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

June 21, 2019

Abstract

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

*The authors gratefully acknowledge ...

TO DO:

- Details on robust LOESS and downweighting.

- Figures work
- Profile/crosscut consistency

1 Background

Forensic pattern anaysis 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 microimperfections 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 believed? to be areas which will bear unique patterns of striation marks which are reproduced on any bullet fired through the same individual? barrel. 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 1; 2; 3; 4; 5; 6; 7). The data used in such algorithms are high resolution 3D scans of LEAs, such as that pictured in Figure 2.

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 comparison compares the patterns of striation marks engraved on the LEA, which can be extracted from a 3D LEA scan as a 2D signature which details a pattern of peaks and valleys representing striation marks.

Signatures are extracted by first obtaining a horizontal slice of the 3D scan, called a

crosscut profile?, followed by removing the overall bullet curvature which dominates the data structure of each crosscut profile?. Figure 3 depicts the process of translating a 3D LEA scan to a 2D LEA signature.

Currently accepted best practice for capturing 3D LEA scans involves capturing portions of the neighboring groove engraved areas (GEAs), which introduce 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.

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 compare performance of the two approaches on three separate test sets of bullets.

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 (8). 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 consists of 35 bullets fired from 10 consecutively rifled Ruger P85 barrels. There are two known bullets for each of the ten barrels, as well as 15 additional questioned bullets. Each fired bullet in Hamby Set 44 has 6 LEAs; every LEA was scanned for each of the 35 bullets, 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 consists 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. There are 6 LEAs for each of the fired bullets, producing a total of 198 individual land engraved areas.

The Houston-test set consists of 45 bullets fired from at least 10 barrels more information on the type of barrels?. There are three known bullets for each of the ten barrels, as well as 24 questioned bullets that are fired from a combination of the 10 known barrels and additional, out-of-set barrels. There are 6 LEAs for each of the fired bullets, producing a total of 414 individual land engraved areas.

A crosscut was extracted from each scan by identifying an optimal crosscut using `x3p_crosscut_optimize` in the `bulletxtrctr` package in `R` (9). Then, an averaged profile was calculated by averaging across ten consecutive crosscuts which fall directly to either side of the optimal crosscut identified. 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

3.1 Global Structure Removal

The first stage of our GEA removal process focuses on removing the bullet curvature from each profile crosscut?. 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. Figure which demonstrates two patterns on profile. 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 as a directly circular structure. We use more flexible non-parametric locally weighted regression (LOESS), which can model 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 which minimize:

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

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 pre-specified decreasing function, traditionally a particular type of weight.

An approach which mitigates the impact of GEA data is robust LOESS, a process which iteratively updates weights XXX based on the magnitude of each residual $e_i = z_i - \widehat{z}_i$. Larger values of e_i result in a lower weight for that data point, z_i. The process is as follows:

1. Fit a LOESS model to a LEA crosscut, predicting height z_i using relative location x_i. Assign weights XXX of 1 to each data point.

2. Obtain predicted height values $\widehat{z}_i$, and corresponding residual values $e_i = z_i - \widehat{z}_i$.

3. Calculate updated weights XXX using residual values e_i:

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

   where MAD is the median absolute deviation of residuals e_i. This is known as the bisquare function.

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

5. After k iterations of updating the weight vector $w_k(x_i)$, fit a LOESS model and obtain residual values e_i.

Reweighting data as in Step 3 reduces influence of data points with large absolute residual values. When GEA data is present on a crosscut, largest residual values will occur in areas where GEA data begins as it presents a competing structure with overall LEA curvature.

Figure 5 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 (see Figure 6. Thus, we make a small adaptation to the procedure to function more effectively with these data structures:

1. Fit a LOESS model to a LEA crosscut, predicting height z_i using relative location x_i. Assign weights XXX of 1 to each data point.

2. Obtain predicted height values $\widehat{z}_i$, and corresponding residual values $e_i = z_i - \widehat{z}_i$.

3. Calculate updated weights XXX using residual values e_i:

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

   where MAD is the median absolute deviation of residuals e_i. This is known as the bisquare function.

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

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

6. After k iterations of updating the weight vector $w_k(x_i)$, fit a LOESS model and obtain residual values e_i.

This adapted robust LOESS, while a small procedural change, more adeptly fits bullet cruvature. Figure 7 demonstrates the impact of the adaptation on predicted height values z_i. This impact differs for each of the bullet test sets. Figure w/pulling down of predictions demonstrates the differing impact dependent on bullet test set.

Once the adapted robust LOESS procedure has been applied to a LEA crosscut, 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 statistical 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 crosscut. One method, using supervised two-class classification techniques, aims to classify each data point as being part of the LEA or GEA. The second approach is an unsupervised method based on changepoint analysis that seeks to identify data points where the distribution of the residuals changes.

3.2 Two-Class Classification

Shoulder location can be predicted by first classifying data points 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 themselves should have different distributions, residuals alone are not enough to classify data with high accuracy.

As already demonstrated by the differences in Figure 5 and Figure 6, certain ammunition or barrel types may result in more pronounced striae on an LEA. 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 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 values present on the scan. 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 similar scale.

For variables which instead deal with differences in the x direction, such as depth from the center of a scan, values will be mapped to a $(0, 1)$ range, with the maximum x location on the scan acting as the divisor.

The full list of features developed are as follows:

`rlo_resid_std`: Robust LOESS residual value, standardized by dividing by standard deviation of residual values from middle 50% of **x** values.

`(rlo_resid_std)`$^{\mathbf{2}}$: Squared term of `rlo_resid_std`.

`side`: Whether data point is to left or right of median x value.

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

`side:depth_std`: Interaction between `side` and `depth_std` variables.

`xint1_std`: Predicted x intercept of robust LOESS on left side of land, standardized by dividing by maximum x value (a proxy for the range of x).

`xint2_std`: Predicted x intercept of robust LOESS on right side of land, standardized by dividing by maximum x value (a proxy for the range of x).

`range_50_std`: Range of residual values within a 50-point window around data point, standardized by dividing by standard deviation of residual values from middle 50% of x values.

`numNA_50`: Number of missing values within a 50-point window around data point.

`ind_2mad`: Indicator of whether `rlo_resid` is greater than `2*MAD(rlo_resid)`.

`numpos_50`: Number of positive residual values within a 50-point window around data point.

`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 9.

