# SANDIA REPORT

SAND2012-6961
Unlimited Release
Printed August 2012

# Early Time-Series Classification with Reliability Guarantee

Hyrum S. Anderson, Nathan Parrish, Maya R. Gupta

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

**NOTICE:** This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. 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 United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2012-6961
Unlimited Release
Printed August 2012

# Early Time-Series Classification with Reliability Guarantee

Hyrum S. Anderson
Data Analysis & Data Exploitation
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0401
hander@sandia.gov

Nathan Parrish
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-4322

Maya R. Gupta
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-4322

**Abstract**

We consider the early classification of (incomplete) time-series data given a complete time-series training set. The early classification problem arises naturally when test sample data is collected over time, or when costs must be incurred to collect the data. For example, for missile defense, it is important to determine the target type long before it reaches its target. A practical goal is to assign a class label as soon as enough data is available to make a good decision. This objective is formalized through the notion of reliability—the probability that a label assigned to the early, incomplete data matches that assigned to the complete data, and we propose a method to classify incomplete data only if a user-specified reliability threshold is met. Our approach models the complete data as a random variable whose distribution is dependent on the current incomplete data and the training data. The method differs from standard strategies in that our focus is on determining the reliability of the early classification decision, not only the accuracy. Proposed methods are tested on a set of open-domain time-series datasets; where the goal is to classify the time-series as early as possible while still guaranteeing that the reliability threshold is met.

# Contents


**1 Introduction** . . . . . **9**

**2 Incomplete Decision Rules** . . . . . **10**

**3 Defining a Set $A$ that Contains Measure $\tau$ of $X$** . . . . . **12**
3.1 Chebyshev Construction for Set $A$ . . . . . 12
3.2 Naive Bayes Constructions for Set $A$ . . . . . 12

**4 Efficient Solutions for Linear or Quadratic Discriminants** . . . . . **14**
4.1 Two-class problems . . . . . 14
4.2 Multi-class Classifiers . . . . . 17
4.3 General Multi-class Decision Process . . . . . 18

**5 Estimation of the Complete Test Data Distribution** . . . . . **20**
5.1 Joint Gaussian Estimation . . . . . 20
5.2 GMM Based Estimation . . . . . 20

**6 Related Work** . . . . . **22**
6.1 Other Early Classification Work . . . . . 22
6.2 Related Work on Missing and Noisy Features . . . . . 22
6.3 Optimal Stopping Rules . . . . . 23
6.4 Feature Selection . . . . . 23
6.5 Online and Incremental Learning . . . . . 24
6.6 Sequential Hypothesis Testing . . . . . 24

**7 Experiments** . . . . . **25**
7.1 Experimental Set-up and Details . . . . . 25
7.2 Comparison of Construction of Sets of Measure $\tau$ . . . . . 26
7.3 Comparison of Estimation Methods . . . . . 27
7.4 Dimensionality Reduction Features . . . . . 29
7.5 Comparison to Other Methods . . . . . 31

**8 Discussion and Some Open Questions** . . . . . **37**

**References** . . . . . **38**


# Appendix


**A Proofs** . . . . . **42**

**B Gradient Descent Solution for the Quadratic Min and Max Problems** . . . . . **44**

**C Derivation of the Variance for GMM Based Estimation** . . . . . **45**

# Figures


1 The figure illustrates incomplete time-series classification. The incomplete time signal is shown in green. The complete signal is treated as random, and its distribution is estimated assuming that it is iid with the training signals. From the pdf of the complete data one can estimate the pdf of the feature data $p(x|z)$, and, using this distribution, one can check whether or not a reliable decision can be made. . . . 10

2 **Left:** A two-dimensional feature space and a linear class decision boundary. The mass of $X$ lies mostly to the left of the decision boundary. For values of $\tau$ that are smaller than the mass of $X$ that falls to the left of the decision boundary, the ideal incomplete decision rule would choose to classify rather than wait. **Right:** The entire mass of $X$ falls on one side of the decision boundary, and thus the ideal incomplete decision rule would choose to classify rather than wait for any value of $\tau$. On the other hand, the computable incomplete decision rule constructs some set $A$ that captures a fraction $\tau$ of the mass of $X$, then requires that entire set $A$ to lie on one side of the decision boundary. For the choice of $A$ shown here, the set $A$ crosses the decision boundary, and thus the computable decision rule would choose to wait for more data. . . . . 11

3 The sets $A$ containing mass $\tau$ of the conditional p.d.f. of $X$ using the three different construction methods proposed Section **??**. . . . . 12

4 Three different scenarios for incomplete data classification. In the leftmost plot, the classifier withholds making a decision. In the center and rightmost plots, $A$ lies completely on a single side of the decision boundary, so the classifier assigns a label to the incomplete data. . . . . 15

5 The left figure shows the time required by the SDP vs gradient descent solutions for different reliabilities. The right figure verifies that the solution for the methods is identical. . . . . 17

6 Average classification time vs test reliability for local QDA (left column) and linear SVM (right column) using jointly Gaussian prediction. The symbols correspond to choices of $\tau \in \{0.001, 0.1, 0.25, 0.9\}$. . . . . 27

7 Average classification time vs test reliability for local QDA (left column) and linear SVM (right column) using the näive Bayes quadratic constraint set with $\tau$ varied between $[0.001, 0.1, 0.25, 0.9]$. . . . . 28

8 Average classification time vs test reliability for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN. . 33

9 Average classification time vs test reliability for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN. . 34

10 Average classification time vs test accuracy for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN. . 35

11 Average classification time vs test accuracy for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN. . 36

# Tables


1 Time-series Datasets . . . . . 25
2 Average test time per sample, in seconds, for the three different constraint sets. . . . 26
3 Average test time per sample, in seconds, for the two different estimation methods. 29
4 Time-series length and the number of features after LDG dimensionality reduction as well as a comparison of the testing time, in milliseconds, required to perform reliable local QDA classification with the näive Bayes quadratic constraint set and jointly Gaussian estimation. The test time shown measures the average time, per test sample, to perform reliable classification at time $t = 1$. Therefore, this is a worst case test time in terms of real-time performance as the number of unknowns in the optimization problem for reliable classification is maximized at time $t = 1$. . 30

# 1 Introduction

Early classification is important any time there is a penalty for collecting additional features of the test data. For example, in medical applications, it is beneficial to make a diagnosis after subjecting a patient to as few tests as possible. When working with time-series data, it is beneficial to make time-sensitive decisions as early as possible. Of course, the earliness of a decision must be balanced with the fact that, intuitively, as more data is collected, one may make a more informed decision.

The early classification problem is a special case of the missing features problem, to which our methods are also applicable. Generally, features may be missing because of data drop-out, deletion, no-responses on surveys, etc. Although the experiments and emphasis in this work are for time-series data in which features are missing only because one has yet to collect them (if one were to wait long enough, the incomplete data would become complete), our approach is also useful in scenarios where the data is innately missing and cannot be recovered. The classifiers we design may be applied to any incomplete data with the guarantee that the decision made using the incomplete data has a high probability of matching the decision that would be made if the complete data were available. Our approach also makes it possible to answer the related question, "If we classify based on the current incomplete data, what is the probability that the class decision will be the same as classifying from the complete data?"

Our approach also relates to adaptive sampling and active learning, in which one chooses *which* features are to be collected first, so that a decision can be made on the best features available. This scenario is similar to the medical diagnosis example given above, in which a doctor requests future tests based on previous results. In this work, we do not consider adaptivity, and are agnostic to the order in which features are discovered; however, given a current set of "best" features in incomplete data, our methods can provide guarantees on the classification results over the yet-to-be-observed feature space.

First, we propose optimal and practical decision rules for classifying incomplete data. In Sections 3, 4, and 5 we provide the details on how to efficiently and accurately implement the proposed practical decision rule for classifiers that use linear or quadratic discriminants, such as linear support vector machines and linear or quadratic discriminant analysis (LDA or QDA). In Section 6, we review related work on classifying with missing features and related work on early classification of time-series data. Experiments in Section 7 show that the proposed incomplete decision rule consistently provides enhanced reliability over the state of the art in classifying incomplete data. We further discuss the results and some open questions in Section 8.

This paper significantly extends a prior conference paper [1], where we tackled the same problem but proposed a more conservative decision rule. Here, we propose a more optimal, but still computationally tractable, decision rule. We also show how the new rule can be used with different kinds of classifiers, show that our approach can be applied to different features, and provide substantially more analysis and experimental results.

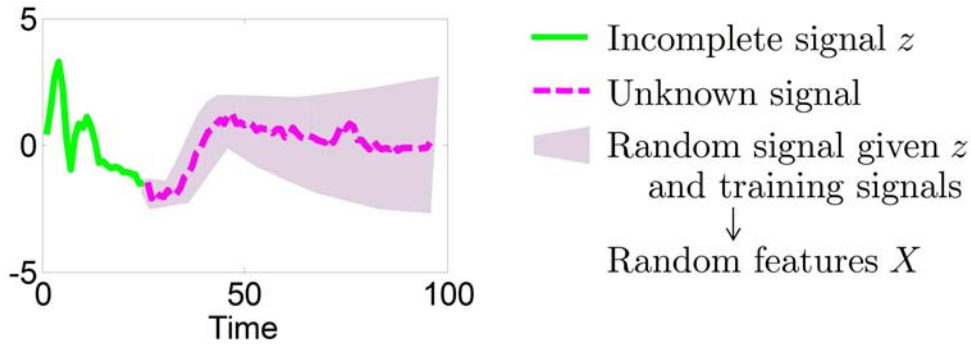


Figure 1: The figure illustrates incomplete time-series classification. The incomplete time signal is shown in green. The complete signal is treated as random, and its distribution is estimated assuming that it is iid with the training signals. From the pdf of the complete data one can estimate the pdf of the feature data $p(x|z)$, and, using this distribution, one can check whether or not a reliable decision can be made.

# 2 Incomplete Decision Rules

Let $\hat{g}(x)$ be a classifier function that assigns a class label to test sample $x$. However, suppose that at test time, one is given an incomplete data given as a vector $z$ in place of $x$. We wish to classify $z$ if it provides enough data to make a good decision, otherwise, we delay making a decision until more data is available. To that end, we consider decision rules that answer the question: "If we classify $z$, can we be assured that, with some minimum probability threshold, the same class label would be chosen when classifying $x$?" We use the term *reliability* to mean the probability that the class label assigned to $z$ matches that assigned to $x$.

To estimate reliability, we model the classification features derived from the complete data as a random variable $X$ that is jointly distributed with $Z$ — a random variable modeling the incomplete data. Given a desired reliability $\tau \in [0,1]$ and a realization of the incomplete data $z$, an ideal incomplete decision rule is to classify as class $g$ if

$$P(X \text{ s.t. } \hat{g}(X) = g | Z = z) = \int_{x \text{ s.t. } \hat{g}(x)=g} p(x|z)\, dx \geq \tau, \tag{1}$$

and otherwise to wait for more data. Figure 2 illustrates this rule.

The ideal condition in (1) could be checked in several ways. A straightforward approach is to compute the integral directly and see if it is greater than or equal to $\tau$. An alternative approach is to consider all sets $A$ in the domain of $X$ such that $P(X \in A | Z = z) \geq \tau$, and see if $\hat{g}(x)$ maps all $x$ in one such set to a single class $g$. In general, we expect these approaches to be computationally intractable.

