Statistical Approaches to Automated Groove Engraved Area Identification in 3D Bullet Land Scans

Kiegan Rice *
Department of Statistics, Iowa State University
and
Nathaniel Garton
Department of Statistics, Iowa State University
and
Ulrike Genschel
Department of Statistics and CSAFE, Iowa State University
and
Heike Hofmann
Department of Statistics and CSAFE, Iowa State University

October 2, 2019

Abstract

Abstract will be written via the Gelman method once the introduction is OK'd.

Keywords: 3 to 6 keywords, that do not appear in the title

*The authors gratefully acknowledge ...

TO DO:

- Abstract

1 Introduction

Forensic pattern analysis aims to address the same-source problem: whether two impressions were generated by the same object. One of the most significant aspects of the same-source problem in firearms analysis is the determination of whether two bullets were fired through the same gun barrel. The evidence used in visual pattern comparison of bullets are striation marks, which are engraved on the bullet by micro imperfections in the barrel.

For barrels with traditional (i.e., not polygonal) rifling, striation marks found on land engraved areas (LEAs) are the primary evidence used. LEAs bear marks engraved by alternating sections of the barrel, and are considered to be areas which will bear unique patterns of striation marks which are reproduced on any bullet fired through the same barrel (1). Figure 1 depicts lands inside a barrel, as well as land engraved areas on a fired bullet.

The recent application of high resolution 3D scanning technology to bullet LEAs coupled with concerns about the objectivity of visual pattern comparison have motivated the development of several image-analysis algorithms which aim to complete automated, quantitative analyses of bullet evidence (see 2; 3; 4; 5; 6; 7; 8). The data used in such algorithms are high resolution 3D scans of LEAs, such as that pictured in Figure 2 (top).

Bullets can be algorithmically compared to one another by completing individual LEA-to-LEA image comparisons of each LEA scan from one bullet to each LEA scan from a second bullet. Each LEA-to-LEA pairing compares the patterns of striation marks engraved on the LEA, which can be extracted from a 3D scan as a 2D signature which details a pattern of peaks and valleys representing striation marks.

Signatures are extracted by obtaining a horizontal slice of the 3D scan, called a profile, followed by removing the overall bullet curvature which dominates the data structure of each profile. Figure 2 depicts the process of translating a 3D LEA scan to a 2D LEA signature.

Currently accepted best practice for collecting 3D LEA scans involves capturing portions of the neighboring groove engraved areas (GEAs), introducing a secondary data structure on the edges of the LEA scan which is not attributed to the pattern of interest: striation marks left on the LEA.

In order to accurately represent the striation pattern as a 2D LEA signature, the extraneous data structure introduced by GEA data must be identified and removed. The identification of GEA data, while quite straightforward for the human visual system, is a difficult process to automate using computer vision techniques. A prior GEA identification method was detailed as one step of the LEA matching algorithm proposed in Hare et al. (8). The method, "rollapply", uses data smoothing to reduce noise measured in data and subsequently searches for the local minima closest to the edges of each LEA. It often fails to adequately identify large scale structural changes in favor of smaller structural anomalies.

This would replace the paragraph below: This paper describes two methods for automated identification and removal of GEA data. We first describe the data structure and source of test set bullets, followed by a description of methods for GEA data identification. Both methods begin by using an adapted version of a robust statistical modeling technique to remove bullet curvature. Two distinct approaches are then applied to separate GEA data from LEA data using statistical techniques. Finally, a performance assessment of the two approaches is completed by comparing performance of each LEA matching algorithm as it functions in the automated analysis pipeline for three different bullet test sets.

We describe here two methods for automated identification and removal of GEA data. First, we use an adapted version of a robust statistical modeling technique to remove bullet curvature. Then, we describe two methods to separate GEA data from LEA data using statistical techniques. We then assess performance of the two approaches on three separate test sets of bullets by comparing performance of the Hare et al. (8) LEA matching algorithm when each proposed method is applied within the automated analysis pipeline.

2 Data Source

We use high resolution 3D scans of bullet LEAs. Scans were captured at Iowa State University's High Resolution Microscopy Facility on a Sensofar confocal light microscope

at 20x magnification resulting in a resolution of 0.645 microns per pixel. Scans are stored as x3p files, conforming to the ISO5436-2 standard (9). x3p is the industry standard format for capture and storage of 3D microscopic topography of bullets. Objects are stored digitally as a 2-dimensional matrix with (x, y) locations corresponding to locations on the physical object; a relative height value z is measured and recorded for each (x, y) location.

All six LEAs from each bullet were captured for bullets from three separate test sets: Hamby set 44, Phoenix PD set, and Houston-test set. Each test set contains a combination of "known" bullets, identified as originating from a particular gun barrel, and "unknown" bullets. In a closed test set, each unknown bullet originates from one of the known barrels. In an open test set, unknown bullets can originate from either a known barrel or a barrel outside of the set entirely.

Hamby set 44 is a closed test set consisting of 35 bullets fired from ten consecutively rifled Ruger P85 barrels. There are two known bullets for each of the ten barrels, as well as 15 additional questioned bullets. Every LEA was scanned for each of the 35 fired bullets in the set, producing data for 210 individual land engraved areas. Two lands – Barrel 9, Bullet 2, Land 3 and Unknowns, Bullet L, Land 5 – were removed from consideration due to "tank rash". Tank rash results from a bullet striking the bottom of a water recovery tank after exiting the barrel, thereby creating marks on the land that are not due to the contact with the barrel.

The Phoenix PD set is a closed test set consisting of 33 bullets fired from 8 barrels more information on the type of barrels?. There are three known bullets for each of the eight barrels, as well as 9 additional questioned bullets. Every LEA was scanned for each of the 33 fired bullets, producing a total of 198 individual land engraved areas.

The Houston-test set is an open test set consisting of 69 bullets fired from more than 10 barrels more information on the type of barrels?. Reference to where more info can be found. There are at least three known bullets for each of the ten barrels, with a total of 33 known bullets. There are 24 questioned bullets that were fired from a combination of the 10 known barrels and additional, out-of-set barrels. Every LEA was scanned for each of the 69 fired bullets, producing a total of 414 individual land engraved areas.

A vertical crosscut location was extracted from each scan by identifying an optimal

crosscut using `x3p_crosscut_optimize` in the `bulletxtrctr` package in `R` (10). We calculated a profile by averaging across ten consecutive crosscuts which fall directly to either side of the optimal crosscut identified. In the following, we utilize relative horizontal locations on the profile, x_i, and their relationship with relative height values on the profile, z_i, for data points $i = 1, ..., n$ on each individual profile. The following GEA data identification methods are applied to these averaged profiles, for a total of 820 profiles across the three test sets.

3 Methodology

We propose two methods for separating GEA data from LEA data. The first method, logistic LASSO, uses supervised two-class classification techniques to classify each data point as being part of the LEA or GEA. The competing LEA and GEA stuctures are a natural fit to two-class classification techniques that predict membership in one of two classes. The second approach is Bayesian changepoint analysis, an unsupervised method that seeks to identify points where the distribution of the residuals changes. The structure of the residuals makes the identification of GEAs highly amenable to models similar to those considered in (11), in which the changepoint method was shown to be effective. Moreover, it is reasonable to expect that there are significant differences in features of the data across different types of bullets or firearms. For example, (6) showed that there was significant systematic variation in the width of bullet LEAs depending on firearm manufacturer. Because it is unsupervised, the effectiveness of the Bayesian changepoint method should not depend on the types of bullets available for training a supervised model. Thus, as long as our assumptions about the structure of the data are reasonable, the Bayesian changepoint method may generalize more effectively than a supervised model to unseen bullets. Should add why we started to try Bayesian changepoint analysis, for example good performance in literature or similar data. Both methods require first removing the global structure of the LEA scan, a non-trivial statistical procedure we will describe before further developing each method.

3.1 Global Structure Removal

The first stage of both GEA removal processes requires removing the bullet curvature from each profile. GEAs on the edge of each LEA represent a change in the primary data structure typically characterized by a sharp increase in measured height values, as demonstrated in Figure 2 (middle). Removal of the primary curvature should leave the secondary structure intact – namely, the sharp increase in height values. The increase in values can then be used to separate LEA data from GEA data.

Bullets are subjected to significant pressure in the process of being fired through a gun barrel. Thus, we cannot assume completely circular curvature on bullet LEAs. Further, we cannot model bullet curvature directly as a circular structure. Rather, we use more flexible non-parametric locally weighted regression (LOESS), which can model the large-scale structure of the height values z_i. LOESS is flexible enough to model structural defects within the bullet curvature. However, the flexibility also leaves LOESS susceptible to modeling the secondary structure of the GEA.

