Document Title:	A Hybrid Machine Learning Approach for DNA Mixture Interpretation
Author(s):	Michael A. Marciano, M.S., Kevin S. Sweder, Ph.D.
Document Number:	251804
Date Received:	July 2018
Award Number:	2014-DN-BX-K029

A Hybrid Machine Learning Approach for DNA Mixture Interpretation

Final Draft Summary Report
Reporting Period End Date: April 2016

Submitted to:
United State Department of Justice, Office of Justice Programs, National Institute of Justice

Submission Date
June 6, 2016

Funding Opportunity #:
Grants.gov No. NIJ-2014-3744

Period of Performance
January 01,2015- June 30,2016

Awardee Address
113 Bowne Hall
Syracuse University
Syracuse, NY 13244-1200

DUNS:
Fed EIN:

Co-PI and Technical Point of Contact:
Michael A Marciano, M.S.
Office: 315-443-5279
Fax: 315-443-4040
Email: mamarcia@syr.edu

PI
Kevin S. Sweder, Ph.D.
Office: 315-443-3396
Email: kssweder@syr.edu
Kevin S. Sweder

Submitting Official and Administrative Point of Contact:
Amy Graves
Office: 315-443-9360
Fax:315-443-9361
ajgraves@syr.edu

Signature Amy J. Graves

Authors: Michael A. Marciano, M.S.[1,2] and Jonathan Adelman, M.S.[1,2]

[1.] Forensic & National Security Sciences Institute, Syracuse University, Syracuse, NY 13244
[2.] Both authors contributed equally.

Introduction

DNA mixtures are defined as a mixture of the DNA of two or more donors. The ability to separate or "deconvolute" the individual donors from a DNA mixture remains one of the most critical challenges in the field of forensic DNA analysis. Several metrics are required to accurately interpret and deconvolute DNA mixtures; a selection of the most critical include the number of contributors, the minimum expected heterozygote balance, the ratio of contributors, the DNA template, and the probabilities of allele drop-out and drop-in. Specifically, the number of contributors is widely considered the most critical component in leading to an accurate DNA mixture deconvolution, in large part due to the deconvolution's sensitivity to whichever number of contributors is assumed. Likelihood-based deconvolution methods require the potentially erroneous assumption that the number of contributors is known to the analyst. Indeed, the assumption of the number of contributors can greatly affect the resulting conclusions (SWGDAM Interpretation Guidelines for Autosomal STR Typing, 2011); establishing the number of contributors permits the analyst to set a range of potential alleles at a particular locus within the sample and proceed with mixture deconvolution, but the use of incorrect assumptions regarding the number of contributors can have at times extremely adverse effects on the resulting likelihood ratios (Benschop et al. 2012) and therefore the mixture interpretation as a whole. Therefore, making high-probability estimates of the number of contributors in a given mixture should be considered a vital component of DNA mixture deconvolution. Although the goal of this project was to investigate a new machine learning-based method of mixture deconvolution, a secondary outcome was a superior model for estimation of the number of contributors in a DNA mixture.

Machine learning refers to the development of systems that can learn from data. A machine learning algorithm can, after exposure to an initial set of data, be used to generalize; that is, it can evaluate new, previously unseen examples and relate them to the initial "training" data. Machine learning is a widely used approach with an incredibly diverse range of applications, with examples such as object recognition (Duygulu et al. 2002), natural language processing (Jurafsky & Martin 2009), and DNA sequence classification (Cho & Won 2003). It is ideally suited for classification problems involving implicit patterns, and is most effective when used in conjunction with large amounts of data. Although machine learning has not previously been used within the domain of DNA mixture analysis, the problem area is well-suited to such an endeavor due to two key problem characteristics: there exists a large repository of human DNA mixture data in electronic format, and these data are high-dimensional and complex; patterns in such data are often non-obvious and beyond the effective reach of manual analysis but can be statistically evaluated using one or more machine learning algorithms.

In the following draft report, we describe a novel method to probabilistically deconvolute DNA mixtures using a machine learning approach. The conclusions generated are based on the use of both categorical (qualitative) data such as allele labels, dye channels and continuous and discrete (quantitative) data such as stutter rates, peak heights, heterozygote balance, and mixture ratios that describe the DNA sample. The method is computationally inexpensive, and results are obtained in a maximum of 10 seconds using a standard desktop or laptop computer with 6-8 GB RAM and an Intel i5 1.9gHz processor.

Materials and Methods

Data acquisition and exportation

The system was trained, tested and validated using electronic data (.fsa files) obtained from 1405 non-simulated DNA mixture samples comprised of 1-4 contributors and generated from a combination of 16 individuals. This set of 1405 samples included 35 different template amounts from 0.0125 ng to 10 ng (0.0125, 0.025, 0.05, 0.0625, 0.075, 0.1, 0.125, 0.15, 0.2, 0.25, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1.0, 1.2, 1.3, 1.5, 1.6, 1.7, 2.0, 2.5, 3.0, 3.3, 3.5, 4.0, 5.0, 6.0, 7.0, 7.5, 8.0, 9.0, 10.0) and 28 different ratios of contributors (1.6:3:1:1, 1.6:3:1:2, 1.6:3:2:1, 1.6:3:2:2, 1.6:6:1:1, 3:3:1:1, 3:6:1:1, 1.6:12:1, 1.5:3:1, 1.5:3:2, 1.5:6:1, 1.6:3:4, 12:1:1, 1:3:1/3:1:1/1:1:3, 3:1:2, 3:1:4, 3:2:1, 3:3:1, 3:4:1, 6:1:1, 6:3:1, 1:0, 1:1, 1:19/19:1, 1:2/2:1, 1:4/4:1, 1:9/9:1, 10:1). These samples were previously amplified using the AmpFLSTR® Identifiler® PCR Amplification Kit (Thermo Fisher Scientific Inc.) with detection performed on five different 3130/3100 Genetic Analyzers (Thermo Fisher Scientific Inc.). Data were exported from GeneMarkerHID (SoftGenetics LLC) for further analysis; Figures 1 through 4 detail the GeneMarker settings and resulting report used for exporting data in the format required for subsequent feature selection and model development.

Locus-sample-specific threshold calculation

We have devised a locus and sample specific threshold (LSST) scheme for application of the analytical threshold to capillary electrophoretic data. LSST was created to increase the discriminatory power of each locus independent of one another, avoiding both the global and dye channel-specific thresholds, which may cause true peaks to be discarded as noise. In addition, this method will enable the concurrent analyses of samples generated from different capillary electrophoresis instruments.

LSST is calculated for each locus by taking the mean of the region flanking each locus, outside of the calling region; refer to Table 1 for threshold bounds for each locus when using the Identifiler kit (Life Technologies). Two notable exceptions exist for FGA and D18. These loci are the largest loci in the Identifiler kit; although the calling ranges are large, those loci with the highest frequency across many populations exist in the smaller base pair size regions (Kidd et al. 2003, Rajeevan et al. 2003, Rajeevan et al. 2005, Rajeevan et al. 2011). More than 21 alleles in the D18 locus have a mean frequency of < 0.0075 ± 0.008 across 168 populations and similarly, FGA >28 alleles have and average frequency of 0.0055 ± 0.006. Therefore, the FGA and D18 calling regions at base pair positions of the >28 and >21 alleles, respectively, will serve as a lower bound for the right flanking region of the locus. This captures the local baseline environment of each locus based on position within each dye channel.

