

Nonparametric supervised learning by linear interpolation with maximum entropy

Maya R. Gupta, Robert M. Gray, Richard A. Olshen

Abstract

Nonparametric neighborhood methods for learning entail estimation of class conditional probabilities based on relative frequencies of samples that are ‘near-neighbors’ of a test point. We propose and explore the behavior of a learning algorithm that uses linear interpolation and the principle of maximum entropy (LIME). We consider some theoretical properties of the LIME algorithm: LIME weights have exponential form, the estimates are consistent, and the estimates are robust to additive noise. In relation to bias reduction, we show that near-neighbors contain a test point in their convex hull asymptotically. The common linear interpolation solution used for regression on grids or look-up-tables is shown to solve a related maximum entropy problem. LIME simulation results support use of the method, and performance on a pipeline integrity classification problem demonstrates that the proposed algorithm has practical value.

Index Terms

nonparametric statistics, probabilistic algorithms, pattern recognition, maximum entropy, linear interpolation

I. THE SUPERVISED LEARNING PROBLEM

WE observe a labeled set or sequence of correlated pairs $\{(X_i, Y_i); i = 1, \dots, n\}$, where the Y_i are scalar labels of the d -dimensional real vectors X_i . Typically, each random object in the sequence is assumed to be drawn independently according to a common distribution P_X or P_{XY} . We

M. R. Gupta is with the Department of Electrical Engineering, University of Washington, Seattle, WA 98195. R. M. Gray is with the Information Systems Lab, Department of Electrical Engineering, Stanford University, Stanford, CA 94305. R. A. Olshen is with the Department of Health, Research and Policy, and by courtesy also with the Departments of Electrical Engineering and Statistics, Stanford University, Stanford, CA 94305.

This work was supported by National Science Foundation Grants CCR-0073050 and MIP-9706284, by a National Science Foundation Graduate Fellowship, by NIBIB/NIH Grant 5RO1 EB002784, and by Norsk Electro Optikk.

assume that P_X is absolutely continuous with respect to Lebesgue measure and hence has a well-defined density function f . Suppose then that another sample vector $X = x$ is drawn from P_X ; several questions might be asked:

- What is the best estimate $\hat{f}(x)$ of the true but unknown probability density function at x , $f(x)$? (*density estimation*)
- What is the minimum mean-squared estimate $\hat{Y}(x)$ of some correlated random variable Y ? Equivalently, what is the best estimate of the conditional expectation $\hat{E}(Y|X = x)$? (*regression or estimation*)
- What is the maximum a posteriori guess of the value of Y , or, equivalently, $\operatorname{argmax}_y \hat{P}_{Y|X}(y|x)$? (*statistical classification or detection*)

The second and third questions above extend easily to general cost functions to define minimum average Bayes risk regression and classification. In all three cases the estimates implicitly depend on the training set. The statistical literature on all three problems is extensive, and often the problems are treated together because of their similarities.

A standard approach makes use of a “kernel,” wherein a real-valued (often nonnegative) function $K(u)$ is defined on \mathcal{R}^d , constrained to have unit integral. With the traditional kernel approach there is a scale $h > 0$ (the “bandwidth”) with the resulting modified kernels $K_h(u) = K(u/h)/h^d$. Dependence on n is suppressed in the notation unless required for clarity. There results an estimate of the density,

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - X_i), \quad (1)$$

and of the regression function

$$\hat{Y}(x) = \frac{\frac{1}{n} \sum_{i=1}^n K_h(x - X_i) Y_i}{\hat{f}(x)}, \quad (2)$$

and a classifier

$$\operatorname{argmax}_y \frac{1}{n} \sum_{i=1}^n K_h(x - X_i) 1(Y_i = y), \quad (3)$$

where $1(F)$ is the indicator function that is 1 if F is true and 0 otherwise. Regression and classification are often termed “supervised statistical learning” or “pattern recognition” [1]. Here we assume simply that the training vectors are random vectors in d -dimensional Euclidean space with well-behaved distributions,

and that P_Y is absolutely continuous in the regression problem and discrete in the classification problem.

There are many variations and extensions of kernel methods, a common one being to adapt the kernel locally, either to the observation x or to the training vectors X_i . One common method is to fix the radius R or volume V of a sphere around the point x and only include in the sums those training vectors X_i that fall inside the sphere. This ensures that only points close to x influence the estimate. In this case the scaling h is usually chosen to be proportional to $(k/(nV))^{1/d}$, where k is the number of neighbors in the volume V . The radius of the sphere can be adapted to the observed point x . This approach is still basically a fixed kernel method, but the kernel is truncated so as to consider only training vectors within a fixed distance of the observation.

Loftsgaarden and Quesenberry [2] introduced an alternative approach to density estimation based on the nearest-neighbor approach of Fix and Hodges [3]. They fixed the number k of nearest neighbors to be used in the estimate. In the fixed k case, the volume V is the volume of the sphere with x at the center and radius $R(x)$ equal to the distance from x to its k th nearest neighbor in the training set. This yields a density estimator of the form

$$\hat{f}(x) = \frac{1}{nR(x)^d} \sum_{i=1}^n K\left(\frac{x - X_i}{R(x)}\right), \quad (4)$$

which is the general multivariate k -nearest neighbor (kNN) density estimate studied by Mack and Rosenblatt [4], who generalized the Loftsgaarden and Quesenberry example which implicitly used a uniform kernel $K(x) = 1(|x| \leq 1)$. Unfortunately, even if K integrates to 1, the estimator given in (4) does not integrate to 1; rather the integral increases without bound as the distance from x to the origin increases.

These kNN approaches can be considered as variable kernel or adaptive kernel methods as defined in (6.78) of Scott [5]:

$$\hat{f}(x) = \frac{1}{nh_x^d} \sum_{i=1}^n K\left(\frac{x - X_i}{h_x}\right), \quad (5)$$

where the ‘‘adaptive’’ bandwidth or scaling function or smoothing function h_x depends on the observed vector x (and the training sequence). For the Loftsgaarden and Quesenberry method, h_x is simply the distance from the observation to the k th nearest neighbor. Another approach to kNN for regression is the weighted generalized nearest neighbors approach of Stone [6], who assumed a weighting sequence

$w = w_n(x) = \{w_{ni}(x); i = 1, \dots, n\}$ for which $\sum_{i=1}^n w_{ni}(x) = 1$ and formed an estimate

$$\hat{Y}(x) = \sum_{i=1}^n w_{ni}(x)Y_i. \quad (6)$$

If the weighting function assigns zero weight to any training vectors farther away than the k th nearest neighbor, then this is another way of defining a nonuniformly weighted kNN estimate. Stone provides general conditions under which such weighted-average estimators give universally consistent estimates. The kernel approach and the weighted approach are immediately seen to be equivalent in the case where the weights are related to the kernels by

$$w_{ni}(x) = \frac{K\left(\frac{x-X_i}{R(x)}\right)}{\sum_{j=1}^n K\left(\frac{x-X_j}{R(x)}\right)}, \quad (7)$$

which yields a valid weight function, in the sense that the weight function sums to 1. The discrete weighting w and the continuous argument kernel related in this way together entail estimators of a density function and regression function, and also a classifier.

There is a large literature using assumptions made either on the kernels or on the weighting functions and demonstrating a variety of properties of such estimators under those assumptions. Properties of particular importance include universal consistency, robustness to noise, and the bias and variance of the estimators.

It is standard in kernel design to make the kernel symmetric so that the first moment of the kernel is zero,

$$\int_{\mathcal{R}^d} uK(u)du = 0, \quad (8)$$

and the second moment is finite. We propose adaptive kernels that do not have zero first moment, and we will argue that the asymmetric kernel considered provides a simple, easily computable, and effective estimator. Simulations and heuristics support the claim that such asymmetric kernels may be superior to symmetric kernels in kNN applications.

We propose an estimator that is based on linear interpolation and maximum entropy (LIME) and is a generalized weighted nearest neighbor method as described in Stone [6]. Stone's arguments provide a means of demonstrating the estimator's basic universal consistency properties for regression. Recent extensions [7] of Kullback's minimum discrimination information approach [8] provide a way of showing

that the LIME weights can be expressed in terms of an adaptive kernel. The LIME technique can be used for density estimation and takes on the form of the Mack-Rosenblatt estimator, except that the kernel does not satisfy their zero first moment constraint. We present a heuristic argument to show that in fact this lack of symmetry may significantly lessen the serious drawback observed by Mack and Rosenblatt that kNN estimates are doomed to severe problems of bias when the observation is drawn from the tails of the distribution.

The paper provides several basic results regarding the behavior of the LIME weights under simplified assumptions, along with simulation results exemplifying the behavior, and applications to real data. The paper concludes with a discussion of the method and the technical issues arising in extensions of the theoretical properties.