We propose that a more conservative, but computable, incomplete decision rule is to classify as class $g$ if

$$\hat{g}(x) = g \;\text{ for all }\; x \in A \text{ for some set } A \text{ such that } P(X \in A | Z = z) \geq \tau. \tag{2}$$

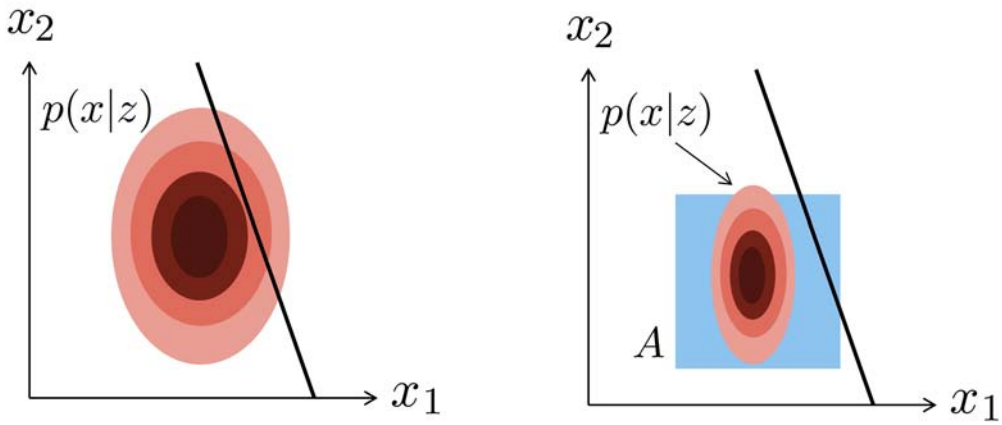


Figure 2: **Left:** A two-dimensional feature space and a linear class decision boundary. The mass of $X$ lies mostly to the left of the decision boundary. For values of $\tau$ that are smaller than the mass of $X$ that falls to the left of the decision boundary, the ideal incomplete decision rule would choose to classify rather than wait. **Right:** The entire mass of $X$ falls on one side of the decision boundary, and thus the ideal incomplete decision rule would choose to classify rather than wait for any value of $\tau$. On the other hand, the computable incomplete decision rule constructs some set $A$ that captures a fraction $\tau$ of the mass of $X$, then requires that entire set $A$ to lie on one side of the decision boundary. For the choice of $A$ shown here, the set $A$ crosses the decision boundary, and thus the computable decision rule would choose to wait for more data.

Rule (2) differs from (1) in that only one set $A$ that contains at least $\tau$ measure of $X$ must be checked. This rule is more conservative than (1) because it does not check all sets $A$, and thus (1) could be satisfied without (2) being satisfied (but not vice-versa).

Implementing the proposed rule in (2) requires three steps. First, one must estimate the conditional density $p(x|z)$. Second, an appropriate set $A$ must be constructed. Third, one must compute the probability and verify that the threshold $\tau$ in (2) is satisfied. We first discuss the construction of a set $A$ in Section 3, and show that our construction only requires estimates of the first and second conditional moments of $X$. Then in Section 4, we show how rule (2) can be efficiently checked for classifiers that have linear or quadratic class discriminant functions. We delay the discussion of how to estimate the necessary moments of $X$ until Section 5.

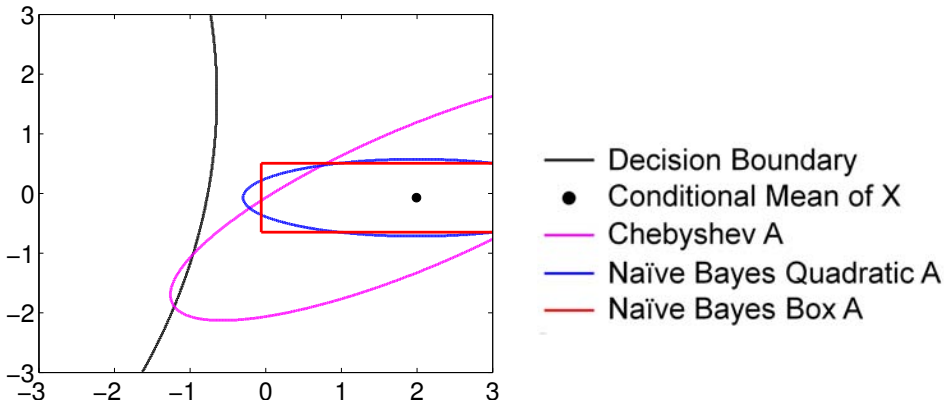


Figure 3: The sets $A$ containing mass $\tau$ of the conditional p.d.f. of $X$ using the three different construction methods proposed Section 3.

# 3 Defining a Set $A$ that Contains Measure $\tau$ of $X$

To implement the decision rule (2), one must construct a set $A$ that contains at least $\tau$ measure of $X$ given $Z = z$. In this section we propose three ways to construct a set $A$. Figure 3 compares these three constructions.

## 3.1 Chebyshev Construction for Set $A$

The first construction for set $A$ assumes only that the distribution $p(x|z)$ has finite first and second moments which can be estimated. Then a set $A$ may be constructed using the multidimensional Chebyshev inequality, which states that for $X \in \mathbb{R}^d$ with known mean $m$ and covariance $R$, and any $\alpha > 0$

$$P\left((X-m)^T R^{-1}(X-m) \leq \alpha^2\right) \geq 1 - \frac{d}{\alpha^2}.$$

Thus to satisfy $P(X \in A|Z = z) \geq \tau$, define

$$A = \left\{ x \text{ s.t. } (x-m)^T R^{-1}(x-m) \leq \frac{d}{1-\tau} \right\}. \tag{3}$$

The set $A$ defined by (3) is non-empty for $\tau \in (-\infty, 1]$, although $\tau \leq 0$ does not provide an informative bound for classifier reliability.

## 3.2 Naive Bayes Constructions for Set $A$

The Chebyshev construction given in the previous section can be overly conservative, as it makes no assumptions about the conditional distribution of $X$ other than a finite mean and covariance. If one assumes more about the distribution, a tighter constraint set $A$ may be found that results in a less

conservative decision rule, and therefore earlier classification for the same reliability requirement $\tau$. For example, if the conditional distribution is assumed to be Gaussian[1], then the quadratic set $A$, centered about the mean, that covers $\tau$ measure of $X$ is

$$A = \left\{x \text{ s.t. } (x-m)^T R^{-1}(x-m) \leq \text{erf}(\tau)\right\}, \tag{4}$$

where one must compute the inverse cdf to determine the value $\text{erf}(\tau)$ to achieve a set $A$ with measure $\tau$. For a multi-dimensional Gaussian, computing the inverse cdf for (4) is non-trivial. Equation (4) can be simplified by making the conservative näive Bayes assumption that the components of $X$ are independent, and thus $R$ is diagonal. Then the quadratic function in (4) becomes $\sum_{\ell=1}^{d}\left(\frac{x(\ell)-m(\ell)}{\sqrt{R(\ell,\ell)}}\right)^2$. Under the independent Gaussian assumption, $\sum_{\ell=1}^{d}\left(\frac{X(\ell)-m(\ell)}{\sqrt{R(\ell,\ell)}}\right)^2$ is a chi-squared random variable with $d$ degrees of freedom; thus, the $\text{erf}(\tau)$ function in (4) is easily computed using the inverse cdf of a chi-squared random variable.

A related option is to force the set $A$ to be a box that is centered about the mean. Again employing the näive Bayes assumption that elements of $X$ are independent, one may express $p(x|z) = \prod_{\ell=1}^{d} p(x(\ell)|z)$. Therefore, a set $A$ may be defined as

$$A = \{x \text{ s.t. } x(\ell) \in [m(\ell) - s_\tau(\ell), m(\ell) + s_\tau(\ell)] \; \forall \; \ell = 1, ..., d\}, \tag{5}$$

where $s_\tau$ is a vector defining the width of the box in each dimension such that the total measure of the box is $\tau$. In this paper, we implement this constraint by assigning each dimension equal measure $\tau^{1/d}$ while assuming that each marginal distribution $X(\ell)$ is Gaussian.

Both sets in (4) and (5) employ the same assumptions about the conditional distribution of $X$, but (4) finds the ellipsoidal footprint of the Gaussian that has measure $\tau$, while (5) treats the dimensions completely independently, giving each of the marginals measure $\tau^{1/d}$.

[1]The Gaussian assumption is often justified by a maximum entropy argument or a simplicity argument.

# 4 Efficient Solutions for Linear or Quadratic Discriminants

In this section, we show that the incomplete data classification rule (2) with the constraint sets $A$ proposed in Section 3 can be computed efficiently for classifiers of the form

$$\hat{g}(x) = \arg\max_{g} f_g(x), \tag{6}$$

where $f_g(x)$ is a linear or quadratic discriminant function for the $g^{\text{th}}$ class, and according to (6), the classifier assigns $x$ to the class with maximum discriminant. For example, the linear support vector machine (SVM) has a linear discriminant, while the quadratic discriminant analysis (QDA) classifier has a quadratic discriminant [24].

Nearest-neighbor classifiers using a Euclidean (or Mahalanobis) distance have a discriminant that over the set $x \in A$ requires taking the minimum of a set of quadratic discriminants:

$$f_g(x) = \min_{x_i : y_i = g} (x - x_i)^T (x - x_i). \tag{7}$$

An optimal method for checking the incomplete decision rule (2) for this discriminant is an open question. A conservative reliability decision can be made by treating each sample as its own class in (6). That is, let $f_i(x) = (x - x_i)^T (x - x_i)$, solve (6) for the resulting quadratic discriminant, and then classify as the class $y_i$. A computationally simpler approach (but one that is not strictly conservative), is to only consider each class's nearest neighbor to the posterior mean, which produces one quadratic discriminant per class. In experiments, we do something similar to the latter approach using a local QDA classifier.

We begin with the two-class problem, and then show how this rule can be extended to multi-class problems.

## 4.1 Two-class problems

We first consider a two-class problem, where the set of class labels is $\mathscr{G} = \{1, 2\}$. Let $f_1(x)$ and $f_2(x)$ be the discriminants for classes one and two, and define

$$f(x) = f_2(x) - f_1(x).$$

An equivalent classifier to (6) using only $f(x)$ can be defined by noting that $f(x) = 0$ defines the decision boundary between classes 1 and 2. Therefore, classification rule (6) is equivalent to

$$\hat{g}(x) = \begin{cases} 1 & \text{if } f(x) \leq 0 \\ 2 & \text{if } f(x) > 0 \end{cases}. \tag{8}$$

Then the proposed incomplete data decision rule (2) is implemented

$$\hat{g}(z) = \begin{cases} 1 & \text{if } \max_{x \in A} f(x) \leq 0 \\ 2 & \text{if } \min_{x \in A} f(x) > 0 \\ \text{no decision} & \text{otherwise.} \end{cases} \tag{9}$$

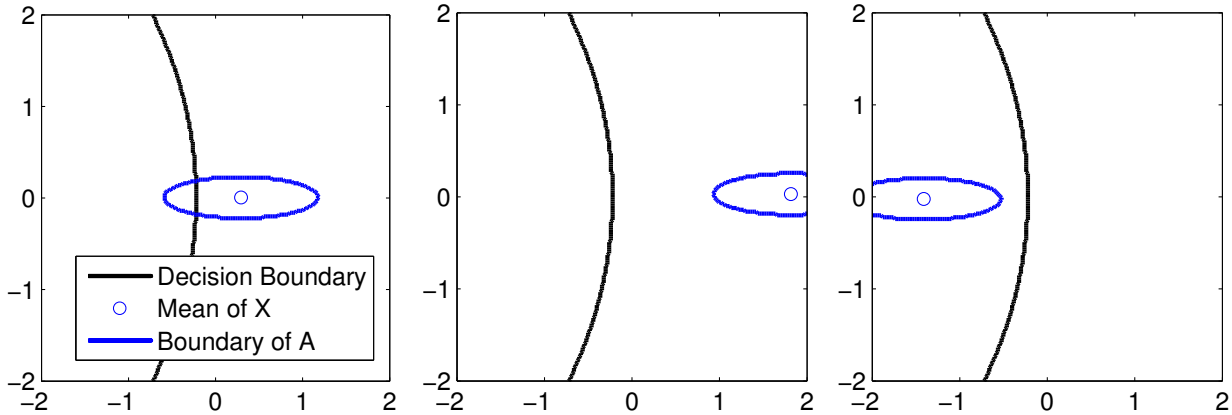


Figure 4: Three different scenarios for incomplete data classification. In the leftmost plot, the classifier withholds making a decision. In the center and rightmost plots, $A$ lies completely on a single side of the decision boundary, so the classifier assigns a label to the incomplete data.

Note that the decision rule (9) is dependent on the incomplete data through the dependence of $A$ on $z$.

The decision rule in (9) classifies the incomplete data if the constraint set $A$ lies completely on one side of the decision boundary or the other. If the set $A$ lies across the decision boundary, then the rule withholds making a decision. The three different conditions in (9) are shown for a quadratic discriminant (and hence quadratic decision boundary) and a quadratic construction of the set $A$ in Figure 4.

