FINAL REPORT

Robust Detection, Discrimination, and Remediation of UXO: Statistical Signal Processing Approaches to Address Uncertainties Encountered in Field Test Scenarios

SERDP Project MR-1663

JANUARY 2012

Leslie M. Collins, Ph.D.
Duke University

This document has been cleared for public release

SERDP
DOD ▪ EPA ▪ DOE

This report was prepared under contract to the Department of Defense Strategic Environmental Research and Development Program (SERDP). The publication of this report does not indicate endorsement by the Department of Defense, nor should the contents be construed as reflecting the official policy or position of the Department of Defense. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Department of Defense.

Robust Detection, Discrimination, and Remediation of UXO: Statistical Signal Processing to Address Uncertainties Encountered in Field Test Scenarios MR-1663 Final Report

Performing Organization: Duke University, Department of Electrical & Computer Engineering

Principal Investigator: Leslie M. Collins, Ph.D.

Response to Comments on Final Report Submitted December 13, 2011

January 3, 2012

Following are the responses to the comments on the draft final report for MM-1663, Robust Detection, Discrimination, and Remediation of UXO: Statistical Signal Processing to Address Uncertainties Encountered in Field Test Scenarios, submitted on December 13, 2011:

- Page viii: FUDS = **Formerly** Used Defense Sites.
    - The correction has been made.

- Page 6, 2nd paragraph: In next to last line, remove is between that and provides.
    - The correction has been made.

- Page 21, 2nd paragraph: Figure 16 should be Figure 18.
    - The correction has been made.

- Page 30: Is this section based on MetalMapper data from SLO? If so, please state that. If
- not, please describe the sensor and data set to which this section's discussion applies.
    - The first sentence has been revised to indicate that this section does use the MetalMapper data from SLO. The revised text is:
      A generalized model for the EMI sensor response was developed to support the model inversion studies of the MetalMapper data collected as part of the SLO demonstration.

- Page 39: In the 3rd line, correct the spelling of revealed.
    - The correction has been made.

Table of Contents

Table of Contents i
List of Tables iii
List of Figures iv
List of Acronyms viii
Keywords ix
Acknowledgements ix
Abstract 1
Objectives 1
Technical Approach 1
Results 1
Benefits 2
Objective 3
Background 4
Robust Target Classification with Limited Training Data 6
Classifier Sensitivities Related to the Minimum P_{FA} at P_D=1 Performance Goal 6
Classifiers for Small Training Sets 13
Bootstrap Aggregation of Classifiers (Bagging) 17
Adaptive Kernel RVM 20
Techniques to Modify the Training Set 22
Remove Overly Influential Data Points 22
Modify Prior Assumptions Regarding Target Class Distributions 25
Improving Performance Prediction at P_D=1 28
Efficient and Robust Model Inversion 30
Sensor Response Model 30
Model Inversion 31
Information-Theoretic Inversion Augmentation 34
Multiple Instance Learning 37
Results and Discussion 39

Conclusions and Implications for Future Research 39
Literature Cited 41

List of Tables

Table 1: Parameters defining the Gaussian clusters for the example binary classification problem. 6

List of Figures

Figure 1: Training data for the example binary decision classification problem. 7
Figure 2: RVM decision surface resulting from the training data shown in Figure 1 for the example binary decision classification problem. 8
Figure 3: Training data for the example binary decision classification problem superimposed on the resulting RVM decision surface. 9
Figure 4: Decision boundary ($\beta = 0.5$) perfectly separating the training data for the H_0 and H_1 classes. 9
Figure 5: ROC for the training data, showing perfect separation of H_0 and H_1 classes at the threshold $\beta = 0.5$. 9
Figure 6: Decision boundary ($\beta = 0.45$) perfectly separating the training data for the H_0 and H_1 classes, and providing a greater margin for H_1. 10
Figure 7: ROC for the training data, showing perfect separation of H_0 and H_1 classes at the threshold $\beta = 0.45$. 10
Figure 8: Decision boundary ($\beta = 0.45$) imperfectly separating testing data for the H_0 and H_1 classes. 11
Figure 9: ROC for the training data, showing imperfect separation of H_0 and H_1 classes ($P_{FA} = 0.02$, $P_D = 0.985$) at the threshold $\beta = 0.45$. 11
Figure 10: Decision boundary (β = 1e-12) required to correctly find all of the H_1 targets. 12
Figure 11: ROC for the training data, showing performance ($P_{FA} = 0.585$, $P_D = 1.0$) at the threshold required to correctly find all of the H_1 targets (β = 1e-12). 12
Figure 12: Scatter plot of the MetalMapper features from the San Luis Obispo demonstration study (provided by Snyder Geophysics) utilized in the studies of classifier robustness. P_{0T} is the transverse polarizability [$P_{0T} = (P_{0y}+P_{0z})/2$] and P_{0R} is the polarizability ratio [$P_{0R} = P_{0x}/P_{0T}$]. 14
Figure 13: Pictorial representation of 2-folds cross-validation. The data is randomly divided into 2 groups (or folds). One fold is used for training, while the other is used for testing, and vise versa. This process results in two estimates of performance. The entire process is repeated N times to generate distributions of the performance metrics. 16
Figure 14: Example classifier performance evaluation. The KSD pdf estimate of P_{FA} at $P_D = 1$ is shown in the grey line (vertical axis directed to the left), the maximum P_{FA} at $P_D = 1$ is denoted by the blue square, and the mean P_{FA} at $P_D = 1$ is denoted by the green circle. 17
Figure 15: Example classifier decision surfaces for 4 example bags with an RVM classifier. Notice the variations in the UXO and clutter decision regions depending on the bootstrap samples selected for training. 18

Figure 16: Example classifier decision surfaces for an RVM classifier without (left) and with (right) classifier bagging. The bagged classifier was generated using 4 bags. 19
Figure 17: Classifier performance without and with classifier bagging for the $DLRT_3$ (top-left), GMM-GLRT (top-right) and RVM (bottom-center). In each panel, performance without bagging is shown on the left and performance with bagging is shown on the right. 20
Figure 18: Example RVM classifier decision surfaces with a fixed kernel width (left) and adaptive kernel width (right). 21
Figure 19: RVM classifier performance a fixed kernel width (left) and adaptive kernel width (right). 22
Figure 20: Illustration of the concepts of influence and leverage. Top-left: A data point with high leverage and low influence. Top-right: A data point with high leverage and high influence. Bottom-center: A data point with low leverage and high influence. 23
Figure 21: Example RVM decision surfaces with overly influential training data points removed. 24
Figure 22: ROCs corresponding to the decision surfaces presented in Figure 21. 25
Figure 23: Example GMM-GLRT decision surfaces using both clutter and UXO training data (top-left), using only UXO training data (top-right), and using only clutter training data (bottom-center). 26
Figure 24: Classifier performance as a function of the training data utilized (UXO and clutter, UXO only, or clutter only) for the $DLRT_3$ classifier (left) and the GMM-GLRT classifier (right). 27
Figure 25: Classifier performance as a function of the training data utilized (UXO and clutter, UXO only, or clutter only) for the bagged $DLRT_3$ classifier (left) and the bagged GMM-GLRT classifier (right). 27
Figure 26: Pictorial representation of 2-folds cross-validation. The data is randomly divided into 2 groups (or folds). One fold is used for training, while the other is used for testing, and vise versa. This process results in two estimates of performance. The decisions statistics may also be aggregated across folds to generate a single performance estimate. 28
Figure 27: Classifier performance comparing individual cross-validation folds and aggregate cross-validation folds for the bagged $DLRT_3$ (top-left), bagged GMM-GLRT (top-right), and bagged RVM (bottom-center). 29
Figure 28: Sensor response model block diagram. The sensor's transmitter and receiver geometry are defined independently of the target response model. 1) The primary field at the transmitter is calculated. 2) The primary field is propagated from the transmitter to the target. 3) The interaction of the incident primary field on the target with the target is calculated. 4) The resultant secondary field after interaction

with the target is propagated back to the receiver. 5) The receiver response to the secondary field is calculated. 30
Figure 29: Classification performance for RVM (blue line) and bagged GMM-GLRT (green line) classifiers with features (decay rates) derived from inverting the simple model. Top-left: Model inversion assuming the target is a BoR. Top-right: Model inversion assuming the target is a BoR, and including the GoF parameter as a feature. Bottom-left: Model inversion without assuming the target is a BoR. Bottom-right: Model inversion without assuming the target is a BoR and including the GoF parameter as a feature. 32
Figure 30: Classification performance for RVM (blue line) and bagged GMM-GLRT (green line) classifiers with features (resonant frequencies) derived from inverting the full model. Top-left: Model inversion assuming the target is a BoR. Top-right: Model inversion assuming the target is a BoR, and including the GoF parameter as a feature. Bottom-left: Model inversion without assuming the target is a BoR. Bottom-right: Model inversion without assuming the target is a BoR and including the GoF parameter as a feature. 33
Figure 31: .Example application of Fisher Information to augment least squares model inversion to estimate the parameters of a Gaussian pulse. Example least square objective function surface (top-left), in which preferred solutions have smaller sum of errors. Example Fisher Information surface (top-right), in which preferred solutions have higher FI. Model fits (bottom-center). The green curve more closely approximates the true parameters used to generate the measured data. Least squares alone selected the orange curve. Least squares augmented with FI selects the purple curve. 35
Figure 32: Classification performance for RVM (blue line) and bagged GMM-GLRT (green line) classifiers with features (resonant frequencies) derived from inverting the full model and applying Fisher Information to guide model parameter selection. Top-left: Model inversion assuming the target is a BoR. Top-right: Model inversion assuming the target is a BoR, and including the GoF parameter as a feature. Bottom-left: Model inversion without assuming the target is a BoR. Bottom-right: Model inversion without assuming the target is a BoR and including the GoF parameter as a feature. 36
Figure 33: Graphical depiction of multiple instance learning. All measurements, termed “instances” for a given target are placed in a “bag.” Each instance consists of an N-dimensional feature vector. 37
Figure 34: Classification performance for ppmm MIL with kMeans clustering and an SVM classifier using features (decay rates) derived from inverting the simple model. Left: Model inversion without assuming the target is a BoR. Right: Model inversion

without assuming the target is a BoR, and including the GoF parameter as a feature. 38

List of Acronyms

AUC ..... Area Under the ROC Curve
BoR ..... Body of Revolution
BRAC ..... Base Realignment and Closure
$DLRT_N$ ..... Distance Likelihood Ratio Test using N neighbors
EM ..... Expectation-maximization
EMI ..... Electromagnetic Induction
FI ..... Fisher Information
FUDS ..... Formerly Used Defense Sites
GoF ..... Goodness-of-Fit
GLRT ..... Generalized Likelihood Ratio Test
GMM ..... Gaussian Mixture Model
GMM-GLRT ..... Generalized Likelihood Ratio Test with Gaussian Mixture Model distributions
H_0 ..... Null Hypothesis (Clutter Class)
H_1 ..... Alternate Hypothesis (UXO Class)
KNN ..... k-Nearest Neighbor
KSD ..... Kernel Smoothing Density [pdf estimate]
LS ..... Least squares
MIL ..... Multiple Instance Learning
P_{0R} ..... Polarizability Ratio
P_{0T} ..... Transverse Polarizability
P_{0x} ..... Polarizability in the x-direction
P_{0y} ..... Polarizability in the y-direction
P_{0z} ..... Polarizability in the z-direction
Pdf ..... probability density function
P_D ..... Probability of Detection
P_{FA} ..... Probability of False Alarm
ppmm ..... p-posterior mixture model
SERDP ..... Strategic Environmental Research and Development Program
ROC ..... Receiver Operating Characteristic [curve]
RVM ..... Relevance Vector Machine
Rx ..... Receiver
SLO ..... San Luis Obispo, California Demonstration Study
Tx ..... Transmitter
UXO ..... Unexploded Ordnance

Keywords

Unexploded ordnance, classifier sensitivity, model inversion.

Acknowledgements

The authors gratefully acknowledge the support of the Strategic Environmental Research and Development Program (SERDP) (Project MM-1663).

Abstract

Objectives

The objective of this work was to develop methodologies that will allow the human analyst to be removed from the processing loop. It has been shown in a number of recent demonstrations that when the most skilled practitioners process geophysical data, select data chips for analysis, select features for classification, select one of a suite of classifiers, and manually tune the classifier boundaries, excellent classification performance can be achieved. Here, we aim to develop techniques to improve target characterization and reduce classifier sensitivity to imprecision in the target characterizations, thereby reducing the need for an expert human analyst.

Technical Approach

The technical approach focuses on two main areas of research: 1) robust automated model inversion and 2) robust target classification with limited training data. The efficacy of information-theoretic measures to improve model inversion robustness is investigated. Specifically, Fisher Information is investigated as a mechanism to select from among multiple candidate feature sets that all model the measured data similarly well. Multiple instance learning, a machine learning technique that facilitates learning the features that are indicative of the class of interest even if those features are not present for every measurement for the class of interest, is also investigated. With this more sophisticated machine learning approach to classification, a simpler, and potentially more robust inversion can be implemented. Classifiers that are robust with limited training data are investigated through sensitivity studies. Several different classifiers are considered, as well as techniques to modify the training data set by removing some training data points that may be less informative to the classifier. In addition, the potential impacts of the cross-validation method are investigated.

Results

The classifier sensitivity studies revealed that the more robust classifiers tended to have decision surfaces that gradually transition from decision statistics that are strongly indicative of UXO to decision statistics that are strongly indicative of clutter. These classifiers tended to produce moderate decision statistics in the vicinity of the UXO clusters, thereby allowing test UXO that may have features somewhat different from the training UXO to be assigned decision statistics that are not strongly indicative of clutter. The model inversion investigations revealed that although information-theoretic approaches do provide some benefit, they are not necessarily consistent in doing so. Multiple Instance Learning, however, appears to provide a substantial computational benefit, in that it can attain performance similar to that obtained with the full model inversion, but with a small fraction of the computation time.

Benefits

Given a performance goal of minimizing P_{FA} at $P_D = 1$, there are two important aims: 1) ensuring consistent UXO characterization via features, and 2) ensuring the chosen classifier is insensitive to UXO target features that may lie somewhat outside the cluster of most UXO features. The first aim ensures that when UXO are characterized in training, that characterization can be repeated in testing, even if site conditions vary. The second aim ensures that if, for some reason, a UXO target's characterization is not completely consistent with previously observed UXO, the classifier still produces a decision statistic which allows for the target to be classified as UXO (i.e., the decision statistic is not strongly indicative of clutter). Both of these aims reduce risk by improving the quality UXO characterization via features and reducing classifier sensitivity to the precision of estimated UXO features.

Objective

The objective of the work described here was to develop methodologies that will allow the human analyst to be removed from the processing loop. It has been shown in a number of recent demonstrations that when the most skilled practitioners process geophysical data, select data chips for analysis, select features for classification, select one of a suite of classifiers, and manually tune the classifier boundaries, excellent classification performance can be achieved. Generalizing this performance to the entire contractor community will require procedures that are more automated and standardized, and this is the research direction we pursued in this effort.

Over the last several years, modern geophysical techniques have been developed that merge more sophisticated sensors, underlying physical models, statistical signal processing algorithms, and adaptive training techniques. These new approaches have dramatically reduced false alarm rates, although for the most part they have been applied to data collected at sites with relatively benign topology and anomaly densities (e.g. [1, 2, 3]). Current fielded and demonstrated UXO classification strategies are also constrained by the use of significant human interaction to hand-select data, computationally inefficient and poorly parameterized model inversion strategies, and the limited availability of UXO and site-specific clutter data for supervised training. Additionally, in actual cleanup scenarios there are be a variety of challenges, driven by physical constraints in the data collection, the presence of native clutter, and natural UXO placement, all of which will deleteriously impact performance if ignored [4]. Therefore, there are still fundamental research issues that must be addressed under conditions more representative of actual cleanup scenarios to enable development of automated, field-ready, and robust UXO classification strategies.

This effort has two basic research thrust areas that are focused on developing a robust data-to-decision processing architecture that removes the expert human analyst from the loop. The foci of the research are to: (1) investigate elements necessary for robust model inversion techniques: alternate objective functions for model inversion and automated information-theoretic channel selection prior to model inversion; and (2) investigate issues that impact robustness in classifier design, such as overtraining due to the limited availability of UXO and clutter data for supervised learning, class membership ambiguity for some training samples, and techniques to mitigate the potential challenges associated with estimating classifier performance at the desired performance point of $P_D = 1$. These issues are investigated primarily using sensor data collected by advanced sensors at the former Camp San Luis Obispo, CA demonstration, specifically the Geometrics MetalMapper sensor, to determine the benefits provided by automated, principled, approaches to robust model inversion and classifier design.

Background

There are many areas in the United States and throughout the world that are contaminated or potentially contaminated with unexploded ordnance. In the United States alone there are 1900 Formerly Used Defense Sites (FUDS) and 130 Base Realignment and Closure (BRAC) installations that need to be cleared of UXO. Using current technologies, the cost of identifying and disposing of UXO in the United States is estimated to range up to $500 billion. Site specific clearance costs vary from $400/acre for surface UXO to $1.4 million/acre for subsurface UXO [5]. These approaches, however, usually require significant amounts of human analyst time, and thus those additional costs, which are currently necessary parts of ongoing demonstrations, are not factored into these numbers. Thus, there is a clear need to effectively and cost-efficiently remediate UXO contaminated lands using *automated* procedures, rendering them safe for their current or intended civilian uses. Development of new UXO detection technologies with improved data analysis has been identified as a high priority requirement for over a decade.

