
| 5 | Mean PH |
| 6 | Median PH |
| 7 | Standard deviation PH |
| 8 | Minimum AF | The lowest or highest allele frequency (AF) of an allele, or the sum of the allele frequencies of the alleles, or the percentage of alleles that are within the population database. *I.e.*: |
| 9 | Maximum AF |
| 10 | Sum AF |
| 11 | Percentage of AF |

Supplementary Table 5. Top 50 ranked features and their partial correlation (correlation for TAC and MAC). Features are ordered based on the sequential algorithm described in section 2.2. Further details on these features are presented in Supplementary Tables 3 and 4.

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Feature** | **Sample/ locus feature** | **Partial Correlation** |
| 1 | MAC | Sample | 0.9209 |
| 2 | TAC | Sample | 0.9133 |
| 3 | Standard Deviation Allele Count | Sample | 0.4214 |
| 4 | Allele count\_D3S1358 | Locus | 0.2904 |
| 5 | AC 5-6 | Sample | 0.2577 |
| 6 | Minimum NOC\_Penta E | Locus | 0.2100 |
| 7 | Minimum NOC\_Penta D | Locus | 0.2123 |
| 8 | AC 0 | Sample | 0.1962 |
| 9 | Standard deviation PH\_vWA | Locus | 0.1911 |
| 10 | Match Probability | Sample | 0.1972 |
| 11 | Number of Peaks below ST | Sample | 0.1621 |
| 12 | Minimum NOC\_TPOX | Locus | 0.1547 |
| 13 | Minimum NOC | Sample | 0.1811 |
| 14 | Minimum NOC\_CSF1PO | Locus | 0.1435 |
| 15 | Minimum NOC\_D16S539 | Locus | 0.1518 |
| 16 | Sum AF\_TH01 | Locus | 0.1188 |
| 17 | Allele count\_TPOX | Locus | 0.1242 |
| 18 | Percentage of AF\_D1S1656 | Locus | 0.1290 |
| 19 | Allele count\_D8S1179 | Locus | 0.1174 |
| 20 | AC 3-4 | Sample | 0.1264 |
| 21 | Standard deviation PH\_D2S441 | Locus | 0.1330 |
| 22 | Minimum AF\_D22S1045 | Locus | 0.1369 |
| 23 | Sum AF\_SE33 | Locus | 0.1266 |
| 24 | Minimum AF\_Penta E | Locus | 0.1224 |
| 25 | Allele count\_D18S51 | Locus | 0.1264 |
| 26 | Minimum AF\_D5S818 | Locus | 0.0978 |
| 27 | Minimum AF\_D8S1179 | Locus | 0.0933 |
| 28 | Ratio peaks above/below ST | Sample | 0.0978 |
| 29 | Median PH\_D19S433 | Locus | 0.1038 |
| 30 | Minimum NOC\_vWA | Locus | 0.0951 |
| 31 | Median PH\_D7S820 | Locus | 0.0847 |
| 32 | Median PH\_Penta D | Locus | 0.1021 |
| 33 | Sum AF\_FGA | Locus | 0.1022 |
| 34 | Median PH\_TPOX | Locus | 0.0942 |
| 35 | Median PH\_D2S441 | Locus | 0.1190 |
| 36 | Maximum PH\_Penta D | Locus | 0.0964 |
| 37 | Maximum PH\_Penta E | Locus | 0.1266 |
| 38 | Minimum NOC\_D3S1358 | Locus | 0.0928 |
| 39 | Mean PH\_D19S433 | Locus | 0.0909 |
| 40 | Minimum AF\_CSF1PO | Locus | 0.0887 |
| 41 | Minimum NOC\_FGA | Locus | 0.0898 |
| 42 | Minimum AF\_D2S441 | Locus | 0.0898 |
| 43 | Standard deviation PH\_TH01 | Locus | 0.0917 |
| 44 | Minimum NOC\_D12S391 | Locus | 0.0806 |
| 45 | Percentage of AF\_D7S820 | Locus | 0.0837 |
| 46 | Minimum NOC\_D7S820 | Locus | 0.1172 |
| 47 | Sum AF\_D10S1248 | Locus | 0.0980 |
| 48 | Minimum PH\_Penta E | Locus | 0.0840 |
| 49 | Median PH\_Penta E | Locus | 0.0954 |
| 50 | Mean PH\_Penta E | Locus | 0.1616 |