**Linear Discriminants**

In order to efficiently check (9), we must be able to efficiently compute the maximum and minimum of $f(x)$ over the set $x \in A$. For linear discriminant functions, $f(x)$ is also linear. Then for a quadratic set $A$, such as the Chebyshev or näive Bayes quadratic sets $A$ given in Section 3, finding the maximum and minimum are quadratically constrained linear programs:

$$\begin{aligned} \max_{x \in A} f(x) = & \max_{x} \beta^T x + b \\ & \text{s.t. } (x-m)^T R^{-1} (x-m) \leq \delta \end{aligned} \tag{10}$$

$$\begin{aligned} \min_{x \in A} f(x) = & \min_{x} \beta^T x + b \\ & \text{s.t. } (x-m)^T R^{-1} (x-m) \leq \delta. \end{aligned} \tag{11}$$

**Proposition 1:** *The solutions to (10) and (11) are, respectively*

$$\max_{x\in A} f(x) = \beta^T m + \sqrt{\delta} \parallel R^{1/2}\beta \parallel_2 + b$$
$$\min_{x\in A} f(x) = \beta^T m - \sqrt{\delta} \parallel R^{1/2}\beta \parallel_2 + b.$$

The proof is given in Appendix A.

For a linear set $A$ such as the näive Bayes box constraint set given in (5), the maximum and minimum are

$$\max_{x\in A} f(x) = \max_x \beta^T x + b \quad \text{s.t. } m(\ell) - s_\tau(\ell) \leq x \leq m(\ell) + s_\tau(\ell) \; \forall \; \ell = 1, ..., d \tag{12}$$

$$\min_{x\in A} f(x) = \min_x \beta^T x + b \quad \text{s.t. } m(\ell) - s_\tau(\ell) \leq x \leq m(\ell) + s_\tau(\ell) \; \forall \; \ell = 1, ..., d. \tag{13}$$

The solution of (12) is $\beta^T m + |\beta^T| s_\tau + b$, and the solution of (13) is $\beta^T m - |\beta^T| s_\tau + b$.

### Quadratic Discriminants

If the class discriminant functions are quadratic, then $f(x) = f_2(x) - f_1(x)$ will also be quadratic and, thus, can be written

$$f(x) = (x - v)^T V (x - v) + b. \tag{14}$$

Since (14) is the difference of two quadratics, $V$ will generally be indefinite even if $f_2(x)$ and $f_1(x)$ are both positive semi-definite.

First consider finding the maximum and minimum of (14), as required by the incomplete decision rule (9), over a quadratic constraint set $A$. Since $V$ is indefinite, this is a non-convex optimization problem. However, strong duality holds for finding the minimum or maximum of any quadratic function subject to a quadratic constraint, as shown by [5, Appendix B]. The dual problem is a semi-definite program (SDP), and can therefore be solved using convex optimization such as an interior point methods. However, in our experiments, we found the SDP solution to be prohibitively slow. Therefore, we instead propose to use the two-step gradient descent approach described in Appendix B. [30] showed that there is at most one local non-global solution to this non-convex problem. Also, since we need only know if the minimum or maximum of $f(x)$ is less than or greater than zero, we can often stop the gradient descent before convergence. Figure (5) shows a run-time comparison between the SDP solution solved using `CVX` [22] running `SeDuMi` [42] and the gradient-descent solution.

Now consider finding the maximum and minimum of (14) over the box set $A$. An efficient solution is obtained by first performing a change of variables that diagonalizes $V$. Define $y = V^{1/2}x$

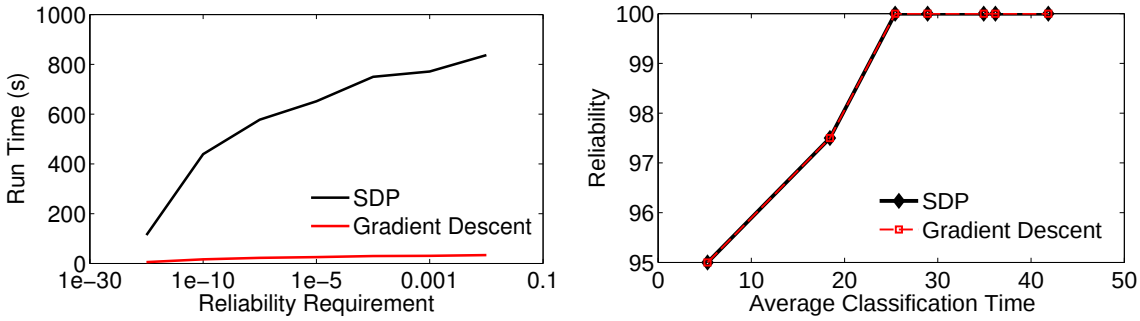


Figure 5: The left figure shows the time required by the SDP vs gradient descent solutions for different reliabilities. The right figure verifies that the solution for the methods is identical.

and $w = V^{1/2}v$, then $f(x) = f(y) = \| y - w \|^2 + b$. After this change of variables, we can greatly simplify the maximum and minimum computations required by the incomplete decision rule (9) by making the näive Bayes assumption on the random variable $Y = V^{1/2}X$ as opposed to on $X$. Defining the mean of $Y$ as $m_y = V^{1/2}m$,

$$\max_{x \in A} f(x) = \max_{y \in A} \| y - w \|^2 + b \tag{15}$$

$$\min_{x \in A} f(x) = \min_{y \in A} \| y - w \|^2 + b, \tag{16}$$

with

$$A = \left\{ y \text{ s.t. } y(\ell) \in [m_y(\ell) - s_\tau(\ell),\ m_y(\ell) + s_\tau(\ell)],\ \forall\, \ell = 1, \ldots, d \right\},$$

where the $s_\tau(\ell)$ are determined by the inverse cdf of $Y(\ell)$.

After this change of variables, the $y$ that maximizes (15) is found by assigning each $y(\ell)$ to the edge of the box that maximizes the distance from $w(\ell)$. Similarly, the $y$ that minimizes (16) assigns $y(\ell) = w(\ell)$ if $w(\ell) \in [m_y(\ell) - s_\tau(\ell),\ m_y(\ell) + s_\tau(\ell)]$. Otherwise, $y(\ell)$ is assigned to the edge of the box that minimizes the distance to $w(\ell)$.

## 4.2 Multi-class Classifiers

We now extend the results of the previous section to multi-class classifiers. For multi-class classifiers, the classification rule (6) can be expressed as

$$\hat{g}(x) = c \ \text{ if } \ f_c(x) - f_h(x) \geq 0 \ \text{ for all } \ h \neq c. \tag{17}$$

Using (17), the proposed incomplete data classification rule (2) can be written

$$\hat{g}(z) = \begin{cases} c & \text{if } \min_{x \in A} f_c(x) - f_h(x) \geq 0 \text{ for all } h \neq c \\ \text{no decision} & \text{otherwise} \end{cases}. \tag{18}$$

That is, classify $z$ as class $c$ if the set $A$ lies completely within the decision region for some class $c$, and do not decide at the requested reliability if the set $A$ straddles a decision boundary.

If there are $G$ total classes, then there are $2\binom{G}{2}$ possible checks, $\min_{x\in A} f_c(x) - f_h(x) \geq 0$, implied by (18). However, if the set $A$ contains the posterior mean $m$ (as it does in all of our proposed constructions), then a decision can be made with $G-1$ checks:

*Guess:* Let $c = \arg\max_g f_g(m)$.

*Check:* Sequentially check if $\min_{x\in A} f_c(x) - f_h(x) \geq 0$ for $h = 1, 2, \ldots, G$, $h \neq c$. If the check fails for any $h$, stop, and output the result *no decision*. If the check holds for all $h$, then classify early as class $c$.

In the next section, we prove that regardless of the construction of set $A$, one must compute at most $2(G-1)$ checks of $\min_{x\in A} f_c(x) - f_h(x) \geq 0$.

## 4.3 General Multi-class Decision Process

We provide a provably efficient multi-class decision process for arbitrary constructions of the constraint set $A$. We say that class $c$ *dominates* class $h$ and that class $h$ is *dominated by* $c$ if $f_c(x) - f_h(x) \geq 0$ for all $x \in A$. If neither class dominates the other one, then the two classes are called *tied*. To classify the incomplete data $z$ early as class $c$, class $c$ must dominate all other classes.

**Proposed Decision Process:**

*Initialize:* Begin with all $G$ classes labeled `candidate`.

*Compare:* Choose any two classes $c$ and $h$ that are labeled `candidate` and check whether or not $\min_{x\in A} f_c(x) - f_h(x) \geq 0$. If yes, then label $h$ as `dominated`. If no, then perform a second check to see if $\min_{x\in A} f_h(x) - f_c(x) \geq 0$, and if so then label $c$ as `dominated`, and otherwise label both classes as `tied`. Continue this process until fewer than two classes are labeled `candidate`. If no classes remain that are labeled `candidate`, then output *no decision*. If one class is labeled `candidate`, then proceed to the *Final Comparison*.

*Final Comparison:* Check if the last class labeled `candidate` dominates every class labeled `tied`. If yes, classify the incomplete data as the class labeled `candidate`, if no, output *no decision*.

**Proposition 2:** *The above decision process correctly determines the dominating class or that there is no dominating class.*
The proof is given in Appendix A.

**Proposition 3:** *Given G classes, the above decision process requires at most* $2(G-1)$ *and at least* $G-1$ *checks of* $\min_{x \in A} f_h(x) - f_c(x) \geq 0$.
The proof is given in Appendix A.

# 5 Estimation of the Complete Test Data Distribution

In order to construct the sets $A$ in Section 3, one must estimate the mean $m$ and covariance $R$ of the complete test data $X$. We do this by leveraging the incomplete data about the test signal that is currently available along with the prior knowledge of the structure of the test signal that is available in the corpus of training data. We present two estimation methods: 1) joint Gaussian estimation, and 2) Gaussian mixture model (GMM) estimation. These approaches are similar to those used in missing feature imputation, for example in speech recognition as described by [34]. However, our approach differs from that of missing feature imputation in that the latter constructs only a point estimate of the unknown data, whereas we construct estimates of the mean and covariance of the unknown data.

## 5.1 Joint Gaussian Estimation

For joint Gaussian estimation, we assume that the complete data $X$ is distributed jointly Gaussian with the incomplete data $Z$. Therefore, the model is

$$\begin{bmatrix} X \\ Z \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} \bar{x} \\ \bar{z} \end{bmatrix}, \begin{bmatrix} \Sigma_{x,x} & \Sigma_{x,z} \\ \Sigma_{z,x} & \Sigma_{z,z} \end{bmatrix} \right). \tag{19}$$

We estimate the model parameters in (19) from the training data. The mean and covariance parameters of $X$ conditioned on the realization of the incomplete data $Z = z$ are

$$\begin{aligned} m &= \hat{\bar{x}} + \hat{\Sigma}_{x,z}\hat{\Sigma}_{z,z}^{-1}(z - \hat{\bar{z}}) \\ R &= \hat{\Sigma}_{x,x} - \hat{\Sigma}_{x,z}\hat{\Sigma}_{z,z}^{-1}\hat{\Sigma}_{z,x}, \end{aligned}$$

where $\hat{\bar{x}}$, $\hat{\bar{z}}$, and $\hat{\Sigma}$ are the estimated parameter values.

## 5.2 GMM Based Estimation

We assume that the joint distribution of the complete data, $X$, and the incomplete data, $Z$, is a Gaussian mixture model, where the elements of the Gaussian mixture are the class-conditional distributions. Under these assumptions the model is

$$\begin{bmatrix} X \\ Z \end{bmatrix} \sim \sum_{g \in \mathcal{G}} w(g) p\left( \begin{bmatrix} x \\ z \end{bmatrix} | g \right), \tag{20}$$

where $w(g)$ is the weight of the class $g$ Gaussian and

$$p\left( \begin{bmatrix} x \\ z \end{bmatrix} | g \right) = \mathcal{N}\left( \begin{bmatrix} \bar{x}_g \\ \bar{z}_g \end{bmatrix}, \begin{bmatrix} \Sigma_{x,x}(g) & \Sigma_{x,z}(g) \\ \Sigma_{z,x}(g) & \Sigma_{z,z}(g) \end{bmatrix} \right).$$

We can again estimate the parameters of the model (the means, covariances, and weights) from the training data.