Several sensor modalities have been explored for the detection and identification of surface and buried UXO. These include electromagnetic induction (EMI), magnetometry, radar, and seismic sensors. These sensors generally experience little difficulty detecting UXO, thus detection does not create the bottleneck that results in the high cost of remediating sites. The primary contributor to the costs and time associated with remediating a UXO-contaminated site is the high false-alarm rate caused by the significant amount of non-UXO clutter and shrapnel typically found on battlefields and military ranges.

On sites where anomalies are well separated, statistical signal processing algorithms that exploit recent advances in sensor design and phenomenological modeling have been successfully employed and substantial improvements in performance over traditional “mag and flag” approaches have been demonstrated [1, 2, 6, 7, 8, 9, 10, 11]. Recent results from the Camp Sibert demonstration clearly demonstrate that good classification can be effected, but to attain this performance level significant human interaction and a fairly benign demonstration scenario were required. Results from the JPG-V study indicate performance may not attain desired goals in actual cleanup scenarios where the controlled conditions of earlier tests (e.g. JPG-IV) cannot be guaranteed. In the JPG-V study, a decrease in classification performance was attributed to the unanticipated sensor positional uncertainty and lack of knowledge about the clutter objects because of limited training data [4]. Such conditions will clearly be present in many actual cleanup scenarios, although position uncertainty is less of an issue for multi-axis sensors. Additionally, the classification algorithms need to be effective in highly-cluttered environments that contain anomalies in close proximity with each other, resulting in overlapping target and clutter signatures. The statistical signal processing algorithms developed to date have not been robust under these conditions. Rigorous testing is required

for validation and algorithms developed to date will most likely require refinement or more sophisticated approaches to perform well in diverse conditions. Development of a robust UXO classification system will require a principled approach to optimizing all signal processing aspects in the system. Classifier design, training methods, and the model inversion process which produces features for classification and segmentation of the sensor data all need to be formulated carefully.

Robust Target Classification with Limited Training Data

Classifier Sensitivities Related to the Minimum P_{FA} at P_D=1 Performance Goal

Commonly, an "average value" performance metric, such as **a**rea **u**nder the ROC **c**urve (AUC), is utilized to assess classifier performance. "Average value" performance metrics are often fairly stable across training/testing data set realizations and typically do not exhibit sensitivity to the precise location of the transition between H_0 (null hypothesis) and H_1 (alternate hypothesis) decision regions or the slope of the decision statistic between decision regions. While there is typically some variation in performance across training/testing data sets, the variability is often fairly concentrated about a mean performance value. The P_{FA} at $P_D = 1$ performance metric, however, is an "extreme value" performance metric. As such, it may be unstable across training/testing data set realizations and often may exhibit high sensitivity to the precise location of the transition between decision regions or the slope of the decision statistic between decision regions. In addition, performance across training/testing sets may be quite variable, sometimes spanning nearly the entire range of possible performance values. Clearly, the choice of operational goal (i.e., maximizing AUC, minimizing P_{FA} at P_D=1, as well as others) may influence the overall classifier design

Suppose the chosen classifier provides a bounded decision statistic between 0 and 1, with a decision statistic of 0 indicating strong belief that the object under test belongs to the null hypothesis (H_0) class and a decision statistic of 1 indicating strong belief that the object under test belongs to the alternate hypothesis (H_1) class. In this application, the null hypothesis corresponds to the clutter class, and the alternate hypothesis corresponds to the UXO class. A relevance vector machine (RVM) is one example of a classifier that provides bounded decision statistics.

Suppose that the classification problem under consideration is a binary classification decision using two features, with a known set of training data. An example set of training data consisting of two Gaussian clusters defined by the parameters in Table 1 is shown in Figure 1. The blue circles represent the H_0 data and the red squares represent the H_1 data. In this example, there is clear separation between the two classes; one can easily imagine a curve drawn from the upper-left to the lower-right which separates the features space into H_0 and H_1 regions. In fact, the data shown here are linearly separable, meaning a straight line can be drawn between the two clusters to separate them.

Table 1: Parameters defining the Gaussian clusters for the example binary classification problem.

	H_0 cluster	H_1 cluster
Mean	$[-2.0 \quad 0.0]$	$[0.0 \quad 2.0]$
Covariance	$\begin{bmatrix} 1.0 & -0.7 \\ -0.7 & 1.0 \end{bmatrix}$	$\begin{bmatrix} 1.0 & -0.7 \\ -0.7 & 1.0 \end{bmatrix}$

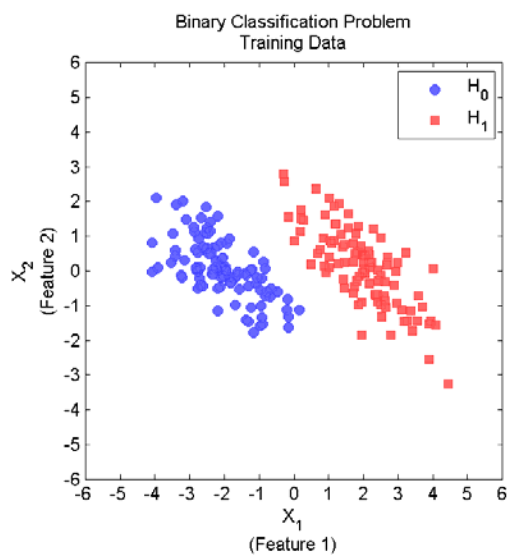

Figure 1: Training data for the example binary decision classification problem.

Given a set of available training data, such as that shown in Figure 1, the RVM parameters (optimized for that particular set of training data) can be determined. Figure 2 shows the RVM decision surface resulting from the example training data displayed in Figure 1. In this image, the color white represents a decision statistic of 1, the color yellow represents a decision statistic of 0.5, and the color dark blue represents a decision statistic of 0. There is a dark blue region corresponding to the H_0 cluster (blue circles) and a white region corresponding to the H_1 cluster, (red squares) and a thin yellow strip separating the two regions. This decision surface suggests that the RVM is functioning as desired; it is identifying the locations of the H_0 and H_1 clusters and provides a nonlinear boundary (the yellow strip) between the two regions.

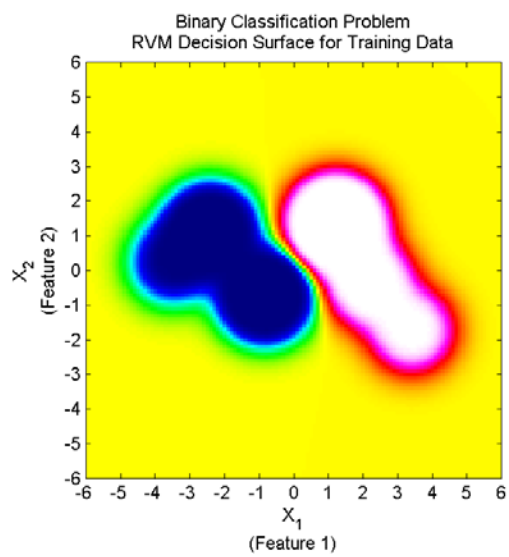

Figure 2: RVM decision surface resulting from the training data shown in Figure 1 for the example binary decision classification problem.

The training data is superimposed on the RVM decision surface in Figure 3 to illustrate the relationship between the data and the decision surface. This figure clearly demonstrates that the RVM provides perfect separation of the H_0 and H_1 classes. All the blue (H_0) data points are located in the blue to green portion of the decision surface and all the red (H_1) data points appear in the orange to white portion of the decision surface. A decision boundary corresponding to a decision threshold of $\beta = 0.5$ is drawn in Figure 4 (teal line). It follows the yellow strip separating the blue and red regions in the decision surface and perfectly classifies the data. The corresponding ROC is displayed in Figure 5, with the operating point for a decision threshold of $\beta = 0.5$ denoted by the green asterisk ($P_{FA} = 0.0$, $P_D = 1.0$).

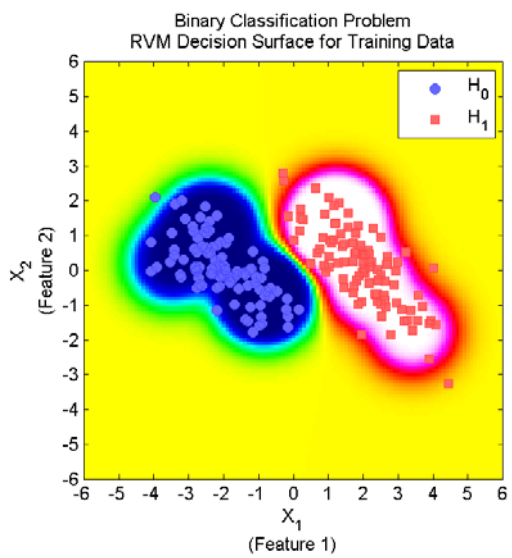

Figure 3: Training data for the example binary decision classification problem superimposed on the resulting RVM decision surface.

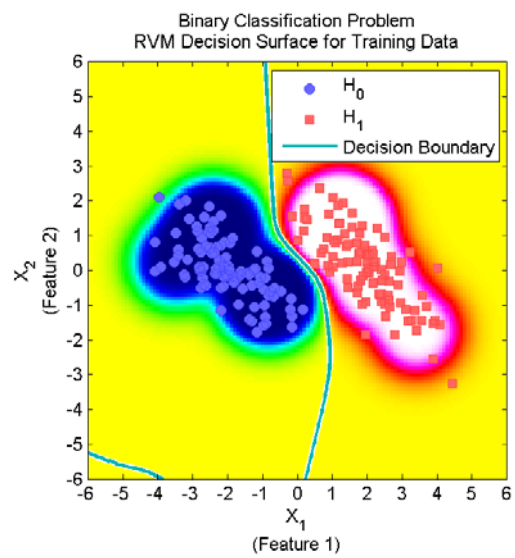

Figure 4: Decision boundary ($\beta = 0.5$) perfectly separating the training data for the H_0 and H_1 classes.

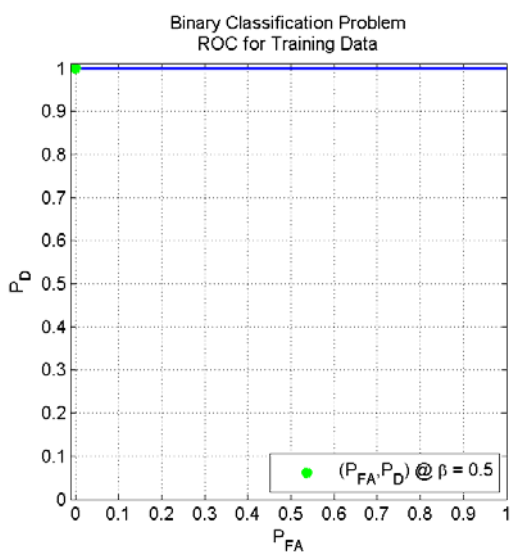

Figure 5: ROC for the training data, showing perfect separation of H_0 and H_1 classes at the threshold $\beta = 0.5$.

For the desired operational goal of minimizing P_{FA} at $P_D = 1$, a slightly lower decision threshold may be chosen so as to allow for correct detection of H_1 targets that may fall somewhat outside the H_1 decision region inferred from the training data. In this example, the minimum decision statistic associated with H_1 targets is approximately 0.55. Choosing a decision threshold of $\beta = 0.45$ would allow for a greater margin around H_1, while still perfectly classifying the training

data. The RVM decision surface and decision boundary for a threshold of $\beta = 0.45$ are shown in Figure 6. Notice that the decision boundary is moved away from the H_1 cluster and toward the H_0 cluster, but still closely follows the yellow strip in the decision surface between the H_0 and H_1 regions. The ROC and the corresponding operating point for a decision threshold of $\beta = 0.45$ denoted by a green asterisk ($P_{FA} = 0.0$, $P_D = 1.0$) follow in Figure 7. These figures demonstrate that lowering the decision threshold to $\beta = 0.45$ moves the decision boundary further away from the majority of the H_1 targets, thereby providing a margin of safety for detecting H_1 targets, and maintains perfect classification performance for the training data.

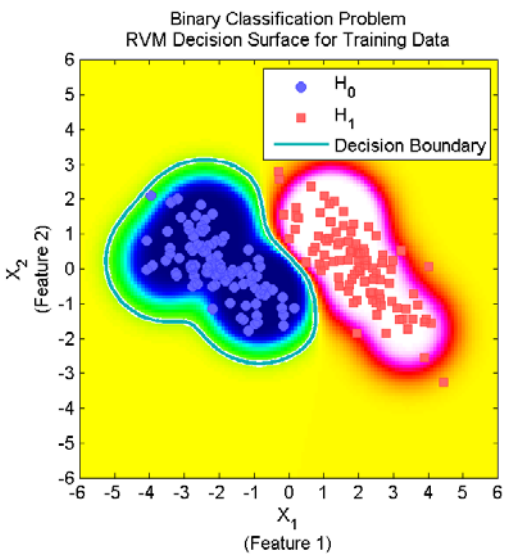

Figure 6: Decision boundary ($\beta = 0.45$) perfectly separating the training data for the H_0 and H_1 classes, and providing a greater margin for H_1.

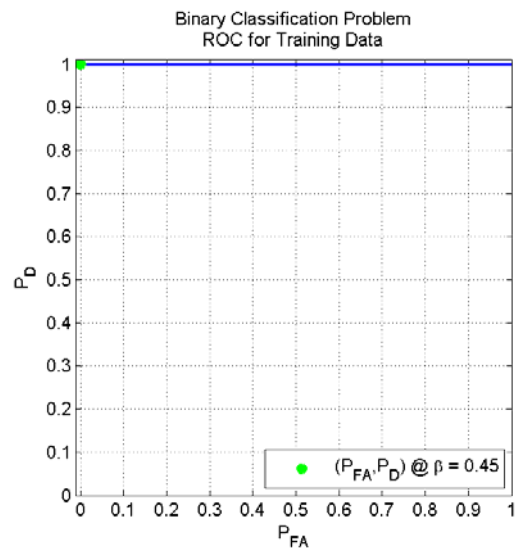

Figure 7: ROC for the training data, showing perfect separation of H_0 and H_1 classes at the threshold $\beta = 0.45$.

In practice, a classifier is developed (designed and trained) using training data, and then put to practice on separate testing data sets. The data that the classifier operates on in practice, termed testing data, is not the same data that was used to determine its parameters (the training data), so the performance in practice may differ from the performance found with the training data.

A second set of data, representing testing data, was generated according to the parameters in Table 1. Thus, the training and testing data were generated by exactly the same process. These testing data are shown in Figure 8 superimposed on the RVM decision surface previously determined using the training data. Also shown is the decision boundary corresponding to a decision threshold of $\beta = 0.45$, the decision boundary previously determined to provide a margin of safety for H_1 target detection. Even though the training data were generated by exactly the same process as the testing data, classification performance on the testing data is

not perfect (P_{FA} = 0.02, P_D = 0.985); there are 2 H_0 data points that fall in the H_1 decision region (causing false alarms) and 3 H_1 data points that fall within the H_0 decision region (causing missed detections). More importantly, detection performance at this decision threshold (β = 0.45) is less than 1. The ROC curve for the testing data is shown in Figure 9, along with the operating point corresponding to a decision threshold of β = 0.45. Although the ROC is still quite good as measured by the AUC metric, performance is significantly degraded as measured by the P_{FA} at P_D = 1 performance metric; P_{FA} at P_D = 1 has increased from P_{FA} = 0.00 for the testing data to P_{FA} = 0.585 for the training data.

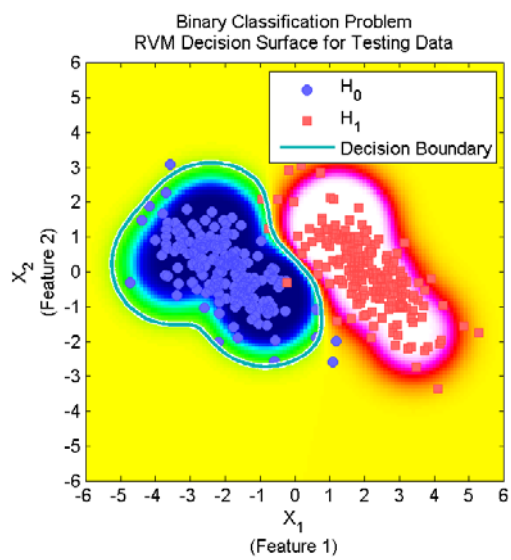

Figure 8: Decision boundary (β = 0.45) imperfectly separating testing data for the H_0 and H_1 classes.

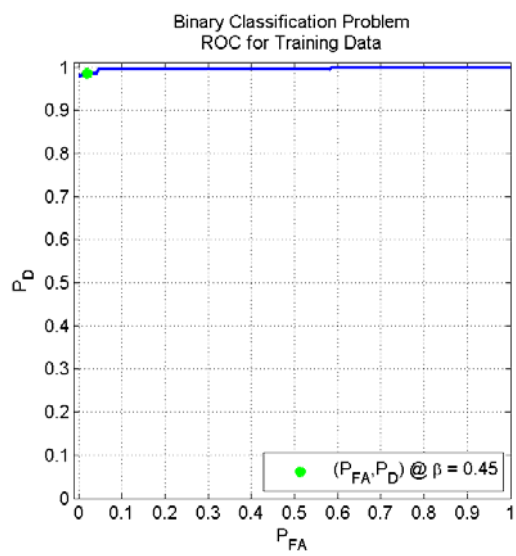

Figure 9: ROC for the training data, showing imperfect separation of H_0 and H_1 classes (P_{FA} = 0.02, P_D = 0.985) at the threshold β = 0.45.