Traditional LOESS predicts height values $\widehat{z}_i$ as a function of location x_i by estimating values β_0, β_1, minimizing:

$$\arg\min_{\beta} \sum_{k=1}^{n} w_k(x_i)(z_k - (\beta_0 + \beta_1 x_k))^2,$$

where $w_k(x_i)$ is a weight assigned to each data point x_k based on its proximity to x_i. Weights w_k decrease as distance to x_i increases, so that data points closest to x_i influence the prediction $\widehat{z}_i$ most. Weights $w_k(x_i)$ are defined using a prespecified decreasing function, traditionally a tricube weight. The decreasing nature of the weight function results in predicted $\widehat{z}_i$ values in the GEA data are which are primarily based on other GEA data, and thus do not accurately capture the overall bullet curvature. Failure to correctly estimate and remove bullet curvature causes a mischaracterization of the extracted LEA signature. Finish explanation of how method is applied before switching to robust LOESS? how does it end? Similar to my comment on page 7 bottom.

An alternative approach which mitigates the impact of GEA data is robust LOESS (see (12)), a process which iteratively updates weights $w_k(x_i)$ by multiplying by a robustness weight, δ_k based on the magnitude of each residual $e_k = z_k - \widehat{z}_k$. Larger values of e_k result

in a lower weight for that data point, z_k. The process is as follows:

1. Fit a LOESS model to a LEA profile, predicting height z_i using relative location x_i. Assign robustness weights δ_k of 1 to each data point (x_k, y_k).

2. Obtain predicted height values $\widehat{z}_k$, and corresponding residual values $e_k = z_k - \widehat{z}_k$.

3. Calculate updated robustness weights using residual values e_k:

$$\delta_k \times w_k(x_i) = \left(1 - \left(\frac{e_k}{6 * MAD}\right)^2\right)^2 \times w_k(x_i) \qquad \text{if} \quad \left|\frac{e_k}{6 * MAD}\right| < 1,$$

   where MAD is the median absolute deviation of residuals e_k. δ_k is known as the bisquare function.

4. Repeat steps 1-3 with updated weights at each iteration for m iterations, with 20 iterations as the default.

5. After $m = 20$ iterations of updating the weight vector $\delta_k w_k(x_i)$, fit a LOESS model and obtain residual values e_i for each data point (x_i, z_i).

How do we know when to stop? (How many iterations m?)

Re-weighting data as in Step 3 reduces influence of data points with large absolute residual values. When GEA data are present on a profile, largest residual values will occur in areas where GEA data begin as they present a competing structure with overall LEA curvature.

Figure 3 depicts the impact this iterative re-weighting process has on curvature removal for an example LEA from Hamby set 44. Predictions $\widehat{z}_i$ are updated and more closely follow the primary structure of bullet curvature.

However, this method fails to mitigate GEA data impact when GEA structures are more pronounced. Thus, we make an adaptation to the procedure to function more effectively with these data structures, such as the one in Figure 4. Re-weighting in Step (3) is only applied to *positive* residuals in our adaptation, which iteratively moves the boundary predictions downwards towards the LEA curvature:

1. Fit a LOESS model to a LEA profile, predicting height z_i using relative location x_i. Assign robustness weights δ_k of 1 to each data point (x_k, y_k).

2. Obtain predicted height values $\widehat{z}_k$, and corresponding residual values $e_k = z_k - \widehat{z}_k$.

3. Calculate updated robustness weights using residual values e_k:

$$\delta_k \times w_k(x_i) = \left(1 - \left(\frac{e_k}{6 * MAD}\right)^2\right)^2 \times w_k(x_i) \qquad \text{if} \quad \left|\frac{e_k}{6 * MAD}\right| < 1,$$

   where MAD is the median absolute deviation of residuals e_k. δ_k is known as the bisquare function.

4. Assign updated weights as in Step 3 if $e_k > 0$. Else, leave weights as $w_k(x_i)$.

5. Repeat steps 1-4 with updated weights at each iteration for m iterations, with 20 iterations as the default.

6. After m iterations of updating the weight vector $w_k(x_i)$, fit a LOESS model and obtain residual values e_i for each data point (x_i, z_i). The default number of iterations, $m = 20$, is typically plenty to mitigate the influence of GEA data.

This adapted robust LOESS, more adeptly fits bullet curvature. Figure 4 demonstrates the impact of the adaptation on predicted height values z_i and residual value e_i. Figure 5 depicts the differing levels of impact our adjustment has on each bullet test set. Once I have more barrel information, I could go into a little more detail here about differing barrel types/ammo types or something, but no more than a sentence to bring the point home.

Once the adapted robust LOESS procedure has been applied to a LEA profile (Did you describe this anywhere? That is what is described above; the process applied to LEA profiles.), the resulting residuals e_i follow a reliable pattern: small residuals closer to zero in locations associated with LEA data, and sharply increasing, larger residuals in locations associated with GEA data. The resulting residual structure lends itself to changepoint analysis techniques to separate the two structures more effectively.

The subsequent GEA identification methods are based on residuals e_i calculated from adapted robust LOESS fit to the global structure of the profile. Both methods result in

"shoulder location predictions", predictions for the locations (x_L, x_R) at which LEA data end and GEA data begin on the left and right side of the profile, respectively.

3.2 Logistic LASSO

This next section is hard to follow:

Shoulder location can be predicted by first classifying each residual point as one of two classes ("LEA" or "GEA"), and subsequently gathering the range of values classified as "LEA" points.

Classification into "LEA" or "GEA" is first approached by a process of feature engineering based on adapted robust LOESS residuals. While the residuals e_i should demonstrate differing patterns of magnitude, residuals alone are not sufficient to classify data with high accuracy. Add reason already here.

Figure 6 demonstrates that each test set has distinct patterns of striation depth. Thus, standardization of features is imperative for transferability of fitted model parameters. The LEA scan context requires some non-traditional standardization practices.

For example, consider the distribution of residual values e_i resulting from adapted robust LOESS. There is reason to believe that the distribution will be quite skewed, which means a standard deviation will not be a good proxy for the spread of the distribution. Thus, rather than the standard deviation, we consider instead the standard deviation of residual values from the middle 50% of x_i values present in each profile. This alternative acts as a proxy for the depth of striae on each LEA, with higher standard deviations found for lands with deeper striae. Standardizing residual values by this proxy puts all residuals on a comparable scale.

For variables which represent differences in the x direction, such as depth from the center of a scan, values will be mapped to a $(0, 1)$ range.

The full list of standardized features is as follows:

> **Standardized residuals** e_i (`rlo_resid_std`): Robust LOESS residual value e_i, standardized by dividing by standard deviation of residual values from middle 50% of x_i values.

Standardized residuals squared, e_i^2 (`(rlo_resid_std)^2`): Squared term of `rlo_resid_std`.

Side of scan (`side`): Whether data point is to left or right of median x_i value.

Standardized depth from scan center (`depth_std`): Distance of data point from median x_i value, standardized by dividing by maximum x_i value (a proxy for the range of x values).

Side of scan, depth interaction (`side:depth_std`): Interaction between `side` and `depth_std` variables.

Left x_i **intercept** (`xint1_std`): Predicted location x_i at which adapted robust LOESS crosses y axis on left side of profile, standardized by dividing by maximum x_i value (a proxy for the range of x values).

Right x_i **intercept** (`xint2_std`): Predicted location x_i at which adapted robust LOESS crosses y axis on right side of profile, standardized by dividing by maximum x_i value (a proxy for the range of x values).

Range of local residuals e_i (`range_50_std`): Range of residual values e_i within a 50-point window (x_{i-25}, x_{i+25}) around data point x_i, standardized by dividing by standard deviation of residual values from middle 50% of x_i values.

Number of local missing values (`numNA_50`): Number of missing values within a 50-point window (x_{i-25}, x_{i+25}) around data point x_i.

Magnitude indicator for residual e_i (`ind_2mad`): Indicator of whether `rlo_resid` is greater than `2*MAD(rlo_resid)`, where MAD is the median absolute deviation of residual values e_i for an entire profile.

Number of local positive e_i **values** (`numpos_50`): Number of positive residual values within a 50-point window (x_{i-25}, x_{i+25}) around data point x_i.

Outside edges indicator (`ind_edges`): Indicator of whether data point is to the left of `xint1` or to the right of `xint2`. Values between `xint1` and `xint2` receive a value of 0, while values on the outside of the two values receive a value of 1.

Examples of the distributions of some of these features can be seen in Figure 7. Where does the number 50 come from in Figure 7 bottom graph?