II. LINEAR INTERPOLATION WITH MAXIMUM ENTROPY

Let $\mathcal{T} = \{X_i; i = 1, \dots, n\}$ denote training vectors $X_i \in \mathcal{R}^d$, and let $x \in \mathcal{R}^d$. The observation x can be thought of as a sample vector of a random vector X drawn independently from the same distribution as used to generate the training vectors X_i . We preserve for now the lowercase notation for x as a specific realized sample vector, but later some of the theoretical results will be concerned with averages over the random observation vector X . In this section we describe an algorithm for producing an n -dimensional weight vector $w = \{w_i(x); i = 1, \dots, n\}$ from \mathcal{T} and x that can be used for regression and classification, and we show that the weight vector can be expressed as an adaptive kernel so that it can also be used for density estimation. The algorithm is based on linear interpolation and maximizing entropy; the resulting weights will be referred to as the LIME weights. The remainder of the paper is devoted to developing several basic properties of the LIME weights and demonstrating the LIME algorithm's performance.

Three additional definitions are required. First, we assume a distortion measure $D(u, v)$, a nonnegative function of two d -dimensional real vectors for which $D(u, v) = 0$ if and only if $u = v$. We further assume that the distortion measure is a difference distortion measure and adopt the usual notational convention that $D(u - v) = D(u, v)$, and we assume that D is convex. This is sufficient for the algorithm to be well defined, but we focus on the common case of squared error, $D(u, v) = |u - v|^2 = \sum_{i=1}^d |u_i - v_i|^2$, in our examples. Second, the Shannon entropy of w is defined in the usual way by $H(w) = -\sum_{i=1}^n w_i \ln w_i$.

Third, define a neighborhood $J_n = J_n(x) \subset \{1, \dots, n\}$ as a subset of the indices of the training vectors. The most important example for the present work is the neighborhood of the k nearest neighbors to x in the training set $J_n = \{j : D(x, X_j) \leq D_k(x, \{X_1, \dots, X_n\})\}$ where $D_k(x, \{X_1, \dots, X_n\})$ denotes the distance from x to its k th-nearest neighbor in the training set with respect to distortion D . An alternative choice is to specify a radius R and include all training vectors within distance R of x : $J_n = \{j : D(x, X_j) \leq R\}$. In both cases the idea is to base the estimator on local training vectors, but the specifics of how “local” is defined are not important for defining the LIME estimator. Training vectors whose index is not in the neighborhood J_n are given zero weight. The training vectors can be reindexed in order of nearness to x for convenience.

A. Lime Weights

The LIME weight vector w is defined in terms of a parameter λ . Let \mathcal{W} be the collection of all probability mass functions w , that is, all n -tuples for which $w_i \geq 0$ if $i \in J_n$ and $w_i = 0$ otherwise, and $\sum_{i \in J_n} w_i = 1$. Then the LIME weights w^* solve

$$\operatorname{argmin}_w \left(D\left(\sum_{i \in J_n} w_i X_i - x\right) - \lambda H(w) \right). \quad (9)$$

A minimizer w^* for (9) exists and is unique if the function D is a continuous convex function of w . Henceforth, we make that assumption. For example, any l_p norm or monotonically increasing function thereof will work. For convex distortion functions D , the LIME objective function (9) is a sum of convex functions and thus convex, so that w^* can be found using standard convex optimization methods. In our numerical results, mean-squared error (squared l_2 distance) is used for D , and the optimization of (9) is done with a fast primal-dual log-barrier interior-point method from Saunders [9].

To illustrate LIME’s behavior, consider some extreme cases. First, if λ is nearly zero, then the estimator concentrates on minimizing the distortion. If the observation is contained in the convex hull of its neighbors, then the distortion can usually be forced to zero if λ is sufficiently small, and the distortion can be forced to zero for all cases if the distortion function D is an *exact penalty distortion* such as an l_p norm [7]. Given λ small enough to achieve zero distortion, the LIME objective will be minimized by the choice of weights that have maximum entropy of all weight vectors that yield zero distortion.

Under commonly adopted asymptotic assumptions on k and n , with probability one the observation x will lie in the convex hull of its k nearest neighbors (see Theorem 3). If the observation x does not lie within the convex hull of its k nearest neighbors, then small λ will yield w^* such that $\sum_{i \in J_n} w_i^* X_i$ approximates x .

For illustration, in Fig. 1, four different LIME weights are shown for a test point $x = 0$ and different sets of five training samples. For these examples, λ is set relatively small at $\lambda = .0001$.

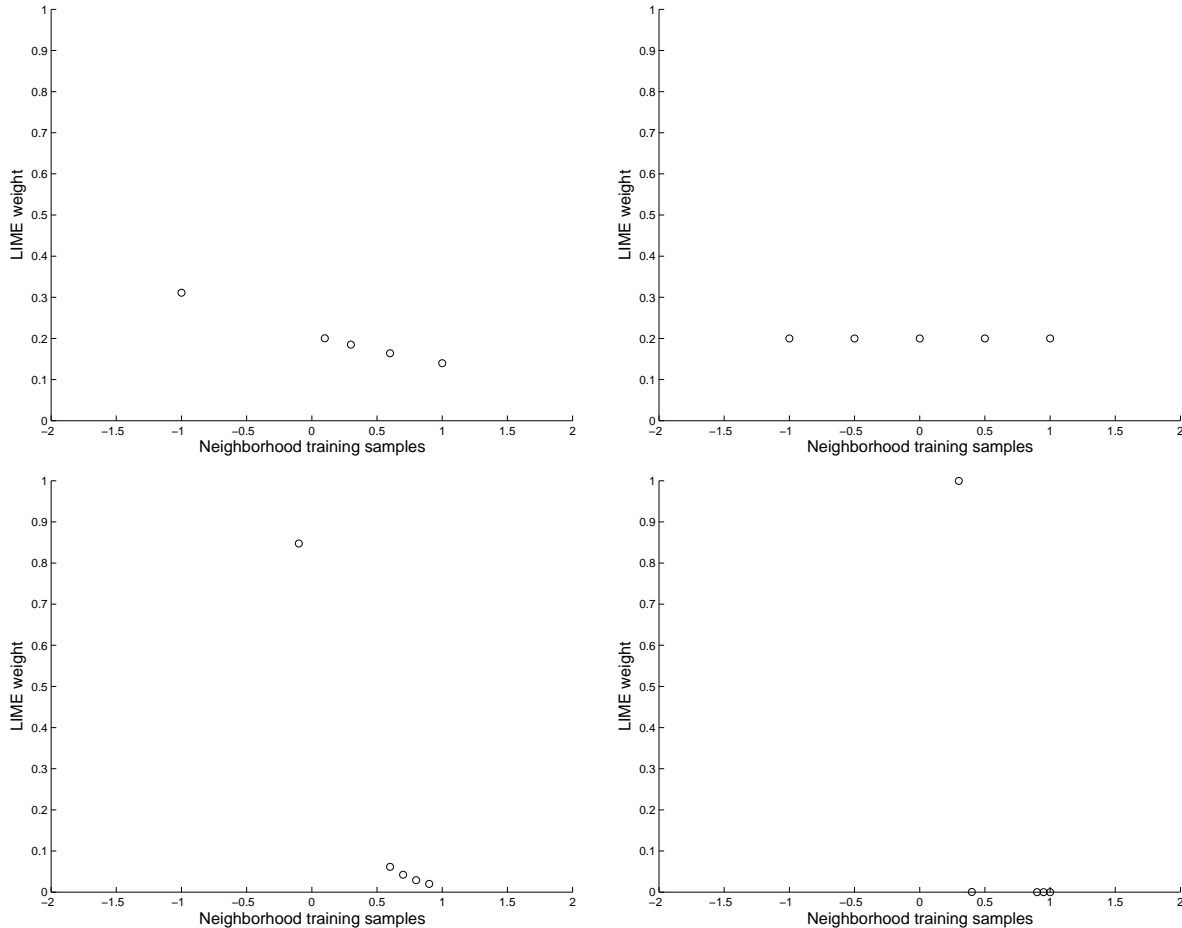


Fig. 1. LIME weights form an adaptive kernel. In these examples, the test point is at 0, and λ is small, emphasizing minimum distortion D . Top left: neighborhood training samples at -1, .1, .3, .7 and 1. Top right: neighborhood training samples at -1, -.5, 0, .5 and 1. Bottom left: neighborhood training samples at -1, .6, .7, .8 and .9. Bottom right: neighborhood training samples at .3, .4, .9, .95 and 1.

Consider the opposite extreme with λ very large. Then the emphasis is on making the weights as uniform as possible with less concern for approximating x . In fact, for a fixed set of neighbors J_n , the LIME weights will converge to the uniform weights:

Lemma 1 (LIME weights converge to uniform): Let $u_i = 1/k$ for all $i \in J_n$. Then

$$\lim_{\lambda \rightarrow \infty} \|w^* - u\|_1 \rightarrow 0 \quad (10)$$

The choice of λ may follow the Shannon source coding approach where a designer picks λ to reflect subjective judgement, or the choice of λ may be dependent on the scaling of the feature space compared to entropy, or may reflect the amount of noise in the training vectors or labels. A common approach in statistical applications of regression and classification is to train parameters such as λ using cross-validation, choosing the λ that is optimal in the sense of yielding the smallest empirical cross-validated distortion.

As another extreme case, suppose that J_n is chosen to contain only a single nearest neighbor, so that the weighting vector becomes a scalar constant. In this case the entropy will be zero and the weight on the one nearest-neighbor will be $w_1 = 1$ to satisfy the normalization constraint. Hence LIME includes the standard 1-nearest neighbor as a special case (and with it the Cover and Hart's guarantee of asymptotic classification performance within twice the Bayes optimal [10], and the Cover guarantee of asymptotic estimation performance within twice the Bayes risk [11]).