The developed features can be used in the fit of a logistic LASSO model, which is a form of penalized regression. LASSO parameter values for p covariates are found by identifying:

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

which adds a penalty to the traditional ordinary least squares minimization problem, and uses a tuning parameter λ.

Cross-validated LASSO models were fit using the `cv.glmnet` function in the `glmnet` package (10). For each model, 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.

Two separate models were fit: "LASSO Basic", which uses each of the features listed above, and "LASSO Full", which uses each of the features along with pairwise interactions for each of them. These parameter values were trained on the Hamby set 44 data. The resulting parameter values were used to calculate predicted values between 0 and 1; the closer to 1, the higher probability of membership in the "GEA" class.

Do we even need this paragraph?: Traditional two-class classification techniques call for finding an equal error rate cutoff to classify the predicted values for each data point into each of the two classes; 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. However, since scans are primarily of the land engraved area, there is significant class imbalance in our response categories. This unbalanced response means that any criterion used to determine a reasonable cutoff value for classification needs to be adjusted to account for differences in the size of the response classes. The larger class size of "LEA" means an equal error rate would allow for more false negatives (i.e., more "GEA" data points being classified as "LEA") - exactly what we are trying to avoid.

Thus, Instead of looking for thecutoff that gives raw equal error rate (where sensitivity and specificity are equal), we found an equal error rate based on the overall number of data points. This allows for an equal number of errors in each category, rather than equal percentage of errors. This tactic more fairly penalizes points that should be classified

"GEA" but are predicted to be "LEA".

The final process for shoulder location identification is as follows:

1. Use adapted robust LOESS procedure to remove bullet curvature.
2. Calculate features based on x locations and residual height values from Step 1.
3. Use fit parameter values from either (a) lassobasic or (b) lassofull models to calculate probabilities of membership in GEA class.
4. Apply cutoff of (a) .33 or (b) .34 to classify higher probabilities as GEA data points, and lower probabilities as LEA data points.
5. Gather the x range of data points predicted as members of the LEA class. These are the final shoulder location predictions. Make this more mathematically formulated?

This process results in two methods for shoulder location identification in the `R` package `bulletxtrctr`: "lassobasic" and "lassofull".

3.3 Bayesian Changepoint Analysis

The idea behind the changepoint approach is that within either the left GEA, right GEA, or the LEA, there are consistent patterns which can either be described by a line with negative slope for the left GEA, a line with zero slope for the LEA, or a line with positive slope for the right GEA. Our model will be 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 parameters and estimated in the same manner as any other parameter in a statistical model.

This approach was proposed in the more general context of Bayesian changepoint detection in Stephens (11). In practice there are also complex additional patterns (e.g., striae), but assuming the aforementioned large linear structures remains generally reasonable. The complex smaller scale patterns can be thought of as the dependence in the data after accounting for the larger structures.

Because of the nature of the model which we consider, it becomes necessary for computational reasons to perform a couple of additional data preprocessing steps. Specifically,

we will scale the residuals from the robust LOESS procedure, and we will impute missing values. In the next section, we describe the model that we will use to identify changepoints, after which we will describe the estimation procedure which we use. Details of the additional data preprocessing steps can be found in the appendix.

3.3.1 Bayesian Model Formulation

Before introducing the model, we introduce some notation. First, 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. For simplicity, also assume that $x_1 < x_2 < ... < x_n$. Also, 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. Also, denote the median centered x values as $x_i' = x_i - \tilde{x}$ where $\tilde{x}$ is the median x value. As mentioned in the previous paragraph, the 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,

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

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

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

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. Indpendence is assumed between each of these three distributions for simplicity. 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 actually 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,

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

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

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

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

We see that estimating the changepoint locations also involves selecting the most appropriate model from these four. 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, these above four models are each individually fitted, and each model above is given a prior. 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 each of the parameters in each model as well as each 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.

3.3.2 Bayesian Model Estimation

As was noted in Stephens (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 each model, 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 Gelman et al. (12). It is also worth mentioning that the zero changepoint model does not require Gibbs sampling at all, and we perform estimation there using a

RWMH algorithm.

We now provide the two basic steps of the Gibbs sampler for the two changepoint case. 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 after warmup

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 $(\hat{\theta}, \hat{M}) = \underset{(\theta,M)}{\text{argmaxlog}}(p(\theta, M|Y))$, 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. 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. Five different shoulder location identification methods will be inserted into the automated bullet matching algorithm process:

(1) rollapply, the method proposed by (7),

(2) LASSO basic,

(3) LASSO full,

(4) Bayesian changepoint, and

(5) 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 land-to-land comparisons between signatures that originate from the same land, and closer to 0 for land-to-land comparisons between signatures from different lands.

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 are completed. We can 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.

There should be visual separation between "same source" and "different source", as is present for Manual ID predictions for all three test sets, seen in Figure 10, Figure 11, and Figure 12. LASSO basic, LASSO full, and Bayesian changepoint all show improvement in separation for all three test sets; however, there is still much room for improvement when compared to Manual ID distributions in the Phoenix PD and Houston-test sets.

Quantitatively, there is also significant improvement in AUC values for all three test sets, 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, we set a cutoff rate for random forest scores based on a controlled false positive rate of .01.

Accuracy is overall high regardless of shoulder location identification method; however, there was a significant reduction in number of false negatives for the LASSO basic, LASSO full, 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 LASSO basic and LASSO full methods in AUC and reduction of false negatives.

5 Conclusions

All three proposed approaches show significant improvement both visually and quantitatively over the "rollapply" method proposed by (7). However, both LASSO methods show 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.

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 methods trained on Hamby set 44 generalized effectively to new data.

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 methods should include re-training LASSO models 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 References

7 Appendix

7.1 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 warmup iterations are required for this purpose. This turns out to be a bad idea in this context for two reasons. The first reason is that the warmup 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 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 $\frac{2.4^2}{2}\hat{Var}((c_l, c_r)^{(1:w)}) + \text{diag}(1)$.

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

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

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

4. 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.

5. 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.

6. 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.

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

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

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

10. 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.