A logistic LASSO model, a form of penalized regression, was fit using the developed features. LASSO parameter values for p features were estimated by identifying:

$$\hat{\beta}_\lambda = \overset{\arg\min}{\beta \in \mathbb{R}^p} \left\{ (Y - X\beta)'(Y - X\beta) + \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

where X is a matrix with n rows, and a column for each data feature described above. $\hat{\beta}$ is a vector of estimated parameter values associated with each data feature, and Y is the vector of length n of response class values, either a 1 for GEA class, or a 0 for LEA class. This is quite the leap from the ending of the previous sentence: LASSO adds a penalty to the traditional ordinary least squares minimization problem, and uses a tuning parameter λ (13). Predicted probabilities of membership in the GEA class can be calculated as: I would first describe how to use the features as you do further down.

$$\widehat{p_i} = \frac{\exp\{X\hat{\beta}_\lambda\}}{1 + \exp\{X\hat{\beta}_\lambda\}}.$$

A cross-validated logistic LASSO model was fit using the `cv.glmnet` function in the `glmnet` package in `R` (14). Parameter values from the model with λ_{1se} were used. λ_{1se}, a standard when using LASSO, is the tuning parameter which results in the simplest model that still has cross-validation error within one standard deviation of the best model.

The resulting model uses each of the data features listed above along with pairwise interactions for each of them. The $\hat{\beta}$ vector was estimated using Hamby set 44 data. Estimated parameter values were used to calculate predicted values of GEA membership between 0 and 1; the closer to 1, the higher the probability of membership in the "GEA" class.

Two-class classification techniques traditionally employ a cutoff for predicted probabilities and assign predicted class membership using that cutoff; i.e., values above a certain cutoff are classified as part of the "GEA" class, and values below the cutoff are classified as part of the "LEA" class. An equal error rate is typically used for this purpose. Equal error rate is defined as the cutoff at which sensitivity (true positive rate) and specificity (true negative rate) are equal. However, since LEA scans consist of a majority of LEA data by

nature, significant class imbalance in our response variable ("GEA" vs. "LEA") dictates a change in procedure for assigning a cutoff for class membership. Employing an equal error rate cutoff to predict class membership would result in a higher number of false positives, here predicting data which is part of the "LEA" class as "GEA" data. We employ an equal *number of errors* rate rather than an equal error rate to ameliorate this imbalance.

The pipeline of this method for shoulder location identification is as follows:

1. Use adapted robust LOESS procedure to remove bullet curvature and obtain residual values e_i.

2. Extract data features based on x_i locations and residual height values e_i from Step 1.

3. Use fit parameter values from trained logistic LASSO model to calculate probabilities of membership in GEA class.

4. Apply cutoff of $\hat{p} = .34$: classify higher probabilities as GEA data points, and lower probabilities as LEA data points.

5. Identify the minimum x_i value which is predicted as a member of the LEA class, x_L. Identify the maximum x_i value which is predicted as a member of the LEA class, x_R. (x_L, x_R) are then the predicted shoulder locations.

This method for shoulder location identification can be found in the `R` package `bulletxtrctr` as the function `get_grooves_lassofull`. Another method, `get_grooves_lassobasic`, is also available through `bulletxtrctr`, but is slightly less accurate and thus is not laid out here.

3.3 Bayesian Changepoint Analysis

Our second approach is also based on the idea that residuals e_i resulting from adapted robust LOESS predictions will follow a consistent pattern: decreasing values in the left GEA, values with no discernable slope change in the LEA, and increasing values in the right GEA. This can be thought of as a line with negative slope for the left GEA, line with

zero slope for the LEA, and a line with positive slope for the right GEA. The next model will therefore be defined in a piecewise fashion. The points of global structural change are what we will call changepoints. Changepoint locations can be treated as unknown parameters and estimated in the same manner as any other parameter in a statistical model.

Identifying such structural points was proposed more generally as Bayesian changepoint detection in (11). In the context of LEA residuals, there are additional complex patterns (i.e., striation marks), but the overall linear structure remains on a larger scale. We will consider the smaller scale striation patterns as dependence in the data after accounting for the large scale structures we are estimating.

We first perform some additional data processing steps to streamline computation. First, we scale the residuals e_i from the robust LOESS procedure by dividing by the standard deviation of residuals e_i for the entire profile. The purpose of this is purely to make certain priors easier to specify. Secondly, we impute missing residual values e_i to maintain equidistance between x_i locations.

We next develop a model which we will use to identify changepoints, and describe estimation procedures. Additional details on data processing can be found in the appendix.

3.3.1 Bayesian Model Formulation

For model formulation, we will define random variables associated with residual values e_i. Let $\{Y(x_i) : i = 1, 2, ..., n\}$ denote the set of random variables representing the residuals from the robust LOESS procedure at the values x_i. Also assume that $x_1 < x_2 < ... < x_n$. Let c_l be the value of the left changepoint and c_r be the value of the right changepoint. Here, the left changepoint is where the left GEA meets the LEA, and the right changepoint is where the right GEA meets the LEA. Denote the median centered x values as $x_i' = x_i - \tilde{x}$ where $\tilde{x}$ is the median x value. Complex small scale patterns, such as the striae, will be modeled through a covariance structure on the data that will be allowed to differ between each GEA and between the GEAs and LEA. We will construct the covariance matrices from the exponential covariance function $K(x, x'; \sigma, \ell) = \sigma^2 e^{-\frac{|x-x'|}{\ell}} = cov(Y(x), Y(x'))$. The differences in covariance matrices for the GEAs and LEA will be reflected in the parameters σ and ℓ. The data model that we consider is then,

$$(Y(x_1), Y(x_2), ..., Y(x_{k_1}))^\top \sim N(\beta_{01}\mathbb{1} + \beta_{11}x_{1:k_1}, \Sigma_1(\sigma_1, \ell_1)) \quad (1)$$

$$(Y(x_{k_1+1}), Y(x_{k_1+2}), ..., Y(x_{k_2}))^\top \sim N(0, \Sigma_2(\sigma_2, \ell_2)) \quad (2)$$

$$(Y(x_{k_2+1}), Y(x_{k_2+2}), ..., Y(x_n))^\top \sim N(\beta_{02}\mathbb{1} + \beta_{12}x_{k_2+1:n}, \Sigma_3(\sigma_3, \ell_3)), \quad (3)$$

where $x_{k_1} < c_l \leq x_{k_1+1}$ and $x_{k_2} < c_r \leq x_{k_2+1}$ Here, $x_{1:k}$ denotes the column vector $(x_1, x_2, ..., x_k)^\top$, and $\mathbb{1}$ denotes the vector of ones. Independence is assumed between each of these three distributions for two reasons. Firstly, it simplifies the computations significantly. Secondly, the robust loess residual values are the sum of the relevant global structure (GEA or LEA) and finer local structure (e.g. striae). The global structure component of any two residual values should be independent conditioned on the known global structure at the location of each residual. The local structure tends to be meaningful only over small regions of the land, and so any dependence between the GEAs and LEA due to local structure should be small enough to safely ignore. The parameters that need to be estimated include the four mean parameters in the GEAs, the six covariance parameters (two for each of the three areas), and the two changepoint parameters, c_l and c_r.

The above model encapsulates the essence of the approach. However, there are a few difficulties. The first difficulty is that there are not always two GEAs in a particular land. There may be one GEA, or the land may only consist of the LEA. Thus, the above model is conditional on there being two GEAs in the data. We also define models for when there is one GEA on the left, one GEA on the right, or no GEAs. The models are defined in an essentially identical way. Conditional on there being only one GEA, the left GEA model is defined as,

$$(Y(x_1), Y(x_2), ..., Y(x_k))^\top \sim N(\beta_0\mathbb{1} + \beta_1 x_{1:k}, \Sigma_1(\sigma_1, \ell_1)) \quad (4)$$

$$(Y(x_{k+1}), Y(x_{k+2}), ..., Y(x_n))^\top \sim N(0, \Sigma_2(\sigma_2, \ell_2)), \quad (5)$$

and the right GEA model is defined as,

$$(Y(x_1), Y(x_2), ..., Y(x_k))^\top \sim N(0, \Sigma_1(\sigma_1, \ell_1)) \tag{6}$$

$$(Y(x_{k+1}), Y(x_{k+2}), ..., Y(x_n))^\top \sim N(\beta_0 \mathbb{1} + \beta_1 x_{k+1:n} \Sigma_2(\sigma_2, \ell_2)). \tag{7}$$

Finally, conditional on there being no GEAs in the data, the model is simply

$$(Y(x_1), Y(x_2), ..., Y(x_n))^\top \sim N(0, \Sigma(\sigma, \ell)). \tag{8}$$

Estimating the changepoint locations also involves selecting the most appropriate model from these four. We use m_0 to denote the model with no GEA, m_{1l} and m_{1r} to denote the models with one left or one right GEA, respectively, and we use m_2 to denote the model with two GEAs. In order to avoid confusion, we have slightly abused notation and, for example, $\Sigma_1(\sigma_1, \ell_1)$ as it is estimated in the two changepoint model is *not* the same as $\Sigma_1(\sigma_1, \ell_1)$ from either of the one changepoint models, and $\Sigma_1(\sigma_1, \ell_1)$ is also *not* the same between the two one changepoint models. As another example, β_0 is *not* the same between each of the one changepoint models. So, to be clear, duplication of notation in *different* models is not meant to imply that those parameters are shared between models.