The LIME minimization can be restated in a form incorporating a well-known problem of Kullback [8], which in turn resembles (as Kullback notes on p. 37) Shannon's rate for a source relative to a fidelity criterion [12] and Jaynes's maximum entropy estimates [13]. Suppose that w^* are the LIME weights and define $\hat{x} = \sum_{i \in J_n} w_i^* X_i$. Let \mathcal{C} be the convex hull of the training set: the set of all vectors \tilde{x} such that $\tilde{x} = \sum_{i \in J_n} w_i X_i$ for some $w \in \mathcal{W}$, and let $\mathcal{W}(\hat{x})$ be the subset of \mathcal{W} for which $\sum_{i \in J_n} w_i X_i = \hat{x}$. Then, let

$$\begin{aligned} \rho(\lambda, x) &= D(\hat{x} - x) - \lambda H(w^*) \\ &\geq D(\hat{x} - x) - \lambda \max_{w \in \mathcal{W}(\hat{x})} H(w) \\ &\geq \inf_{\tilde{x} \in \mathcal{C}} \left(D(\tilde{x} - x) - \lambda \max_{w \in \mathcal{W}(\tilde{x})} H(w) \right) \end{aligned} \quad (11)$$

The supremum of $H(w)$ is a maximum because H is concave and continuous in w . Conversely, suppose

that \hat{x} and some \hat{w} yield a value of $D(\hat{x} - x) - \lambda H(\hat{w})$ within $\epsilon > 0$ of the minimum value, then

$$\begin{aligned} \inf_{\tilde{x} \in \mathcal{C}} \left(D(\tilde{x} - x) - \lambda \max_{w \in \mathcal{W}(\tilde{x})} H(w) \right) + \epsilon &\geq D(\hat{x} - x) - \lambda H(\hat{w}) \\ &\geq \rho(\lambda, x), \end{aligned} \quad (12)$$

which proves the following lemma:

Lemma 2: Given a continuous, convex distortion D , the LIME weights w are the solution to

$$\min_{\hat{x} \in \mathcal{C}} \left(D(\hat{x} - x) - \lambda \max_{w \in \mathcal{W}(\hat{x})} H(w) \right) \quad (13)$$

where \mathcal{C} is the convex hull of the training set. The minimizing reproduction \hat{x} and the LIME weights w^* are related by $\hat{x} = \sum_{i \in J_n} w_i^* X_i$.

The interpretation here is that for x , every possible approximation \tilde{x} in the convex hull of the training set yields both a reproduction distortion $D(\tilde{x} - x)$, which we want to be small, and a maximum entropy $\max_{w \in \mathcal{W}(\tilde{x})} H(w)$, which we want to be large. The parameter λ specifies the desired tradeoff between these two quantities.

B. LIME Weighting Is Exponential

The maximum entropy solution is a variation of a discrete case of Kullback's minimum discrimination information formulation with a uniform prior (see Kullback [8]). Below, we present a simple proof that the weights will have exponential form based on the divergence inequality of information theory (cf. [14]); see [7] for more results on this class of extensions of the Kullback approach. Since maximum entropy solutions and maximum likelihood solutions often coincide [15], [14], maximum likelihood solutions may also be exponential, such as testing hypotheses about means in the context of empirical likelihood [16].

From Lemma 2, the LIME weights maximize $H(w)$ subject to the constraints $\sum_{i \in J_n} w_i X_i = \hat{x}$ for some $\hat{x} \in \mathcal{C}$ and $w \in \mathcal{W}$. This is equivalent to minimizing

$$\sum_{i \in J_n} w_i (\ln w_i - a'(X_i - x) - c) = \sum_{i \in J_n} w_i \ln \left(w_i / e^{-a'(X_i - x) - c} \right), \quad (14)$$

where a is a d -dimensional Lagrange multiplier corresponding to the mean constraint and c is a Lagrange multiplier corresponding to the weight normalization constraint. The inclusion of x in (14) is for con-

venience, so that we can “centralize” the training vectors around the observation vector. Moreover, the inclusion of x reflects the fact that $\sum_{i \in J_n} w_i (X_i - x) = \hat{x} - x$.

The divergence inequality states that for two pmfs w and q , the relative entropy of Kullback-Leibler divergence satisfies the inequality $\sum_{i \in J_n} w_i \ln w_i / q_i \geq 0$ with equality if and only if $w = q$. Equation (14) has the form of a divergence if the denominator of c sums to 1; that is, if $q_i = e^{-a'(X_i - x) - c}$ which implies that $\sum_{i \in J_n} e^{-a'(X_i - x) - c} = 1$. Thus the quantity to be minimized given in (14) is bounded to be greater than or equal to 0. By expansion,

$$\sum_{i \in J_n} w_i \ln w_i - w_i \ln (e^{-a'(X_i - x) - c}) \geq 0. \quad (15)$$

Imitating Kullback, we define

$$M(a) = \sum_{i \in J_n} e^{-a'(X_i - x)} = e^c. \quad (16)$$

Recognizing $M(a)$ and the entropy term $H(w)$ in (15), and re-arranging its terms, one sees that

$$H(w) \leq \sum_{i \in J_n} w_i a'(X_i - x) + \ln M(a), \quad (17)$$

with equality if and only if $w_i = q_i$ such that $w_i = e^{-a'(X_i - x)} / M(a)$. This yields the LIME weights w_i^* if the vector a satisfies

$$\hat{x} = \sum_{i \in J_n} w_i^* X_i = \sum_{i \in J_n} \frac{e^{-a'(X_i - x)}}{M(a)} X_i, \quad (18)$$

which is accomplished by choosing each component a_l so that

$$\begin{aligned} \frac{\partial}{\partial a_l} \ln M(a) &= \frac{\frac{\partial}{\partial a_l} M(a)}{M(a)} \\ &= - \sum_{i \in J_n} (X_i - x)_l \frac{e^{-a'(X_i - x)}}{M(a)} \\ &= -(\hat{x} - x)_l. \end{aligned} \quad (19)$$

In summary, the LIME weights can be expressed as an exponential pmf proportional to $e^{-a'(X_i - x)}$, where the weighting vector a depends on the entire training set and the observed vector x . Thus far, the development is a minor variation on Kullback. Taking advantage of the exponential form of the weights yields the following characterization of the LIME weights.

Lemma 3: The LIME weights are given by

$$w_i^* = \frac{e^{-a'(X_i-x)}}{M(a)} \quad (20)$$

where $M(a) = \sum_{i \in J_n} e^{-a'(X_i-x)}$ and $\partial \ln M(a) / \partial a_l = (\hat{x} - x)_l$.

The weight solution of (20) resembles the form of an adaptive kernel and a product multivariate kernel (see e.g. Scott [5], sections 5.3 and 6.6). The LIME kernel adapts to the entire training set and the observation x jointly, not to the individual training vectors X_i as in the first case considered in section 6.6.1 of Scott. The form is not strictly a product kernel, however, because the normalization in the denominator can not be written as a product. The overall multivariate kernel does not have precisely the form of an adaptive kernel as in (6.78) of Scott, but the individual terms in the product do have the form of adaptive scalar kernels since the vector a depends on the observation x . In particular, the weight w_i^* does not assume the traditional form $K_{h_x}(u) = K(u/h_x)/nh_x^d$, where K has unit integral. Although neither category of product kernel nor adaptive multivariate kernel as usually defined is an exact fit, LIME can be thought of as a close relation to both kernels. Depending on the choice of the neighborhood J_n chosen for the nonzero weights, the algorithm can be considered as a member of the generalized kernel or a generalized nearest-neighbor method with an adaptive kernel. As noted in Scott, even the usual adaptive kernel K_{h_x} does not integrate to 1 even though K does. Thus the LIME implicit adaptive kernel of is no worse in this regard than the traditional case. Both can be forced to have unit integral by confining their support to a bounded region and normalizing them suitably. The primary difference between the LIME kernel and traditional kNN and adaptive kernels is that it is not symmetric and does not have a zero first moment as is usually assumed for asymptotic analysis [4], [5]. An asymmetric kernel has the potential to ease a famous problem of bias popularly considered to be a major drawback of kNN methods. This potential is supported by simulations presented here, and we are in the process of developing an asymptotic analysis of bias using LIME, but the intuition behind the claim is easily stated. In asymptotic bias analysis for kNN estimators as in [4], the large bias term occurs in the tails of the distribution due to a power of the density being estimated occurring in the denominator of a second moment term of the kernel. The zero first moment term implies there is no linear term in the expansion to compensate for the quadratic term. In the tail region, however, the density will be asymmetric about the observation. A symmetric kernel

will weight samples on all sides of a test point neighborhood equally, underweighting the probabilistically sparse training vectors in the distribution tail, causing bias. The asymmetric LIME kernel yields lower weights where there are more sample points, and higher weights where there are fewer sample points. This asymmetry results in a nonzero linear term in the asymptotic expansion which can ameliorate the effect of the quadratic term, and which is not possible with a symmetric kernel.