For locus l:

$$X_l = \begin{cases} x = 0, for\ x < 0 \\ x = x\ , for\ x > 0 \end{cases}$$

$$\frac{\left(\sum_{l_{PRE_{ST}}}^{l_{PRE_E}} x_{l_{PRE}} + \sum_{l_{PST_{ST}}}^{l_{PST_E}} x_{l_{PST}}\right)}{n_{PRE}+n_{PST}} \qquad (2)$$

…where $\ell_{PRE_{ST}}$ is the pre-locus threshold start, $\ell_{PRE_{E}}$ is the pre-locus threshold end, $\ell_{PST_{ST}}$ is the post-locus threshold start, $\ell_{PST_{E}}$ is the post-locus threshold end, and x_{ℓ} is the height of the baseline at n for locus ℓ.

The calculation of each LSST is dependent on the availability of post-analysis trace data. The SoftGenetics GeneMarker HID program makes these data readily available and exportable. Once exported, the trace data must be associated with base pair size and the calling regions identified, with each base equivalent to 10 data points. Each locus has two LSST regions identified, a pre-locus region and a post-locus region. The heights of the baseline at each data point within these regions are averaged and the threshold is set at four standard deviations above the mean. The calculation includes two additional corrections. First, any negative baseline values are set to zero, which provides for an increasingly conservative estimation of the threshold while avoiding any software-based corrections due to the application of spectral corrections or artifacts. Second, elevated peak heights in a peak in one dye channel can lead to an artificial increase in the baseline in other dye channels, commonly referred to as pull-up. This artificial increase in regions used for threshold calculation can cause the threshold to become unnecessarily elevated. To account for this phenomenon a filter is applied to those regions used in the calculation of the LSST.

Data partitioning

A fully-trained machine learning algorithm, on its own, cannot be generalized to new, unfamiliar data. Such an algorithm is capable of creating a model that makes good predictions only if future data are selected from the initial training library. To generalize, the learner must be both trained and tested, and the library of DNA mixtures must correspondingly be partitioned into training and testing subsets. For all modeling efforts herein, the training dataset was created by randomly selecting 75% of the initial data, with the other 25% used for testing how generalizable the learned model is.

All hyperparameters of machine learning algorithms were set to default values from Python's Scikit-Learn library (Pedregosa et al. 2011), unless otherwise noted. Any tuning of hyperparameters for an algorithm were performed using 5-fold cross-validation on the training data set, thereby created a partitioned validation set.

Feature scaling

Learning algorithms make use of data instances, each one of which has a corresponding feature vector. Most machine learning algorithms cannot appropriately utilize the raw features in this vector because feature scales can be wildly different from one another. The template DNA feature, for example, has mean and variance several orders of magnitude smaller than those of the maximum peak height feature. Distinct means and variances can lead to importance of some features being artificially inflated by learning algorithms, which are spending disproportionate amounts of time minimizing the larger errors produced by the features with the larger variances. While many researchers choose to resolve this concern by simply normalizing feature data via min-max scaling to a range of [0, 1], some learning algorithms learn model weights more quickly and are more robust in the face of data outliers if features are instead standardized:

$$X_{std}^{(i)} = \frac{X^{(i)} + \mu_x}{\sigma_x} \qquad (3)$$

Here $X^{(i)}$ is a given feature, μ_x is the feature's mean, and σ_x is the corresponding standard deviation. All feature scaling in this study was performed using Equation 3.

Feature selection

Once LSST, pull-up, and stutter calculations have been performed on all sample data, candidate features will be exported for analysis and subsequent use. All candidate features are ranked by Kullback–Leibler divergence (Equation 4), which is a measure of the reduction in entropy of the class variable (in this case, the true number of contributors) after the value for the feature is observed.

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}. \qquad (4)$$

All calculations were performed using the Weka Knowledge Analysis Environment, version 3.8 (Hall et al. 2009). Any candidate feature with a divergence below 0.01 was removed prior to machine learning.

Machine learning algorithms

The following six machine learning algorithms were used to create classification models: (1) k-Nearest Neighbor, (2) Classification and Regression Trees (CART) (Quinlan 2014), (3) Multinomial Logistic Regression, (4) Multilayer Perceptron (MLP), (5) Support Vector Machine (SVM), and (6) AdaBoost. In all cases, each model derived from these algorithms was evaluated based on both its classification accuracy and the convergence of its training and testing accuracies as shown in a learning curve, with the latter meant to gauge how well the model might perform when faced with new, previously unseen data. All machine learning algorithms were implemented using Python's Scikit-Learn library (Pedregosa et al. 2011) with the exception of the multilayer perceptron, which was implemented in Weka (Hall et al. 2009). All hyperparameters were set as shown in Table 2.

The accuracy of the NOC model classifications (e.g. 1,2,3 or 4 contributors) was determined through comparison of the results obtained from the classification of the testing data set with the known number of contributors. Correct calls represent that class (number of contributors) that yield the highest probability compared to the known number of contributors. For example, given the results for a hypothetical sample *A*: Pr(NOC:1) = 0.000117; Pr(NOC:2) = 0.000154; Pr(NOC:3) = 0.977103; Pr(NOC:4) = 0.022627, the NOC model classifies sample *A* as a mixture with 3 contributors; given the known number of contributors is 3, the sample is identified as a correct classification.

Results

Results pertaining to the deconvolution of DNA mixtures remain in progress and have relied upon the installation of an allele dropout model and the NOC model. The following results represent progress of the deconvolution models as of April 2016. The reported models were developed using 1-3 contributor samples.

Two models were developed to deconvolute mixtures, a model to deconvolute the major contributor and a second to deconvolute the entire sample. For both major and full deconvolution, the model using non-linear, kernel-based support vector machines provided the highest level of accuracy, 82.4% and 54% respectively. These results can be interpreted as the percentage of correct answers given samples that were previously unseen by the model. When considering the accuracy by contributor number of the model for the deconvolution of the major contributor we see 100% accuracy for the single source samples, 81.6% for the 2-contributor samples and 79.6% for the 3- contributor. When considering the model for full-deconvolution, accuracy of single source samples is 96%, 53.6% for 2 contributors and 16.6% for 3-contributor. These results reflect the accuracy based on the highest probability outcome. When the results are further broken down by template, the major contributor model accuracy (Figure 5) is highest in the 0.5-4.0ng range, with the highest proportion of misclassifications above 5.0ng. The same pattern is present in the full-deconvolution model (Figure 6). Further optimization of the models is ongoing, including the implementation of a dropout model, expansion to 4 contributors and inclusion of the NOC model. Significantly improved results are expected.

Number of Contributors

Kullback–Leibler divergence was calculated for ten candidate features (Table 3), and the base pair size of a locus was removed from the list of candidate features after achieving a divergence of 0. All other features were kept, and used in subsequent machine learning.

Summary metrics for all learning algorithms are found in Table 4. Support for the number of contributors in the testing data set is as follows: 472 samples with 1 contributor, 756 samples with 2 contributors, 350 samples with 3 contributors, and 171 samples with 4 contributors. It is evident from these results that a kernel-based support vector machine produces a tightly converging model with high classification accuracy rates, scoring most highly of all classifiers in both convergence and accuracy. The learning curve for a kernel-based SVM model (Figure 7) illustrates model convergence for the top-performing algorithm's model.