Let $\hat{\bar{x}}_g$, $\hat{\bar{z}}_g$, and $\hat{\Sigma}(g)$ be the estimated model parameters for class $g$, and define

$$\begin{aligned} m_g &= \hat{\bar{x}}_g + \hat{\Sigma}_{x,z}(g)\hat{\Sigma}_{z,z}^{-1}(g)(z - \hat{\bar{z}}_g) \\ R_g &= \hat{\Sigma}_{x,x}(g) - \hat{\Sigma}_{x,z}(g)\hat{\Sigma}_{z,z}^{-1}(g)\hat{\Sigma}_{z,x}(g) \\ p(g|z) &= \frac{w_g p(z|G=g)}{\sum_{h\in\mathscr{G}} w_h p(z|G=h)}. \end{aligned}$$

Given a realization $Z = z$, the conditional mean of $X$ can be computed as

$$m = \mathrm{E}[X|z] = \sum_{g\in\mathscr{G}} \mathrm{E}[X, G|z] = \sum_{g\in\mathscr{G}} m_g p(g|z).$$

Furthermore, as shown in Appendix C, the condtional covariance of $X$ is

$$R = \sum_{g\in\mathscr{G}} p(g|z)\left(R_g + m_g m_g^T\right) - \sum_{q\in\mathscr{G}}\sum_{h\in\mathscr{G}} m_q m_h^T p(q|z)p(h|z).$$

# 6 Related Work

We detail the related work in early classification and missing features, then we contrast the proposed with optimal stopping, feature selection, online and incremental learning, and sequential hypothesis ratio testing.

## 6.1 Other Early Classification Work

Xing et al. [49] considered the problem of making an early prediction on time-series data that matches that of a full length one nearest-neighbor classifier. Suppose that the labeled training dataset is $\{(x_i, g_i)\}_{i=1}^n$, where $x_i \in \mathbb{R}^d$. Their approach, called early classification on time-series (ECTS), is motivated by the idea of the *minimum prediction length* (MPL) of a training time-series $x_i$. Define $x_i(1:t) \in \mathbb{R}^t$ to be the first $t$ samples of $x_i$. Furthermore, define, $\text{RNN}(x_i(1:t))$ to be the reverse nearest neighbors of $x_i(1:t)$ which is the set of training samples that choose $x_i$ to be their nearest neighbor at time $t$. The MPL of $x_i$ is the smallest time index $k$ such that for all $k \leq \ell \leq d$ the following holds $\text{RNN}(x_i(1:\ell)) = \text{RNN}(x_i(1:d)) \neq \emptyset$. By this definition, the MPL is the smallest time index at which the reverse nearest neighbors of $x_i$ do not change as the rest of the time-series is revealed. At test time, a training point $x_i$ can be used to assign a label to a test sample $x(1:t)$ once $t \geq \text{MPL}(x_i)$, the minimum prediction length of $x_i$.

The authors found that the above procedure was too conservative; therefore, they proposed a slightly modified way to find the MPL for ECTS. They first clustered the training data using a hierarchical clustering method and then selected the MPL for each training time-series depending on its cluster membership. They also introduced a parameter to control the 'earliness' of their approach called *minimum support* – a ratio that varies between zero and one, with zero resulting in the earliest classifier. However, the minimum support parameter is different from our $\tau$ parameter in that it does not provide an explicit guarantee on the reliability of the early decision.

Xing et al. cite [36] as the only existing study mentioning early classification on time-series data. [36] propose to classify a time-series using a *literal* based classifier, where a literal is a descriptor describing what happens during a specified interval of the time-series. For example, the literal *increases* would be set to one if the time-series increases during the specified interval, and would be set to zero otherwise. The authors mention that for early classification of time-series some of the literals will not yet have a value because the interval that they are measured in has not occurred yet. The authors propose to 'omit' these literals from the classifier in order to classifier early. This approach to early classification is the näive approach, as it does not consider whether or not a reliable decision can be made with the incomplete data.

## 6.2 Related Work on Missing and Noisy Features

Another related body of work is imputing (estimating) missing features. If missing features occur in the training data, then standard methods of classifier training cannot be used. One method of

dealing with this problem, called single imputation, is to fill in the missing features with their estimated values. The missing features can be estimated using a multivariate regressor that is trained using the subset of training data with no missing features. [39] and [37] review missing feature methods for training data.

On the other hand, if features are missing in the test data, then they do not necessarily need to be imputed. A classifier can be trained for this test data by removing the same features that are missing in test data from the training data prior to classifier training. However, this approach requires the classifier to be retrained for each different subset of missing test features. This retraining can be avoided by estimating the missing features and then classifying the test data with the missing features filled in by their estimates. Our method differs from methods that classify the imputed test features in that we use the estimated mean and variance in order to measure the reliability of a test decision made with the incomplete data. Our work is complementary to imputation methods, and different methods than the ones we used in Section 5 could be easily substituted into the proposed approach.

If features are noisy rather than missing, then estimating the clean feature values can improve test accuracy. This problem arises, for example, in automatic speech recognition (ASR) systems when the test signal is masked by noise [7, 33, 34]. [34] compare MAP estimates for noisy features in ASR systems using Gaussian and GMM based estimators with models similar to those that we describe in Section 5.

## 6.3 Optimal Stopping Rules

Quoting [17], "The *theory of optimal stopping* is concerned with the problem of choosing a time to take a given action based on sequentially observed random variables in order to maximize an expected pay-off or to minimize an expected cost." While the high-level goal is the same, the optimal stopping perspective requires specification of misclassification costs and delay costs, which are often difficult to specify. Given such costs, an optimal stopping rule approach would attempt to estimate the probability of each class given the current incomplete data, and determine the expected costs of making a decision or waiting.

## 6.4 Feature Selection

A related problem in classification is to determine the best subset of features to use in classification. For example, the classic *forward selection method* sequentially adds in features based on their marginal value. Different stopping rules have been proposed to decide when to stop sequentially adding the features [8]. Generally stopping rules are not applicable to the problem we focus on because they assume that all increasing sets of features can be compared, rather than that one only has the incomplete set of features and must make a decision. In addition, stopping rules are based only on the training data statistics, and from our perspective are strictly suboptimal in that they do not consider the current incomplete data.

## 6.5 Online and Incremental Learning

In this paper we assume that a fixed set of training data is given, and that incremental features of a test sample become available. These assumptions differ from the usual set-up of online learning (also known as incremental learning), which assumes that incrementally more training data becomes available to train the classifier over time (e.g. [31], [14], [10]). Also assuming the online learning set-up, [19] propose a stopping rule for deciding when enough training samples have been received to classify with confidence.

## 6.6 Sequential Hypothesis Testing

The sequential probability ratio test (SPRT) [46] is a greedy alternate to the proposed work, designed for use with probabilistic models of two hypotheses. In the context of binary classification, and a generative model $p(y|x_k)$, it accumulates the log-likelihood ratio:

$$S_k = S_{k-1} + \log p(y_1|x_k) - \log(y_2|x_k), \tag{21}$$

and if $S_k$ exceeds a preset threshold $t_1$, the signal would be called for class 1, and if $S_k$ goes below a preset negative threshold $t_2$, the signal would be called for class 2. The thresholds are set to achieve desired error levels on class 1 and class 2 respectively.

[2] expanded SPRT for the multi-hypothesis case and applied it to linear discriminant analysis classification (in which each class is assumed to be drawn from a distribution with the same covariance matrix) for a different problem than the one treated here: given a sequence of iid samples from one class, he prescribed how to use SPRT to give a rule for how and when to determine the class.

A key difference between the proposed approach and the SPRT approach is that (21) is greedy: new features do not change the contribution to the log-likelihood already made by previous features, which stems from the standard SPRT assumption successive observations are independent. But the classifiers we consider in this paper are not trained to consider the features independently. Further, we assume correlations between the features in order to estimate a probability distribution over the unknown part of the feature vector, which we use to define a constraint set.

| Dataset | Time-series Length | Number of Classes | Training Samples | Test Samples |
|---|---|---|---|---|
| Chlorine Concentration | 166 | 3 | 467 | 3840 |
| Italy Power Demand | 24 | 2 | 67 | 1029 |
| Face (All) | 131 | 14 | 560 | 1690 |
| Medical Images | 99 | 10 | 381 | 760 |
| Non-Invasive Fetal ECG 1 | 750 | 42 | 1800 | 1965 |
| Non-Invasive Fetal ECG 2 | 750 | 42 | 1800 | 1965 |
| Starlight Curves | 1024 | 3 | 1000 | 8236 |
| Swedish Leaf | 128 | 15 | 500 | 625 |
| Synthetic Control | 60 | 6 | 300 | 300 |
| Two Patterns | 128 | 4 | 1000 | 4000 |
| U Wave Gesture Library X | 315 | 8 | 896 | 3582 |
| U Wave Gesture Library Y | 315 | 8 | 896 | 3582 |
| U Wave Gesture Library Z | 315 | 8 | 896 | 3582 |
| Wafer | 152 | 2 | 1000 | 6174 |
| Yoga | 426 | 2 | 300 | 3000 |

Table 1: Time-series Datasets

# 7 Experiments

Section 7.1 details the datasets, experimental set-up, and classifiers used. We first compare the proposed methods to construct sets $A$ of measure $\tau$, reported in Section 7.2, and the proposed estimation methods for the moments of $P_{X|z}$, reported in Section 7.3. Then in Section 7.4, we show that applying a dimensionality reduction method can greatly reduce the computation needed at test time. Lastly, we compare our recommended reliable classifier to other approaches to early classification. Complete experimental code can be downloaded from the publications page of http://idl.ee.washington.edu/.

## 7.1 Experimental Set-up and Details

We demonstrate performance using all of the time-series datasets available on the *UCR Time-Series Classification and Clustering Page* [26] that have at least five hundred test samples and at least 15 training examples per class when this paper was written. We use the given training and test splits, so all results can be reproduced. We also use the Synthetic Control dataset from this repository, a dataset of Gaussian data that has only three hundred test samples, to further illustrate the differences between the constraint sets and estimation methods that we have described for the proposed incomplete decision rule. Table 1 gives details for the used datasets.

The time-series classification experiments are performed as follows. The test dataset consists of $n$ sampled time-series vectors and corresponding labels $\{x_i, g_i\}_{i=1}^n$, with $x_i \in \mathbb{R}^d$ and $g \in \mathscr{G}$. At time $t$, the incomplete data for the $i^{\text{th}}$ test time-series is $z_i \in \mathbb{R}^t$, the first $t$ samples of $x_i$. At each time $t$ we check the proposed incomplete decision rule and classify $z_i$ if the reliability condition is

met for $\tau$. We plot results for a set of choices of $\tau$.

Let $t_i(\tau)$ be the minimum time at which the $i^{\text{th}}$ test signal can be classified with reliability constraint $\tau$, and let $\hat{g}(z_i(\tau))$ be the class label assigned to $z_i$ at this time. We measure the test reliability as $\frac{1}{n}\sum_{i=1}^{n} \mathrm{I}(\hat{g}(z_i(\tau)) = \hat{g}(x_i))$, where $\hat{g}(x_i)$ is the label assigned to the complete data and $\mathrm{I}(\cdot)$ is one if the argument is true and zero otherwise. We also measure the average classification time as the mean of the $t_i(\tau)$. Ideally, we would like to classify with the smallest average classification time while still meeting reliability requirement $\tau$.

We perform incomplete classification experiments with two different classifiers. The first classifier is local QDA [20]. Local QDA learns the mean and covariance for the class $g$ discriminant function for test point $x$, $f_g(x)$, by estimating them using the $k$ nearest class $g$ training points to test point $x$. We choose $k \in \{1, 2, 4, 8, 16, 32, 64, 128\}$ by cross-validation on the training data. In our implementation of local QDA, we use a diagonal covariance matrix, and we regularize the covariance estimate by adding $10^{-4}\mathrm{I}$, where I is the identity matrix. Since we do not have the complete data $x$, we instead estimate the mean and covariance for $f_g(x)$ by finding the nearest class $g$ neighbors to the mean of $X$. The second classifier that we use is a linear SVM which we implement using LibSVM [6] with default settings.

## 7.2 Comparison of Construction of Sets of Measure $\tau$

We first compare the three set construction methods proposed Section 3, the Chebyshev set (3), the Gaussian näive Bayes quadratic set (4), and the Gaussian näive Bayes box set (5).

We vary the reliability parameter between four values $\tau = [0.001, 0.1, 0.25, 0.9]$, and we perform prediction using the jointly Gaussian model (19). Figure 6 plots the results for the Synthetic Control, Medical Images, and Two Patterns datasets. In all cases, the empirical reliability rate exceeds the reliability requirement $\tau$. Additionally, these plots verify that the Chebyshev set is the most conservative, as it waits the longest to classify the test data, and the näive Bayes quadratic set is the least conservative.