C. LIME Weights Vanish Asymptotically

We close this section with a useful asymptotic boundedness property of LIME weights. Suppose that an increasing set of training data results in a sequence of LIME weight vectors w_n^* for $n = 1, 2, \dots$. The following lemma states that the maximum value of the weight vector tends to 0 as $n \rightarrow \infty$. The result holds pointwise and does not require assumptions on the distribution of the training set.

Lemma 4: Given a training sequence $X_i = x_i$ for $i = 1, 2, \dots$, let w_n^* denote the LIME weights for X_1, \dots, X_n . Then $\lim_{n \rightarrow \infty} \|w_n^*\|_\infty = 0$.

III. SIMULATIONS

To explore the differences between the LIME algorithm and other neighborhood methods, a simulation example is used that combines ideas of Kohonen et al. [17], [18] and of Hastie et al. [1] (pp 384–385). Different runs show the effects of varying the neighborhood size k , the effect of increasing the number of iid training dimensions, the effect of varying λ , and a comparison of the empirical bias and variance of the classifiers. Comparisons are made with other neighborhood methods and the Bayes classifier (which uses knowledge of the class conditional densities to achieve the minimal Bayes risk). For the d -dimensional simulation, we draw d -dimensional random feature vectors $X \sim f(x) = .5f^0(x) + .5f^1(x)$ where $f^g(x)$ is the conditional pdf given that the class label $Y = g$, and Y is equally likely to be 0 or 1.

Class 1 is a mixture of two Gaussians, $f^0 = .5\mathcal{N}(0, \Sigma) + .5\mathcal{N}(0, 9\Sigma)$, where the covariance matrix Σ is the $d \times d$ identity matrix. Class 2 is distributed as $f^1 = \mathcal{N}(0, 4\Sigma)$. Thus, the feature vectors X are drawn from a mixture of three Gaussians all centered at the origin forming a spherical cloud. The points from Class 2 are surrounded inside and outside by points from Class 1, and hence a variation on this simulation [1] (pp. 384–385) is called the “skin of the orange” test. The class conditional distributions are

shown for one feature dimension in Fig. 2 (left), and for the two feature dimension simulation, the Bayes decision boundaries are shown in Fig. 2 (right). There are many nonparametric neighborhood classifiers

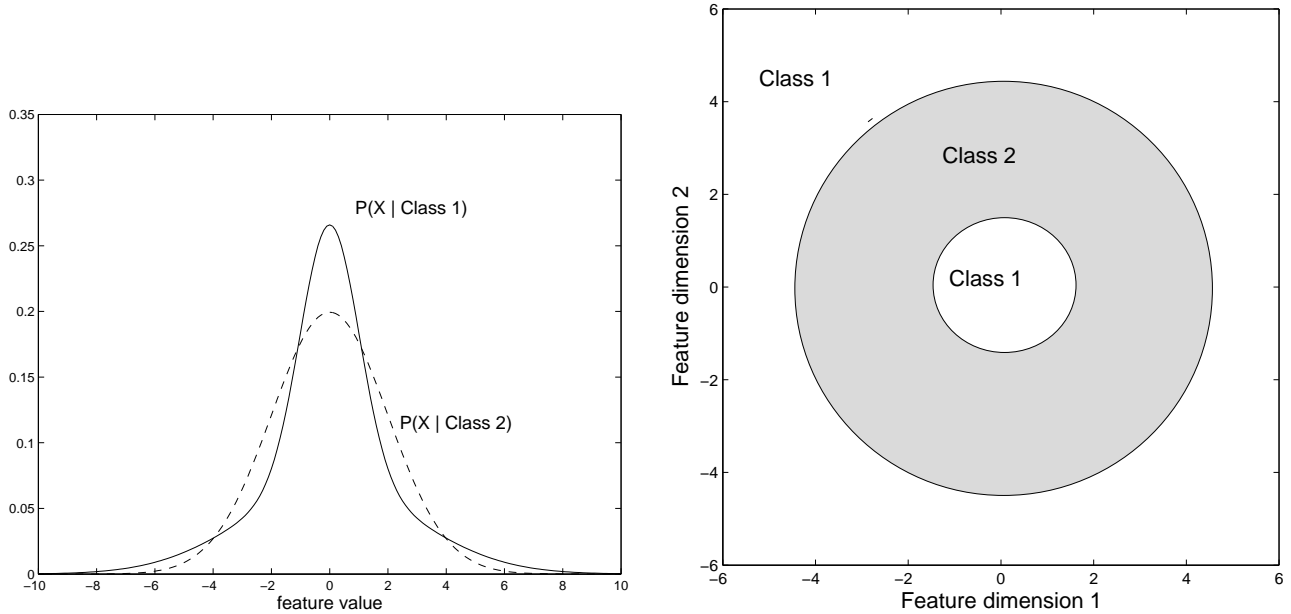


Fig. 2. Left: Class conditional pdf's for the simulation run with one feature dimension. Right: The Bayes decision regions for the simulation run with two feature dimensions.

(see [19], [1], [20], [21] for reviews and discussion). LIME is compared to the basic kNN classifier, the popular tricube kernel [1] (p. 168), and local linear regression [1] (pp. 168–172). The simple kNN technique achieves competitive performance over a wide range of classification problems, even when compared to state-of-the-art classifiers [1], [22]. The tricube kernel is representative of the general class of positive, symmetric, smoothing kernels.

As with LIME, the local linear regression weights are adaptive and asymmetric, but local linear regression differs from LIME in that it uses the neighborhood training labels Y_i to define the kernel. As with Stone [6], the local linear regression estimate for a test point X fits a hyperplane with the least-squared error to a neighborhood of observations J_n , so that the hyperplane parameters a and b solve

$$\operatorname{argmin}_{a,b} \sum_{i \in J_n} (Y_i - (a'X_i + b))^2. \quad (21)$$

Then an estimate of the label corresponding to X is $\hat{Y} = a'X + b$. Similarly, local linear regression can be used for two-class classification in which for 0-1 loss classification, $\hat{Y} = \operatorname{argmin}_g \min(g - a'X + b)^2$, where g is a class label. For these simulations we consider a two-class classification where $Y_i, Y \in \{0, 1\}$.

Local linear regression can be expressed as an adaptive nonparametric estimation method with weights on the neighborhood training samples that satisfy $\mathbf{1}'w = 1$, but there is no constraint that $w \in [0, 1]^k$.

Local linear regression and LIME take different approaches to reducing the estimation bias. As analyzed in [21], local linear regression performs “automatic kernel carpentry” by fitting a hyperplane to the training samples, and this eliminates the first-order bias (assuming the regression target can be expanded in a Taylor series). Similarly, *local polynomial regression* fits higher-order polynomials and thus enables elimination of higher-order bias terms. LIME does not depend on the training observations Y_i , instead reducing the bias by driving down $D(\sum_{i \in J_n} w_i X_i - x)$.

A. Simulation Varying the Neighborhood Size k

As per the simulation architecture previously described, we randomly generated 2000 training points and 10,000 test points in a twenty-dimensional feature space. The algorithms are compared for increasing size of neighborhood k . The LIME algorithm has two parameters, λ and the neighborhood size k . For this simulation, the LIME estimates are calculated with λ set to a default value of 10^{-9} : an extreme value so that the LIME weights are (to numerical accuracy used) the weights that maximize the entropy given that they solve $\operatorname{argmin}_w \sum_{i \in J_n} w_i X_i - X$. This is not the optimal value of λ for this dataset, but by using a default λ value none of the algorithms needs to be trained for this comparison, shown in Fig. 3.

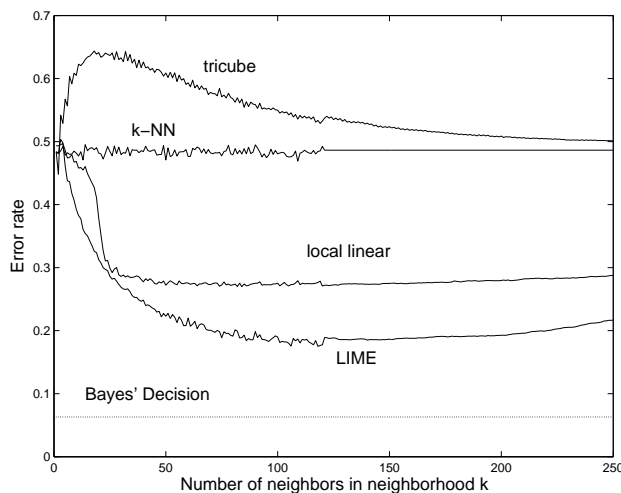


Fig. 3. Simulation varying the neighborhood size k for twenty feature dimensions and 2000 training points.

For a neighborhood size of $k = 1$, all the algorithms put all the weight on the nearest-neighbor and

perform equally well. As neighbors are added to kNN and tricube, their performance quickly deteriorates. Adding neighbors generally brings down the variance of estimates, but the bias of these estimates increases for larger neighborhood size in this simulation. The effect can be seen even in the one-dimensional feature space shown in Fig. 2. Consider the test point $x = 2$, which is outside the Bayes decision boundary; and thus the Bayes estimate for this point is class 2. Given only a few training samples, the probability distribution of training samples around x suggests that x 's near-neighbors are more likely to be on x 's left rather than its right, because the density of training samples is greater to the left of x . However, points to the left of x are more likely to be from class 1, and thus x 's nearest neighbors are more likely to be from class 1, causing symmetric neighborhood classifiers to mislabel x as class 1. LIME mitigates this bias problem by weighting the near neighbors based on their joint spatial relationship. For equidistant training samples, LIME assigns more weight where the samples are sparse, and less weight where the samples are dense, in order to minimize $D(\sum_{i \in J_n} w_i X_i - x)$. In higher-dimensional feature spaces, the training feature vectors are sparse, and bias problems near decisions boundaries are increasingly of concern.