The performance of the SVM-derived classification model was compared to the Maximum Allele Count (MAC) method, as well as to results from NOCit software (Figure 8). Note that results from MAC and NOCit are based on samples amplified using 0.25 – 0.016 ng of DNA, which represents a smaller range of template amounts that would arguably be easier to classify than would be the testing data set used in this study for machine learning. The SVM-derived model outperformed MAC in all cases, performed similarly to NOCit for one- and two-contributor samples, and strongly outperformed NOCit for three- and four-contributor samples despite running ~3,600 times faster (~9 hours vs. ~9 seconds). It should be noted that classification performance suffered most when the SVM-derived model attempted to classify samples containing allelic drop-out. While NOCit has already been optimized to detect drop-out and predict accordingly, the model has yet to incorporate drop-out in any form.

Training and testing sets were compiled from the aforementioned samples using a proportionally stratified sampling of the overall data set (Tables 5-6). The contributor classes were proportionally represented in the training and testing sets, with no overlap in the samples included in each set. Classifications resulting from the samples in the testing set are independent of the samples used to create the NOC model. The overall model accuracy is 98.2%, meaning 98.2% of the sample classifications (e.g. 1,2,3 or 4 contributors) were correct based on a comparison of the model's classifications with the known number of contributors. Classification of unknown single source and 4 contributor samples yielded 100% accuracy, with 94 and 30 samples, respectively. The 100% accuracy of the 4 contributor samples will be further explained in the following section. The 2- and 3- contributor samples displayed 98.1% (151/154) and 94.6% (70/74) accuracy, respectively (Table 7).

Those samples with incorrect classifications were misclassified by a maximum of ±1 contributor; the 3 misclassifications in the 2 contributors group included 2 samples misclassified as a single source and 1 as a 3-contributor sample. Similarly, the 4 samples misclassified in the 3 contributor group were classified as either 2 contributor or 4 contributor samples. Figures 9-10 display the accuracy of the model across the DNA template (ng) amplified. As anticipated 6 of the 7 misclassifications occur below 0.75ng of template DNA amplified. Further data regarding the misclassifications can be observed in Table 8.

Discussion

The proposed method for mixture deconvolution, including determining the number of contributors, is a robust and reproducible method that was developed using an expansive AmpFlSTR® Identifiler® PCR Amplification Kit (Thermo-Fisher Scientific). The use of the Identifiler® data set was due to current availability; however, similar training data tests can be compiled from multiple laboratories' validation with additional samples run. A noteworthy aspect of the mixture deconvolution method is its independence based on instrument and injection parameters. The data presented were compiled from 5 different capillary electrophoresis instruments at 2 different laboratories, in addition the same data had varied injection times (2-22 seconds) and kV used for injection (1-5). This is a significant advantage that would permit the model to be easily transferred between laboratories and not require significant resources to perform internal validation. More detailed discussion of the optimized system will be addressed in the Final Report.

The NOC model's accuracy is greater than 98%, with only 7 misclassifications observed in the 2- and 3- contributor sample groups. The 100% accuracy experienced in the 4 contributor group is primarily due to the lack of 5 (or greater) contributor samples for training; this group can be more accurately thought of as a "≥4 contributor" group. The system probabilistically determined these samples to not be 1-,2- or 3- contributor samples and classified them as having ≥4. The accuracy would be expected to drop slightly from the 100% classification accuracy if 5 or 6 contributor samples are included in the training data. In practice, the value of classifying a profile with having ≥4 contributors is significantly high, as many laboratories choose to not interpret profiles over greater than 3 contributors.

The NOC system is proposed as a valuable tool in the analyst assessment of the number of contributors. Of the 7 misclassifications (out of 351 total samples) 4 could be corrected if an analyst briefly reviewed the data, through the identification of artifactual peaks. The remaining 3 were due to dropout, allele sharing and high template whereas an analyst or software did not have significant evidence to accurately predict the number of contributors. Therefore, with analyst influence the system has an accuracy rate of over 99.0%.

While the training data used in this research are far larger than any other human DNA mixture data known to the authors, it is important to note that these data cannot be immediately placed into a feature vector for machine learning. A typical forensic laboratory would manually perform stutter calculations, but would use an analytical threshold (often set to 50 rfu) instead of relying on an approach such as LSST. While machine learning represents a novel solution to the problems of both NOC estimation and mixture deconvolution as a whole, of equal importance is the ability to extract as much high-value information from the initial sea of data as is possible. Maximizing the amount of signal acquired from the noise of an electropherogram seems to us an additional low-hanging fruit for the deconvoluting multi-contributor DNA samples. Indeed, LSST was one such attempt on our part. While the threshold itself has only modest predictive value as a machine learning feature, the overall process modified peak heights in such a way that many other features were indirectly affected for the better; preliminary findings suggest that a machine learning approach unaccompanied by LSST performs less accurately than does an LLST-enabled model.

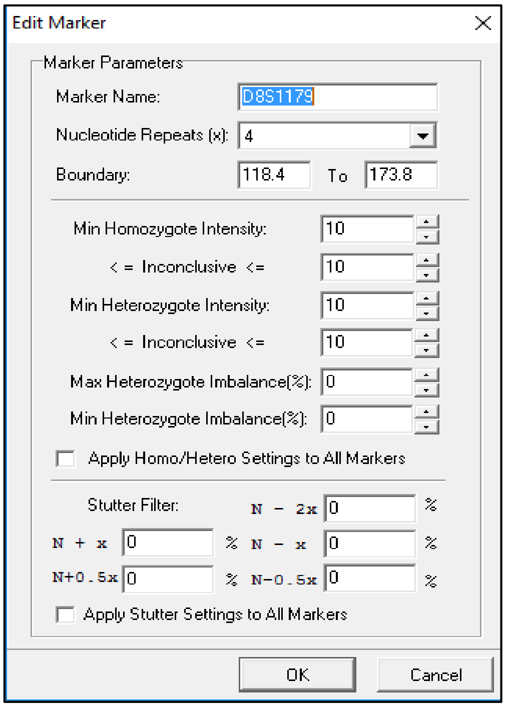
Conclusion

We have presented a new and potentially highly valuable means of deconvolution of DNA mixtures including a method to estimate the number of contributors in a sample of DNA.

Achieving high classification accuracy, especially with complex mixtures containing many contributors, is a vital prerequisite to full mixture deconvolution, which we believe is also well-suited for a machine learning approach. Even with a basic model that does not incorporate dropout, we have seen a non-incremental increase in such accuracy. A key research goal for the field of forensic science, then, may be the subsequent evaluation of machine learning algorithms as tools for DNA mixture deconvolution, as well as the computational extraction of high-value information during the deconvolution process.

Additionally, of great interest to the community may be a formal evaluation of a machine learning-derived model alongside other top performers in a variety of laboratories operating under a variety of conditions; while model validation via testing data performance was a vital component of this study, the potential for poor model generalization still exists if the underlying training data set poorly reflects the reality of DNA mixtures. While the dataset itself is massive enough to allay such concerns, any novel approach in a scientific discipline connected to law and court-based proceedings is likely to face increased levels of scrutiny and mistrust, and should be held to extremely high standards. Lab-based validation is therefore an important subsequent stage in this research.