Table 2 compares the average testing time per test sample for the three different constraint sets when $\tau = 0.9$. This table shows that the näive Bayes box set is the least computationally complex, followed by the näive Bayes quadratic set, and finally the Chebyshev set.

| | Local QDA | | | Linear SVM | | |
|---|---|---|---|---|---|---|
| | Chebyshev | Quadratic | Box | Chebyshev | Quadratic | Box |
| Synthetic Control | 1.8 | 0.7 | 0.4 | 0.4 | 0.3 | 0.3 |
| Medical Images | 27.1 | 2.7 | 1.4 | 1.9 | 1.0 | 0.8 |
| Two Patterns | 12.45 | 2.0 | 1.0 | 1.8 | 0.5 | 0.3 |

Table 2: Average test time per sample, in seconds, for the three different constraint sets.

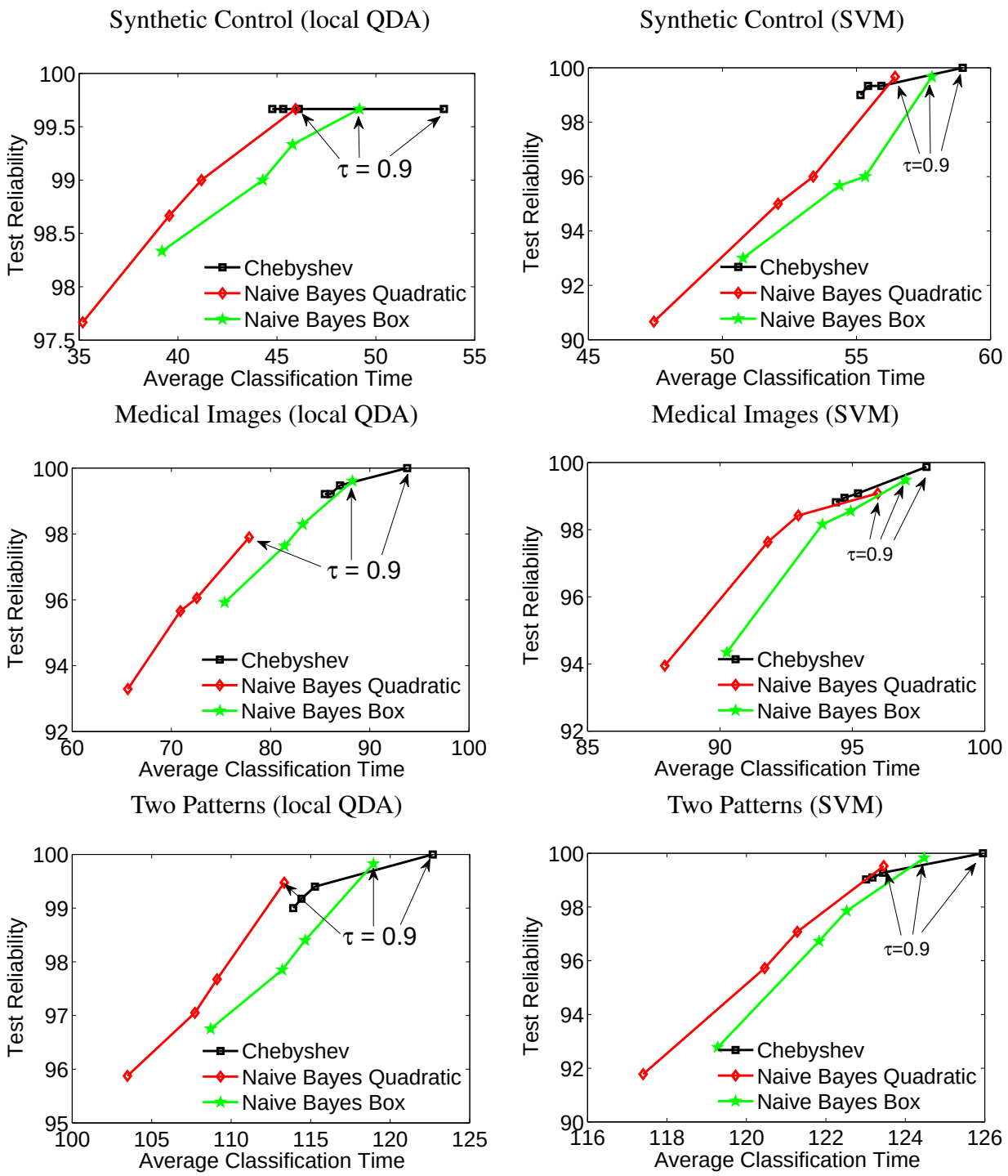


Figure 6: Average classification time vs test reliability for local QDA (left column) and linear SVM (right column) using jointly Gaussian prediction. The symbols correspond to choices of $\tau \in \{0.001, 0.1, 0.25, 0.9\}$.

## 7.3 Comparison of Estimation Methods

In this section we compare the performance of reliable incomplete classification using jointly Gaussian estimation (19) to that using GMM estimation (20). We use the same classifiers and values

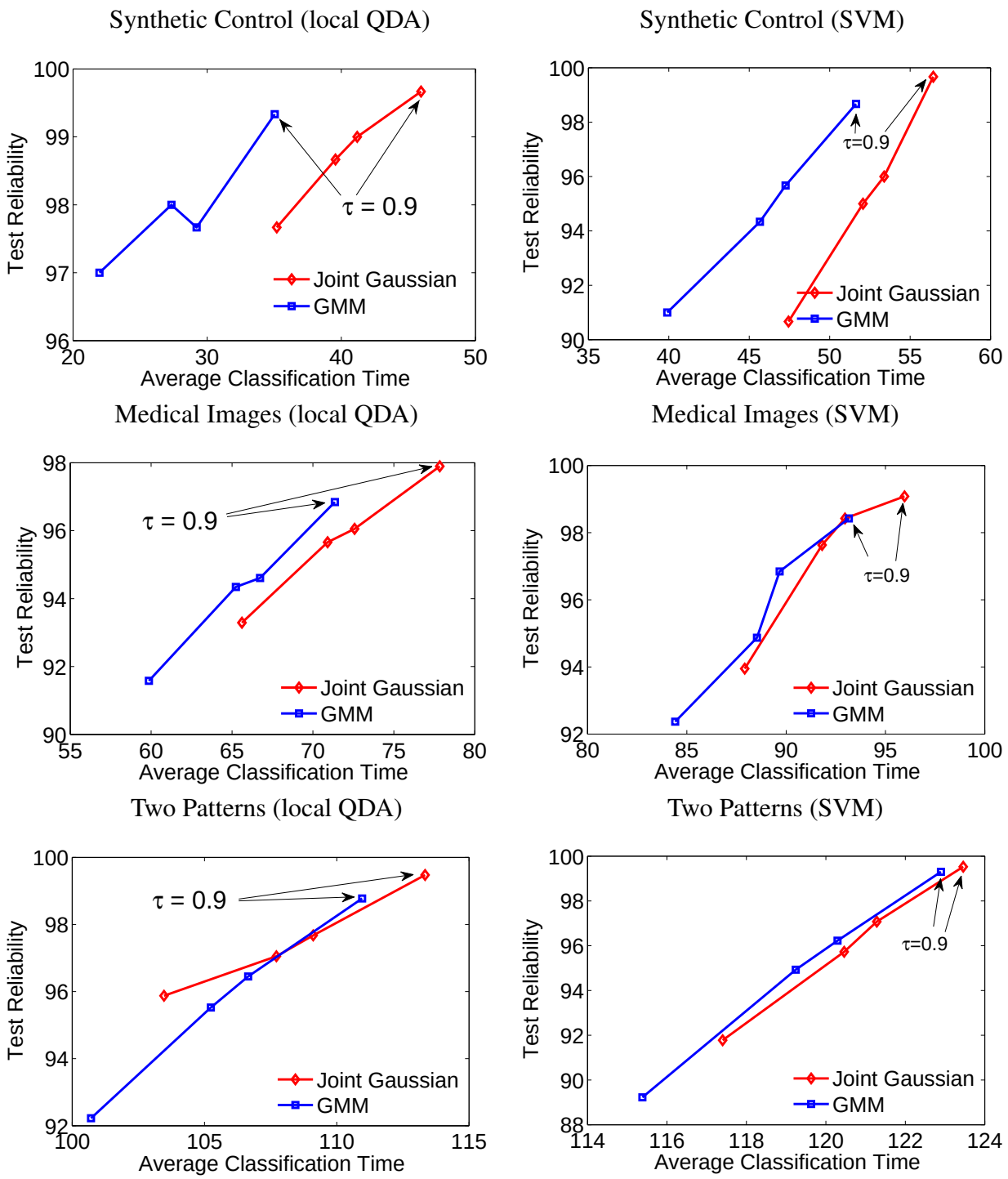


Figure 7: Average classification time vs test reliability for local QDA (left column) and linear SVM (right column) using the näive Bayes quadratic constraint set with $\tau$ varied between [0.001, 0.1, 0.25, 0.9].

for $\tau$ as given in the previous section.

| | Local QDA | | Linear SVM | |
|---|---|---|---|---|
| | Joint Gaussian | GMM | Joint Gaussian | GMM |
| Synthetic Control | 0.7 | 0.9 | 0.3 | 0.7 |
| Medical Images | 2.7 | 5.5 | 1.0 | 2.7 |
| Two Patterns | 2.0 | 3.4 | 0.5 | 0.8 |

Table 3: Average test time per sample, in seconds, for the two different estimation methods.

Figure 7 plots the average classification time vs. test reliability for the jointly Gaussian and GMM estimation methods using the näive Bayes quadratic constraint set. The figure shows that on the Synthetic Control and Medical Images datasets, the GMM method dominates the jointly Gaussian over all $\tau$ values for both classifiers. On the Two Patterns dataset with local QDA classification, the GMM method is not uniformly better than the jointly Gaussian method.

Table 3 compares the total testing time of the two approaches, and as expected, the GMM method requires more test time than the simpler jointly Gaussian method.

## 7.4 Dimensionality Reduction Features

An advantage of our reliable incomplete classification approach is that it can use any features derived from the data for which we can estimate the mean and covariance. As an example alternative to using the time-series samples as the features, we select a smaller feature set by first preprocessing the time-series using supervised linear dimensionality reduction. Linear dimensionality reduction finds a matrix $B \in \mathbb{R}^{\ell \times d}$, $\ell < d$ that maps the data from $d$-dimensional to $\ell$-dimensional space. Supervised dimensionality reduction uses the label information in the training data to find a reduced space where the data is also separated by class.

When using linear dimensionality reduction, the complete data becomes a vector $y = Bx \in \mathbb{R}^{\ell}$ as opposed to $x \in \mathbb{R}^{d}$. Let $Y$ be a random variable representing the distribution of the complete data. Due to the linear relationship $Y = BX$, it is straighforward to compute the mean and covariance parameters of $Y$ using the models described in Section 5.

Linear dimensionality reduction can provide two advantages over classifying the time-series features. First, it can diminish the impact of noisy or non-discriminative features in the time-series data, thus providing increased accuracy. Second, reducing the number of features reduces the computational complexity. For a time-series with $d$ samples, there are $d-t$ unknown samples at time $t$. Thus, if we simply use the time-series samples as the features for classification, the optimization problem that the reliable incomplete classifier must solve has $d-t$ free variables. For a long time series, this can cause the computational complexity to become extreme when $t$ is small. However, performing linear dimensionality reduction reduces the number of unknowns to $\ell$ which can greatly reduce the number of variables in the optimization for reliable classification.

We use local discriminative Gaussian (LDG) dimensionality reduction [32] to learn $B$. We choose LDG dimensionality reduction because 1) it can separate multi-modal data, 2) the solution

| Dataset | Time-series length | Number of LDG features | Test time at t=1 (ms) | LDG test time at t=1 (ms) |
|---|---|---|---|---|
| Chlorine Concentration | 166 | 42 | 76 | 4 |
| Italy Power Demand | 24 | 2 | 2 | 1 |
| Face (All) | 131 | 30 | 40 | 2 |
| Medical Images | 99 | 11 | 18 | 2 |
| Non-Invasive Fetal ECG 1 | 750 | 30 | 6,107 | 4 |
| Non-Invasive Fetal ECG 2 | 750 | 23 | 5,789 | 3 |
| Starlight Curves | 1024 | 26 | 15,697 | 2 |
| Swedish Leaf | 128 | 20 | 35 | 2 |
| Synthetic Control | 60 | 7 | 9 | 1 |
| Two Patterns | 128 | 22 | 31 | 2 |
| U Wave Gesture Library X | 315 | 12 | 418 | 2 |
| U Wave Gesture Library Y | 315 | 6 | 382 | 1 |
| U Wave Gesture Library Z | 315 | 10 | 393 | 2 |
| Wafer | 152 | 17 | 55 | 2 |
| Yoga | 426 | 26 | 945 | 2 |

Table 4: Time-series length and the number of features after LDG dimensionality reduction as well as a comparison of the testing time, in milliseconds, required to perform reliable local QDA classification with the näive Bayes quadratic constraint set and jointly Gaussian estimation. The test time shown measures the average time, per test sample, to perform reliable classification at time $t = 1$. Therefore, this is a worst case test time in terms of real-time performance as the number of unknowns in the optimization problem for reliable classification is maximized at time $t = 1$.

is fast, requiring only a maximal eigenvalue decomposition, and 3) it has been shown to work well even when few training samples are provided and the input dimensionality is large. Furthermore, we can choose the best input dimensionality by performing cross-validation on the training dataset to find a reduced space that is both small and accurate. Table 4 shows the dimensionality of the training data after LDG dimensionality reduction. The table also compares the testing time required to perform reliable local QDA classification with the näive Bayes quadratic constraint set with jointly Gaussian estimation at time $t = 1$ with and without LDG dimensionality reduction. On the datasets with more than three hundred time-series samples, using LDG dimensionality reduction results in an orders of magnitude decrease in the testing time.

