Automated Groove Identification in 3D Bullet Land Scans (we'll change the title)

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

January 25, 2019

Abstract

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

*The authors gratefully acknowledge ...

1 Background

Images for background: bullet and LEA scan (fig 1), LEA scan to profile??

In forensic firearms analysis, visual feature comparison is used to analyze bullets to address the same source-difference source problem. Striation marks act as features which provide evidence to help determine whether two bullets were propelled through the same gun barrel. Land engraved areas (LEAs), alternating sections of the bullet that make the closest contact with the gun barrel, are the primary source of striation marks.

A cornerstone of forensic firearms analysis is that two bullets fired through the same barrel will bear more similar striation marks on their LEAs than two bullets fired from different barrels. The AFTE Theory of Identification (1) is used to make decisions when comparing two bullets under a comparison microscope: **add details here from AFTE Theory**.

Technological advances coupled with concerns about the objectivity of visual campirson have produced several image-analysis algorithms which aim to complete automated, quantitative analyses of bullet evidence.

Many of these efforts have capitalized on the introduction of high resolution 3D scanning technology to forensic science (see 2; 3; 4). 3D scans of land engraved areas have been used to develop several methods for the automated comparison of bullet lands (e.g. 5; 6; 7; 8).

In the last several decades, advances in technology have led to an increase in research focused on developing image-analysis algorithms to complete automated, quantitative analyses of bullet evidence. The main technological development that has created a pathway for image-analysis techniques is the introduction of high resolution 3D scanning technology to the field of forensic science (e.g. 2; 3; 4). These 3D data have since been used in the development of several methods of varying complexity for automated comparison of land engraved areas (e.g. 6; 7; 8).

One such method, proposed by Hare et al. (8), is a random forest algorithm based on features calculated from 2D horizontal slides of the 3D images. These slices, called profiles, have a data structure which is dominated by the global structure of the bullet land: they are curved. Feature comparison in firearms analysis is concerned with the similarity of straitions impressed by the barrel; whether two bullets are both curved does not answer

the forensic inquiry. As such, a crucial step is to remove the global structure and distill the information down the deviations from the curve. The remaining pattern of peaks and valleys - called a signature - is a much more useful representation of the striations.

While removal of a curve from data is typically a straightforward statistical problem, 3D scans of LEAs contain a unique data structure which obfuscates this task. Currently established best practice for collection of 3D images of bullet LEAs dictates that scanning begin and end slightly past the edges of the LEA, in the neighboring groove engraved areas (GEAs). This introduces a secondary structure that needs to be removed before algorithms can be reliably applied to data.

Correctly separating LEA and GEA data is one of the most challenging - and important - aspects of data pre-processing. Without removing GEA data, algorithms will be using extraneous data which can misrepresent the character of the striations present on a given LEA. In order to distinguish between these areas, we aim to identify "shoulder locations", the locations at which the LEA ends and the GEAs begin.

The method described in Hare et al. (8), based on data smoothing and local minima, fails to reliably separate the GEA data from LEA data. **Add more expanded description of two-step process to follow in the paper here**. The following work first describes a robust methodology for fitting the global curvature of 2D profiles. Three proposed methods to categorize the remaining data patterns into LEA and GEA are then described and tested.

2 Data Source

3 Methodology

We first need to remove the global structure of the bullet land.

3.1 Global Structure Removal

The non-traditional data structure necessitates employing non-traditional methods to model and remove the global structure. The data are made up of two competing structures: the LEA data, of which we would like to model the global structure, and the GEA data, which

we would like to consider as outlying data. Traditional statistical modeling techniques minimize the least squared vertical distance from each data point to a fit line; this results in undue influence by GEA points, which pull any fit lines towards their unusual points.

While bullets are traditionally circular, it is unwise to use a rigidly quadratic model to fit the global structure. We cannot assume that fired bullets will retain a neatly circular shape, especially at the level of detail scans are captured. The significant amount of physical pressure that acts upon bullets as they are fired through a barrel also can lead to some warping or slight deformations *(find a citation from JFS or AFTE about warping/deformation of bullets?)*. Finally, the placement of the land relative to the plane of reference when a 3D scan is being captured can vary slightly, meaning that the 2D crosscuts can be slightly tilted or rotated. This will not translate into a clean quadratic-shaped(?) crosscut.

To avoid the potential risks arising from using a quadratic linear model, we instead use a locally weighted regression (LOESS) which fits linear regression models on small pieces of the data and combines predictions to result in a non-parametric predicted fit of the data structure.

However, since LOESS is still rooted in traditional regression techniques, it is unable to adequately identify and address the separation between GEA and LEA data structures. To address this, we implement a robust version of LOESS which iteratively downweights unusual data points and re-fits a LOESS model to each land. This robust LOESS is an adapted version of the robust LOESS proposed by (9).

This model is fit as follows: (add more formulaic language here...)

1. Fit a LOESS model (span = 1) to an entire crosscut to predict y using values of x. Assign weights of 1 to each data point for this fitting procedure.

2. Obtain predicted values of y from the model fit in step 1.

3. Calculate residual values using the predicted y values.

4. Calculate bisquare weights for each residual value using the following formula:

$$\max(1 - (residual/(6 * MAR))^2, 0)^2,$$

where MAR is the median absolute residual for the crosscut.