B. Simulation with Varying Feature Space Dimension

Next we compare the performance of the algorithms with trained parameters. We randomly generated 2000 training points for 1, 2, 5, 8, 10, 15, and 20 dimensional feature spaces. Each algorithm's parameters were trained by leave-one-out cross-validation on the training set; the number of neighbors k was varied by steps of one, while the LIME parameter λ was trained in multiplicative steps of 2. The trained algorithms were compared on 10,000 test points. The error rates are recorded in Fig. 4. The results show that the difference in performance between local linear regression weights and LIME weights is small for low-dimensional feature spaces, but that LIME outperforms the other algorithms as the number of feature dimensions grows. Since each feature dimension provides additional information about Y , the Bayes error decreases with increasing feature dimensions. LIME's error also decreases with increasing feature dimensions. For ten feature dimensions and up, the other algorithms' error rates increase, showing that the extra information from the additional feature dimensions is confuses these algorithms, rather than helping them.

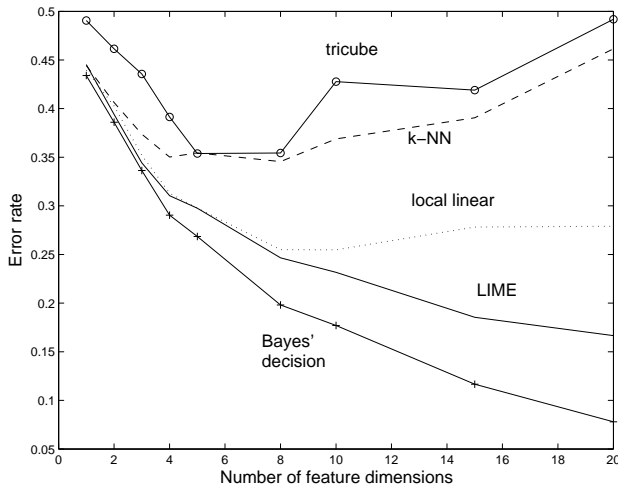


Fig. 4. Simulation for increasing number of feature dimensions, with $n = 2000$ training samples.

C. Simulation with Varying λ

Fig. 5 shows simulation results for LIME with λ varying from 2^{-15} to 2^{12} for 2000 test points estimated using 2000 training points with a neighborhood size of $k = 127$, as cross-validated in the preceding subsection.

As expected, the graphs show that the distortion D grows as λ grows, and that the entropy of the LIME weights also grows as λ grows. The error rate, as shown in the upper-left graph, is lowest for some tradeoff between the two objectives; here for $\lambda = 1$. For very large λ , the weights approach uniformity, thus for very large λ LIME's behavior is similar to that of kNN.

D. Simulation to Compare Classification Bias and Variance

We start with 1000 test points in twenty feature dimensions. Then, we draw 100 independent sets of 2000 training points, and form 100 estimates for each of the 1000 test points. The parameters $k = 127$ and $\lambda = 1$ are used for all 100 training sets and for all algorithms compared here. Shown in Fig. 6 are the resulting histograms of classification bias (left figures) and classification variance (right figures) of the one hundred estimates for each of the 1000 test points.

For the two-class classification problem, with $Y \in \{0, 1\}$, we define $\text{bias}(x)$ to be the bias of the classification corresponding to test feature vector x , defined as the expectation over the joint probability distribution of the training data,

$$\text{bias}(x) = E_{\{X_i, Y_i\}}[\hat{Y}(x) - Y]. \quad (22)$$

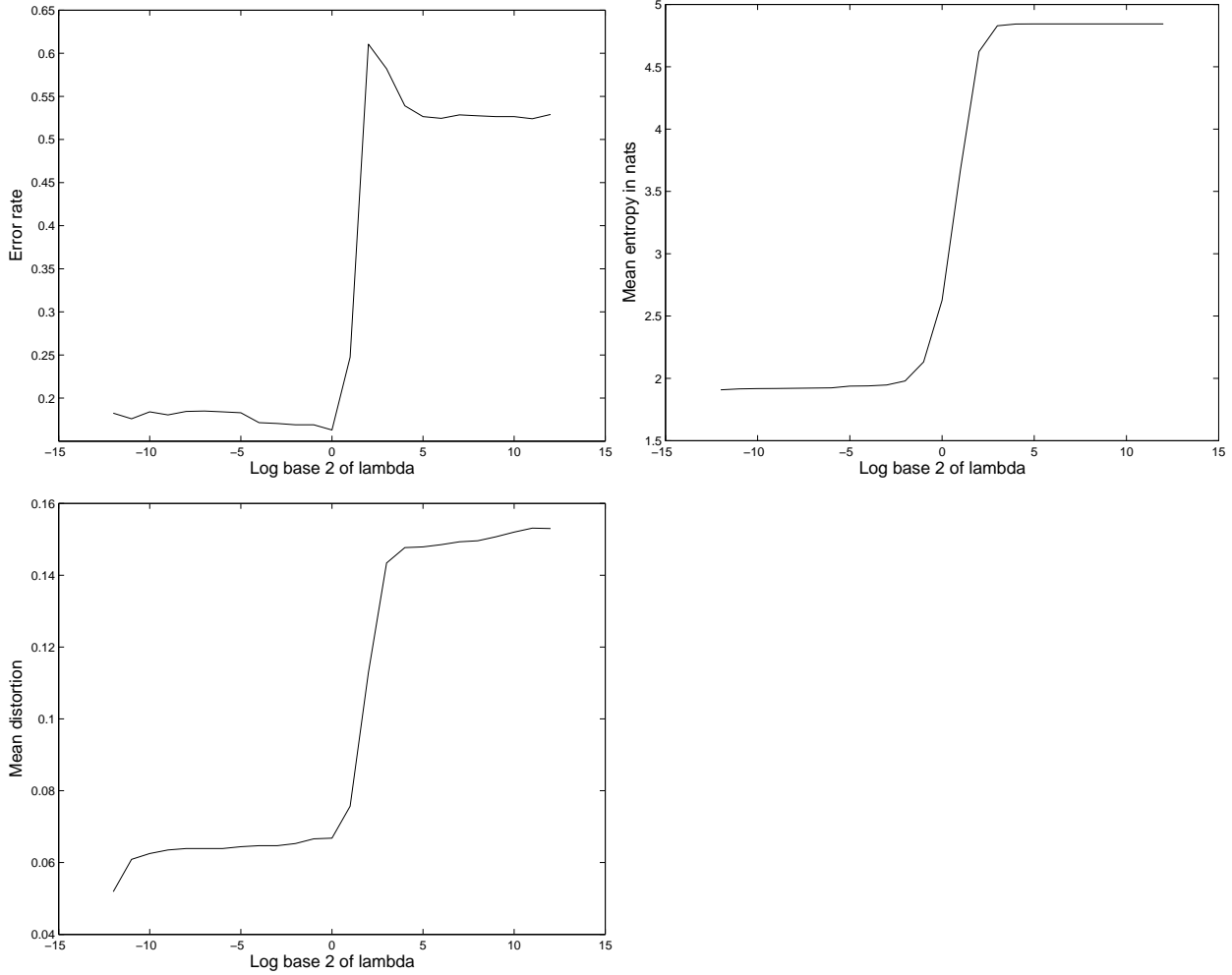


Fig. 5. LIME simulation results for varying λ on 2000 test points in twenty dimensions with 127 neighbors. Top left: Classification error rate. Top right: Entropy of the LIME weights averaged over the test set. Bottom: Mean squared error between the test point X and its LIME weighted neighbors averaged over the test set.

The classification variance $\text{var}(x)$ corresponding to test feature vector x is defined as an expectation over the joint probability distribution of the training data,

$$\text{var}(x) = E_{\{X_i, Y_i\}}[(\hat{Y}(x) - E[\hat{Y}])]^2. \quad (23)$$

An alternative bias definition for two-class classification problems is given by Friedman [23], who defines bias and variance for two-class problems in terms of the estimated conditional probability: $P(\hat{Y}(x) = 1|x)$. However, as Friedman analyzes [23], the link between the conditional probability bias and the classification error is nonlinear. Thus we have chosen to demonstrate classification bias and classification variance directly instead.

Error in classification is either due to bias of the classification (22), to the variance of the classification

(23), or to the irreducible variability of the test label Y itself. For example, the optimal Bayes classifier has zero variance (the Bayes classifier does not change with different training sets); the Bayes error rate is due entirely to the randomness of the test observation Y .

The total error over all 100,000 test points in this simulation was 53,799 errors for kNN, 25,296 errors for local linear, and 16,791 errors for LIME. Did the LIME reduction in error come generally from a reduction in bias or from a reduction in variance? As is seen in the histograms displayed in the right column of Fig. 6, the variance of the LIME estimates was in fact generally larger than the kNN or local linear variance. Thus, the error reduction for LIME must have come from reduced bias. The left column of Fig. 6 confirms this: the LIME estimation bias has predominantly small magnitude, whereas the kNN and local linear biases have peaks at -1 and 1 , respectively.

