FINAL REPORT

Detection of Buried Targets via Active Selection of Labeled Data: Application to Sensing Subsurface UXO

SERDP Project MM-1283

JUNE 2007

Lawrence Carin
Duke University

This document has been approved for public release.

DoD EPA DOE SERDP
Strategic Environmental Research and Development Program

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.

Project Summary

The principal focus of this project was on developing new statistical algorithms for analysis of electromagnetic induction (EMI) and magnetometer data measured at actual former bombing ranges. To address this challenge, four key technologies have been developed:

- **Active learning**: Active learning is a framework whereby the labeled (training) data used to design a classifier are defined *in situ* directly on the site under test. This is performed using information-theoretic measures, in which one asks which signatures from the site under test would be most informative for classifier design if the associated labels could be acquired. These labels are then acquired via excavation. In this framework one assumes no *a priori* labeled (training) data, and the initial phase of excavation is performed for the purpose of learning an algorithm. Once the information-theoretic measure indicates that no further information is available from further excavations, the excavations for the purpose of learning are terminated. At this point, using the acquired labeled data, a dig list is provided, defining an ordered list of which items are most likely to be UXO. Since the output of this process is defined in terms of the *probability* of being UXO, a risk-based analysis may also be employed, to define when to stop excavating (when the risk analysis indicates that the expected cost is minimized by leaving the remaining items unexcavated).

- **Concept drift**: In the above discussion it was assumed that no labeled (training) data were available. However, in practice one may have labeled data, for example from previous excavations at different former ranges. The challenge is that the data from a previous site may not be entirely relevant for the new site under test. Speaking in a statistical sense, the site under test may be characterized as one "concept", and there may be a "concept drift" from this new site relative to data from previous sites analyzed previously. Therefore, the objective is to learn the "concept drift" between the site of interest and previous sites, and once this statistical relationship is understood/learned, one may *appropriately* utilize previous existing labeled data. In concept-drift-based learning, one again employs active learning, whereby an initial set of excavations are performed for the purpose of learning. However, in this context one is interested in learning the inter-relationship between the current site under test and previous labeled data sets that are available. The final dig list, after excavation for the purpose of learning, utilizes all of the excavated labeled data from the site of interest, as well as an appropriately weighted version of the labeled data from previous sites. In this sense the algorithm learns to appropriately utilize all available data.

- **Semi-supervised learning**: Most existing classification algorithms are supervised. This implies that a set of labeled data are provided to the classifier, from which learning is constituted. The classifier so learned is then applied *one by*

one to each of the unlabeled signatures, for which a classification (UXO/non-UXO) is desired. In UXO sensing one typically has access to all of the unlabeled data simultaneously (since all of the data are collected at once, typically). Therefore, there is an opportunity to place the classification of any given signature within the context of all other unlabeled signatures (since all unlabeled data are available simultaneously). When one takes account of the properties of the unlabeled data when learning a classifier, this is termed semi-supervised learning. Duke has developed novel semi-supervised algorithms, and applied them successfully to measured UXO data.

- **Sensor management**: The active-learning and concept-drift algorithms address the issue of what is termed, in the statistics community, "incomplete" or "missing" data. Specifically, the unlabeled data from a given site of interest is "missing" labels, and in that sense it is "incomplete". The active-learning framework seeks to fill in missing data in an optimal manner, with targeted (information-theory-based) excavations, for the purpose of learning the statistical characteristics of the new site under test. Another form in which missing data is manifested is if only a subset of sensors is deployed in a given region. For example, assume that EMI and magnetometer sensors are available. A given portion of a site may be interrogated by EMI, another region by magnetometer, and a third by both of these sensors. There is "missing data" in those regions for which only one of the two sensors is deployed. Similar issues may exist for the spatial sample rate of the data, even for a single sensor: If data are sampled in spatial increments Δx, there is missing data, for example, at finer sample rates. We may again employ information theory to optimally deploy additional sensors, with the purpose of optimally completing the data, and deployment of additional sensors is terminated when the expected information gain saturates. This framework has been developed for multi-sensor UXO sensing.

Publications

The research reported here has resulted in several publications, in leading journals. It has also been published at the foremost conferences in machine learning (*i.e.*, many of the ideas developed here are novel at the basic statistics/machine-learning level, beyond the specific applicability for UXO). As requested by SERDP, we have recently written papers that are targeted directly to the UXO community, to improve the accessibility of the ideas we have developed. Those papers are appended here as a part of this final report. The specific papers are:

- Y. Zhang, X. Liao and L. Carin, "Detection of buried targets via active selection of labeled data: application to sensing subsurface UXO," *IEEE Trans. Geosc. Remote Sensing,* vol. 42, pp. 2535-2543, Nov. 2004.

- D. Williams, C. Wang, X. Liao and L. Carin, "Classification of unexploded ordnance using incomplete multi-sensor multiresolution data," accepted for publication in *IEEE Trans. Geoscience & Remote Sensing*

- Q. Liu, X. Liao and L. Carin, "Detection of unexploded ordnance via efficient semi-supervised and active learning," submitted to *IEEE Trans. Geoscience & Remote Sensing*

- X. Liao and L. Carin, "Migratory logistic regression for learning concept drift between two data sets: Application to UXO sensing," submitted to *IEEE Trans. Geoscience & Remote Sensing*

Transitions

Many of the ideas developed under this SERDP project are being utilized within the context of an ESTCP project being executed by Signal Innovations Group, Inc., where they are being demonstrated on actual sites.

Detection of Buried Targets via Active Selection of Labeled Data: Application to Sensing Subsurface UXO

Yan Zhang, Xuejun Liao and Lawrence Carin
Department of Electrical and Computer Engineering
Duke University
Box 90291
Durham, NC 27708-0291

Abstract – When sensing subsurface targets, such as landmines and unexploded ordnance (UXO), the target signatures are typically a strong function of environmental and historical circumstances. Consequently, it is difficult to constitute a universal training set for design of detection or classification algorithms. In this paper we develop an efficient procedure by which information-theoretic concepts are used to design the basis functions and training set, directly from the site-specific measured data. Specifically, assume that measured data (*e.g.,* induction and/or magnetometer) are available from a given site, unlabeled in the sense that it is not known *a priori* whether a given signature is associated with a target or clutter. For N signatures the data may be expressed as $\{\mathbf{x}_i, y_i\}_{i=1,N}$, where $\mathbf{x}_i$ is the measured data for buried object i and y_i is the associated *unknown* binary label (target/non-target). Let the N $\mathbf{x}_i$ define the set $\mathbf{X}$. The algorithm works in four steps: (i) The Fisher information matrix is used to select a set of basis functions for the kernel-based algorithm, this step defining a set of n signatures $\mathbf{B}_n \subset \mathbf{X}$ that are most informative in characterizing the signature distribution of the site; (ii) the Fisher information matrix is used again, to define a small subset $\mathbf{X}_s \subset \mathbf{X}$, composed of those $\mathbf{x}_i$ for which knowledge of the associated labels y_i would be most informative in defining the weights for the basis functions in $\mathbf{B}_n$; (iii) the buried objects associated with the signatures in $\mathbf{X}_s$ are excavated, yielding the associated labels y_i, represented by the set $\mathbf{Y}_s$; and (iv) using $\mathbf{B}_n$, $\mathbf{X}_s$ and $\mathbf{Y}_s$ a kernel-based classifier is designed, for use in classifying all remaining buried objects. This framework is discussed in detail, with example results presented for an actual buried-UXO site.

I. INTRODUCTION

It is well known that sensor signatures of buried targets such as landmines and unexploded ordnance (UXO) are a strong function of their history and soil environment. For example, radar and seismic sensing of landmines is a strong function of the soil properties [1]. Electromagnetic induction (EMI) and magnetometer [2,3] sensors are typically less sensitive to soil properties when the target is of high metal content, such as UXO. However, the complexity of the UXO sensing problem is strongly influenced by which ordnance are present, on how the ordnance impacted the soil, and on the surrounding man-made conducting clutter and UXO fragments. All of these issues are dependent on the history of a given UXO site.

These characteristics of the subsurface-sensing problem significantly complicate design of detection and classification algorithms, since it is difficult to define a set of landmine or UXO sensor signatures that are, for algorithm-training purposes [4], generally representative (for all landmine and UXO sites). In this paper we investigate a technique whereby detection and classification algorithms may be designed for sensing buried landmines and UXO without requiring a separate training set of representative target and clutter signatures. The approach is based on the realization that, when sensing landmines and UXO, one will eventually excavate buried targets based on the sensor data. The approach developed here chooses which items to excavate initially, based on their importance in design of the associated detection and classification algorithm.

Let $\{\mathbf{x}_i\}_{i=1,N}$ represent the *known* measured signatures of the N subsurface objects at a given site, with the set of all $\mathbf{x}_i$ denoted as $\mathbf{X}$. Further, let $\{y_i\}_{i=1,N}$ represent the associated *unknown* binary labels (target/non-target) of the signatures, to be determined in the detection phase. We here develop a kernel-based classifier, by which an observed signature or feature vector $\mathbf{x}$ is classified using the function

$$f(\mathbf{x}) = \sum_{i=1}^{n} w_i K(\mathbf{x}, \mathbf{b}_i) + w_o \qquad (1)$$

where $\mathbf{b}_i$ is the ith basis function, w_i are scalar weights, w_o is a scalar offset or bias, and $K(\mathbf{x},\mathbf{b}_i)$ is a general kernel that defines the similarity of $\mathbf{x}$ to $\mathbf{b}_i$. For a prescribed threshold

t, $\mathbf{x}$ is deemed associated with the +1 class if $f(\mathbf{x}) \geq t$, and associated with the -1 class if $f(\mathbf{x})<t$, and by varying the threshold t one yields the receiver operating characteristic[1] (ROC) [4]. Algorithms that utilize the form in (1) include the support vector machine (SVM) [5-6], the relevance vector machine (RVM) [7], kernel matching pursuits (KMP) [8], as well as many other related algorithms [9-11].

In design of a classifier of the form in (1), the $\mathbf{b}_i$ typically come from a separate training set, for which the associated labels y_i are known. In this case the goal is to design a classifier of the form in (1) that correctly identifies the labels of the training data (when $\mathbf{x}=\mathbf{b}_i$ for any i), with the hope that this will generalize to data $\mathbf{x}$ not observed while training. The aforementioned variability of subsurface-target signatures makes the idea of utilizing a separate training set undesirable and often impractical.

In the approach taken in this paper, the set of basis functions $\mathbf{B}_n = \{\mathbf{b}_i\}_{i=1,n}$ is selected from the set of observed data $\mathbf{X}$, *i.e.* $\mathbf{B}_n \subset \mathbf{X}$. This is done because such basis functions will be well matched to the data to be classified, *vis-à-vis* other data that may have come from a different site. The set $\mathbf{B}_n$ is defined by selecting those signatures from $\mathbf{X}$ that are most representative of the measured data from the site of interest, using fundamental information-theoretic considerations to be detailed below. Note that the labels (identities) of the subsurface objects associated with $\mathbf{B}_n$ are not required at this point. Having defined the basis set for (1), we must determine the associated model weights $\{w_i\}_{i=1,n}$ and w_o (denoted collectively by the vector $\mathbf{w}$), and for this task we require labeled data. Therefore, we define a subset of signatures $\mathbf{X}_s \subset \mathbf{X}$ for which knowledge of the associated labels $\mathbf{L}_s$ would be most informative in the context of defining the model weights. The set of signatures $\mathbf{X}_s$ is again determined via information-theoretic metrics detailed below. Note that the sets $\mathbf{B}_n$ and $\mathbf{X}_s$ may overlap, but they are in general distinct. After excavating the items associated with $\mathbf{X}_s$, yielding $\mathbf{L}_s$, the algorithm

[1] Rigorously speaking, the ROC was originally developed for a likelihood-ratio test [4], and therefore what we consider here is arguably ROC-like. For notational convenience, throughout we refer to such as an ROC.

in (1) is trained as usual [8] and then applied to $\mathbf{x} \notin \mathbf{X}_s$. The key point is that the training set ($\mathbf{X}_s$,$\mathbf{L}_s$) is determined adaptively on the observed site-dependent data, via fundamental information-theoretic metrics.

We demonstrate using measured EMI and magnetometer data from an actual UXO site that the sets $\mathbf{X}_s$ and $\mathbf{L}_s$ are often of small dimension, thereby minimizing the amount of excavation required for algorithm design. Once the algorithm has been designed, the fact that it is well matched to the environment often yields a significant reduction in the false-alarm rate, thereby ultimately reducing the total number of excavations (*i.e.* the false-alarm probability is reduced, and therefore less excavation is required of clutter).

The remainder of the paper is organized as follows. In Sec. II we discuss the selection of basis functions $\mathbf{B}_n$ and labeled data $\mathbf{X}_s$. We present in Sec. III example results on EMI and magnetometer detection of UXO from an actual UXO site. The work is summarized in Sec. IV.

II. ACTIVE CLASSIFIER DESIGN

A. Model structure

The decision function in (1), using n basis functions, may be expressed concisely as [8]

$$f_n(\mathbf{x}) = \sum_{i=1}^{n} w_{n,i} K(\mathbf{x}, \mathbf{b}_i) + w_{n,0} = \mathbf{w}_n^T \boldsymbol{\phi}_n(\mathbf{x}) \tag{2}$$

where

$$\boldsymbol{\phi}_n(\mathbf{x}) = [1, K(\mathbf{x}, \mathbf{b}_1), K(\mathbf{x}, \mathbf{b}_2), ..., K(\mathbf{x}, \mathbf{b}_n)]^T \tag{3}$$

$$\mathbf{w}_n = [w_{n,0}, w_{n,1}, w_{n,2}, \cdots, w_{n,n}]^T \tag{4}$$

Assume that the basis set $\mathbf{B}_n = \{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is known. Moreover, assume that the item associated with signature $\mathbf{x}_i$ is excavated (this is termed an "experiment"), from which we

learn the associated label y_i, where by construction y_i=1 for one class and y_i=-1 for the other class (target/no-target). The label found by the experiment is related to the prediction $f_n(\mathbf{x})$ by

$$y_i = \mathbf{w}_n^T \boldsymbol{\phi}_n(\mathbf{x}_i) + \varepsilon_i \tag{5}$$

where $\varepsilon(\mathbf{x}_i)$ is the error term resulting from imperfections in the model. In algorithm design one desires weights **w** that minimize the error observed on training data, for which the data and labels are known. If the training data are well matched to the subsequent testing data, then the algorithm is likely to constitute a robust detection procedure. As indicated above, in many subsurface-sensing problems it is impractical to have a separate training set, with this addressed by the information-theoretic techniques discussed below.

B. Selection of basis functions

If we assume that the ε_i in (5) are Gaussian and independent with variance σ_i^2, then the Fisher information matrix associated with **X** and $\mathbf{B}_n$ is expressed as [12]

$$\mathbf{M}_n = \sum_{i=1}^{N} \sigma_i^{-2} \boldsymbol{\phi}_n(\mathbf{x}_i) \boldsymbol{\phi}_n^T(\mathbf{x}_i) = \sum_{i=1}^{N} \sigma_i^{-2} \boldsymbol{\phi}_{n,i} \boldsymbol{\phi}_{n,i}^T \tag{6}$$

where $\boldsymbol{\phi}_{n,i} \equiv \boldsymbol{\phi}_n(\mathbf{x}_i)$. Note that in computing $\mathbf{M}_n$ we do not require the labels associated with $\mathbf{B}_n$ and **X** (this is a result of the fact that the model in (2) is linear in the weights $\mathbf{w}_n$). As discussed by Fedorov [12], the Fisher information matrix in (6) is associated with the uncertainty in the model weights **w**, as defined through all N measured $\mathbf{x}_i$ and the basis $\mathbf{B}_n$. By appending a new basis function to $\boldsymbol{\phi}_n(\cdot)$, one obtains

$$\boldsymbol{\phi}_{n+1}(\cdot) = \begin{bmatrix} \boldsymbol{\phi}_n(\cdot) \\ \phi_{n+1}(\cdot) \end{bmatrix} \tag{7}$$

where $\phi_{n+1}(\cdot) = K(\cdot\,, \mathbf{b}_{n+1})$ and $\mathbf{b}_{n+1} \in \mathbf{X}$, $\mathbf{b}_{n+1} \notin \mathbf{B}_n$. Following (2), we can write from $\boldsymbol{\phi}_{n+1}$ the augmented classifier f_{n+1}, for which the Fisher information matrix is found to be

$$\mathbf{M}_{n+1} = \sum_{i=1}^{N} \sigma_i^{-2} \begin{bmatrix} \boldsymbol{\phi}_{n,i} \\ \phi_{n+1,i} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_{n,i}^T & \phi_{n+1,i} \end{bmatrix} = \begin{bmatrix} \mathbf{M}_n & \sum_{i=1}^{N} \sigma_i^{-2} \boldsymbol{\phi}_{n,i} \phi_{n+1,i} \\ \sum_{i=1}^{N} \sigma_i^{-2} \phi_{n+1,i} \boldsymbol{\phi}_{n,i}^T & \sum_{i=1}^{N} \sigma_i^{-2} \phi_{n+1,i}^2 \end{bmatrix} \tag{8}$$

where $\phi_{n+1,i} \equiv \phi_{n+1}(\mathbf{x}_i)$. The expression in (8) is again associated with fitting the model to the N measured $\mathbf{x}_i$, but now using an (n+1)-dimensional basis $\mathbf{B}_{n+1}$, *vis-à-vis* the n-dimensional basis $\mathbf{B}_n$ in (6). We develop a metric which compares (6) and (8), thereby quantifying the information gain by adding the new basis $\mathbf{b}_{n+1}$.

There are many ways of comparing the information content reflected by $\mathbf{M}_n$ and $\mathbf{M}_{n+1}$, and here we employ the so-called D-optimal procedure [12], defined as the determinant of the information matrix. The logarithm of the determinant of $\mathbf{M}$ is denoted q_n, and it may be shown that

$$q_{n+1} = q_n + \ln r(\mathbf{b}_{n+1}) \tag{9}$$

where

$$r(\mathbf{b}_{n+1}) = \sum_{i=1}^{N} \sigma_i^{-2} \phi_{n+1,i}^2 - \sum_{i=1}^{N} \sigma_i^{-2} \phi_{n+1,i} \boldsymbol{\phi}_{n,i}^T \mathbf{M}_n^{-1} \sum_{i=1}^{N} \sigma_i^{-2} \boldsymbol{\phi}_{n,i} \phi_{n+1,i} \tag{10}$$

Since $N \geq n$ the matrix $\mathbf{M}_n$ is full rank and its inverse exists (assuming that n of the vectors $\{\boldsymbol{\phi}_n(\mathbf{x}_i)\}_{i=1,N}$ are linearly independent). Under these conditions, it can be shown that $r > 0$, and therefore $\ln r$ in (9) is generally valid. Note that if $r(\mathbf{b}_{n+1})$=0, then $\mathbf{M}_{n+1}$ is rank-deficient, and we delete the basis function from the candidate set and proceed to select from the remaining ones (however, we haven't found $r(\mathbf{b}_{n+1})$=0 in practice).

It is known from information theory [13] that the inverse of $\mathbf{M}_n$ gives the Cramer-Rao lower bound (CRLB) of the covariance matrix of the estimate of $\mathbf{w}_n$ and the reciprocal of q_n lower bounds the product of its eigenvalues. The CRLB is here the actual covariance, assuming the Gaussian model. A larger q_n implies low variances of the components of $\mathbf{w}_n$. Given the nth order decision function f_n, q_n is fixed, and one relies on maximization of $\ln r(\mathbf{b}_{n+1})$ to obtain a large value of q_{n+1}. This can be achieved by

conducting a "greedy" search for the new $\mathbf{b}_{n+1}$ in $\mathbf{X}$ with the previously selected support data excluded

$$\mathbf{b}_{n+1} = \arg\max_{\mathbf{b}\in\mathbf{X},\mathbf{b}\notin\mathbf{B}_n} \ln r(\mathbf{b}) \tag{11}$$

Using the procedure outlined above, basis elements $\mathbf{b}_n$ are added until the information gain reflected in q_{n+1}-q_n is no longer deemed significant. Note from (9)-(10) that evaluation of (11) does not require knowledge of the target labels y_i, and therefore no excavation is required to determine the basis $\mathbf{B}_n$. The greedy method is suboptimal, but in practice often provides good results.

C. Selection of labeled data, for model training

Assume that the procedure discussed above selects n bases from the observed data $\mathbf{X}$. We now require labeled data to optimize the associated model weights $\mathbf{w}$. In a manner analogous to the previous discussion, we select those $\mathbf{x}_i \in \mathbf{X}$ for which knowledge of the associated labels y_i would be most informative in the context of defining $\mathbf{w}$. Those $\mathbf{x}_i$ that are so selected define a subset of signatures $\mathbf{X}_s \subset \mathbf{X}$, and these items are excavated to yield the respective set of labels $\mathbf{L}_s$. The set of signatures and labels ($\mathbf{X}_s$, $\mathbf{L}_s$) are then used to define the weights $\mathbf{w}$ in a least-squares sense, and the resulting model $f(\mathbf{x})$ is used to specify which of the remaining signatures $\mathbf{x} \notin \mathbf{X}_s$ are likely targets of interest.

Assume that there are J signatures in $\mathbf{X}_s$, denoted $\mathbf{X}_{s,J}$. We quantify the information context in $\mathbf{X}_{s,J}$ in the context of estimating the model weights $\mathbf{w}$, and further ask which $\mathbf{x}_i \notin \mathbf{X}_{s,J}$ would be most informative if it and its label were added for determination of $\mathbf{w}$. Analogous to (6), we have

$$\mathbf{M}_n(\mathbf{X}_{s,J}) = \sum_{i:\mathbf{x}_i\in\mathbf{X}_{s,J}} \sigma_i^{-2}\boldsymbol{\phi}_{n,i}\boldsymbol{\phi}_{n,i}^T \tag{12}$$

The expressions in (6) and (12) both employ an n-dimensional basis set $\mathbf{B}_n \subset \mathbf{X}$. The distinction is that in (6) we are interested in defining $\mathbf{B}_n$, and we sum over all observed signatures $\{\mathbf{x}_i\}_{i=1,N}$. By contrast, in (12) the basis set $\mathbf{B}_n$ is known and fixed, and we are only summing over those signatures $\mathbf{X}_{s,J}$ for which knowledge of the associated labels is most informative in defining the model weights $\mathbf{w}$.

After adding a new signature $\mathbf{x}_i \in \mathbf{X}$, $\mathbf{x}_i \notin \mathbf{X}_{s,J}$, we now have $\mathbf{X}_{s,J+1}$ and $\mathbf{M}_n$ is updated as

$$\mathbf{M}_n(\mathbf{X}_{s,J+1}) = \mathbf{M}_n(\mathbf{X}_{s,J}) + \sigma_{i_{J+1}}^{-2} \boldsymbol{\phi}_{n,i_{J+1}} \boldsymbol{\phi}_{n,i_{J+1}}^T \tag{13}$$

where i_{J+1} represents the index of the new signature selected for $\mathbf{X}_{s,J+1}$. Using the matrix identity $\det(\mathbf{A}+\mathbf{F}\mathbf{F}^T)=\det(\mathrm{I}+\mathbf{F}^T\mathbf{A}^{-1}\mathbf{F})\det(\mathbf{A})$, one obtains from (13)

$$q_n(\boldsymbol{X}_{s,J+1}) = q_n(\boldsymbol{X}_{s,J}) + \ln \rho(\boldsymbol{x}_{i_{J+1}}) \tag{14}$$

with

$$\rho(\mathbf{x}_{i_{J+1}}) = 1 + \sigma_{i_{J+1}}^{-2} \boldsymbol{\phi}_{n,i_{J+1}}^T \mathbf{M}_n^{-1}(\mathbf{X}_{s,J}) \boldsymbol{\phi}_{n,i_{J+1}} \tag{15}$$

Care is needed with regard to evaluating the inverse of $\mathbf{M}_n$, since if $J<n$ the matrix is rank deficient. We have considered addressing this in either of two ways. A standard approach for inversion of such matrices is to add a small diagonal term to $\mathbf{M}_n$, such that its inverse exists. Alternatively, by construction one can assume that the items associated with the basis $\mathbf{B}_n$ are all associated with $\mathbf{X}_{s,J}$, yielding a minimum of n labeled data and therefore assuring that the matrix is full rank. We have examined both procedures, and they yield comparable results. We use the second approach in all examples presented in Sec. III.

Having addressed the inverse of $\mathbf{M}_n$, one iteratively maximizes $\ln \rho(\mathbf{x}_{i_{J+1}})$ to obtain

$$\mathbf{x}_{i_{J+1}} = \arg\max_{\mathbf{x} \in \mathbf{X}, \mathbf{x} \notin \mathbf{X}_{s,J}} \ln \rho(\mathbf{x}) \tag{16}$$

Note that to define $\mathbf{x}_{i_{J+1}}$ we again do not require the signature labels. The elements of $\mathbf{X}_s$ are selected iteratively, in a "greedy" fashion as indicated in (16), until the information gain is below a prescribed threshold. After J iterations we have defined those signatures $\mathbf{X}_{s,J}$ for which knowledge of the labels will best approximate the weights $\mathbf{w}$. These items are excavated, yielding the labels $\mathbf{L}_{s,J}$.

For the assumptions underlying the linear model in (5), and assuming knowledge of $\mathbf{B}_n$ and ($\mathbf{X}_{s,J}$, $\mathbf{L}_{s,J}$) the optimal estimation for the weights $\mathbf{w}$ is expressed as [8,12]

$$\mathbf{w} = [\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}]^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{y} \tag{17}$$

where **y** represents the set of labels determined via the J excavations

$$\mathbf{y} = \{y_{i_1}, y_{i_2}, ..., y_{i_J}\}^{\mathrm{T}} \tag{18}$$

and the $J \times (n+1)$ matrix $\mathbf{\Phi}$ is defined as

$$\mathbf{\Phi} = \begin{bmatrix} \boldsymbol{\phi}_n^{\mathrm{T}}(\mathbf{x}_{i_1}) \\ \boldsymbol{\phi}_n^{\mathrm{T}}(\mathbf{x}_{i_2}) \\ \cdot \\ \cdot \\ \cdot \\ \boldsymbol{\phi}_n^{\mathrm{T}}(\mathbf{x}_{i_J}) \end{bmatrix} \tag{19}$$

where, for example, $\mathbf{x}_{i_1}$ corresponds to y_{i_1}.

In the classification stage we consider $\mathbf{x} \notin \mathbf{X}_{s,J}$ and compute $f(\mathbf{x})$. For a prescribed threshold t, **x** is deemed associated with the +1 class if $f(\mathbf{x}) \geq t$, and associated with the -1 class if $f(\mathbf{x})<t$, and by varying the threshold t one yields the receiver operating characteristic (ROC) [4]. The key component of the model $f(\mathbf{x})$ is that it is linear in the weights **w**, which yields a closed-form procedure for selection of $\mathbf{B}_n$ and $\mathbf{X}_{s,J}$, as indicated in the previous sections.

III. THEORETICAL MOTIVATION FOR CLASSIFIER DESIGN

In the previous two sections we have presented procedures for selecting basis functions for a kernel-based classifier, based on a set of unlabeled data. After designing the basis set, we have also addressed selection of which signatures would be most informative for classifier training, if the associated signature labels were known. In this section we provide theoretical justification for these design procedures, and in Sec. IV example results are presented for UXO sensing.

A. Basis-function selection

To simplify notation, we utilize matrix expressions in our derivation. Let the basis functions $\boldsymbol{\phi}_n(\cdot)$ be evaluated for all initially unlabeled data points $\{\mathbf{x}_i\}_{i=1,N}$, and stacked to form the matrix $\widetilde{\boldsymbol{\Phi}}_n = [\boldsymbol{\phi}_n(\mathbf{x}_1), \boldsymbol{\phi}_n(\mathbf{x}_2), ..., \boldsymbol{\phi}_n(\mathbf{x}_N)]^{\mathrm{T}}$. Let the data labels be denoted $\mathbf{y} = \{y_1, y_2, ..., y_N\}^{\mathrm{T}}$, although these labels are not required when designing the basis functions. The difference between the true labels and those output by the classifier (2), for all $\{\mathbf{x}_i\}_{i=1,N}$ is expressed in vector form as

$$\begin{aligned}\mathbf{y} - \widetilde{\boldsymbol{\Phi}}_n(\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}}\widetilde{\boldsymbol{\Phi}}_n)^{-1}\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}}\mathbf{y} &\overset{1}{=} (\mathbf{I}_N - \widetilde{\boldsymbol{\Phi}}_n(\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}}\widetilde{\boldsymbol{\Phi}}_n)^{-1}\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}})\mathbf{y} \\ &\overset{2}{\approx} (\mathbf{I}_N - \widetilde{\boldsymbol{\Phi}}_n(\lambda\mathbf{I}_{n+1} + \widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}}\widetilde{\boldsymbol{\Phi}}_n)^{-1}\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}})\mathbf{y} \\ &\overset{3}{=} (\mathbf{I}_N + \frac{1}{\lambda}\widetilde{\boldsymbol{\Phi}}_n\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}})^{-1}\mathbf{y}\end{aligned} \tag{20}$$

where $\mathbf{I}_N$ is a $N \times N$ identity matrix ($\mathbf{I}_n$ is defined similarly), and λ is a small positive number. The equality 3 in (20) is due to the Sherman-Morrison-Woodbury formula. From (20) the squared error between the true and estimated labels is

$$e_n^2 \approx \mathbf{y}^{\mathrm{T}}(\mathbf{I}_N + \frac{1}{\lambda}\widetilde{\boldsymbol{\Phi}}_n\widetilde{\boldsymbol{\Phi}}_n^{\mathrm{T}})^{-2}\mathbf{y} \tag{21}$$

The expression in (21) shows that for given basis functions $\boldsymbol{\phi}_n(\cdot)$ we have approximately expressed the squared error as a quadratic form of the labels $\mathbf{y}$, with a coefficient matrix $\mathbf{C}_n^{-2}$ with $\mathbf{C}_n = \mathbf{I}_N + \widetilde{\boldsymbol{\Phi}}_n\widetilde{\boldsymbol{\Phi}}_n^T/\lambda$. The approximation can be made as accurate as desired by making λ sufficiently small. Without knowing $\mathbf{y}$, we prefer $\mathbf{C}_n$ to have large eigenvalues, to make the error e_n^2 small. This is accomplished by making the determinant of $\mathbf{C}_n$ large. The logarithmic determinant of $\mathbf{C}_n$ is

$$\begin{aligned}q_n^{(2)} &\overset{1}{=} \ln\det(\mathbf{C}_n) \overset{2}{=} \ln\det(\mathbf{I}_N + \frac{1}{\lambda}\widetilde{\boldsymbol{\Phi}}_n\widetilde{\boldsymbol{\Phi}}_n^T) \\ &\overset{3}{=} \ln\frac{\det(\lambda\mathbf{I}_{n+1} + \widetilde{\boldsymbol{\Phi}}_n^T\widetilde{\boldsymbol{\Phi}}_n)}{\lambda^{n+1}} \\ &\overset{4}{=} \ln\frac{\det(\lambda\mathbf{I}_{n+1} + \mathbf{M}_n)}{\lambda^{n+1}}\end{aligned} \tag{22}$$

where equality 3 is due to the property of matrix determinants and equality 4 is due to (6). Adding a new basis function to $\boldsymbol{\phi}_n(\cdot)$, we get $\boldsymbol{\phi}_{n+1}(\cdot)$ as given in (7). The logarithmic determinant of $\mathbf{C}_{n+1} = \mathbf{I}_N + \widetilde{\boldsymbol{\Phi}}_{n+1}\widetilde{\boldsymbol{\Phi}}_{n+1}^T/\lambda$ is

$$q_{n+1}^{(2)} = \ln \frac{\det(\lambda \mathbf{I}_{n+2} + \mathbf{M}_{n+1})}{\lambda^{n+2}} \tag{23}$$

Following the method of obtaining (9)-(10), we can show that $q_n^{(2)}$ and $q_{n+1}^{(2)}$ are related by

$$q_{n+1}^{(2)} = \ln q_n^{(2)} + \ln \frac{r^{(2)}(\phi_{n+1})}{\lambda} \tag{24}$$

with

$$r^{(2)}(\phi_{n+1}) = \lambda + \sum_{i=1}^{N} \phi_{n+1,i}^2 - \sum_{i=1}^{N} \phi_{n+1,i} \boldsymbol{\phi}_{n,i}^T (\lambda \mathbf{I}_{n+1} + \mathbf{M}_n)^{-1} \sum_{i=1}^{N} \boldsymbol{\phi}_{n,i} \phi_{n+1,i} \tag{25}$$

where $\boldsymbol{\phi}_{n,i} \equiv \boldsymbol{\phi}_n(\mathbf{x}_i)$ and $\phi_{n+1,i} \equiv \phi_{n+1}(\mathbf{x}_i)$. Since we wish for a $\mathbf{C}_{n+1}$ with large determinant, we want to make $\ln \frac{r^{(2)}(\phi_{n+1})}{\lambda}$ or equivalently $\ln r^{(2)}(\phi_{n+1})$ large, as λ is a constant.

Comparing (25) to (10), we find that $r^{(2)}$ is approximately equal to r when λ is small. Since λ can be made as small as desired, the approximation can be made arbitrarily accurate. Therefore the basis function obtained in (11) is the one that minimizes the determinant of $\mathbf{C}_{n+1}$ given $\mathbf{C}_n$, which in consequence will minimize the eigenvalues of $\mathbf{C}_{n+1}$, minimizing the squared error e_{n+1}^2 .