## 7.5 Comparison to Other Methods

In this section, we compare the performance of our reliable incomplete data classifier to ECTS[2] [49] and several baselines. For all experiments in this section, we use the näive Bayes quadratic constraint set because it proved to be uniformly better than the box constraint set across all experiments in Section 7.2, while not being as overly conservative as the Chebyshev set. We also use the jointly Gaussian estimation method as it is faster to compute than the GMM method, particularly for the long time-series with many classes (the three U Wave Gesture Library datasets and two Non-Invasive Fetal ECG datasets). We also only show results for local QDA, as the reliable local QDA classifier classified earlier than reliable SVM in all experiments of Sections 7.2 and 7.3.

ECTS trades off between the objectives of classifying early and ensuring that early labels meet final labels by using a parameter, MS, that varies between zero and one, with zero resulting in the earliest classification time. However, we emphasize that this parameter is not the same as our reliability parameter $\tau$, in that it provides no guarantee on reliability of the early predictions, but is instead a knob that the user can tune to trade off between earliness and reliability. [49] set this parameter to 0 in the majority of their experiments. We compare to ECTS by varying this parameter $\text{MS} \in \{0, 0.05, 0.1, 0.2, 0.4, 0.8\}$.

We also compare to the performance of two baseline methods that we call *Fixed-time local QDA* and *Fixed-time 1-NN*. These methods use no predictive power, but instead classify all test signals at some user specified time: $t$ samples.

The reliability results are shown in Figures 8 and 9. Reliable incomplete local QDA classification and reliable incomplete local QDA classification with LDG features perform well across all experiments. Reliable local QDA using the time series samples as features (the red line in the plots) dominates fixed-time local QDA in test reliability for all datasets over all values of $\tau$. The only times that reliable local QDA with LDG features fails to dominate fixed-time local QDA are when $\tau = 0.001$ on the Italy Power Demand, U Wave Gesture Library Y, and Wafer datasets.

We also note that the leftmost pink circle in these plots is the earliest possible average classification time that ECTS can achieve, as $\text{MS} = 0$ is the smallest possible value for the minimum support parameter. On the other hand, reliable early classification can achieve earlier times than

[2]The authors graciously provide code for ECTS at http://zhengzhengxing.blogspot.com/p/research.html.

those shown in the figures by setting $\tau < 0.001$. Therefore, if someone wanted to set $\tau$ by cross-validation on the training dataset, the reliable incomplete classifier offers more flexibility than ECTS.

Figures 10 and 11 plot the test accuracy of the various approaches. The accuracy plots show that for some datasets, local QDA achieves higher accuracy than 1-NN; therefore, ECTS suffers in comparison to reliable local QDA due to the fact that it attempts to match a less accurate classifier.

The accuracy plots also show that although ECTS is typically more reliable than fixed-time 1-NN, it is less accurate for at least one value of MS on twelve of the fourteen datasets. On the other hand, reliable local QDA using the time-series samples as features is less accurate than fixed-time local QDA on only the Medical Images and Chlorine Concentration datasets. However, on the Chlorine Concentration dataset, reliable local QDA with LDG features exceeds the accuracy of fixed-time local QDA. Furthermore, we found that reliable local QDA classification using GMM estimation (not shown) exceeds the accuracy of fixed-time local QDA on the Medical Images dataset. The proposed reliable classification approach can be used with a wide variety of features, classifiers, and estimation methods in order to maximize accuracy for a particular application.

Finally, the accuracy in the Starlight Curves dataset of Figure 11 demonstrates an important consideration for ECTS and reliable local QDA. Both of these methods try to match the labels that are assigned by the respective fixed-time classifiers at time 1024. However, the fixed-time local QDA plot shows that the underlying local QDA accuracy remains essentially flat from time two hundred to time 1024, and fixed-time 1-NN shows that accuracy decreases. Therefore, both incomplete classifiers would be better served by treating the complete data as only the first two hundred samples rather than all 1024.

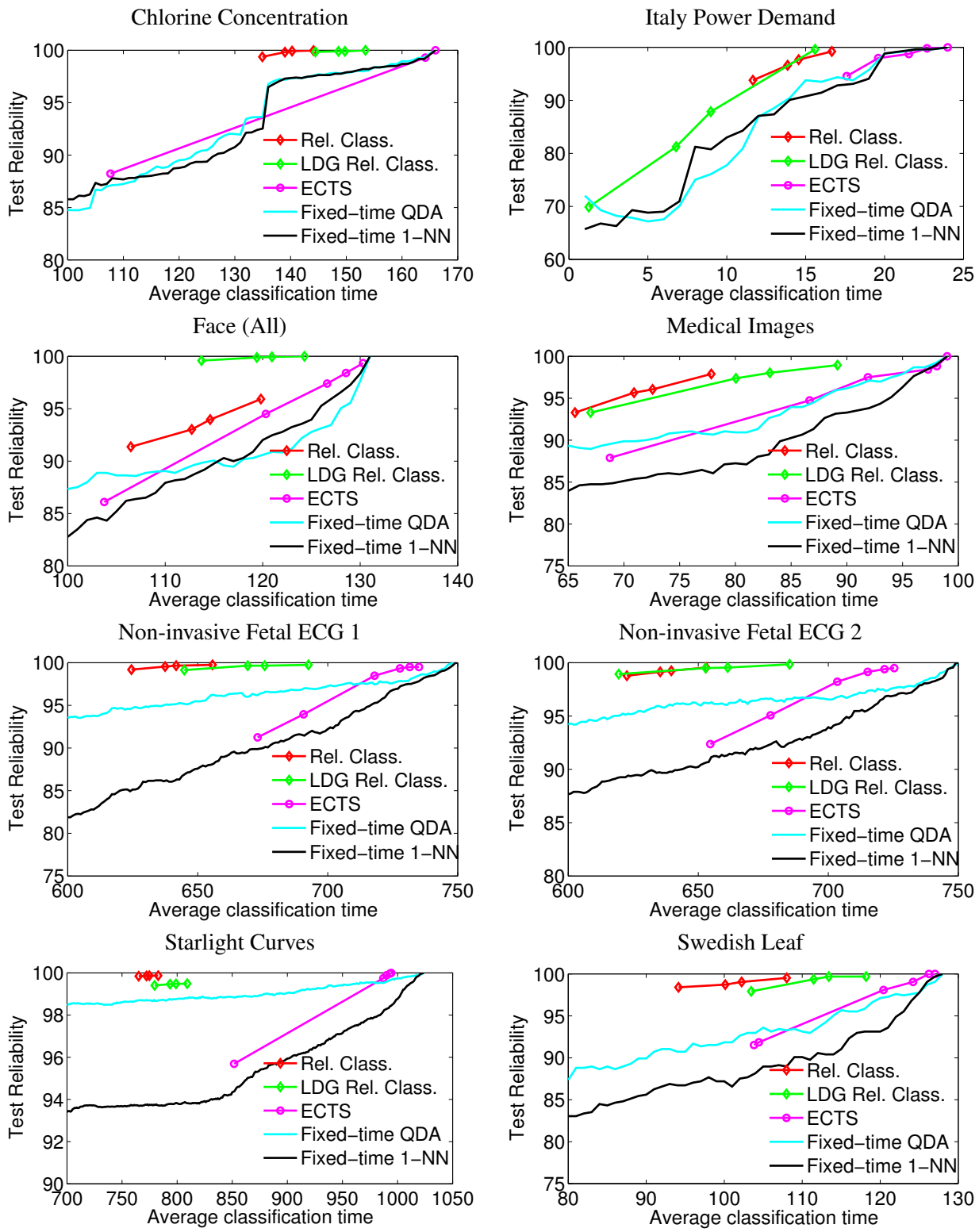


Figure 8: Average classification time vs test reliability for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN.

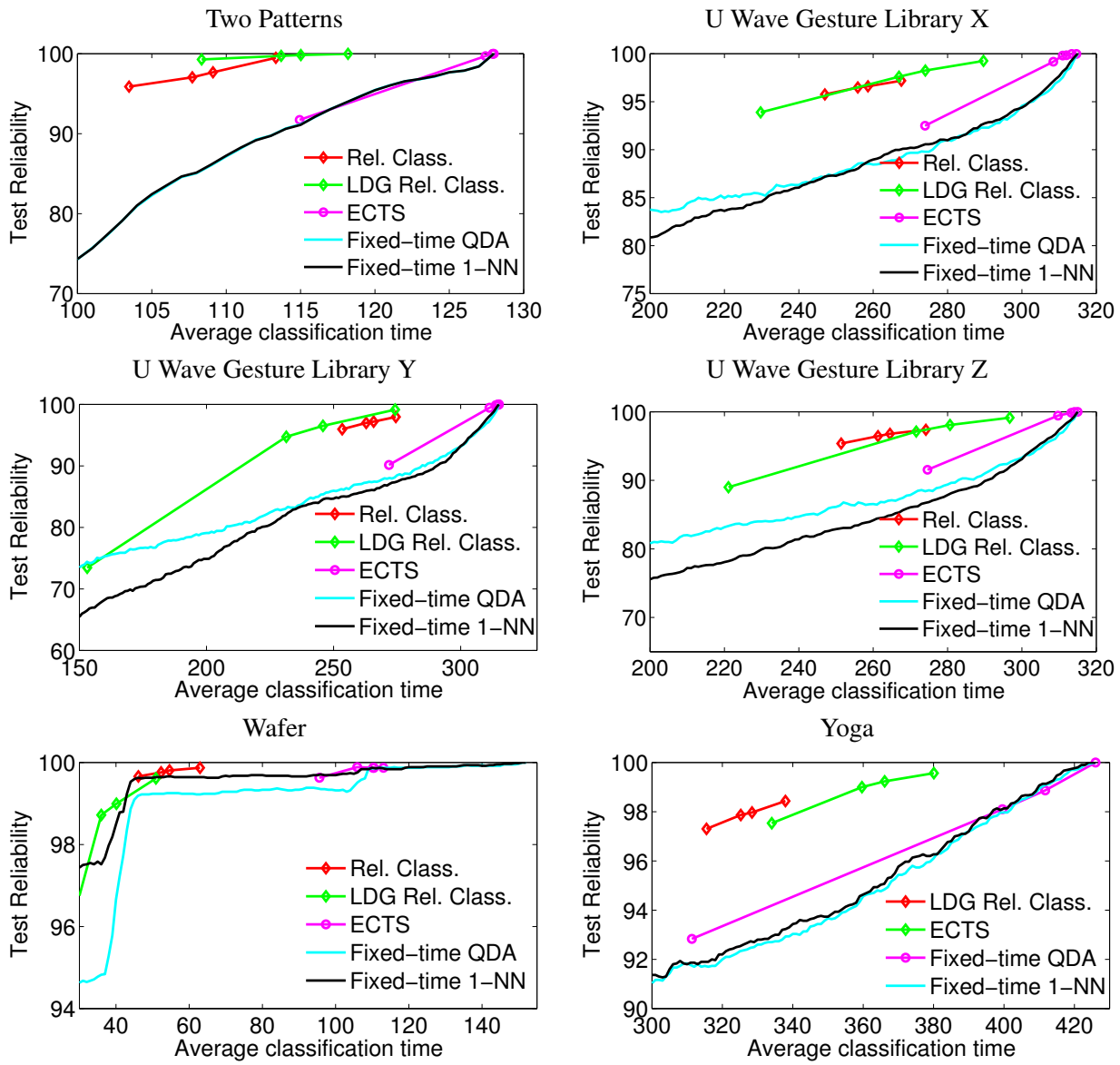


Figure 9: Average classification time vs test reliability for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN.