The decision boundary corresponding to the decision threshold required to achieve P_D=1, β = 1e-12, is shown in Figure 10. Clearly, this decision threshold leaves many H_0 targets outside the H_0 decision region (the interiors of the two circular boundaries drawn on the dark blue portion of the decision surface), thus producing numerous false alarms. This is reflected in the operating point on the ROC, shown in Figure 11, where the minimum P_{FA} for P_D = 1 is 0.585.

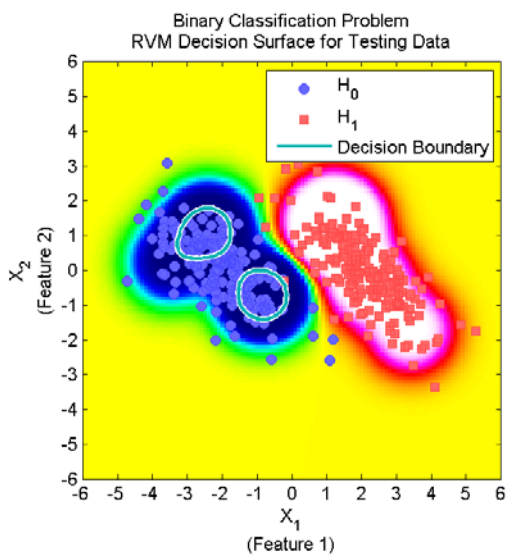

Figure 10: Decision boundary (β = 1e-12) required to correctly find all of the H_1 targets.

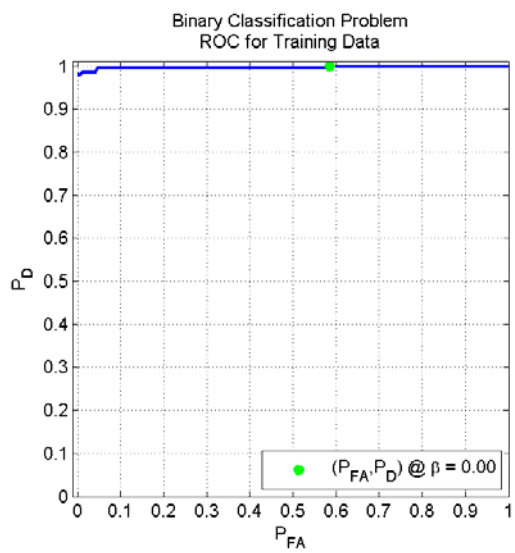

Figure 11: ROC for the training data, showing performance (P_{FA} = 0.585, P_D = 1.0) at the threshold required to correctly find all of the H_1 targets (β = 1e-12).

This example illustrates a potential downfall of relying on a small data set to fully characterize the feature distributions for the H_0 and H_1 classes. In this example, the training data set resulted in a classifier and decision boundary that perfectly separated the H0 and H1 classes, but when the same classifier and decision boundary were applied to a testing data set generated by the same stochastic process, less than perfect performance was achieved. This example illustrates that to minimize the P_{FA} at P_D = 1 performance metric, it will be preferable to design classifiers such that any UXO targets that are reasonably close to the UXO target clusters are not assigned very low decision statistics that are highly indicative of clutter.

Classifiers for Small Training Sets

As demonstrated in the preceding example, classifier performance depends in part on the similarity of the statistical characterizations of the training and testing data sets. Due to the limited amount of data available in small training data sets, they may not provide a statistical characterization of the data that is optimal for a given testing data set or robust across all possible testing sets. This motivates investigating classifiers which are robust to the uncertainties inherently associated with characterizing the data using small training data sets.

The data utilized in this study to assess the characteristics of classifiers which make them robust to small training data sets are MetalMapper features from the San Luis Obispo, California demonstration study (SLO) provided by Snyder Geophysics. The specific features utilized are the transverse polarizability [$P_{0T} = (P_{0y}+P_{0z})/2$] and the polarizability ratio [$P_{0R} = P_{0x}/P_{0T}$]. The feature set was limited to two features to facilitate visualizing the decision surfaces and better understanding the relationship between the classifier decision surface characteristics and classifier robustness. A scatter plot of the selected features is shown in Figure 12. The features for the UXO targets cluster fairly well, though there are a few UXO targets that lie somewhat outside the main clusters. For example, the UXO target with features (P_{0T},P_{0R}) of about (0,-0.7) is outside the main clusters of UXO features. Due to its distance from the UXO clusters, this target is an example of a target which is likely to be among the last UXO targets detected, thereby potentially contributing to a larger number of false alarms before reaching $P_D = 1$. The classifier decision surface determines how many false alarms occur before reaching the final UXO target. As will be shown in the following sections, a decision surface that quickly transitions from a high decision statistic in the vicinity of the UXO feature clusters to a low decision statistic outside those clusters is more likely to result in a high P_{FA} at $P_D = 1$ than a decision surface that more gradually transitions to a low decision statistic outside the UXO clusters. The reason for this is a sharper transition to a low decision statistic outside the UXO clusters results in it being more likely that a UXO target somewhat outside the UXO clusters will have a very low decision statistic (i.e., a clutter-like decision statistic). In contrast, a more gradual transition to a low decision statistic outside the UXO feature clusters results in it being likely that a UXO target near, but outside, the UXO clusters will have a more moderate decision statistic (i.e., a moderate-value decision statistic not strongly associated with either UXO or clutter).

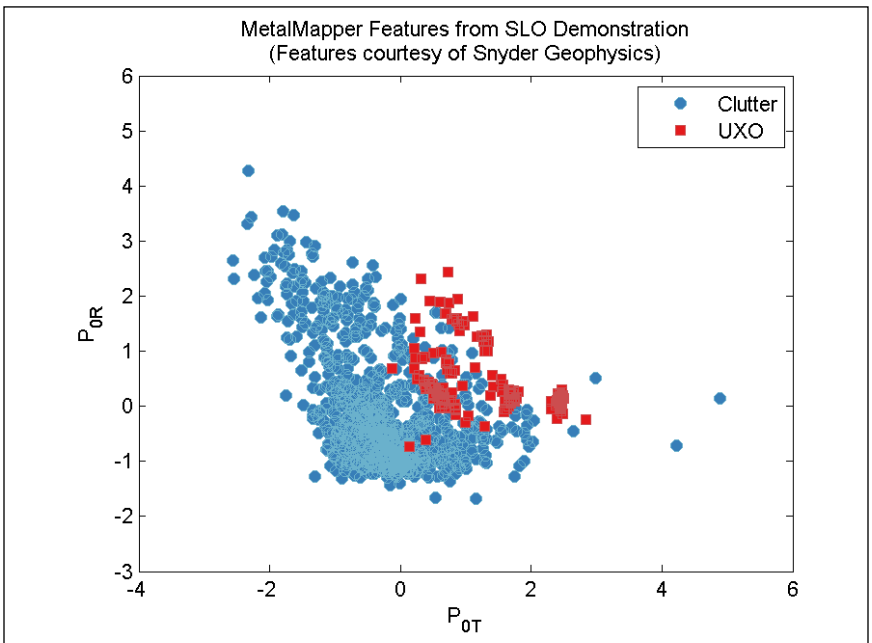

Figure 12: Scatter plot of the MetalMapper features from the San Luis Obispo demonstration study (provided by Snyder Geophysics) utilized in the studies of classifier robustness. P_{0T} is the transverse polarizability [$P_{0T} = (P_{0y}+P_{0z})/2$] and P_{0R} is the polarizability ratio [$P_{0R} = P_{0x}/P_{0T}$].

The classifiers considered in this study are the distance likelihood ratio test (DLRT) [12], the generalized likelihood ratio test with Gaussian mixture models for the distributions (GMM-GLRT), and the relevance vector machine (RVM) [13, 14].

The DLRT is an approximation to the likelihood ratio test with the required probability density functions estimated via K-nearest neighbors (KNN) density estimation, and is given by [12]

$$\hat{\lambda}(\mathbf{x}) = \log\left(\frac{n_{H_0}}{n_{H_1}}\right) + D\left[\log\left(\Delta_{K^{(0)}}\right) - \log\left(\Delta_{K^{(1)}}\right)\right], \tag{1}$$

where n_{H_i} is the number of training data points associated with hypothesis H_i, D is the dimensionality of the feature space, and $\Delta_{K^{(i)}}$ is the distance to the kth neighbor from hypothesis H_i. Here, the distance Δ is measured using Euclidean distance. In practice, a monotonic function of the DLRT,

$$\hat{\lambda}'(\mathbf{x}) = \log\left(\Delta_{K^{(0)}}\right) - \log\left(\Delta_{K^{(1)}}\right), \tag{2}$$

is used to calculate the decision statistic for this classifier.

The GMM-GLRT is an approximation to the likelihood ratio test where the required probability density functions are represented by Gaussian mixture models. It is given by

$$\hat{\lambda}(\mathbf{x}) = \frac{\sum_{n=1}^{N} \rho_n \times \frac{1}{(2\pi)^{k/2} |\mathbf{\Sigma}_n|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_n)^T \mathbf{\Sigma}_n^{-1} (\mathbf{x} - \boldsymbol{\mu}_n) \right)}{\sum_{m=1}^{M} \rho_m \times \frac{1}{(2\pi)^{k/2} |\mathbf{\Sigma}_m|^{1/2}} \exp\left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_m)^T \mathbf{\Sigma}_m^{-1} (\mathbf{x} - \boldsymbol{\mu}_m) \right)}, \tag{3}$$

where each Gaussian in the mixture model is parameterized by a mean $\boldsymbol{\mu}$ and a covariance $\mathbf{\Sigma}$. The Gaussians are mixed according to the weights ρ, where $\sum_{n=1}^{N} \rho_n = 1$ and $\sum_{m=1}^{M} \rho_m = 1$. The sets of model parameters for each hypothesis, $\boldsymbol{\mu}$, $\mathbf{\Sigma}$, and $\boldsymbol{\rho}$, are estimated from the training data via expectation-maximization (EM).

The RVM is a kernel-based classifier that provides a non-linear decision boundary [13, 14]. It selects a (typically small) number of training data points, termed relevant vectors, to be the locations where the kernels are centered. Each kernel $\phi_n(\mathbf{x})$ has an associated weight w_n, and the weighted sum of the kernels produces the decision surface. Given a set of kernels, located at the relevant vectors (kernel centers), the RVM decision surface is given by

$$\lambda(\mathbf{x}) = \sum_{n=1}^{N} w_n \phi_n(\mathbf{x}). \tag{4}$$

The techniques considered to improve classifier robustness with small training data sets are evaluated through simulations based on the aforementioned MetalMapper features from SLO. The simulations consist of repeated 2-folds cross-validation. This approach results in 536 data points in each fold (443 clutter targets and 93 UXO targets). The cross-validation approach is presented pictorially in Figure 13. The 1072 data points are randomly divided into 2 groups (or folds), ensuring that the clutter and UXO classes are represented proportionally in each group. One fold is used for training, while the other is used for testing, and vise versa. This process results in two estimates of performance. The entire process is repeated N times, and the 2N performance estimates are then used to generate distributions of the performance metrics.

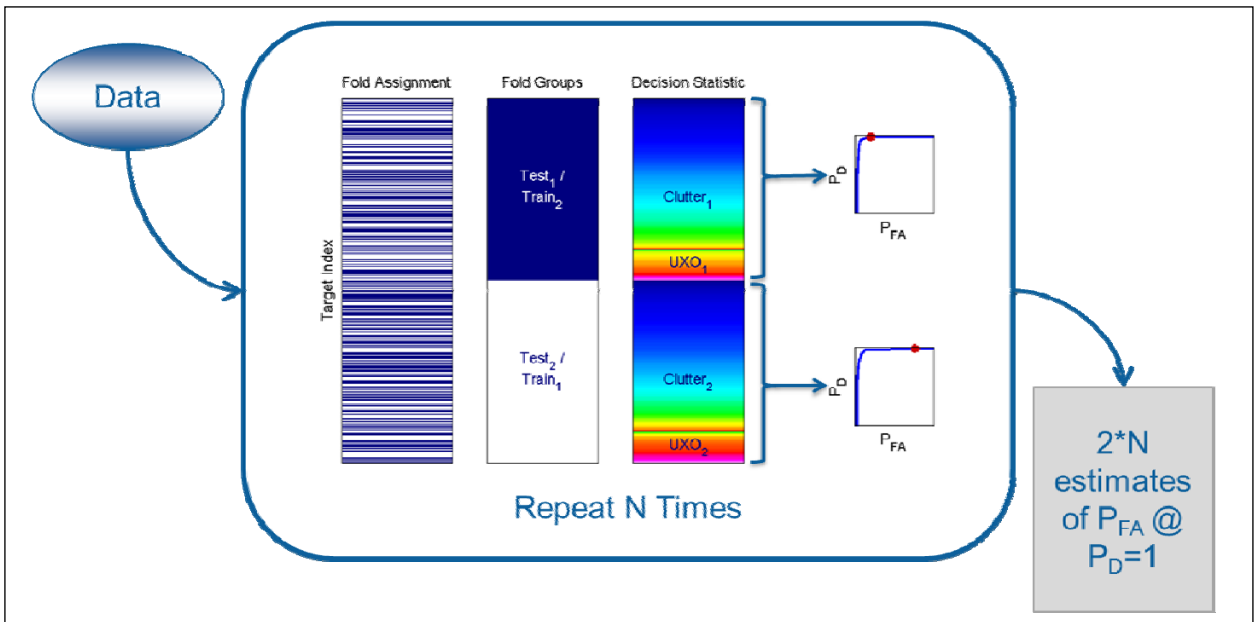

Figure 13: Pictorial representation of 2-folds cross-validation. The data is randomly divided into 2 groups (or folds). One fold is used for training, while the other is used for testing, and vise versa. This process results in two estimates of performance. The entire process is repeated N times to generate distributions of the performance metrics.

Performance is evaluated using the mean P_{FA} at $P_D = 1$, the maximum P_{FA} at $P_D = 1$, and the kernel smoothing density (KSD) estimate of the probability density function (pdf) of P_{FA} at $P_D = 1$. An example classifier performance evaluation is shown in Figure 14. The KSD pdf estimate of P_{FA} at $P_D = 1$ is shown in the grey line, with the vertical axis directed to the left. The maximum P_{FA} at $P_D = 1$ is denoted by the blue square, and the mean P_{FA} at $P_D = 1$ is denoted by the green circle. The goal is to minimize P_{FA} at $P_D = 1$, so lower values of P_{FA} at $P_D = 1$ are preferred. The maximum P_{FA} at $P_D = 1$ provides a measure of the worst-case scenario; it is estimated to be the worst (highest) minimum P_{FA} at $P_D = 1$. The KSD pdf estimate serves to display the range and distribution of P_{FA} at $P_D = 1$ values obtained through the cross-validation procedure. With this, the consistency of classifier performance can also be assessed visually. For example, notice in the example performance evaluation shown in Figure 14 the distribution of P_{FA} at $P_D = 1$ is bimodal, with the majority of samples around 0.25 and 1. In this example, considering only the mean value of P_{FA}, even concurrently with the standard deviation, would not reveal the fairly large number of samples of P_{FA} near 1 at $P_D = 1$, and one could come to erroneous conclusions regarding the P_{FA} at $P_D = 1$ performance of this classifier across testing data sets.

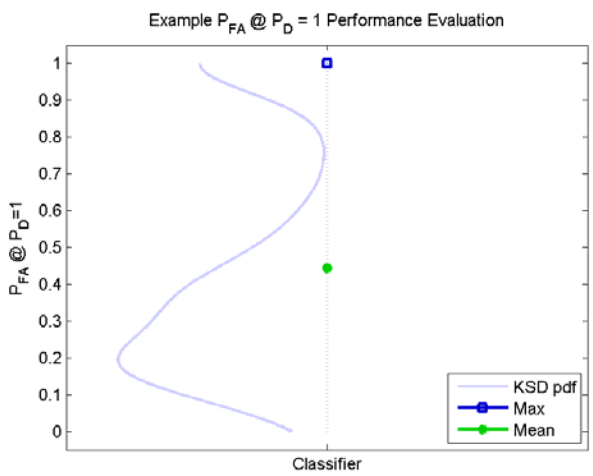

Figure 14: Example classifier performance evaluation. The KSD pdf estimate of P_{FA} at P_D = 1 is shown in the grey line (vertical axis directed to the left), the maximum P_{FA} at P_D = 1 is denoted by the blue square, and the mean P_{FA} at P_D = 1 is denoted by the green circle.

Bootstrap Aggregation of Classifiers (Bagging)

Bootstrap aggregation of classifiers (Bagging) is a technique in which a number of classifiers are developed, each using a bootstrap sample with replacement of the data, and the final classifier decision statistic is the average of the decision statistics from each of the individual bootstrapped classifiers. Bagged classifiers tend to bring about consistency in situations where classifier performance can be highly variable, such as when employing an "extreme value" performance measure like P_{FA} at P_D = 1. Thus, bagged classifiers have the potential to improve consistency of performance when the operational goal is to minimize P_{FA} at P_D = 1.