B. Selection of examples for labeling

Assume the basis functions $\boldsymbol{\phi}_n(\cdot)$ have been selected in the manner discussed above. Moreover, assume we have selected the subset $\mathbf{X}_{s,J} = \{\mathbf{x}_{i_1}, \mathbf{x}_{i_2}, ..., \mathbf{x}_{i_J}\}$ of J signatures for which the associated labels will be acquired. The Fisher information matrix

associated with $\mathbf{X}_{s,J}$ is $\mathbf{M}_n(\mathbf{X}_{s,J}) = \sum_{k=1}^{J} \boldsymbol{\phi}_{n,i_k} \boldsymbol{\phi}_{n,i_k}^T$. The Fisher information matrix for an augmented set $\mathbf{X}_{s,J} \bigcup \{\mathbf{x}\} = \{\mathbf{x}_{i_1}, \mathbf{x}_{i_2}, ..., \mathbf{x}_{i_J}, \mathbf{x}\}$ is

$$\mathbf{M}_n(\mathbf{X}_{s,J} \bigcup \{\mathbf{x}\}) = \mathbf{M}_n(\mathbf{X}_{s,J}) + \boldsymbol{\phi}_n(\mathbf{x})\boldsymbol{\phi}_n^T(\mathbf{x}) \tag{26}$$

Suppose we have two classifiers, $f_n^{\mathbf{X}_{s,J} \cup \{\mathbf{x}\}}(\cdot)$ and $f_n^{\mathbf{X}_{s,J}}(\cdot)$, which are trained using $\mathbf{X}_{s,J} \bigcup \{\mathbf{x}\}$ and $\mathbf{X}_{s,J}$, respectively. We test $f_n^{\mathbf{X}_{s,J} \cup \{\mathbf{x}\}}(\cdot)$ and $f_n^{\mathbf{X}_{s,J}}(\cdot)$ on $\mathbf{x}$ and examine how the two results are related. As given in [14, page 121], we have

$$[f_n^{\mathbf{X}_{s,J}}(\mathbf{x}) - y(\mathbf{x})]^2 = \frac{[f_n^{\mathbf{X}_{s,J} \cup \{\mathbf{x}\}}(\mathbf{x}) - y(\mathbf{x})]^2}{1 - \boldsymbol{\phi}_n^T(\mathbf{x})[\mathbf{M}_n(\mathbf{X}_{s,J}) + \boldsymbol{\phi}_n(\mathbf{x})\boldsymbol{\phi}_n^T(\mathbf{x})]^{-1}\boldsymbol{\phi}_n(\mathbf{x})} \tag{27}$$

By using the Sherman-Morrison-Woodbury formula, we obtain

$$\begin{aligned}
&[\mathbf{M}_n(\mathbf{X}_{s,J}) + \boldsymbol{\phi}_n(\mathbf{x})\boldsymbol{\phi}_n^T(\mathbf{x})]^{-1} \\
&= \mathbf{M}_n^{-1}(\mathbf{X}_{s,J}) - \frac{\mathbf{M}_n^{-1}(\mathbf{X}_{s,J})\boldsymbol{\phi}_n(\mathbf{x})\boldsymbol{\phi}_n^T(\mathbf{x})\mathbf{M}_n^{-1}(\mathbf{X}_{s,J})}{1 + \boldsymbol{\phi}_n^T(\mathbf{x})\mathbf{M}_n^{-1}(\mathbf{X}_{s,J})\boldsymbol{\phi}_n(\mathbf{x})}
\end{aligned}$$

which is set into (27) to give

$$[f_n^{\mathbf{X}_{s,J} \cup \{\mathbf{x}\}}(\mathbf{x}) - y(\mathbf{x})]^2 = \frac{[f_n^{\mathbf{X}_{s,J}}(\mathbf{x}) - y(\mathbf{x})]^2}{1 + \boldsymbol{\phi}_n^T(\mathbf{x})\mathbf{M}_n^{-1}(\mathbf{X}_{s,J})\boldsymbol{\phi}_n(\mathbf{x})} \tag{28}$$

Equation (28) shows that by including $\mathbf{x}$ in the training data set, the squared test error on $\mathbf{x}$ will drop by a factor

$$\rho^{(2)}(\mathbf{x}) = 1 + \boldsymbol{\phi}_n^T(\mathbf{x})\mathbf{M}_n^{-1}(\mathbf{X}_{s,J})\boldsymbol{\phi}_n(\mathbf{x}) \tag{29}$$

If $\rho^{(2)}(\mathbf{x}) \approx 1$, we do not require the label for $\mathbf{x}$, as it does not important for inclusion in the training set. On the other hand, if $\rho^{(2)}(\mathbf{x}) >> 1$, inclusion of $\mathbf{x}$ in the training set is important. Therefore, the $\mathbf{x}$ that maximizes $\rho^{(2)}(\mathbf{x})$ should be selected to seek the associated label *y*. Comparing (29) to (15) we note that $\rho^{(2)}(\mathbf{x})$ is exactly equivalent to $\rho(\mathbf{x})$, and thus the $\mathbf{x}$ that maximizes $\rho(\mathbf{x})$ is the one that contributes the maximally to make the squared test error small.

IV. APPLICATION TO UXO DETECTION

The active-training methodology addressed in this paper may be applied to any detection problem for which the data labels are expensive to acquire, and for which there is no distinct training data. In particular, we consider the detection of buried UXO. For UXO remediation, the label of a potential target is acquired by excavation, a dangerous and time consuming task. The overwhelming majority of UXO cleanup costs come from excavation of non-UXO items. In this context, note that *a priori* excavations are required for the procedure in Sec. II (to obtain labeled training data). However, if the false-alarm rate is reduced at the desired detection probability, then overall cleanup costs may diminish substantially (*i.e.*, overall, less non-UXO items need be excavated).

The results presented here are for data collected at an actual UXO site: Jefferson Proving Ground in the United States. The technique in Sec. II is compared with results obtained using existing procedures. Specifically, the principal challenge in UXO sensing is development of a training set, for design of the detection algorithm. At an actual UXO site there is often a significant quantity of UXO, UXO fragments and man-made clutter *on the surface*. It has been recognized that the characteristics of the surface UXO and clutter is a good indicator of what will be found in the subsurface. Consequently, in practice, a subset of the surface UXO and clutter are buried, and magnetometer and induction data are collected for these items, for which the labels are obviously known. The measured data and associated labels (UXO/non-UXO) are then used for training purposes. Of course, the process of burying, collecting data, and then excavating these emplaced items is time consuming and dangerous (for the UXO items), with this procedure eliminated by the techniques outlined in Sec. II.

A. Magnetometer and electromagnetic induction sensors

Magnetometer and electromagnetic induction (EMI) sensors are widely applied in sensing buried conducting/ferrous targets, such as landmines and UXO. The

magnetometer is a passive sensor that measures the change of the earth's background magnetic field due to the presence of a ferrous target. Magnetometers measure static magnetic fields. An EMI sensor actively transmits a time-varying electromagnetic field, and consequently senses the dynamic induced secondary field from the target. To enhance soil penetration, EMI sensors typically operate at kilohertz frequencies. We here employ a frequency-domain EMI sensor that transmits and senses at several discrete frequencies [15]. Magnetometers only sense ferrous targets, while EMI sensors detect general conducting and ferrous items.

Parametric models have been developed for both magnetometer and EMI sensors [16-18]. The target features **x** are extracted by fitting the EMI and magnetometer models to measured sensor data. The vector **x** has parameters from both the magnetometer and EMI data, and therefore in this sense the data from these two sensors are "fused". The one place where these two models have overlapping parameters is in specification of the target position. The magnetometer data often yields a very good estimation of the target position, and therefore such are used in **x**. In fact, the target position specified by the magnetometer data is explicitly utilized as prior information when fitting EMI data to the EMI parametric model. Details on the magnetometer and EMI models, and on the model-fitting procedure, may be found in [18].

The features employed are as in [18], and the features are centered and normalized. Specifically, using the training data, we compute the mean feature vector $\mathbf{x}_{mean}$ and the variance of each feature component (let σ_i^2 represent the variance of the ith feature). Before classification, a given feature vector **x** is shifted by implementing $\mathbf{x}_{shift}=\mathbf{x}-\mathbf{x}_{mean}$, and then the ith feature component of $\mathbf{x}_{shift}$ is divided by σ_i to effect the normalization.

B. Measured sensor data from the Jefferson Proving Ground

Jefferson Proving Ground (JPG) is a former military range that has been utilized for UXO technology demonstrations since 1994. We consider data collected by Geophex,

Ltd. in the latest phase (Phase V) of the JPG demonstration. The goal of the JPG V is to evaluate the UXO detection and discrimination abilities under realistic scenarios, where man-made and natural clutter coexist with UXO items. Our results are presented with the GEM-3 and magnetometer data from two adjoining areas, constituting a total of approximately five acres. There are 433 potential targets detected from sensor anomalies, 40 of which are proven to be UXO and the others are clutter. The excavated UXO items include 4.2 inch, 60 mm, and 81mm mortars; 5 inch, 57 mm, 76 mm, 105 mm, 152 mm, and 155 mm projectiles; and 2.75 inch rockets.

This test was performed with US Army oversight. One of the two JPG areas was assigned as the training area, for which the ground truth (UXO/non-UXO) was given. The trained detection algorithms are then tested on the other area, and the associated ground truth was revealed later to evaluate performance. It was subsequently recognized that several UXO types were found in equal number in each of the two areas. This indicates an effort to match the training data to the detection data, in the manner discussed above, involving burial of known UXO and non-UXO collected on the surface.

Each sensor anomaly is processed by fitting the associated magnetometer and EMI data to the parametric models [18], and the estimated parameters define $\mathbf{x}$. In addition, the model-fitting procedure functions as a prescreening tool. Any sensor anomaly failing to fit well to the model is regarded as having been generated by a clutter item. Therefore, a total of 300 potential targets remain after this prescreening stage, 40 of which are UXO. In the training area, there are 128 buried items, 16 of which are UXO.

C. Detection results

Before presenting classification results, we examine the characteristics of the basis functions selected in the first phase of the algorithm, prior to adaptively selecting training data. In Fig. 1 we consider the first three basis functions $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ selected by the first stage of the algorithm. For each feature vector $\mathbf{x}$ (from all UXO and non-UXO), we compute a three-dimensional vector ($K(\mathbf{x},\mathbf{b}_1)$, $K(\mathbf{x},\mathbf{b}_2)$, $K(\mathbf{x},\mathbf{b}_3)$). By examining this three-dimensional vector for all $\mathbf{x}$, we may observe the degree to which UXO and non-

UXO features are distinguished via the features and kernel. A radial basis function kernel is employed here, corresponding to the kernel used to select the basis functions (see discussion below concerning the selected kernel). By examining Fig. 1 we observe that the UXO and non-UXO features are relatively separated, although there is significant overlap, undermining classification performance.

The detection results are presented in the form of the receiver operating characteristic (ROC), quantifying the probability of detection (Pd) as a function of the false alarm count. We present ROC curves using the adaptive-training approach discussed in Sec. II, with performance compared to results realized by training on the distinct training region discussed above (the latter approach reflects current practice). With regard to conventional training, the algorithm employed is of identical form as (2), with model weights determined iteratively using kernel matching pursuits (KMP). Details on the KMP algorithm may be found in [8] (we have employed the prefitting algorithm in [8]). To make the comparison appropriate, the adaptive training and KMP implementation employ an identical radial basis function (RBF) kernel [10]

$$K(\mathbf{x},\mathbf{b}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp[-\|\mathbf{x}-\mathbf{b}\|_2^2 / \sigma^2] \qquad (30)$$

The variance σ^2 is adaptively adjusted after each basis vector is selected before the model weights are determined, and it is not related to the labeled training data. In particular, a gradient search is applied to refine σ^2 with equation (11), by maximizing $\ln r(\mathbf{b})$.

As indicated above, the designated training area has 128 labeled items, and conventionally classifiers are tested on the remaining signatures, in this case constituting 172 items. As a first comparison, the adaptive technique discussed in Sec. II is employed to select J=128 items from the original 300, with these "excavated" to learn the associated labels. This therefore defines the set $\mathbf{X}_{s,J}$ and associated labels $\mathbf{L}_{s,J}$. The basis set $\mathbf{B}_n$ is also determined adaptively using the original 300 signatures, and here n=10. The performance of the adaptive learning algorithm is then tested on the remaining 172 $\mathbf{x}_i \notin \mathbf{X}_{s,J}$, although these are generally not the same testing examples used via traditional

training of the KMP algorithm (the training sets do not overlap completely). For comparison, we also show training and testing results implemented via KMP, in which the 128 training examples are selected randomly from the original 300 signatures **X**. Performance comparisons are shown in Fig. 2, wherein we present results for active data selection (algorithm in Sec. II), KMP results using the assigned 128 training examples, and average results for randomly choosing the 128 examples for KMP training (100 random selections were performed). In addition, for the latter case we also place error bars on the results; the length of the error bar is twice the standard derivation of the Pd for the associated false-alarm count. Therefore, if the result is Gaussian distributed, 95% of the values lie within the error bar.

Before proceeding we note that the ROC curves are generated by varying the threshold t, as applied to the estimated label y. For the binary UXO-classification problem considered here, by design we choose the label y=1 for UXO and y=0 for non-UXO. In practice one must choose one point on the ROC at which to operate. A naïve choice of the operating point would be 0.5 (*i.e.*, if the classifier maps a testing feature vector $\boldsymbol{x}$ to a label y>0.5 the item is declared UXO, and otherwise it is declared non-UXO). However, we must account for the fact that in practice the number of non-UXO items is often much larger than the number of UXO. We have therefore invoked the following procedure.

We assume that the error (noise) between the true label (y=1 or y=0) and the estimated label is *i.i.d.* Gaussian with variance of σ^2, as in (5). Let N_0 and N_1 represent respectively the number of non-UXO and UXO items in the training set. Considering the UXO (y=1) data, an unbiased estimator of the label y will yield a mean of one and a minimum variance of σ^2/N_1. Similarly, considering the non-UXO data (y=0), an unbiased estimator of the label y will have zero mean and minimum variance σ^2/N_0. Let H_1 and H_0 correspond to the UXO and non-UXO hypotheses. Based upon the above discussion, we model the probability density function of y for the H_1 and H_0 hypotheses as

$p(y|H_1) = N(1, \sigma^2 / N_1)$ and $p(y|H_0) = N(1, \sigma^2 / N_0)$. Rather than setting the threshold at t=0.5, we set the threshold at that value of y for which $p(y|H_1) = p(y|H_0)$, yielding

$$t = \frac{N_1 - \sqrt{N_1^2 - (N_1 - N_0)(N_1 + \sigma^2 \ln \frac{N_0}{N_1})}}{N_1 - N_0} \tag{31}$$

Assuming σ is a small, we omit $\sigma^2 \ln \frac{N_0}{N_1}$, obtaining

$$t = \frac{\sqrt{N_1}}{\sqrt{N_1} + \sqrt{N_0}} \tag{32}$$

From (32), the appropriate threshold is t=0.5 only if $N_0 = N_1$.

For example, in Fig. 2, only 15 of the 128 actively selected training data are UXO, and therefore N_1 = 15, N_0 = 113. If we set the threshold to be t=0.5, we detect 16% of the UXO with two false alarms. By contrast, using the procedure discussed above (for which t=0.27), we detect 88% of the UXO with 25 false alarms. The operating point corresponding to t=0.27 is indicated in Fig. 2. We similarly plot this point in all subsequent ROCs presented below.

We observe from the results in Fig. 2 that the active data selection procedure produces the best ROC results (for Pd>0.7, which is of most interest in practice), with the KMP results from the specified training area almost as good. It is observed that the average performance based on choosing the training set randomly is substantially below that of the two former approaches, with significant variability reflected in the error bars. These results demonstrate the power of the active-data-selection algorithm introduced in Sec. II, and also that the training data defined for JPG V is well matched to the testing data.

In the first example we set J=128 to be consistent with the size of the training area specified in the JPG V test. The algorithm in Sec. II can be implemented for smaller values of J, reflecting less excavation required in the training phase (for determination of

target labels). It is of interest to examine algorithm performance as J is decreased from 128. In this case training is performed using signatures and labels from the J "excavated" items, and testing is performed on the remaining 300-J examples. Results are presented for the active training procedure and for randomly choosing J training examples (100 random instantiations), as in Fig. 2. In Figs. 3-5 results are presented for J=90, 60 and 40. Using J=90 rather than J=128 results in very little degradation in ROC performance (comparing Figs. 2 and 3), with a slight performance drop for J=60, and a more substantial drop for J=40. It is interesting to note that with decreasing J, the number of test items 300-J increases, therefore increasing the number of false-alarm opportunities. This further highlights the quality of the results in Figs. 3-5, *vis-à-vis* Fig. 2. In all of these and subsequent examples, the size of the basis set $\mathbf{B}_n$ is n=10.

In the above examples J was specified to be matched to the size of a specified training set, or it was varied for comparison to such. However, the procedure in Sec. II may be employed to adaptively determine the size of the desired training set $\mathbf{X}_{s,J}$, based on the information gain as J is increased. Specifically, we track $q_n(\mathbf{X}_{s,J}) - q_n(\mathbf{X}_{s,J-1})$ for increasing J, and terminate the algorithm when the information gain is minimal. At this point, adding a new datum to the training dataset does not provide significant additional information to the classifier design.

For the JPG V data, the information gain $q_n(\mathbf{X}_{s,J}) - q_n(\mathbf{X}_{s,J-1})$ is plotted in Fig. 6 as a function of J, and the change in information gain is given in Fig. 7 for visualization assistance. Based on Fig. 6-7 the size of the training set is set to J=65. In Fig. 8 results are shown for J=65, with comparison as before to KMP results in which the J=65 training examples are selected randomly. Examining the results in Fig. 8, we observe that the active selection of training data yields a detection probability of approximately 0.95 with approximately 35 false alarms; *on average* one encounters about five times this number of false alarms to achieve the same detection probability (when selecting the training data randomly).

V. CONCLUSIONS

There are many remote-sensing problems for which one collects data from a given site, and the task is to specify the identity of the object responsible for each signature (*e.g.,* detection and classification). Due to the variability and site-dependent character of many target signatures, it is often difficult to have reliable training data *a priori* for algorithm design. In this paper we have therefore developed an information-theoretic framework in which the training data are selected adaptively from the observed site-dependent data, without requiring an *a priori* training set. Specifically, the algorithm specifies those signatures for which knowledge of the associated labels (*e.g.,* target/non-target) would be most relevant in the context of detector design. An "experiment" is then performed to learn the target labels, where in the context of landmine and UXO sensing this corresponds to excavating the respective buried items. This is a reasonable procedure, since landmines and UXO need be excavated ultimately anyway, and therefore the algorithm essentially prioritizes the order in which items are excavated, with the goal of ultimately excavating fewer non-targets (false alarms) via proper algorithm training. The algorithm has been demonstrated successfully on measured magnetometer and EMI data from an actual former bombing range, addressing the sensing of UXO.

There are several items that deserve further attention. It was demonstrated that the gain in information content is a good measure of which items should be excavated for learning of associated labels. The results in Figs. 6-8 demonstrated the effectiveness of this procedure, although the actual selection of the number of training examples, J, was determined in a somewhat *ad hoc* manner. Further work is required to make this procedure more rigorous and automated.

In addition, for the results presented here the detection algorithm was trained once using the adaptively determined training set. However, in the subsequent testing phase a "dig list" is specified for those items that are deemed to be associated with targets of interest (here UXO). Once each item is excavated, and the associated label revealed, the algorithm should be successively retrained and applied to the remaining data. The order of the dig list - and therefore the order in which we learn the labels of the

testing data - is also of interest since it may be used to further refine the algorithm sequentially, as a given site is cleaned (*e.g.,* of landmines or UXO).

Acknowledgments

This research has been supported by the US Strategic Environmental Research and Development Program (SERDP) under project UX-1281, and by the Defense Advanced Research Project Agency (DARPA) under a Multidisciplinary University Research Initiative (MURI) dedicated to Adaptive Multi-Modality Inverse Scattering.

References

[1] C.T. Schroder, W.R. Scott, and G.D. Larson, "Elastic waves interacting with buried land mines: A study using the FDTD method," *IEEE Trans. Geosc. Remote Sensing*, vol. 40, pp. 1405-1415, June 2002

[2] B. Barrow and H.H. Nelson, "Model-based characterization of electromagnetic induction signatures obtained with the MTADS electromagnetic array," *IEEE Trans. Geosc. Remote Sensing*, vol. 39, pp. 1279-1285, June 2001

[3] H.H. Nelson and J.R. McDonald, "Multisensor towed array detection system for UXO detection," *IEEE Trans. Geosc. Remote Sensing*, vol. 39, pp. 1139-1145, June 2001.

[4] R. O. Duda and P. E. Hart, *Pattern classification and scene analysis*, John Wiley, New York, 1973.

[5] V. Vapnik, *Statistical Learning Theory*, John Wiley, New York, 1998.

[6] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, 2(2), pp. 121-167, 1998.

[7] M. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," *Journal of Machine Learning Research*, vol. 1, pp. 211-244, 2001.

[8] P. Vincent and Y. Bengio, "Kernel matching pursuit," *Machine Learning*, vol. 48, pp. 165-187, 2002.

[9] B. Schölkopf, K. K. Sung, C. J. C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik, "Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers," *IEEE Trans. Signal Processing*, vol. 45, pp. 2758-2765, Nov. 1997.

[10] N. Cristianini and J. Shawe-Taylor, *Support Vector Machines and Other Kernel Based Learning Methods*, Cambridge University Press, Cambridge, UK, 2000.

[11] B. Schölkopf and A. Smola, *Learning with Kernels, Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, 2002.

[12] V. V. Fedorov, *Theory of Optimal Experiments*, Academic Press, New York, 1972.

[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley, New York, 1991.

[14] M. Stone, "Cross-validatory choice and assessment of statistical predictions", Journal of the Royal Statistical Society, Series B, 36, pp. 111-147, 1974.

[15] I. J. Won, D. A. Keiswetter, and D. R. Hanson, "GEM-3: A monostatic broadband electromagnetic induction sensor," *J. Environ. Eng. Geophys.*, vol. 2, pp. 53-64, Mar. 1997.

[16] N. Geng, C. E. Baum, and L. Carin, "On the low-frequency natural response of conducting and permeable targets," *IEEE Trans. Geosci. Remote Sensing*, vol. 37, pp. 347-359, Jan. 1999.

[17] L. Carin, H. Yu, Y. Dalichaouch, A. R. Perry, P. V. Czipott, and C. E. Baum, "On the wideband EMI response of a rotationally symmetric permeable and conducting target," *IEEE Trans. Geosci. Remote Sensing*, vol. 39, pp. 1206 -1213, June 2001.

[18] Y. Zhang, L. M. Collins, H. Yu, C. E. Baum, and L. Carin, "Sensing of unexploded ordnance with magnetometer and induction data: Theory and signal processing," *IEEE Trans. Geosci. Remote Sensing*, vol. 41, pp. 1005-1015, May 2003.

Figure Captions

Figure 1. For the first three basis functions selected, $\mathbf{b}_1$, $\mathbf{b}_2$, and $\mathbf{b}_3$, the three-dimensional vector ($K(\mathbf{x},\mathbf{b}_1)$, $K(\mathbf{x},\mathbf{b}_2)$, $K(\mathbf{x},\mathbf{b}_3)$) is plotted. Considered are feature vectors $\mathbf{x}$ for all UXO and non-UXO targets considered in this study.

Figure 2. Receiver operating characteristic (ROC) curves based on 128 training examples, for which the target labels were known. In one case the training set was carefully designed *a priori*, and in the other the training examples were chosen adaptively using the algorithm of Sec. II. For comparison, results are also shown when the 128 training examples are chosen randomly, 100 times. In the latter case average results are shown, as well as the associated range of variability. The indicated point on the ROC corresponds to (32).

Figure 3. As in Fig. 2, but now results are only shown for adaptive training-data selection (Sec. II) and for random selection. In the latter case results are presented as in Fig. 1. Results are shown for J=90 training examples.

Figure 4. As in Fig. 3, with J=60.

Figure 5. As in Fig. 4, with J= 40.

Figure 6. Information gain of adding a new datum, as a function of the number of the training examples J, selected adaptively.

Figure 7. Difference in the information gain, as a function of the number of training examples J.

Figure 8. ROC curves based on J=65 training examples, comparing the adaptive procedure (Sec. II) to random training data selection. Number of training examples chosen based on Figs. 6-7.

Figure 1

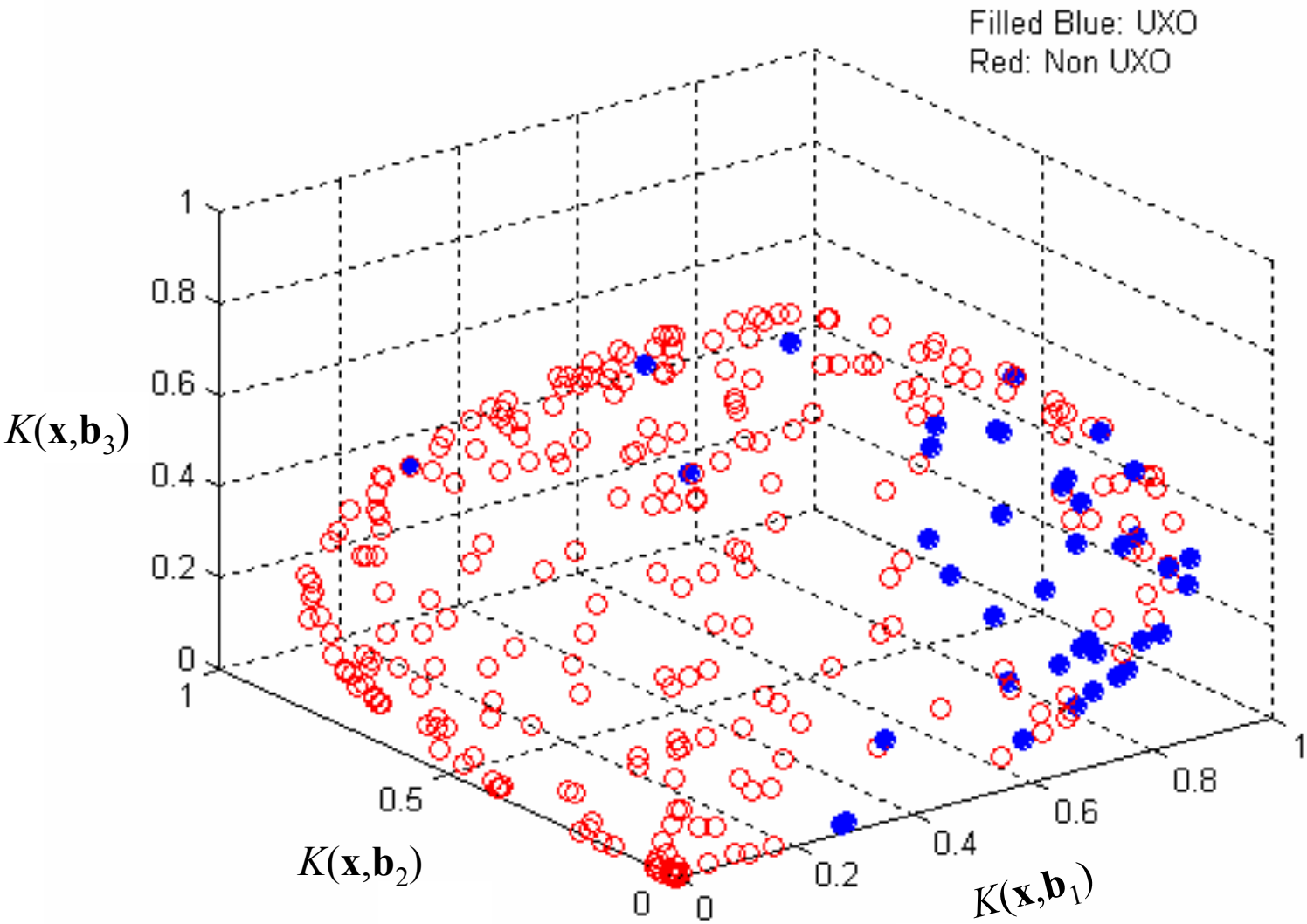

Filled Blue: UXO
Red: Non UXO
$K(\mathbf{x},\mathbf{b}_3)$
$K(\mathbf{x},\mathbf{b}_2)$
$K(\mathbf{x},\mathbf{b}_1)$
0
0.2
0.4
0.6
0.8
1
0
0.5
1
0
0.2
0.4
0.6
0.8
1

Figure 2

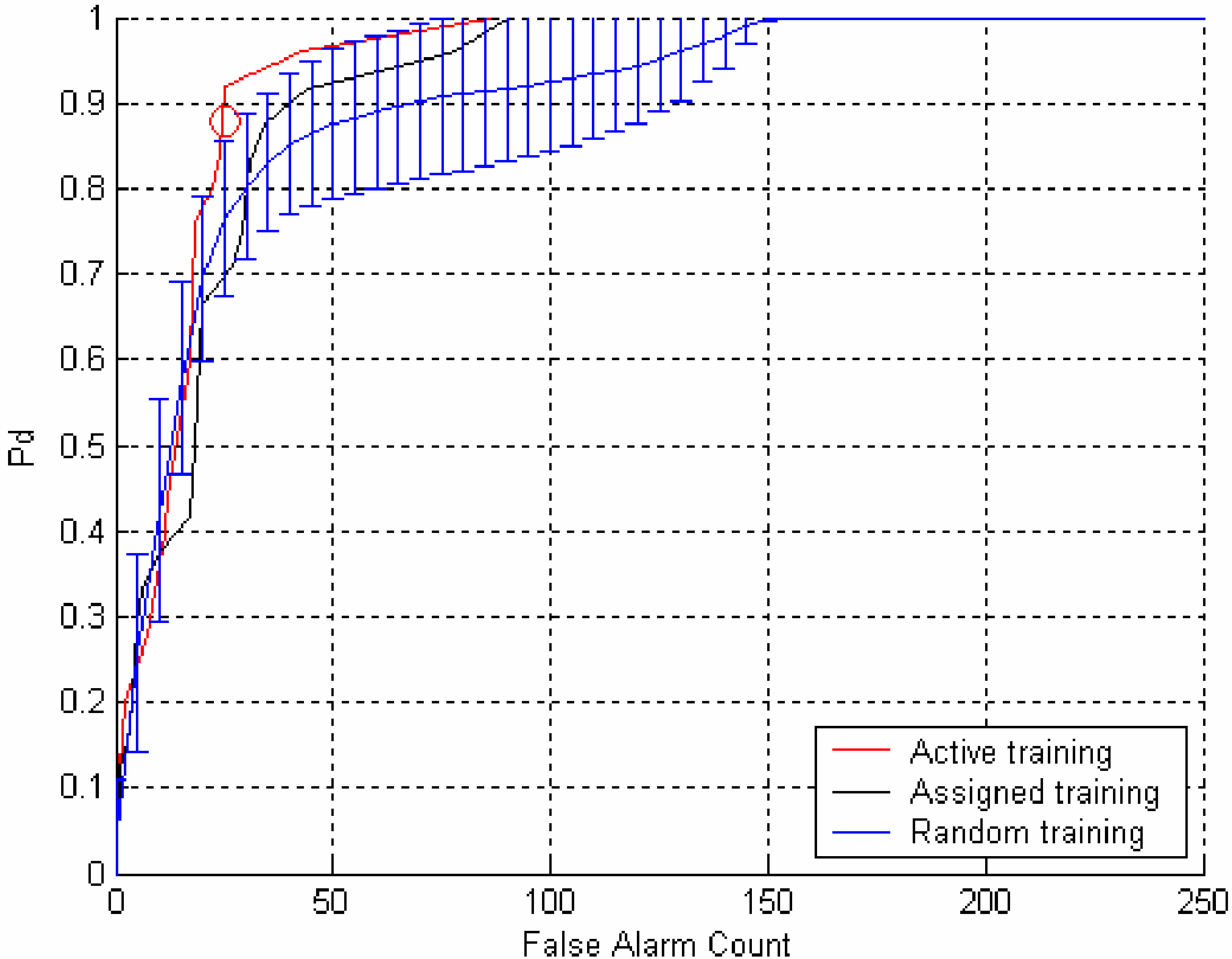

1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Pd
0
50
100
150
200
250
False Alarm Count
Active training
Assigned training
Random training

Figure 3

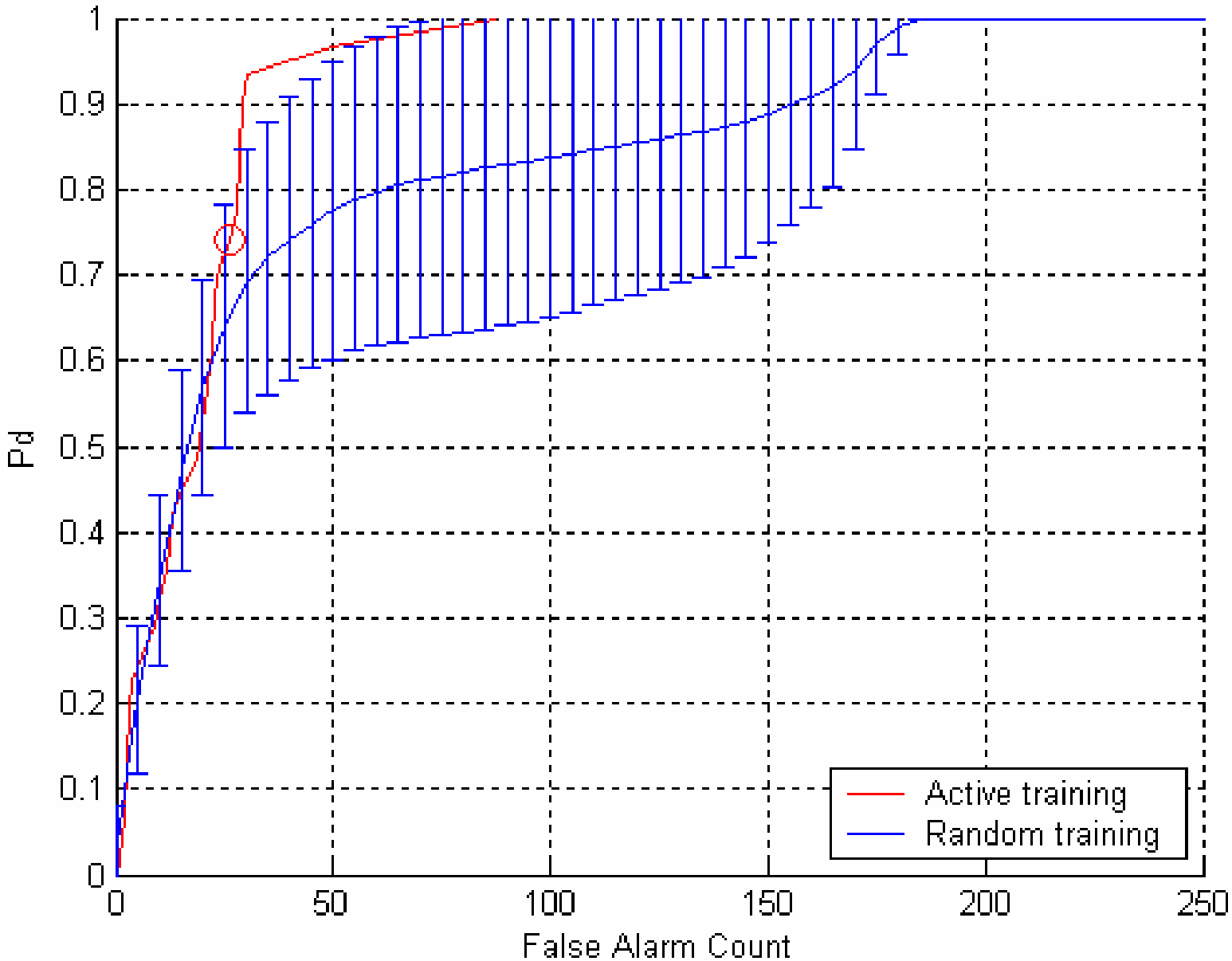

1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Pd
0
50
100
150
200
250
False Alarm Count
Active training
Random training