IV. ANALYSIS OF THE INTEGRITY OF PIPELINES

Recently developed optical inspection tools provide images from the inside of natural gas pipelines to monitor pipeline integrity. Experts can classify the images with labels such as ‘normal,’ ‘weld,’ or ‘corrosion blisters.’ Two example images are shown in Fig. 7. In [24] the feasibility of an automatic classification system was studied using pipeline images from Norsk Electro Optikk (NEO). Twenty-two features were developed to differentiate twelve classes of events. Misclassification costs were estimated by engineers at NEO and varied greatly between pairs of classes.

Classification results on a expert-labeled set of 228 images are shown in Table I for linear discriminant analysis (LDA) [1], regularized quadratic discriminant analysis (QDA) [1], LIME, and Friedman’s boosted decision tree algorithm MART¹ [1]. LDA and QDA were chosen for their robustness to the application’s small sample size given the twenty-two dimensional feature size. Each of the twenty-two features was developed to help differentiate between two confusable classes, and thus it was expected that a decision tree method would work well.

For each sample x , any classifier parameters were estimated based on the other 227 sample points and the estimated class of x was determined using these parameters. Notably, the expected LIME cost was more than 20% lower than the expected cost with the MART decision tree. These results show LIME to

¹MART was implemented using code available at <http://www-stat.stanford.edu/~jhf/>

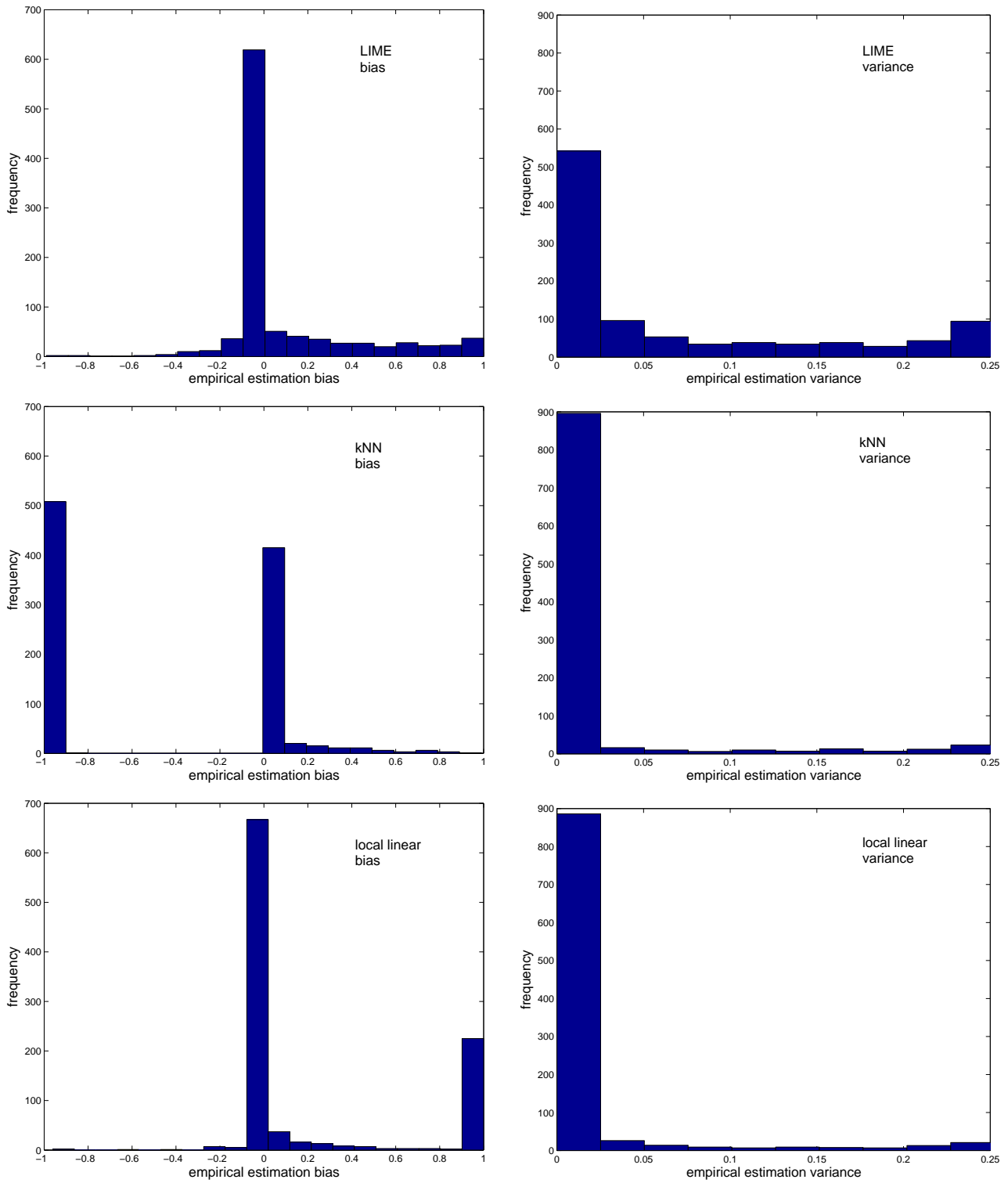


Fig. 6. Left column: Histograms of the empirical biases of the classification of 1000 test points, where the empirical bias is an average over 100 different training sets. Right column: Histogram of the empirical variance of the classification of 1000 test points, where the empirical variance is an average over 100 different training sets. Right top: LIME variance. Right middle: kNN variance. Right bottom: Local linear variance.

be a competitive classifier in a practical situation.

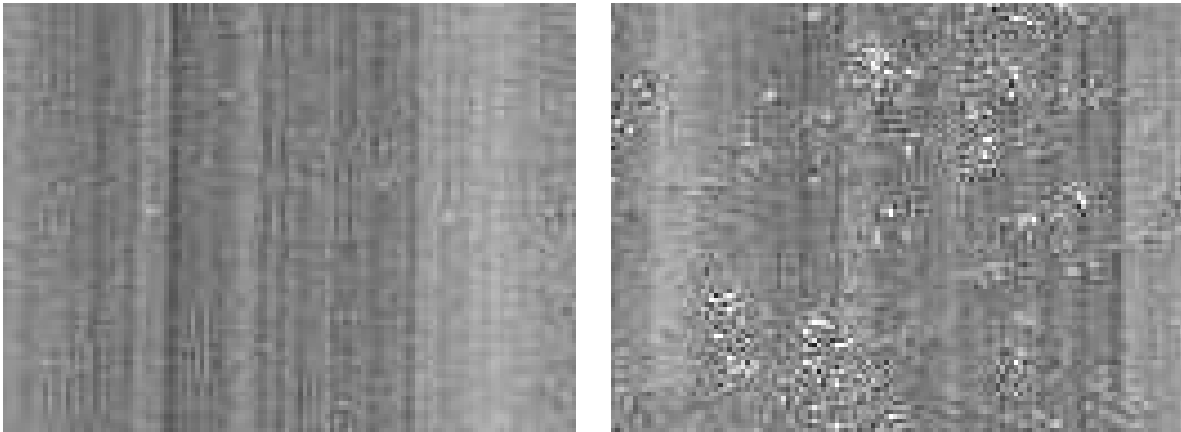


Fig. 7. A normal image (left) and osmosis blister image (right) from the inside of a natural gas pipeline. Photos courtesy of NEO.

V. CONSISTENCY

A guarantee that an algorithm will achieve optimal risk asymptotically is useful in practice, and validates the inductive method philosophically [25], [26]. Learning algorithms that fit a model of the class densities, or that model the decision boundaries, are often not flexible enough to approach optimal rates with an increasing number of training samples. A method for supervised learning is L^r consistent if when (X, Y) , $(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)$ are iid, Y is real-valued, $r > 1$, and $E[|Y|^r] < \infty$, then $\hat{Y}(X) \rightarrow E[Y|X]$ in L^r . A method is said to be *universally* consistent if it is consistent regardless of the distribution of X . Many nonparametric supervised learning algorithms are consistent; see [19] for a review of consistency properties of various algorithms. Algorithms of interest in this paper are non-adaptive in that they are of the form

$$\hat{Y}(X) = \sum_{i=1}^n w_{ni}(X, X_1, \dots, X_n) Y_i ; ,$$

that is, the weights $w_{ni}(X) = w_{ni}(X, X_1, \dots, X_n)$ do not depend on Y_1, \dots, Y_n .