The development of an example bagged classifier, using the RVM classifier, is shown in Figure 15 and Figure 16. Figure 15 shows four example classifier decision surfaces for different bootstrap samples of the training data. Notice the variations in the UXO and clutter decision regions depending on the bootstrap samples selected for training. These example decision surfaces demonstrate the potential sensitivity of a classifier to the specific data points that are available for training. All four of these classifiers are then averaged to produce the bagged classifier decision surface shown in the right panel in Figure 16.

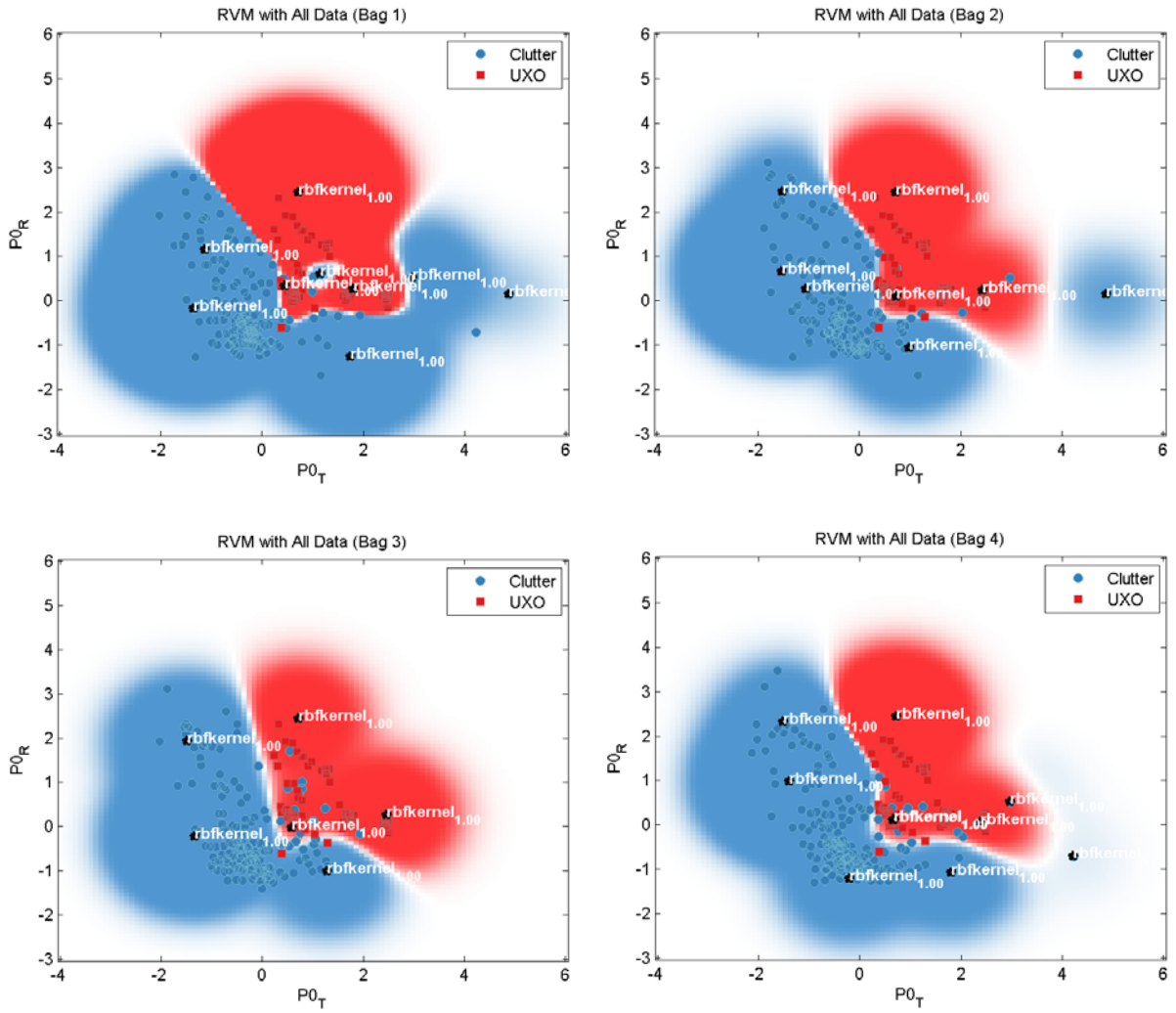

Figure 15: Example classifier decision surfaces for 4 example bags with an RVM classifier. Notice the variations in the UXO and clutter decision regions depending on the bootstrap samples selected for training.

The bagged RVM classifier decision surface is compared to the conventional RVM classifier decision surface in Figure 16. Even with only four bootstrapped classifiers contributing to the bagged classifier, the softening of the decision boundary (the white strip between the blue and red regions) is already evident. The conventional RVM decision surface (left) shows a fairly sharp transition between the clutter and UXO decision regions. In contrast, the bagged RVM decision surface (right) shows a more gradual transition between the two regions in some areas, as evidence by the light pink and light blue colors that appear adjacent to the boundary between the blue (clutter) and red (UXO) regions. With a larger number of bootstrap samples in the bagged classifier, the transition in the decision statistic between the two decision regions becomes more gradual in areas where there is some overlap between the features for the two classes, but remains a sharp transition in the regions where there is good separation between the two classes.

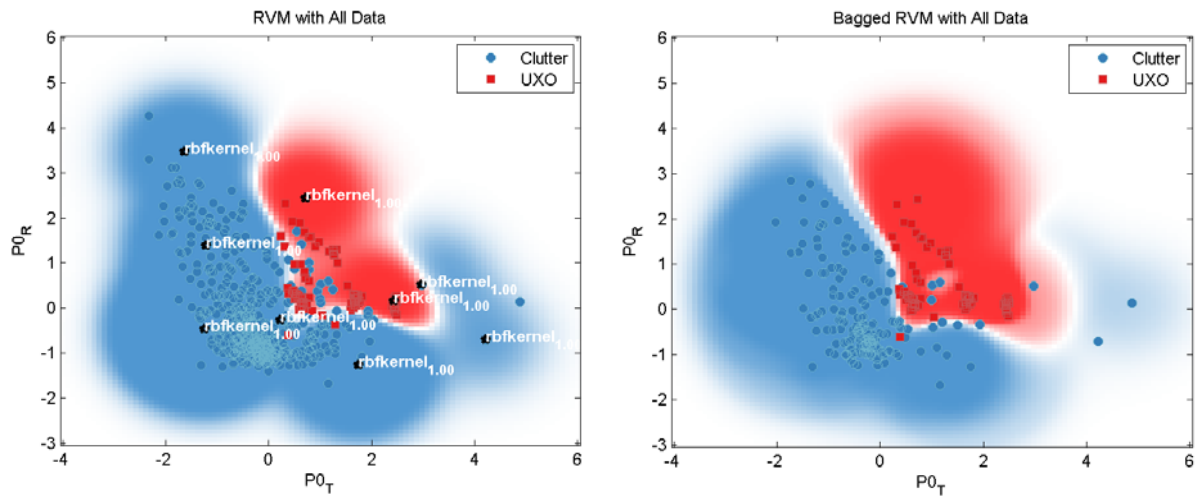

Figure 16: Example classifier decision surfaces for an RVM classifier without (left) and with (right) classifier bagging. The bagged classifier was generated using 4 bags.

For each of the three classifier types considered in this study ($DLRT_3$, GMM-GLRT, and RVM), bagging the classifier improved performance by reducing the maximum P_{FA} at $P_D = 1$ obtained over all folds. Classifier performance is shown in Figure 17. The GMM-GLRT classifier generally provides the best performance, and also exhibits the most significant performance improvement with bagging; the maximum P_{FA} at $P_D = 1$ improved from about 0.85 to about 0.48, which is significantly better than the maximum P_{FA} at $P_D = 1$ of about 0.9 achieved with the $DLRT_3$ and the RVM. Overall, there are not significant differences in the distributions of the P_{FA} at $P_D = 1$ performance measure between the unbagged and bagged classifiers, other than a slight downward shift in the distributions.

It is believed that the bagged GMM-GLRT provides much better performance than other classifiers because it is approximating an optimal Bayesian solution, under the assumption that the data for both classes is properly modeled by a Gaussian mixture model. Given that the GMM is a flexible model capable of modeling a wide-variety of data distributions, the assumption that it is a good model for the data is likely not a poor assumption. If the likelihood ratio were formulated with the number of Gaussian mixture components along with their means and covariances as uncertain parameters, the optimal likelihood ratio (under the assumption that the Gaussian mixture model is the proper model for the data) would integrate over the uncertainty in the GMM parameters under each hypothesis (number of mixture components, and each component's mean and covariance). This integration can be well-approximated using Monte Carlo integration, which is a technique in which the integral is well-approximated by a sum of random samples of the integrand. Often, prior distributions for the uncertain parameters are assumed, and the integrand is sampled according to the assumed priors. However, the distributions over the uncertain parameters may also be estimated

empirically from the data. In this case, each bootstrap sample from the data provides an empirical estimate of the uncertain random parameters. Developing a classifier based on the empirical estimates from a single bootstrap sample constitutes randomly sampling the integrand. Averaging all the classifiers developed from all the bootstrap samples of the data (which lead to random samples of the integrand) is equivalent to numerically integrating over the uncertain parameters via Monte Carlo integration with empirical estimates of the priors on the uncertain parameters. Thus, the bagged GMM-GLRT classifier provides a numerical estimate of an optimal Bayesian solution under the assumption that a GMM is a proper model for the data.

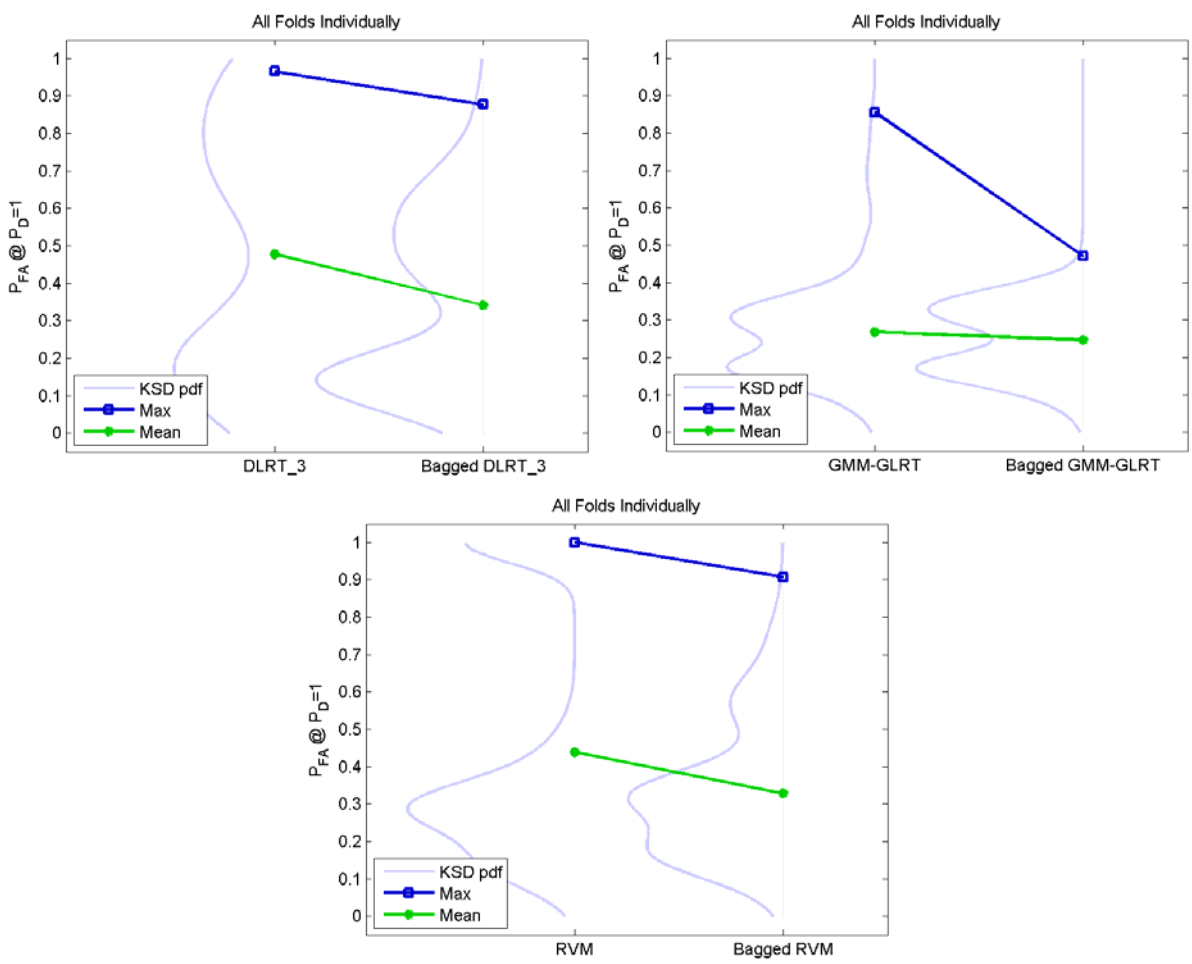

Figure 17: Classifier performance without and with classifier bagging for the $DLRT_3$ (top-left), GMM-GLRT (top-right) and RVM (bottom-center). In each panel, performance without bagging is shown on the left and performance with bagging is shown on the right.

Adaptive Kernel RVM

The RVM classifier typically uses a pre-defined kernel, or set of kernels. The radial basis function is a common choice of kernel, and has a parameter specifying the kernel width. If the

feature clusters are larger than the kernel width, then multiple kernels can be used to represent the kernel. On the other hand, if the feature clusters are smaller than the kernel width, then even just a single kernel may not represent tightly clustered features very well. This motivates investigating the utility of an adaptive kernel RVM, where the kernel width is inversely proportional to the local density of the data. With this approach, a region in which the local density of the data is low will be represented by a kernel with a large width, and a region in which the local density of the data is high will be represented by a kernel with a small width. Adaptive kernel width may allow the RVM to better represent the tightly clustered data associated with the UXO targets.

Example decision surfaces are shown in Figure 18 for the RVM classifier with a fixed kernel width (left) and an adaptive kernel width (right). These example decision surfaces show the RVM with the adaptive kernel width does, in fact, place smaller kernels at the UXO clusters. In addition, the boundary between the clutter and UXO classes is more clearly defined with the adaptive kernel RVM than the fixed kernel RVM.

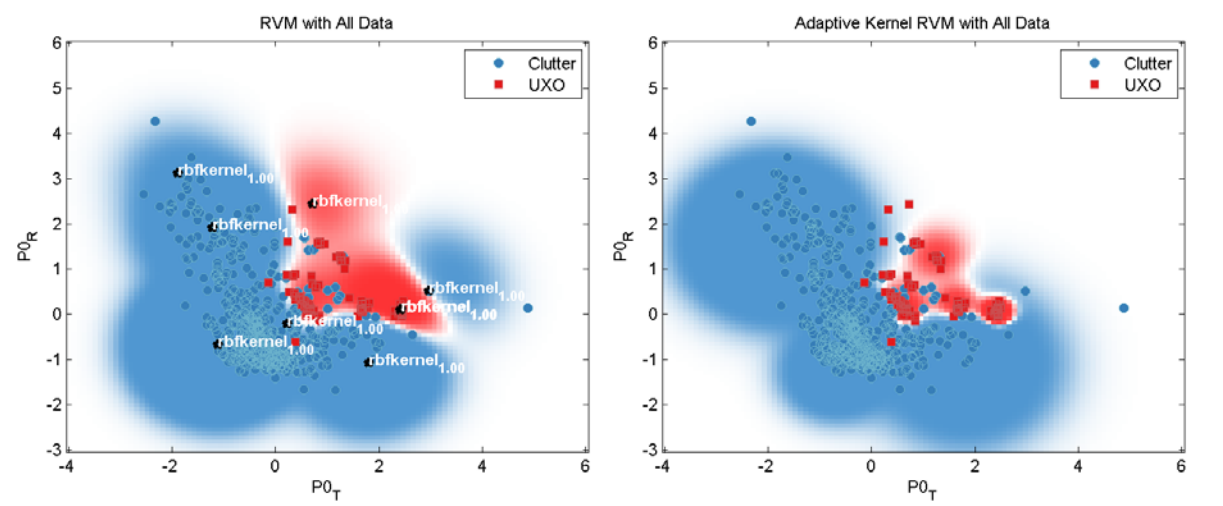

Figure 18: Example RVM classifier decision surfaces with a fixed kernel width (left) and adaptive kernel width (right).

Performance for the fixed kernel and adaptive kernel RVM are shown in Figure 19. The choice of a fixed or adaptive kernel width does not impact classifier performance. Neither the maximum nor the mean P_{FA} at $P_D = 1$ is affected by the choice of kernel width. In addition, the KSD pdf estimates of the distributions are also largely unaffected by the choice of kernel width.