Figure 4

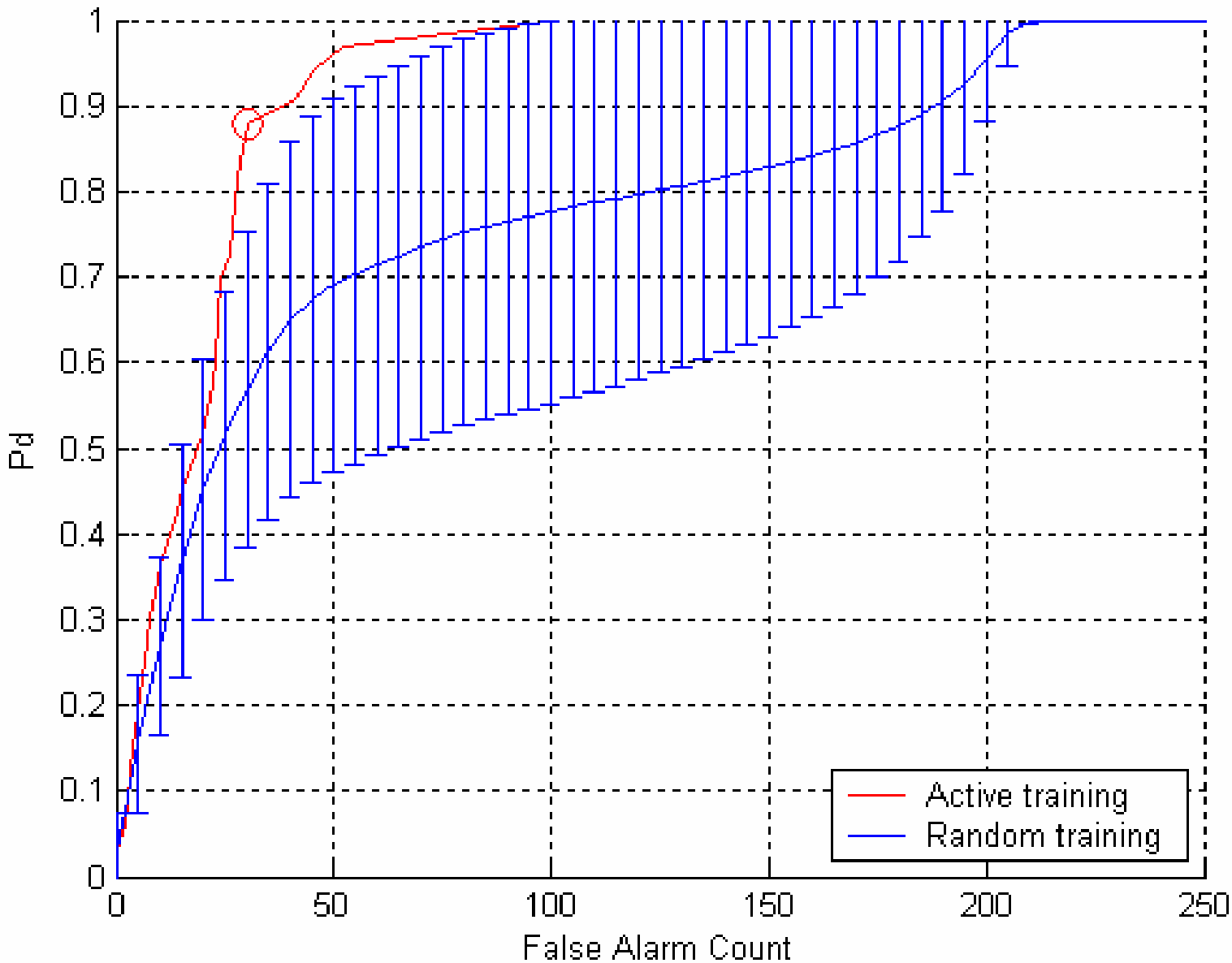

1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Pd
0
50
100
150
200
250
False Alarm Count
Active training
Random training

Figure 5

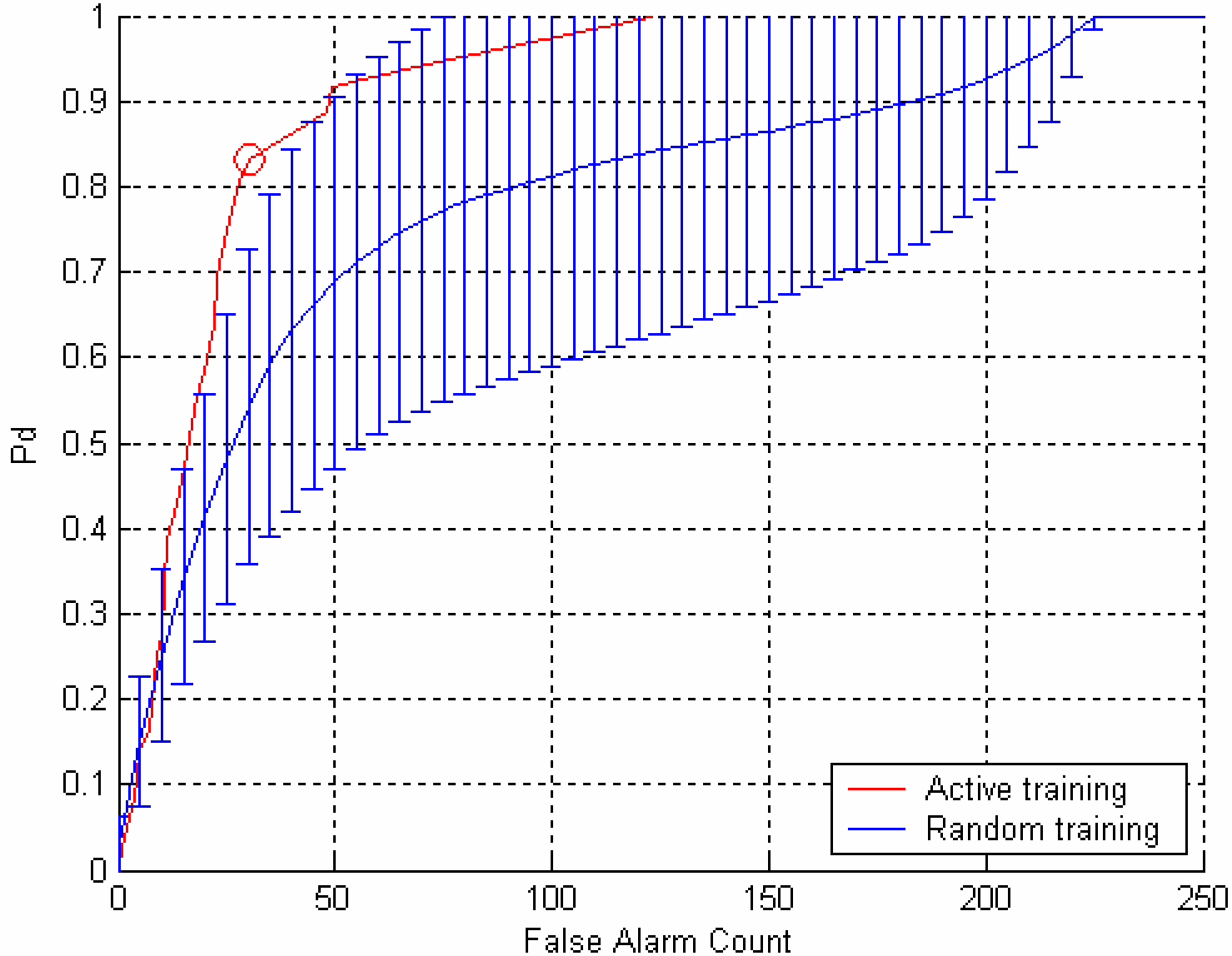

1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Pd
0
50
100
150
200
250
False Alarm Count
Active training
Random training

Figure 6

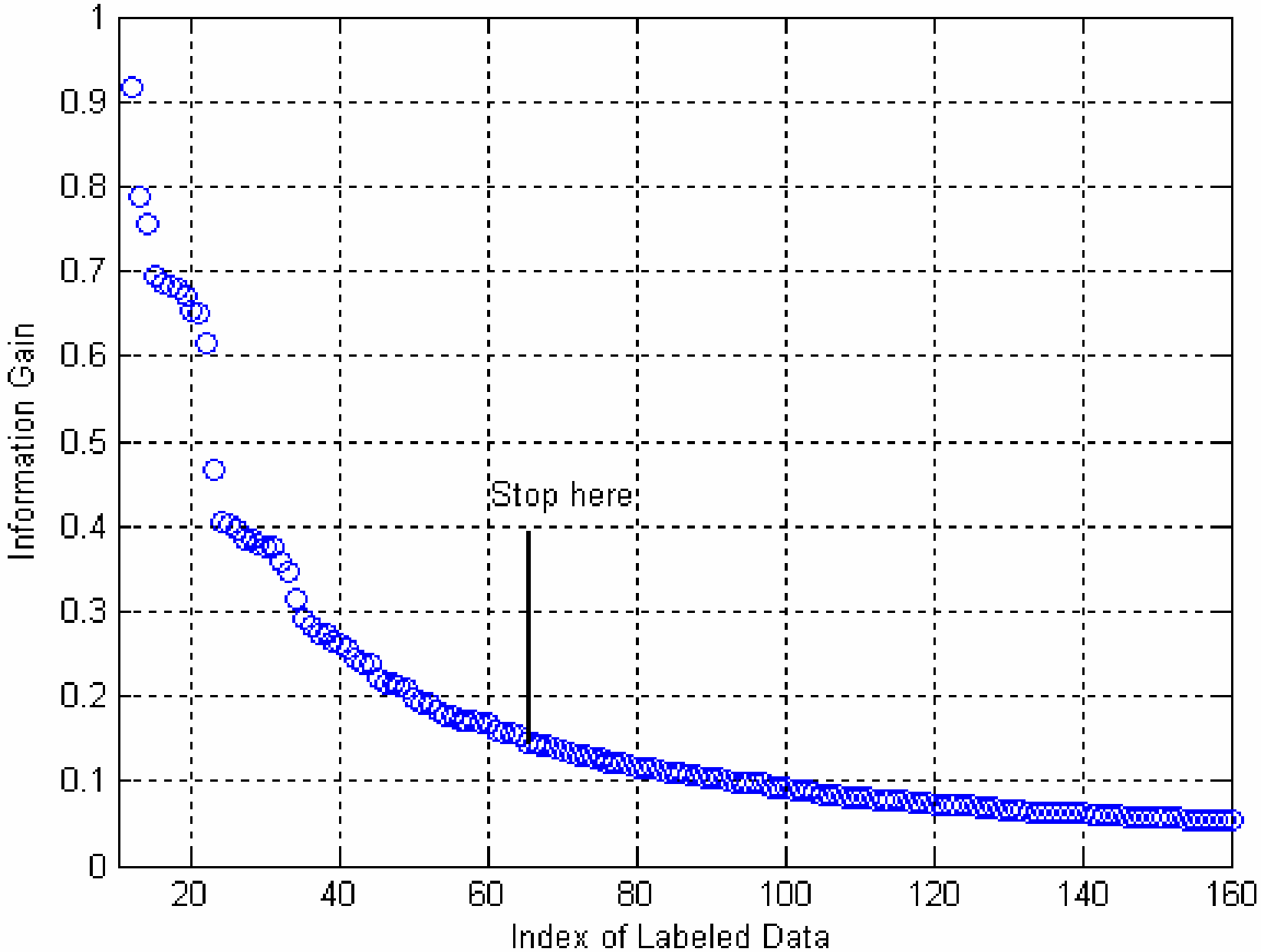

Information Gain
Index of Labeled Data
Stop here
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
20
40
60
80
100
120
140
160

Figure 7

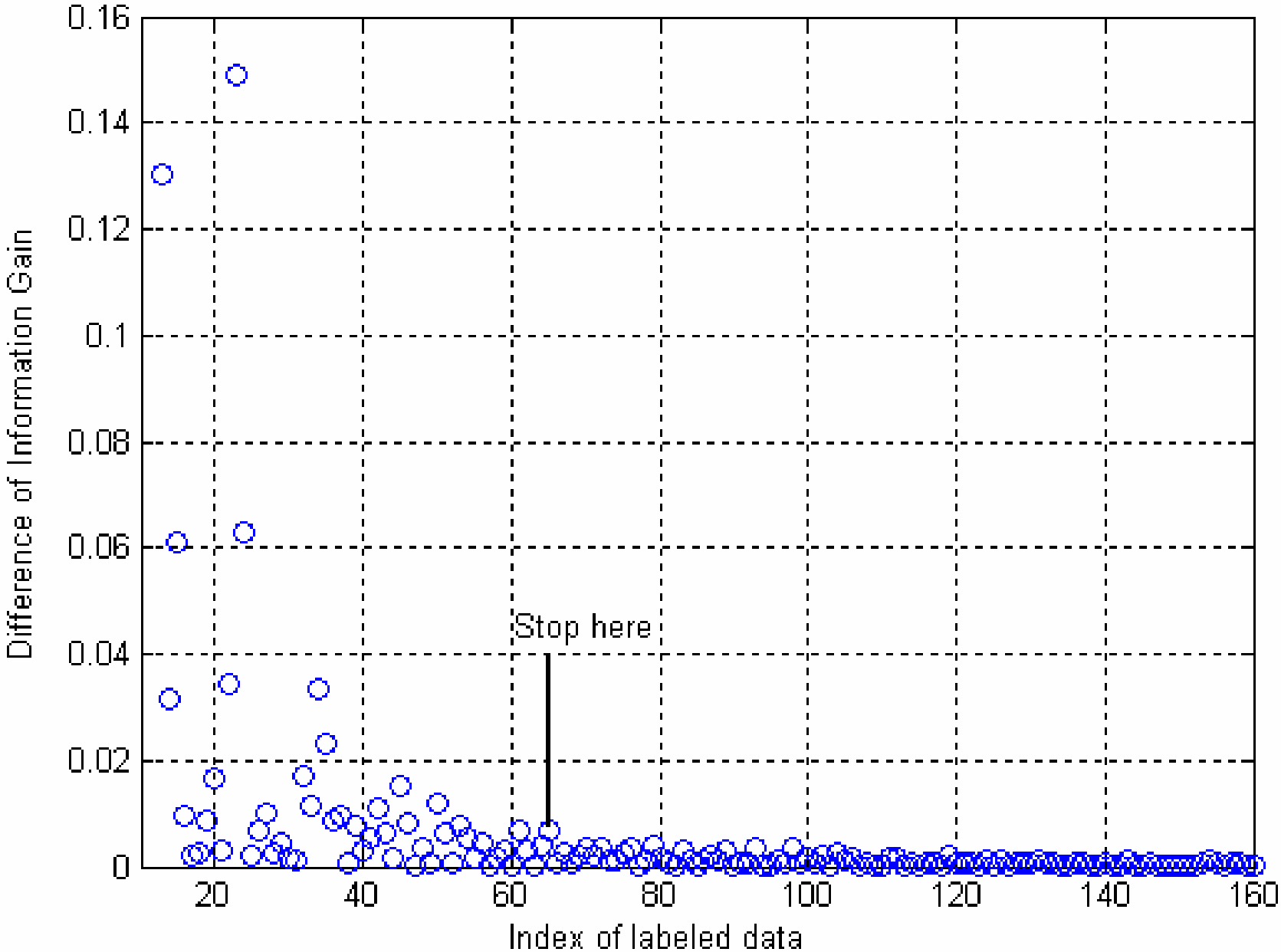
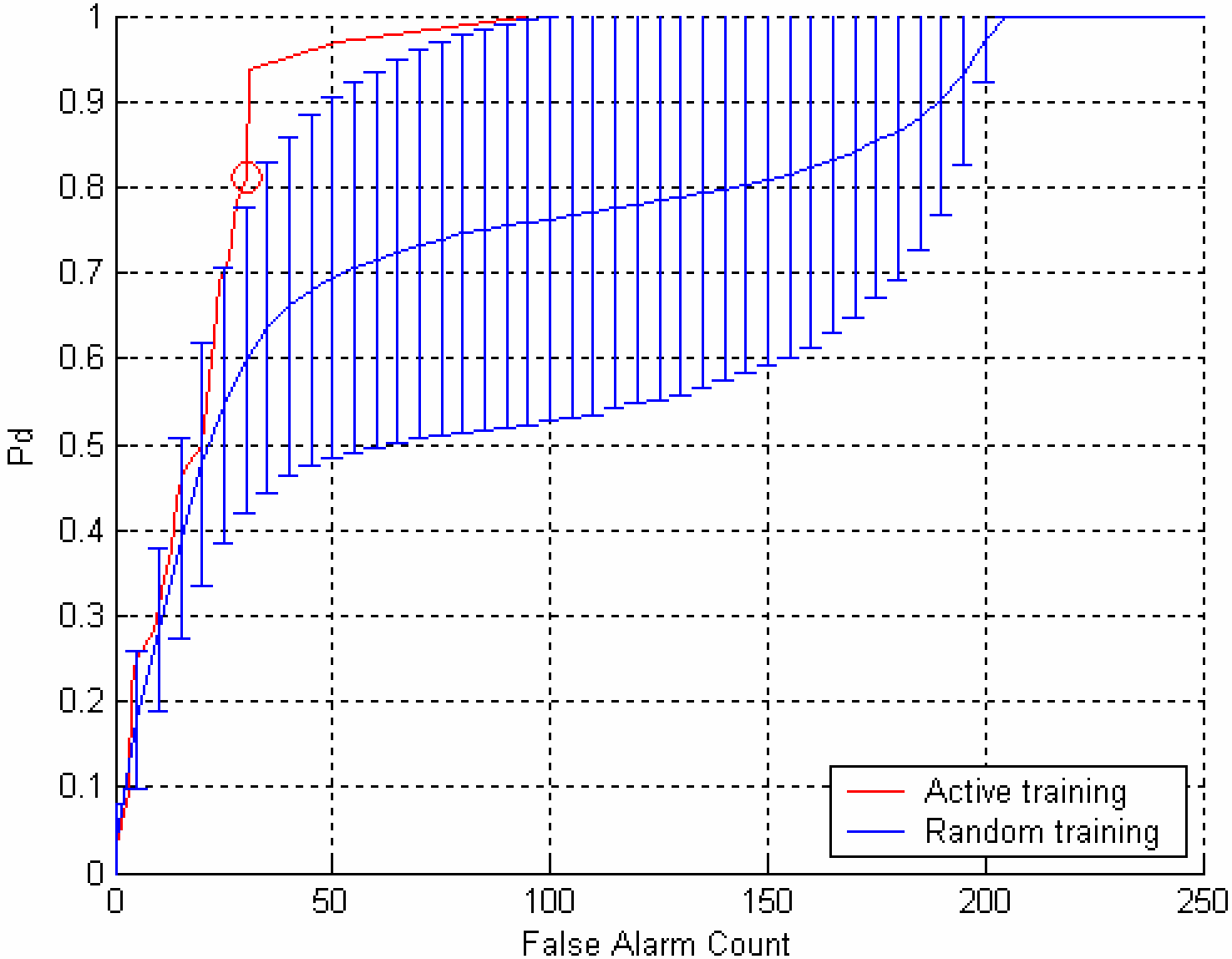

0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
Difference of Information Gain
Stop here
20
40
60
80
100
120
140
160
Index of labeled data

Figure 8


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Pd
0
50
100
150
200
250
False Alarm Count
Active training
Random training

Migratory Logistic Regression for Learning Concept Drift Between Two Data Sets

Xuejun Liao and Lawrence Carin

Department of Electrical and Computer Engineering
Duke University
Durham, NC 27708-0291, USA

{xjliao, lcarin}@ee.duke.edu

June 1, 2007

Abstract

To achieve good generalization in supervised learning, the training and testing examples are usually required to be drawn from the same source distribution. In this paper we propose a method to relax this requirement in the context of logistic regression. Assuming $\mathcal{D}^p$ and $\mathcal{D}^a$ are two sets of examples drawn from two different distributions T and A (called concepts, borrowing a term from psychology), where $\mathcal{D}^a$ are fully labeled and $\mathcal{D}^p$ partially labeled, our objective is to complete the labels of $\mathcal{D}^p$. We introduce an auxiliary variable μ for each example in $\mathcal{D}^a$ to reflect its mismatch with $\mathcal{D}^p$. Under an appropriate constraint the μ's are estimated as a byproduct, along with the classifier. We also present an active learning approach for selecting the labeled examples in $\mathcal{D}^p$. The proposed algorithm, called *migratory logistic regression* (MigLogit), is demonstrated successfully on simulated data as well as on real measured data of interest for unexploded ordnance (UXO) cleanup.

1 Introduction

In supervised classification problems, the goal is to design a classifier using the training examples (labeled data) such that the classifier predicts the labels correctly for unlabeled test data. The accuracy of the predictions is significantly affected by the quality of the training examples, which are assumed to contain essential information about the test instances for which predictions are desired. A common assumption utilized by learning algorithms is that the training examples and the test instances are drawn from the *same* source distribution.

As a practical example, consider the detection of a concealed entity based on sensor data collected in a non-invasive manner. This problem is of relevance in several practical problems,

for example in the medical imaging of potential tumors or other hidden anomalies. In the context of remote sensing, one is often challenged with the problem of detecting and characterizing a concealed (e.g., underground) target based on remotely collected sensor data. In an example that will be considered explicitly in this paper, consider the detection of buried unexploded ordnance (UXO) (Zhang et al. 2003). An unexploded ordnance is a bomb that did not explode upon impact with the ground, and such items pose great danger if disturbed (excavated) without care. Sensors used for detecting and characterizing UXO include magnetometers and electromagnetic induction (Zhang et al. 2003). In designing an algorithm for characterization of anomalies detected by such sensors, to determine if a given buried item is UXO or clutter, one typically requires training data. Such training data typically comes from other former bombing sites that have been cleaned, and there is a significant issue as to whether such extant labeled sensor data are relevant for a new site under test. The challenge addressed in this paper involves learning the relevance and relationship of existing labeled (training) data for analysis of a new unlabeled or partially labeled data set of interest. This type of problem has significant practical relevance for UXO sensing, for which results are presented on measured data, as well as for the aforementioned classes of problems, for which there is uncertainty concerning the appropriateness of existing labeled data for a new set of unlabeled data of interest.

To place this problem in a mathematical setting, let $\mathcal{T}(\mathbf{x}, y)$ be the probability distribution (or concept, borrowing a term from psychology [1]) from which test instances (each including a feature vector $\mathbf{x}$ and the associated class label y) are drawn. The goal in classifier design is to minimize a loss function $L(y, \zeta(\mathbf{x}))$, which is a quantitative measure for the loss incurred by the classifier when it predicts $\zeta(\mathbf{x})$ for $\mathbf{x}$ whose true label is y. The minimization is performed for N independent training examples $(\mathbf{x}, y)$ drawn from $\mathcal{T}(\mathbf{x}, y)$, leading to the empirical loss minimization (Vapnik 1998 1999)

$$\min_{\zeta} \frac{1}{N} \sum_{i=1}^{N} L(y_i, \zeta(\mathbf{x}_i)), \quad \text{with} \quad (\mathbf{x}, y) \sim \mathcal{T}(\mathbf{x}, y) \tag{1}$$

The empirical loss is known to approach the true loss when $N \to \infty$.

A learning algorithm based on the empirical loss minimization in (1) implicitly assumes that the future test instances are also drawn from $\mathcal{T}(\mathbf{x}, y)$. It is this assumption that assures that the classifier generalizes to test instances when it is trained to minimize empirical loss on training examples. This assumption, however, is often violated in practice, since training examples and test instances may correspond to different collections of measurements (likely performed at different times under different experimental conditions) and the class memberships of the measurements may also change. These issues can introduce statistical *differences* between the training examples and the test instances; the UXO-sensing problem discussed above constitutes an important example for which the aforementioned statistical issues hold, concerning the utility of existing labeled (training) data.

Assume that one has training examples from a distribution $\mathcal{A}(\mathbf{x}, y)$ which is different from $\mathcal{T}(\mathbf{x}, y)$. For convenience of exposition, we call $\mathcal{T}(\mathbf{x}, y)$ the primary or target distribution and

[1] Traditionally, the (probabilistic) mapping $\Pr(y|\mathbf{x})$ is called a concept, and $\Pr(\mathbf{x})$ is called a virtual concept (language describing the concept) (Widmer and Kubat 1993). For simplicity, usually they are collectively called a concept.

call $\mathcal{A}(\mathbf{x}, y)$ the auxiliary distribution. Accordingly, the examples drawn from $\mathcal{T}(\mathbf{x}, y)$ are called primary data and the examples drawn from $\mathcal{A}(\mathbf{x}, y)$ are called auxiliary data.

In order to write the empirical loss for $\mathcal{T}$ in terms of examples drawn from $\mathcal{A}$, one may employ the technique of importance sampling (Robert and Casella 1999). By doing so, one makes the following modifications to the expression of empirical loss (1)

$$\frac{1}{N}\sum_{i=1}^{N}\frac{\mathcal{T}(\mathbf{x}, y)}{\mathcal{A}(\mathbf{x}, y)}L(y_i, \zeta(\mathbf{x}_i)), \quad \text{with} \quad (\mathbf{x}, y) \sim \mathcal{A}(\mathbf{x}, y) \tag{2}$$

where $\frac{\mathcal{T}(\mathbf{x},y)}{\mathcal{A}(\mathbf{x},y)}$ is the importance weight. It is known that the modification does not change the asymptotic behavior provided that $\mathcal{A}(\mathbf{x}, y)$ has the same nonzero support as $\mathcal{T}(\mathbf{x}, y)$.

Unfortunately, both $\mathcal{A}(\mathbf{x}, y)$ and $\mathcal{T}(\mathbf{x}, y)$ are unknown to the algorithm; all that is available are samples from $\mathcal{A}(\mathbf{x}, y)$. The challenge, therefore, is to learn a classifier on training examples drawn from $\mathcal{A}(\mathbf{x}, y)$ such that the resulting classifier still generalizes to test instances drawn from $\mathcal{T}(\mathbf{x}, y)$. Clearly, without assuming any knowledge about the relationship between $\mathcal{A}(\mathbf{x}, y)$ and $\mathcal{T}(\mathbf{x}, y)$, there is little one can do but to treat the training examples as if they are from the target distribution. This will, of course, introduce errors, which may be intolerable when the difference between $\mathcal{A}(\mathbf{x}, y)$ and $\mathcal{T}(\mathbf{x}, y)$ is large.

The problem of learning on examples from one distribution with the goal of generalizing to instances from a different distribution has been addressed in different contexts, using different names. In the following, we provide a brief review of this previous work.

1.1 Tackling Concept Drifts in Time-Varying Data

Many data come naturally in streams, collected over a period of time. Such applications include weather recordings, sales and customer data, surveillance video streams, to name a few. For streamed data, it is natural to consider online learning, in which the leaner is dynamically presented with the true label after it makes a prediction and updates its hypothesis based on newly received true labels. When the streamed data are recorded over an extended period of time, the statistics in the data are likely to change. The time-dependent variation of statistics in streamed data are termed concept drift in the literature (Klinkenberg and Joachims 2000; Tenenbaum 1999; Wang et al. 2003; Widmer and Kubat 1996 1993).

Concept drift falls under the general formulation in (2). Here the target distribution characterizes the statistics of the most recent recordings, and there is an auxiliary distribution characterizing the recordings in each time interval in the past. The goal in concept-drift learning is to employ the available recordings to build up the target concept (i.e., the mapping from a feature vector to the associated class label) for the current moment. An important notion is the age of each recording, which determines the utility of the recording to the current prediction.

Three widely used methods for handling concept drift are: time windows (Widmer and Kubat 1996), instance weighting (Klinkenberg and Ruping 2003), and ensemble learning (Wang et al. 2003). In the first method, a time window is applied to the data stream and the data within the window are employed to build the classifier for the current concept. The window keeps moving towards the future so that the most recent recording is always included in the window. The window acts like a limited memory, with the data outside the window forgotten. A key issue

is to determine the window size, which should accurately capture the rate of concept drift. The method of instance weighting is based on the observation that the importance of a past example to the current concept does not change abruptly but rather decreases gradually. A weight is assigned to each past example to reflect its importance. Assuming that concept drift is monotonic (i.e., newer examples are always more important than older ones), one can construct the weights according to the age of each example; for example, one can choose weights that decrease exponentially with the age of examples (Klinkenberg and Ruping 2003). The method of ensemble learning maintains an ensemble of classifiers, instead of a single one, for the current concept. This is done by dividing the data stream into chunks and learning a classifier based on each data chunk. The relevance of each classifier to the current concept is evaluated by the generalization error when applying the classifier to the most recent data chunk. The relevance is employed as a weight applied to each classifier and the weighted classifiers are employed to make predictions for the current data chunk.

1.2 Sample Selection Bias in Econometrics

In econometrics, the observed data are often a nonrandomly selected sample of the true distribution of interest. If the distribution of interest is $\mathcal{T}$, the selection bias results in samples drawn from $\mathcal{A}$ which is different from $\mathcal{T}$. Heckman (1979) developed a method to correct the sample-selection bias for linear regression models. The basic idea of Heckman's method is that if one can estimate the probability of an observation being selected into the sample, one can use this probability estimate to correct the selection bias.

Heckman's model has recently been extended to classification problems (Zadrozny 2004), where it is assumed that the test instances are drawn from $\mathcal{T}(\mathbf{x}, y) = \Pr(\mathbf{x}, y)$ while the training examples are drawn from $\mathcal{A}(\mathbf{x}, y) = \Pr(\mathbf{x}, y|s = 1)$, where the variable s controls the selection of training examples: if $s = 1$, $(\mathbf{x}, y)$ is selected into the training set; if $s = 0$, $(\mathbf{x}, y)$ is not selected into the training set. Evidently, unless s is independent of $(\mathbf{x}, y)$, $\Pr(\mathbf{x}, y|s = 1) \neq \Pr(\mathbf{x}, y)$ and hence $\mathcal{T}(\mathbf{x}, y)$ is different from $\mathcal{T}(\mathbf{x}, y)$. By Bayes rule,

$$\frac{\Pr(\mathbf{x}, y)}{\Pr(\mathbf{x}, y|s = 1)} = \frac{\Pr(s = 1)}{\Pr(s = 1|\mathbf{x}, y)} \tag{3}$$

or

$$\frac{\mathcal{T}(\mathbf{x}, y)}{\mathcal{A}(\mathbf{x}, y)} = \frac{\Pr(s = 1)}{\Pr(s = 1|\mathbf{x}, y)} \tag{4}$$

plugging this into (2), one has

$$\frac{1}{N}\sum_{i=1}^{N} \frac{\Pr(s = 1)}{\Pr(s = 1|\mathbf{x}, y)} L(y_i, \zeta(\mathbf{x}_i)), \quad \text{with} \quad (\mathbf{x}, y) \sim \Pr(\mathbf{x}, y|s = 1) \tag{5}$$

which implies that if one has access to $\frac{\Pr(s=1)}{\Pr(s=1|\mathbf{x},y)}$ one can correct the selection bias by using the empirical loss as expressed in (5). In the special case when $\Pr(s = 1|\mathbf{x}, y) = \Pr(s = 1|\mathbf{x})$, one may estimate $\Pr(s = 1|\mathbf{x})$ from a sufficient sample of $\Pr(\mathbf{x}, s)$ if such a sample is available (Zadrozny 2004). In the general case, however, it is difficult to estimate $\frac{\Pr(s=1)}{\Pr(s=1|\mathbf{x},y)}$, as we do not have a sufficient sample of $\Pr(\mathbf{x}, y, s)$ (if we do, we already have a sufficent sample of $\Pr(\mathbf{x}, y)$, which contradicts the assumption of the problem).

1.3 Overview of This Work

In this paper we propose an efficient algorithm for solving the general problem of learning on examples from $\mathcal{A}$ with the goal of generalizing to instances from $\mathcal{T}$, when $\mathcal{A}$ is different from $\mathcal{T}$. We consider the case in which we have a fully labeled auxiliary data set $\mathcal{D}^a$ and a partially labeled primary data set $\mathcal{D}^p = \mathcal{D}^p_l \cup \mathcal{D}^p_u$, where $\mathcal{D}^p_l$ are labeled and $\mathcal{D}^p_u$ unlabeled. We assume that $\mathcal{D}^p$ are examples of the primary concept $\mathcal{T}$ (the concept we are interested in) and $\mathcal{D}^a$ are examples of the auxiliary concept $\mathcal{A}$ (the one providing indirect and low-quality information about $\mathcal{T}$). Our objective is to use a mixed training set $\mathcal{D}^{tr} = \mathcal{D}^p_l \cup \mathcal{D}^a$ to train a classifier that predicts the labels of $\mathcal{D}^p_u$ accurately, with the hope that $\mathcal{D}^p_l$ is required to have a small number of examples.

Assume $\mathcal{D}^p \sim \Pr(\mathbf{x}, y)$. In light of (3), we can write $\mathcal{D}^a \sim \Pr(\mathbf{x}, y|s = 1)$ as long as the source distributions of $\mathcal{D}^p$ and $\mathcal{D}^a$ have the same support of nonzero probability[2]. As explained previously, it is difficult to correct the mismatch by directly estimating $\frac{\Pr(s=1)}{\Pr(s=1|\mathbf{x},y)}$. Therefore we take an alternative approach. We introduce an auxiliary variable μ_i for each $(\mathbf{x}^a_i, y^a_i) \in \mathcal{D}^a$ to reflect its mismatch with $\mathcal{D}^p$ and to control its participation in the learning process. The μ's play a similar role as the weighting factors $\frac{\Pr(s=1)}{\Pr(s=1|\mathbf{x},y)}$ in (5). However, unlike the weighting factors, the auxiliary variables are estimated along with the classifier in the learning. We employ logistic regression as a specific classifier and develop our method in this context.

The remainder of the paper is organized as follows. A detailed description of the proposed method is provided in Section 2, followed by description of a fast learning algorithm in Section 3 and a theoretical discussion in Section 4. In Section 5 we present a method to actively define $\mathcal{D}^p_l$ when $\mathcal{D}^p_l$ is initially empty. In Section 6 we demonstrate the ideas presented here using simulated data, as well as real data of interest for detecting unexploded ordnance (UXO). Finally, Section 7 provides conclusions.

2 Migratory Logistic Regression (MigLogit): Learning Jointly on the Primary and Auxiliary Data

We assume $\mathcal{D}^p_l$ are fixed and nonempty, and without loss of generality, we assume $\mathcal{D}^p_l$ are always indexed prior to $\mathcal{D}^p_u$, i.e., $\mathcal{D}^p_l = \{(\mathbf{x}^p_i, y^p_i)\}_{i=1}^{N^p_l}$ and $\mathcal{D}^p_u = \{(\mathbf{x}^p_i, y^p_i) : y^p_i \text{ missing}\}_{i=N^p_l+1}^{N^p}$. We use N^a, N^p, and N^p_l to denote the size (number of data points) in $\mathcal{D}^a$, $\mathcal{D}^p$, and $\mathcal{D}^p_l$, respectively. In Section 5 we discuss how to actively determine $\mathcal{D}^p_l$ when $\mathcal{D}^p_l$ is initially empty. We consider the binary classification problem and the labels $y^a, y^p \in \{-1, 1\}$. For notational simplicity, we let $\mathbf{x}$ always include a 1 as its first element to accommodate a bias (intercept) term, thus $\mathbf{x}^p, \mathbf{x}^a \in \mathbb{R}^{d+1}$ where d is the number of features. For a primary data point $(\mathbf{x}^p_i, y^p_i) \in \mathcal{D}^p_l$, we follow standard logistic regression to write

$$\Pr(y^p_i|\mathbf{x}^p_i; \mathbf{w}) = \sigma(y^p_i \mathbf{w}^T \mathbf{x}^p_i) \tag{6}$$

[2]For any $\Pr(\mathbf{x}, y|s = 1) \neq 0$ and $\Pr(\mathbf{x}, y) \neq 0$, there exists $\frac{\Pr(s=1)}{\Pr(s=1|\mathbf{x},y)} = \frac{\Pr(\mathbf{x},y)}{\Pr(\mathbf{x},y|s=1)} \in (0, \infty)$ such that (3) is satisfied. For $\Pr(\mathbf{x}, y|s = 1) = \Pr(\mathbf{x}, y) = 0$, any $\frac{\Pr(s=1)}{\Pr(s=1|\mathbf{x},y)} \neq 0$ makes (3) satisfied.

where $\mathbf{w} \in \mathbb{R}^{d+1}$ is a column vector of classifier parameters and $\sigma(\eta) = \frac{1}{1+\exp(-\eta)}$ is the sigmoid function. For a auxiliary data point $(\mathbf{x}_i^a, y_i^a) \in \mathcal{D}^a$, we define

$$\Pr(y_i^a|\mathbf{x}_i^a; \mathbf{w}, \mu_i) = \sigma(y_i^a \mathbf{w}^T \mathbf{x}_i^a + y_i^a \mu_i) \tag{7}$$

where μ_i is an auxiliary variable. Assuming the examples in $\mathcal{D}_l^p$ and $\mathcal{D}^a$ are drawn i.i.d., we have the log-likelihood function

$$\begin{aligned} &\ell(\mathbf{w}, \boldsymbol{\mu}; \mathcal{D}_l^p \cup \mathcal{D}^a) \\ &= \textstyle\sum_{i=1}^{N_l^p} \ln \sigma(y_i^p \mathbf{w}^T \mathbf{x}_i^p) + \sum_{i=1}^{N^a} \ln \sigma(y_i^a \mathbf{w}^T \mathbf{x}_i^a + y_i^a \mu_i) \end{aligned} \tag{8}$$

where $\boldsymbol{\mu} = [\mu_1, \cdots, \mu_{N^a}]^T$ is a column vector of all auxiliary variables.