Tables and Figures



Edit Marker

Marker Parameters

Marker Name: D8S1179

Nucleotide Repeats (x): 4

Boundary: 118.4 To 173.8

Min Homozygote Intensity: 10

< = Inconclusive <= 10

Min Heterozygote Intensity: 10

< = Inconclusive <= 10

Max Heterozygote Imbalance(%): 0

Min Heterozygote Imbalance(%): 0

□ Apply Homo/Hetero Settings to All Markers

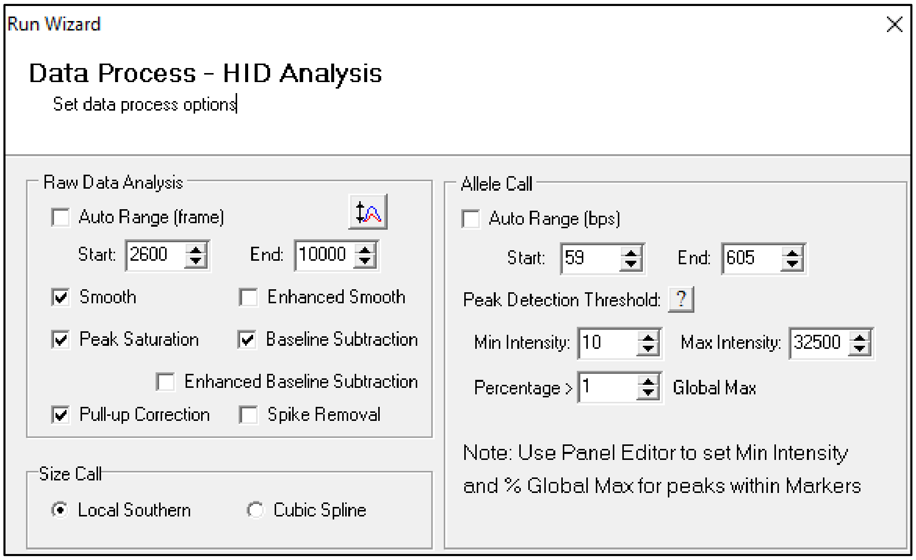
Stutter Filter: N - 2x 0 %

N + x 0 % N - x 0 %

N+0.5x 0 % N-0.5x 0 %

□ Apply Stutter Settings to All Markers

Figure 1: Marker settings for all markers within a panel. Note that there is no stutter filter applied, and that exceedingly low values were used for homozygote and heterozygote intensities and imbalance.



Run Wizard

Data Process - HID Analysis

Set data process options

Raw Data Analysis

□ Auto Range (frame)

Start: 2600 End: 10000

☑ Smooth □ Enhanced Smooth

☑ Peak Saturation ☑ Baseline Subtraction

□ Enhanced Baseline Subtraction

☑ Pull-up Correction □ Spike Removal

Size Call

● Local Southern ○ Cubic Spline

Allele Call

□ Auto Range (bps)

Start: 59 End: 605

Peak Detection Threshold: ?

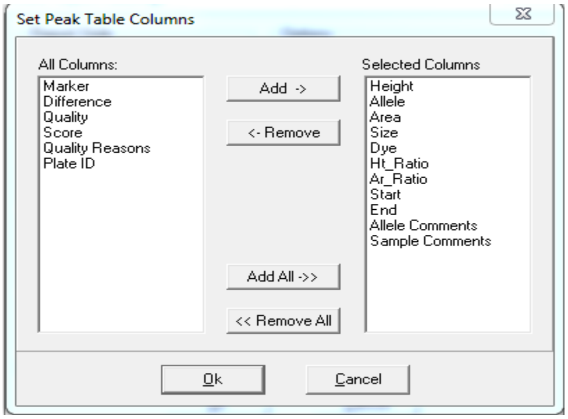
Min Intensity: 10 Max Intensity: 32500

Percentage > 1 Global Max

Note: Use Panel Editor to set Min Intensity and % Global Max for peaks within Markers

Figure 2: Analysis settings for GeneMarker projects.

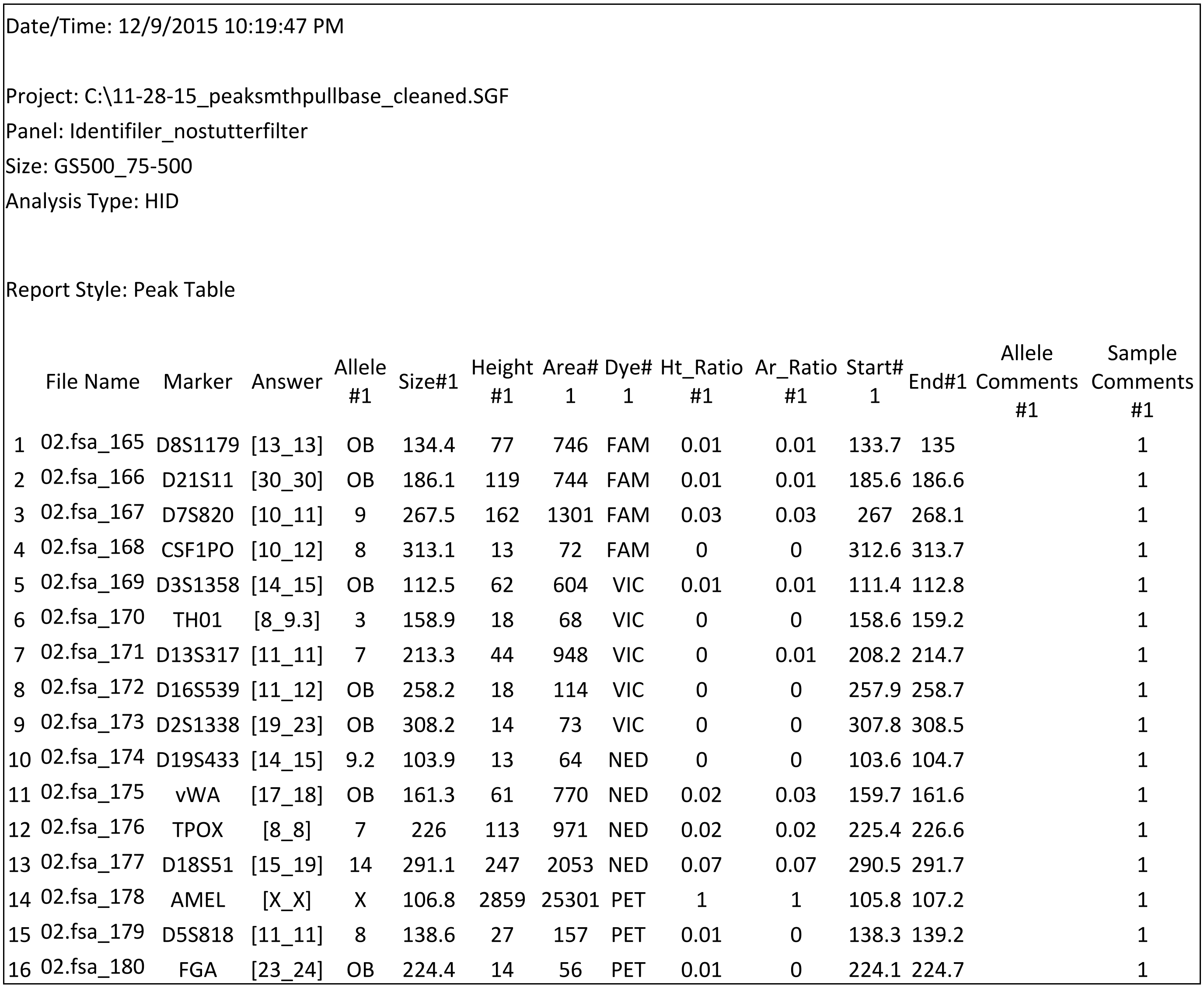
This resource was prepared by the author(s) using Federal funds provided by the U.S. Department of Justice. Opinions or points of view expressed are those of the author(s) and do not necessarily reflect the official position or policies of the U.S. Department of Justice.