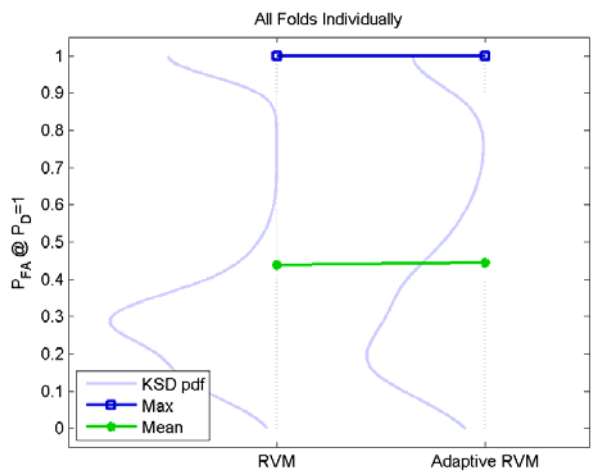

Figure 19: RVM classifier performance a fixed kernel width (left) and adaptive kernel width (right).

Techniques to Modify the Training Set

A second line of inquiry into techniques to improve the robustness of classifier trained with small training data sets is to investigate methods to modify the training set to remove overly influential or potentially unreliable training data points.

Remove Overly Influential Data Points

In general, influence is the degree to which a data point affects a regression result. In the case of classifiers, influence is the degree to which a data point affects the resulting classifier decision surface. Leverage is the potential for the data point to influence the regression result. Generally, a data point has high leverage if it is in a region where there are no, or few, additional data points. The concepts of influence and leverage are illustrated in Figure 20 for the example of linear regression. The true underlying line is shown in the dotted red line. The noisy data are represented by the open blue circles. The linear regression to the noisy data is shown in the solid red line. The top-left panel illustrates a data point with high leverage, but low influence. It is the only data point in the vicinity of x=15 giving it high leverage, but it does not significantly affect the regression result, meaning it has low influence. The top-right panel illustrates a data point with high leverage and high influence. In this case, this single data point significantly alters the regression result, meaning it has high influence. This type of data point is often referred to as an outlier. The bottom-center panel illustrates a data point with low leverage and high influence. Here, the highlighted data point is in the vicinity of many other data points, giving it little leverage. However, it is different enough from the other data points near x=0 that it alters the regression result in a meaningful way, so it has high influence. This type of data point is often referred to as an inlier. These examples illustrate the potential for a single data point with high influence to significantly alter the regression result. Similarly, a

small number of unreliable data points the training set may have the potential to significantly affect the resulting classifier decision surface.

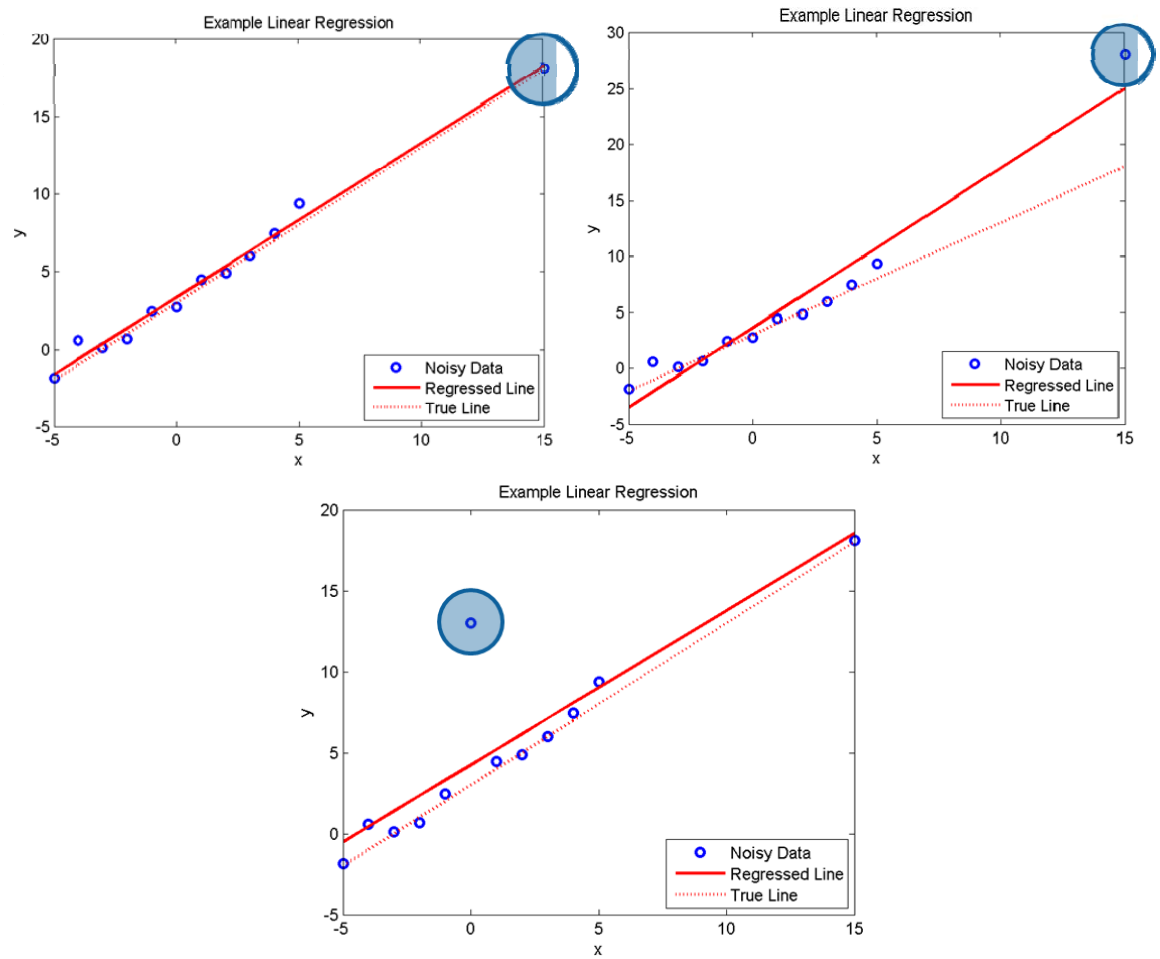

Figure 20: Illustration of the concepts of influence and leverage. Top-left: A data point with high leverage and low influence. Top-right: A data point with high leverage and high influence. Bottom-center: A data point with low leverage and high influence.

Example decision surfaces resulting from training an RVM with overly influential training data points removed are shown in Figure 21. The top-left panel shows the RVM decision surface with all training data included, as a reference. The top-right panel shows the RVM decision surface with both influential clutter and UXO training points removed. Removing influential clutter and UXO data points tends to make the decision surface more decisive – the decision statistics tend to be either strongly indicative of clutter or strongly indicative of UXO. There are no moderate decision statistics where the UXO and clutter classes overlap. The bottom-center panel shows the RVM decision surface when only influential clutter training points are removed. The approach results in a smooth decision boundary between the two classes, with the UXO decision region encompassing most of the UXO targets.

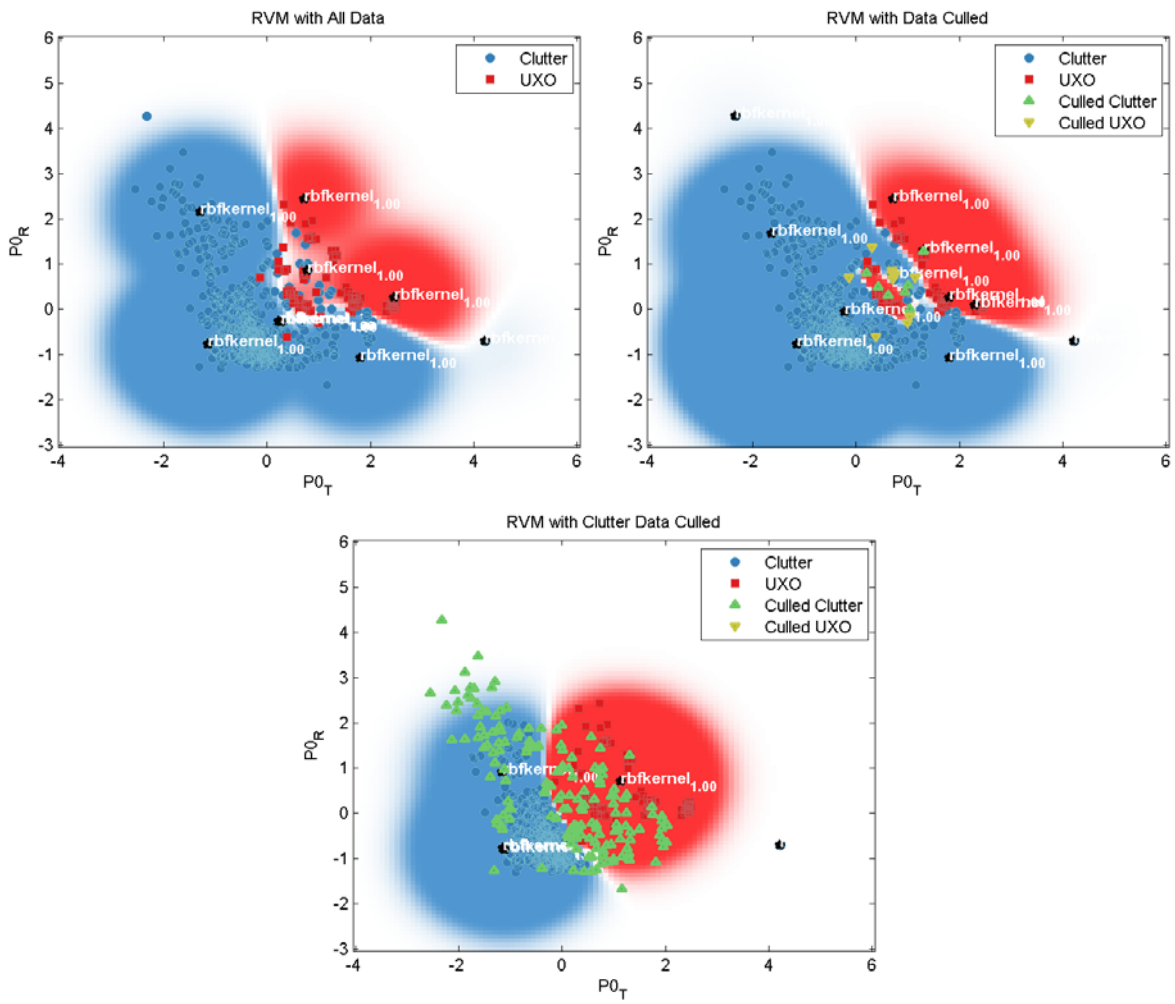

Figure 21: Example RVM decision surfaces with overly influential training data points removed.

ROCs corresponding to the decision surfaces shown in Figure 21 are presented in Figure 22, along with asterisks at the P_{FA} where P_D reaches 1. The ROC when all data is utilized is shown in blue; the ROC when highly influential clutter and UXO are removed is shown in green; and the ROC when only highly influential clutter is removed in shown in red. Removing both influential clutter and UXO results in $P_{FA} = 1$ at $P_D = 1$, which is much higher than the P_{FA} at $P_D = 1$ when all the data is retained (about 0.32). Removing only influential clutter data also increases P_{FA} at $P_D = 1$, to about 0.41, which is worse than retaining all the data, but much better than removing both influential clutter and influential UXO data points. The ROC curve, however, has generally lower performance than either retaining all the data or removing both influential clutter and influential UXO. This occurs because removing the data points that are not near the majority of the data points in the clusters has the effect of sharpening the decision boundary between the clutter and UXO classes. This effect is contrary to what has been observed to improve

performance – softening the transition in the decision surface between the two decision regions so that targets that are within the regions where the classes overlap are not assigned decision statistics that are strongly indicative of either class.

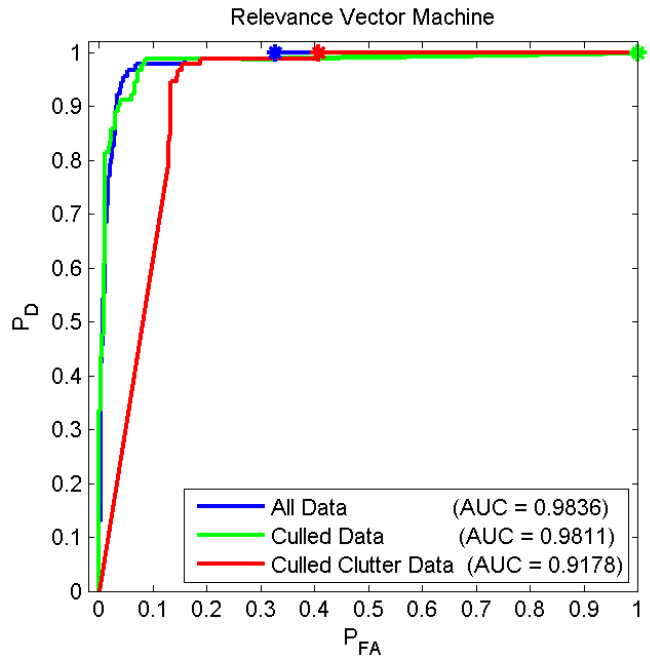

Figure 22: ROCs corresponding to the decision surfaces presented in Figure 21.

Modify Prior Assumptions Regarding Target Class Distributions

The DLRT and GMM-GLRT classifiers both attempt to characterize the distributions of the data for both the UXO and clutter classes, and the quality of the characterizations impacts the classifier performance. If either of those distributions is difficult to characterize, classifier performance may be adversely impacted. An alternate approach is to develop a classifier based only on the UXO data, in which the decision statistic is related to proximity to the UXO training data, or a classifier based only on the clutter data, in which the decision statistic is related to the distance from the clutter training data. Both of these approaches are considered for the DLRT and GMM-GLRT classifiers.

Example decision surfaces for the GMM-GLRT are shown in Figure 23 when both clutter and UXO training data is used (top-left), when only UXO training data is used (top-right), and when only clutter training data is used (bottom-center). These surfaces illustrate that when all the data is used, both proximity to UXO and distance from clutter are utilized to form the decision surface. However, when only UXO data is utilized, the decision surface demonstrates the decision statistics are related to the proximity to the UXO data. In contrast, when only clutter data is utilized, it is distance from the clutter data that is related to the decision statistics.

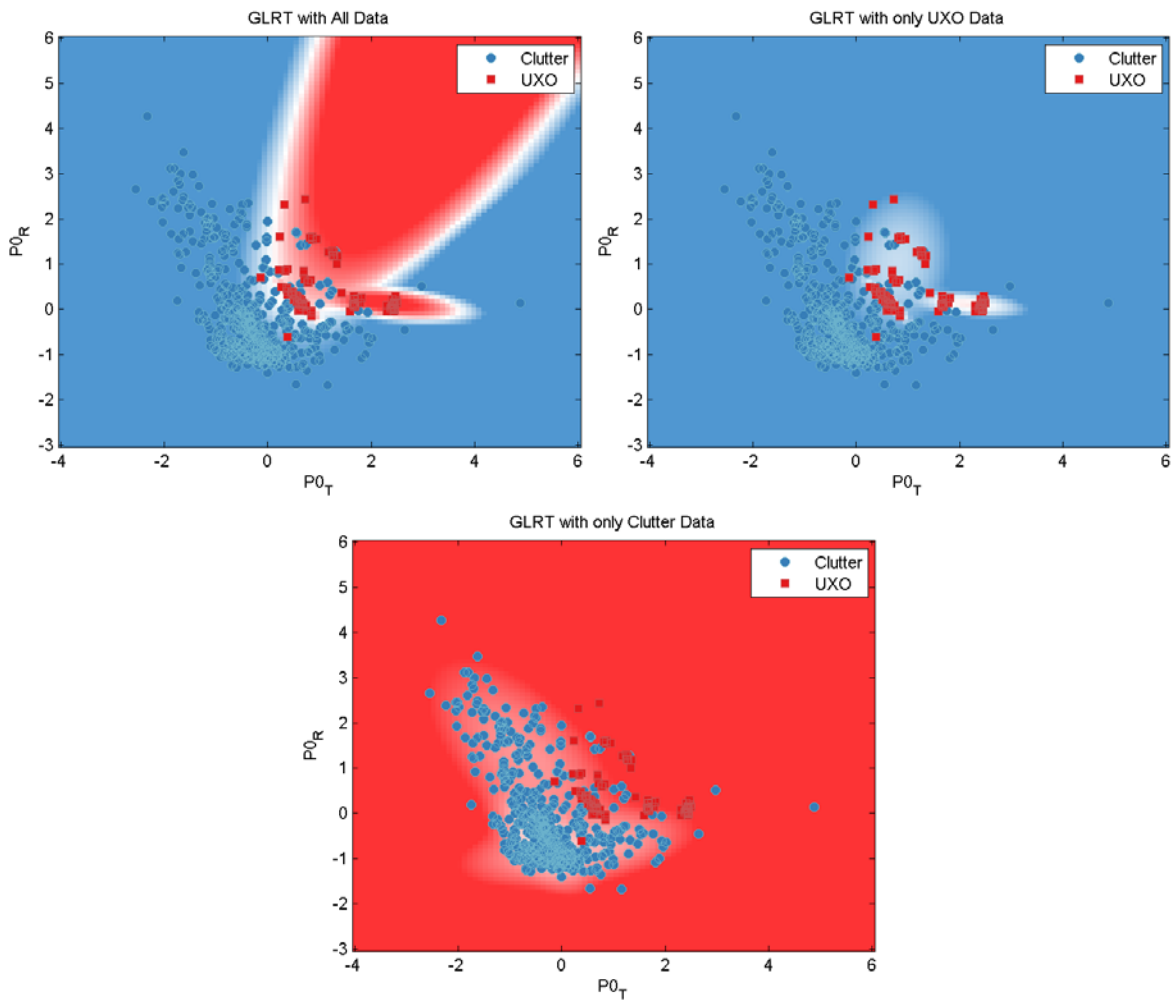

Figure 23: Example GMM-GLRT decision surfaces using both clutter and UXO training data (top-left), using only UXO training data (top-right), and using only clutter training data (bottom-center).