The auxiliary variable μ_i is introduced to reflect the mismatch of $(\mathbf{x}_i^a, y_i^a)$ with $\mathcal{D}^p$ and to control its participation in the learning of $\mathbf{w}$. A larger $y_i^a \mu_i$ makes $\Pr(y_i^a|\mathbf{x}_i^a; \mathbf{w}, \mu_i)$ less sensitive to $\mathbf{w}$. When $y_i^a \mu_i = \infty$, $\Pr(y_i^a|\mathbf{x}_i^a; \mathbf{w}, \mu_i) = 1$ becomes completely independent of $\mathbf{w}$. Geometrically, the μ_i is an extra intercept term that is uniquely associated with $\mathbf{x}_i^a$ and causes it to migrate towards class y_i^a. If $(\mathbf{x}_i^a, y_i^a)$ is mismatched with the primary data $\mathcal{D}^p$, $\mathbf{w}$ cannot make $\sum_{i=1}^{N_l^p} \ln \sigma(y_i^p \mathbf{w}^T \mathbf{x}_i^p)$ and $\ln \sigma(y_i^a \mathbf{w}^T \mathbf{x}_i^a)$ large at the same time. In this case $\mathbf{x}_i^a$ will be given an appropriate μ_i to allow it to migrate towards class y_i^a, so that $\mathbf{w}$ is less sensitive to $(\mathbf{x}_i^a, y_i^a)$ and can focus more on fitting $\mathcal{D}_l^p$. Evidently, if the μ's are allowed to change freely, their influence will override that of $\mathbf{w}$ in fitting the auxiliary data $\mathcal{D}^a$ and then $\mathcal{D}^a$ will not participate in learning $\mathbf{w}$. To prevent this from happening, we introduce constraints on μ_i and maximize the log-likelihood subject to the constraints:

$$\max_{\mathbf{w}, \boldsymbol{\mu}} \quad \ell(\mathbf{w}, \boldsymbol{\mu}; \mathcal{D}_l^p \cup \mathcal{D}^a) \tag{9}$$

$$\text{subject to} \quad \frac{1}{N^a} \textstyle\sum_{i=1}^{N^a} y_i^a \mu_i \leq C, \quad C \geq 0 \tag{10}$$

$$y_i^a \mu_i \geq 0, \quad i = 1, 2, \cdots, N^a \tag{11}$$

where the inequalities in (11) reflect the fact that in order for $\mathbf{x}_i^a$ to fit $y_i^a = 1$ (or $y_i^a = -1$) we need to have $\mu_i > 0$ (or $\mu_i < 0$), if we want μ_i to exert a *positive* influence in the fitting process. Under the constraints in (11), a larger value of $y_i^a \mu_i$ represents a larger mismatch between $(\mathbf{x}_i^a, y_i^a)$ and $\mathcal{D}^p$ and accordingly makes $(\mathbf{x}_i^a, y_i^a)$ play a less important role in determining $\mathbf{w}$. The classifier resulting from solving the problem in (9)-(11) is referred to as *migratory logistic regression* (MigLogit).

The C in (10) reflects the average mismatch between $\mathcal{D}^a$ and $\mathcal{D}^p$ and controls the average participation of $\mathcal{D}^a$ in determining $\mathbf{w}$. It can be learned from data if we have a reasonable amount of $\mathcal{D}_l^p$. However, in practice we usually have no or very scarce $\mathcal{D}_l^p$ to begin with. In this case, we must rely on other information to set C. We will come back to a more detailed discussion on C in Section 4.

3 Fast Learning Algorithm

The optimization problem in (9), (10), and (11) is concave and any standard technique can be utilized to find the global maxima. However, there is a unique μ_i associated with every

$(\mathbf{x}_i^a, y_i^a) \in \mathcal{D}^a$, and when $\mathcal{D}^a$ is large using a standard method to estimate μ's can consume most of the computational time.

In this section, we give a fast algorithm for training MigLogit, by taking a block-coordinate ascent approach (Bertsekas 1999), in which we alternately solve for $\mathbf{w}$ and $\boldsymbol{\mu}$, keeping one fixed when solving for the other. The algorithm draws its efficiency from the analytic solution of $\boldsymbol{\mu}$, which we establish in the following theorem. Proof of the theorem is given in the appendix, and Section 4 contains a discussion that helps to understand the theorem from an intuitive perspective.

Theorem 1: *Let $f(z)$ be a twice continuously differentiable function and its second derivative $f''(z) < 0$ for any $z \in \mathbb{R}$. Let $b_1 \leq b_2 \leq \cdots \leq b_N$, $R \geq 0$, and*

$$n = \max\{m : mb_m - \textstyle\sum_{i=1}^m b_i \leq R, 1 \leq m \leq N\} \tag{12}$$

Then the problem

$$\max_{\{z_i\}} \quad \textstyle\sum_{i=1}^N f(b_i + z_i) \tag{13}$$

$$\textit{subject to} \quad \textstyle\sum_{i=1}^N z_i \leq R, \quad R \geq 0 \tag{14}$$

$$z_i \geq 0, \quad i = 1, 2, \cdots, N \tag{15}$$

has a unique global solution

$$z_i = \begin{cases} \frac{1}{n}\sum_{j=1}^n b_j + \frac{1}{n}R - b_i, & 1 \leq i \leq n \\ 0, & n < i \leq N \end{cases} \tag{16}$$

For a fixed $\mathbf{w}$, the problem in (9)-(11) is simplified to maximizing $\sum_{i=1}^{N^a} \ln \sigma(y_i^a \mathbf{w}^T \mathbf{x}_i^a + y_i^a \mu_i)$ with respect to $\boldsymbol{\mu}$, subject to $\frac{1}{N^a}\sum_{i=1}^{N^a} y_i^a \mu_i \leq C$, $C \geq 0$, and $y_i^a \mu_i \geq 0$ for $i = 1, 2, \cdots, N^a$. Clearly $\ln \sigma(z)$ is a twice continuously differentiable function of z and its second derivative $\frac{\partial^2}{\partial z^2} \ln \sigma(z) = -\sigma(z)\sigma(-z) < 0$ for $-\infty < z < \infty$. Thus Theorem 1 applies. We first solve $\{y_i^a \mu_i\}$ using Theorem 1, then $\{\mu_i\}$ are trivially solved using the fact $y_i^a \in \{-1, 1\}$. Assume $y_{k_1}^a \mathbf{w}^T \mathbf{x}_{k_1}^a \leq y_{k_2}^a \mathbf{w}^T \mathbf{x}_{k_2}^a \leq \cdots \leq y_{k_{N^a}}^a \mathbf{w}^T \mathbf{x}_{k_{N^a}}^a$, where $k_1, k_2, \cdots, k_{N^a}$ is a permutation of $1, 2, \cdots, N^a$. Then we can write the solution of $\{\mu_i\}$ analytically,

$$\mu_{k_i} = \begin{cases} \frac{1}{n} y_{k_i}^a \sum_{j=1}^n y_{k_j}^a \mathbf{w}^T \mathbf{x}_{k_j}^a \\ \quad + \frac{N^a}{n} y_{k_i}^a C - \mathbf{w}^T \mathbf{x}_{k_i}^a \end{cases}, \quad 1 \leq i \leq n \\ \quad 0, \qquad n < i \leq N^a \tag{17}$$

where

$$n = \max\Big\{m : m y_{k_m}^a \mathbf{w}^T \mathbf{x}_{k_m}^a - \textstyle\sum_{i=1}^m y_{k_i}^a \mathbf{w}^T \mathbf{x}_{k_i}^a \leq N^a C, \\ 1 \leq m \leq N^a\Big\} \tag{18}$$

For a fixed $\boldsymbol{\mu}$, we use the standard gradient-based method (Bertsekas 1999) to find $\mathbf{w}$. The main procedures of the fast training algorithm for MigLogit are summarized in Table 1, where the gradient $\nabla_{\mathbf{w}} \ell$ and the Hessian matrix $\nabla_{\mathbf{w}}^2 \ell$ are computed from (8).

Table 1: Fast Learning Algorithm of Migratory Logistic Regression (MigLogit)

Input: $\mathcal{D}^a \cup \mathcal{D}_l^p$ and C; Output: $\mathbf{w}$ and $\{\mu_i\}_{i=1}^{N^a}$

1. Initialize $\mathbf{w}$ and $\mu_i = 0$ for $i = 1, 2, \cdots, N^a$.
2. Compute the gradient $\nabla_{\mathbf{w}}\ell$ and Hessian matrix $\nabla^2_{\mathbf{w}}\ell$.
3. Compute the ascent direction $\mathbf{d} = -(\nabla^2_{\mathbf{w}}\ell)^{-1}\nabla_{\mathbf{w}}\ell$.
4. Do a linear search for the step-size $\alpha^* = \arg\max_\alpha \ell(\mathbf{w} + \alpha\mathbf{d})$.
5. Update $\mathbf{w}$: $\mathbf{w} \leftarrow \mathbf{w} + \alpha^*\mathbf{d}$.
6. Sort $\{y_i^a \mathbf{w}^T\mathbf{x}_i^a\}_{i=1}^{N^a}$ in ascending order. Assume the result is $y_{k_1}^a \mathbf{w}^T\mathbf{x}_{k_1}^a \leq y_{k_2}^a \mathbf{w}^T\mathbf{x}_{k_2}^a \leq \cdots \leq y_{k_{N^a}}^a \mathbf{w}^T\mathbf{x}_{k_{N^a}}^a$, where $k_1, k_2, \cdots, k_{N^a}$ is a permutation of $1, 2, \cdots, N^a$.
7. Find the n using (18).
8. Update the auxiliary variables $\{\mu_i\}_{i=1}^{N^a}$ using (17).
9. Check the convergence of ℓ: exit and output $\mathbf{w}$ and $\{\mu_i\}_{i=1}^{N^a}$ if converged; go back to 2 otherwise.

4 Auxiliary Variables and Choice of C

Theorem 1 and its constructive proof in the appendix offers some insight into the mechanism of how the mismatch between $\mathcal{D}^a$ and $\mathcal{D}^p$ is compensated through the auxiliary variables $\{\mu_i\}$. To make the description easier, we think of each data point $\mathbf{x}_i^a \in \mathcal{D}^a$ as getting principal importance $y_i^a \mathbf{w}^T\mathbf{x}_i^a$ from $\mathbf{w}$ and additional importance $y_i^a\mu_i$ from a given budget totaling N^aC (C represents the average budget for a single $\mathbf{x}^a$). From the appendix, N^aC is distributed among the auxiliary data $\{\mathbf{x}_i^a\}$ by a "smallest-first" rule: the smallest $\mathbf{x}_{k_1}^a$ (that which has the smallest $y_{k_1}^a \mathbf{w}^T\mathbf{x}_{k_1}^a$), gets a portion $y_{k_1}^a\mu_{k_1}$ from N^aC first, and when the total importance $y_{k_1}^a \mathbf{w}^T\mathbf{x}_{k_1}^a + y_{k_1}^a\mu_{k_1}$ reaches the value of the second smallest $\mathbf{x}_{k_2}^a$, N^aC becomes equally distributed to $\mathbf{x}_{k_1}^a$ and $\mathbf{x}_{k_2}^a$ such that their total importances are always equal. Then, when $y_{k_1}^a \mathbf{w}^T\mathbf{x}_{k_1}^a + y_{k_1}^a\mu_{k_1} = y_{k_2}^a \mathbf{w}^T\mathbf{x}_{k_2}^a + y_{k_2}^a\mu_{k_2}$ reach the importance of the third smallest, N^aC becomes equally distributed to three of them to make them equal. The distribution continues in this way until the budget N^aC is used up. The "smallest-first" rule is essentially a result of the concavity of the logarithmic sigmoid function $\ln\sigma(\cdot)$. The goal is to maximize $\sum_{i=1}^{N^a} \ln\sigma(y_i^a \mathbf{w}^T\mathbf{x}_i^a + y_i^a\mu_i)$. The concavity of $\ln\sigma(\cdot)$ dictates that for any given portion of N^aC, distributing it to the smallest makes the maximum gain in $\ln\sigma$.

The C is used as a means to compensate for the loss that $\mathcal{D}^a$ may suffer from $\mathbf{w}$. The classifier $\mathbf{w}$ is responsible for correctly classifying both $\mathcal{D}^a$ and $\mathcal{D}^p$. Because $\mathcal{D}^a$ and $\mathcal{D}^p$ are mismatched, $\mathbf{w}$ cannot satisfy both of them: one must suffer if the other is to gain. As $\mathcal{D}^p$ is the primary data set, we want $\mathbf{w}$ to classify $\mathcal{D}^p$ as accurately as possible. The auxiliary variables are therefore introduced to represent compensations that $\mathcal{D}^a$ get from C. When $\mathbf{x}^a$ gets small contribution from $\mathbf{w}$ and is small, it is because $\mathbf{x}^a$ is mismatched and in conflict with $\mathcal{D}^p$ (assuming perfect separation of $\mathcal{D}^a$, no conflict exists among themselves). By the "smallest first" rule, the most mismatched $\mathbf{x}^a$ gets compensation first.

A high compensation $y_i^a\mu_i$ whittles down the participation of $\mathbf{x}_i^a$ in learning $\mathbf{w}$. This is readily seen from the contribution of $(\mathbf{x}_i^a, y_i^a)$ to $\nabla_{\mathbf{w}}\ell$ and $\nabla^2_{\mathbf{w}}\ell$, which are obtained from (8) as $\sigma(-y_i^a \mathbf{w}^T\mathbf{x}_i^a - y_i^a\mu_i)y_i^a\mathbf{x}_i^a$ and $-\sigma(-y_i^a \mathbf{w}^T\mathbf{x}_i^a - y_i^a\mu_i)\sigma(y_i^a \mathbf{w}^T\mathbf{x}_i^a + y_i^a\mu_i)\mathbf{x}_i^a\mathbf{x}_i^{a\,T}$, respectively. When

$y_i^a \mu_i$ is large, $\sigma(-y_i^a \mathbf{w}^T \mathbf{x}_i^a - y_i^a \mu_i)$ is close to zero and hence the contribution of $(\mathbf{x}_i^a, y_i^a)$ to $\nabla_{\mathbf{w}} \ell$ and $\nabla_{\mathbf{w}}^2 \ell$ are ignorable. We in fact do not need an infinitely large $y_i^a \mu_i$ to make the contributions of $\mathbf{x}_i^a$ ignorable, because $\sigma(\mu)$ is almost saturated at $\mu = \pm 6$. If $y_i^a \mathbf{w}^T \mathbf{x}_i^a = -6$, $\sigma(-y_i^a \mathbf{w}^T \mathbf{x}_i^a) = 0.9975$, implying a large contribution of $(\mathbf{x}_i^a, y_i^a)$ to $\nabla_{\mathbf{w}} \ell$, which happens when $\mathbf{w}$ assigns $\mathbf{x}_i^a$ to the correct class y_i^a with probability of $\sigma(y_i^a \mathbf{w}^T \mathbf{x}_i^a) = \sigma(-6) = 0.0025$ only. In this nearly worst case, a compensation of $y_i^a \mu_i = 12$ can effectively remove the contribution of $(\mathbf{x}_i^a, y_i^a)$ because $\sigma(-y_i^a \mathbf{w}^T \mathbf{x}_i^a - y_i^a \mu_i) = \sigma(6-12) = \sigma(-6) = 0.0025$. To effectively remove the contributions of N^m auxiliary data, one needs a total budge $12N^m$, resulting in an average budget $C = 12N^m/N^a$.

To make a right choice of C, the N^m/N^a should represent the rate that $\mathcal{D}^a$ are mismatched with $\mathcal{D}^p$. This is because we want $N^a C$ to be distributed only to that part of $\mathcal{D}^a$ that is mismatched with $\mathcal{D}^p$, thus permitting us to use the remaining part in learning $\mathbf{w}$. The quantity N^m/N^a is usually unknown in practice. However, $C = 12N^m/N^a$ gives one a sense of at least what range C should be in. As $0 \leq N^m \leq N^a$, letting $0 \leq C \leq 12$ is usually a reasonable choice. In our experiences, the performance of MigLogit is relatively robust to C, as demonstrated in Section 6.2.

5 Active Selection of $\mathcal{D}_l^p$

In Section 2 we assumed that $\mathcal{D}_l^p$ had already been determined. In this section we describe how $\mathcal{D}_l^p$ can be actively selected from $\mathcal{D}^p$, based on the Fisher information matrix (Fedorov 1972; MacKay 1992). The approach is known as active learning (Cohn et al. 1995; Krogh and Vedelsby 1995).

Let $\mathbf{Q}$ denote the Fisher information matrix of $\mathcal{D}_l^p \cup \mathcal{D}^a$ about $\mathbf{w}$. By definition of the Fisher information matrix (Cover and Thomas 1991), $\mathbf{Q} = \mathbb{E}_{\{y_i^p\},\{y_i^a\}} \frac{\partial \ell}{\partial \mathbf{w}} \frac{\partial \ell}{\partial \mathbf{w}}^T$, and substituting (8) into this equation gives (a brief derivation is given in the appendix)

$$\mathbf{Q} = \textstyle\sum_{i=1}^{N_l^p} \sigma_i^p (1-\sigma_i^p) \mathbf{x}_i^p \mathbf{x}_i^{p\,T} + \sum_{i=1}^{N^a} \sigma_i^a (1-\sigma_i^a) \mathbf{x}_i^a \mathbf{x}_i^{a\,T} \tag{19}$$

where $\sigma_i^p = \sigma(\mathbf{w}^T \mathbf{x}_i^p)$ for $i = 1, 2, \ldots, N_l^p$, and $\sigma_i^a = \sigma(\mathbf{w}^T \mathbf{x}_i^a + \mu_i)$ for $i = 1, 2, \ldots, N^a$, and $\mathbf{w}$ and $\{\mu_i\}$ represent the true classifier and auxiliary variables.

It is well known the inverse Fisher information $\mathbf{Q}^{-1}$ lower bounds the covariance matrix of the estimated $\mathbf{w}$ (Cover and Thomas 1991). In particular, $[\det(\mathbf{Q})]^{-1}$ lower bounds the product of variances of the elements in $\mathbf{w}$. The goal in selecting $\mathcal{D}_l^p$ is to reduce the variances, or uncertainty, of $\mathbf{w}$. Thus we seek the $\mathcal{D}_l^p$ that maximize $\det(\mathbf{Q})$.

The selection proceeds in a sequential manner. Initially $\mathcal{D}_u^p = \mathcal{D}^p$, $\mathcal{D}_l^p$ is empty, and $\mathbf{Q} = \sum_{i=1}^{N^a} \sigma_i^a (1-\sigma_i^a) \mathbf{x}_i^a \mathbf{x}_i^{a\,T}$. Then one at a time, a data point $\mathbf{x}_i^p \in \mathcal{D}_u^p$ is selected and moved from $\mathcal{D}_u^p$ to $\mathcal{D}_l^p$. This causes $\mathbf{Q}$ to be updated as: $\mathbf{Q} \leftarrow \mathbf{Q} + \sigma_i^p (1-\sigma_i^p) \mathbf{x}_i^p (\mathbf{x}_i^p)^T$. At each iteration, the selection is based on

$$\begin{aligned} &\max_{\mathbf{x}_i^p \in \mathcal{D}_u^p} \det \left\{ \mathbf{Q} + \sigma_i^p (1-\sigma_i^p) \mathbf{x}_i^p (\mathbf{x}_i^p)^T \right\} \\ &= \max_{\mathbf{x}_i^p \in \mathcal{D}_u^p} \left\{ 1 + \sigma_i^p (1-\sigma_i^p) (\mathbf{x}_i^p)^T \mathbf{Q}^{-1} \mathbf{x}_i^p \right\} \end{aligned} \tag{20}$$

where we assume the existence of $\mathbf{Q}^{-1}$, which can often be assured by using sufficient auxiliary data $\mathcal{D}^a$.

Evaluation of (20) requires the true values of $\mathbf{w}$ and $\{\mu_i\}$, which are not known *a priori*. We follow (Fedorov 1972) and replace them with the $\mathbf{w}$ and $\{\mu_i\}$ that are estimated from $\mathcal{D}^a \cup \mathcal{D}_l^p$, where $\mathcal{D}_l^p$ are the primary labeled data selected up to the present.

6 Results

In this section the performance of MigLogit is demonstrated and compared to the standard logistic regression. The MigLogit is trained using $\mathcal{D}^a \cup \mathcal{D}_l^p$, where $\mathcal{D}_l^p$ are either randomly selected from $\mathcal{D}^p$, or actively selected from $\mathcal{D}^p$ using the method in Section 5. When $\mathcal{D}_l^p$ are randomly selected, 50 independent trials are performed and the results are obtained as an average over the trials. Three logistic regression classifiers are trained using different combinations of $\mathcal{D}^a$ and $\mathcal{D}_l^p$: $\mathcal{D}^a \cup \mathcal{D}_l^p$, $\mathcal{D}_l^p$ alone, and $\mathcal{D}^a$ alone, where $\mathcal{D}_l^p$ are identical to the $\mathcal{D}_l^p$ used by MigLogit. The four classifiers are tested on $\mathcal{D}_u^p = \mathcal{D}^p \setminus \mathcal{D}_l^p$ to produce the test-error rate or the area under the ROC curve. Calculation of test error rates is based on the following decision rule: declare $y^p = -1$ if $\sigma(\mathbf{w}^T\mathbf{x}^p) \leq 0.5$ and $y^p = 1$ otherwise, for any $\mathbf{x}^p \in \mathcal{D}_u^p$.

The performance of MigLogit is demonstrated on two problem domains. The first is a simulated example and the second is detection of unexploded ordnance (UXO) where the UXO signatures are site-sensitive.

Throughout this section the C in MigLogit is set to $C = 6$ when the comparison is made to logistic regression. In addition, we present a comparison of MigLogit with different C's, to examine the sensitivity of MigLogit's performance to C.

6.1 Synthesized Data

In the first example, the primary data are simulated as two bivariate Gaussian distributions representing class "-1" and class "$+1$", respectively. In particular, we have $\Pr(\mathbf{x}^p|y^p = -1) = \mathcal{N}(\mathbf{x}^p; \boldsymbol{\mu}_0, \boldsymbol{\Sigma})$ and $\Pr(\mathbf{x}^p|y^p = 1) = \mathcal{N}(\mathbf{x}^p; \boldsymbol{\mu}_1, \boldsymbol{\Sigma})$, where the Gaussian parameters $\boldsymbol{\mu}_0 = [0, 0]^T$, $\boldsymbol{\mu}_1 = [2.3, 2.3]^T$, and $\boldsymbol{\Sigma} = \begin{bmatrix} 1.75 & -0.433 \\ -0.433 & 1.25 \end{bmatrix}$. The auxiliary data $\mathcal{D}^a$ are then a selected draw from the two Gaussian distributions, as described in (Zadrozny 2004). We take the selection probability $\Pr(s|\mathbf{x}^p, y^p = -1) = \sigma(w_0 + w_1 K(\mathbf{x}^p, \boldsymbol{\mu}_0^s; \boldsymbol{\Sigma}))$ and $\Pr(s|\mathbf{x}^p, y^p = +1) = \sigma(w_0 + w_1 K(\mathbf{x}^p, \boldsymbol{\mu}_1^s; \boldsymbol{\Sigma}))$, where σ is the sigmoid function, $w_0 = -1$, $w_1 = \exp(1)$, $K(\mathbf{x}^p, \boldsymbol{\mu}_0^s; \boldsymbol{\Sigma}) = \exp\{-0.5(\mathbf{x}^p - \boldsymbol{\mu}_0^s)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}^p - \boldsymbol{\mu}_0^s)\}$ with $\boldsymbol{\mu}_0^s = [2, 1]^T$, and $K(\mathbf{x}^p, \boldsymbol{\mu}_1^s; \boldsymbol{\Sigma}) = \exp\{-0.5(\mathbf{x}^p - \boldsymbol{\mu}_1^s)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}^p - \boldsymbol{\mu}_1^s)\}$ with $\boldsymbol{\mu}_1^s = [0, 3]^T$. We obtain 150 samples of $\mathcal{D}^p$ and 150 samples of $\mathcal{D}^a$, which are shown in Figure 3.

The MigLogit and logistic regression classifiers are trained and tested as explained at the beginning of this section. The results are represented as test error rate as a function of number of primary labeled data used in training, and are shown in Figures 1 and 2. Each curve in Figure 1 is an average over 50 independent trials, with each trial having an independent random selection of $\mathcal{D}_l^p$. Figure 2 presents the active learning results, with $\mathcal{D}_l^p$ actively selected as described in Section 5.

Several observations are made from inspection of Figures 1 and 2.

- The MigLogit consistently outperforms the three standard logistic regression classifiers,

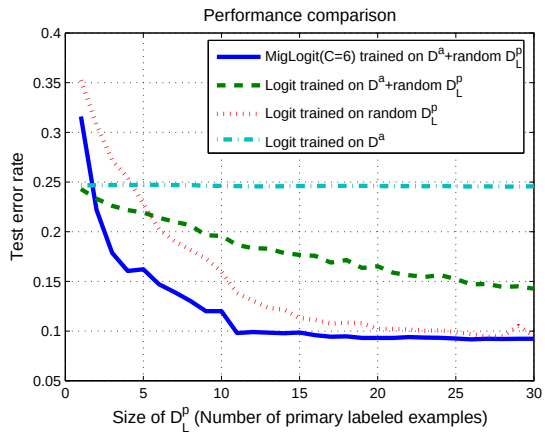


Figure 1: Test error rates of MigLogit and logistic regression on the toy data, as a function of size of $\mathcal{D}_l^p$. The primary labeled data $\mathcal{D}_l^p$ are randomly selected from $\mathcal{D}^p$. The error rates are an average over 50 independent trials of random selection of $\mathcal{D}_l^p$.

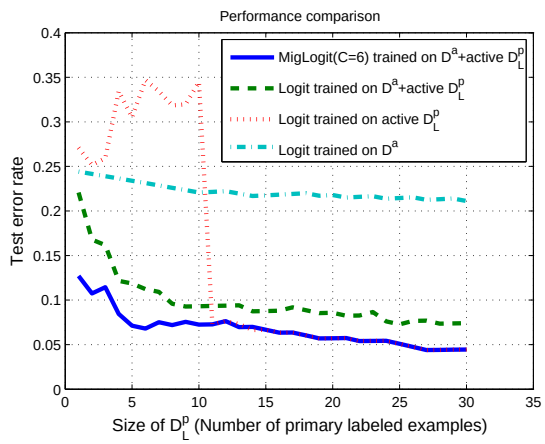


Figure 2: Error rates of MigLogit and logistic regression on the toy data, as a function of size of $\mathcal{D}_l^p$. The primary labeled data $\mathcal{D}_l^p$ are actively selected from $\mathcal{D}^p$, using the method in Section 5.

by a considerable margin. This improvement is attributed to a selective usage of the examples in $\mathcal{D}^a$. In particular, each example in $\mathcal{D}^a$ is employed according to its agreement with $\mathcal{D}_l^p$: a good agreement warrants a higher contribution to determination of the classifier while a poor agreement makes the contribution discounted. The selectivity is implemented through the auxiliary variables which are estimated based on a few examples from $\mathcal{D}^p$.

- The performance of the logistic regression trained on $\mathcal{D}_l^p$ alone changes significantly with the size of $\mathcal{D}_l^p$. This is understandable, considering that $\mathcal{D}_l^p$ are the only examples determining the classifier. The abrupt drop of errors from iteration 10 to iteration 11 in Figure 2 may be because the label found at iteration 11 is critical to determining $\mathbf{w}$.

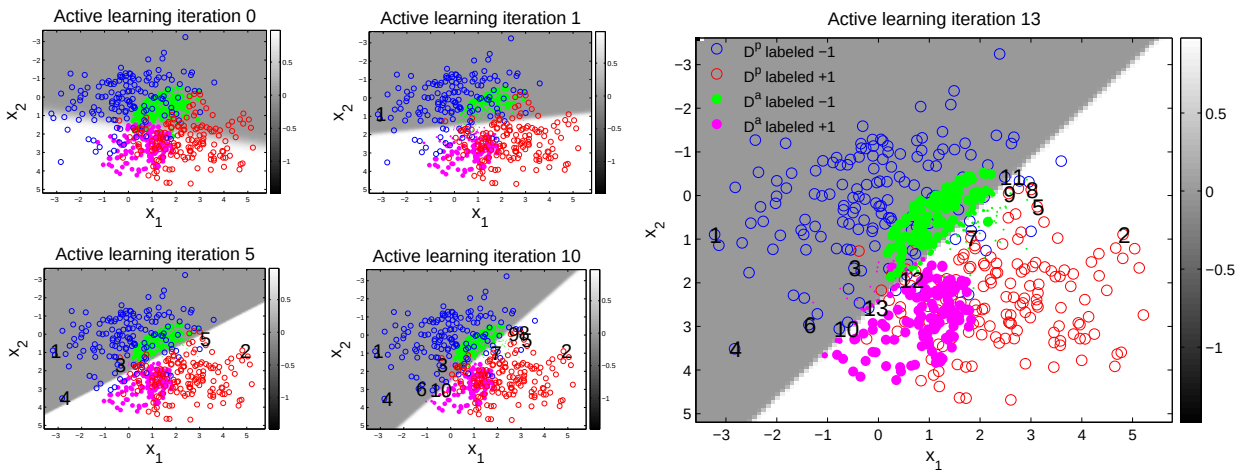


Figure 3: Illustration of active data selection by MigLogit. Only iterations 0,1,5,10,13 are shown. The different symbols are defined as: blue ○ = $\mathcal{D}^p$ labeled "-1", red ○ = $\mathcal{D}^p$ labeled "$+1$", green ● = $\mathcal{D}^a$ labeled "-1", and magenta ● = $\mathcal{D}^a$ labeled "$+1$". The numbers in black denote $\mathcal{D}_l^p$ and represent the order of selection. The smaller ● near the decision boundaries symbolize weakened participation of the associated $\mathcal{D}^a$ in determining $\mathbf{w}$. This may only be visible in the zoomed figure (iteration 13).

- The logistic regression trained on $\mathcal{D}^a$ alone performs significantly worse than MigLogit, reflecting a marked mismatch between $\mathcal{D}^a$ and $\mathcal{D}^p$.
- The logistic regression trained on $\mathcal{D}^a \cup \mathcal{D}_l^p$ improves, but mildly, as $\mathcal{D}_l^p$ grows, and it is ultimately outperformed by the the logistic regression trained on $\mathcal{D}_l^p$ alone, demonstrating that some data in $\mathcal{D}^a$ are mismatched with $\mathcal{D}^p$ and hence cannot be correctly classified along with $\mathcal{D}^p$, if the mismatch is not compensated.
- As $\mathcal{D}_l^p$ grows, the logistic regression trained on $\mathcal{D}_l^p$ alone finally approaches to MigLogit, showing that without the interference of $\mathcal{D}^a$, a sufficient $\mathcal{D}_l^p$ can define a correct classifier.
- All four classifiers benefit from the actively selected $\mathcal{D}_l^p$, and this is consistent with the general observation with active learning (Cohn et al. 1995; Krogh and Vedelsby 1995).

The labeled primary examples $\mathcal{D}_l^p$ play double roles in the learning process. On the one hand they help to find the correct $\mathbf{w}$, and on the other hand they serve as representative primary labeled data in finding the degree of agreement of each auxiliary example with primary data (i.e., estimating the auxiliary variables). Suppose that it requires n_1 primary labeled examples to find the auxiliary auxiliary variables for compensating the mismatch between $\mathcal{D}^a$ and $\mathcal{D}^p$, and that it requires n_2 labeled primary examples alone (without using auxiliary examples) to find the correct classifier. One may conjecture that n_1 is smaller than n_2. Although we have not proven this rigorously, the results in Figures 1 and 2 provide empirical evidence for this being true: note that MigLogit uses much fewer primary labeled examples to find the correct classifier than Logit trained on $\mathcal{D}_l^p$ does.

The double roles assumed by the primary labeled examples make it a critical issue how to select these examples. We already see that actively selected examples give significant boost to the performance. This makes it clear that active learning is a more appropriate strategy than pure random selection, and contributes in an important manner to the proposed method.

To better understand the active selection process, we show in Figure 3 the first few iterations of active learning. Iteration 0 corresponds to the initially empty $\mathcal{D}_l^p$, and iterations 1, 5, 10, 13 respectively correspond to 1, 5, 10, 13 data points selected accumulatively from $\mathcal{D}_u^p$ into $\mathcal{D}_l^p$. Each time a new data point is selected, the $\mathbf{w}$ is re-trained, yielding the different decision boundaries. As can be seen in Figure 3, the decision boundary does not change much after 10 data are selected, demonstrating convergence.

In Figure 3, each auxiliary data point $\mathbf{x}_i^a \in \mathcal{D}^a$ is symbolically displayed with a size in proportion to $\exp(-y_i^a \mu_i/12)$, hence a small symbol of auxiliary data corresponds to large $y_i^a \mu_i$ and hence indicates a discounted contribution of the i-th auxiliary example to determination of $\mathbf{w}$. The auxiliary data that cannot be correctly classified along with the primary data are de-emphasized by the MigLogit. Usually the auxiliary data near the decision boundary are de-emphasized.

6.2 Application to Detection of Site-Sensitive Unexploded Ordnance (UXO)

Unexploded ordnance (UXO) consists of ordnance that did not explode upon impact with the ground. The UXO items are typically buried and consist of significant quantities of metal. Sensing of UXO is typically performed using electromagnetic induction (EMI) and magnetometer sensors. The principal challenge involves distinguishing actual UXO from buried non-ordnance conducting materials. For a more detailed general description of UXO sensing, see (Zhang et al. 2003).