Set Peak Table Columns

All Columns:
Marker
Difference
Quality
Score
Quality Reasons
Plate ID

Selected Columns
Height
Allele
Area
Size
Dye
Ht_Ratio
Ar_Ratio
Start
End
Allele Comments
Sample Comments

Figure 3: Ordered columns representing the required data and order of columns to ensure properly formatted exportation of data for feature selection


Date/Time: 12/9/2015 10:19:47 PM

Project: C:\11-28-15_peaksmthpullbase_cleaned.SGF
Panel: Identifiler_nostutterfilter
Size: GS500_75-500
Analysis Type: HID

Report Style: Peak Table

	File Name	Marker	Answer	Allele #1	Size#1	Height #1	Area# 1	Dye# 1	Ht_Ratio #1	Ar_Ratio #1	Start# 1	End#1	Allele Comments #1	Sample Comments #1
1	02.fsa_165	D8S1179	[13_13]	OB	134.4	77	746	FAM	0.01	0.01	133.7	135		1
2	02.fsa_166	D21S11	[30_30]	OB	186.1	119	744	FAM	0.01	0.01	185.6	186.6		1
3	02.fsa_167	D7S820	[10_11]	9	267.5	162	1301	FAM	0.03	0.03	267	268.1		1
4	02.fsa_168	CSF1PO	[10_12]	8	313.1	13	72	FAM	0	0	312.6	313.7		1
5	02.fsa_169	D3S1358	[14_15]	OB	112.5	62	604	VIC	0.01	0.01	111.4	112.8		1
6	02.fsa_170	TH01	[8_9.3]	3	158.9	18	68	VIC	0	0	158.6	159.2		1
7	02.fsa_171	D13S317	[11_11]	7	213.3	44	948	VIC	0	0.01	208.2	214.7		1
8	02.fsa_172	D16S539	[11_12]	OB	258.2	18	114	VIC	0	0	257.9	258.7		1
9	02.fsa_173	D2S1338	[19_23]	OB	308.2	14	73	VIC	0	0	307.8	308.5		1
10	02.fsa_174	D19S433	[14_15]	9.2	103.9	13	64	NED	0	0	103.6	104.7		1
11	02.fsa_175	vWA	[17_18]	OB	161.3	61	770	NED	0.02	0.03	159.7	161.6		1
12	02.fsa_176	TPOX	[8_8]	7	226	113	971	NED	0.02	0.02	225.4	226.6		1
13	02.fsa_177	D18S51	[15_19]	14	291.1	247	2053	NED	0.07	0.07	290.5	291.7		1
14	02.fsa_178	AMEL	[X_X]	X	106.8	2859	25301	PET	1	1	105.8	107.2		1
15	02.fsa_179	D5S818	[11_11]	8	138.6	27	157	PET	0.01	0	138.3	139.2		1
16	02.fsa_180	FGA	[23_24]	OB	224.4	14	56	PET	0.01	0	224.1	224.7		1

Figure 4: Example GeneMarker report

Marker	Dye	Marker Start Size	Marker End Size	Pre-locus threshold start	Pre-locus threshold end	Post-locus threshold start	Post-locus threshold end
D8S1179	FAM	119	173.2	**110.6**	118	174.2	181.6
D21S11	FAM	182.6	244.1	174.2	181.6	245.1	250
D7S820	FAM	251	295.4	245.1	250	296.4	299.5
CSF1PO	FAM	300.5	345.5	296.4	299.5	346.5	349.6
D3S1358	VIC	107.6	143.5	93.3	106.6	144.5	157.8
THO1	VIC	158.8	201.7	144.5	157.8	202.7	211.4
D13S317	VIC	212.4	248.1	202.7	211.4	249.1	251.1
D16S539	VIC	252.1	296.4	249.1	251.1	297.4	301.7
D2S1338	VIC	302.7	362.3	297.4	301.7	363.3	367.6
D19S433	NED	97.6	139.4	87.9	96.6	140.4	149.1
vWA	NED	150.1	210.4	140.4	149.1	211.4	216.7
TPOX	NED	217.7	253.6	211.4	216.7	254.6	260.7
D18S51	NED	261.7	344.2	254.6	260.7	317	342
Amelogenin	PET	106.3	112	89.5	105.3	113	128.8
D5S818	PET	129.8	175.3	113.0	128.8	176.3	209.3
FGA	PET	210.3	354.4	179.3	209.3	285	310

Table 1: Locus and sample specific threshold bounds for each locus within the AmpFLSTR Identifiler PCR Amplification Kit (Life Technologies).

Classifier	Hyperparameter settings
KNN	n_neighbors=10, p=2, metric='minkowski'
CART	min_samples_leaf=100, random_state=0
Logistic regression	penalty='l2', C=0.1, random_state=0, solver='lbfgs', verbose=1, multi_class='multinomial', warm_start=True, max_iter=1000
MLP	default Weka hyperparameters only
SVM (linear)	C=0.01, random_state=0
SVM (kernel)	kernel='rbf', random_state=0, gamma='auto', C=2.0, probability=True, decision_function_shape='multinomial'

Table 2: Hyperparameters for all machine learning algorithms used in the study.

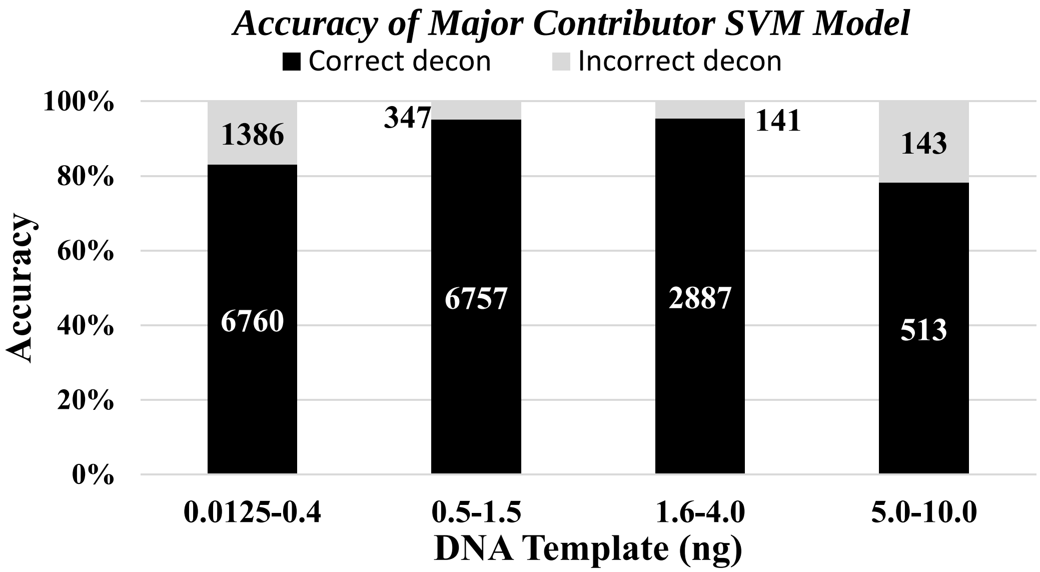


Figure 5: Accuracy of Major Contributor SVM Model.