Performance for the P_{FA} at $P_D = 1$ performance measure is shown in Figure 24 for the $DLRT_3$ (left) and the GMM-GLRT (right). The $DLRT_3$ shows a dramatic decrease in the maximum P_{FA} at $P_D = 1$ when only UXO training data are utilized. In addition, the KSD pdf estimate of P_{FA} at $P_D = 1$ shows variation with the data that is utilized for training. The GMM-GLRT, however, has a fairly consistent maximum P_{FA} at $P_D = 1$ despite the data that is utilized for training, though the KSD pdf estimates vary significantly with the choice of training data. These results suggest that the choice of classifier may influence the choice of training data. Conversely, the availability of reliable training data may influence the choice of classifier.

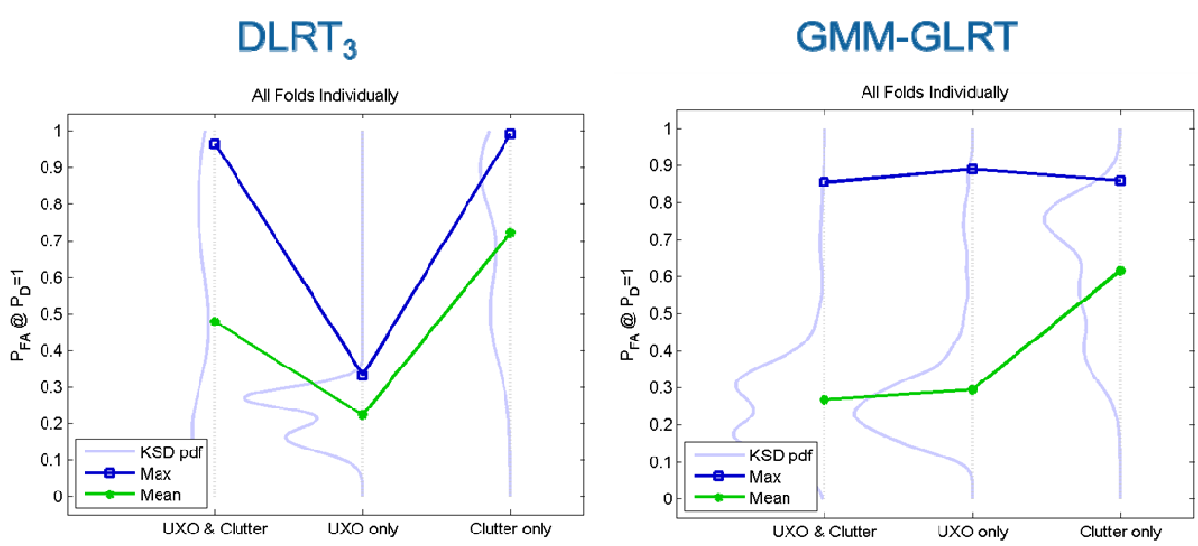

Figure 24: Classifier performance as a function of the training data utilized (UXO and clutter, UXO only, or clutter only) for the $DLRT_3$ classifier (left) and the GMM-GLRT classifier (right).

Performance results for the corresponding bagged classifiers are shown in Figure 25. While performance for the bagged $DLRT_3$ follows similar trends as for the $DLRT_3$ classifier, performance for the bagged GMM-GLRT classifier is quite different from performance for the GMM-GLRT classifier. The bagged GMM-GLRT classifier utilizing both UXO and clutter data has significantly lower P_{FA} at $P_D = 1$ than the classifier that utilizes only UXO or only clutter data for training.

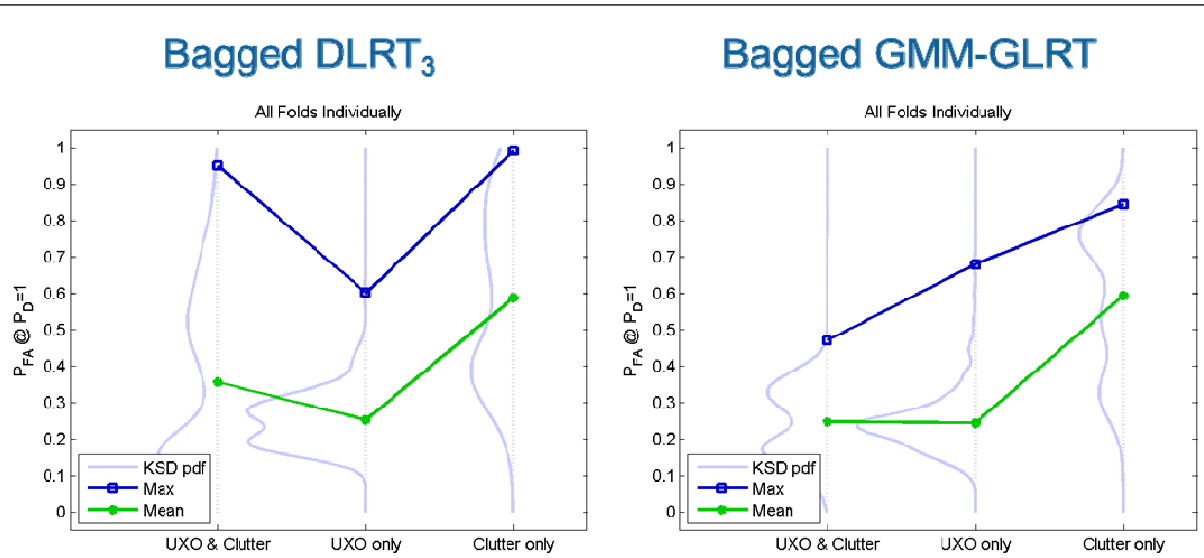

Figure 25: Classifier performance as a function of the training data utilized (UXO and clutter, UXO only, or clutter only) for the bagged $DLRT_3$ classifier (left) and the bagged GMM-GLRT classifier (right).

Improving Performance Prediction at P_D=1

Cross-validation is a technique commonly employed to obtain non-incestuous performance estimates from training data. The performance estimates may be affected by the specific cross-validation approach that is employed. Once again, an "extreme value" performance measure such as P_{FA} at $P_D = 1$ may exhibit more sensitivity to the cross-validation method than an "average value" performance measure such as AUC. The difference between considering the folds individually and in aggregate is depicted in Figure 26. The K folds (K=2 in this illustration) may be considered individually, in which case K-folds cross-validation yields K performance estimates, or all the folds may be considered in aggregate which yields a single performance estimate. The example ROCs shown in Figure 26 illustrate that the ROC is generally fairly consistent across individual folds as well as when the folds are considered in aggregate. The P_{FA} at $P_D = 1$ performance measure, denoted by the red asterisk on each of the ROC curves, can vary significantly across folds.

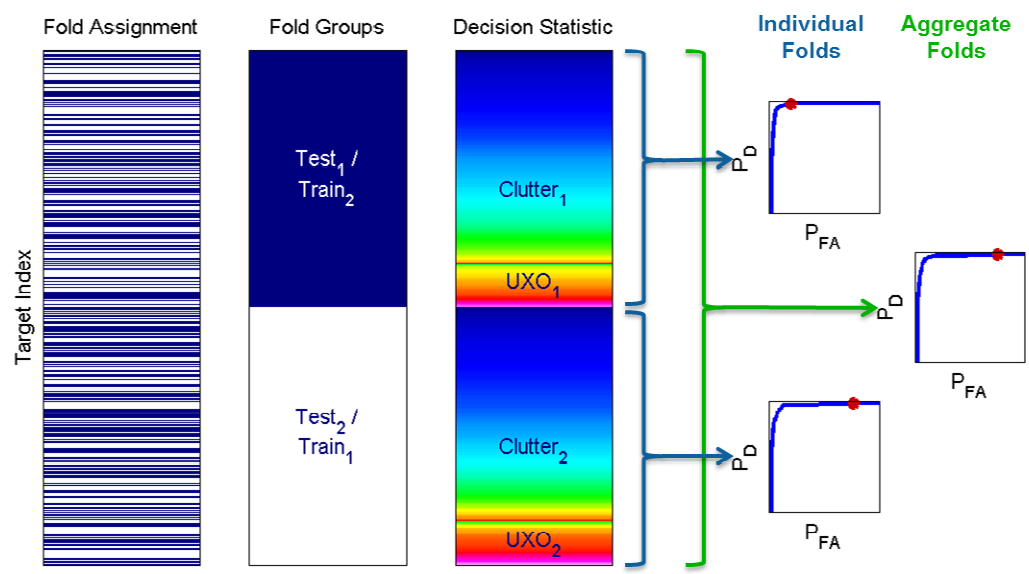

Figure 26: Pictorial representation of 2-folds cross-validation. The data is randomly divided into 2 groups (or folds). One fold is used for training, while the other is used for testing, and vise versa. This process results in two estimates of performance. The decisions statistics may also be aggregated across folds to generate a single performance estimate.

Cross-validated performance for the P_{FA} at $P_D = 1$ performance metric are shown in Figure 27 for the bagged $DLRT_3$ (top-left), bagged GMM-GLRT (top-right), and the bagged RVM (bottom-center). Across all three classifiers, the aggregated folds result in a higher estimate of maximum P_{FA} at $P_D = 1$. This suggests that considering the folds individually may result in a somewhat optimistic performance estimate of P_{FA} at $P_D = 1$, and aggregating the folds may provide a more realistic estimate of P_{FA} at $P_D = 1$. This also highlights the potential challenges associated with predicting performance using the P_{FA} at $P_D = 1$ metric, because this

performance measure is highly influenced by a single data point. When the folds are considered individually, the most challenging UXO target will be present in only a single fold, and only that fold will expose the impact of that UXO target on P_{FA} at $P_D = 1$. When the folds are considered in aggregate, however, the most challenging UXO target will be present in that set of decision statistics from which P_{FA} at $P_D = 1$ is estimated, and its impact will be included in the performance estimate. It should be noted, however, that the maximum P_{FA} at $P_D = 1$ estimated across all individual folds is not equal to maximum P_{FA} at $P_D = 1$ estimated in the aggregate folds.

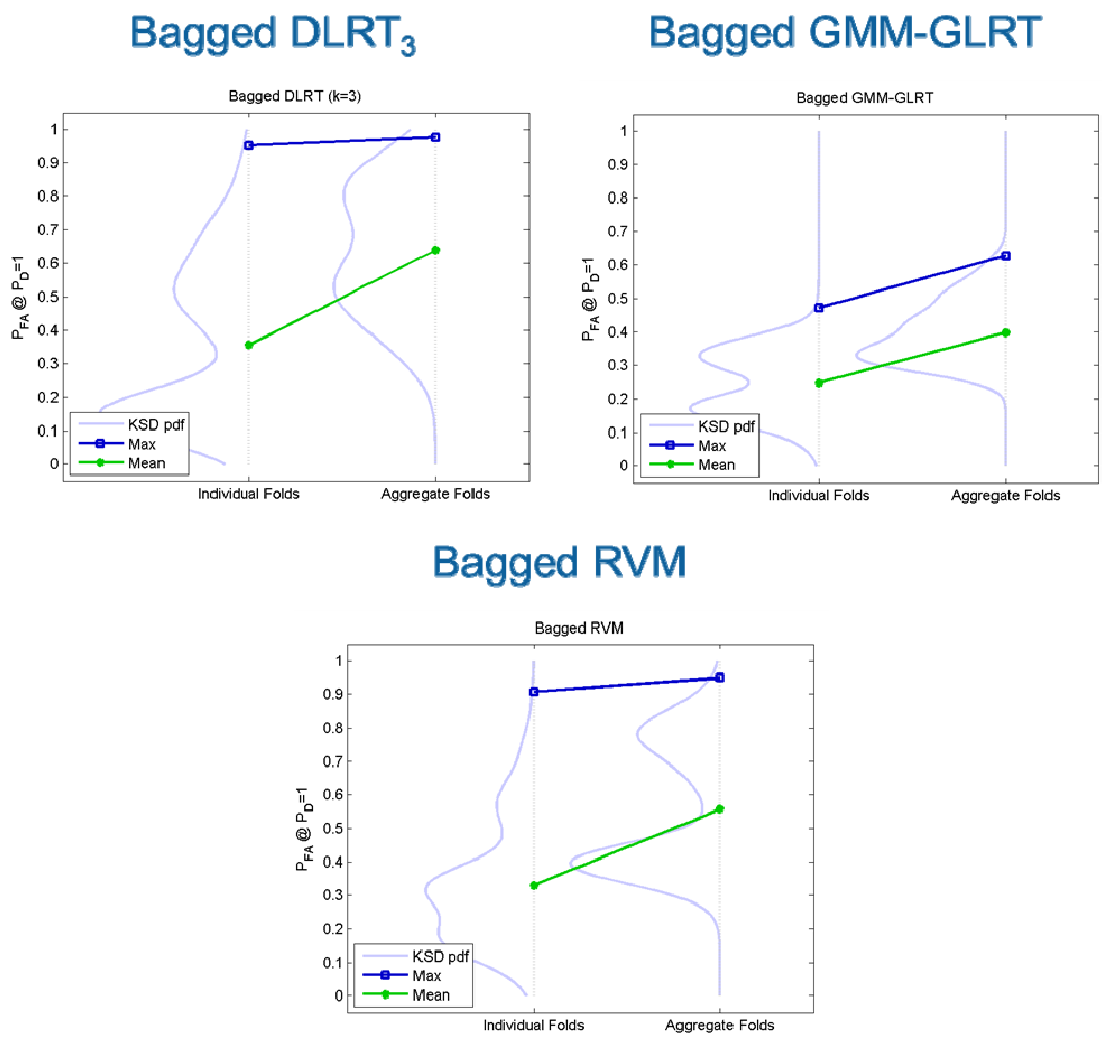

Figure 27: Classifier performance comparing individual cross-validation folds and aggregate cross-validation folds for the bagged $DLRT_3$ (top-left), bagged GMM-GLRT (top-right), and bagged RVM (bottom-center).

Efficient and Robust Model Inversion

Sensor Response Model

A generalized model for the EMI sensor response was developed to support the model inversion studies of the MetalMapper data collected as part of the SLO demonstration. The model is modular, in that the sensor's geometry (sensor coil locations and shapes) and the target model (i.e., dipole, dipole with BoR assumption) are individual components that are defined independently. A block diagram of the sensor response model is shown in Figure 28. First, the primary field at the transmitter is calculated. The primary field is then propagated to the target. The interaction of the primary field with the target is calculated to determine the secondary field due to the target. The secondary field is then propagated back to the receiver, and the receiver response to the secondary field is then calculated.

The transmitter and receiver coils may be modeled using either the Biot-Savart Law or Huygens' Principle. The Biot-Savart Law represents each coil using piecewise linear segments, and models the current in the coil as the line integral over currents in the individual segments with which it is modeled. Huygens' Principle models the coil as a grid of very small dipoles. When all the dipoles are considered in aggregate, the currents in the interior cancel, leaving only the currents around the exterior of the grid to model the current in the coil. The target may be modeled either as a sum of three magnetic dipoles, or as a sum of two magnetic dipoles (the BoR assumption). The models for the transmit coil, target, and receive coil are defined independently, thus facilitating the implementation of models for new sensors. In all cases, the propagation between the transmitter and target, and target and receiver, is modeled using a far-field assumption.

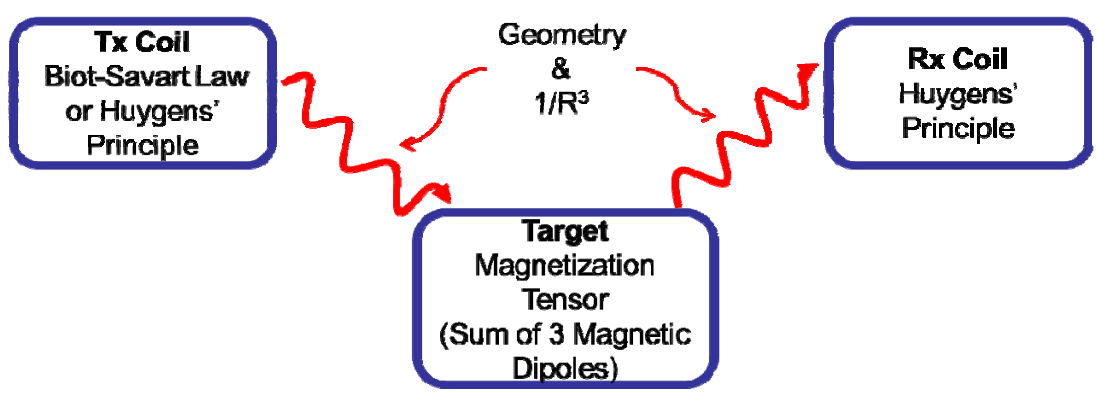

Figure 28: Sensor response model block diagram. The sensor's transmitter and receiver geometry are defined independently of the target response model. 1) The primary field at the transmitter is calculated. 2) The primary field is propagated from the transmitter to the target. 3) The interaction of the incident primary field on the target with the target is calculated. 4) The resultant secondary field after interaction with the target is propagated back to the receiver. 5) The receiver response to the secondary field is calculated.