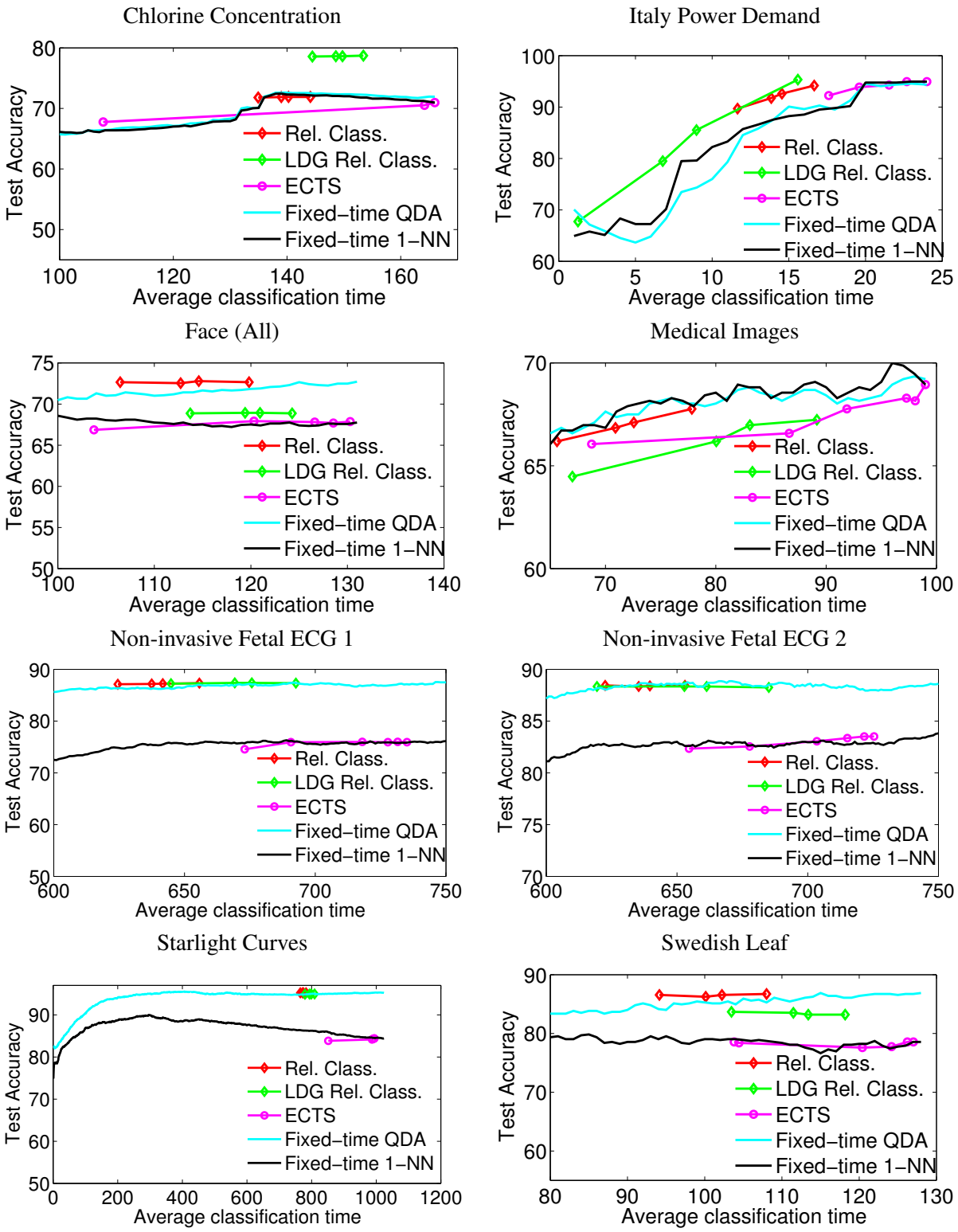


Figure 10: Average classification time vs test accuracy for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN.

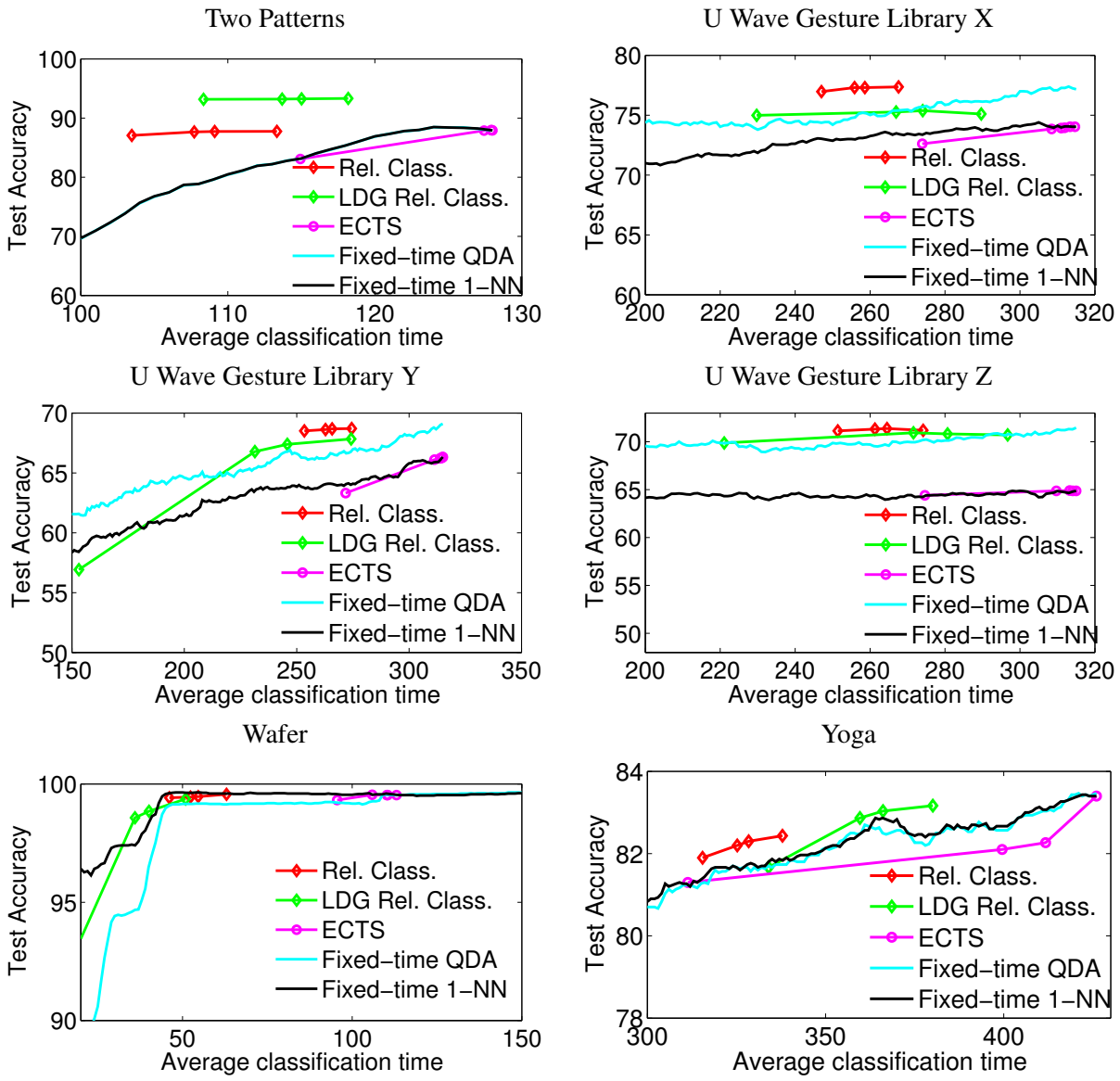


Figure 11: Average classification time vs test accuracy for reliable incomplete local QDA classification (Rel. Class.), reliable incomplete local QDA classification with LDG features (LDG Rel. Class.), ECTS, Fixed-time local QDA, and Fixed-time 1-NN.

# 8 Discussion and Some Open Questions

We have proposed a practical incomplete decision rule that is a conservative approximation of the optimal rule. Experiments on a set of time-series data showed consistently earlier and more reliable predictions on average than other approaches. We showed that for linear or quadratic classifiers the proposed decision rule can be checked either with an analytic solution or using convex optimization. We only touched on applying the proposed rule to nearest neighbor classifiers, and it is an open question how to apply this approach efficiently to other classification strategies. In particular, we suspect the proposed approach could also be implemented efficiently with decision trees that use a cascade of linear discriminants by casting each decision into a form similar to (8).

This paper has focused on answering the question "With probability $\tau$, will the classification decision from this incomplete data be the same as from the complete data?" The presented tools can also be used to answer the related question: "If we classify based on the current incomplete data, what is the probability that assigned label will match that which would be chosen using the complete data?" The answer can be computed by finding the largest $\tau$ that makes the first question a "Yes," which may require guessing a $\tau$, solving the first question, refining $\tau$ up or down depending on the answer, and iterating.

Another related question that can be answered is, "Can we reliably classify as class $g$ with this incomplete data?" That is, there may be only one class (or a subset of classes) which we would like to identify with incomplete data. For example, in determining if a cyst is cancerous or benign, doctors will often have a patient come back every few months to see how it changes over time. There is generally no rush to call it benign, but one would like to classify it as cancerous as soon as that is a reliable class label. This question can be answered by applying the incomplete decision rule given in (2) only to the class of interest.

# References


[1] H. S. Anderson, N. Parrish, K. Tsukida, and M. R. Gupta. Reliable early classification of time series. In *IEEE Intl. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.

[2] P. Armitage. Sequential analysis with more than two alternative hypotheses, and its relation to discriminant function analysis. *J. Royal Statistical Society. Series B (Methodological)*, 12(1):137–144, Jan. 1950.

[3] A. Barla, F. Odone, and A. Verri. Histogram intersection kernel for image classification. *Intl. Conf. Image Processing (ICIP)*, 3:III–513–16 vol.2, 2003.

[4] S. Bengio, J. Weston, and D. Grangier. Label embedding trees for large multi-class tasks. In *NIPS*, 2010.

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2008.

[6] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[7] M. Cooke, P. Green, L. Josifovski, and A. Vizinho. Robust automatic speech recognition with missing and unreliable acoustic data. *Speech Communication*, 34(3):267 – 285, 2001.

[8] M. C. Costanza and A. A. Afifi. Comparison of stopping rules in forward stepwise discriminant analysis. *J. American Statistical Association*, 74(368):777–785, Jan. 1979.

[9] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *J. Machine Learning Research*, 2001.

[10] K. Crammer and Y. Singer. Ultraconservative online algorithms for multiclass problems. *J. Machine Learning Research*, 3:951–991, 2003.

[11] J. Deng, A. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *European Conf. Computer Vision (ECCV)*, 2010.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conf. Computer Vision Pattern Recognition (CVPR)*, 2009.

[13] J. Deng, S. Satheesh, A. C. Berg, and L. Fei-Fei. Fast and balanced: Efficient label tree learning for large scale object recognition. In *NIPS*, 2011.

[14] M. Dredze, K. Crammer, and F. Pererira. Confidence-weighted linear classification. *Intl. Conf. Machine Learning (ICML)*, 2008.

[15] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Machine Learning Research*, 12:2121–2159, 2011.

[16] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, 1998.

[17] T. Ferguson. *Optimal Stopping Rules and Applications*. E-book available on the author's website., 2001.

[18] K. S. Fu. *Sequential methods in pattern recognition and machine learning (Mathematics in science and engineering)*. Academic Press, 1st edition, 1968.

[19] W. J. Fu, E. R. Dougherty, B. Mallick, and R. J. Carroll. How many samples are needed to build a classifier: a general sequential approach. *Bioinformatics*, 21(1):63–70, Jan. 2005.

[20] E. K. Garcia, S. Feldman, M. R. Gupta, and S. Srivastava. Completely lazy learning. *IEEE Trans. Knowledge and Data Engineering*, 22(9):1274 –1285, Sept. 2010.

[21] D. Grangier and S. Bengio. A discriminative kernel-based model to rank images from text queries. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(8):1371–1384, 2008.