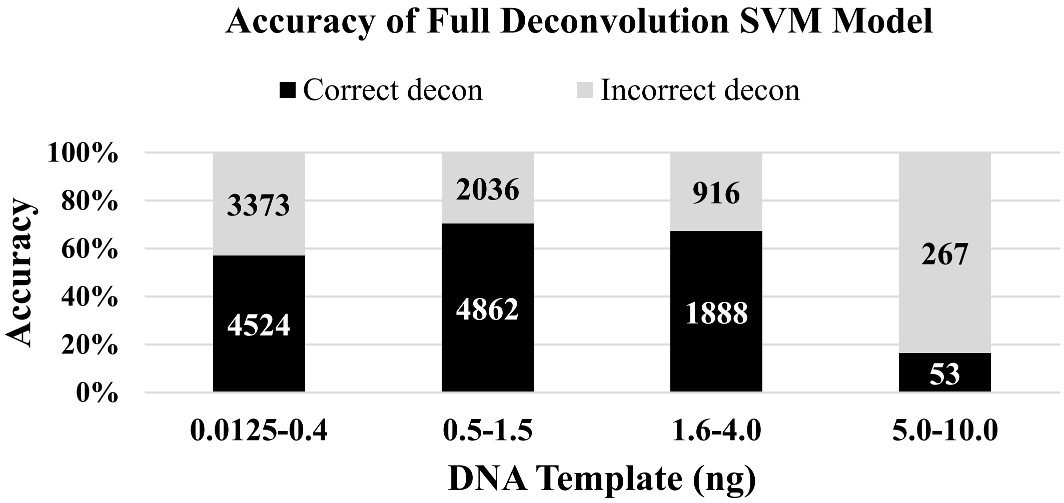


Figure 6: Accuracy of Full Deconvolution SVM Model.

D_{KL}	features
1.4625	sample-wide peak count
1.2436	maximum number of contributors
0.9964	minimum number of contributors
0.8251	template DNA amplified
0.4932	locus-specific peak count
0.0914	locus- and allele-specific threshold
0.0823	minimum observed peak height
0.0375	maximum observed peak height
0	size of locus

Table 3: Kullback–Leibler divergence of nine candidate features

classifier	NOC	precision	recall	f1-score	training/testing accuracies
KNN	1	0.94	0.97	0.96	0.944 / 0.935
	2	0.96	0.95	0.96	
	3	0.87	0.92	0.89	
	4	0.91	0.80	0.85	
CART	1	0.97	0.96	0.97	0.941 / 0.939
	2	0.96	0.96	0.96	
	3	0.86	0.93	0.90	
	4	0.90	0.78	0.84	
Logistic regression	1	0.94	0.95	0.95	0.920 / 0.927
	2	0.95	0.95	0.95	
	3	0.88	0.90	0.89	
	4	0.89	0.82	0.85	
MLP	1	0.96	0.98	0.97	0.946 / 0.936
	2	0.97	0.97	0.97	
	3	0.91	0.92	0.92	
	4	0.84	0.87	0.86	
SVM (linear)	1	0.93	0.94	0.93	0.880 / 0.890
	2	0.90	0.92	0.91	
	3	0.79	0.85	0.82	
	4	0.95	0.72	0.82	
SVM (kernel)	1	0.95	0.98	0.97	0.953 / 0.951
	2	0.97	0.96	0.97	
	3	0.91	0.94	0.92	
	4	0.94	0.85	0.89	

Table 4: Summary metrics for six machine learning algorithm learned models for NOC classification. Training and testing accuracies are used to evaluate model convergence, and represent total accuracy across all four classes.

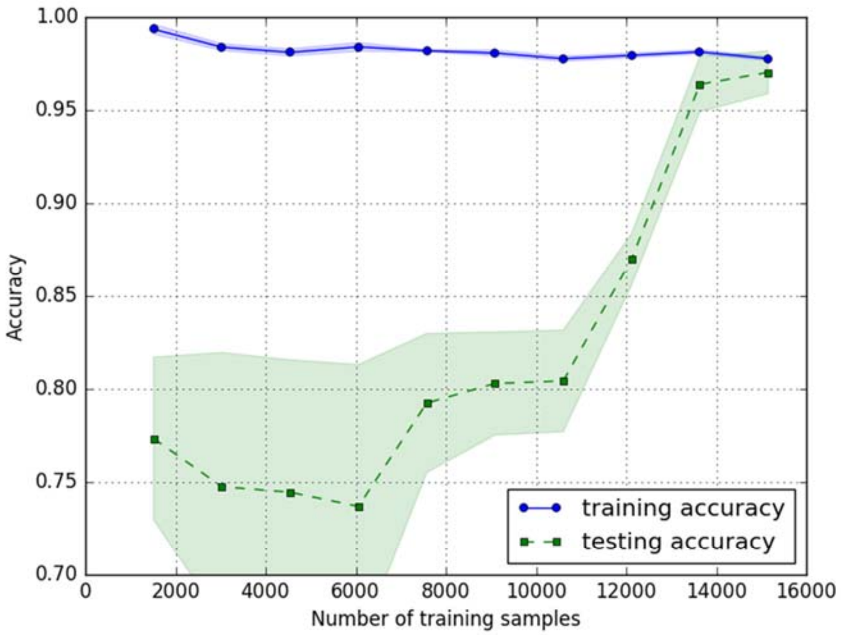


Figure 7: Learning curve for a NOC estimation model derived from a kernel-based support vector machine. Training accuracy: 0.953; testing accuracy: 0.951.