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

List of Figures

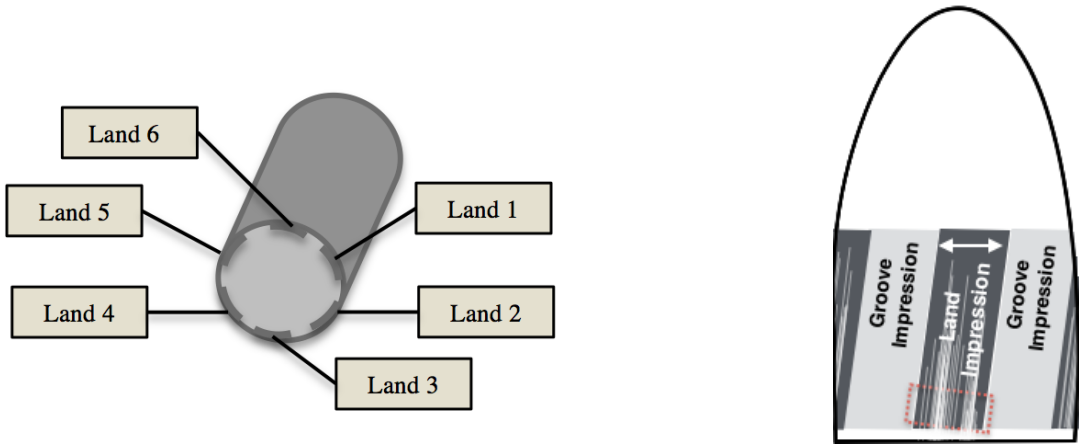

Figure 1: (Left) A sketch depicting lands inside a traditionally rifled barrel with six lands. (Right) A sketch of a land engraved area and striation marks engraved on the bullet. Groove engraved areas are found between land engraved areas. The red area denotes the area of a bullet which would be captured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

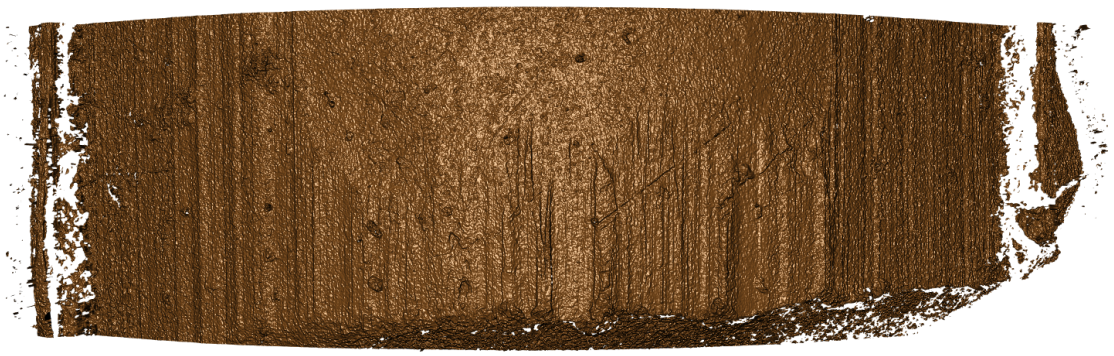

Figure 2: Computer rendering of a high resolution 3D scan of a bullet land engraved area (LEA).

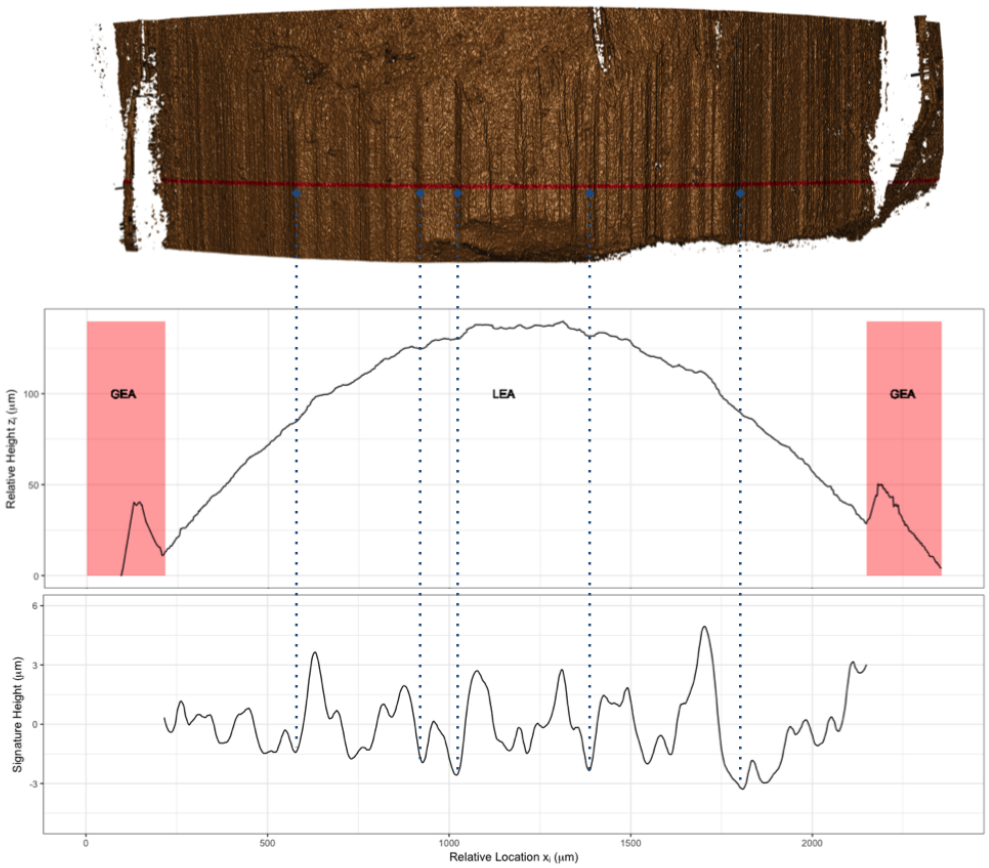

Figure 3: The process of extracting a 2D signature from a high-resolution 3D scan of a land engraved area (LEA) on a bullet. (Top) Computer rendering of a high-resolution 3D bullet LEA scan. Red line denotes horizontal crosscut which is extracted from the scan. (Middle) 2D extracted profile. Red boxes 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.