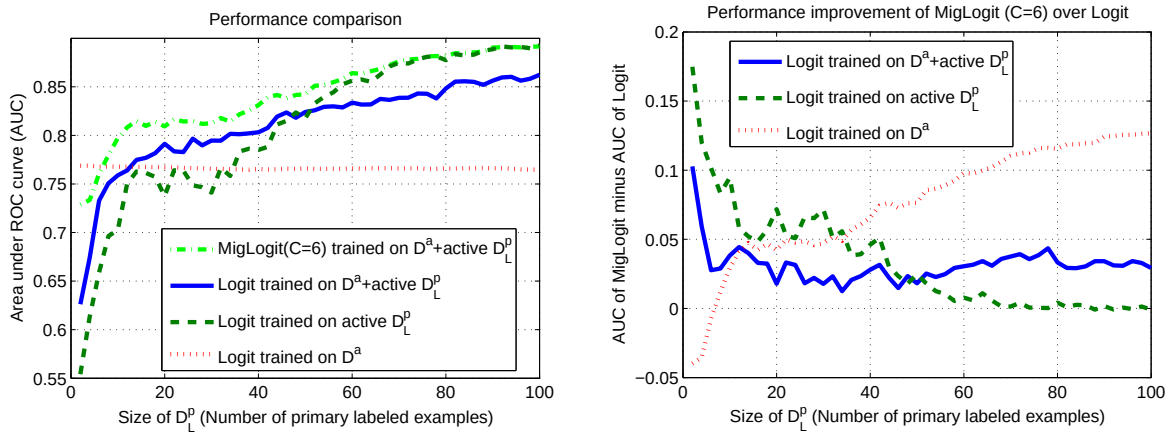
The sensor signature of a given UXO item is dependent on the soil properties as well as the history of the site in which it is located, the latter having a particular strong influence on the signature. The site history is dictated by complex factors such as co-located ordnance, the way the ordnance impacted the soil, and the surrounding man-made conducting clutter and UXO fragments. Therefore UXO detection is a typical site-sensitive problem.

The site-sensitivity makes standard supervised classification techniques an inappropriate choice for UXO detection, due to the difficulty in constituting a universal training set for classifier design. The training examples collected at previous sites are often not appropriate for use for analysis of the current site since the current site is often different from the previous ones (in the sense described above). Despite these disparities, the examples from previous sites are not totally useless; indeed, they can provide quite useful information about the examples for the current site (particularly for the UXO, since the ordnance types at different sites are often the same or similar; the clutter signatures are most often site specific). The usefulness of existing labeled data for a new site of interest is dictated by the characteristics of the new site, as well as on the characteristics of the sites from which the labeled data were acquired; these interrelationships are complex and often difficult to characterize *a priori* (often accurate records are not available about the history of a former bombing site).

Let the examples at the current UXO cite be distributed according to $\mathcal{T}(\mathbf{x}, y)$, and the examples at a previous UXO cite be distributed according to $\mathcal{A}(\mathbf{x}, y)$. It is seen that the empirical loss

for detection of UXO at the current cite is well described by (2). Therefore one can employ the technique of MigLogit to design the desired classifier.

To demonstrate the utility of MigLogit in UXO detection, we here consider two UXO sites and design the classifier for the primary site (the one we are interested in) by using examples from another site (the auxiliary site). The auxiliary site is called *Jefferson Proving Ground (JPG)*, for which one is provided with the EMI and magnetometer measurements as well the associated labels (which are binary: UXO or non-UXO). The examples from the auxiliary site constitutes the auxiliary data $\mathcal{D}^a$. The primary site we are interested in is called *Badlands*, for which we have unlabeled EMI and magnetometer measurement for constituting the primary data $\mathcal{D}^p$. The labeled JPG data consists of 104 total items, of which 16 are UXO and 88 are non-UXO. The Badlands site consists of a total of 492 items, 57 of which are UXO and the remaining 435 are non-UXO. These two former bombing ranges exist at two very different geographical locations within the United States.



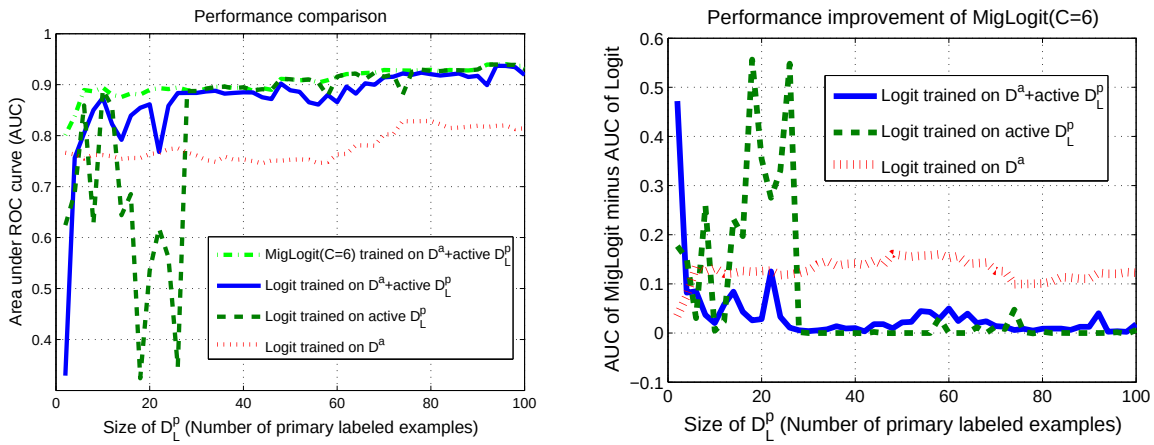
(a) AUC

(b) AUC of MigLogit minus AUC of logistic regression

Figure 4: (a) The area under ROC curve (AUC) of MigLogit and logistic regression on the UXO data, as a function of size of $\mathcal{D}_l^p$ (b) The AUC of MigLogit minus the AUC's of logistic regression. The auxiliary data are collected at *Jefferson Proving Ground (JPG)* and the primary data are collected at *Badlands*. The primary labeled data $\mathcal{D}_l^p$ are randomly selected from $\mathcal{D}^p$. Each curve is an average over 50 independent trials of random selection of $\mathcal{D}_l^p$.

The UXO sensor measurements are mapped to four dimensional feature vectors $[\log(M_p)$, $\log(M_z)$, z, $\log(\frac{M_p}{M_z})]$, where M_p and M_z are the dipole moments perpendicular and parallel to the target axis, respectively, and z is the approximate target depth (Zhang et al. 2003). These parameters are estimated by fitting the EMI and magnetometer measurements to a physical model (Zhang et al. 2003); the features from this study are available upon request to the authors. Each feature is normalized to have zero mean and unitary variance. In UXO detection, one is interested in the receiver's operating characteristic (ROC) curve, particularly the area under ROC curve (AUC) (Hanley and McNeil 1982).

The results are presented in Figures 4(a) and 5(a), where each curve is the area under ROC curve as a function of the size of $\mathcal{D}_l^p$. The results in Figure 4(a) are obtained by randomly la-



(a) AUC

(b) AUC of MigLogit minus AUC of logistic regression

Figure 5: (a) The area under ROC curve (AUC) of MigLogit and logistic regression on the UXO data, as a function of size of $\mathcal{D}_l^p$ (b) The AUC of MigLogit minus the AUC's of logistic regression. The auxiliary data are collected at *Jefferson Proving Ground (JPG)* and the primary data are collected at *Badlands*. The primary labeled data $\mathcal{D}_l^p$ are actively selected from $\mathcal{D}^p$, based on the method in Section 5.

beling primary data and by averaging the AUC's over 50 independent trials. The results in Figure 5(a) are obtained by actively labeling primary data using the method in Section 5. For a better view of the improvement achieved by MigLogit, we plot in Figures 4(b) and 5(b) the AUC of MigLogit with the AUC of each logistic regression classifier subtracted. A positive difference indicate performance improvement while a negative difference indicates performance degradation. We have the following observations:

- With $\mathcal{D}_l^p$ determined randomly, MigLogit outperforms all logistic regression classifiers except at the early part of the curves, where there are very few examples in $\mathcal{D}_l^p$. As discussed at the end of Section 6.1, the primary labeled examples are critical to the performance of MigLogit. With a few randomly selected examples one may not be able to find the appropriate auxiliary variables, leading to a poor compensation of the mismatch between $\mathcal{D}^p$ and $\mathcal{D}^a$ and therefore performance degradation.

- With $\mathcal{D}_l^p$ actively determined, MigLogit outperforms all logistic regression classifiers, regardless of the number of primary labeled examples. This verifies that a good choice of $\mathcal{D}_l^p$ is important to the performance of MigLogit.

- Active learning is not only beneficial to MigLogit, but to other classifiers as well, again demonstrating the advantage of active learning.

- All conclusions observed in the simulated results extend to the UXO results here.

These observations suggest that MigLogit successfully leverages the auxiliary data from previous UXO sites to quickly find the correct classifier for the new site, requiring much fewer

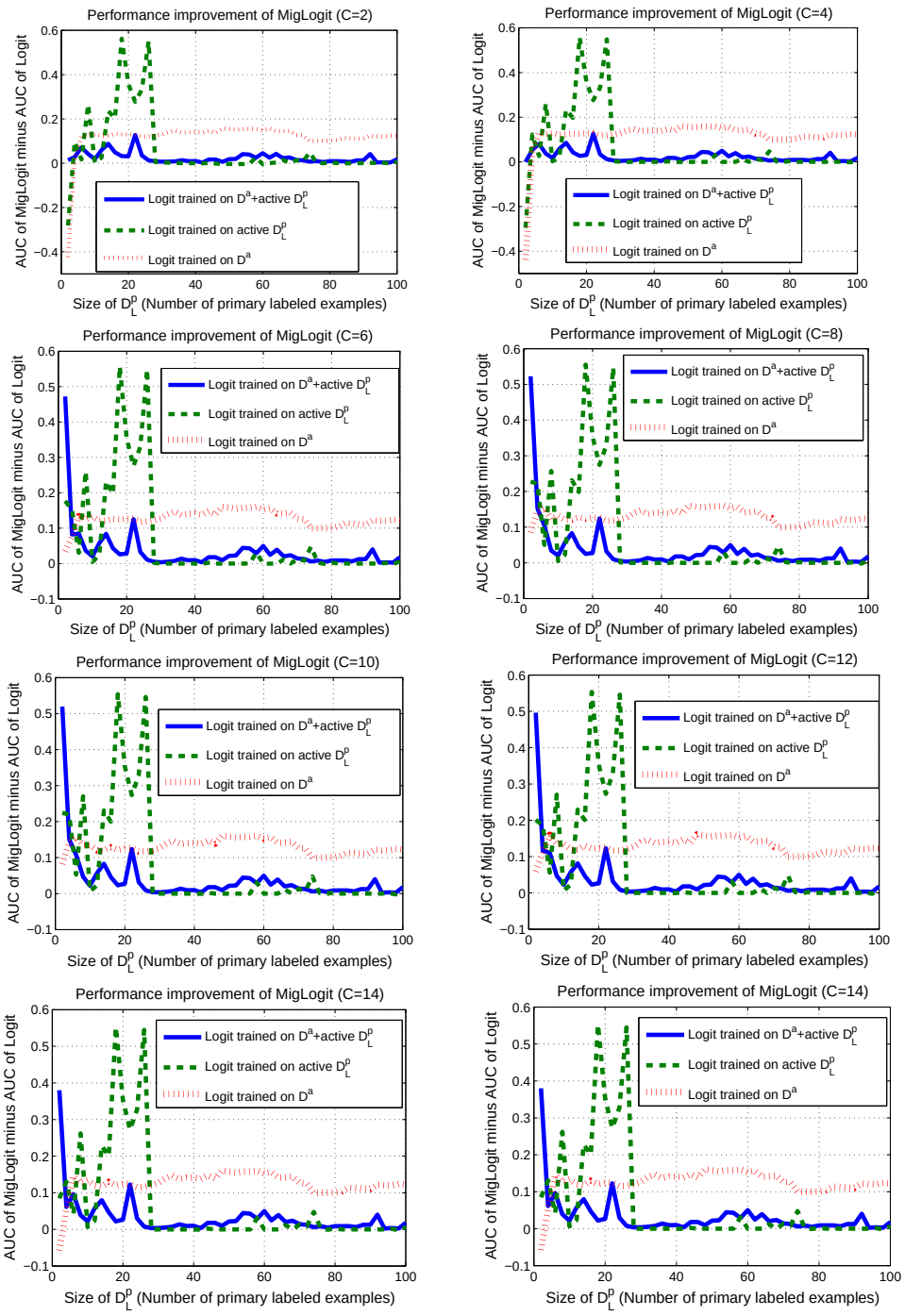


Figure 6: Performance of MigLogit with different choices of C, in the UXO detection problem. The vertical axis is the AUC of MigLogit minus the AUC's of logistic regression. The primary labeled data $\mathcal{D}_l^p$ are actively selected from $\mathcal{D}^p$. From top-left to right-bottom, $C = 2, 4, 6, 8, 10, 12, 14, 16$.

labeled data from the new site than standard classifiers. The results for the actual (measured) UXO data suggest that the algorithm captures the concept drift associated with realistic prob-

lems of practical importance.

6.3 Robustness of MigLogit

We have discussed in (4) how to choose C in MigLogit. In this subsection, we show that when the choice is not accurate, MigLogit still yields robust results.

We consider the same UXO data and use the same experimental settings as in the previous subsection, except that we vary C in MigLogit to examine its robustness. The $\mathcal{D}_l^p$ are determined by active learning as described in Section 5. We consider eight different values of C, $C = 2, 4, 6, 8, 10, 12, 14, 16$, to examine the differences in the results obtained under these settings. The results are shown in Figure 6. It is seen that over this wide range of choices for C, MigLogit consistently yields superior performances except in a few cases, which occur when the size of $\mathcal{D}_l^p$ is very small and C is either too large or too small. These results demonstrate the robustness of MigLogit to the choice of C, particularly when active learning is invoked. With different C, the $\mathcal{D}_l^p$ are also selected differently, which counteracts the effect of C and increase the robustness of MigLogit.

7 Conclusions

We have proposed an algorithm, called *migratory logistic regression* (MigLogit), for learning in the presence of concept change between the (auxiliary) training data $\mathcal{D}^a$ and the (primary) testing data $\mathcal{D}^p$. The basic idea of our method is to introduce an auxiliary variable μ_i for each example $(\mathbf{x}_i^a, y_i^a) \in \mathcal{D}^a$, which allows $\mathbf{x}_i^a$ to migrate to the class y_i^a when it cannot be correctly classified along with $\mathbf{x}^p$ by the classifier. The migrations of $\mathcal{D}^a$ are controlled by the inequality constraint $\frac{1}{N^a}\sum_{i=1}^{N^a} y_i^a \mu_i \leq C$, where $C \geq 0$ is an appropriate bound limiting the average migration. The primary labeled data $\mathcal{D}_l^p$ play a pivotal role in correctly learning the classifier, and we have presented a method to actively selecting $\mathcal{D}_l^p$, which enhances the adaptivity of the entire learning process. We have developed a fast learning algorithm to enhance the ability of MigLogit to handle large auxiliary data sets.

The results from both synthesized data and data collected at actual unexploded ordnance (UXO) sites show that MigLogit yields significant improvements over the standard logistic regression, demonstrating that if the classifier trained on $\mathcal{D}^a$ is to generalize well to $\mathcal{D}^p$, the mismatch between $\mathcal{D}^a$ and $\mathcal{D}^p$ must be compensated.

In the work presented here it was assumed that we had an existing set of labeled data $\mathcal{D}^a$, for which the goal was to learn relationships with a primary set of (unlabeled or partially labeled) data $\mathcal{D}^p$. In some problems we may have $M-1$ existing labeled data sets, indexed by $m = 1, 2, \cdots, M-1$, and we are interested in learning the characteristics (concept) of a new (M-th) unlabeled or partially labeled data set. Using the method presented here, all data in the existing $M-1$ data sets would be combined to define $\mathcal{D}^a$. There is a question as to whether this is the most effective way to address this problem. A related technique for handling multiple data sets is based on the notion of multitask learning (MTL) or inductive transfer (Baxter 2000; Caruana 1997; Yu et al. 2005). Here a task refers to classifier design based on a specific data set. The goal in multitask learning is to enhance training examples used to learn a given task by borrowing information from related tasks. Information borrowing is accomplished by learning

the multiple tasks simultaneously under a unified framework. This is particularly beneficial when each task has limited training examples, since information borrowing allows examples of related tasks to be utilized when learning the target task. Note MTL does not require the tasks to be ordered in time; it only assumes that the tasks are related in some manner.

Existing MTL algorithms are distinguished by the way information borrowing is implemented. In a neural network, in which each output node can encode a task (Bakker and Heskes 2003; Caruana 1997; Liao and Carin 2006), information borrowing is implemented by a common internal representations such as hidden nodes and input-to-hidden weights. The method in (Evgeniou et al. 2005) employs a task-kernel to capture the similarity between any two tasks. The task-kernel is used to construct a quadratic regularization term for the parameters across all tasks, which implements information borrowing from one task to another. In Bayesian hierarchical models (Dominici et al. 1997), a common prior distribution is placed over the model parameters in different tasks to represent the information shared between tasks. In nonparametric Bayesian models (Xue et al. 2007), information borrowing is carried out by by a common Dirichlet process (DP) (Ferguson 1973) employed to generate the nonparametric prior distribution over the model parameter in each task. The method in (Wu and Dietterich 2004) learns a classifier based on a weighting of two tasks, with the auxiliary task given lower weight to reflect that it has a discounted contribution to the classifier learning. Here the target task borrows information from the auxiliary one, through the discounted contribution.

An interesting direction for future research involves examination of the relationship between the concept-drift algorithm presented here and the aforementioned MTL approaches, each of which constitutes a method for implementing transfer learning. In particular, it is of interest to examine the value in retaining the separation of the $M-1$ labeled data sets, as in MTL, versus aggregating them to define $\mathcal{D}^a$. For the UXO problem considered as a practical problem in this paper, one may have cleaned $M-1$ previous sites before considering the M-th (although this was not the case in the example considered here, in which we only had labeled data from one previous site). This line of investigation will be the focus of a future study.

Acknowledgments

The research reported here was supported by the Strategic Environmental Research and Development Program (SERDP).

References

B. Bakker and T. Heskes. Task clustering and gating for Bayesian multitask learning. *Journal of Machine Learning Research*, pages 83–99, 2003.

J. Baxter. A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12:149–198, 2000.

D. P. Bertsekas. *Nonlinear Programming (2nd Edition)*. Athena Scientific, 1999.

R. Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Advances in Neural Information Processing Systems*, 7:705–712, 1995.

T.M. Cover and J.A. Thomas. *Elements of Information Theory*. John Wiley, New York, 1991.

F. Dominici, G. Parmigiani, K.H. Reckhow, and R.L. Wolpert. Combining information from related regressions. *Journal of Agricultural, Biological, and Environmental Statistics*, 2(3):313–332, 1997.

T. Evgeniou, C. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research*, pages 615–637, 2005.

V. V. Fedorov. *Theory of Optimal Experiments*. Academic Press, New York, 1972.

T. Ferguson. A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1: 209–230, 1973.

J. Hanley and B. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143:29–36, 1982.

J. Heckman. Sample selection bias as a specification error. *Econometrica*, 47:153–161, 1979.

R. Klinkenberg and T. Joachims. Detecting concept drift with support vector machines. In P. Langley, editor, *Proceedings of the Seventeenth International Conference on Machine Learning (ICML)*, pages 487–494, 2000.

R. Klinkenberg and S. Ruping. Concept drift and the importance of examples. In J. Franke, G. Nakhaeizadeh, and I. Renz, editors, *Text Mining – Theoretical Aspects and Applications*, pages 55–77. Physica-Verlag, Heidelberg, Germany, 2003.

A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. *Advances in Neural Information Processing Systems*, 7:231–238, 1995.

X. Liao and L. Carin. Radial basis function network for multi-task learning. In *Advances in Neural Information Processing Systems 18*, pages 795–802. 2006.

D. J. C. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4:590–604, 1992.

C.P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer-Verlag, 1999.

J. B. Tenenbaum. Bayesian modeling of human concept learning. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems*, volume 11. Cambridge, MA: MIT Press, 1999.

V. Vapnik. *Statistical Learning Theory*. John Wiley, New York, 1998.

V. N. Vapnik. An overview of statistical learning theory. *IEEE Trans. Neural Networks*, 10(5): 988–999, 1999.

H. Wang, W. Fan, P. S. Yu, and J. Han. Mining concept-drifting data streams using ensemble classifiers. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 226–235, New York, NY, USA, 2003. ACM Press. ISBN 1-58113-737-0.

G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Machine Learning*, 23:69–101, 1996.

G. Widmer and M. Kubat. Effective learning in dynamic environments by explicit concept tracking. In Pavel B. Brazdil, editor, *European Conference on Machine Learning*, pages 227–24. SpringerVerlag, 1993.

P. Wu and T. G. Dietterich. Improving svm accuracy by training on auxiliary data sources. *Proceedings of the 21st ICML*, 2004.

Y. Xue, X. Liao, L. Carin, and B. Krishnapuram. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research (JMLR)*, 8:35–63, 2007.

K. Yu, V. Tresp, and A. Schwaighofer. Learning gaussian processes from multiple tasks. In *The 22nd International Conference on Machine Learning (ICML)*, 2005.

B. Zadrozny. Learning and evaluating classifiers under sample selection bias. *Proceedings of the 21st ICML*, 2004.

Y. Zhang, L. Collins, H. Yu, C.E. Baum, and L. Carin. Sensing of unexploded ordnance with magnetometer and induction data: Theory and signal processing. *IEEE Trans. Geoscience Remote Sensing*, 41(5):1005–1015, 2003.

Appendix

Proof of Theorem 1: Let $f'(z)$ be the first derivative of $f(z)$. We have $\sum_{i=1}^{N} f(b_i + z_i) = \sum_{i=1}^{N} f(b_i) + \sum_{i=1}^{N} \int_0^{z_i} f'(x + b_i)dx$. The first term on the right side is a constant and hence, the problem in (13) is equivalent to

$$\max_{\{z_i\}} \quad \sum_{i=1}^{N} \int_0^{z_i} f'(b_i + x)dx \tag{A-1}$$

Because $f''(z) < 0$, we have for any $\tau_1 \leq \tau_2$ that $f'(\tau_1 + x) \geq f'(\tau_2 + x)$ and consequently

$$\begin{array}{c} \int_0^{\Delta} f'(\tau_1 + x)dx \geq \int_0^{\Delta} f'(\tau_2 + x)dx \\ \forall \quad \tau_1 \leq \tau_2 \quad \text{and} \quad \Delta \geq 0 \end{array} \tag{A-2}$$

By (12), there exists $0 \leq r < n(b_{n+1} - b_n)$ such that $R = nb_n - \sum_{k=1}^{n} b_k + r = \sum_{k=1}^{n} k\Delta_k$ where $\Delta_k = b_{k+1} - b_k$ for $k = 1, \cdots, n-1$, and $\Delta_n = r/n$. We now use (A-2) to distribute $\Delta_1, 2\Delta_2, \cdots, n\Delta_n$ to $z_1, z_2, \cdots, z_N$ such that the resulting $\{z_i\}$ maximize (A-1). As $\Delta_k \geq 0$ for

$k = 1, \cdots, n$, and any distribution of $\{k\Delta_k\}_{k=1}^N$ to $\{z_k\}_{k=1}^N$ makes $\sum_{i=1}^N z_i = \sum_{k=1}^n k\Delta_k = R$, the constraints of (14) and (15) are automatically satisfied.

Initially $z_i = 0$ for $i = 1, 2, \cdots, N$.

As $\Delta_1 = b_2 - b_1 \geq 0$, by (A-2), $\int_0^{\Delta_1} f'(b_1 + x)dx \geq \int_0^{\Delta_1} f'(b_2 + x)dx$, therefore Δ_1 is distributed to z_1, i.e., $z_1 \leftarrow z_1 + \Delta_1$, which makes $b_1 + z_1 = b_2$.

Similarly $\Delta_2 = b_3 - b_2 \geq 0$, by (A-2), $\int_0^{\Delta_2} f'(b_2 + x)dx \geq \int_0^{\Delta_2} f'(b_3 + x)dx$, therefore $2\Delta_2$ is equally distributed to z_1 and z_2, i.e., $z_1 \leftarrow z_1 + \Delta_2$ and $z_2 \leftarrow z_2 + \Delta_2$, which makes $b_1 + z_1 = b_2 + z_2 = b_3$.

Generally, $k\Delta_k$ is equally distributed to $z_1, z_2, \cdots, z_k$. After the distribution of $k\Delta_k$, $k = 1, 2, \cdots, n$, we have $z_k = \sum_{i=k}^n \Delta_i$ for $k = 1, 2, \cdots, n$ and $z_k = 0$ for $k = n+1, n+2, \cdots, N$, which is equal to the solution in (16). Because the problem is strictly concave, the solution is unique and globally optimal. □

Derivation of Equation (19): By definition of logistic regression, $\mathbf{w}$ is the parameter of the conditional distribution $\Pr(y|\mathbf{x}) = \sigma(y\mathbf{w}^T\mathbf{x})$, with $\mathbf{x}$ given and fixed. Let $\mathbf{g} = \partial \ln \sigma(y\mathbf{w}^T\mathbf{x})/\partial \mathbf{w} = [1 - \sigma(y\mathbf{w}^T\mathbf{x})]y\mathbf{x}$. Then $\mathbb{E}_y(\mathbf{g}\mathbf{g}^T) = \sum_{y=-1,1} \sigma(y\mathbf{w}^T\mathbf{x})[1 - \sigma(y\mathbf{w}^T\mathbf{x})]^2\mathbf{x}\mathbf{x}^T$. Using $\sigma(-\mathbf{w}^T\mathbf{x}) = 1 - \sigma(\mathbf{w}^T\mathbf{x})$, we obtain $\mathbb{E}_y(\mathbf{g}\mathbf{g}^T) = \sigma(\mathbf{w}^T\mathbf{x})[1 - \sigma(\mathbf{w}^T\mathbf{x})]\mathbf{x}\mathbf{x}^T$. Summing $\mathbb{E}(\mathbf{g}\mathbf{g}^T)$ over all primary and auxiliary data points (assuming the data are independent), we obtain the formula of $\mathbf{Q}$. □

Detection of Unexploded Ordnance via Efficient Semi-Supervised and Active Learning

Qiuhua Liu, Xuejun Liao, and Lawrence Carin
Department of Electrical and Computer Engineering
Duke University
Durham, NC 27708-0291, USA

***Abstract*— Semi-supervised learning and active learning are considered for UXO detection. Semi-supervised learning algorithms are designed using both labeled and unlabeled data, where here labeled data corresponds to sensor signatures for which the identity of the buried item (UXO/non-UXO) is known; for unlabeled data one only has access to the corresponding sensor data. Active learning is used to define which unlabeled signatures would be most informative to improve classifier design, if the associated label could be acquired (where for UXO sensing the label is acquired by excavation). A graph-based semi-supervised algorithm is proposed, employing the idea of a random Markov walk on a graph, thereby exploiting knowledge of the data manifold (where the manifold is defined by both the labeled and unlabeled data). The resulting algorithm is then used to infer labels for the unlabeled data, providing a probability that a given unlabeled signature corresponds to a buried UXO. An efficient active-learning procedure is developed for this algorithm, based on a mutual-information measure. In this manner one initially performs excavation with the purpose of acquiring labels to improve the classifier, and once this active-learning phase is completed the resulting semi-supervised classifier is then applied to the remaining unlabeled signatures, to quantify the probability that each such item is UXO. Example classification results are presented for an actual UXO site, based on electromagnetic induction and magnetometer data. Performance is assessed in comparison to other semi-supervised approaches, as well as to supervised algorithms.**

I. Introduction

Unexploded ordnance (UXO) correspond to explosive devices ($e.g.$, bombs) that did not explode upon impact with the ground, and that are subsequently buried intact or partially intact. Some UXO may also exist on the surface of the ground, but we here assume these are removed via manual inspection, and therefore this paper focuses on detecting buried UXO. There are several sensing techniques that have been developed over the last several decades for detection of buried UXO. Most widely used among these are electromagnetic induction (EMI) [1, 2, 3] and magnetometers [4]. Both of these approaches are based on sensing magnetic signatures. An EMI sensor is an active approach, whereby electromagnetic radiation is emitted, and one measures the signals scattered off targets. Such that one achieves sufficient ground penetration, EMI systems are typically designed to operate at kilohertz frequencies (in the inductive regime). By contrast, magnetometers are passive sensors, which measure the static magnetic field of the earth, and hence the presence of ferrous targets, which yield a corresponding perturbation to the earth's magnetic field.

A UXO is any explosive device that has not detonated, and therefore UXO are dangerous if disturbed. However, although the device didn't detonate, in many cases the item is deformed upon impact, with possible components ($e.g.$, tail wings) broken off. In addition, there are many different types of explosives (bombs) that may have been deployed. These factors significantly complicate one's ability to distinguish UXO from non-UXO based on the EMI and/or magnetometer signature. Specifically, many types of buried benign metal items are often readily confused for UXO, based on the sensor signature. Consequently, the unnecessary excavation of non-UXO items often constitutes the principal cost of UXO cleanup (there are typically far more non-UXO buried metal items than there are actual UXO). Therefore, classification of UXO constitutes a significant sensing and classification challenge.

Classification using EMI and magnetometer sensors is typically not performed directly on the measured data, but on features extracted therefrom. Specifically, parametric models have been developed for the response of targets as viewed from such sensors, with most of these models based on a dipole approximation [3]. The parameters extracted from the models, when fitting is performed to the measured data, are typically employed to constitute feature vectors within the subsequent classification algorithm. Most of these algorithms are supervised, in the following sense. A set of labeled feature vectors are assumed given (the identity, UXO/non-UXO, of each feature vector is known), and these data are used to design a classifier. Numerous such classifiers have been considered for UXO detection, such as kernel matching pursuits [5], support vector machines [3], and likelihood-ratio tests [3]. There are two limitations of such approaches: (i) the assumption of the presence of an $appropriate$ labeled data set is tenuous in many cases, and (ii) even when such labeled data are available, a purely supervised algorithm doesn't exploit the contextual information provided by the unlabeled data.

General interest in these latter two issues has motivated recent research in the machine learning community. Specifically, active learning [6, 7] is a framework whereby the acquisition of labeled data is integrated within classifier design. Using appropriate information-theoretic measures, an active-learning algorithm asks which of the unlabeled feature vectors would be most informative for classifier design if the associated labels could be made available. This idea has been applied previously in the context of UXO detection [5]. The new aspect of the

work considered here is that this active-learning framework is placed within the context of a semi-supervised learning setting. Specifically, in addition to actively acquiring the labeled data (performing item excavations selectively for the purpose of algorithm learning), a semi-supervised algorithm exploits contextual information provided by all of the unlabeled data (the classification of any one unlabeled feature vector is placed within the context of all unlabeled feature vectors). In the UXO problem the EMI/magnetometer data are often all collected at once, typically using a cart-based system [4]; recall that the UXO of interest are all buried, and therefore they are not dangerous until excavation begins. Therefore, one may perform feature extraction on all of the signatures at once, and the contextual information provided by these data may be of utility in improving classification performance.

Semi-supervised learning has been an area of significant recent interest in the machine-learning community [8, 9, 10, 11, 12, 13, 14, 15], where exploitation of the information available in the unlabeled data has been demonstrated to often add value. To our knowledge this paper represents the first use of such an approach as applied to the UXO problem. As discussed below, the semi-supervised approach proposed here is new in its own right, and has advantages relative to other such techniques currently in the literature.

To date, there have been several semi-supervised methods developed. The generative-model method, an early semi-supervised method, estimates the joint probability of data and labels via expectation-maximization (EM), treating the missing labels of unlabeled data as hidden variables; this method was studied in statistics for mixture estimation [16] and has been reformulated for semi-supervised classification [15]. Co-training [13], another early method, exploits two independent subvectors of features, using one to provide the label estimates for the other; co-training has received renewed interest recently, particularly theoretically. The semi-supervised support vector machine (SVM) [12] represents a more recent method, which maximizes the margin between classes, taking into account both labeled and unlabeled data. Graph-based methods [11, 10, 14, 9], the main focus of current research in semi-supervised learning, exploits the assumption that strongly connected data points (in feature space) should share the same label, and utilizes spectral graph theory to quantify the between-data connectivity. For a more complete review of the literature, see [8].

Most graph-based algorithms operate in a transductive fashion, i.e., they directly learn the labels of the unlabeled data, instead of learning a classifier first and then using the classifier to infer the unseen labels (inductive learning). While transductive algorithms avoid the problem of model selection for a classifier, they lack a principled way of predicting the labels of data out of the training set. The work in [9] addresses this problem by constructing a graph-based prior distribution on the parameters of a classifier and learns the classifier by maximizing the posterior (MAP estimation); the prior utilizes both labeled and unlabeled data, thus enforcing semi-supervised learning. Several drawbacks are inherent in the algorithm in [9]. For example, the hyper-parameter balancing the importance of the prior relative to the data likelihood needs to be learned.

In this paper, with a focus on the UXO-sensing application, we present an algorithm for learning parametric classifiers on a partially labeled data manifold, by representing the manifold as a graph; each vertex on the graph represents a data point and the weighted edge between two vertices manifests the immediate connectivity between the corresponding data points. We are motivated by the work in [11] and build the t-step connectivity between data points via a Markov random walk on a manifold. To account for heterogeneities in the data manifold, we let the random walk take different step-sizes at different data locations; each step-size dictates a Markov transition matrix and we select the step-size to assemble the transition matrix for the entire manifold.

The remainder of the paper is organized as follows. In Sections II and III we, respectively, discuss the semi-supervised learning algorithm and the active-learning framework. These discussions are presented in a general setting, applicable to any remote-sensing problem for which (i) all of the unlabeled data are available simultaneously, and (ii) there is an opportunity to selectively acquire labels on a subset of the unlabeled feature vectors. In the work considered here these labels may be acquired via selective excavation, while in other settings one may employ a human analyst or potentially another (higher-resolution) sensor, selectively deployed. The specific application to UXO sensing is discussed in Sec. IV, wherein the sensors and feature vectors considered are described. Results are presented for an actual UXO site, using magnetometer and/or EMI sensor data, and comparisons are made to other approaches (other classes of supervised and semi-supervised algorithms). Conclusions from this work are provided in Section V.

II. The Semi-supervised Learning Algorithm

A. The Graph Representation of a Partially Labeled Data Manifold

Let $G = (\mathcal{X}, \mathbf{W})$ be a graph, where $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\}$ is the set of vertices and $\mathbf{W} = [x_{ij}]_{N\times N}$ is the affinity matrix with the (i,j)-th element w_{ij} indicating the strength of immediate connectivity between vertices $\mathbf{x}_i$ and $\mathbf{x}_j$. For the purpose of data classification, the vertex set $\mathcal{X}$ coincides with the set of data points (labeled or unlabeled), and w_{ij} is a quantitative measure of the closeness of data points $\mathbf{x}_i$ and $\mathbf{x}_j$. In the semi-supervised setting, only a subset of $\mathcal{X}$ are provided with class labels, and the remaining data points are unlabeled, and therefore we have a partially labeled graph.

Although there are many alternative ways of defining the connectivity w_{ij}, here we consider a radial basis function

$$w_{ij} = \exp(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma_i^2}) \tag{1}$$

where $\|\cdot\|$ represents the Euclidean norm. While the affinity matrix may provide a reasonable local similarity among the data points, it is not a good representation of the global similarity measure of the data sets. Following [11], we construct a Markov random walk based on the affinity measure, which is

capable of incorporating both the high-density clustering property and the manifold structure of the data set. Specifically, we induce a Markov transition matrix $\mathbf{A} = [a_{ij}]_{N\times N}$, where the (i,j)-th element

$$a_{ij} = \frac{W_{ij}}{\sum_{k=1}^{N} W_{ik}} \tag{2}$$

gives the probability of walking from $\mathbf{x}_i$ to $\mathbf{x}_j$ by taking a single step. In general we are interested in a t-step random walk, the transition matrix of which is given by $\mathbf{A}$ raised to the power of t, *i.e.*, $\mathbf{A}^t = [a_{ij}^{(t)}]_{N\times N}$. The $\mathbf{A}^t$ is row stochastic, where each element $a_{ij}^{(t)}$ represents the probability that the Markov process starts from $\mathbf{x}_i$ and ends at $\mathbf{x}_j$ by taking t-step random walks. As a special case, $\mathbf{A}^t$ degenerates to an identity matrix when $t = 0$, which means one can only stay at a single data point when no walk is performed.