Ultimately, models m_0, m_{1l}, m_{1r}, and m_2 are each given a prior and individually fitted on every profile. From there, we do model selection in the formal Bayesian way, selecting number and location of changepoints by maximizing the estimated posterior distribution.

In order to complete a Bayesian model specification, we need priors on the parameters in each model as well as the model itself. We will assume independence between each parameter a priori. For each length scale ℓ, we will assume $\ell \sim \text{Gamma}(3, 5)$. For each standard deviation, we will assume $\sigma \sim \text{Half-Normal}^+(0, 1)$, where $\text{Half-Normal}^+(\cdot, \cdot)$ is notation for the normal distribution restricted to the positive real numbers. For intercept parameters, $\beta_{01}, \beta_{02}, \beta_0 \sim N(0, 10)$. For the slope parameters, the preceding trend deviates slightly. For any slope that corresponds to the *left* GEA, β_1 or β_{01}, we will assume that the slope can not be positive. That is, $\beta_1, \beta_{01} \sim \text{Half-Normal}^-(0, 10)$, where $\text{Half-Normal}^-(\cdot, \cdot)$ is notation for the normal distribution restricted to the negative real numbers. Contrastingly, for any slope that corresponds to the *right* GEA, β_1 or β_{02}, we will assume that the slope can not be negative. That is, $\beta_1, \beta_{01} \sim \text{Half-Normal}^+(0, 10)$. For the changepoint

locations, we assume a uniform prior $\pi(c_l, c_r) \propto I(a < c_l < c_r - \gamma < b - \gamma)$. Here, a and b are some values close to the edges of the data. How close those values are to the edges is a parameter that is set manually. Further, we include another hyperparameter, γ, which can be set so that the changepoints are not allowed to be too close to each other. This is also a parameter that is set manually. Lastly, we assume a uniform prior over all four models, i.e. $P(M = m_0) = P(M = m_{1l}) = P(M = m_{1r}) = P(M = m_2) = 1/4$.

3.3.2 Bayesian Model Estimation

As was noted in (11), for any model including a changepoint, the likelihood is not a smooth function of the changepoint location. This is because, holding all other parameters fixed, shifting the changepoint value will result in zero change to the likelihood until it crosses the nearest point to the right or left, at which point the likelihood makes a jump. This makes maximum likelihood estimation in the standard way infeasible, but Bayesian estimation can be done in a fairly straightforward way via Markov chain Monte Carlo (MCMC). The basic idea is that, for models m_{1l}, m_{1r}, and m_2, we can construct a two step Gibbs sampler. In step 1 we sample from the posterior distribution of the mean and covariance parameters given the changepoint locations, and in step 2 we sample from the changepoint locations given the mean and covariance parameters. Because of the non-conjugacy in our model, we perform both sampling steps using a random walk Metropolis-Hastings (RWMH) step with Gaussian proposals. For details on Gibbs sampling and the Metropolis-Hastings algorithm see (15). It is also worth mentioning that the model m_0 does not require Gibbs sampling at all, and we perform estimation there using a RWMH algorithm in the same way that we do for the other models.

We now provide the two basic steps of the Gibbs sampler for $M = m_2$. The algorithms to sample from the other three models are omitted, and are nearly identical except for the smaller number of parameters that need to be sampled. Denote collection of mean and covariance parameters for the left GEA as θ_1, the LEA as θ_2, and the right GEA as θ_3. Then, at iteration t

1. given changepoint locations $(c_l^{(t-1)}, c_r^{(t-1)})$, sample $(\theta_1^{(t)}, \theta_2^{(t)}, \theta_3^{(t)})$ using independent RWMH steps for each θ_i

2. given $(\theta_1^{(t)}, \theta_2^{(t)}, \theta_3^{(t)})$, sample $(c_l^{(t)}, c_r^{(t)})$ using a single RWMH step.

After running the MCMC for each model, parameter estimates and the most likely model are jointly chosen according to the largest joint posterior value. That is, we arrive at estimates $(\hat{\theta}, \hat{M}) = \underset{(\theta,M)}{\text{argmax}} \log(p(\theta, M|Y))$, where M is the random variable associated with the choice of model, θ is the associated parameter vector for the appropriate model, and Y is all of the available data. Final estimates $(c_l^{(t)}, c_r^{(t)})$ are then (x_L, x_R), predicted shoulder locations. Additional MCMC details can be found in the appendix.

4 Results

To assess the degree of improvement in automated shoulder location identification, we want to quantify the impact each prediction method has on an automated bullet matching algorithm's accuracy. Four different shoulder location identification methods will be inserted into the automated bullet matching algorithm process:

(1) rollapply, the method proposed by (8),

(2) logistic LASSO,

(3) Bayesian changepoint, and

(4) manual identifications, the "gold standard" for identification.

Shoulder location predictions were found using each of these methods, and used to remove GEA data from each profile. This is followed by extracting a signature, and using extracted signatures for each land engraved area to calculate pairwise similarity scores for all LEA signatures within each test set. Pairwise similarity scores were calculated using the random forest algorithm in the `bulletxtrctr` package.

Random forest scores should be close to 1 for LEA-to-LEA comparisons between signatures that originate from the same land, and closer to 0 for LEA-to-LEA comparisons between signatures from different lands.

This section needs to be expanded and more details need to be given: We will investigate these scores for each individual test set: Hamby set 44, Phoenix PD set, and Houston-test

set. All pairwise comparisons within a test set were completed. We look at both visual representations of the random forest score distributions as well as investigate the random forest method's accuracy in determining whether two lands originate from the same source, for each (x_L, x_R) identification method.

There should be visual separation between "same source" and "different source" distributions, as is present for Manual ID predictions for all three test sets, seen in **??**. Logistic LASSO and Bayesian changepoint both show improvement in separation for all three test sets; however, there is still room for improvement when compared to Manual ID distributions in the Phoenix PD and Houston-test sets.

There is also significant improvement in AUC values for all three test sets over rollapply, as seen in Table 1, Table 2, and Table 3. Of interest is the classification accuracy with a fixed false positive rate. Since false positives - in this case identifying two land engraved areas as same source when they are different source - are the worst possible mistake in the forensic context, we set a cutoff rate for random forest scores based on a controlled false positive rate of .01.

Accuracy is high regardless of shoulder location identification method; however, there was a significant reduction in number of false negatives for the logistic LASSO and Bayesian changepoint methods for both Hamby set 44 and Phoenix PD sets. For the Houston-test set, Bayesian changepoint improves upon rollapply, but lags behind the logistic LASSO method in AUC and reduction of false negatives.

5 Conclusions

Both the logistic LASSO and Bayesian changepoint approaches show significant improvement over the "rollapply" method proposed by (8). However, the logistic LASSO method shows greater improvement than Bayesian changepoint on the Houston-test set. While manual identification of shoulder locations is still the most accurate method, and considered the "gold standard", the reduction in time to get predictions when using LASSO methods is advantageous and allows for less human involvement in the overall automated matching process. Although manual identification of GEA data would still be the best safeguard against false positives in LEA-to-LEA matching, as it leads to the most accurate

results, either Bayesian changepoint or logistic LASSO could be applied in the automated process if the matching scores are used as a validation of a human examiner's decision. Examiners can either use the automated process as a validation of their decision, or determine at which point in the process they disagree with the automated algorithm and explain how and why their decision differs. When used as a validation support tool, the implications of a false positive from the automated process are much less problematic. What are the implications of a false positive? That depends on whether I am willing to give up accuracy in favor of "speed" and more (incorrect) objectivity.

The initial appeal of unsupervised methods such as Bayesian changepoint was the lack of dependence on training data and hence, potential generalizability advantages. This advantage did not bear out in the test sets presented here, and it appears the LASSO method trained on Hamby set 44 generalized effectively to new data. However, all test sets considered here consist of 9mm? bullets; Bayesian changepoint as an unsupervised approach may translate more effectively than logistic LASSO to additional gun types and calibers.

While improvement is apparent on all three test sets using the LASSO and Bayesian changepoint methods, there is clear room for additional precision. Future work on the LASSO method should include re-training the LASSO model on a wider variety of LEA types rather than just the Hamby set 44 to avoid over-fitting to a specific type of LEA.