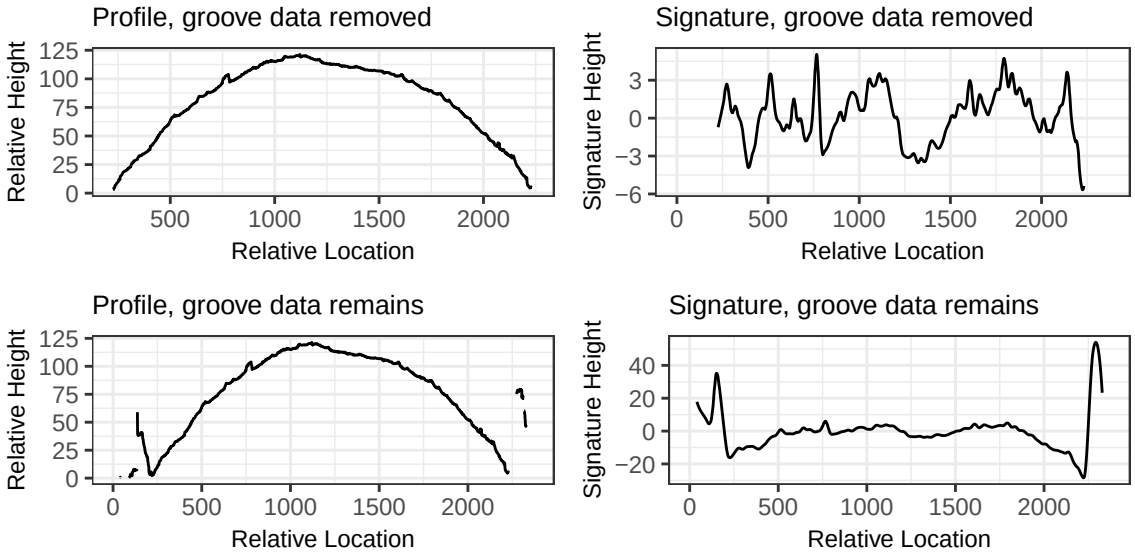

Figure 4: An example of the impact failure to remove GEA data can have on an extracted 2D signature. Important data features are obfuscated in the signature by remaining GEA structure.

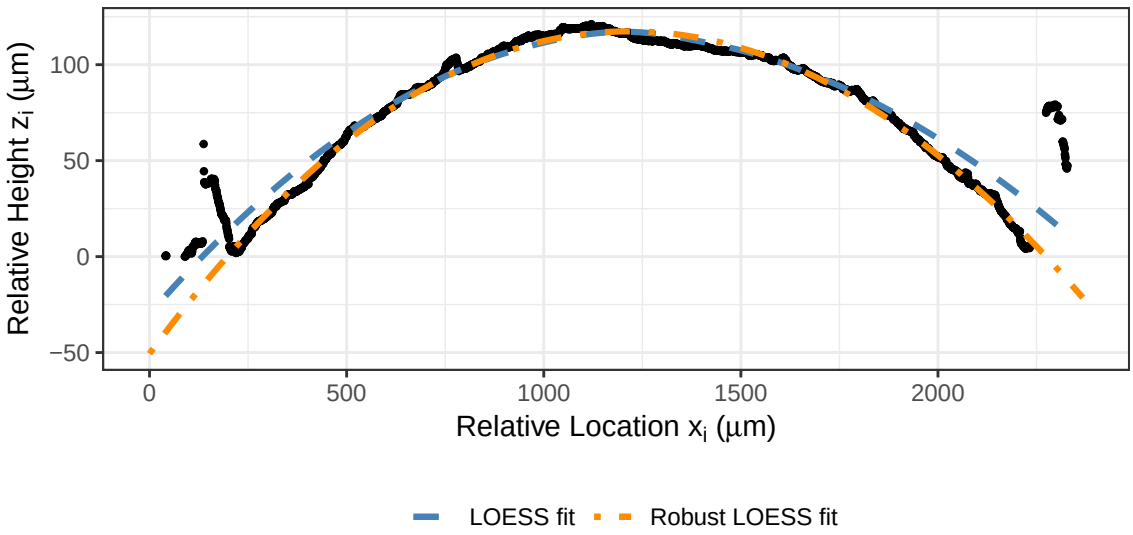

Figure 5: An example of the difference between traditional LOESS fit and robust LOESS fit to an LEA profile from Hamby set 44.

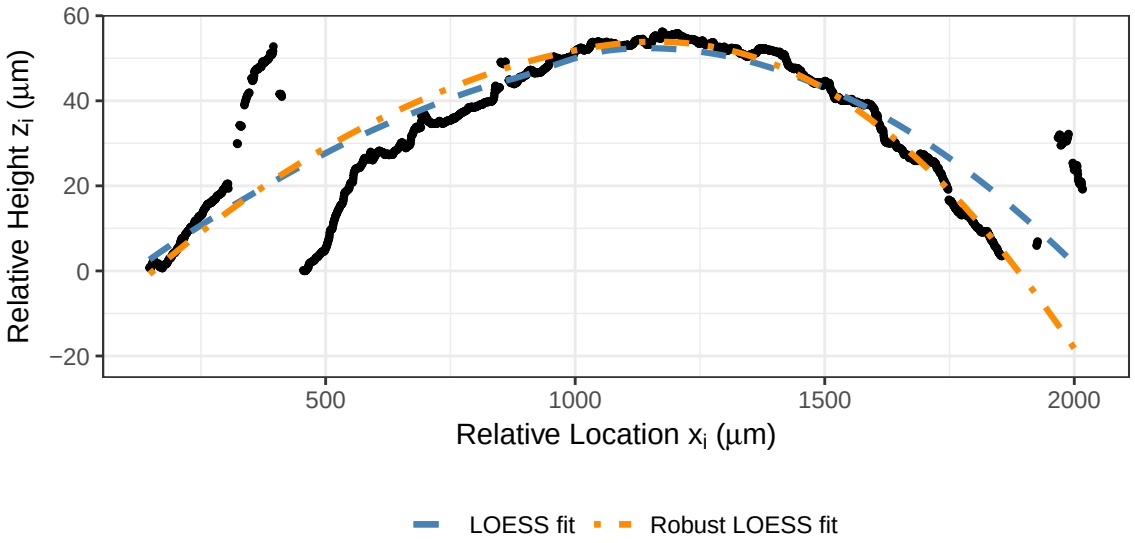

Figure 6: An example of the difference between traditional LOESS fit and robust LOESS fit to an LEA profile from the Houston-test set. While the smaller GEA structure on the right side is accounted for using the robust procedure, the larger GEA structure on the left is still problematic.