In specifying the Markov transition matrix in (1) we have used a distinct σ for each data point $\mathbf{x}$. In the random walk, σ can be thought of as the step-size. Therefore location-dependent step-sizes allow one to account for possible heterogeneities in the data manifold — at locations where data are densely distributed a small step-size is enough, whereas at locations where data are sparsely distributed a large step-size is necessary to connect a data point to its nearest neighbor. A simple choice of the heterogeneous σ is to let σ_i to be a fraction of the shortest Euclidean distance between $\mathbf{x}_i$ and all other data points in $\mathcal{X}$. This ensures each data point is immediately connected to at least one neighbor.

B. Neighborhood-Based Learning

Any two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are said to be t-step neighbors, denoted as $\mathbf{x}_j \overset{t}{\sim} \mathbf{x}_i$, if $a_{ij}^{(t)} > 0$. Then $\mathcal{N}_t(\mathbf{x}_i) = \{\mathbf{x} : \mathbf{x} \overset{t}{\sim} \mathbf{x}_i\} \subseteq \mathcal{X}$, which represents the set of t-step neighbors of $\mathbf{x}_i$, is called the t-step neighborhood of $\mathbf{x}_i$. When $t = 0$, the neighborhood shrinks to a single data point, $\mathcal{N}_0(\mathbf{x}_i) = \{\mathbf{x}_i\}$. We define the probability of label y_i given the t-step neighborhood of $\mathbf{x}_i$ as

$$p(y_i|\mathcal{N}_t(\mathbf{x}_i), \boldsymbol{\theta}) = \sum_{j=1}^{N} a_{ij}^{(t)}\, p(y_i|\mathbf{x}_j, \boldsymbol{\theta}) \tag{3}$$

where the magnitude of $a_{ij}^{(t)}$ automatically determines the contribution of $\mathbf{x}_j$ to the neighborhood, thus we are allowed to run the index j over the entire $\mathcal{X}$. Expression $p(y_i|\mathbf{x}_j, \boldsymbol{\theta})$ is the probability of label y_i given a single data point $\mathbf{x}_j$ (zero-step neighborhood) and it's represented by a standard probabilistic classifier parameterize by $\boldsymbol{\theta}$. In this paper we consider binary classification with $y \in \{-1, 1\}$, and choose the form of $p(y_i|\mathbf{x}_i, \boldsymbol{\theta})$ as logistic regression classifier

$$p(y_i|\mathbf{x}_j, \boldsymbol{\theta}) = \frac{1}{1 + \exp(-y_i \boldsymbol{\theta}^T \mathbf{x}_j)} \tag{4}$$

where we assume a constant element 1 is prefixed to each feature vector $\mathbf{x}$ (the prefixed $\mathbf{x}$ is still denoted as $\mathbf{x}$ for notational simplicity), thus the first element in $\boldsymbol{\theta}$ is a bias term. Arbitrarily one may set $y = 1$ as corresponding to a UXO, and $y = -1$ as corresponding to a non-UXO.

We distinguish between the classifier in (3) and the typical logistic regression classifier

$$p(y_i|\mathbf{x}_i, \boldsymbol{\theta}) = \frac{1}{1 + \exp(-y_i \boldsymbol{\theta}^T \mathbf{x}_i)} \tag{5}$$

The fundamental difference between these two is that the logistic-regression classifier predicts y_i using $\mathbf{x}_i$ alone, while the semi-supervised approach considered here predicts y_i by $\mathbf{x}_i$ and the feature vectors in the neighborhood of $\mathbf{x}_i$. The neighborhood of $\mathbf{x}_i$ is formed by all $\mathbf{x}_j$'s that can be reached from $\mathbf{x}_i$ by t-step random walks, with each $\mathbf{x}_j$ contributing to the prediction of y_i in proportion to $a_{ij}^{(t)}$, the probability of walking from $\mathbf{x}_i$ to $\mathbf{x}_j$ in t steps. The role of neighborhoods is then conspicuous — in order for $\mathbf{x}_i$ to be labeled y_i, each neighbor $\mathbf{x}_j$ must be labeled consistently with y_i, in the degree proportional to $a_{ij}^{(t)}$; in such a manner, y_i implicitly propagates over the neighborhood. By taking the neighborhoods into account, it is possible to learn a classifier with only a few labels present and yet the classifier learned is much less subject to over-fitting than when ignoring the neighborhoods. This is addressed in greater detail below.

Let $\mathcal{L} \subseteq \{1, 2, \cdots, N\}$ denote the set of indices of labeled data. Assuming the labels are conditionally independent, we obtain the likelihood function

$$\begin{aligned} p(\{y_i, i \in \mathcal{L}\}|\{\mathcal{N}_t(\mathbf{x}_i) : i \in \mathcal{L}\}, \boldsymbol{\theta}) &= \prod_{i\in\mathcal{L}} p(y_i|\mathcal{N}_t(\mathbf{x}_i), \boldsymbol{\theta}) \\ &= \prod_{i\in\mathcal{L}} \sum_{j=1}^{N} a_{ij}^{(t)}\, p(y_i|\mathbf{x}_j, \boldsymbol{\theta}) \end{aligned} \tag{6}$$

which is the joint probability of observed labels given the t-step neighborhood of each corresponding data point. Estimation of θ may be achieved by maximizing the log-likelihood, which however may yield over-fitting, especially when the number of labeled samples is small. To enforce sparseness of $\boldsymbol{\theta}$ (sparseness has been demonstrated as an important property [17], discouraging overfitting), we impose a zero-mean Gaussian prior on each dimension of $\boldsymbol{\theta}$,

$$p(\boldsymbol{\theta}|\Lambda) = \frac{|\Lambda|^{1/2}}{(2\pi)^{d/2}} \exp(-\frac{1}{2}\boldsymbol{\theta}^t \Lambda \boldsymbol{\theta}) \tag{7}$$

where $\Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_d\}$ are hyper-parameters, d is the dimensionality of $\mathbf{x}$. Each hyper-parameter has an independent Gamma distribution, resulting in

$$\begin{aligned} p(\Lambda|\alpha, \beta) &= \prod_{i=1}^{d} \mathrm{Gamma}(\lambda_i|\alpha_i, \beta_i) \\ &= \prod_{i=1}^{d} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} \lambda_i^{\alpha_i - 1} \exp(-\lambda_i \beta_i) \end{aligned} \tag{8}$$

Marginalizing Λ, we obtain the prior distribution conditional directly on α and β,

$$p(\boldsymbol{\theta}|\alpha, \beta) = \int p(\boldsymbol{\theta}|\Lambda) p(\Lambda|\alpha, \beta)\, d\Lambda \tag{9}$$

The posterior of $\boldsymbol{\theta}$ follows from (6) and (9),

$$p(\boldsymbol{\theta}|\alpha, \beta, \{y_i, \mathcal{N}_t(\mathbf{x}_i) : i \in \mathcal{L}\})$$

$$= Z^{-1} \prod_{i \in \mathcal{L}} \sum_{j=1}^{N} a_{ij}^{(t)} \, p(y_i|\mathbf{x}_j, \boldsymbol{\theta}) \int p(\boldsymbol{\theta}|\Lambda) p(\Lambda|\alpha, \beta) \, d\Lambda \quad (10)$$

where Z is a normalization constant. We are interested in the maximum *a posterior* (MAP) estimate of $\boldsymbol{\theta}$, which maximizes (10) or, equivalently,

$$\begin{aligned} \ell(\boldsymbol{\theta}) &\stackrel{def.}{=} \ln p(\boldsymbol{\theta}|\alpha, \beta, \{y_i, \mathcal{N}_t(\mathbf{x}_i) : i \in \mathcal{L}\}) + \ln Z \\ &= \sum_{i \in \mathcal{L}} \ln \sum_{j=1}^{N} a_{ij}^{(t)} \, p(y_i|\mathbf{x}_j, \boldsymbol{\theta}) \\ &\quad + \ln \int p(\boldsymbol{\theta}|\Lambda) p(\Lambda|\alpha, \beta) \, d\Lambda \end{aligned} \quad (11)$$

The $\boldsymbol{\theta}$ obtained by maximization of $\ell(\boldsymbol{\theta})$ generally is not subject to over-fitting due to two reasons — the neighborhoods incorporated into the first term of $\ell(\boldsymbol{\theta})$ encourages smoothness along the manifold, and the second term of $\ell(\boldsymbol{\theta})$ enforces sparseness of $\boldsymbol{\theta}$.

C. The Learning Algorithm

We maximize (11) by employing an expectation-maximization (EM) algorithm. For any $\{\delta_{ij} : \delta_{ij} \geq 0, \sum_{j=1}^{N} \delta_{ij} = 1\}$ and $\{q(\Lambda) :\in q(\Lambda) d\Lambda = 1\}$, we apply Jensen's inequality to the righthand side of (11) to obtain the lower bound

$$\begin{aligned} \ell(\boldsymbol{\theta}) \geq Q(\boldsymbol{\theta}|\delta, q) &\stackrel{def.}{=} \sum_{i \in \mathcal{L}} \sum_{j=1}^{N} \delta_{ij} \ln \frac{a_{ij}^{(t)} p(y_i|\mathbf{x}_j, \boldsymbol{\theta})}{\delta_{ik}} \\ &\quad + \int q(\Lambda) \ln \frac{p(\boldsymbol{\theta}|\Lambda) p(\Lambda|\alpha, \beta)}{q(\Lambda)} d\Lambda \end{aligned} \quad (12)$$

where the equality holds when

$$\delta_{ij} = \frac{p(y_i|\mathbf{x}_j, \boldsymbol{\theta}) a_{ij}^{(t)}}{\sum_{k=1}^{N} p(y_i|\mathbf{x}_k, \boldsymbol{\theta}) a_{ik}^{(t)}} \quad (13)$$

$$q(\Lambda) = \frac{p(\theta|\Lambda) p(\Lambda|\alpha, \beta)}{\int p(\theta|\Lambda) p(\Lambda|\alpha, \beta) d\Lambda} \quad (14)$$

The EM algorithm consists of iteration of the following two steps.

1) E-step: computing $\{\delta_{ij}\}$ and $q(\boldsymbol{\Lambda})$ using (13) and (14);
2) M-step: compute the re-estimate of $\boldsymbol{\theta}$ as

$$\boldsymbol{\theta} = \arg\max_{\widehat{\boldsymbol{\theta}}} Q(\widehat{\boldsymbol{\theta}}|\delta, q) \quad (15)$$

The convergence is monitored by checking $\ell(\boldsymbol{\theta})$, which is guaranteed to monotonically increase over the EM iterations.

There are two noticeable points regarding the technical details. First, since (8) is conjugate to (7), $q(\Lambda)$ is of the same form as (8) with updated hyper-parameters α, β,

$$\begin{aligned} q(\Lambda) &= \prod_{i=1}^{d} \text{Gamma}(\lambda_i|\alpha_i + \frac{1}{2}, \beta_i + \frac{1}{2}\theta_i^2) \\ &= \prod_{i=1}^{d} \frac{(\beta_i + \frac{1}{2}\theta_i^2)^{\alpha_i + \frac{1}{2}}}{\Gamma(\alpha_i + \frac{1}{2})} \lambda_i^{\alpha_i - \frac{1}{2}} e^{-\lambda_i(\beta_i + \frac{1}{2}\theta_i^2)} \end{aligned} \quad (16)$$

and the integral in the dominator of (14) has an analytic form

$$\begin{aligned} &\int p(\boldsymbol{\theta}|\Lambda) p(\Lambda|\alpha, \beta) d\Lambda \\ &= \frac{1}{(2\pi)^{d/2}} \prod_{i=1}^{d} \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} \frac{\Gamma(\alpha_i + \frac{1}{2})}{\left(\beta_i + \frac{1}{2}\theta_i^2\right)^{\alpha_i + \frac{1}{2}}} \end{aligned} \quad (17)$$

which is useful in checking the convergence of $\ell(\boldsymbol{\theta})$ in (11).

Secondly, in computing $Q(\widehat{\boldsymbol{\theta}}|\delta, q)$ by (12), one needs to compute $\gamma(\widehat{\boldsymbol{\theta}}) \stackrel{def.}{=} \int q(\Lambda) \ln p(\widehat{\boldsymbol{\theta}}|\Lambda) d\Lambda$, and it is found that

$$\begin{aligned} \gamma(\widehat{\boldsymbol{\theta}}) &= -\frac{1}{2} \widehat{\boldsymbol{\theta}}^T \mathbb{E}_q(\Lambda|\theta) \widehat{\boldsymbol{\theta}} \\ &= -\frac{1}{2} \widehat{\boldsymbol{\theta}}^T \text{diag}\left[\mathbb{E}_q(\lambda_1), \mathbb{E}_q(\lambda_2), \cdots, \mathbb{E}_q(\lambda_d)\right] \widehat{\boldsymbol{\theta}} \end{aligned} \quad (18)$$

with

$$\mathbb{E}_q(\lambda_i) = \frac{\alpha_i + \frac{1}{2}}{\beta_i + \frac{1}{2}\theta_i^2}. \quad (19)$$

III. Active Learning

In the UXO-classification problem, it is a given that excavation will ultimately be performed. The principal objective is to excavate as high a percentage of UXO as possible, while leaving as much of the non-UXO as possible unexcavated. Recall that the primary expense in UXO cleanup is the excavation of non-UXO items, since the density of such is typically much higher than the amount of UXO, and the sensor signatures of UXO are often very similar to those of many types of non-UXO. Given that excavation will be performed in any case, one may ask whether the initial set of excavations may be performed with the purpose of improving the performance of the algorithm. Specifically, one may ask which unlabeled sensor signature would be most informative to improved classifier performance if the associated label could be made available. As discussed below, this question is answered in a quantitative information-theoretic manner. When the expected information content of such an excavation drops below a prescribed threshold, excavation for the purpose of improved learning is terminated, and then the algorithm is used to define the probability that all remaining unlabeled signatures correspond to UXO. Importantly, in active learning the algorithm desires to learn about the properties of the UXO *and* non-UXO at the site, and therefore in this phase an excavated non-UXO should not be termed a "false alarm". Such active learning has been performed previously in a related UXO-cleanup study [5]; the distinct character of the algorithm discussed below is that this process is here placed within the context of semi-supervised learning.

A. Active Learning with Semi-Supervised Classifier

For active label selection, we consider a Gaussian approximation of the posterior of the classifier

$$p(\theta|D) \simeq \mathcal{N}(\theta|\hat{\theta}, \mathsf{H}^{-1}) \quad (20)$$

where $\hat{\theta}$ is the estimate of the classifier learned from the above EM algorithm, and H is the posterior precision matrix $\mathsf{H} = \nabla^2(-\log p(\boldsymbol{\theta}|\{y_i, \mathcal{N}_t(\mathbf{x}_i) : i \in \mathcal{L}\})$. By treating $\gamma(\widehat{\boldsymbol{\theta}})$ in (18)

as deterministic, we obtain an evidence-type approximation [17]:

$$\mathbf{H} = \sum_{i\in\mathcal{L}} \sum_{j=1}^{N} \delta_{ij}\, p(y_i|\mathbf{x}_j,\boldsymbol{\theta})(1-p(y_i|\mathbf{x}_j,\boldsymbol{\theta}))\mathbf{x}_j\mathbf{x}_j^T - \nabla^2 \ln\gamma(\widehat{\boldsymbol{\theta}}) \tag{21}$$

With one more data point x_{i*} with label y_{i*} as the next labeled data, assuming that the MAP estimate of $\widehat{\boldsymbol{\theta}}$ remains the same after including the new data point, then the posterior precision changes to

$$\mathbf{H}' = \sum_{i'\in\mathcal{L}\cup\{i*\}} \sum_{j=1}^{N} \delta_{i'j}\, p(y_{i'}|\mathbf{x}_j,\boldsymbol{\theta})(1-p(y_{i'}|\mathbf{x}_j,\boldsymbol{\theta}))\mathbf{x}_j\mathbf{x}_j^T - \nabla^2 \ln\gamma(\widehat{\boldsymbol{\theta}}) \tag{22}$$

For active label selection, we could further simplify the equation for the precision matrix by considering the degenerated connectivity matrix $A^{(t=0)}$, which is an identity matrix, such that

$$\delta_{ij} = \begin{cases} 1, & \text{for } i=j \\ 0, & \text{for } i\neq j \end{cases} \tag{23}$$

Following this, the new precision matrix becomes

$$\mathbf{H}' = \mathbf{H} + p(y_{i*}|\mathbf{x}_{i*},\boldsymbol{\theta})(1-p(y_{i*}|\mathbf{x}_{i*},\boldsymbol{\theta}))\mathbf{x}_{i*}\mathbf{x}_{i*}^T \tag{24}$$

Our criterion for active learning is to choose the feature vector for labeling that maximizes the mutual information between the classifier $\boldsymbol{\theta}$ and the new data point to be labeled, which is the expected decrease of the entropy of $\boldsymbol{\theta}$ after x_{i*} and y_{i*} are observed,

$$\begin{aligned} I &= \frac{1}{2}\log\frac{|\mathbf{H}'|}{|\mathbf{H}|} \\ &= \frac{1}{2}\log\big\{1+p(y_{i*}|\mathbf{x}_{i*},\boldsymbol{\theta})[1-p(y_{i*}|\mathbf{x}_{i*},\boldsymbol{\theta})]\mathbf{x}_{i*}^T H^{-1}\mathbf{x}_{i*}\big\} \end{aligned} \tag{25}$$

The mutual information I is large when $p(y_{i*}|\mathbf{x}_{i*},\boldsymbol{\theta}) \approx 0.5$, therefore, our active learning prefers label acquisition on samples with uncertain classification, based on the current classifier based upon available labeled data. Further, considering the term $\mathbf{x}_{i*}^T H^{-1}\mathbf{x}_{i*}$, the mutual information criterion prefers samples with high variance.

The assumption that the mode of the posterior distribution of the classifier remains unchanged with one more labeled data point is not good at the beginning of the active learning procedure. However, empirically we have found that it is a very good approximation after the active learning procedure has acquired as few as 15 labels, for the examples considered here. Further, this assumption obviates the need to re-train the classifier after each new label is acquired, thus saving computational cost.

B. Other Active-Learning Approaches

When presenting results, we will make comparisons to other semi-supervised learning algorithms, and therefore we briefly discuss how active learning is implemented in these approaches. In the semi-supervised work of [9], the authors also proposed a scheme for active label acquisition, which reduces to the same criterion as (25). However, our classifier is different from theirs, and consequently active learning based on our classifier will yield different results from those produced by the algorithm in [9].

As pointed out in the Introduction, our work was motivated by that in [11], where a similar Markov random walk graph is defined. Instead of training a parametric inductive classifier as in (3), the EM algorithm in [11] learns the classification probability of the unlabeled data directly (transductive). Specifically, the probability of the label for $\mathbf{x}_i$ is defined as

$$p(y_i|\mathcal{N}_t(\mathbf{x}_i)) = \sum_{j=1}^{N} a_{ij}^{(t)}\, p(y_i|\mathbf{x}_j) \tag{26}$$

and the estimation criterion is to maximize the log-likelihood of the labeled data points,

$$\sum_{i\in\mathcal{L}} \log p(\{y_i|\mathcal{N}_t(\mathbf{x}_i)) = \sum_{i\in\mathcal{L}} \log \sum_{j=1}^{N} a_{ij}^{(t)}\, p(y_i|\mathbf{x}_j) \tag{27}$$

which may be performed via an EM method. There is no active learning algorithm provided in [11]. Here we therefore propose an active-learning criterion for the non-parameteric classifier in [11], which selects $\mathbf{x}_{i*}$ to be the next labeled data point by maximizing

$$p(y_{i*}|\mathbf{x}_{i*})(1-p(y_{i*}|\mathbf{x}_{i*})) \tag{28}$$

In this approach we simply acquire a label on that unlabeled sample for which the current classifier is most uncertain; clearly this simple approach may be applied to any classifier.

IV. Application to UXO Detection

A. Magnetometer and EMI Sensor Data Considered

To evaluate the proposed algorithm, we applied it to a UXO data set from an actual former bombing range. This data set was collected by the Multi-sensor Towed Array Detection System (MTADS) [4]. This system is composed of arrays of full-field cesium vapor magnetometers and time-domain electromagnetic pulsed induction sensors. The magnetometers were Geometrics Model 822ROV, while the EMI sensors were highly-modified Geonics EM-61 sensors. The data were collected at a bombing target on the Badlands Bombing Range on the Ogala Sioux Reservation in Pine Ridge, South Dakota. The UXO items present at the site included M 38 (100 lb.) sand-filled practice bombs, M 57 (250 lb.) practice bombs, 2.25 in. and 2.75 in. rocket bodies and rocket warheads, and ordnance scrap (such as tail fins and casing parts). The details of how these measured data are analyzed with magnetometer and EMI dipole models has been described in detail elsewhere [3], and these same techniques were applied to extract feature vectors from the data considered here.

The data associated with a given item under test was manifested in one of three variations: (i) only magnetometer data were available, (ii) only EMI data were available, or (iii) both magnetometer and EMI data were available. These different variations were tied to the details of the data collection, and to the quality of the data acquired for each of the two modalities.

For EMI sensor data alone, there are 230 clutter cases (non-UXOs) and 44 UXOs. For the magnetometer sensor data alone, there are 719 non-UXOs and 79 UXOs. Concerning the case for which data from both the magnetometer and EMI sensors are available, there are 228 Non-UXOs and 44 UXOs. For the EMI data the feature vector is of dimension 10, for the magnetometer the feature vector is of length 9, and when both are used the two types of features are concatenated. Before processing each feature is centered and normalized. Specifically, we compute the mean and variance for each dimension of the features; each feature is shifted by subtracting its mean and then divided by its variance. The feature vectors from this data set are available to other researchers, upon request to the authors.

B. Detection Results for Non-Active Classification: Transductive

In semi-supervised learning, as discussed above, there are two frequently applied settings. In a transductive algorithm [11] it is assumed that all of the labeled and unlabeled data are available simultaneously, and the algorithm is designed to classify the unlabeled data, employing the data-manifold information provided by both the labeled and unlabeled data. Importantly, if a new unlabeled example was added, then the whole transductive learning process would have to begin anew. In an inductive semi-supervised learning algorithm [9] one again has both labeled and unlabeled data with which an algorithm is designed, exploiting the data manifold. Once this algorithm is designed, it may be applied to the existing unlabeled data, as well as to new unlabeled data, without having to redo the learning process.

In many UXO-sensing settings all of the data are collected at once, and therefore a transductive semi-supervised learning algorithm may be sufficient. However, if data is collected incrementally on a large UXO-cleanup site, the inductive framework may be attractive. The semi-supervised algorithm developed here is inductive, but clearly it may be applied in a transductive setting as a special case. However, there are existing semi-supervised algorithms of interest that are only transductive, the algorithm of Szummer & Jaakkola [11] being an important example.

In this subsection we compare our results with performance achieved using [11]. We also make a comparison to results computed using a logistic-regression classifier, where the graph considered here was as a prior to regularize the learning process (imposing smoothness of the classifier along the data manifold [9]). Like our algorithm, the approach in [9] may operate in an inductive setting. However, such that the comparisons are fair, for all examples considered in this section, the unlabeled data on which classification is performed is the same unlabeled data used for semi-supervised algorithm learning (consistent with the requirements of a transductive algorithm). The performance is evaluated in terms of classification accuracy, defined as the ratio of the number of correctly classified UXOs and non-UXOs over the total number of data being used. For this, a threshold 0.5 is used to the classification probability. In the discussion that follows the algorithms considered will be referred to as follows: (i) the method in [11] is denoted RW-Transductive; (ii) the method developed in this paper is termed RW-Inductive; (iii) the method in [9] is termed Logistic-GRF (for Gaussian random field prior); and (iv) the supervised solution is termed Logistic-Regression, with this equivalent to the algorithm in [9] without the graphical prior.

To ensure a fair comparison, we apply the same Markov random walk graph $A^{(t)}$ with $t = 50$ and kernel width $\sigma_i = 1/3 \min_{k=1:N} |\mathbf{x}_i - \mathbf{x}_k|$ for both RW-Transductive and RW-Inductive. Since the graphical prior for Logistic-GRF [9] must be symmetric, we symmetrize the graph by $(A^{(t)} + A^{(t)'})/2$, where $A^{(t)'}$ represents the transpose of $A^{(t)}$. Denote by $\mathcal{X}$ any of the three data sets and $\mathcal{Y}$ the associated label set.

For the results in Figure 1, we randomly sample $\mathcal{X}_L \subset \mathcal{X}$ and assume the associated label set $\mathcal{Y}_L$ are available. The semi-supervised algorithms are trained using $\mathcal{X} \cup \mathcal{Y}_L$ and tested on $\mathcal{X} \setminus \mathcal{X}_L$. The supervised algorithm is trained on $\mathcal{X}_L \cup \mathcal{Y}_L$ and tested on $\mathcal{X} \setminus \mathcal{X}_L$. From Figure 1, we observed that all the semi-supervised algorithms outperform the supervised algorithm. In addition, the RW-Inductive consistently outperforms the Logistic-GRF and is comparable in performance to RW-Transductive. Further, by comparing these three subfigures, we observe that the magnetometer data yields the best detection results on the data considered; this issue will be reconsidered when the labeled data are chosen actively, rather than randomly, as considered in this example.

C. Detection Results for Non-Active Classification: Inductive

In Figure 2 we test the algorithms in an inductive mode, and therefore in this case we only compare RW-Inductive with Logistic-GRF. For this example we randomly sample 200 exmplars $\mathcal{X}_{test} \subset \mathcal{X}$ as testing data, from all the magnetometer data. From the remaining data, we randomly select a subset $\mathcal{X}_L$ of feature vectors, for which the associated labels $\mathcal{Y}_L$ are assumed available, and the remaining feature vectors $\mathcal{X}_U$ are left as unlabeled. The semi-supervised algorithms are trained by using $\mathcal{X}_L \cup \mathcal{Y}_L \cup \mathcal{X}_U$ and tested on $\mathcal{X}_{test}$. In Figure 2 we observe that the RW-Inductive semi-supervised algorithm performs on average superior to Logistic-GRF, although the "error bars" overlap. The length of the error bar is twice the standard deviation of the detection accuracy. Therefore, if the result is Gaussian distributed, 95% of the values lie within the error bar.

D. Detection Results with Active Label Acquisition

In Figures 3 through 5 we consider active learning for the three data sets presented in Figure 1; we first randomly select one UXO and one non-UXO feature vector, and the other labeled data are selected by the active learning algorithm; to design the classifier we require at least one feature vector from each class, but after active learning proceeds sufficiently the large number of labeled examples determined adaptively typically dominate the two labeled examples with which we commence. Compared to Figure 1, we observe that the active learning results are much better than those performed with non-active learning ($i.e.$, random selection of the labeled data),

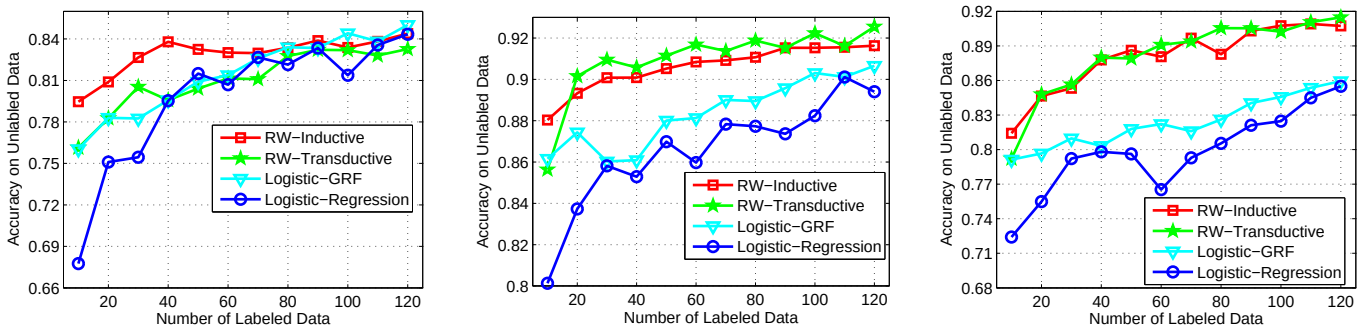


Fig. 1. Comparison of semi-supervised and supervised algorithms for the case in which all unlabeled data are available when algorithm learning is performed. Each curve is an average from 25 independent trials. The horizontal axis is the size of $\mathcal{X}_L$. The algorithms are tested on $\mathcal{X}_U$. From left to right in the sub-figures, the results are for EMI data, magnetometer data and for both EMI and magnetometer data.

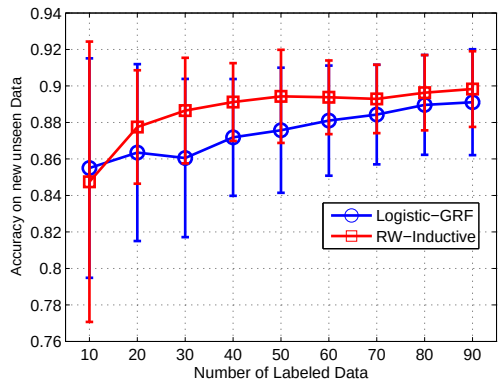


Fig. 2. Semi-supervised results on magnetometer data for the case in which different unlabeled data are used when testing than those used while learning. Each curve is an average from 50 independent trials. The algorithms are tested on 200 data in $\mathcal{X}_{test}$, that are randomly sampled from all magnetometer data $\mathcal{X}$. The horizontal axis is the size of $\mathcal{X}_L$, which is randomly selected from $\mathcal{X} \setminus \mathcal{X}_{test}$. Error bars represent standard deviations.

which demonstrates the effectiveness of the active learning schemes. Although all three semi-supervised active-learning algorithms perform well, the RW-Inductive results appear to be best on average, particularly after a relatively large number of labeled examples are acquired. Note that for a relatively small number of excavations for label acquisition, the magnetometer results are superior but, encouragingly, as the number of labels acquired extends to 120, the fusion of the magnetometer and EMI data yields slightly improved performance relative to either sensor alone.

Further analyzing Figure 3, we observe that with 120 labeled magnetometer examples, these requiring 120 evacuations for the purpose of learning, the RW-Inductive algorithm achieves an accuracy rate as high as 96%, but for the non-active learning, the best algorithm performance for the four methods shown in Figure 1 is only 85%. When both EMI and magnetometer data are considered, from Figure 5, we observe that with 120 labeled data, RW-Inductive achieves an accuracy of 97.5%, but for the non-active learning results in Figure 1, RW-Transductive and RW-Inductive achieve 92% accuracy, and Logistic-GRF and Logistic-Regression only reached 86%.

The results in Figures 3 - 5 present the final classification accuracy for different numbers of labeled data. In practice one will have to employ the information-theoretic measure in (25) to decide when to stop excavating for the purpose of learning, with the acquired labels used to classify all remaining items. In Figure 6 we plot the mutual information of the next item to be labeled via active learning, as a function of the number of items labeled. Results are shown for each of the ten different cases considered to generate the results in Figures 3 - 5. Note that the performance is relatively independent of which two items were considered to constitute the initial labeled UXO and non-UXO feature vectors.

Considering Figure 6, in the next set of examples we terminate the learning phase when the expected gain in mutual information is less than 0.1 (this is an arbitrary setting, but one observes from Figure 6 that this is a point at which the subsequent information gains are relatively small). In Table I we note that with this threshold the algorithm consistently labeled approximately 90 items, out of the possible 272 total items. Interestingly, 19 of the UXO excavated in this active-learning phase were common among all of the ten trials. We also reiterate that the non-UXO excavated in this phase are best not termed "false alarms", since the algorithm desires to learn the properties of both the UXO and non-UXO.

Once the active learning is completed, a final classifier is designed, and this may be applied to the remaining unlabeled data. In Figure 7 we plot a typical receiver operating characteristic (ROC) curve, which corresponds to varying the threshold on the output of the final classifier. We also place a circle at the point on the ROC for which the threshold is set to 0.5; the ROC is computed for all unlabeled data not excavated in the active-learning phase. As indicated, the results in Figure 7 are typical of all of the ten trials considered above, but

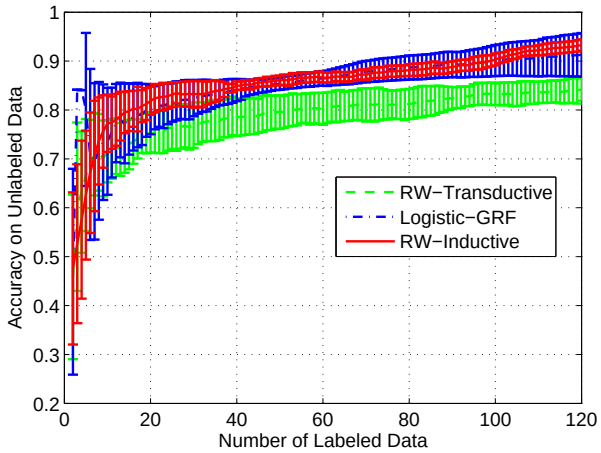


Fig. 3. Active learning results for EMI data. The first two labeled data include one UXO and one Non-UXO, both are sampled randomly. Each curve is an average from 50 independent trials. The horizontal axis is the size of $\mathcal{X}_L$. The algorithms are tested on $\mathcal{X}_U$. Error bars shown are standard deviations.

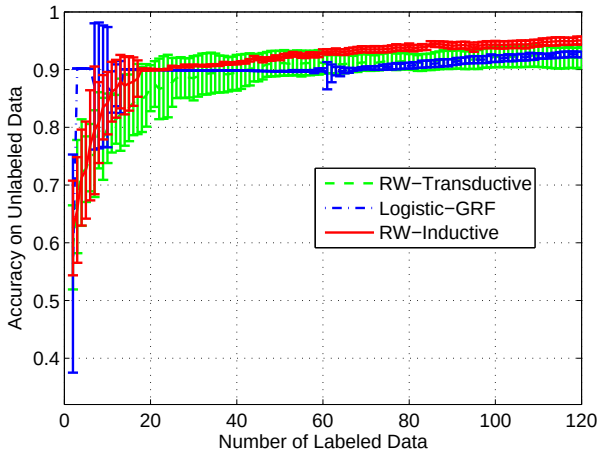


Fig. 4. Active learning results for magnetometer data. The first two labeled data include one UXO and one non-UXO, both are sampled randomly. Each curve is an average from 50 independent trials. The horizontal axis is the size of $\mathcal{X}_L$. The algorithms are tested on $\mathcal{X}_U$. Error bars shown are standard deviations.