6 Appendix

6.1 MCMC Details

As a practical note, it turns out that the posterior distribution is almost always multimodal, and it can happen that the sampler gets stuck in a suboptimal mode for a large number of iterations. It is also the case that the suboptimal modes need not even be close to the groove locations. It has, however, been our experience that the optimal mode corresponds well to the actual groove locations, which are often somewhat close to the edges of the data. With this in mind, starting values and the RWMH proposal variances play a very important role in the success of the sampling algorithm. Fortunately, it seems to be the

case that by setting the initial changepoint values close to the edges of the data and making the proposal variance small (around 100 seems to work well) allows the sampler to wander inwards, and even with a modest number of iterations (say 5000), typically pass through the largest mode corresponding to the groove locations. This is not always the case, and it is possible that increasing the number of iterations produces better results.

In our implementation of this algorithm, the sampling functions were originally written with the intention of tuning the proposal variances to potentially accelerate convergence, and thus several warm-up iterations are required for this purpose. The concept of warm-up iterations and tuning proposal variances based on warm-up iterations is described in Chapter 12.2 of (15). This turns out to be a bad idea in this context for two reasons. The first reason is that the warm-up iterations allow the sampler to wander past the global modes and get stuck in suboptimal modes far from the groove locations, from which the sampler may or may not find its way back to the optimal modes in just a few thousand iterations. Secondly, if the sampler does wander past the optimal modes, which are usually on the edges of the data, the tuned proposal variance can be quite large. The large proposal variance might not be a huge problem if it weren't for the fact that the width of the modes are almost always quite small. This means that it can take a very, very long time for the sampler to move from a suboptimal mode to the global mode. In order to mitigate this problem, we are currently setting the number of warmup iterations to be relatively small (somewhere in 100 to 500 seems to work well). In future, our implementation of the algorithm will not require any warmup iterations.

Initially, the Metropolis proposal variance for each θ_i is diagonal with diagonal elements all equal to $1/2$. The proposal variance for (c_l, c_r) is initially set to be diagonal with elements equal to 10^2. Note that because of the currently necessary warmup iterations, the variances after warmup for each θ_i becomes $\frac{2.4^2}{d}\hat{Var}(\theta_i^{(1:w)}) + \text{diag}(0.1)$, where d is the dimension of θ_i (which is not constant between GEAs and LEA), and $\hat{Var}(\theta_i^{(1:w)})$ is the estimated variance covariance matrix from the w warmup iterations. Note that the approximately optimal proposal variance as described in Chapter 12.2 of (15) is $\frac{2.4^2}{d}\hat{Var}(\theta_i^{(1:w)})$. The addition of a diagonal matrix with entries 0.1 is to avoid the case when most or all warmup iterations have the same value. Similarly, the proposal variance for (c_l, c_r) after warmup becomes

$\frac{2.4^2}{2}\hat{Var}((c_l, c_r)^{(1:w)}) + \text{diag}(1)$.

6.2 Data Preprocessing for MCMC

Before running the MCMC to do the changepoint detection, we first perform two data preprocessing steps. The first step is to scale the residuals from the robust loess procedure by the standard deviation calculated from the entire set of residuals. The reason for this is simply to make priors for standard deviation and slope parameters easier to specify. For example, ensuring that the residuals are scaled to have standard deviation one means that the standard deviation parameters in our model should also be close to one. This scaling also ensures that slopes values are not very large.

The second preprocessing step is a bit more involved. In order to enable the algorithm to run reasonably fast, we need to take advantage of the sparse precision matrix structure that is induced by the exponential covariance function. Indeed, this was the reason for choosing this covariance function in the first place. Unfortunately, it is challenging to do this unless the observations are evenly spaced in the domain. In our case, this would be true if there were no missing values. In order to remedy this problem, we impute the missing data, but only in the case that there exist non-missing observations outside of the missing values. In the case that the missing values exist on the edges of the data, we simply do not consider those domain values in the model.

We perform the imputation by treating the observations as coming from an unknown function, and infer the missing values from the known function values. In order to do this, we model the data with a Gaussian process and the squared exponential covariance function. That is, we suppose that

$$Y(x) \sim \mathcal{GP}(0, K(x, x'; \sigma^2, \ell)),$$

where now $K(x, x'; \sigma^2, \ell) = \sigma^2 e^{-(x-x')^2/(2\ell^2)}$ is the squared exponential covariance function. We emphasize for clarity that this is a different covariance function than we use in the changepoint model. The main reason for this is that in imputing values, it seems desirable to allow dependencies beyond immediately neighboring points to influence predictions as the function that we are trying to predict generally has a smooth global structure. For

all of our experiments, we set $\sigma = 0.8$ and $\ell = 15$. These values were chosen from doing maximum likelihood estimation for a representative bullet.

When we impute the missing values, we compute the conditional mean of the missing values. To be clear, denote the distribution of the observed and missing data as

$$(Y, Y^*)^\top \sim N\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}\Sigma_{yy} & \Sigma_{yy^*}\\ \Sigma_{y^*y} & \Sigma_{y^*y^*}\end{bmatrix}\right).$$

Here, Y is observed data and Y^* is the missing data, and the covariance matrix above is constructed from the squared exponential covariance function. We then use normal distribution theory to calculate the imputed values

$$E(Y^*|Y = y) = \Sigma_{y^*y}\Sigma_{yy}^{-1}y$$

.

References

1. AFTE Glossary. Theory of Identification as it Relates to Toolmarks. AFTE Journal 1998;30(1):86–88.

2. De Kinder J, Prevot P, Pirlot M, Nys B. Surface topology of bullet striations: an innovating technique. AFTE Journal 1998;30(2):294–299.

3. De Kinder J, Bonifanti M. Automated comparison of bullet striations based on 3D topography. Forensic Science International 1999;101:85–93.

4. Bachrach B. Development of a 3D-based Automated Firearms Evidence Comparison System. Journal of Forensic Sciences 2002;47(6):1253–1264.

5. Ma L, Song J, Whitenton E, Zheng A, Vorburger T, Zhou J. NIST bullet signature measurement system for RM (Reference Material) 8240 standard bullets. Journal of Forensic Sciences 2004;49(4):649–59.

6. Chu W, Song T, Vorburger J, Yen J, Ballou S, Bacharach B. Pilot study of automated bullet signature identification based on topography measurements and correlations. Journal of Forensic Sciences 2010;55(2):341–47.

7. Chu W, Thompson RM, Song J, Vorburger TV. Automatic identification of bullet signatures based on consecutive matching striae (CMS) criteria. Forensic Science International 2013;231(1-3):137–41.

8. Hare E, Hofmann H, Carriquiry A. Automatic matching of bullet land impressions. The Annals of Applied Statistics 2017 12;11:2332–2356.

9. Geometrical product specifications (GPS) – Surface texture: Profile method; Measurement standards – Part 2: Software measurement standards. Geneva, CH: International Organization for Standardization; 2012.

10. Hofmann H, Vanderplas S, Krishnan G. bulletxtrctr: Analyze bullet striations using nonparametric methods (Computer Program]; 2019. https://heike.github.io/bulletxtrctr/.

11. Stephens DA. Bayesian Retrospective Multiple-Changepoint Identification. Journal of the Royal Statistical Society Series C (Applied Statistics) 1994;43(1):159–178. https://www.jstor.org/stable/2986119.

12. Cleveland WS. Robust Locally Weighted Regression and Smoothing Scatterplots. Journal of the American Statistical Association 1979;74(368):829–836.

13. Tibshirani R. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society Series B (Methodological) 1996;58(1):267–288. http://www.jstor.org/stable/2346178.

14. Friedman J, Hastie T, Tibshirani R, Simon N, Narasimhan B, Qian J. glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models (Computer Program]; 2018. https://cran.r-project.org/web/packages/glmnet/glmnet.pdf.

15. Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB. Bayesian Data Analysis. Chapman and Hall; 2013.

List of Figures

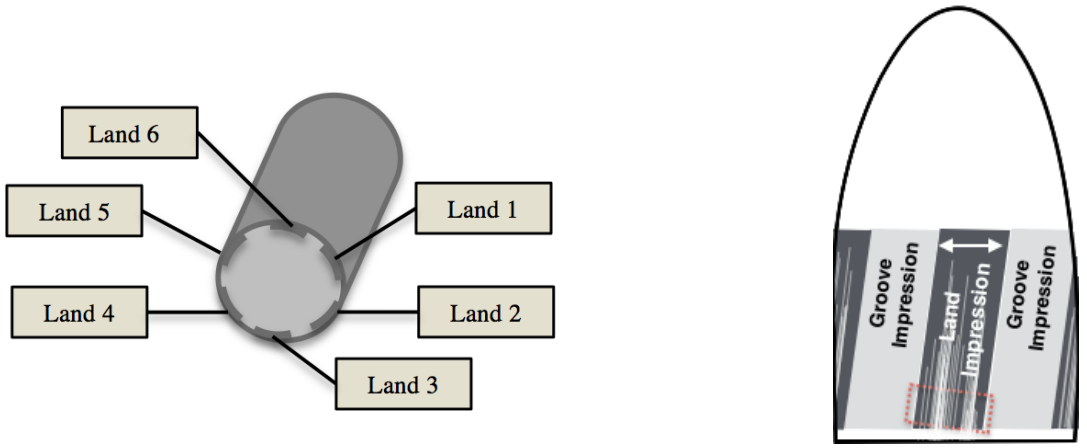

Figure 1: (Left) A sketch depicting lands inside a traditionally rifled barrel with six lands. (Right) A sketch of a land engraved area and striation marks engraved on the bullet. Groove engraved areas are found between land engraved areas. The red area denotes the area of a bullet which would be captured as part of a LEA scan.