Model Inversion

Target parameters are estimated from the measured data via two approaches: 1) a simple model in which the sensor geometry is not taken into consideration, and 2) a rigorous model in which the sensor geometry is taken into consideration.

In the first approach, is it assumed that the response for each channel (Tx-Rx pair) may be modeled as a weighted sum of decaying exponentials. This model is based on the physics, as the kernel of the full physics-based dipole model is also a weighted sum of decaying exponentials, but is less constrained than the full physics-based model as the weights on each of the decaying exponentials are not explicitly modeled by the target-sensor geometry. The full physical model, in contrast, also models the weights associated with each of the decaying exponential terms, with each of these weights being a function of the target-sensor geometry. These models trade-off greater computational complexity for the potential for greater fidelity in the target parameters; the simple model may not offer as high fidelity on the target parameters, but the computational complexity may be several orders of magnitude lower.

In both cases, a numerical optimization (Levenburg-Marquardt) procedure is employed to find the model parameters that minimize the residual (sum of squared errors) between the modeled data and the measured data. Many times, the parameter estimation process is sensitive to the initial conditions selected for the numerical optimization. That is, different initial conditions often lead to different solutions. The residual at these solutions can be used to select a single, global, solution. It is not uncommon, however, for multiple solutions to have similarly low residuals.

To mitigate some of the sensitivity to the initial conditions, the parameters for the simple model are estimated via a sequential process, in which the estimates found for the model of order p are used to guide the initial conditions when estimating the parameters for the model of order $p + 1$. The parameters found for the simple model are then used to guide the initial conditions for inverting the full physical model. The full physical model is inverted under the assumption that the target may be modeled as a BoR, as well as with that assumption removed.

Classification results after inverting the simple model are shown in Figure 29 and results after inverting the full model follow in Figure 30. For both models, classification performance is found with an RVM (blue line) and a bagged GMM-GLRT (green line) classifier using 10-folds cross-validation, and there are four cases considered. In the top-left, the result of model inversion with the BoR assumption is shown. The result of model inversion with the BoR assumption and using goodness-of-fit (GoF) as a feature in addition to the model parameters is shown in the top-right. The bottom-left and bottom-right show performance when the BoR assumption is dropped, without (left) and with (right) the GoF as an additional feature.

Performance with the simple model is comparable to performance with the full model, with a fraction of the computation complexity (approximately 1/30th the time to invert the simple modele compared to the full model). Performance with the simple model tends to be slightly stronger when the BoR assumption is removed. However, including the GoF as an additional feature does not offer much benefit, either with or without the BoR assumption in the inversion.

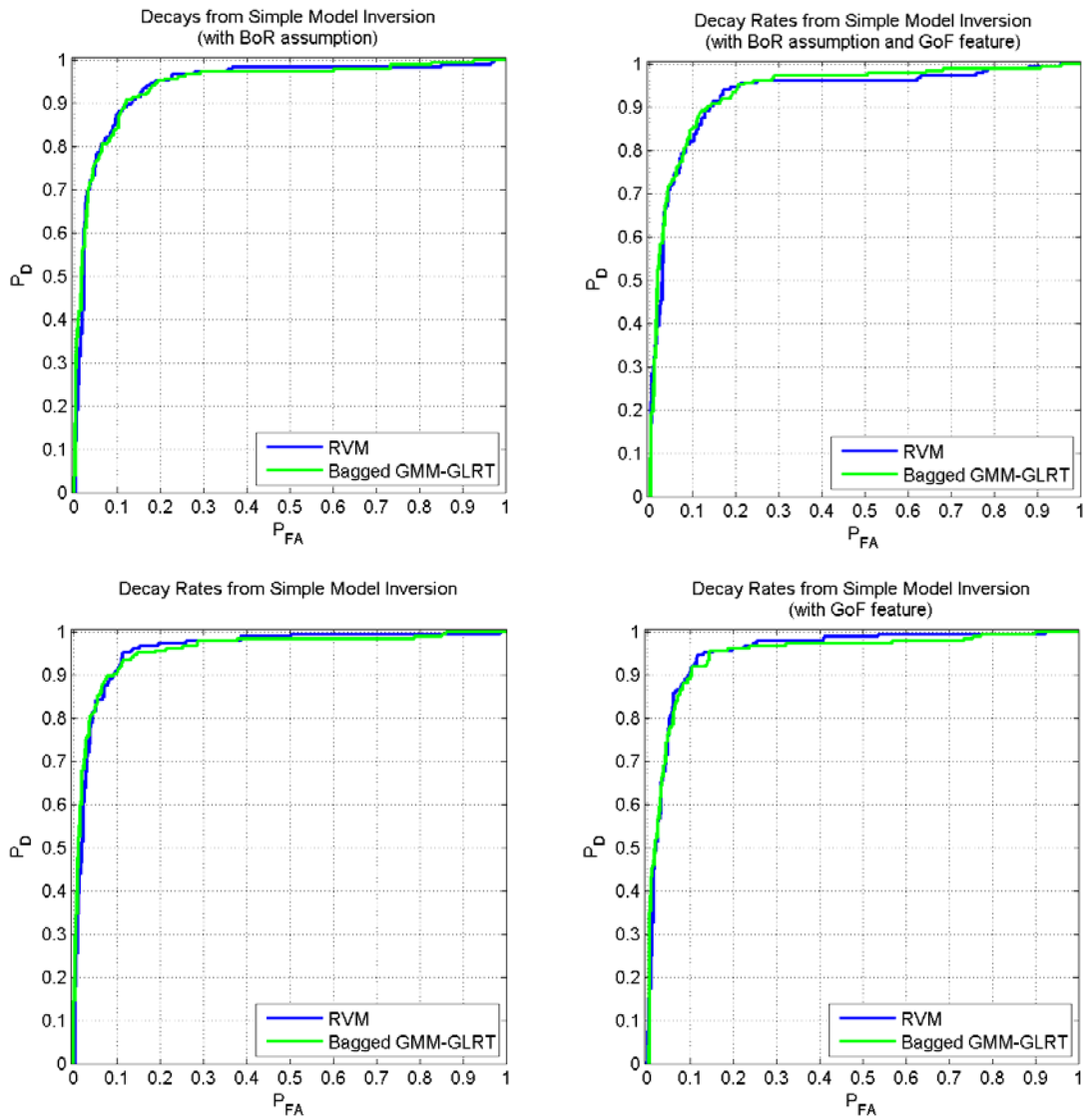

Figure 29: Classification performance for RVM (blue line) and bagged GMM-GLRT (green line) classifiers with features (decay rates) derived from inverting the simple model. Top-left: Model inversion assuming the target is a BoR. Top-right: Model inversion assuming the target is a BoR, and including the GoF parameter as a feature. Bottom-left: Model inversion without assuming the target is a BoR. Bottom-right: Model inversion without assuming the target is a BoR and including the GoF parameter as a feature.

Performance with the full model shows some trends similar to performance with the simple model. Specifically, the RVM and bagged GMM-GLRT classifiers perform similarly, and including GoF as an additional feature does not offer much benefit whether the BoR assumption is employed, or not. In contrast to the simple model, increasing the model order by removing the BoR assumption does not appear improve performance, and in some cases may degrade it slightly. Thus, the additional model complexity, and associated increase in computational complexity, is not providing a commensurate improvement in classification performance.

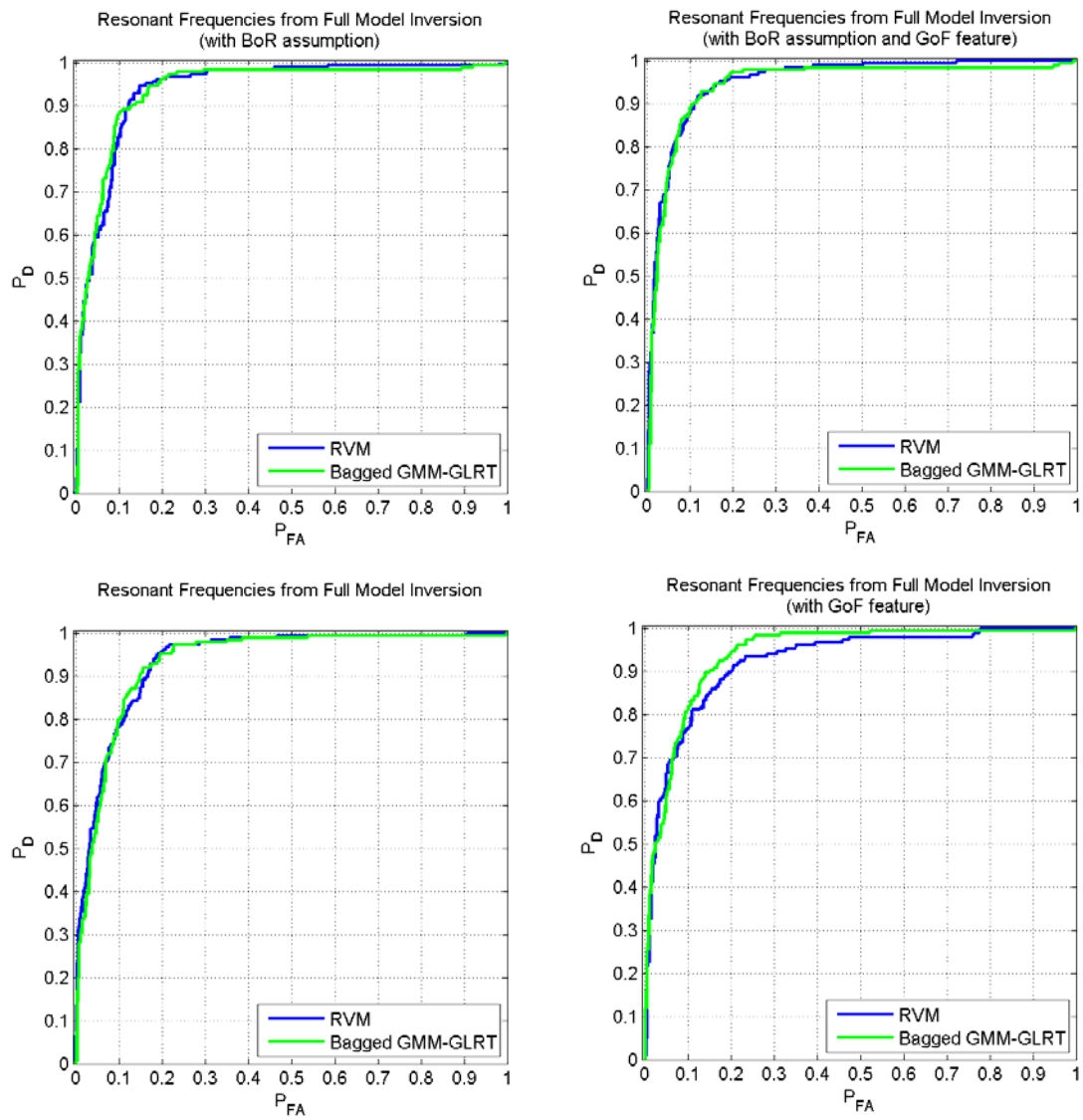

Figure 30: Classification performance for RVM (blue line) and bagged GMM-GLRT (green line) classifiers with features (resonant frequencies) derived from inverting the full model. Top-left: Model inversion assuming the target is a BoR. Top-right: Model inversion assuming the target is a BoR, and including the GoF parameter as a feature. Bottom-left: Model inversion without assuming the target is a BoR. Bottom-right: Model inversion without assuming the target is a BoR and including the GoF parameter as a feature.

Information-Theoretic Inversion Augmentation

Fisher Information (FI) is a measure of the information conveyed about the model by the model parameters; high FI indicates the model parameters are highly informative with respect to the model. In this application, FI is used to augment the model inversions by guiding the selection of model parameters from among multiple candidate sets of model parameters with similarly small residuals.

Fisher Information is a function of the model parameters (including the spatial and temporal sampling parameters); it does not depend on the measured data. Thus, FI offers an assessment of the model parameters that is independent of the measured data. Effectively, FI acts as a regularizer that provides a mechanism for consistent selection of model parameter estimates from among multiple candidates that have similarly low residuals.

For a given set of K independent and identically distributed measurements from an assumed model $m(\boldsymbol{\theta})$ corrupted by Gaussian noise with variance σ_n^2, FI is given by

$$\mathrm{FIM}(\boldsymbol{\theta}) = \frac{1}{\sigma_n^2} \sum_{k=1}^{K} \mathrm{FIM}_k(\boldsymbol{\theta}), \tag{5}$$

where

$$\mathrm{FIM}_k(\boldsymbol{\theta}) = \left[\frac{\partial m(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]\left[\frac{\partial m(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]^T.$$

To describe Fisher Information, consider the problem of estimating the parameters of a Gaussian pulse, $GP(\mathbf{x};b,c) = \exp\left[\frac{-(\mathbf{x}-b)^2}{2c}\right]$, when only two measured data points are available, illustrated in Figure 31. The least squares objective function surface is shown in the top-left, and demonstrates there are two candidate solutions with similarly low error. The Gaussian pulses corresponding to those two parameter sets are shown in the bottom-center in the green and orange curves. (The green curve more closely approximates the true curve from which the two measured data points were generated.) Clearly, both of these curves fit the measured data very well. Fisher Information as a function of the mean (b) and variance (c) parameters is shown in the top-right. (The speckle in the image is due to the numerical artifacts of calculating the determinant of the matrix.) Here, it can be seen that the parameters for the first candidate solution (the green curve) have higher FI, and so those parameters are more informative. If only the least square error were considered, the second (incorrect) candidate solution (orange curve) would be selected. When Fisher Information is used to augment the least squares solution, the first (correct) candidate solution (green curve) is

selected. This example illustrates that when least squares alone results in multiple candidate solutions, FI has the potential to help guide selecting a single solution.

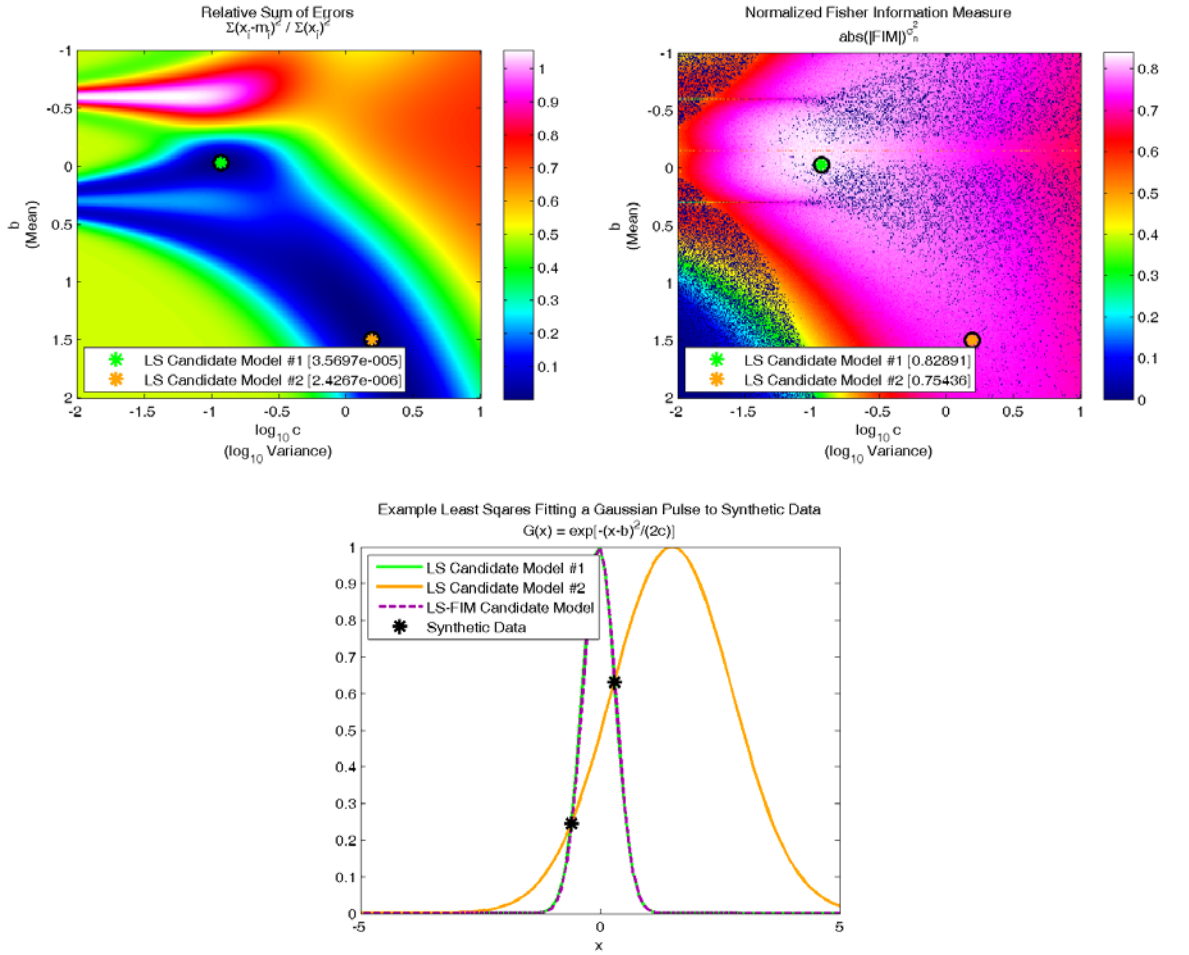

Figure 31: Example application of Fisher Information to augment least squares model inversion to estimate the parameters of a Gaussian pulse. Example least square objective function surface (top-left), in which preferred solutions have smaller sum of errors. Example Fisher Information surface (top-right), in which preferred solutions have higher FI. Model fits (bottom-center). The green curve more closely approximates the true parameters used to generate the measured data. Least squares alone selected the orange curve. Least squares augmented with FI selects the purple curve.