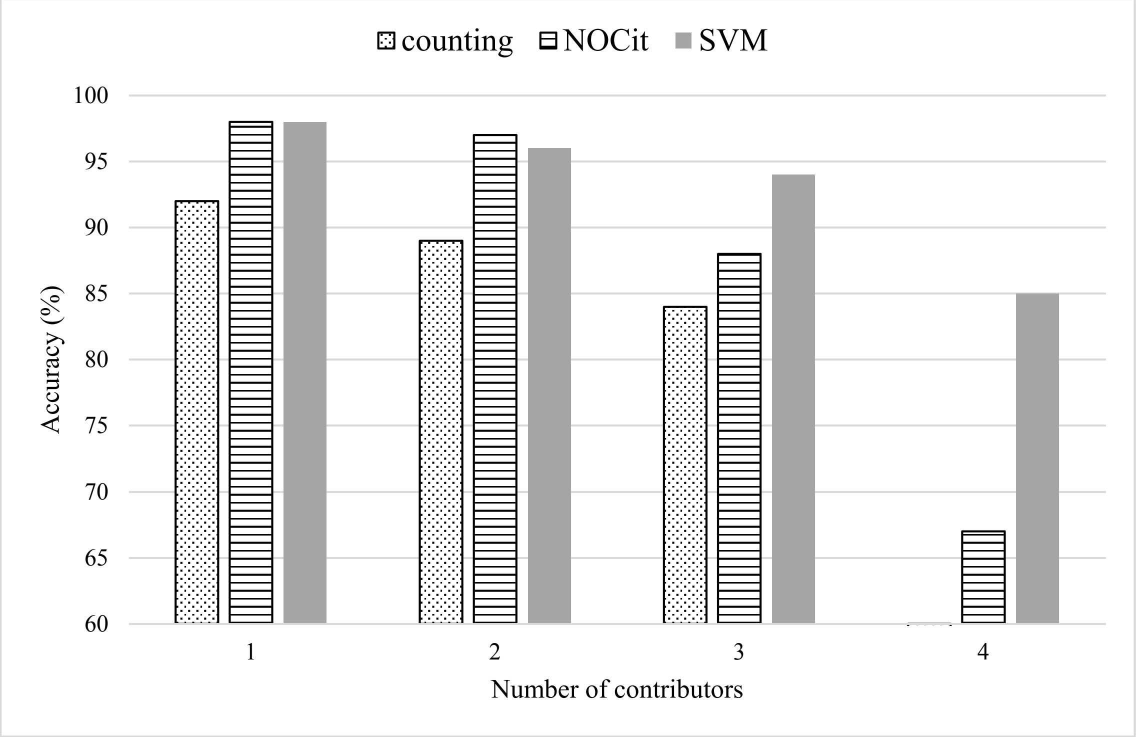


Figure 8: Accuracy rates for several NOC estimation models. "Counting" refers to the MAC method described earlier in this paper. "NOCit" refers to the NOCit software and reported results (Swaminathan et al. 2015). "SVM" refers to the model created by a kernel-based support vector machine. The classification accuracy for MAC with four contributors is less than 30%, and therefore not present here due to the histogram's axis range.

This resource was prepared by the author(s) using Federal funds provided by the U.S. Department of Justice. Opinions or points of view expressed are those of the author(s) and do not necessarily reflect the official position or policies of the U.S. Department of Justice.

Contributor Number	Training Set		Testing Set	
	Sample Count	Percentage	Sample Count	Percentage
1	378	26.9%	94	26.8%
2	607	43.3%	154	43.9%
3	276	19.7%	74	21.1%
4	142	10.1%	29	8.3%
Total	1403		351	

Table 5: Sample sets used for the training and testing of NOC classification models created by machine learning algorithms. Stratified sampling was used to ensure a proportional representation of each contributor class in the two distinct sample sets.

DNA Template Amplified (ng)	Training Set			Testing Set		
	Sample-Locus Instance Count	Sample Count	Percent Representation	Sample-Locus Instance Count	Sample Count	Percent Representation
0.0125	100	6	0.4%	0	0	0.0%
0.025	153	10	0.7%	31	2	0.6%
0.05	235	15	1.1%	31	2	0.6%
0.0625	1152	72	5.1%	253	16	4.6%
0.075	287	18	1.3%	64	4	1.1%
0.1	303	19	1.4%	159	10	2.8%
0.125	1211	76	5.4%	319	20	5.7%
0.15	2800	175	12.5%	694	43	12.3%
0.2	768	48	3.4%	128	8	2.3%
0.25	1360	85	6.1%	416	26	7.4%
0.3	224	14	1.0%	112	7	2.0%
0.4	528	33	2.4%	160	10	2.8%
0.5	3216	201	14.3%	848	53	15.1%
0.6	96	6	0.4%	32	2	0.6%
0.7	272	17	1.2%	48	3	0.9%
0.8	336	21	1.5%	48	3	0.9%
1	3938	246	17.5%	944	59	16.8%
1.2	96	6	0.4%	48	3	0.9%
1.3	80	5	0.4%	16	1	0.3%
1.5	320	20	1.4%	48	3	0.9%
1.6	48	3	0.2%	16	1	0.3%
1.7	128	8	0.6%	32	2	0.6%
2	1552	97	6.9%	336	21	6.0%
2.5	192	12	0.9%	64	4	1.1%
3	384	24	1.7%	144	9	2.6%
3.3	96	6	0.4%	16	1	0.3%
3.5	96	6	0.4%	32	2	0.6%

4	1443	90	6.4%	304	19	5.4%
5	192	12	0.9%	48	3	0.9%
6	256	16	1.1%	64	4	1.1%
7	320	20	1.4%	112	7	2.0%
7.5	48	3	0.2%	16	1	0.3%
8	96	6	0.4%	0	0	0.0%
9	96	6	0.4%	32	2	0.6%
10	32	2	0.1%	0	0	0.0%
Total	22454	1404		5615	351	

Table 6: Sampling Structure for the training and testing data sets based on the amount of DNA template amplified (ng).

Contributor #	% Correct	Incorrect Count	Correct Count	Over-estimate	Under-estimate
1	100%	0	94	0	0
2	98.1%	3	151	1	2
3	94.6%	4	70	2	2
4	100%	0	30	0	0

Table 7: NOC classification model accuracy rates when used to classify samples from the testing set.

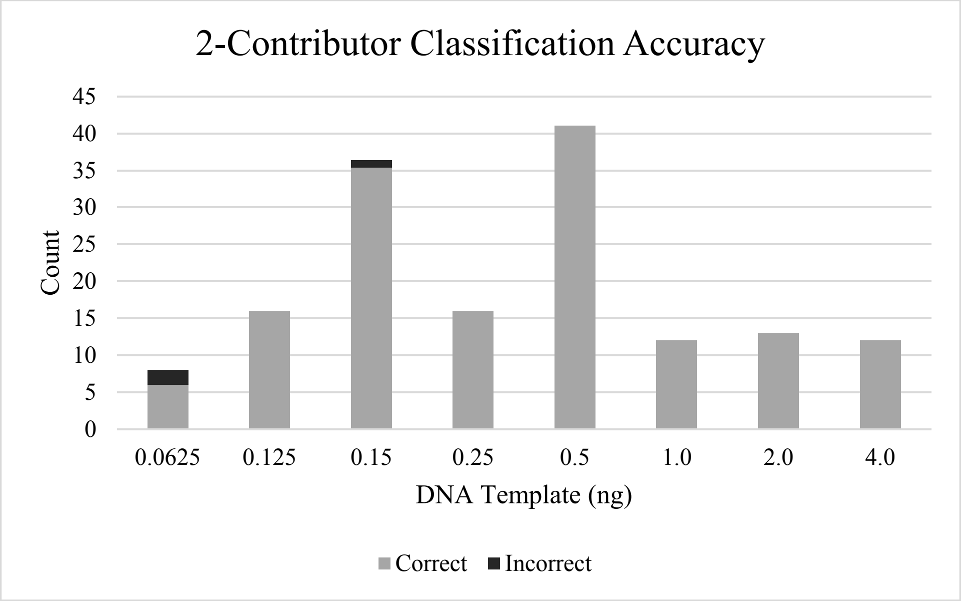


Figure 9: Accuracy rates for NOC classification of two-person mixtures at various template amounts.