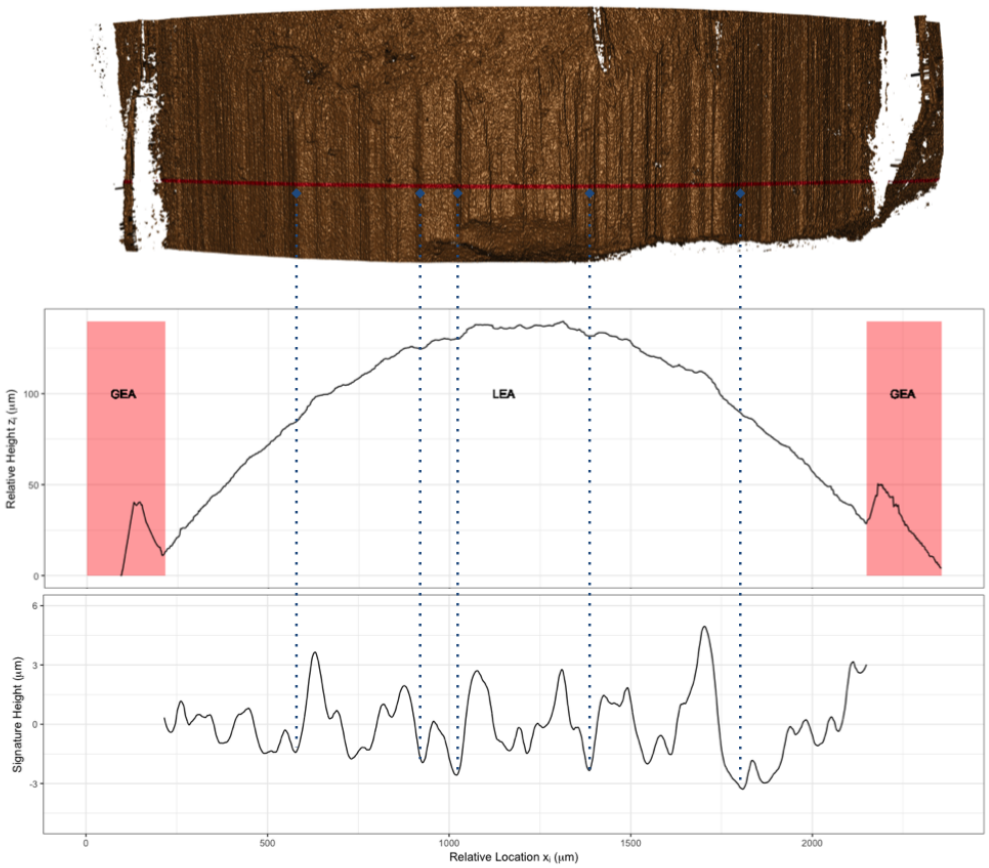

Figure 2: The process of extracting a 2D signature from a high-resolution 3D scan of a land engraved area (LEA) on a bullet. (Top) Computer rendering of a high-resolution 3D bullet LEA scan. Red line denotes horizontal crosscut which is extracted from the scan. (Middle) 2D extracted profile. Red boxes denote data which are part of the GEAs to the left and right sides of the LEA data. (Bottom) 2D extracted LEA signature with bullet curvature removed. Signatures are a representation of the striation pattern on each LEA. Vertical lines depict alignment of valleys with prominent striation marks.

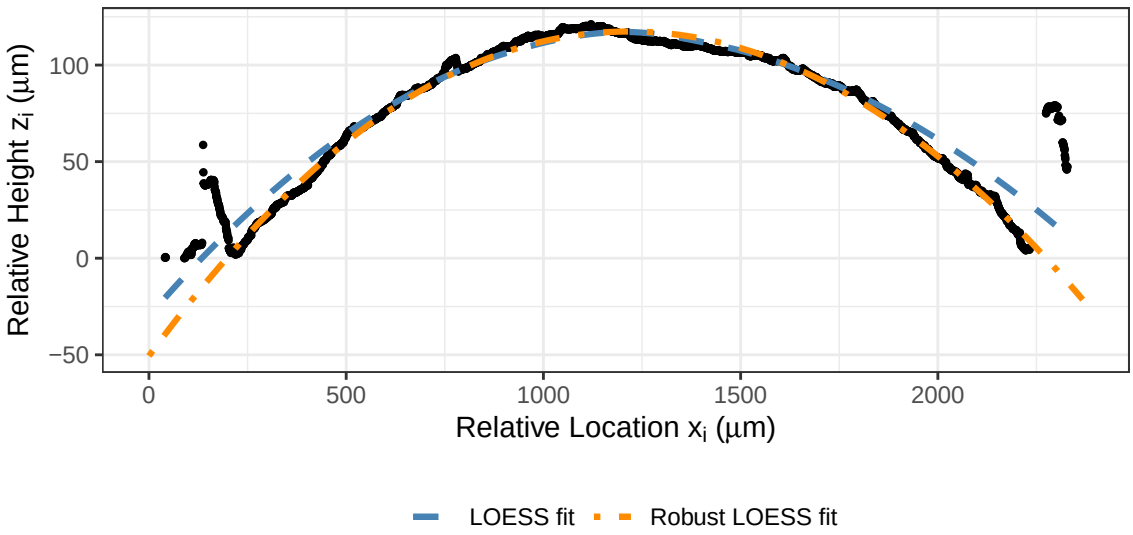

Figure 3: An example of the difference between traditional LOESS fit and robust LOESS fit to an LEA profile from Hamby set 44.