The full physical model was inverted from multiple initial conditions, and FI was calculated for the solution returned for each initial condition. The solutions were ranked with respect to their residual (low to high residual) and their associated FI (high to low FI). The solution with the lowest sum of ranks was then selected as the single solution. If multiple solutions had identical sum of ranks, the tie was broken using the lowest residual.

Although Fisher Information has been shown in some instances to have the potential to guide selection of stable model parameters, in this case it does not consistently show performance benefits. Interestingly, including GoF as an additional feature noticeably degraded

performance. This can be attributed to FI guiding the selection of solutions that did not produce the lowest residual. Thus, the residual loses its significance as a measure of how well the target response resembles UXO. When the residual is the sole determinant of model parameters, the residual tends to be smaller for UXO than for clutter because UXO are often generally ellipsoidally shaped, while clutter are not, so the dipole model with three terms tends to be a better approximation to a UXO response than a clutter response.

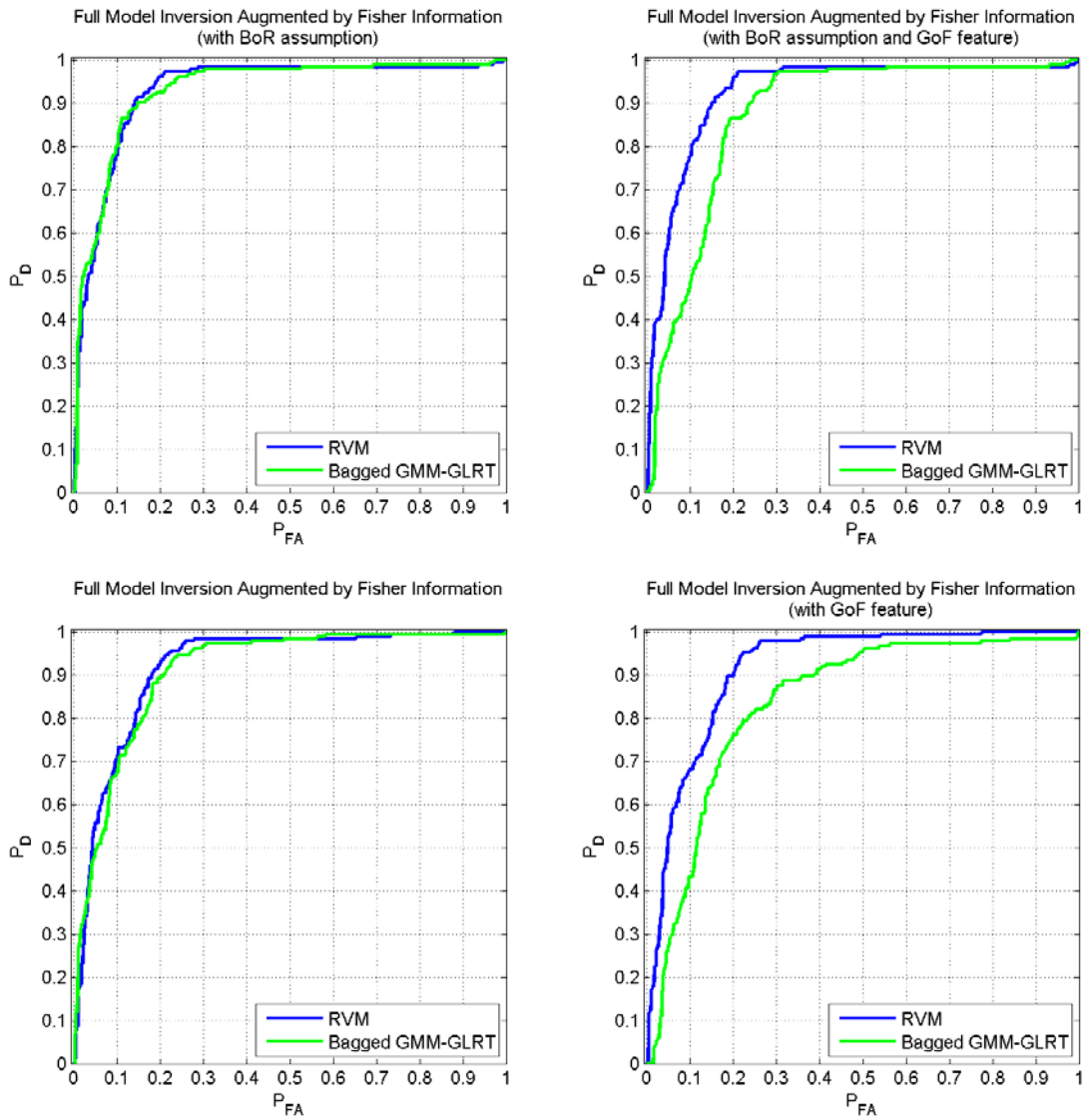

Figure 32: Classification performance for RVM (blue line) and bagged GMM-GLRT (green line) classifiers with features (resonant frequencies) derived from inverting the full model and applying Fisher Information to guide model parameter selection. Top-left: Model inversion assuming the target is a BoR. Top-right: Model inversion assuming the target is a BoR, and including the GoF parameter as a feature. Bottom-left: Model inversion without assuming the target is a BoR. Bottom-right: Model inversion without assuming the target is a BoR and including the GoF parameter as a feature.

Multiple Instance Learning

Multiple Instance Learning (MIL) is a machine learning technique that enables automated learning of the features that are indicative of the H_1 class from multiple measurements for a given target, even if the informative features are not present for *every* measurement for a target from the H_1 class. The data organization for MIL is depicted in Figure 33. All the measurements for a given target, termed "instances" are placed in a "bag." Each instance consists of a N-dimensional feature vector. A bag is deemed to belong to the H_1 class if at least one instance has features that are indicative of the H_1 class. The MIL algorithm utilized here is the adaptive p-posterior mixture model (ppmm) kernel MIL [15].

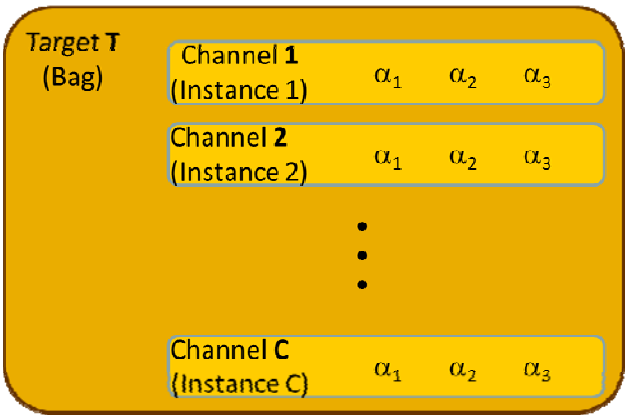

Figure 33: Graphical depiction of multiple instance learning. All measurements, termed "instances" for a given target are placed in a "bag." Each instance consists of an N-dimensional feature vector.

For the simple model in which parameter estimates are derived from each channel individually, it is not certain which channels produce parameters that reliably distinguish between UXO and clutter. Multiple instance learning provides a framework in which data of this type can be analyzed. As long as at least one channel (instance) produces features that are indicative of UXO, the data fits within the MIL paradigm. Prior to applying ppmm MIL, the intances were culled so that only channels that produced valid decay rates (between the lower and upper bounds of 50 and 40000, respectively) were retained. This process resulted in a varying number of instances across targets, but at least several instances remained for each target.

Classification performance results for the ppmm MIL algorithm applied to decay rates that are estimated from each channel individually are shown in Figure 34. Performance with the decay rates alone is shown on the left, and performance with GoF as an additional features is shown on the right. For comparison, a baseline energy detector is also shown (black line). The number of clusters is a parameter that must be chosen. Performance was evaluated for a wide range of values for the number of clusters (2 to 30), and the one or two values that simultaneously provided the lowest P_{FA} at $P_D = 1$ and high AUC are presented. The performance

of these classifiers is on par with performance shown for the simple model and full model inversions. The benefit of this approach is the large reduction in computational complexity. The total computation time required to obtain these results is about 2% of the time required for the full model inversion, with the overwhelming majority of the difference in the time required for model parameter estimation.

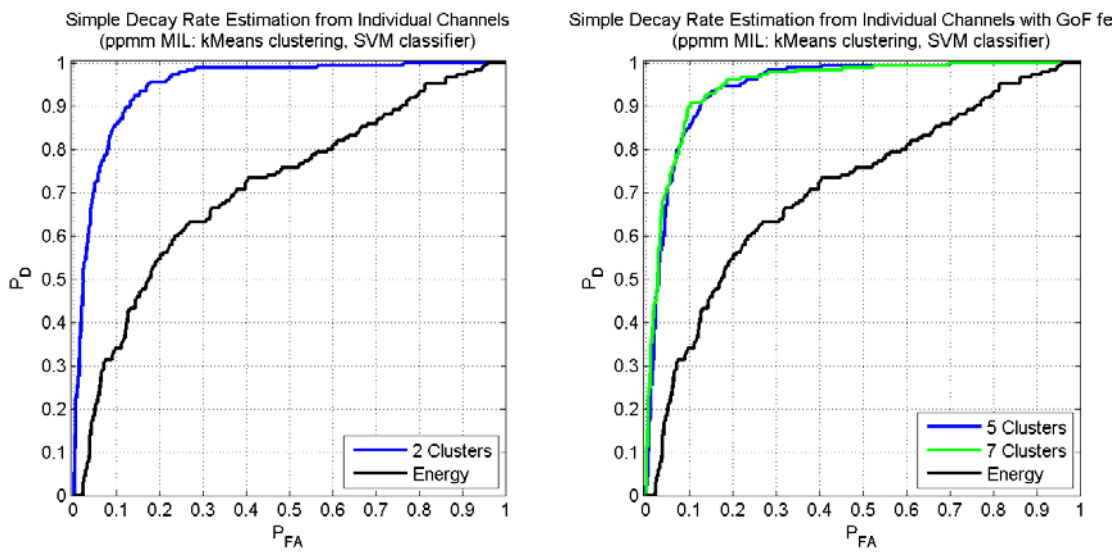

Figure 34: Classification performance for ppmm MIL with kMeans clustering and an SVM classifier using features (decay rates) derived from inverting the simple model. Left: Model inversion without assuming the target is a BoR. Right: Model inversion without assuming the target is a BoR, and including the GoF parameter as a feature.

Results and Discussion

The two major thrusts of this work, investigating classifier sensitivity with respect to the minimum P_{FA} at $P_D = 1$ performance goal and investigating efficient and robust model inversion have yielded interesting results. The classifier sensitivity studies revealed that some classifiers are more sensitive at the P_{FA} at $P_D = 1$ operating point than others. Those classifiers that exhibited the most robust performance tended to be less decisive at the boundary; the decision surface more gradually transitioned from a high decision statistic strongly indicative of UXO to a low decision statistic strongly indicative of clutter. The classifiers that showed the most sensitivity tended to be very decisive at the boundary, with a very sharp transition from a high (UXO) decision statistic to a low (clutter) decision statistic. The robust classifiers, with the more gradual boundary, allowed for a greater margin of error in the UXO features for the targets under test. This manifested itself in lower P_{FA} at $P_D = 1$ across a large number of simulated non-incestuous test data sets. The classifier sensitivity studies showed that the choice of classifier can have a significant impact on P_{FA} at $P_D = 1$ performance, and considering this sensitivity to select a classifier that minimizes the maximum P_{FA} at $P_D = 1$ (i.e., the best, worst-case scenario) can improve classification performance at P_{FA} at $P_D = 1$.

The model inversion investigations suggested that it may be possible to achieve efficient and robust model inversion. With a machine learning algorithm that is well-suited for the problem at hand, such as ppmm MIL, it may be possible to overcome lower fidelity in the model parameter estimates from a simpler model inversion process to achieve strong performance, and to do so in a much more computationally efficient manner. Improved computational efficiency may allow, in the future, real-time decisions in the field as cued target interrogations take place. This immediate feedback could improve the efficiency of site clearance operations by enabling real-time evaluation of the data quality, thereby reducing the number of targets denoted as “Can’t Analyze.”

Conclusions and Implications for Future Research

The research has demonstrated the need for and benefits of classifiers which are robust at the P_{FA} at $P_D = 1$ performance goal, as well as the potential for efficient and robust model inversion methods. More importantly, the interplay between the two (classifiers and inversion) has been demonstrated by showing that a sophisticated machine learning algorithm coupled with a simpler inversion procedure can produce strong classification results, with a small fraction of the computation time.

Follow-on research to improve model inversion would be to consider all the objects at a site in aggregate when inverting the parameters for each anomaly, termed full-site model inversion. This approach would enable learning from the objects at the site, without necessarily knowing

the labels (UXO or clutter class) for those objects, via joint inversion and clustering in the feature space directly from the data, and guided by *a priori* knowledge gained from any available training data at this site or inferred from previous sites. We will be requesting a meeting with Dr. Herbert Nelson to present and discuss preliminary results related to full-site model inversion. Machine learning techniques, such as multiple instance learning, in which a more sophisticated learning algorithm and classifier are coupled with a simpler inversion process are also an interesting avenue for further inquiry. Results obtained at SLO show promise for this type of approach, and evaluating it in other scenarios could be valuable.

Literature Cited

[1] L. M. Collins, Y. Zhang, J. Li, L. Carin, S. Hart, S. Rose-Pehrsson, H. Nelson, and J. McDonald. A comparison of the performance of statistical and fuzzy algorithms for unexploded ordnance detection. *IEEE Transactions on Fuzzy Systems*, 9(1):17–30, 2001.

[2] B. Barrow and H. H. Nelson. Model-based characterization of electromagnetic induction signatures obtained with the MTADS electromagnetic array. *IEEE Transactions on Geoscience and Remote Sensing*, 39(6):1279–1285, June 2001.

[3] T. H. Bell, B. J. Barrow, and J. T. Miller. Subsurface discrimination using electromagnetic induction sensors. *IEEE Transactions on Geoscience and Remote Sensing*, 39(6):1286–1293, June 2001.

[4] E. R. Cespedes. Advanced UXO detection/discrimination technology demonstration – U.S. Army Jefferson Proving Ground, Madison, Indiana. Technical Report ERDC/EL TR-01-20, U.S. Army Engineer Research and Development Center, Vicksburg, MS, 2001.

[5] Unexploded ordnance response: Technology and cost, Report to Congress. Technical report, Department of Defense, 2001.

[6] S. L. Tantum and L. M. Collins. A comparison of algorithms for subsurface target detection and identification using time domain electromagnetic data. *IEEE Transactions on Geoscience and Remote Sensing*, 39(6):1299–1306, 2001.

[7] P. Gao, L. M. Collins, P. Garber, N. Geng, and L. Carin. Classification of landmine-like metal targets using wideband electromagnetic induction. *IEEE Transactions on Geoscience and Remote Sensing*, 38(3):1352–1361, May 2000.

[8] P. Runkle, P. Bharadwaj, L. Couchman, and L. Carin. Hidden Markov models for multi-aspect target identification. *IEEE Transactions on Signal Processing*, Under Review, Submitted April 1998.

[9] Y. Dong, P. R. Runkle, L. Carin, R. Damarla, A. Sullivan, M. A. Ressler, and J. Sichina. Multi-aspect detection of surface and shallow-buried unexploded ordnance via ultra-wideband synthetic aperture radar. *IEEE Transactions on Geoscience and Remote Sensing*, 47(6):1259–1270, June 2001.

[10] L. Carin, N. Geng, M. McClure, Y. Dong, Z. Liu, J. He, J. Sichina, M. Ressler, L. Nguyen, and A. Sullivan. Wide-area detection of land mines and unexploded ordnance. *Inverse Problems*, 18(3):575–609, June 2002.

[11] S. Hart, H. Nelson, R. Grimm, S. Rose-Pehrsson, and J. McDonald. Probabilistic neural networks for unexploded ordnance (UXO) classification using data fusion of magnetometry and EM physics-derived parameters. In *UXO/Countermine Forum*, Anaheim, CA, 2000.

[12] J. J. Remus, K. D. Morton, P. A. Torrione, S. L. Tantum, and L. M. Collins. Comparison of a distance-based likelihood ratio test and k-nearest neighbor classification methods. In *Machine Learning for Signal Processing*, pages 362–367, Cancun, Mexico, 2008.

[13] Michael E. Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244, June 2001.

[14] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science, 2006.

[15] H-.Y. Wang, Q. Yang, and H. Zha. Adaptive p-posterior mixture-model kernels for multiple instance learning. In *Proceedings ot the* 25th *International Conference on Machine Learning*, 2008.