In this section, the result is formally stated that the LIME algorithm with any l_p distance function D is Bayes risk consistent for bounded feature spaces under standard asymptotic assumptions, $k \rightarrow \infty$, $n \rightarrow \infty$, and $k/n \rightarrow 0$. The proof of this theorem is given in [7]. This special case of bounded features corresponds to practical situations, and highlights the key ideas of the algorithm without the clutter of detailed technical arguments. Extensions of the consistency result to unbounded features is difficult, and

| | MART | LDA | Reg. QDA | LIME |
|---------------------|--------------|--------------|--------------|--------------|
| Normal | 20.93 | 0.23 | 0.70 | 9.65 |
| Osmosis Blisters | 11.50 | 14.75 | 5.00 | 5.75 |
| Black Lines | 16.79 | 28.57 | 37.50 | 11.79 |
| Corrosion Dots | 6.18 | 117.65 | 52.94 | 29.41 |
| Welds | 12.00 | 2.50 | 0.00 | 1.50 |
| Weld Cavity | 7.89 | 99.47 | 17.37 | 9.47 |
| Welds Too Close | 37.50 | 161.88 | 50.00 | 25.00 |
| Field Joint | 27.50 | 0.00 | 2.50 | 25.00 |
| Grinder Marks | 32.00 | 0.75 | 0.25 | 20.75 |
| MFL Marks | 23.85 | 3.08 | 1.54 | 20.77 |
| Corrosion Blisters | 20.45 | 27.27 | 72.73 | 20.91 |
| Single Dots | 18.00 | 78.00 | 41.33 | 6.00 |
| Average cost | 19.55 | 44.51 | 23.49 | 15.50 |

TABLE I
MEAN EXPECTED COST FOR EACH GIVEN CLASS

some of the technical details are not yet resolved. The extension is further discussed in the final section of the appendix.

Theorem 1 (LIME consistency): Suppose w_n^* is the pmf that solves an l_p LIME minimization problem for X and its $k(n)$ nearest-neighbors, where the nearest-neighbors are ordered in terms of l_p distance. Suppose all training and test feature vectors are random variables drawn iid according to a distribution with bounded support. Then the sequence of weights w_n^* is universally consistent as $k(n) \rightarrow \infty$, $n \rightarrow \infty$, and $k(n)/n \rightarrow 0$.

Stone's consistency is a statement about convergence of the estimated mean over the space of test vectors X . However, one can apply dominated convergence and Fubini's theorem to see that Stone's consistency also implies convergence of the estimated mean when conditioned on $X = x$ for a set of x

with probability one.

A. Robustness to Noise

Real measurements are rarely noise-free. The following lemma states that if the training observations are corrupted by iid zero-mean additive noise, the expected LIME estimate will be unaffected, and further, that the noisy estimate will converge without bias to the clean LIME estimate as the number of neighborhood training samples $k \rightarrow \infty$.

Lemma 5: Suppose that each Y_1, Y_2, \dots is observed with additive noise. Thus, $\tilde{Y}_i = Y_i + \varepsilon_i$, where $\varepsilon, \varepsilon_1, \varepsilon_2, \dots$ are iid. Assume that $\{\varepsilon, \varepsilon_i\}$ is independent of $\{(X, Y), (X_i, Y_i)\}$ and that for some $p > 1$, $E[|\varepsilon|^p] < \infty$. Then LIME regression estimates computed from $\{(X_i, \tilde{Y}_i), i = 1, 2, \dots\}$ are consistent provided $\|X\|$ is finite.

VI. LINEAR INTERPOLATION ON GRIDS

Consider the case in which the known sample points lie on a regular d -dimensional grid, and the test point x is interior to the grid. It is common to interpolate such test points by obtaining weights by performing linear interpolation dimension-by-dimension, and then applying the obtained weights linearly to the associated output variable to form an estimate. This technique is used in color management to interpolate three-dimensional look-up-tables [27]. More generally, the Matlab function *interp*n [28] performs this successive linear interpolation in each dimension, and code is available in Numerical Recipes in C [29]. Let the training vectors $\{x_1, x_2, \dots, x_{2^d}\}$ be the vertices of the d -dimensional unit hypercube. Consider a test point $x \in [0, 1]^d$; the i th weight of the successive linear interpolation weights for x can be written:

$$w_i^* = \prod_{m=1}^d (|1 - x_i - x|)_m, \quad (24)$$

for $i = 1, \dots, 2^d$, where $(x)_m$ is the m th component of the vector x .

Notably, this common form of linear interpolation for grids is the weighting of the grid points that has the maximum entropy out of all solutions that satisfy the linear interpolation equations.

Theorem 2: The successive linear interpolation weights (24) solve

$$\operatorname{argmax}_v H(v) \quad (25)$$

subject to

$$\sum_{i=1}^{2^d} v_i = 1, \quad \sum_{i=1}^{2^d} v_i x_i = x, \quad v \geq 0. \quad (26)$$

VII. CONVEX HULLS OF k -NEAREST NEIGHBORS

As shown in the simulations, the LIME estimates can achieve lower bias than other neighborhood methods. A heuristic argument for the bias reduction is that the linear interpolation equations are approximately satisfied and thus the bias is greatly reduced. This argument is strengthened by the following result that X is in the convex hull of its k -nearest neighbors out of a total of n training sample vectors. Theorem 3 shows that in large enough samples, this condition not only is satisfied, but remains satisfied over the indefinite horizon as $n \rightarrow \infty$.

Theorem 3: Let $\{X, X_i\}$ be iid and take values in \mathcal{R}^d , for some $d < \infty$. Suppose their common distribution is absolutely continuous with Lebesgue density f . Let $\operatorname{supp}(f)$ denote the support of f , and $\operatorname{supp}(f)^\circ$ its interior. Assume that $\mu(\operatorname{supp}(f) \setminus \operatorname{supp}(f)^\circ) = 0$, where μ denotes Lebesgue measure. Assume further that there is a numerical sequence $a_n \rightarrow \infty$ for which $a_n(\log n)/n \rightarrow 0$, and set $k = k(n) = a_n \log n$. Let $X_{1,n}, X_{2,n}, \dots, X_{k,n}$ be the $k(n)$ nearest neighbors of X among X_1, X_2, \dots, X_n . (Note that the existence of f means that almost surely each X_i is uniquely defined.) Let $\mathcal{C}(k, n)$ be the convex hull of $X_{1,n}, X_{2,n}, \dots, X_{k,n}$. Then

$$P \left(\bigcup_{N=1}^{\infty} \bigcap_{n=N}^{\infty} [X \in \mathcal{C}(k, n)] \right) = 1.$$

VIII. DISCUSSION

LIME uses the linear interpolation equations to avoid bias, and the maximum entropy principle to weight all near neighbors as uniformly as possible to keep estimation variance low. LIME regularizes the weights, but differs from methods that trade off between empirical accuracy and complexity [30], [31], [32]. Instead, the LIME weights are designed to trade off between distortion in the feature space (not in the observation space) and diversity in the use of the training data.

The algorithm presented here is a “vanilla” version. Refinements in terms of neighborhood selection, fast neighbor search, feature scaling, distortion functions, reduced weighting over distance, and hybrid estimation techniques might all lead to better performance.

In this paper we have presented several theoretical properties for LIME under simple assumptions. Open questions include the analysis of rates of convergence for the LIME weight, theoretical determination of the parameter λ , and theoretical determination of $k(n)$ when using the k nearest neighbors as a neighborhood. Optimal values of $k(n)$ would depend upon the smoothness of the underlying probability density and on the smoothness of the function being estimated. We believe that approaches of Stone [33] adapted to the nearest neighbor probability structure [34] would be a foundation for such results.

APPENDIX

Proof: [Lemma 1 (LIME Weights Converge to Uniform)]

The proof of Lemma 1 follows easily from an auxillary result characterizing the objective in the limit of large λ :

Lemma 6 (Lemma): Let $u_i = 1/k(n)$ for all n and for all $i = 1, \dots, k(n)$ and $u_i = 0$ otherwise. Let $F(w, \lambda) = D(w) - \lambda H(w)$, where $D(w)$ denotes $D(\sum_{i=1}^k w_i X_i - X)$.

$$\lim_{\lambda \rightarrow \infty} \frac{\inf_w F(w, \lambda) - F(u, \lambda)}{\lambda} = 0 \quad (27)$$

Proof: [Lemma 6] By definition, $\inf_w F(w, \lambda) \leq F(u, \lambda)$. Then, since $\lambda \geq 0$, $F(u, \lambda)/\lambda - \inf_w F(w, \lambda)/\lambda \geq 0$, and thus,

$$\liminf_{\lambda \rightarrow \infty} \left(\frac{F(u, \lambda)}{\lambda} - \inf_w \frac{F(w, \lambda)}{\lambda} \right) \geq 0. \quad (28)$$

Coupling

$$\frac{D(u)}{\lambda} - H(u) - \inf_w \left(\frac{D(w)}{\lambda} - H(w) \right) \leq \frac{D(u)}{\lambda} - H(u) + \sup_w H(w),$$

with

$$\begin{aligned} \lim_{\lambda \rightarrow \infty} \left(\frac{D(u)}{\lambda} + \sup_w H(w) - H(u) \right) &= \left(\sup_w H(w) - H(u) \right) \\ &= 0, \end{aligned}$$

establishes that

$$\limsup_{\lambda \rightarrow \infty} \left(\frac{F(u, \lambda)}{\lambda} - \inf_w \frac{F(w, \lambda)}{\lambda} \right) \leq 0 \quad (29)$$