5. Assign weights to each data point according to its residual value. If the residual value is positive, assign the bisquare downweight. If the residual is zero or negative, leave the weight at 1.

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

7. After k iterations of updating the weight vector, fit a LOESS model (span = 1) and obtained predicted and residual values.

The subsequent prediction methods for shoulder location are based on the residuals calculated from the fit to the global structure of each land. One method uses penalized two-class classification techniques to classify each data point into "LEA" or "GEA", while the second uses Bayesian changepoint analysis to predict the data points at which the shoulders begin on either side.

3.2 Two-Class Classification

Shoulder location can be predicted by classifying data points as one of two classes: "LEA" or "GEA". A shoulder location prediction can then be found by gathering the range of values classified as "LEA" points.

Classification into "LEA" or "GEA" was accomplished by a process of feature engineering (calculating new variables based on the residuals) and LASSO regression, a statistical learning method which penalizes the model for having many large parameters, and thus works to reduce the overall number of model parameters needed.

The following features are used in the LASSO model:

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)^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.

Using these features, two separate models were fit: "LASSO Simple", which uses each of the features listed above, and "LASSO Interactions", which uses each of the features along with pairwise interactions for each of them.

The resulting parameter values for the model fits were used to calculate predicted values between 0 and 1; the closer to 1, the higher probability of membership in the "GEA" class.

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 are many more responses in the "LEA" category than in the "GEA" category. 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.

Thus, instead of looking for the cutoff that gives raw equal error rate (where sensitivity and specificity are equal), we looked for 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". Due to the larger class size of "LEA", 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.

This process results in two methods for groove identification in the `bulletxtrctr` package: "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, the global structure is consistent and can either be described by a line with zero slope, a line with positive slope for the right GEA, or a line with negative slope for the left GEA. Finding the points where the GEAs and LEA meet is treated as a problem of model selection. That is, the best fitting statistical model, in terms of the magnitude of the likelihood, should be the one which assumes that the points at which the global structure changes align with where the GEAs and LEA meet. This approach was proposed in the more general context of Bayesian changepoint detection in Stephens (10). Thus, the points of global structural change are what we will call changepoints. Thus, our model will be defined in a piecewise fashion. In practice there are also complex additional patterns which may exist for a number of reasons, but this large scale structural assumption remains generally reasonable. The complex smaller scale patterns can be thought of as the dependence in the data after accounting for the global structure. 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. 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 (10), 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. (11). 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 a smaller number of parameters 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.

The idea behind the changepoint approach is that within either the left GEA, right GEA, or the LEA, the global structure is consistent and can either be described by a line with zero slope, a line with positive slope for the right GEA, or a line with negative slope for the left GEA. Finding the points where the GEAs and LEA meet is treated as a problem of model selection. That is, the best fitting statistical model, in terms of the magnitude of the likelihood, should be the one which assumes that the points at which the global structure changes align with where the GEAs and LEA meet. This approach was proposed in the more general context of Bayesian changepoint detection in Stephens (10). Thus, the points of global structural change are what we will call changepoints. Thus, our model will be defined in a piecewise fashion. In practice there are also complex additional patterns which may exist for a number of reasons, but this large scale structural assumption remains generally reasonable. The complex smaller scale patterns can be thought of as the dependence in the data after accounting for the global structure. 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.

3.3.3 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,

$$(Y(x_1), Y(x_2), ..., Y(x_{k_1}))^\top \sim N(\beta_{01}\mathbb{1} + \beta_{11}x_{1:k_1}, \Sigma_1(\sigma_1, \ell_1)) \tag{9}$$

$$(Y(x_{k_1+1}), Y(x_{k_1+2}), ..., Y(x_{k_2}))^\top \sim N(0, \Sigma_2(\sigma_2, \ell_2)) \tag{10}$$

$$(Y(x_{k_2+1}), Y(x_{k_2+2}), ..., Y(x_n))^\top \sim N(\beta_{02}\mathbb{1} + \beta_{12}x_{k_2+1:n}, \Sigma_3(\sigma_3, \ell_3)), \tag{11}$$

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,

$$(Y(x_1), Y(x_2), ..., Y(x_k))^\top \sim N(\beta_0 \mathbb{1} + \beta_1 x_{1:k}, \Sigma_1(\sigma_1, \ell_1)) \tag{12}$$

$$(Y(x_{k+1}), Y(x_{k+2}), ..., Y(x_n))^\top \sim N(0, \Sigma_2(\sigma_2, \ell_2)), \tag{13}$$

and the right GEA model is defined as,

$$(Y(x_1), Y(x_2), ..., Y(x_k))^\top \sim N(0, \Sigma_1(\sigma_1, \ell_1)) \tag{14}$$

$$(Y(x_{k+1}), Y(x_{k+2}), ..., Y(x_n))^\top \sim N(\beta_0 \mathbb{1} + \beta_1 x_{k+1:n} \Sigma_2(\sigma_2, \ell_2)). \tag{15}$$

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

$$(Y(x_1), Y(x_2), ..., Y(x_n))^\top \sim N(0, \Sigma(\sigma, \ell)). \tag{16}$$

We see that estimating the changepoint locations also involves selecting the most appropriate model. 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.

3.3.4 Bayesian Model Estimation

As was noted in Stephens (10), 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. (11). 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 a smaller number of parameters 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

5 Conclusions

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

10. 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.

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