[22] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 1.21. `http://cvxr.com/cvx/`, Apr. 2011.

[23] K. Grauman and T. Darrell. The pyramid match kernel: Efficient learning with sets of features. *J. Machine Learning Research*, 8:725–760, Apr. 2007.

[24] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, New York, 2001.

[25] C. Hsu and C. Lin. A comparison of methods for multiclass support vector machines. *IEEE Trans. Neurel Networks*, 13(2):415–425, 2002.

[26] E. Keogh, Q. Zhu, B. Hu, Hao. Y., X. Xi, L. Wei, and C. A. Ratanamahatana. The UCR time series classification/clustering homepage. www.cs.ucr.edu/˜eamonn/time_series_data/, 2011.

[27] Y. LeCun, L. Bottou, G. Orr, and K. Mueller. Efficient backprop. In G. Orr and K. Mueller, editors, *Neural Networks: Tricks of the trade*. Springer, 1998.

[28] T. Leung and J. Malik. Recognizing surface using three-dimensional textons. *Intl. Conf. Computer Vision (ICCV)*, 1999.

[29] A. Makadia, V. Pavlovic, and S. Kumar. A new baseline for image annotation. In *European Conf. Computer Vision (ECCV)*, 2008.

[30] J. M. Martinez. Local minimizers of quadratic functions on Euclidean balls and spheres. *SIAM J. on Optimization*, 4(1):159 – 176, 1994.

[31] S. Pang, S. Ozawa, and N. Kasabov. Incremental linear discriminant analysis for classification of data streams. *IEEE Trans. Systems, Man, and Cybernetics*, 35(5):905–914, Oct. 2005.

[32] N. Parrish and M. R. Gupta. Dimensionality reduction by local discriminative Gaussians. In *Intl. Conf. Machine Learning (ICML)*, 2012.

[33] B. Raj, M. L. Seltzer, and R. M. Stern. Reconstruction of missing features for robust speech recognition. *Speech Communication*, 43(4):275 – 296, 2004.

[34] B. Raj and R. M. Stern. Missing-feature approaches in speech recognition. *IEEE Signal Processing Magazine*, 22(5):101 – 116, 2005.

[35] Robert Kalaba David Middleton Richard Bellman. Dynamic Programming, Sequential Estimation and Sequential Detection Processes. *Proc. National Academy of Sciences of the United States of America*, 47(3):338, Mar. 1961.

[36] J. J. Rodriguez and C. J. Alonso. Boosting interval-based literals: Variable length and early classification. In *ECAI'02 Workshop on Knowledge Discovery from (Spatio-) Temporal Data*, 2002.

[37] A. Rogier, T. Donders, G. J.M.G. van der Heijden, T. Stijnen, and K. G.M. Moons. Review: A gentle introduction to imputation of missing values. *J. Clinical Epidemiology*, 59(10):1087 – 1091, 2006.

[38] J. Sanchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In *CVPR*, 2011.

[39] J. L. Schafer and J. W. Graham. Missing data: Our view of the state of the art. *Psychological Methods*, 7(2):147 – 177, 2002.

[40] B. Schoelkopf, A. J. Smola, and K. R. Müller. Kernel principal component analysis. *Advances in kernel methods: Support vector learning*, pages 327–352, 1999.

[41] S. Srivastava, M. R. Gupta, and B. A. Frigyik. Bayesian quadratic discriminant analysis. *J. Machine Learning Research*, 8:1287–1314, 2007.

[42] J. Sturm. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11:625–653, 1999.

[43] P. D. Tao and L. T. H. An. Convex analysis approach to D.C. programming: Theory, algorithms, and applications. *ACTA Mathematica Vietnamica*, 22(1):289 – 355, 1997.

[44] N. Usunier, D. Buffoni, and P. Gallinari. Ranking with ordered weighted pairwise classification. In *ICML*, 2009.

[45] A. Wald. Sequential tests of statistical hypotheses. *Annals of Mathematical Statistics*, 16:117–186, 1945.

[46] A. Wald. *Sequential Analysis*. John Wiley, 1947.

[47] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Intl. Joint Conf. Artificial Intelligence, (IJCAI)*, pages 2764–2770, 2011.

[48] J. Weston and C. Watkins. Support vector machines for multi-class pattern recognition. In *Proc. European Symposum on Artificial Neural Networks*, 1999.

[49] Z. Xing, J. Pei, and P. S. Yu. Early prediction on time series: A nearest neighbor approach. In *IJCAI*, pages 1297–1302, 2009.

# A Proofs

**Proof of Proposition 1:** For the minimum problem (11), the Lagrange dual function is $g(\lambda) = \beta^T m - \frac{1}{4\lambda} B^T RB + b - \lambda\delta$, a concave function of $\lambda$, and $g(\lambda)$ is maximized for $\lambda^* = \sqrt{\frac{1}{4\delta} B^T RB}$. Since $\lambda^* \geq 0$, it is dual feasible. Since the objective function is convex, strong duality holds, and thus the maximum of the dual problem equals the minimum of the primal problem. A similar analysis can be performed for the maximum problem.

**Proof of Proposition 2:** First, note that each pairwise check reduces the number of classes labeled `candidate` by either two classes if the classes tie, or by one class (the loser) if one class dominates. Second, once a class has tied with another class or has been dominated, it cannot be the correct dominating class. Thus the proposed decision process eventually reduces the number of classes labeled `candidate` to either zero or one. If there are zero classes left labeled `candidate`, then all classes have either tied or been dominated, and the above process correctly chooses not to classify. If there is one class remaining that is labeled `candidate` it must be compared to all the classes that tied on their first comparison. It is not necessary to also compare to the classes labeled `dominated` by the transitivity of the domination rule.

**Proof of Proposition 3:** We first note that in the *Compare* step, the pairwise comparison between classes $c$ and $h$ requires a single minimum computation if $c$ dominates $h$, and two computations if the classes tie or if $h$ dominates $c$. Furthermore, we define $T$ to be the number of pairwise comparisons that result in ties during the *Compare* step, and $D$ as the number of pairwise comparisons that do not result in a tie in the *Compare* step (such evaluations necessarily result in one class that was labeled `candidate` being re-labeled `dominated`). Thus, the compare step requires at most $2T + 2D$ minimum calculations.

On the other hand, each pairwise comparison in the *Final Comparison* check requires only a single minimum computation.

There are two cases to consider

*Case 1:* Consider the case that the *Compare* step in the decision process results in one class left labeled `candidate`. Immediately prior to the *Final Comparison* step, there are $G-1$ classes that have been re-labeled `tied` or `dominated`, and since each tie results in two classes being re-labeled `tied`, it must be that

$$G - 1 = D + 2T. \tag{22}$$

In the *Final Comparison* step, the $G$th class must be compared to at most the $2T$ classes labeled `tied`, each of which requires one minimum calculation. Thus the maximum number of calculations

needed is

$$\begin{aligned}&2T+2D+2T\\&=2(G-1) \text{ by (22).}\end{aligned}$$

Conversely, the best case is that there are no ties, and that each pairwise check requires only a single minimum calculation. This case requires $G-1$ minimum calculations.

*Case 2:* The second case is that at the end of the *Compare* step there are zero classes labeled `candidate`. Therefore,

$$G=D+2T, \tag{23}$$

and the total number of comparisons required is

$$\begin{aligned}&2T+2D\\&=G+D \text{ by (23).}\end{aligned}$$

There can be at most $G-2$ comparisons that result in one class dominating the other (otherwise, one class would remain labeled `candidate` after the *Compare* step), so the maximum number of minimum calculations is again $2(G-1)$.

Since it requires at least one minimum calculation to change a class label from `candidate` to `dominated` or `tied`, the minimum number of calculations is $G$.

# B Gradient Descent Solution for the Quadratic Min and Max Problems

The min problem with quadratic $f(x)$ subject to a quadratic constraint set is written as

$$\min_{x \in A} f(x) = \min_{x} (x-v)^T V (x-v) + b \qquad (24)$$
$$\text{s.t. } (x-m)R^{-1}(x-m) \leq \delta,$$

where $V$ is indefinite and $R$ is positive semi-definite. We propose to solve this problem using the two-step gradient descent approach described in [43].

We first reformulate (24) as the *trust region subproblem* (TRSP). Define

$$\begin{aligned} z &= R^{\frac{-1}{2}}(x-m) \\ A &= 2R^{\frac{1}{2}} V R^{\frac{1}{2}} \\ y &= 2R^{\frac{1}{2}} V (m-v) \\ b_{tsrp} &= b + v^T V v + m^T V m - 2m^T V v. \end{aligned}$$

Then rewrite (24) as

$$\min_{x \in A} f(x) = \min_{z} \frac{1}{2} z^T A z + y^T z + b_{tsrp} \qquad (25)$$
$$\text{s.t. } \| z \| \leq \sqrt{\delta}.$$

Let $\rho$ equal the largest eigenvalue of $A$. The following two-step iteration converges to a $z^*$ that is a local minimum of the TRSP (25):

$$\begin{aligned} &\text{Step 1 :} \quad z_{k+1} = z_k - \frac{1}{\rho}(A z_k + y) \\ &\text{Step 2 :} \quad z_{k+1} = \min \left[ z_{k+1}, \frac{\| z_{k+1} \|}{\sqrt{\delta}} z_{k+1} \right], \end{aligned}$$

where Step 1 computes a gradient step, and Step 2 projects the $z_{k+1}$ found in Step 1 onto the constraint set in (25).

The TRSP has been shown to have at most one local minimum that is not also the global minimum [30], and thus the above algorithm has proven to be robust in finding the minimum of (25).

Furthermore, for the incomplete data decision rule (9), it is not necessary to find the true minimum over $A$ of $f(x)$, but it is instead sufficient to know only whether or not it is less than or equal to zero. Therefore, the iteration can be stopped early if $z_k^T A z_k + y^T z_k + b_{tsrp} \leq 0$.

# C Derivation of the Variance for GMM Based Estimation

Let $m_g = \mathrm{E}[X|g,z]$, $R_g = \mathrm{COV}[X|g,z]$, and $p(g|z)$ be defined as in Section 5.2.

$$
\begin{aligned}
R &= \int_x (x-m)(x-m)^T \; p(x|z)\, dx \\
&= \int_x \sum_{g\in\mathscr{G}} (x-m)(x-m)^T \; p(x,g|z)\, dx \\
&= \sum_{g\in\mathscr{G}} p(g|z) \int_x (x-m)(x-m)^T \; p(x|g,z)\, dx \\
&= \sum_{g\in\mathscr{G}} p(g|z) \int_x \left( xx^T - 2x \sum_{h\in\mathscr{G}} m_h^T p(h|z) + \sum_{q\in\mathscr{G}} \sum_{h\in\mathscr{G}} m_q m_h^T p(q|z) p(h|z) \right) \; p(x|g,z)\, dx \\
&= \sum_{g\in\mathscr{G}} p(g|z) \left( \int_x xx^T - 2xm_g^T + m_g m_g^T \; p(x|g,z)\, dx + \int_x 2xm_g^T - 2x \sum_{h\in\mathscr{G}} m_h^T p(h|z) \; p(x|g,z)\, dx \right. \\
&\quad \left. -m_g m_g^T + \sum_{q\in\mathscr{G}} \sum_{h\in\mathscr{G}} m_q m_h^T p(q|z) p(h|z) \right) \\
&= \sum_{g\in\mathscr{G}} p(g|z) \left( R_g + 2m_g m_g^T - 2m_g \sum_{h\in\mathscr{G}} m_h^T p(h|z) - m_g m_g^T + \sum_{q\in\mathscr{G}} \sum_{h\in\mathscr{G}} m_q m_h^T p(q|z) p(h|z) \right) \\
&= \sum_{g\in\mathscr{G}} p(g|z) \left( R_g + m_g m_g^T - 2m_g \sum_{h\in\mathscr{G}} m_h^T p(h|z) \right) + \sum_{q\in\mathscr{G}} \sum_{h\in\mathscr{G}} m_q m_h^T p(q|z) p(h|z) \\
&= \sum_{g\in\mathscr{G}} p(g|z) \left( R_g + m_g m_g^T \right) - \sum_{q\in\mathscr{G}} \sum_{h\in\mathscr{G}} m_q m_h^T p(q|z) p(h|z).
\end{aligned}
$$

# DISTRIBUTION:

| | | |
|---|---|---|
| 1 | MS 0899 | Technical Library, 9536 (electronic copy) |
| 1 | MS 0359 | D. Chavez, LDRD Office, 1911 |

Sandia National Laboratories