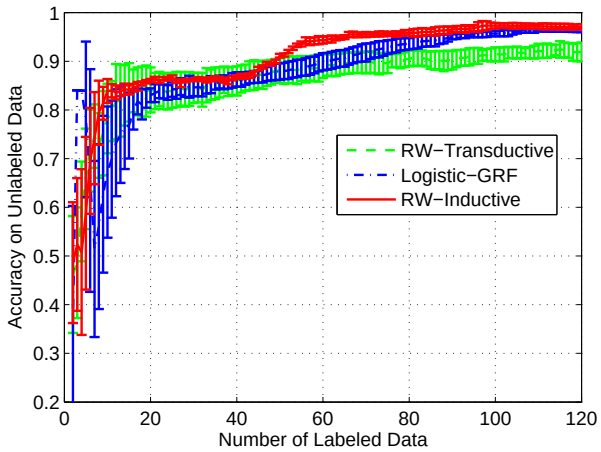


Fig. 5. Active learning results as applied to EMI and magnetometer data. The first two labeled data include one UXO and one non-UXO, both are sampled randomly. Each curve is an average from 50 independent trials. The horizontal axis is the size of $\mathcal{X}_L$. The algorithms are tested on $\mathcal{X}_U$. Error bars shown are standard deviations.

only a single ROC is presented for ease of viewing. Note that the algorithm effectively detects most of the UXO, but the performance saturates around a detection probability of 0.9, and this is because two of the UXOs have features that are very similar to the non-UXO, these constituting challenging targets for classification. The results in Figure 7 correspond to Trail 4 in Table I.

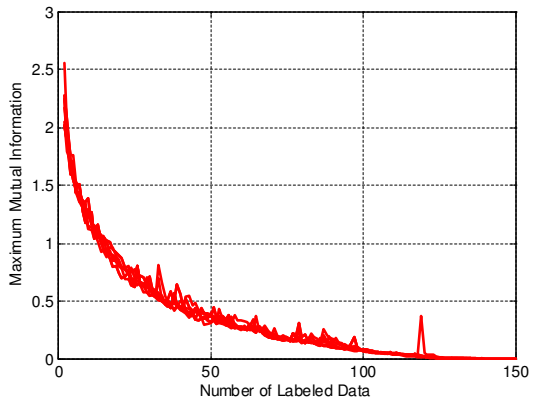


Fig. 6. Active learning mutual information as a function of the next item considered for labeling, as applied to the EMI and magnetometer data. Ten different curves are considered, for different random selection of the initial UXO and non-UXO labeled examples.

Trial	1	2	3	4	5	6	7	8	9	10
Total # of labeled data	89	91	93	95	90	89	89	95	91	93
UXOs	24	24	23	26	25	24	24	25	24	23
non-UXOs	65	67	70	69	65	65	65	70	67	70

TABLE I

SUMMARY OF NUMBER OF ITEMS LABELED FOR EACH OF THE TEN TRIALS IN FIG. 6, WITH THE MUTUAL-INFORMATION-GAIN THRESHOLD SET AT 0.1. LISTED ARE THE NUMBER OF UXO AND NON-UXO EXCAVATED IN THE ACTIVE-LEARNING PHASE.

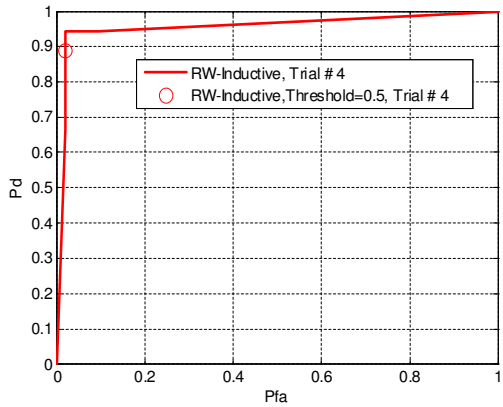


Fig. 7. Receiver operating characteristic for Trial 4 in Table 1. The circle denotes a threshold of 0.5 as applied to the classifier. The labeled data were acquired via active learning, and the results here are as applied to the remaining unlabeled data.

V. CONCLUSIONS

In this paper we have considered the use of semi-supervised learning in the context of UXO detection, based on electromagnetic induction (EMI) and magnetometer data. The algorithms were applied to features extracted from these data, with the features linked to EMI and magnetometer dipole-based parametric models. Semi-supervised learning is particularly well suited to the UXO-sensing problem, because one typically deploys a cart-based system to collect all EMI and magnetometer data at once, for an entire site. Hence, one may perform feature extraction simultaneously on all buried items of interest, and the classification of any one feature vector may be placed within the context of all feature vectors. This contextual information yields information on the characteristics of the data manifold, which has proven useful to improve classification performance in many settings. By contrast, in traditional supervised learning the labeled data alone are employed to learn a classifier, and this classifier is

employed one-by-one to each unlabeled example, in isolation, and consequently contextual information is not employed.

Semi-supervised algorithms typically impose the following condition: two feature vectors that are "close" in feature space should be classified similarly. This implies that the classifier outputs should vary smoothly over the high-density portion of the data manifold, and consequently that the decision boundary in feature space should reside in areas of low data density. These concepts may only be implemented if knowledge of the distribution of all unlabeled data is exploited when performing algorithm learning. The most advanced semi-supervised algorithms developed to date are based on graphical techniques. Specifically, the nodes on the graph correspond to the feature vectors (labeled and unlabeled), and the edge between any two feature vectors is defined by a distance between the two in feature space, where here this is defined by a radial basis function. There are many different ways in which the graph may be employed within a semi-supervised algorithm. For example, one may perform inference directly on the nodes of the graph, thereby inferring labels on the unlabeled nodes. This approach does not generalize to the classification of a general (new) feature vector that is not on the original graph, and therefore if new unlabeled data are acquired, the graph must be reconstituted and learning performed anew. This has been referred to as transductive semi-supervised learning. By contrast, one may also use the graph to learn an "inductive" semi-supervised algorithm, which may be applied to new unlabeled data without having to reconstitute the graph or relearn. In the work presented here we have developed a new inductive semi-supervised algorithm, which extends the transductive algorithm developed in [11]. We have also performed comparisons to another (distinct) inductive semi-supervised algorithm [9], as well as to supervised learning. We have demonstrated that for the measured UXO-sensing data considered here, from an actual UXO cleanup site, that the semi-supervised algorithms perform better than purely supervised learning, implying that there is value in the manifold information associated with UXO sensing, at least for the UXO site considered.

In the UXO-excavation problem, clearly there will be many items manually removed, and the cost of unnecessary excavation of non-UXO items often constitutes the principal cleanup cost. One may therefore ask whether initial excavation may be performed with the purpose of learning. This is termed active learning, and is characterized by asking in an information-theoretic sense which unlabeled feature vectors would be most informative for improving the classifier if the associated label could be acquired (here implemented via targeted excavation). In this sense the algorithm learns adaptively, directly on the site under test. In the examples considered here active learning yielded substantial improvement in UXO-classification performance, relative to selecting the labeled data randomly. One limitation of the active-learning framework, as implemented, is that to commence one needs at least one UXO and one non-UXO labeled example. In practice this is often not a significant limitation, because one typically knows the type of UXO that may be encountered at a given site (from historical information, and also from the items observed on the surface), and an archive of existing labeled UXO data may be used for this target class. Further, since at a typical site the quantity of non-UXO is much larger than the number of UXO, almost any initial excavations will yield at least one non-UXO signature. In the results presented here we examined the sensitivity of the algorithm to the initial UXO and non-UXO labeled exemplars, and found the algorithm to be robust in practice.

For the UXO-sensing data considered, we observed a substantial gain in the performance of the semi-supervised algorithm developed here relative to a corresponding supervised-learning algorithm. However, for the semi-supervised algorithm, the performance of learning using active-learning-determined labeled data was only slightly better to learning with randomly selected labeled data (the latter still using semi-supervised learning). This is a phenomenon we have observed on several different data sets: Since the semi-supervised algorithm exploits the information in the entire data manifold, using labeled and unlabeled data, we have found in practice that it is less sensitive to exactly which labeled data are considered; by contrast, when employing supervised learning the particular labeled data considered is often of significant importance [5].

The most significant direction for future research involves appropriate design of the graph for UXO applications. The weights on the graph edges are adapted to the characteristics of the manifold, via the data-dependent variance in (1). However, in the analysis that followed a t-step walk on the graph was employed, where in the examples considered here t=50. The size of t plays an important role in defining what it means for two feature vectors to be "close" in feature space. It is of interest to develop a principled means of defining an appropriate t for a given data set. We note that the use of t=50 was not carefully tuned for the data considered here, and many similar values ($20 < t < 80$) yielded similar results on the Badlands UXO data. We also note that the need to develop a technique for selecting t is not unique to the RW-Inductive algorithm introduced here, but is of interest for any of the graph-based semi-supervised algorithms.

VI. Acknowledgement

The research reported here was sponsored by the Strategic Environmental Research and Development Program (SERDP). The funding and support of SERDP is greatly appreciated; the views reported here are only those of the authors.

References

[1] N. Geng, C. E. Baum, and L. Carin, "On the low-frequency natural response of conducting and permeable targets," *IEEE Trans. Geoscience and Remote Sensing*, vol. 37, pp. 347–359, 1999.

[2] L. Carin, H. Yu, Y.Dalichaouch, A. R. Perry, P.V.Czipott, and C. Baum, "On the wideband emi response of a rotationally symmetric permeable and counduting targets," *IEEE Trans. Geosience and Remote Sensing*, vol. 39, pp. 1206–1213, 2001.

[3] Y. Zhang, L. M. Collins, H. Yu, C. E. Baum, and L. Carin, "Sensing of unexploded ordance with magnetometer and induction data: Theory and signal processing," *IEEE Trans. Geosience and Remote Sensing*, vol. 41, pp. 1005–1015, 2003.

[4] H. H. Nelson and J. R. McDonald, "Multisensor towed array detection system for uxo detection," *IEEE Trans. Geoscience and Remote Sensing*, vol. 39, p. 11391145, 2001.

[5] Y. Zhang, X. Liao, and L. Carin, "Detection of buried targets via active selection of labeled data: Applications to sensing subsurface uxo," *IEEE Trans. Geosience and Remote Sensing*, vol. 42, pp. 2535–2543, 2004.

[6] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," in *Advances in Neural Information Processing Systems (NIPS)*, 1996.

[7] V. V. Fedorov, *Theory of Optimal Experiments*. New York: Academic Press, 1972.

[8] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*. Cambridge, MA: MIT Press, 2006. [Online]. Available: http://www.kyb.tuebingen.mpg.de/ssl-book

[9] B. Krishnapuram, D. Williams, Y. Xue, A. Hartemink, L. Carin, and M. Figueiredo, "On semi-supervised classification," in *Advances in Neural Information Processing Systems (NIPS)*, 2005.

[10] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *The Twentieth International Conference on Machine Learning (ICML)*, 2003, pp. 912–919.

[11] M. Szummer and T. Jaakkola, "Partially labeled classification with markov random walks," in *Advances in Neural Information Processing Systems (NIPS)*, 2002.

[12] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th International Conf. on Machine Learning (ICML)*. Morgan Kaufmann, San Francisco, CA, 1999, pp. 200–209.

[13] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *The Annual Conference on Learning Theory (COLT)*.

[14] M. Belkin, I. Matveeva, and P. Niyogi, "Regularization and semi-supervised learning on large graphs," in *The Annual Conference on Learning Theory (COLT)*. Springer, 2004.

[15] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine Learning*, vol. 39(2/3), pp. 103–134, 2000.

[16] S. Ganesalingam, "Classification and mixture approaches to clustering via maximum likelihood," *Applied Statistics*, vol. 38, no. 3, pp. 455–466, 1989.

[17] M. Tipping, "Sparse bayesian learning and the relevance vector machines," *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.

Classification of Unexploded Ordnance Using Incomplete Multi-Sensor Multiresolution Data

David Williams, Chunping Wang, Xuejun Liao, and Lawrence Carin

Department of Electrical and Computer Engineering
Duke University
Durham, NC 27708

{dpw,cw36,xjliao,lcarin}@ee.duke.edu

May 2006

Abstract

We address the problem of unexploded ordnance (UXO) detection in which data to be classified are available from multiple sensor modalities and multiple resolutions. Specifically, features are extracted from measured magnetometer and electromagnetic induction (EMI) data; multiple resolution data is manifested when the sensors are separated from the buried targets of interest by different distances (*e.g.,* different sensor-platform heights). The proposed classification algorithm explicitly emphasizes features extracted from fine-resolution imagery over those extracted from less reliable, coarse-resolution data. When fine-resolution features are unavailable (due to undeployed sensors), the algorithm analytically integrates out the missing features via an estimated conditional density function, conditioned on the observed features (from deployed sensors). This density function exploits the statistical relationships that exist among features at different resolutions, as well as those among features from different sensors (in the multi-sensor case). Experimental classification results are shown for real UXO data, on which the proposed algorithm consistently achieves better classification performance than common alternative approaches.

I. Introduction

The problem of unexploded ordnance (UXO) classification continues to receive significant attention in the scientific community [1]–[5]. The objective of a UXO detection (or classification) task is to distinguish buried UXO targets from non-UXO targets (*i.e.,* clutter). To this end, sensors are used to measure data (*e.g.,* magnetic fields) over a two-dimensional grid. These (raw data) sensor measurements can therefore be considered to be in the form of an image. The ultimate classification task is then performed using features extracted from this imagery. This paper addresses the problem of multi-sensor UXO detection when magnetometer and electromagnetic induction (EMI) sensors are employed. In particular, we address the realistic case in which imagery from each of these sensor modalities may be available from multiple *resolutions*. Throughout this paper, the term "resolution" refers to the amount of spatial detail attainable in an image, with this quantity being inversely proportional to the distance between the sensor and the targets (or ground).

Magnetometer and electromagnetic induction (EMI) constitute the principal sensors used in UXO detection and classification [6]–[8]. It is a time-consuming task to deploy these sensors over the large domains that must be interrogated for possible buried UXO. There has therefore been interest in deploying these sensors on helicopters, thereby accelerating the data-collection

process. Although helicopter-borne sensors afford increased collection speed, they also incur the deleterious effect of a loss of signal strength. In particular, for a distance r between the sensor and target, magnetometers and EMI sensors measure field strengths proportional to $1/r^3$ and $1/r^6$, respectively. Therefore, the increased sensor height required by the helicopter manifests a significant loss of signal strength, which undermines the ability to detect small or deeply-buried UXO. In practice, therefore, one may be interested in deploying a helicopter-borne sensor over as large a region as possible, with ground-based sensors applied only locally, over a coarse set of lines running through the site under test.

This paper addresses the problem of classification for data sets in which the features of different data points are extracted from sensor imagery at different resolutions. Additionally, this work considers the more general case in which multiple *sensor modalities* — each of which may operate at multiple resolutions — are employed. In the multi-sensor scenario, incomplete data is manifested when some data points are interrogated by only a subset of the available sensors. Incomplete data also exists in the single-sensor case when not all data points have features extracted from imagery at all resolution levels. Although the classification algorithm introduced in this paper is applicable for data sets that fit the general multi-resolution framework, we focus specifically on the problem of UXO detection. In summary, the novel problem we address in this paper is of multi-sensor, multi-resolution, incomplete-data UXO classification.

It is important to emphasize that this paper addresses a multi-resolution classification problem that has not been examined previously (see [9] for a thorough review of multi-resolution work). In most previous "multi-resolution" image classification work [10], [11], the original imagery actually exists at only a single resolution; the term "multi-resolution" refers simply to a wavelet or other multi-resolution decomposition [12] of the original single resolution imagery. In contrast, this paper utilizes multiple raw images, each at a unique resolution. The ultimate classification objective also distinguishes this work from other multi-resolution image classification work. Most multi-resolution classification work strives for pixel-level classification via image segmentation [13]–[15]. In contrast, in this work, a given image belongs to a single class (UXO or non-UXO).

Several approaches can be employed to handle the missing-data problem in which some data points are characterized by features extracted from only a subset of the possible sensors and/or resolution levels. One approach would build a separate classifier for each type of data. Assuming the set of possible data is relatively small, this approach would be reasonable. The

major drawback with this method, however, is that the dependencies between different types of data are not exploited. In addition to ignoring the correlations between sensors, the severe fragmentation of the data set — based on the combinations of which sensors and resolutions are observed — may leave insufficient data to train each classifier.

A different method would concatenate the features from the various resolutions; incomplete data arising from missing sensors and/or resolutions would be handled in some way, such as by imputation [16]. However, such an approach would treat features obtained from images at different resolutions equally. Intuitively, one should favor using features extracted from high-resolution imagery.

The algorithm proposed in this paper extends the work in [17] — in which missing data is analytically integrated out — to the case of multi-resolution imagery. The algorithm, which does not suffer from any of the drawbacks that plague the aforementioned methods, requires only a single classifier, regardless of the number of sensors or the number of resolutions involved in the problem. Moreover, all data are utilized, so correlations among sensors, as well as among features at different resolutions, are exploited. Additionally, features extracted from different resolutions are not treated equally; rather, fine-resolution features are given more importance. Furthermore, the missing data that exist are handled in a principled manner, avoiding explicit imputation. Specifically, the missing data are integrated out via the use of an estimated conditional density function that relates the dependencies of features both of a single given sensor at different resolutions, as well as of features from different sensors.

The remainder of this paper is organized as follows. Section II explains notation necessary for the proposed classification algorithm introduced in Section III. Section IV describes the UXO model inversion (and feature extraction) processes. Experimental classification results are shown in Section V. Section VI consists of a discussion, followed in Section VII by concluding comments and directions for future work.

II. Notation

Consider the case in which a sensor generates raw data in the form of an image, from which features are extracted subsequently. Assume we possess S such sensors, the s-th of which can operate at $R_s + 1$ resolutions; the resolution is a function of the distance between the sensor and the ground under which the targets are buried. Each of the S sensors may or may not be of the

same modality, and the possible resolutions of each sensor are in general unique. Define Δ_r^s to be the r-th sensor-target separation distance (hereafter, simply "separation distance") of the s-th sensor, for $s = 1, 2, \ldots, S$ and $r = 0, 1, 2, \ldots, R_s$. Let Δ_0^s denote the smallest separation distance of the s-th sensor. The resolution of an image, which is inversely proportional to the separation distance, is written $\mathcal{R}(\cdot)$. The image that results from operating a sensor at its smallest separation distance is referred to as a fine-resolution image. Sensors operating at larger separation distances generate coarse-resolution imagery.

Assume that for a given sensor, the type of features extracted from the raw-image data are fixed, regardless of the separation distance of the sensor that generated the data. That is, for a given sensor, the specific features extracted will be identical for all separation distances, but the feature *values* will in general be unique for each separation distance.

Let $\boldsymbol{x}_i^{(s)} \in \mathbb{R}^{F_s}$ be the F_s features of the s-th sensor for the i-th item (*i.e.,* object, which may be UXO or non-UXO), extracted from data corresponding to the highest resolution image of the s-th sensor. For all larger separation distances, let $\boldsymbol{z}_i^{(s,r)} \in \mathbb{R}^{F_s}$ be the F_s features of the s-th sensor for the i-th item, extracted from the image obtained with the s-th sensor operating at the r-th separation distance. Define $\boldsymbol{x}_i = \left[\boldsymbol{x}_i^{(1)}, \boldsymbol{x}_i^{(2)}, \ldots, \boldsymbol{x}_i^{(S)}\right]$ to be the concatenated feature vectors extracted from imagery at each sensor's respective smallest separation distance. Similarly, define $\boldsymbol{z}_i = \left[\boldsymbol{z}_i^{(1)}, \boldsymbol{z}_i^{(2)}, \ldots, \boldsymbol{z}_i^{(S)}\right]$ to be the concatenated feature vectors extracted from each sensor's coarse-resolution imagery, where $\boldsymbol{z}_i^{(s)} = \left[\boldsymbol{z}_i^{(s,1)}, \boldsymbol{z}_i^{(s,2)}, \ldots, \boldsymbol{z}_i^{(s,R_s)}\right]$. Hereafter, we shall refer to $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$ as *primary* and *auxiliary* features (or data), respectively.

The data can alternatively be partitioned in terms of its observed and missing components. Let o_i^x be the set of sensors for which the i-th item's primary features are observed. Let m_i^x be the (complementary) set of sensors for which the primary features are missing for the i-th item. Similarly, let o_i^z be the set of sensor and separation-distance pairs for which the auxiliary features for the i-th item are observed. Let m_i^z be the (complementary) set of sensor and separation-distance pairs for which the auxiliary features for the i-th item are missing. To simplify notation, we shall suppress the superscripts when doing so will not cause confusion (*e.g.,* $\boldsymbol{x}_i^{o_i^x}$, the primary features (from all sensors) that are observed for the i-th item, will be written as $\boldsymbol{x}_i^{o_i}$). The primary and auxiliary data of the i-th item can thus be written as $\boldsymbol{x}_i = [\boldsymbol{x}_i^{o_i}; \boldsymbol{x}_i^{m_i}]$ and $\boldsymbol{z}_i = [\boldsymbol{z}_i^{o_i}; \boldsymbol{z}_i^{m_i}]$, respectively.

Data for a given item is deemed to be *complete* if we possess all primary features, for all

sensors, for that data point (*i.e.,* $m_i^x = \emptyset$). A data point is otherwise deemed *incomplete*. It should be noted that there exist two different types of incomplete data. First, data would be incomplete if some subset of sensors were never deployed (at any separation distance) for the corresponding item (UXO or non-UXO). Data could also be incomplete even when all sensors were deployed for the item; specifically, the data would still be considered incomplete in this case if the data had not been interrogated at the smallest separation distance of every sensor.

III. Classification with Incomplete Data

Assume we have a set of labeled (incomplete) data

$$\mathcal{D}_L = \{\boldsymbol{x}_i, \boldsymbol{z}_i, y_i, \epsilon_i, o_i^x, o_i^z, m_i^x, m_i^z\}_{i=1}^{N_L} \tag{1}$$

where $y_i \in \{-1, 1\}$ is the label (indicating non-UXO or UXO, respectively) of the i-th item, and $\epsilon_i \in [0, 0.5)$ is the corresponding labeling error rate. The labeling error rate is simply the probability that a true label was flipped (corrupted) to the wrong label (*e.g.,* $\{y_i^{\text{true}} = 1\} \rightarrow \{y_i = -1\}$). Such imperfect labels can be manifested when a human analyst performs the labeling without excavating the buried object.

Let $\boldsymbol{w}_s = \left[w_s^{(1)}, w_s^{(2)}, \ldots, w_s^{(F_s)}\right]$ represent the F_s weights of a classifier on the primary features of the s-th sensor. Let $\boldsymbol{w} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_S]$ be the classifier weights on the primary features of each sensor (*i.e.,* $\boldsymbol{x}_i$). Note that the number of features from each sensor need not be identical. It must be emphasized that the weights — and hence the resulting classifier — are on the features extracted from *only* the fine-resolution imagery. We emphasize this caveat by using different notation for primary features extracted from the fine-resolution imagery ($\boldsymbol{x}_i$) and auxiliary features extracted from coarse-resolution imagery ($\boldsymbol{z}_i$).

In logistic regression (with a hyperplane classifier), the probability of label y_i given feature vector $\boldsymbol{x}_i$ is $p\left(y_i|\boldsymbol{x}_i, \boldsymbol{w}\right) = \sigma(y_i \boldsymbol{w}^T \boldsymbol{x}_i)$, where $\sigma(\eta) = (1 + \exp(-\eta))^{-1}$ is the sigmoid link function and $\boldsymbol{w}$ constitutes a classifier. Accounting for imperfections in the labeling process arising from a known labeling error rate ϵ_i, the probability of label y_i given $\boldsymbol{x}_i$ and ϵ_i is [18]

$$p\left(y_i|\boldsymbol{x}_i, \epsilon_i, \boldsymbol{w}\right) = \epsilon_i + (1 - 2\epsilon_i)\sigma(y_i \boldsymbol{w}^T \boldsymbol{x}_i). \tag{2}$$

Note that the standard case of perfect labels is recovered when $\epsilon_i = 0$. If all features are extracted from complete data, the weights of the classifier can be learned easily by maximizing

the likelihood of the data. Here we consider the case in which the data are in general *incomplete* in the sense described above.

Recall that the classifier is to be designed for only the primary data — the features extracted from the finest-resolution imagery. We first partition $\boldsymbol{x}_i$ into its observed and missing parts, $\boldsymbol{x}_i = [\boldsymbol{x}_i^{o_i}; \boldsymbol{x}_i^{m_i}]$, and then apply the same partition to $\boldsymbol{w}$ to obtain $\boldsymbol{w} = [\boldsymbol{w}_{o_i}; \boldsymbol{w}_{m_i}]$. With $\eta_i = \boldsymbol{w}_{m_i}^T \boldsymbol{x}_i^{m_i}$, (2) can be written as

$$p\left(y_i | \boldsymbol{x}_i^{o_i}, \epsilon_i, \boldsymbol{w}\right) = \epsilon_i + (1 - 2\epsilon_i)\sigma(y_i(\boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i} + \eta_i)). \tag{3}$$

If the missing data $\boldsymbol{x}_i^{m_i}$ is integrated out, the needed probability of y_i given *all* observed features can be written as

$$\begin{aligned} p\left(y_i | \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}, \epsilon_i, \boldsymbol{w}\right) &= \int p\left(y_i | \boldsymbol{x}_i^{o_i}, \epsilon_i, \boldsymbol{w}\right) p\left(\boldsymbol{x}_i^{m_i} | \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}\right) d\boldsymbol{x}_i^{m_i} & (4) \\ &= \epsilon_i + (1 - 2\epsilon_i) \int \sigma(y_i(\boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i} + \eta_i)) p\left(\eta_i | \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}\right) d\eta_i. & (5) \end{aligned}$$

Although the classifier uses only the primary data, the auxiliary data *is* exploited when primary data is missing, via the conditional density function $p\left(\boldsymbol{x}_i^{m_i} | \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}\right)$. That is, when primary data is available, it is utilized; when primary data is unavailable, the auxiliary data becomes relevant and is exploited.

The integration in (5) can be performed analytically by making two mild assumptions. First, we assume that $p\left(\boldsymbol{x}_i, \boldsymbol{z}_i\right)$ is a Gaussian mixture model (GMM), which can accurately model many reasonably well-behaved distributions. This density function describes the relationships among the same features obtained from different resolutions of a given sensor; it also describes the relationships among features from different sensor modalities. It then follows that

$$p(\boldsymbol{x}_i, \boldsymbol{z}_i) = p(\boldsymbol{x}_i^{m_i}, \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}) p(\boldsymbol{z}_i^{m_i} | \boldsymbol{x}_i^{m_i}, \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}), \tag{6}$$

where $p(\boldsymbol{x}_i^{m_i}, \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i})$ is also necessarily a GMM. Introducing the notation $\boldsymbol{\chi}_i^{o_i} = [\boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}]$, this ($K$-component) GMM is

$$p(\boldsymbol{x}_i^{m_i}, \boldsymbol{\chi}_i^{o_i}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(\begin{bmatrix} \boldsymbol{x}_i^{m_i} \\ \boldsymbol{\chi}_i^{o_i} \end{bmatrix}; \begin{bmatrix} \boldsymbol{\mu}_k^{m_i} \\ \boldsymbol{\mu}_k^{o_i} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_k^{m_i m_i} & \boldsymbol{\Sigma}_k^{m_i o_i} \\ \boldsymbol{\Sigma}_k^{o_i m_i} & \boldsymbol{\Sigma}_k^{o_i o_i} \end{bmatrix} \right), \tag{7}$$

where π_k are the non-negative mixing proportions that sum to unity. Moreover, $p(\boldsymbol{x}_i^{m_i} | \boldsymbol{\chi}_i^{o_i})$ is a GMM as well. Because of the linear relation $\eta_i = \boldsymbol{w}_{m_i}^T \boldsymbol{x}_i^{m_i}$, $p\left(\eta_i | \boldsymbol{\chi}_i^{o_i}\right)$ is also a GMM,

$$p\left(\eta_i | \boldsymbol{\chi}_i^{o_i}\right) = \sum_{k=1}^{K} \delta_k^i \mathcal{G}\left(\frac{\eta_i - \boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^{m_i}}{\sqrt{\boldsymbol{w}_{m_i}^T \boldsymbol{\Omega}_k^{m_i} \boldsymbol{w}_{m_i}}} \right), \tag{8}$$

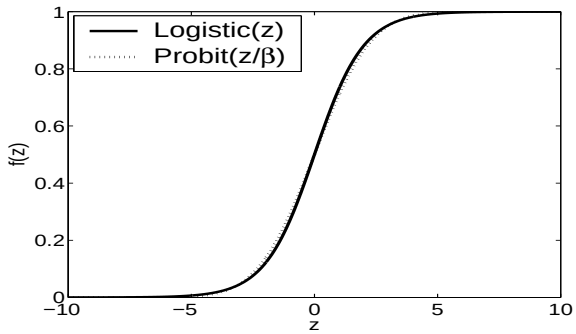


Fig. 1. Illustration of the accuracy of the approximation made between the logistic function and the (scaled) probit function.

with the parameters

$$\delta_k^i = \frac{\pi_k \mathcal{N}\left(\boldsymbol{\chi}_i^{o_i}; \boldsymbol{\mu}_k^{o_i}, \boldsymbol{\Sigma}_k^{o_i o_i}\right)}{\sum_{\ell=1}^{K} \pi_\ell \mathcal{N}\left(\boldsymbol{\chi}_i^{o_i}; \boldsymbol{\mu}_\ell^{o_i}, \boldsymbol{\Sigma}_\ell^{o_i o_i}\right)} \tag{9}$$

$$\boldsymbol{\xi}_k^{m_i} = \boldsymbol{\mu}_k^{m_i} + \boldsymbol{\Sigma}_k^{m_i o_i} (\boldsymbol{\Sigma}_k^{o_i o_i})^{-1} (\boldsymbol{\chi}_i^{o_i} - \boldsymbol{\mu}_k^{o_i}) \tag{10}$$

$$\boldsymbol{\Omega}_k^{m_i} = \boldsymbol{\Sigma}_k^{m_i m_i} - \boldsymbol{\Sigma}_k^{m_i o_i} (\boldsymbol{\Sigma}_k^{o_i o_i})^{-1} \boldsymbol{\Sigma}_k^{o_i m_i} \tag{11}$$

where $\mathcal{G}(\eta_i) = (2\pi)^{-1/2} \exp\left\{-\eta_i^2/2\right\}$ is the standard univariate Gaussian density function with zero mean and unit variance. The requisite GMM density function estimation can be accurately performed using all available data, via the Variational Bayesian Expectation-Maximization algorithm presented in [17].

The second (very accurate) assumption is that the sigmoid function can be approximated as a probit function (*i.e.,* a Gaussian cumulative distribution function)

$$\sigma(\alpha) \approx \int_{-\infty}^{\alpha} \mathcal{G}\left(\frac{u}{\beta}\right) du \tag{12}$$

where $\beta = \frac{\pi}{\sqrt{3}}$. The accuracy of this approximation is shown in Figure 1.

Mirroring the derivation in [17], it can then be shown that the integral in (5) can be computed analytically. The result of this integration is that the probability of y_i given only the observed portions of $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$ can be expressed as a mixture of *sigmoids*:

$$p\left(y_i | \boldsymbol{x}_i^{o_i}, \boldsymbol{z}_i^{o_i}, \epsilon_i, \boldsymbol{w}\right) \approx \epsilon_i + (1 - 2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \sigma\left(\frac{y_i \beta (\boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^{m_i} + \boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i})}{\sqrt{\boldsymbol{w}_{m_i}^T \boldsymbol{\Omega}_k^{m_i} \boldsymbol{w}_{m_i} + \beta^2}}\right). \tag{13}$$

The log-likelihood function of the incomplete data in (1) is then

$$\ell(\boldsymbol{w}) = \log p\left(\{y_i\}_{i=1}^{N_L} | \{\boldsymbol{x}_i^{o_i}\}_{i=1}^{N_L}, \{\boldsymbol{z}_i^{o_i}\}_{i=1}^{N_L}, \{\epsilon_i\}_{i=1}^{N_L}, \boldsymbol{w}\right) \tag{14}$$

$$\approx \sum_{i=1}^{N_L} \log\left[\epsilon_i + (1-2\epsilon_i)\sum_{k=1}^{K} \delta_k^i \sigma\left(\frac{y_i \beta(\boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^{m_i} + \boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i})}{\sqrt{\boldsymbol{w}_{m_i}^T \boldsymbol{\Omega}_k^{m_i} \boldsymbol{w}_{m_i} + \beta^2}}\right)\right]. \tag{15}$$

The objective function (15) to be maximized is no longer concave for two reasons. First, the concavity is destroyed by the imperfect labels resulting from ϵ_i. Even in the case of perfect labels though, (15) is not concave because of the particular form of the argument of the sigmoid function, arising from the incomplete data. Since (15) is not concave, an intelligent initialization of $\boldsymbol{w}$ is valuable for avoiding local maxima. We therefore initialize $\boldsymbol{w}$ as follows. We "complete" the data set by replacing the missing features $\boldsymbol{x}_i^{m_i}$ with the conditional mean $\mathbb{E}[\boldsymbol{x}_i^{m_i}|\boldsymbol{x}_i^{o_i}] = \sum_{k=1}^{K} \delta_k^i \boldsymbol{\xi}_k^{m_i}$, where δ_k^i and $\boldsymbol{\xi}_k^{m_i}$ are defined in (9) and (10), respectively. For the initialization, we also treat all labels as perfect, artificially setting all $\epsilon_i = 0$. This "completed," "perfectly" labeled data set is submitted to the standard logistic regression [19] to obtain $\boldsymbol{w}_0$, which is then used as the initialization of $\boldsymbol{w}$ in maximizing (15) by a modified form of gradient ascent (additional details are shown in the Appendix). Empirical evidence [20] suggests that this initialization successfully avoids most local maxima.

Thus, the maximum-likelihood (ML) logistic regression classifier $\boldsymbol{w}$ can then be obtained, in spite of the missing data (and imperfect labels). Thereafter, the class predictions of an unlabeled testing data point with incomplete (missing) features is computed trivially using (13) (with $\epsilon_i = 0$ since no actual labeling will have transpired).

IV. Multi-Sensor Multiresolution UXO Data

The proposed classification algorithm is designed for data sets consisting of imagery that exists at multiple resolutions for multiple sensors, which is a realistic scenario for UXO detection tasks. Here we consider the case in which we possess two modalities, magnetometer and EMI. Moreover, it is assumed that each sensor can operate at two different image resolution levels. The image resolution is a function of the distance between the sensor and the (buried) target. Therefore, these unique image resolutions are manifested by deploying a given sensor on different platforms (at different heights).

Coarse-resolution imagery is generated by a given sensor when a relatively large distance separates the sensor from the targets; here, this situation corresponds to the case in which the

sensor collects data while located on a low-altitude airborne platform (*e.g.,* a helicopter) that flies above the area of interest. In contrast, fine-resolution imagery is generated by a given sensor when a smaller distance separates the sensor from the targets; here, this situation corresponds to the case in which the sensor collects data while located on a ground-based platform. Because the magnetometer and EMI sensor may not be located on the same platform, each sensor may interrogate unique areas of land that overlap only partially. As a result, some targets may be characterized by imagery from only one of the sensors, which is a case of incomplete data.

A. Feature Extraction Models

The magnetometer and EMI sensor data used in this study are magnetic field measurements as a function of spatial position. The data from each sensor can therefore be considered to be in the format of an image. Features are extracted from this imagery and then subsequently used in the classification stage discussed in Section III. The features we use here are the parameters of UXO models developed in [8] that are fit via a model inversion process. Specifically, the measured (image) data is the input to the inversion, and the model parameters (features) are the output of the inversion. We subsequently employ these fitted model parameters as the features of the classification stage. As a result, regardless of the resolution of the image, the same features (parameters) are extracted. Although the features are identical, the actual *values* of these features extracted from images at different resolutions (for any given data point) will be unique.