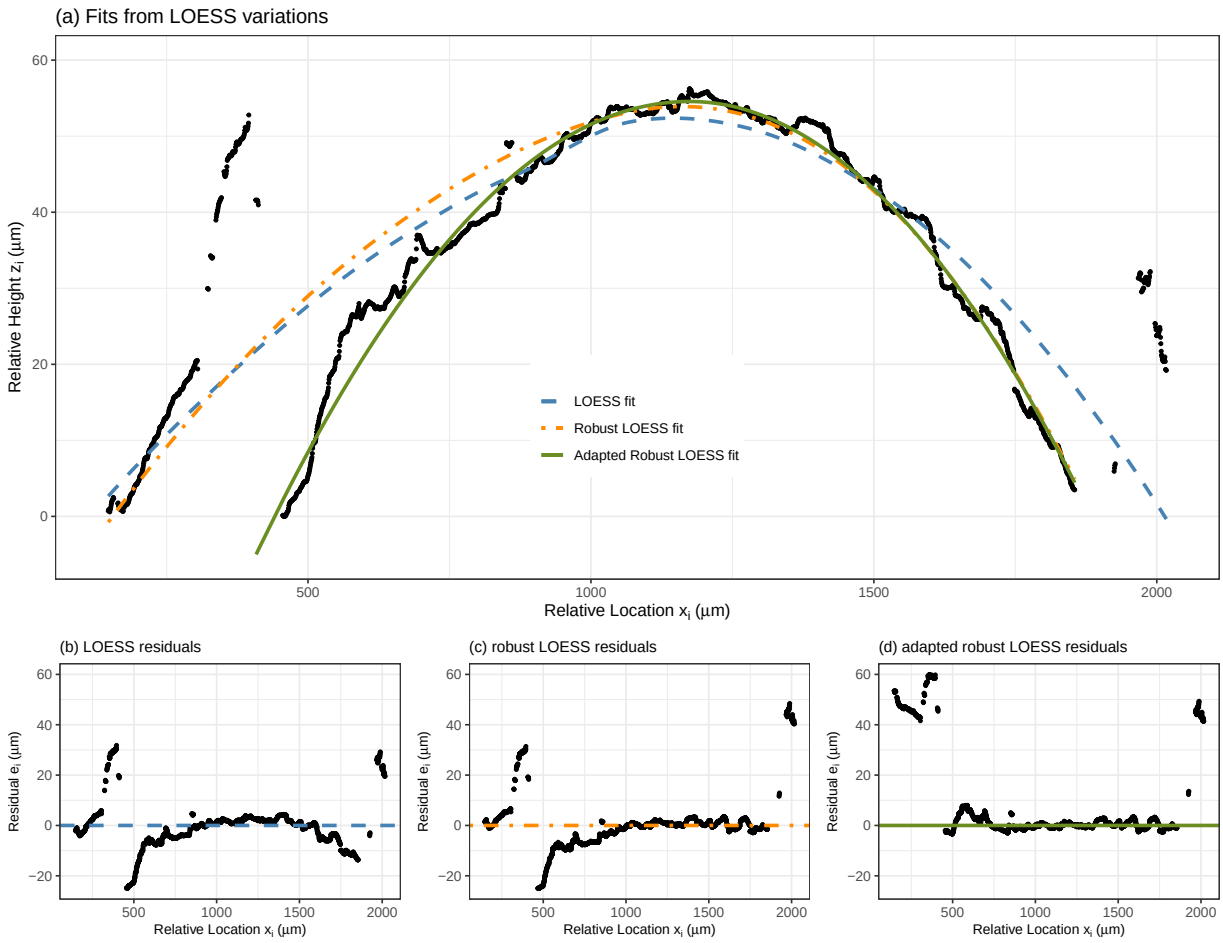

Figure 4: An example of the difference between LOESS, robust LOESS, and adapted robust LOESS fits to an LEA profile from the Houston-test set. (a) depicts predicted curves for all three methods on one LEA. Adapted robust LOESS most closely fits the LEA structure and allows GEA data to remain a separate structure. (b), (c), and (d) depict residuals e_i resulting from each respective prediction method. Adapted robust LOESS in (d) results in the most desirable residual pattern, with LEA data residuals remaining closer to zero, and GEA data residuals being positive and large.

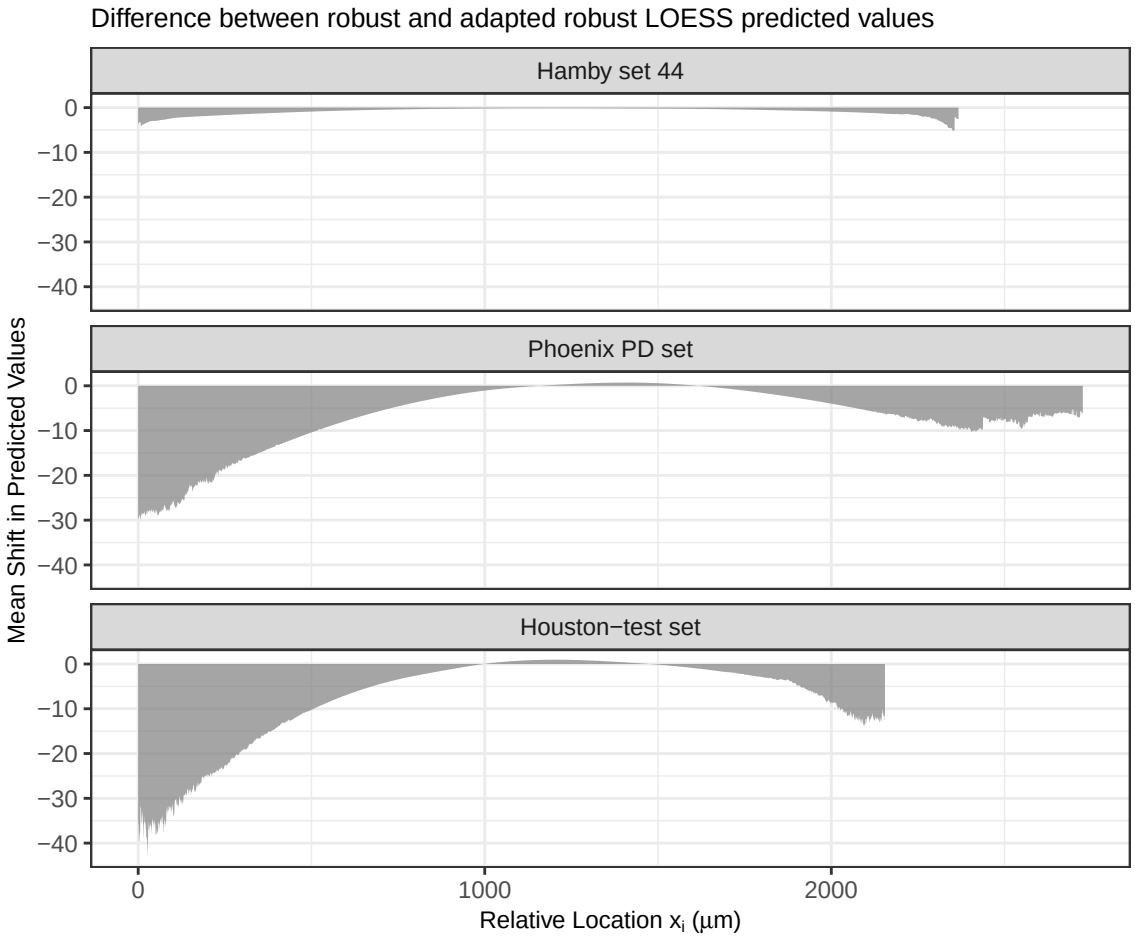

Figure 5: Mean shift in predictions when applying the adapted robust LOESS procedure in place of the traditional robust LOESS procedure. For Hamby set 44 predictions are, on average, very similar. The Phoenix PD and Houston-test sets have more significant downwards shifts in predictions near the left and right boundaries.

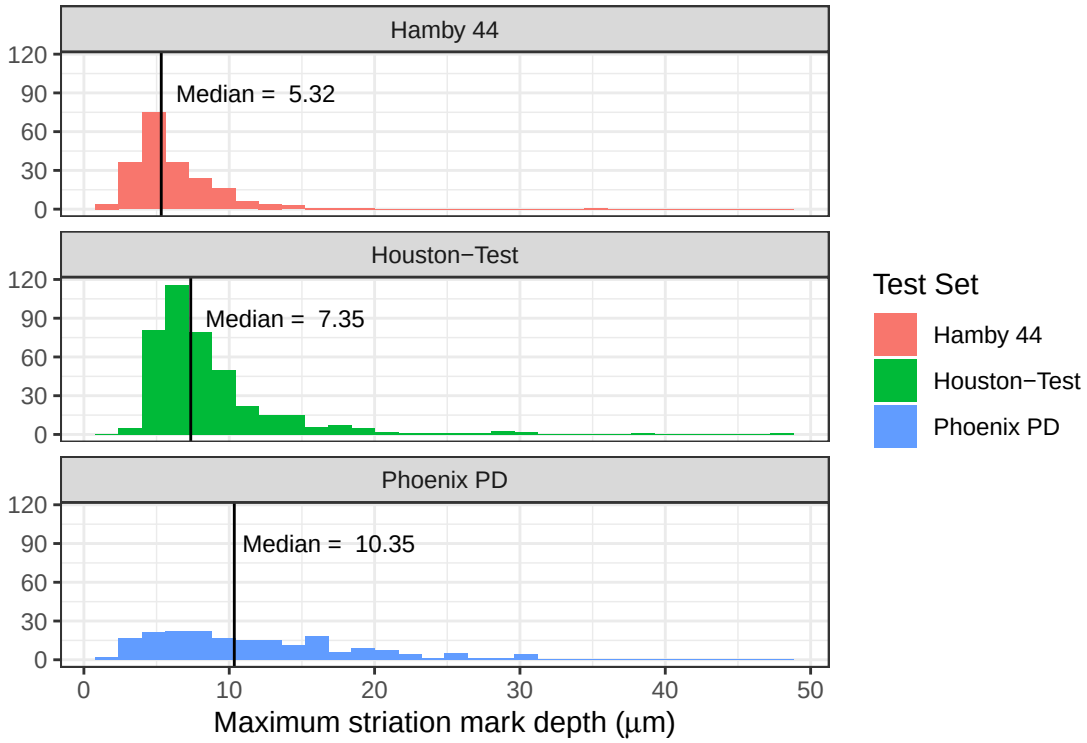

Figure 6: Distribution of maximum striation depth for each of the three bullet test sets. Maximum striation depths are calculated as the largest observed absolute signature value in each individual LEA signature. Black vertical lines represent the median depth for each test set. Each test set has a different distribution, which indicates standardization of residual heights is crucial for generalizability of parameter estimates.