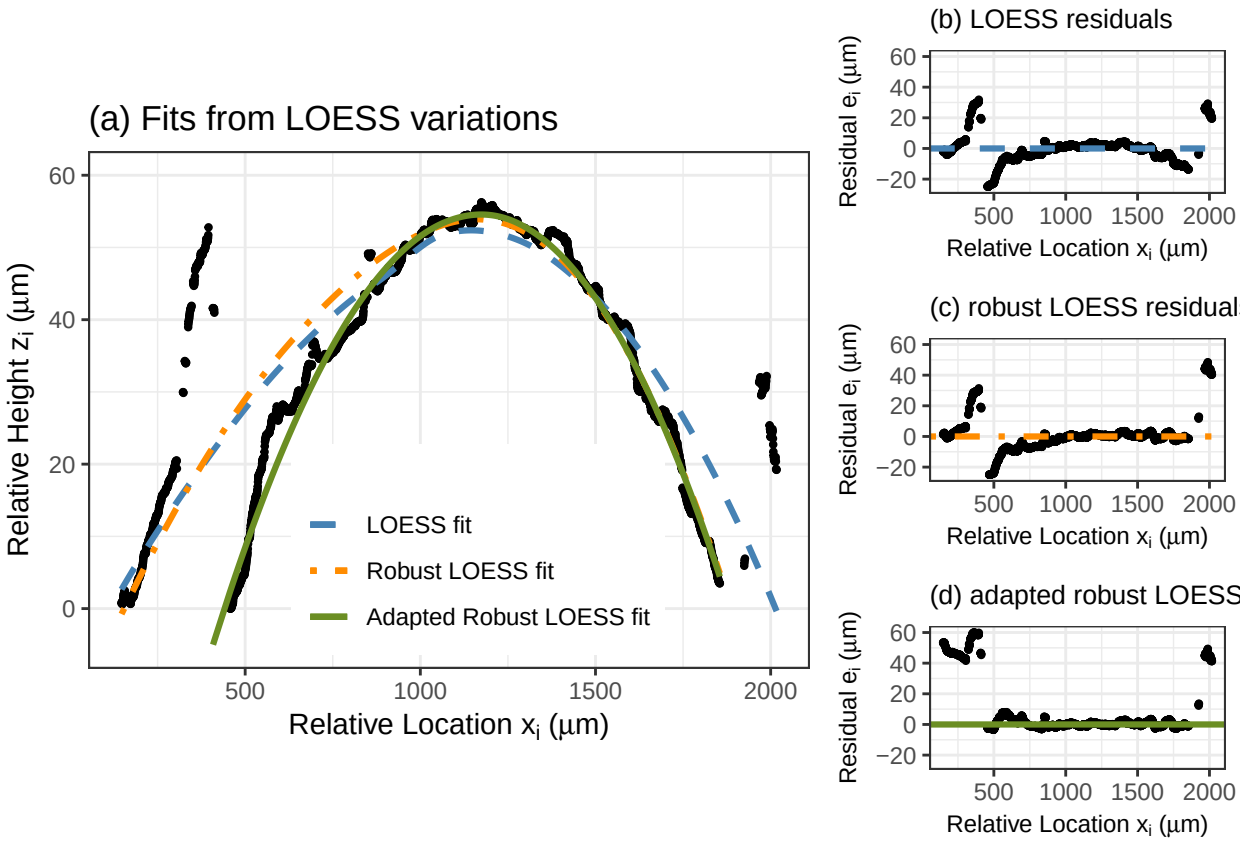

Figure 7: An example of the difference between LOESS, robust LOESS, and adapted robust LOESS fits to an LEA profile from the Houston-test set. Iteratively downweighting only positive residuals results in a significantly different fit which accounts for GEA structures on both the left and right.