1) Magnetometer Model: Ferrous objects cause changes in the observed background magnetic field of the earth; magnetometers sense these changes. It has been shown that the spatially dependent magnetometer signal is well-modeled by a simple magnetic dipole [21]. The success of this magnetic-dipole model for sensing buried UXO [6], [7], [22], [23] motivates us to employ the model for the measured spatially dependent magnetometer data here.

In the $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{z}$ coordinate system of the sensor, let the z-direction be normal to the air-soil interface. Let the position vectors of the sensor (*i.e.,* observation point) and target-dipole be $\boldsymbol{r}_s = \begin{bmatrix} x_s & y_s & z_s \end{bmatrix}$ and $\boldsymbol{r}_t = \begin{bmatrix} x_t & y_t & z_t \end{bmatrix}$, respectively. Define $\boldsymbol{r}_{ts} = \boldsymbol{r}_s - \boldsymbol{r}_t$ to be the vector directed from the dipole to the sensor, with $\boldsymbol{r} = \frac{\boldsymbol{r}_{ts}}{|\boldsymbol{r}_{ts}|}$ the corresponding unit vector. It should also be noted that the orientation of the magnetic-dipole — completely summarized by the angles θ and ϕ — is different from the direction of the ordnance itself.

When the sensor is sufficiently distant from the buried target relative to the target dimensions, the (vector) magnetic field may be represented approximately as [8], [24]

$$\mathbf{H} = \frac{1}{2\pi}\frac{\boldsymbol{m}\cdot\boldsymbol{r}}{|\boldsymbol{r}_{ts}|^3}, \tag{16}$$

where $\boldsymbol{m}$ is the magnetic-dipole moment. The magnetometer employed to collect the data used in this work measures the z-component of the magnetic field as a function of position on the surface. This measurement is subsequently fit to the model in (16) via a simple gradient search. Specifically, the parameters that the model inversion fits are the target position (x, y, and depth z), the magnetic-dipole strength, ($m = |\boldsymbol{m}|$), and the magnetic-dipole orientation (θ and ϕ). We retain the last four parameters (z, m, θ, and ϕ) of the model as features for the classification stage.

2) EMI Model: A model for the EMI response of targets that generalizes the magnetometer model via a frequency-dependent magnetic dipole has been developed in [8]. Specifically, the magnetic-dipole moment $\boldsymbol{m}$ of a target is represented as

$$\boldsymbol{m} = \mathbf{M}\mathbf{H}^{\text{inc}} \tag{17}$$

where $\mathbf{H}^{\text{inc}}$ denotes the incident (excitation) magnetic field, and $\mathbf{M}$ is the magnetization tensor that relates the magnetic field to the magnetic-dipole moment. For a UXO assumed to be rotationally symmetric with the axis of rotation along the z direction, the (frequency-dependent) magnetization tensor can be expressed as a diagonal matrix [22]

$$\mathbf{M} = \text{diag}\left[\ m_{p0} + \textstyle\sum_i \frac{\omega m_{pi}}{\omega - j\omega_{pi}},\quad m_{p0} + \sum_i \frac{\omega m_{pi}}{\omega - j\omega_{pi}},\quad m_{z0} + \sum_k \frac{\omega m_{zk}}{\omega - j\omega_{zk}}\ \right]. \tag{18}$$

The terms m_{z0} and m_{p0} correspond to the zero-frequency magnetic-dipole moments of the target, directed perpendicular to and along the target's axis of rotation, respectively. The terms m_{zk} and m_{pi} in (18) account for the frequency-dependent character of the response, while ω_{zk} and ω_{pi} correspond to EMI resonant frequencies. Because higher order dipole moments in the summations in (18) typically lack significant strength [25], here we use only the first term in each summation, which is representative of the principal dipole mode along each of the principal axes.

If it is assumed that the EMI source responsible for the excitation magnetic field $\mathbf{H}^{\text{inc}}$ can be represented — as seen from the target — as a magnetic dipole with moment $\boldsymbol{m}_s$, then [8]

$$\mathbf{H}^{\text{inc}} = \boldsymbol{r}\frac{1}{2\pi}\frac{\boldsymbol{m}_s\cdot\boldsymbol{r}}{|\boldsymbol{r}_{st}|^3}, \tag{19}$$

where $\boldsymbol{r}_{st}$ is the vector directed from the source to the target center, with $\boldsymbol{r} = \frac{\boldsymbol{r}_{st}}{|\boldsymbol{r}_{st}|}$ the corresponding unit vector. Assuming sufficient proximity of the sensor's source and receiver coils, the total (frequency-dependent) magnetic field observed at the sensor will be [8]

$$\mathbf{H}^{\text{rec}} \propto \frac{\boldsymbol{r}}{|\boldsymbol{r}_{st}|^6} \boldsymbol{r}^T \mathbf{U}^T \mathbf{M} \mathbf{U} \boldsymbol{r}, \tag{20}$$

where the proportionality constant depends on the strength of the dipole source $\boldsymbol{m}_s$ and the characteristics of the receiver.

The 3×3 unitary rotation matrix $\mathbf{U}$ rotates the fields from the coordinate system of the sensor to the coordinate system of the target, and $\mathbf{U}^T$ transforms the dipole fields of the target (in the $\mathbf{M}$ coordinate system) back to the coordinate system of the sensor. Explicitly, the target orientation, in terms of the angles of the target θ and ϕ with respect to the sensor coordinate system, is accounted for by

$$\mathbf{U} = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{21}$$

As with the magnetometer, the EMI sensor employed in this work measures the z-component of the magnetic field as a function of position on the surface. This measurement is subsequently fit to the model in (20) via a form of the Levenberg-Marquardt method [26]. Specifically, the parameters that the model inversion fits are the target position (x, y, and depth z), the target orientation (θ and ϕ), the magnetic dipole strengths (m_{z0}, m_{p0}, m_{zk}, and m_{pi}), and the EMI resonant frequencies (ω_{zk} and ω_{pi}). We retain five parameters (z, θ, ϕ, m_{z0}, and m_{p0}) of the model as features for the classification stage.

In fitting the more sophisticated EMI model, parameters from the magnetometer inversion are used to constrain the search of some of the EMI model parameters. Specifically, the depth (z) and cross-sectional position (x and y) of the target found by the magnetometer inversion are used to initialize the target location in the EMI inversion. This initialization helps avoid some local maxima in the inversion process. To overcome other local maxima, several (model-fitting) solutions are obtained, with each solution resulting from randomly initializing the remaining parameters of the model. The final parameters of the model are taken to be those of the solution that minimizes the mean-square error between the measured and model-fit data.

B. Simulation of Multiresolution Imagery

We possess measured magnetometer and EMI (image) data measured by ground-based sensors; we simulate multi-resolution imagery from the available single-resolution imagery in the following manner. The image-simulation process for the two sensors is identical, so here we explain the process in terms of the magnetometer sensor. During the explanation, we shall reference Figure 2, which illustrates the various stages of the process for one example target.

We begin with ground-based magnetometer data measurements for a target (Figure 2(A)). The magnetometer model inversion explained in Section IV is performed, which provides model parameters. These model parameters are then assumed to be the true model parameters of the target. Using these model parameters, ground-based data can be synthesized (Figure 2(B)). If the model fitting was successful, the measured and synthesized data should be nearly identical. Using these same model parameters, one can instead synthesize coarse-resolution (helicopter-based) data (Figure 2(C)) by increasing the value representing the distance between the sensor and the ground. The distance from the ground-based sensors to the ground surface is $0.3\,\mathrm{m}$, while the distance from the helicopter-based sensors to the ground surface is assumed to be $5.0\,\mathrm{m}$.

As stated until now, this synthesis procedure would be unrealistic because the sensor noise of the helicopter-based sensor should be higher than that of the ground-based sensor. To reflect this fact, white Gaussian noise ($\mathcal{N}(0, \sigma^2)$, where here $\sigma = 1$) — representing sensor noise — is added to this synthesized data to produce noisy data (Figure 2(D)). This noisy helicopter-based sensor data is then taken to be the "raw" coarse-resolution sensor measurements, analogous to the raw ground-based sensor data from which the original model parameters were obtained. This noisy data is subsequently used to obtain coarse-resolution features via the magnetometer model inversion. It is important to reiterate that the particular features (but not the feature values) extracted from any image from a given sensor will be identical, regardless of the image's resolution.

It should be noted that the amplitude of the response in Figure 2(A) and 2(B) is much larger than that in Figure 2(C) and 2(D) because the response is proportional to $1/r_{ts}^3$ where r_{ts} is the distance between the target and the sensor (see (16)). Also note that the (physical) area shown in Figures 2(A) and 2(B) is $3\,\mathrm{m} \times 3\,\mathrm{m}$, while the area in Figures 2(C) and 2(D) is $8\,\mathrm{m} \times 8\,\mathrm{m}$;

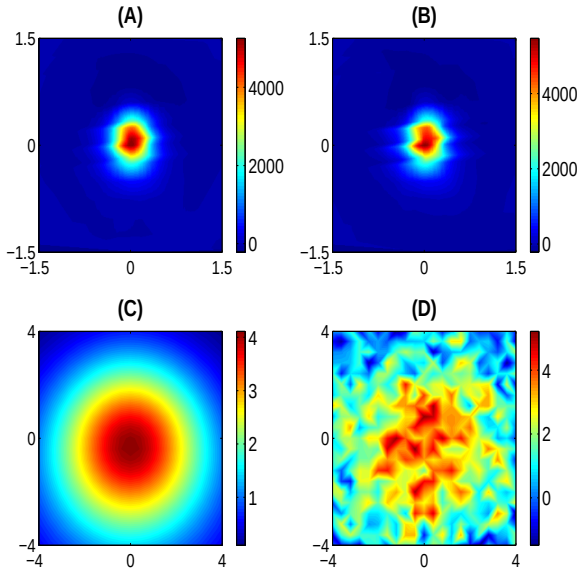


Fig. 2. Process of simulating coarse-resolution (helicopter-based) magnetometer data. (A) Measured fine-resolution (ground-based) magnetometer data. (B) Synthesized fine-resolution (ground-based) magnetometer data using the model parameters obtained from the model inversion with the data in (A). (C) Synthesized noise-free coarse-resolution (helicopter-based) magnetometer data using the model parameters obtained from the model inversion and the data in (A). (D) Same as (C) except with (sensor) noise added.

this larger area must be considered in order to ensure that the full response is captured, for the response expands spatially as the sensor-target distance increases.

V. Experimental Results

To evaluate the proposed incomplete-data classification algorithm, we applied it to a UXO data set consisting of 166 items, 41 of which are UXO. This data set was collected by the Multi-sensor Towed Array Detection System (MTADS) [27]. This system is composed of arrays of full-field cesium vapor magnetometers and time-domain electromagnetic pulsed induction sensors. The magnetometers were Geometrics Model 822ROV, while the EMI sensors were highly-modified Geonics EM-61 sensors. The data was collected at a bombing target on the Badlands Bombing Range on the Ogala Sioux Reservation in Pine Ridge, South Dakota. The UXO items present at the site included M 38 (100 lb.) sand-filled practice bombs, M 57 (250 lb.) practice bombs, 2.25 in. and 2.75 in. rocket bodies and rocket warheads, and ordnance scrap (such as tail fins and casing parts).

For every item, we possess both (measured) fine-resolution and (synthesized) coarse-resolution features from each of the two sensors (a magnetometer and an EMI sensor). As mentioned earlier, four magnetometer features are used to characterize the magnetometer data at each resolution level, while five EMI features are used to characterize the EMI data at each resolution level.

In all experiments, it is assumed that coarse-resolution (helicopter-based sensor) data is available from both sensors for all data points. This choice is motivated by the fact that it would be relatively quick, easy, and inexpensive to acquire such data vis-à-vis ground-based sensor data. In contrast, it is assumed that fine-resolution sensor data will be missing for some data points, with these amounts made specific later. It should be noted, however, that the proposed algorithm can function successfully even when data points are missing data from a given sensor completely (*i.e.,* at all resolutions).

Because data is available from two different sensors, many different combinations of missing data are possible. To conduct an extensive investigation of the proposed algorithm, 36 different combinations of missing data are considered (explained in more detail below). The binary Cartesian product of a set S is the set of ordered pairs

$$S \times S = \{(\alpha, \beta) \,|\, \alpha \in S \text{ and } \beta \in S\}. \tag{22}$$

In (22), let α and β be the fraction of data points that are missing fine-resolution magnetometer features and fine-resolution EMI features, respectively. We conduct experiments using the elements of the binary Cartesian product of the set $S = \{0, 0.1, 0.25, 0.5, 0.75, 0.9\}$ as the pairs of amounts of missing primary (fine-resolution) data. For each of the 36 combinations considered, 100 independent trials are run. Each trial has a random partition of the data set into training and testing data, and randomly selected data points that are assumed to be missing the primary data. Note that primary data will be missing for both training and testing data.

This experimental set-up was employed for three different amounts of training data: when 25%, 50%, and 75% of the data was labeled training data. All classification results shown are for the remaining unlabeled testing data. In all experiments, it was assumed that there was no labeling error ($\epsilon = 0$). In each experiment, four algorithms are applied, each of which handles the multi-resolution data in a different manner. However, a logistic regression classifier is used in all four methods, which are explained below.

The proposed approach builds a classifier for only the primary data; it handles missing primary

data by integrating out the missing data, using the estimated density function relating both the primary and auxiliary data. This density function — a GMM — is accurately estimated using all available data, via the Variational Bayesian Expectation-Maximization algorithm presented in [17]. Because class labels are not used in the estimation, both labeled and unlabeled data can be utilized. This fact ensures that the density function can be accurately estimated even when limited (labeled) training data is available.

The second method builds a separate classifier for data from each resolution. Building separate classifiers for data from each resolution in the case of a single sensor with two resolutions would entail that one classifier be built for features extracted from fine-resolution imagery, and a second classifier be built for features extracted from coarse-resolution imagery. The generalization of this case to multiple sensors with multiple resolutions is employed here as the second method. Specifically, four separate classifiers are constructed, one to handle each sensor-resolution pair combination. A more detailed explanation of this method is provided in the Appendix.

The third method builds a classifier for the concatenated primary and auxiliary data; it handles missing primary data by integrating out the missing data, via the approach used in [17]. The difference between this method and the proposed method is that this method builds a classifier for both auxiliary and primary data, whereas the proposed method does so only on the latter. The fourth method also builds a classifier for the concatenated primary and auxiliary data; however, this method imputes (*i.e.,* "fills in") missing primary data with the unconditional mean of the observed data.

The area under a receiver operating characteristic (ROC) curve (AUC) is given by the Wilcoxon statistic [28]

$$\text{AUC} = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \mathbf{1}_{x_m > y_n} \tag{23}$$

where $x_1, \ldots, x_M$ are the classifier outputs of data belonging to class 1, $y_1, \ldots, y_N$ are the classifier outputs of data belonging to class -1, and $\mathbf{1}$ is an indicator function. As a measure of classification performance, the AUC is a more useful quantity than accuracy (*i.e.,* the fraction of classifications that are correct) when significant class imbalance exists, as it does in this data set. Moreover, the AUC can summarize performance more compactly than an ROC curve. For these reasons, we present the results of the classification experiments in terms of the AUC.

The results of all of the experiments are compactly summarized in Figures 3, 4, and 5. The

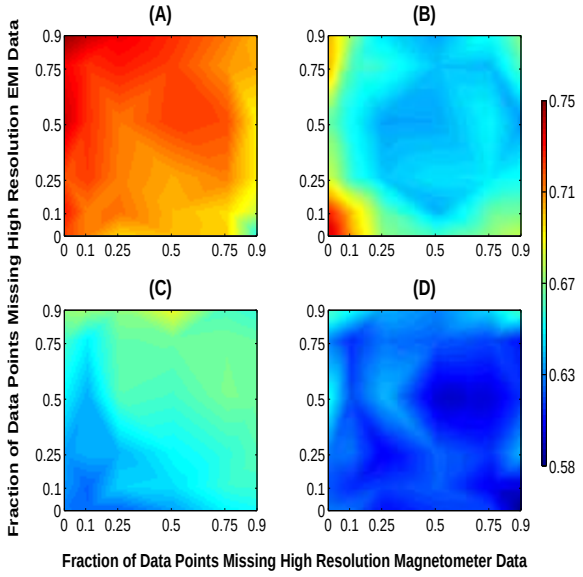


Fig. 3. Experimental results in terms of AUC when 25% of the data set is (labeled) training data. (A) The proposed method; (B) four separate classifiers are built, one for each possible combination of missing data; (C) one classifier is built on all features, with missing data integrated out analytically; and (D) one classifier is built on all features, with missing data handled via unconditional mean imputation.

results are displayed in these figures as images, interpolated from the results of the finite set of 36 pairs of missing-data conditions explained previously. Specifically, the images display the AUC values as a function of the amounts of missing fine-resolution magnetometer and missing fine-resolution EMI sensor data. The results from which the resulting images were interpolated were the mean AUC values over the 100 independent trials of the 36 pairs of conditions. The color scales are identical in the four panels within each figure, so visual comparisons among the methods' results can be made easily.

As can be seen from the three figures, the proposed method consistently performs better than the other three competing methods, regardless of the amounts of missing high-resolution data.

VI. Discussion

It should be emphasized that in the proposed method, the classifier weights are on the primary features, which are extracted from fine-resolution imagery. However, the auxiliary features extracted from coarse-resolution imagery are still utilized in the algorithm when primary data is missing. Specifically, missing primary data is analytically integrated out via the estimated density

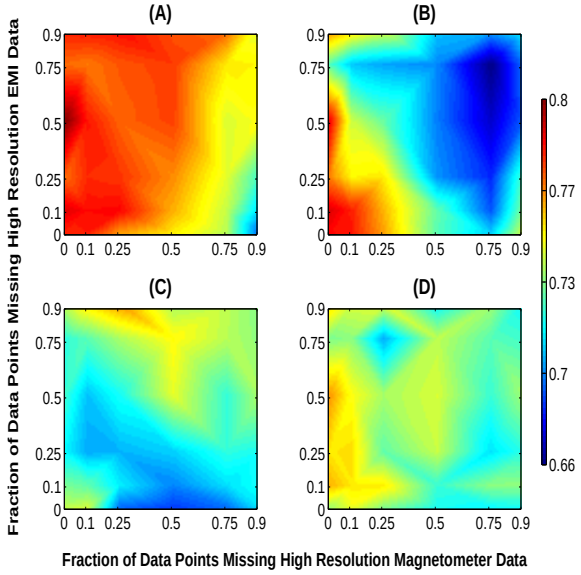


Fig. 4. Experimental results in terms of AUC when 50% of the data set is (labeled) training data. Refer to the caption of Figure 3 for additional details.

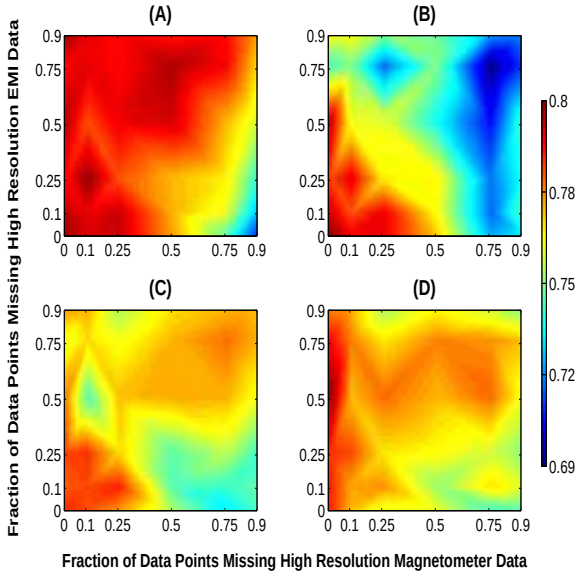


Fig. 5. Experimental results in terms of AUC when 75% of the data set is (labeled) training data. Refer to the caption of Figure 3 for additional details.

function, which models the relationship between the features from coarse-resolution imagery and those same features from fine-resolution imagery. The experimental results consistently show that the proposed method outperforms the alternative methods. Here we explain the reasons underlying this result.

Fine-resolution imagery contains salient aspects that are absent in coarse-resolution imagery. Therefore, features extracted from fine-resolution imagery should be preferred to features extracted from coarse-resolution imagery. Our proposed approach emphasizes the importance of the finer-resolution data by building a classifier with only that data. The coarser-resolution data is still exploited (via the estimated density function), albeit in an auxiliary role.

If a classifier is instead built on the conglomerated features extracted from different resolutions — as it was in two of the alternative methods — the information about the relative "quality" of the features is ignored. Concatenating features extracted from different resolution imagery also causes the feature-dimension to grow quickly, which can in turn lead to overfitting of the training data. In theory, a prior could be incorporated to combat overfitting, but additional complications arise as a result of having incomplete data. In contrast, the proposed method has no such overfitting issues.

The proposed method also consistently outperforms the method that builds a separate classifier for data from each sensor-resolution combination. This result is possible because the proposed method utilizes side information in the form of the estimated density function. By exploiting the statistical relationship that exists among features at different resolutions (as well as among features from different sensors), better performance can be achieved. This result can perhaps best be understood from the viewpoint of super-resolution techniques. Knowledge about a problem (*e.g.,* that noise in an image is Gaussian) can be exploited to resolve a super-resolution image from several blurry images [29]. Similarly, in this problem, knowledge of the statistical relationship between features at different resolutions can be exploited. Importantly, the proposed approach avoids the unnecessary intermediate step of forming an entire super-resolution *image*; instead, the ultimate goal is addressed directly: obtaining the equivalent of "super-resolution *features*."

VII. Conclusion

Acquiring fine-resolution imagery for all data points may be prohibitively expensive. For example, in the UXO detection problem, deploying ground-based sensors is dangerous and

time-consuming. This work presents a principled algorithm to classify imagery that is available at multiple resolutions. Because some data points may possess imagery at only a subset of resolutions, the problem can be viewed as one of incomplete-data classification. The proposed algorithm also naturally handles the case in which multiple sensor modalities — each of which may operate at multiple resolutions — are used to acquire data. In summary, the novel problem we addressed was of multi-sensor, multi-resolution, incomplete-data classification. Experimental results on a challenging UXO classification task employing magnetometer and EMI sensors demonstrated the advantage of the proposed algorithm over common alternatives.

Future work will focus on the development of an active data acquisition algorithm that determines which data points should receive finer-resolution imagery — and at which particular resolution level — in order to most improve performance. This active sensing concept is relevant for many applications, including medical imaging, remote sensing, and video tracking.

ACKNOWLEDGMENTS

The research reported here was supported by the Strategic Environmental Research and Development Program (SERDP).

REFERENCES

[1] S. Billings, C. Pasion, S. Walker, and L. Berans, "Magnetic models of unexploded ordnance," *IEEE Trans. Geoscience Remote Sensing*, vol. 44, pp. 2115–2124, 2006.

[2] S. Chilaka, D. Faircloth, L. Riggs, and H. Nelson, "Enhanced discrimination among UXO-like targets using extremely low-frequency magnetic fields," *IEEE Trans. Geoscience Remote Sensing*, vol. 44, pp. 10–21, 2006.

[3] C. Moss, T. Grzegorczyk, K. O'Neill, and J. Kong, "A hybrid time-domain model of electromagnetic induction from conducting, permeable targets," *IEEE Trans. Geoscience Remote Sensing*, vol. 44, pp. 2916–2926, 2006.

[4] K. Sun, K. O'Neill, F. Shubitidze, I. Shamatava, and K. Paulsen, "Fast data-derived fundamental spheroidal excitation models with application to UXO discrimination," *IEEE Trans. Geoscience Remote Sensing*, vol. 43, pp. 2573–2583, 2005.

[5] J. Stalnaker, M. Everett, A. Benavides, and C. Pierce, "Mutual induction and the effect of host conductivity on the EM induction response of buried plate targets using 3-d finite-element analysis," *IEEE Trans. Geoscience Remote Sensing*, vol. 44, pp. 251–259, 2006.

[6] S. Hart, R. Shaffer, S. Rose-Pehrsson, and J. McDonald, "Using physics-based modeler outputs to train probabilistic neural networks for unexploded ordnance (UXO) classification in magnetometry surveys," *IEEE Trans. Geoscience Remote Sensing*, vol. 39, pp. 797–804, 2001.

[7] C. Nelson, C. Cooperman, W. Schneider, D. Wenstrand, and D. Smith, "Wide bandwidth time-domain electromagnetic sensor for metal target classification," *IEEE Trans. Geoscience Remote Sensing*, vol. 39, pp. 1129–1138, 2001.

[8] Y. Zhang, L. Collins, H. Yu, C. Baum, and L. Carin, "Sensing of unexploded ordnance with magnetometer and induction data: theory and signal processing," *IEEE Trans. Geoscience Remote Sensing*, vol. 41, pp. 1005–1015, May 2003.

[9] A. Willsky, "Multiresolution Markov models for signal and image processing," *Proceedings of IEEE*, vol. 90, no. 8, pp. 1396–1458, 2002.

[10] H. Choi and R. Baraniuk, "Multiscale image segmentation using wavelet-domain hidden Markov models," *IEEE Trans. Image Processing*, vol. 10, no. 9, pp. 1322–1331, September 2001.

[11] J. Li, R. Gray, and R. Olshen, "Multiresolution image classification by hierarchical modeling with two dimensional hidden Markov models," *IEEE Trans. Information Theory*, vol. 46, no. 5, pp. 1826–1841, 2000.

[12] S. Mallat, *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[13] C. Li and R. Wilson, "Image segmentation based on a multiresolution Bayesian framework," in *Proc. IEEE International Conf. Image Processing*, 1998, pp. 761–765.

[14] C. Bouman and B. Liu, "Multiple resolution segmentation of textured images," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 13, no. 2, pp. 99–113, 1991.

[15] C. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Trans. Image Processing*, vol. 3, no. 2, pp. 162–177, March 1994.

[16] D. Rubin, *Multiple Imputation for Nonresponse in Surveys*. Wiley, 1987.

[17] D. Williams, X. Liao, Y. Xue, and L. Carin, "Incomplete-data classification using logistic regression," in *Proc. International Conf. Machine Learning*, 2005, pp. 977–984.

[18] M. Opper and O. Winther, "Gaussian processes and SVM: Mean field and leave-one-out," in *Advances in Large Margin Classifiers*, A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, Eds. MIT Press, 2000, pp. 311–326.

[19] P. McCullagh and J. Nelder, *Generalized Linear Models, 2nd Edition*. Chapman & Hall, 1989.

[20] D. Williams, "Classification and data acquisition with incomplete data," Ph.D. dissertation, Duke University, 2006.

[21] C. Baum, Ed., *Detection and Identification of Visually Obscured Targets*. Taylor and Francis, 1998.

[22] L. Carin, H. Yu, Y. Dalichaouch, A. Perry, P. Czipott, and C. Baum, "On the wideband EMI response of a rotationally symmetric permeable and conducting target," *IEEE Trans. Geoscience Remote Sensing*, vol. 39, pp. 1206–1213, 2001.

[23] N. Geng, C. Baum, and L. Carin, "On the low-frequency natural response of conducting and permeable targets," *IEEE Trans. Geoscience Remote Sensing*, vol. 37, pp. 347–359, Jan. 1999.

[24] W. Panofsky, *Classical Electricity and Magnetism*. Addison-Wesley, 1962.

[25] X. Liao and L. Carin, "Application of the theory of optimal experiments to adaptive electromagnetic-induction sensing of buried targets," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 26, pp. 961–972, 2004.

[26] W. Press, B. F. S. Teukolsky, and W. Vetterling, *Numerical Recipes in C : The Art of Scientific Computing, 2nd Edition*. Cambridge University Press, 1992.

[27] H. Nelson and J. McDonald, "Multisensor towed array detection system for UXO detection," *IEEE Trans. Geoscience Remote Sensing*, vol. 39, pp. 1139–1145, 2001.

[28] J. Hanley and B. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, pp. 29–36, 1982.

[29] R. Tsai and T. Huang, "Multi-frame image restoration and registration," *Advances in Computer Vision and Image Processing*, vol. 1, pp. 317–339, 1984.

APPENDIX

A. Modified Gradient Ascent

The classifier $\boldsymbol{w}$ from Section III is learned via a modified form of gradient ascent. This method uses the gradient and Hessian of the log-likelihood, which we provide explicitly here. For convenience, we first rewrite the log-likelihood function (15) as

$$\ell(\boldsymbol{w}) = \sum_{i=1}^{N_L} \log \left[\epsilon_i + (1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \, \sigma_k^i \right] \tag{24}$$

where

$$\sigma_k^i = \sigma(f_k^i) \tag{25}$$

$$f_k^i = y_i \beta (\boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^i + \boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i})(\gamma_k^i)^{-1} \tag{26}$$

$$\gamma_k^i = \sqrt{\boldsymbol{w}_{m_i}^T \boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} + \beta^2}. \tag{27}$$

The gradient of the log-likelihood is

$$\frac{\partial \ell(\boldsymbol{w})}{\partial \boldsymbol{w}} = \sum_{i=1}^{N_L} \frac{\left[(1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \, \sigma_k^i (1-\sigma_k^i) \frac{\partial f_k^i}{\partial \boldsymbol{w}} \right]}{\left[\epsilon_i + (1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \sigma_k^i \right]}, \tag{28}$$

while the Hessian of the log-likelihood is

$$\begin{aligned} \frac{\partial^2 \ell(\boldsymbol{w})}{\partial \boldsymbol{w} \partial \boldsymbol{w}^T} \;=\; & \sum_{i=1}^{N_L} \left\{ \frac{\left[2(1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \sigma_k^i (1-\sigma_k^i)^2 \frac{\partial^2 f_k^i}{\partial \boldsymbol{w} \partial \boldsymbol{w}^T} \right]}{\left[\epsilon_i + (1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \; \sigma_k^i \right]} \right. \\ & \left. - \frac{\left[(1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \, \sigma_k^i (1-\sigma_k^i) \frac{\partial f_k^i}{\partial \boldsymbol{w}} \right] \left[\sum_{k=1}^{K} \delta_k^i (1-2\epsilon_i) \, \sigma_k^i (1-\sigma_k^i) \frac{\partial f_k^i}{\partial \boldsymbol{w}} \right]^T}{\left[\epsilon_i + (1-2\epsilon_i) \sum_{k=1}^{K} \delta_k^i \sigma_k^i \right]^2} \right\} \end{aligned} \tag{29}$$

where

$$\frac{\partial f_k^i}{\partial \boldsymbol{w}} = \begin{bmatrix} \frac{\partial f_k^i}{\partial \boldsymbol{w}_{o_i}} \\ \frac{\partial f_k^i}{\partial \boldsymbol{w}_{m_i}} \end{bmatrix} = \begin{bmatrix} \frac{y_i \beta \boldsymbol{x}_i^{o_i}}{\gamma_k^i} \\ \frac{y_i \beta \boldsymbol{\xi}_k^i}{\gamma_k^i} - \frac{y_i \beta (\boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^i + \boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i}) \boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i}}{\left(\gamma_k^i\right)^3} \end{bmatrix} \tag{30}$$

$$\frac{\partial^2 f_k^i}{\partial \boldsymbol{w} \partial \boldsymbol{w}^T} = \begin{bmatrix} \frac{\partial^2 f_k^i}{\partial \boldsymbol{w}_{o_i} \partial \boldsymbol{w}_{o_i}^T} & \frac{\partial^2 f_k^i}{\partial \boldsymbol{w}_{o_i} \partial \boldsymbol{w}_{m_i}^T} \\ \frac{\partial^2 f_k^i}{\partial \boldsymbol{w}_{m_i} \partial \boldsymbol{w}_{o_i}^T} & \frac{\partial^2 f_k^i}{\partial \boldsymbol{w}_{m_i} \partial \boldsymbol{w}_{m_i}^T} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \frac{-y_i \beta \boldsymbol{x}_i^{o_i} \left(\boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} \right)^T}{\left(\gamma_k^i\right)^3} \\ \frac{-y_i \beta \boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} \left(\boldsymbol{x}_i^{o_i} \right)^T}{\left(\gamma_k^i\right)^3} & \frac{\partial^2 f_k^i}{\partial \boldsymbol{w}_{m_i} \partial \boldsymbol{w}_{m_i}^T} \end{bmatrix} \tag{31}$$

$$\frac{\partial^2 f_k^i}{\partial \boldsymbol{w}_{m_i} \partial \boldsymbol{w}_{m_i}^T} = \frac{-y_i \beta}{(\gamma_k^i)^3} \left[\boldsymbol{\xi}_k^i \left(\boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} \right)^T + (\boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^i + \boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i}) \boldsymbol{\Omega}_k^i + \boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} \left(\boldsymbol{\xi}_k^i \right)^T \right] + \frac{3 y_i \beta (\boldsymbol{w}_{m_i}^T \boldsymbol{\xi}_k^i + \boldsymbol{w}_{o_i}^T \boldsymbol{x}_i^{o_i}) \boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} \left(\boldsymbol{\Omega}_k^i \boldsymbol{w}_{m_i} \right)^T}{(\gamma_k^i)^5}. \tag{32}$$

B. Classification Method 2

Here we explain in greater detail the second classification method used in the experiments. Define sensor 1 to be the magnetometer, and define sensor 2 to be the EMI sensor. Recall that data from two image resolutions are available for each of the two sensors. Let γ_x^s be the set of data points for which primary data from the s-th sensor is possessed; let γ_z^s be the set of data points for which auxiliary data from the s-th sensor is possessed. Context will elucidate whether the sets contain training or testing data points. Note that $\gamma_x^s \subseteq \gamma_z^s$ in all experiments in this paper because it is assumed that auxiliary data is available for all data points. Let $\gamma_z^s \backslash \gamma_x^s$ denote the set of data points in γ_z^s but not in γ_x^s. Table I compactly summarizes the manner in which the various classifiers of this method are constructed and utilized.

TABLE I

EXPLANATION OF CLASSIFICATION METHOD 2 OF THE EXPERIMENTS

CLASSIFIER	FEATURES ON WHICH THE CLASSIFIER IS BUILT	TRAINING DATA POINTS USED TO TRAIN CLASSIFIER	TESTING DATA POINTS EVALUATED BY CLASSIFIER
1	$[\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}]$	$\gamma_x^1 \cap \gamma_x^2$	$\gamma_x^1 \cap \gamma_x^2$
2	$[\boldsymbol{x}^{(1)}, \boldsymbol{z}^{(2)}]$	$\gamma_x^1 \cap \gamma_z^2$	$\gamma_x^1 \cap (\gamma_z^2 \backslash \gamma_x^2)$
3	$[\boldsymbol{z}^{(1)}, \boldsymbol{x}^{(2)}]$	$\gamma_z^1 \cap \gamma_x^2$	$(\gamma_z^1 \backslash \gamma_x^1) \cap \gamma_x^2$
4	$[\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}]$	$\gamma_z^1 \cap \gamma_z^2$	$(\gamma_z^1 \backslash \gamma_x^1) \cap (\gamma_z^2 \backslash \gamma_x^2)$

For example, a training data point that has both fine-resolution and coarse-resolution magnetometer data and both fine-resolution and coarse-resolution EMI sensor data would be used in the construction of all four classifiers. A testing data point that has both fine-resolution and coarse-resolution magnetometer data, but only coarse-resolution EMI sensor data would be evaluated (*i.e.,* classified) using classifier 2.

To summarize, in the training stage, all data points that possess the requisite features are used to train the classifier. This arrangement allows more data to be used in building the classifiers, and hence allows more accurate classifiers to be obtained. In the testing stage, a given testing data point is submitted to that classifier that fully exploits the fine-resolution features that the data point possesses.