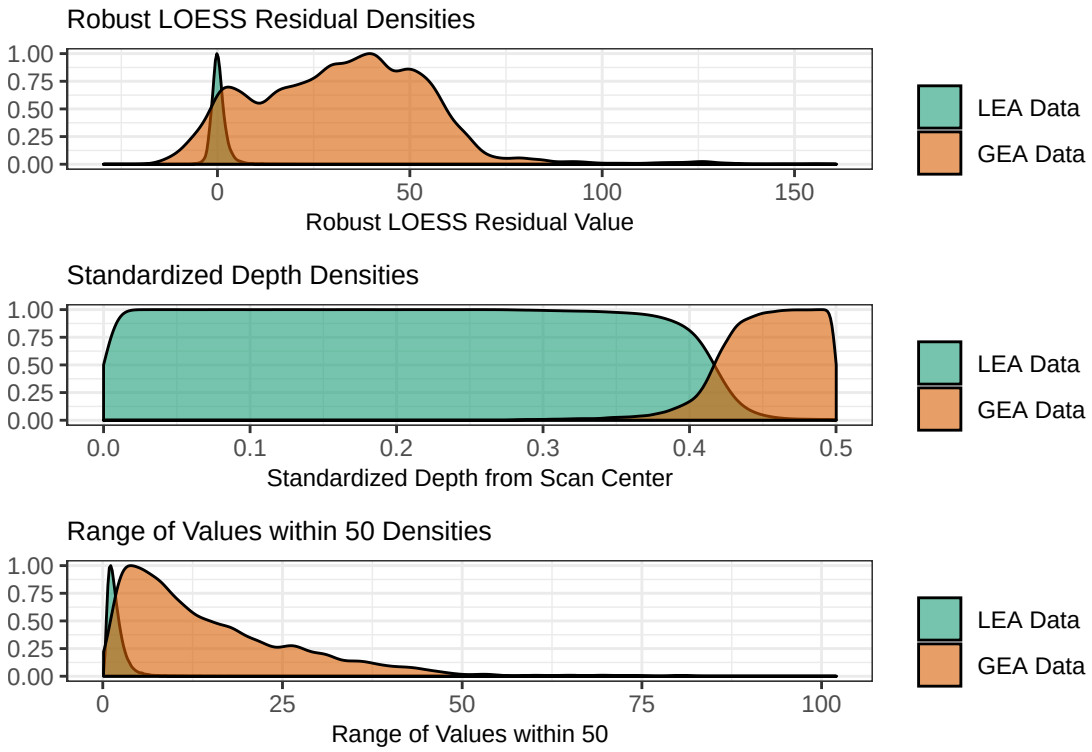

Figure 7: Example distributions of features used in two-class classification from Hamby set 44. While depth shows the most clear separation between GEA and LEA data, it alone will not suffice to classify data correctly.

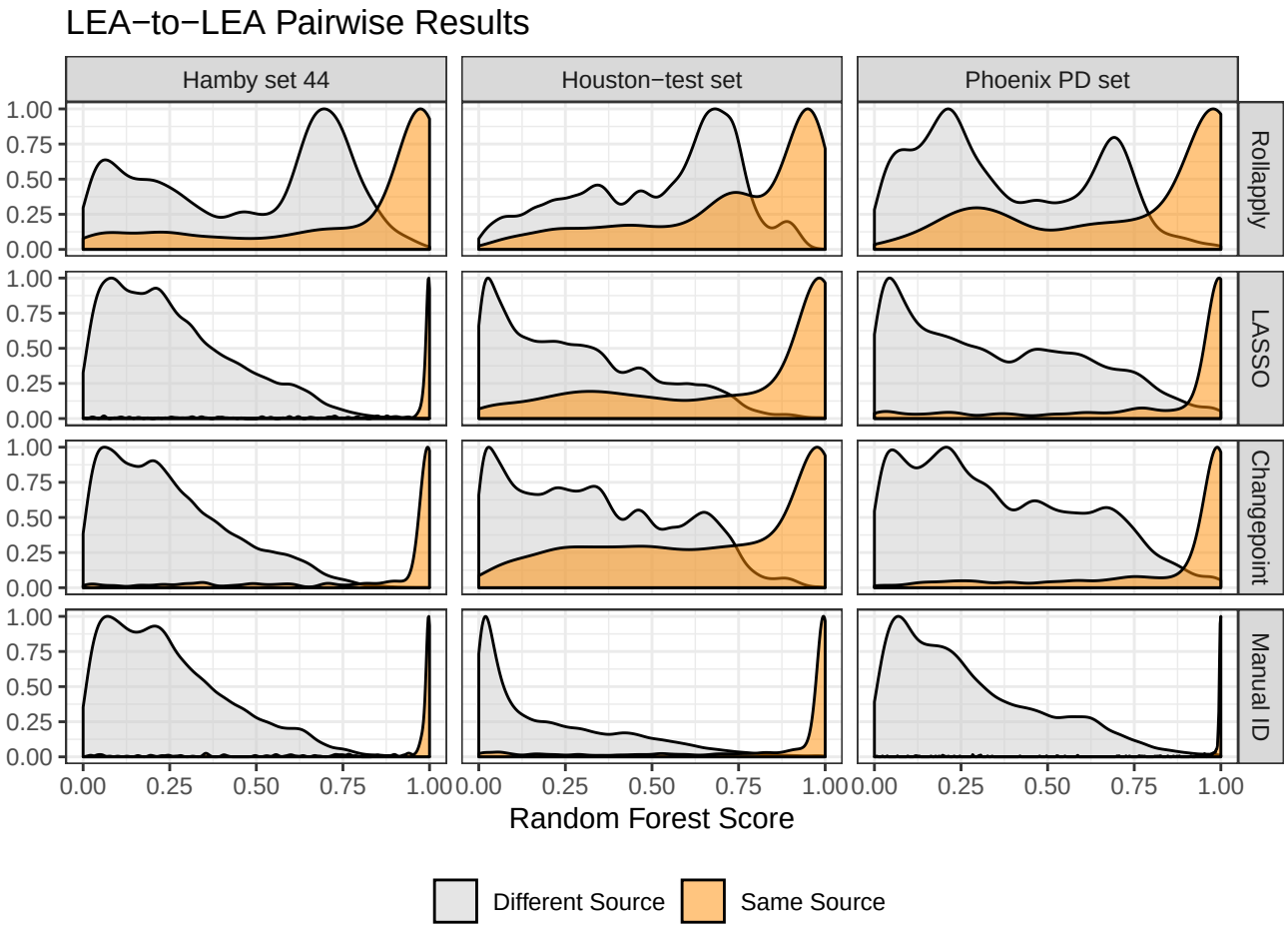

Figure 8: Random forest score distributions for same source and different source land-to-land comparisons for all test sets. Logistic LASSO demonstrates improvement over Rollapply, but is still not as well separated as the Manual ID distributions. Bayesian changepoint demonstrates similar improvement for Hamby set 44 and Phoenix PD sets, but does not improve as much as the LASSO method for the Houston-test set.

List of Tables

		Controlled FPR = .01					
Method	**AUC**	Cutoff	FN	TP	TN	Accuracy	**Time to Calculate**
Rollapply	0.8	0.92	332	420	42126	0.98	1 min.
Logistic LASSO	0.94	0.75	126	626	42106	0.99	6 min.
Bayesian Changepoint	0.93	0.74	152	600	42098	0.99	150 min.
Manual ID	0.94	0.74	124	628	42088	0.99	45 min.

Table 1: LEA-to-LEA comparison results for Hamby set 44. Results are reported as area under the curve (AUC), as well as False Negative, True Positive, and True Negative rates for a controlled false positive rate of .01. Both logistic LASSO and Bayesian changepoint show significant improvement over the rollapply method.

		Controlled FPR = .01					
Method	**AUC**	Cutoff	FN	TP	TN	Accuracy	**Time to Calculate**
Rollapply	0.818	0.9	364	378	38082	0.98	1 min.
Logistic LASSO	0.893	0.953	238	504	38082	0.98	6 min.
Bayesian Changepoint	0.903	0.937	256	486	38080	0.98	140 min.
Manual ID	0.953	0.853	96	646	38084	0.99	45 min.

Table 2: LEA-to-LEA comparison results for the Phoenix PD set. Results are reported as area under the curve (AUC), as well as False Negative, True Positive, and True Negative rates for a controlled false positive rate of .01. Both logistic LASSO and Bayesian changepoint show significant improvement over the rollapply method.

		Controlled FPR = .01					
Method	**AUC**	Cutoff	FN	TP	TN	Accuracy	**Time to Calculate**
Rollapply	0.761	0.91	1652	1016	167118	0.98	2 min.
Logistic LASSO	0.858	0.823	1144	1524	167062	0.98	12 min.
Bayesian Changepoint	0.795	0.863	1552	1116	167096	0.98	280 min.
Manual ID	0.931	0.823	614	2054	167098	0.99	75 min.

Table 3: LEA-to-LEA comparison results for the Houston-test set. Results are reported as area under the curve (AUC), as well as False Negative, True Positive, and True Negative rates for a controlled false positive rate of .01. Logistic LASSO method shows significant improvement over the rollapply method. Bayesian changepoint shows improvement, but less significant than the improvement shown by logistic LASSO.