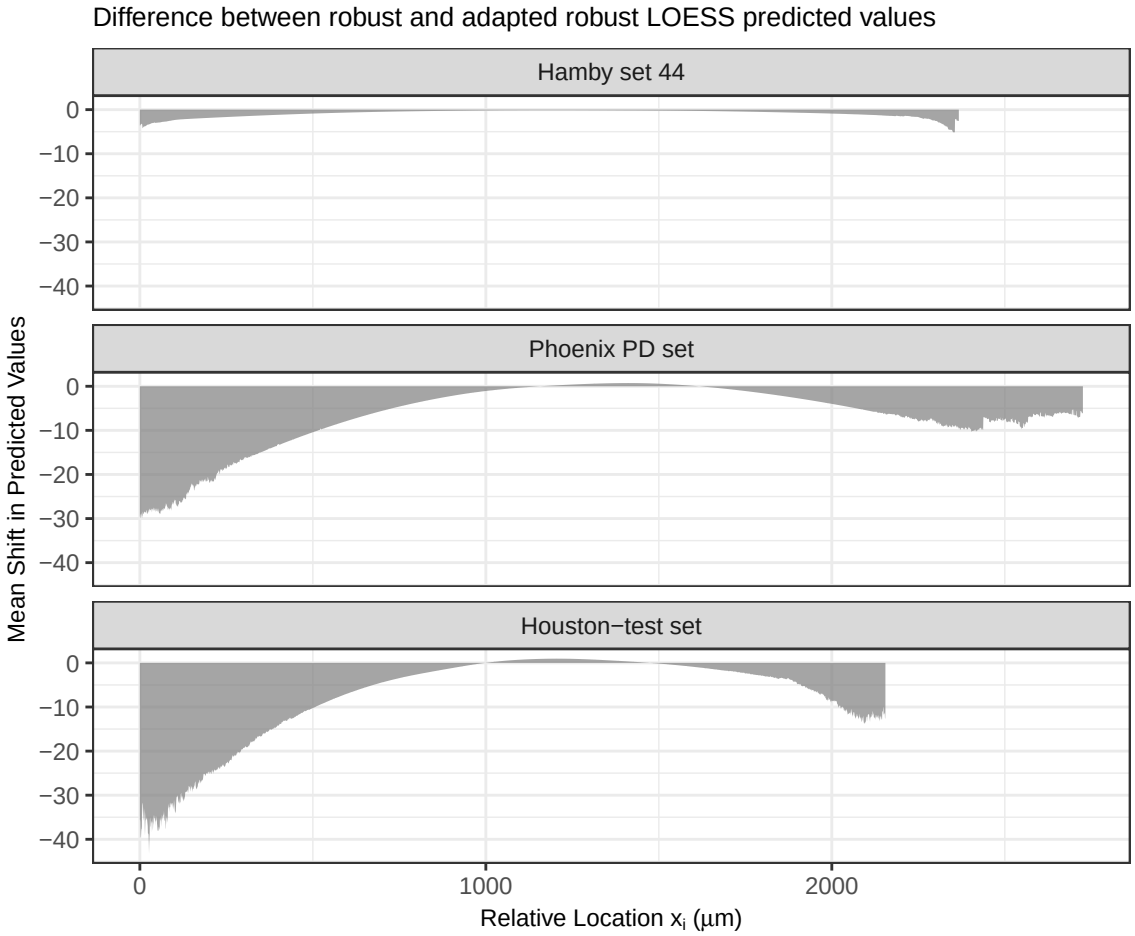

Figure 8: Mean shift in predictions when applying the adapted robust LOESS procedure in place of the traditional robust LOESS procedure. For Hamby set 44 predictions are, on average, very similar. The Phoenix PD and Houston-test sets have more significant downwards shifts in predictions near the left and right boundaries.

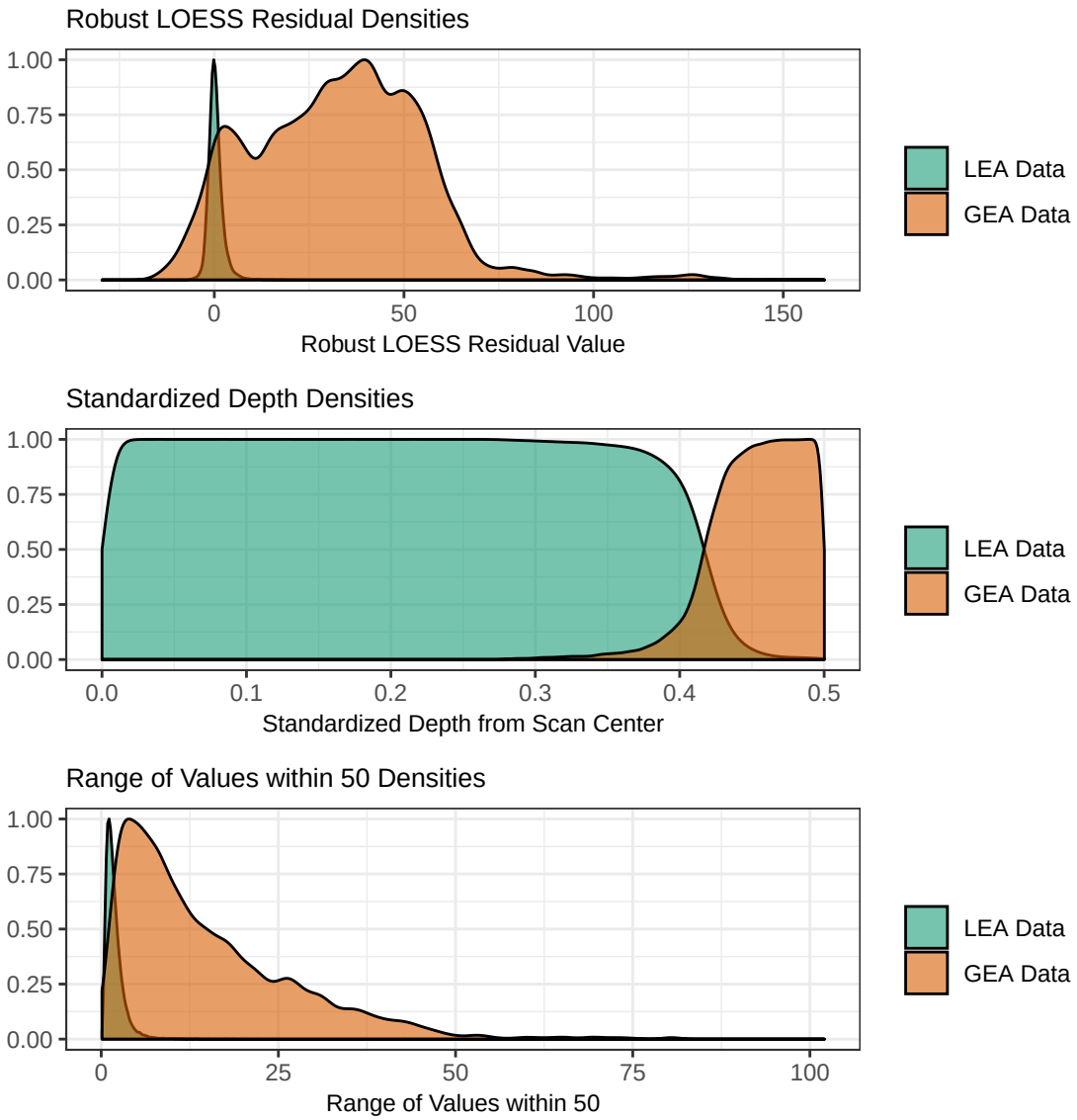

Figure 9: Example distributions of features used in two-class classification from Hamby set 44. While depth shows the most clear separation between GEA and LEA data, it alone will not suffice to classify data correctly. While other distributions are relatively tight for LEA, there is still significant overlap with the wider GEA distributions.