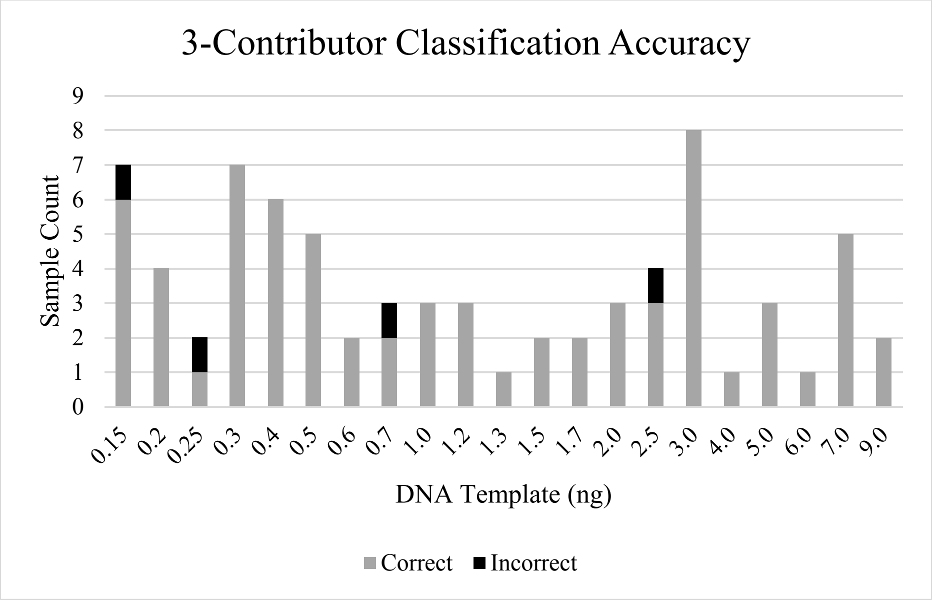


Figure 10: Accuracy rates for NOC classification of three-person mixtures at various template amounts.

Contributor Number	Sample Number	DNA Template (ng)	Ratio of Contributors	NOC model estimate
2	1	0.15	2 to 1	3
	2	0.0625	1 to 1	3
	3	0.0625	1 to 19	1
3	1	0.15	1 to 1 to 3	2
	2	0.7	3 to 1 to 2	4
	3	2.5	12 to 1 to 1	4
	4	0.25	1.5 to 3 to 1	2

Table 8: Summary of misclassifications by the NOC classification model.

References

Benschop, C. C., Haned, H., de Blaeij, T. J., Meulenbroek, A. J., & Sijen, T. (2012). Assessment of mock cases involving complex low template DNA mixtures: a descriptive study. *Forensic Science International: Genetics*, *6*(6), 697-707.

Butler, J. M. (2014). *Advanced topics in forensic DNA typing: interpretation*. Academic Press.

Buckleton, J. S., Curran, J. M., & Gill, P. (2007). Towards understanding the effect of uncertainty in the number of contributors to DNA stains. *Forensic Science International: Genetics*, *1*(1), 20-28.

Clayton, T. M., Whitaker, J. P., Sparkes, R., & Gill, P. (1998). Analysis and interpretation of mixed forensic stains using DNA STR profiling. *Forensic Science International*, *91*(1), 55-70.

Cho, S. B., & Won, H. H. (2003, January). Machine learning in DNA microarray analysis for cancer classification. In *Proceedings of the First Asia-Pacific bioinformatics conference on Bioinformatics 2003-Volume 19* (pp. 189-198). Australian Computer Society, Inc.

Duygulu, P., Barnard, K., de Freitas, J. F., & Forsyth, D. A. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Computer Vision—ECCV 2002* (pp. 97-112). Springer Berlin Heidelberg.

Egeland, T., Dalen, I., & Mostad, P. F. (2003). Estimating the number of contributors to a DNA profile. *International journal of legal medicine*, *117*(5), 271-275.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, *11*(1), 10-18.

Haned, H., Pene, L., Lobry, J. R., Dufour, A. B., & Pontier, D. (2011a). Estimating the number of contributors to forensic DNA mixtures: does maximum likelihood perform better than maximum allele count?. *Journal of forensic sciences*, *56*(1), 23-28.

Haned, H., Pène, L., Sauvage, F., & Pontier, D. (2011b). The predictive value of the maximum likelihood estimator of the number of contributors to a DNA mixture. *Forensic Science International: Genetics*, *5*(4), 281-284.

Jurafsky, D., & Martin, J. H. (2014). *Speech and language processing*. Pearson.

Kidd, K., Rajeevan, H., Osier, M. V., Cheung, K., Deng, H., Druskin, L., Heinzen, R, Kidd, J.R., Stein, S., Pakstis, A.J., Tosches, N.P., Yeh, C.C., & Miller, P.L. (2003, January). ALFRED-the ALlele FREquency database-an update. In *American Journal of Physical Anthropology* (pp. 128-128). DIV JOHN WILEY & SONS INC, 605 THIRD AVE, NEW YORK, NY 10158-0012 USA: WILEY-LISS.

Paoletti, D. R., Doom, T. E., Krane, C. M., Raymer, M. L., & Krane, D. E. (2005). Empirical analysis of the STR profiles resulting from conceptual mixtures. *Journal of forensic sciences*, *50*(6), 1361.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825-2830.

Perez, J., Mitchell, A. A., Ducasse, N., Tamariz, J., & Caragine, T. (2011). Estimating the number of contributors to two-, three-, and four-person mixtures containing DNA in high template and low template amounts. *Croatian medical journal*, *52*(3), 314-326.

Quinlan, J. R. (2014). *C4. 5: programs for machine learning*. Elsevier.

Rajeevan, H., Osier, M. V., Cheung, K. H., Deng, H., Druskin, L., Heinzen, R., Kidd, J.R., Stein, S., Pakstis, A.J., Tosches, N.P. and Yeh, C.C. (2003). ALFRED: the ALelle FREquency Database. Update. *Nucleic Acids Research*, *31*(1), 270-271.

Rajeevan, H., Cheung, K.H., Gadagkar, R., Stein, S., Soundararajan, U., Kidd, J.R., Pakstis, A.J., Miller, P.L. and Kidd, K.K. (2005). ALFRED: an allele frequency database for microevolutionary studies. *Evolutionary Bioinformatics*, *1*.

Rajeevan, H., Soundararajan, U., Kidd, J. R., Pakstis, A. J., & Kidd, K. K. (2011). ALFRED: an allele frequency resource for research and teaching. *Nucleic acids research*, gkr924.

Swaminathan, H., Grgicak, C. M., Medard, M., & Lun, D. S. (2015). NOCIt: A computational method to infer the number of contributors to DNA samples analyzed by STR genotyping. *Forensic Science International: Genetics*, *16*, 172-180.

SWGDAM interpretation guidelines for autosomal STR typing by forensic DNA testing laboratories. Available from: http://www.fbi.gov/about-us/lab/codis/swgdam-interpretation-guidelines. Accessed: June 7, 2011.