Supplementary Table 6. Comparison of the performance of the MAC method, nC-tool and RFC19 machine learning model for 2p-5p PPF6C profiles.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **MAC** | | | **TAC nC-tool** | | | **RFC19 machine learning** | | |
| **True NOC (*n* per method)** | **Under-estimated** | **Correctly estimated** | **Over-estimated** | **Under-estimated** | **Correctly estimated** | **Over-estimated** | **Under-estimated** | **Correctly estimated** | **Over-estimated** |
| 2 (*n*=90) | 0% | 67% | 33% | 0% | 100% | 0% | 0% | 100% | 0% |
| 3 (*n*=88) | 0% | 97% | 3% | 17% | 83% | 0% | 3% | 97% | 0% |
| 4 (*n* =89) | 20% | 77% | 3% | 27% | 70% | 3% | 17% | 83% | 0% |
| 5 (n =87) | 63% | 37% | 0% | 47% | 53% | not applicable | 40% | 60% | not applicable |
| Total (*n* =354) | 21% | 69% | 10% | 23% | 77% | 1% | 15% | 85% | 0% |

1. Taylor, D., J.-A. Bright, and J. Buckleton, *Interpreting forensic DNA profiling evidence without specifying the number of contributors.* Forensic Science International: Genetics, 2014. **13**: p. 269-280.

2. Clayton, T.M., et al., *Analysis and interpretation of mixed forensic stains using DNA STR profiling.* Forensic Science International, 1998. **91**(1): p. 55-70.

3. Haned, H., et al., *Estimating the Number of Contributors to Forensic DNA Mixtures: Does Maximum Likelihood Perform Better Than Maximum Allele Count?* Journal of Forensic Sciences, 2011. **56**(1): p. 23-28.

4. Paoletti, D.R., et al., *Inferring the Number of Contributors to Mixed DNA Profiles.* IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2012. **9**(1): p. 113-122.

5. Swaminathan, H., et al., *NOCIt: A computational method to infer the number of contributors to DNA samples analyzed by STR genotyping.* Forensic Science International: Genetics, 2015. **16**: p. 172-180.

6. Benschop, C., A. Backx, and T. Sijen, *Automated estimation of the number of contributors in autosomal STR profiles.* Forensic Science International: Genetics Supplement Series, 2019. **7**.

7. Marciano, M.A. and J.D. Adelman, *Developmental validation of PACE™: Automated artifact identification and contributor estimation for use with GlobalFiler™ and PowerPlex® fusion 6c generated data.* Forensic Science International: Genetics, 2019. **43**.

8. Bauer, D.W., et al., *Validating TrueAllele® Interpretation of DNA Mixtures Containing up to Ten Unknown Contributors.* Journal of Forensic Sciences, 2020. **65**(2): p. 380-398.

9. Benschop, C.C.G., et al., *Automated estimation of the number of contributors in autosomal short tandem repeat profiles using a machine learning approach.* Forensic Science International: Genetics, 2019. **43**: p. 102150.

10. Kruijver, M., et al., *Estimating the number of contributors to a DNA profile using decision trees.* Forensic Science International: Genetics.

11. Ramon, Y., et al., *A comparison of instance-level counterfactual explanation algorithms for behavioral and textual data: SEDC, LIME-C and SHAP-C.* Advances in Data Analysis and Classification, 2020. **14**(4): p. 801-819.

12. Dandl, S., et al. *Multi-Objective Counterfactual Explanations*. in *Parallel Problem Solving from Nature – PPSN XVI*. 2020. Cham: Springer International Publishing.

13. Sokol, K. and P. Flach. *Desiderata for interpretability: Explaining decision tree predictions with counterfactuals*. 2019.

14. Russell, C. *Efficient search for diverse coherent explanations*. 2019.

15. Wachter, S., B. Mittelstadt, and C. Russell, *Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR.* Harvard journal of law & technology, 2018. **31**: p. 841-887.

16. Mothilal, R.K., A. Sharma, and C. Tan. *Explaining machine learning classifiers through diverse counterfactual explanations*. 2020.

17. Karimi, A.-H., et al. *Model-agnostic counterfactual explanations for consequential decisions*. in *International Conference on Artificial Intelligence and Statistics*. 2020. PMLR.

18. Barocas, S., A.D. Selbst, and M. Raghavan. *The hidden assumptions behind counterfactual explanations and principal reasons*. 2020.

19. Grath, R.M., et al., *Interpretable Credit Application Predictions With Counterfactual Explanations.* ArXiv, 2018. **abs/1811.05245**.

20. Rathi, S., *Generating Counterfactual and Contrastive Explanations using SHAP*. 2019.

21. White, A. and A. Garcez. *Measurable Counterfactual Local Explanations for Any Classifier*. in *ECAI*. 2020.

22. Guidotti, R., et al., *Factual and Counterfactual Explanations for Black Box Decision Making.* IEEE Intelligent Systems, 2019. **34**(6): p. 14-23.

23. Moore, J., N. Hammerla, and C. Watkins, *Explaining deep learning models with constrained adversarial examples*. 2019. p. 43-56.

24. Sokol, K. and P. Flach. *Conversational Explanations of Machine Learning Predictions Through Class-contrastive Counterfactual Statements*. 2018.