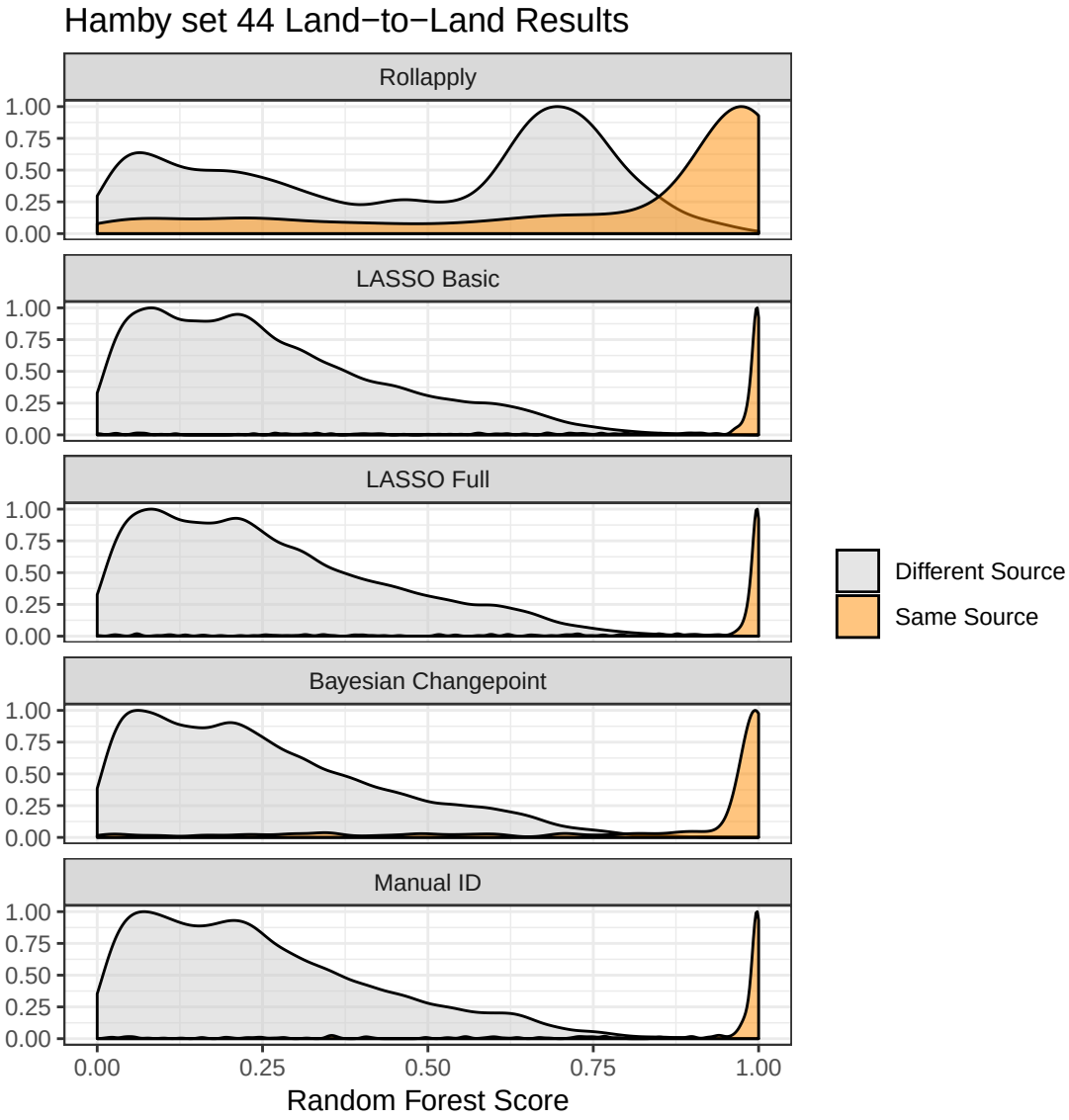

Figure 10: Random forest score distributions for same source and different source land-to-land comparisons for Hamby set 44. Distributions should ideally separate between same source and different source pairs. LASSO Basic, LASSO Full and Bayesian changepoint all demonstrate significant improvement over Rollapply.

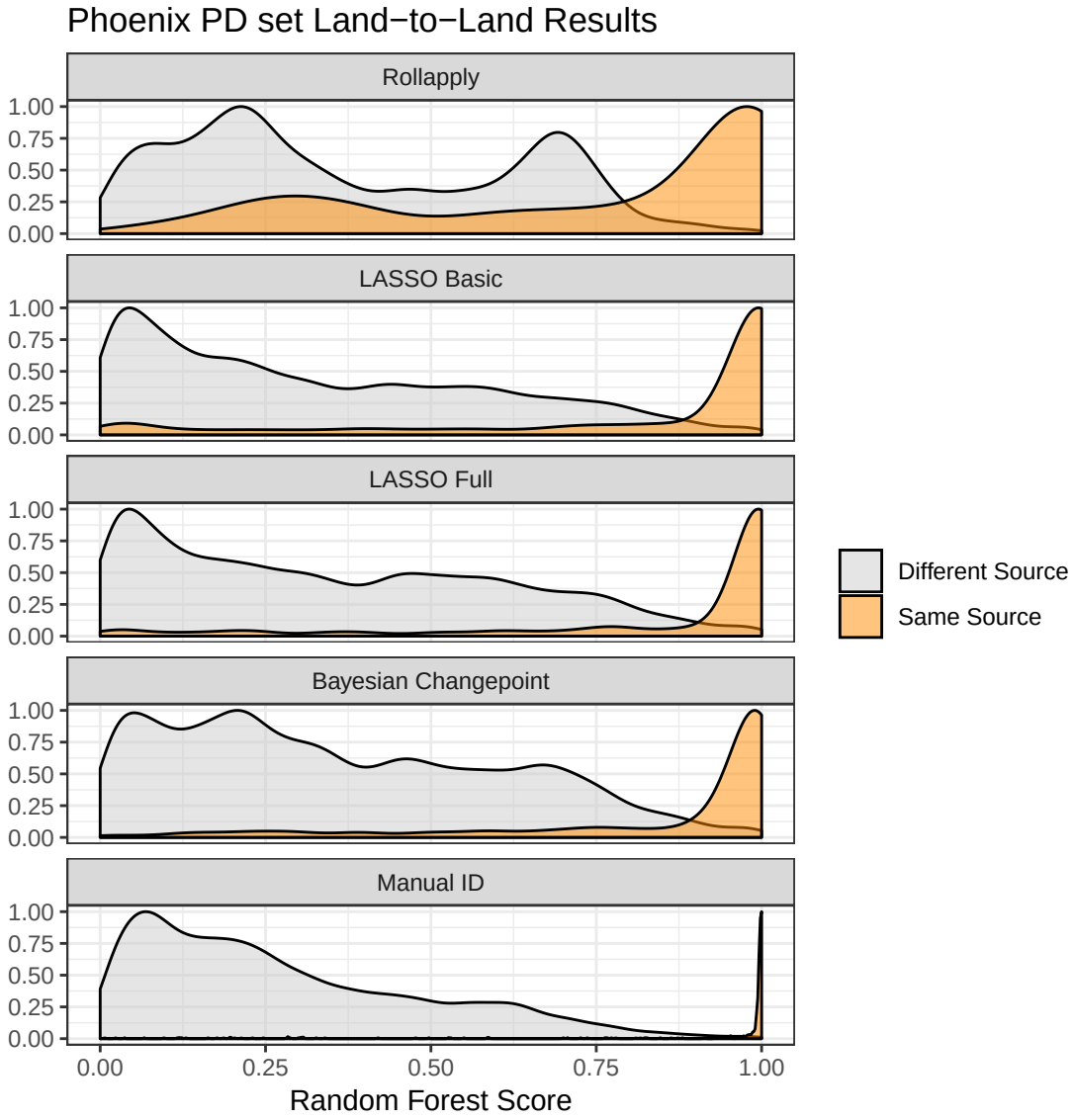

Figure 11: Random forest score distributions for same source and different source land-to-land comparisons for the Phoenix PD set. LASSO Basic, LASSO Full, and Bayesian changepoint all demonstrate significant improvement over Rollapply, but are still not as well separated as the Manual ID distributions.

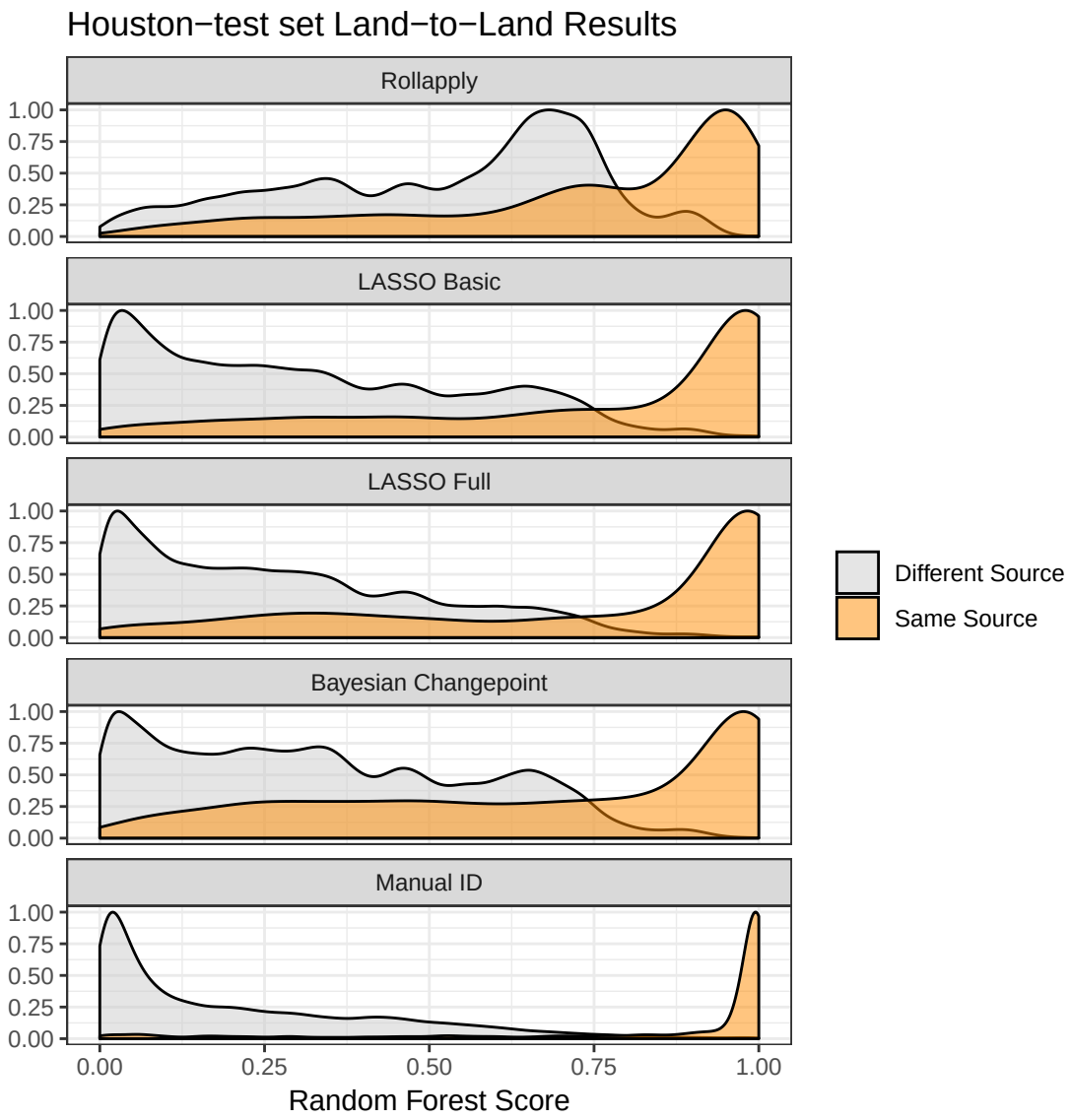

Figure 12: Random forest score distributions for same source and different source land-to-land comparisons for the Houston-test set. LASSO Basic and LASSO Full both demonstrate improvement over Rollapply, but are still not as well separated as the Manual ID distributions. Bayesian changepoint demonstrates minor improvement, but does not improve as much as the LASSO methods.

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.
LASSO basic	0.94	0.75	126	626	42098	0.99	6 min.
LASSO full	0.94	0.75	126	626	42106	0.99	6 min.
Bayesian Changepoint	0.93	0.74	152	600	42098	0.99	
Manual ID	0.94	0.74	124	628	42088	0.99	45 min.

Table 1: Land-to-land comparison results for Hamby set 44.

		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.
LASSO basic	0.877	0.947	256	486	38088	0.98	6 min.
LASSO full	0.893	0.953	238	504	38082	0.98	6 min.
Bayesian Changepoint	0.903	0.937	256	486	38080	0.98	
Manual ID	0.953	0.853	96	646	38084	0.99	45 min.

Table 2: Land-to-land comparison results for the Phoenix PD set.

		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.
LASSO basic	0.852	0.88	1274	1394	167136	0.98	12 min.
LASSO full	0.858	0.823	1144	1524	167062	0.98	12 min.
Bayesian Changepoint	0.795	0.863	1552	1116	167096	0.98	
Manual ID	0.931	0.823	614	2054	167098	0.99	75 min.

Table 3: Land-to-land comparison results for the Houston-test set.