Combining (28) and (29) yields the lemma. ■

From Lemma 6, it can be concluded that for the LIME weights $w^* = \operatorname{argmin}_w F(w, \lambda)$,

$$\lim_{\lambda \rightarrow \infty} (H(w^*) - H(u)) = 0$$

, or equivalently,

$$\lim_{\lambda \rightarrow \infty} H(w^*) = H(u) = \log k. \quad (30)$$

A result from information theory [12, pp. 102–103] relates the l_1 distance and the relative entropy \mathcal{I} of two pmfs p and q , $\|p - q\|_1 \leq \sqrt{2\mathcal{I}(p\|q)}$. Then

$$\begin{aligned} \lim_{\lambda \rightarrow \infty} \|w^* - u\|_1 &\leq \lim_{\lambda \rightarrow \infty} \sqrt{2\mathcal{D}(w^*\|u)} \\ &= \lim_{\lambda \rightarrow \infty} \sqrt{2(\log k - H(w^*))} = 0 \end{aligned}$$

where the last line follows from (30). ■

Proof: [Lemma 4 (Asymptotic Vanishing Weights)]

We show that for any $\delta > 0$, there is some n_0 such that for every $n > n_0$, $\|w_n^*\|_\infty < \delta$. The proof is by contradiction. Consider any weight sequence a_n such that $\|a_n\|_\infty \geq \delta > 0$ infinitely often, and define the set \mathcal{A} such that $n \in \mathcal{A}$ if and only if $\|a_n\|_\infty \geq \delta$. Let the uniform weight sequence over the near neighbors be $u_{i,n} = 1/k(n)$. In this section we use the notation $D(w_n)$ to mean $D(\sum_{i=1}^n w_{i,n} X_i - X)$. The LIME weight sequence minimizes $D(w_n) - \lambda H(w_n)$ for all n . We show that there exists some n_0 such that

$$D(u_n) - \lambda H(u_n) < D(a_n) - \lambda H(a_n) \quad (31)$$

for all n such that $n > n_0$ and $n \in \mathcal{A}$. Thus, no sequence a_n with maximum component $\|a_n\|_\infty \geq \delta$ infinitely often can be the LIME weight sequence.

To show (31), we suppose pessimistically that $D(a_n) = 0$, then show that $H(u_n) - H(a_n)$ grows without bound for $n \in \mathcal{A}$, and that $D(u_n) \rightarrow 0$. Then for fixed $\lambda > 0$ and $\delta > 0$, there exists some n_0 such that

$D(u_n) < \lambda(H(u_n) - H(a_n))$ for all $n > n_0$ and $n \in \mathcal{A}$. Thus (31) holds and no sequence a_n can be the LIME weight sequence.

First we show that $H(u_n) - H(a_n)$ grows without bound for $n \rightarrow \infty$, $n \in \mathcal{A}$. By Fano's Inequality [14, pp.38–40],

$$H(a_n) \leq \delta \ln \left(\frac{1}{\delta} \right) - (1 - \delta) \ln \left(\frac{1 - \delta}{k(n) - 1} \right), \quad (32)$$

for $n \in \mathcal{A}$. Then

$$H(u_n) - H(a_n) \geq \ln(k(n)) + \delta \ln(\delta) + (1 - \delta) \ln \left(\frac{1 - \delta}{k(n) - 1} \right),$$

for $n \in \mathcal{A}$, which (after expanding) is dominated by the $\delta \ln(k(n) - 1)$ term for a given δ as $k(n), n \rightarrow \infty$.

Thus $H(u_n) - H(a_n)$ grows without bound for $n \in \mathcal{A}$ as $k(n) \rightarrow \infty$ and, $n \rightarrow \infty$.

Also,

$$\begin{aligned} D(u_n) &= \frac{1}{k(n)} \left\| \sum_{i \in J_n} X_i - X \right\| \\ &\leq \frac{1}{k(n)} \sum_{i \in J_n} \|X_i - X\| \\ &\leq \|X_{k(n)} - X\|, \end{aligned}$$

where the last line holds by the ordering of the near neighbors and because there are $k(n)$ near neighbors (that is, the cardinality of the set $i \in J_n$ is $k(n)$). The conditions for Lemma 5.1 from [19] (p 63) hold, so that $\|X_{k(n)} - X\| \rightarrow 0$, and thus $D(u_n) \rightarrow 0$. ■

Proof: [Lemma 5 (Robustness to Noise)] Write w_n^* for the LIME weights for a random test point X .

We have shown that w_n^* are consistent under conditions that apply to this lemma. Write

$$\begin{aligned} \hat{Y} &= \sum_{i=1}^n w_{i,n}^* \tilde{Y}_i \\ &= \sum_{i=1}^n w_{i,n}^* Y_i + \sum_{i=1}^n w_{i,n}^* \varepsilon_i. \end{aligned}$$

The first term tends to $E[Y|X]$ because $\{w_n^*\}$ is a consistent sequence of weights. If we replace (X_i, Y_i) pairs by (X_i, ε_i) pairs, then because w_n^* are consistent weights, $\sum w_{i,n}^* \varepsilon_i \rightarrow E[\varepsilon|X]$ as n increases without bound. Because ε and X are independent, $E[\varepsilon|X]$ is almost surely $E[\varepsilon] = 0$. So, $\sum_{i=1}^n w_{i,n}^*(X) \tilde{Y}_i$ tends

in L^p to $E[Y|X]$. ■

Proof: [Theorem 2 (Successive Linear Interpolation Maximizes Entropy)] To solve (25), there are 2^d weights that must be determined, but only $d + 1$ linear interpolation equations given as constraints. Since x is by construction contained in the convex hull of the vertices of the unit hypercube, there exists a continuous and convex set of solutions to the linear interpolation equations given as constraints:

$$\{w : w \in \mathcal{W}, \sum_{i=1}^{2^d} w_i x_i = x\} \quad (33)$$

where \mathcal{W} is the set of all pmfs with 2^d components.

It is easy to show by induction over dimensionality that the successive linear interpolation weights w^* given in (24) are in fact one solution to the linear interpolation equations. It remains to be shown that of the set of feasible pmfs, the successive linear interpolation weights w^* are the solution with maximum entropy.

According to the maximum entropy distribution theorem given in [14] (Theorem 11.1.1, p. 267) and given in [8] (Theorem 2.1, p. 38), the pmf that maximizes entropy given a moment constraint

$$\sum_{i=1}^{2^d} v_i x_i = x \quad (34)$$

is unique and has exponential form, $v_i^* = \gamma e^{-\alpha^T x_i}$, where $\alpha \in \mathcal{R}^d$ and $\gamma \in \mathcal{R}$ satisfy

$$\gamma = \frac{1}{\sum_{i=1}^{2^d} e^{-\alpha^T x_i}} \quad (35)$$

and

$$\gamma \sum_{i=1}^{2^d} x_i e^{-\alpha^T x_i} = x. \quad (36)$$

We present a γ and α that satisfy (35) and (36), and show that the successive linear interpolation weights w^* are equal to the maximum entropy weights v^* that solve (25) and (26).

Denote the m th component of vector x with $(x)_m$. Then let

$$\gamma = \prod_{m=1}^d 1 - (x)_m,$$

and let the m th component of α be,

$$(\alpha)_m = -\ln\left(\frac{(x)_m}{1-(x)_m}\right),$$

for $m = 1, \dots, d$. Substituting γ and α into the equation for the maximum entropy weight distribution $v_i^* = \gamma e^{-\alpha^T x_i}$, the maximum entropy weights are

$$v_i^* = \left[\prod_{m=1}^d (1 - (x)_m) \right] e^{\sum_{m=1}^d (x_i)_m \ln\left(\frac{(x)_m}{1-(x)_m}\right)}.$$

Simplifying,

$$\begin{aligned} v_i^* &= \prod_{m=1}^d (1 - (x)_m) \left(\frac{(x)_m}{1 - (x)_m} \right)^{(x_i)_m} \\ &= \prod_{m=1}^d (|1 - (x_i)_m - (x)_m|), \end{aligned}$$

where the last line follows because every component of x_i is either a one or a zero (since the training points lie on a regular grid). Thus, the maximum entropy weights v^* and the successive linear interpolation weights w^* are equivalent, and the successive linear interpolation weights must be the unique maximum entropy weights given the mean constraint. ■

Proof: [Theorem 3 (Convex Neighborhood)] Readers will note from this proof that if Theorem 3 holds for $d = 2$, then it holds for any d . Before turning to the main proof in two dimensions, some preliminaries are required.

The following theorem (Theorem 4) is needed for this proof. It is proved by a slight variation of the arguments for Theorem 12.2 of [35] and Theorem 3.15 of [36].

Theorem 4: Suppose that $\epsilon > 0$, a dimension $d < \infty$, $p > 0$, and a Vapnik-Chervonenkis class \mathcal{B} are given. Then there exists a constant $c = c(p, \epsilon, d, \mathcal{B})$ such that for all $n \geq N(p, \epsilon, d, \mathcal{B})$ there is a set A , with $P(A) \geq 1 - n^{-(2p+1)}$ on which simultaneously for all $B \in \mathcal{B}$,

$$|\hat{F}_n(B) - F(B)| \leq \epsilon \hat{F}_n(B) + c \frac{\log n}{n}$$

and

$$|\hat{F}_n(B) - F(B)| \leq \epsilon F(B) + c \frac{\log n}{n}.$$

Here $F(B) = P(X \in B)$; $\hat{F}_n(B) = n^{-1} \sum_{i=1}^n I_{x_i \in B}$, where I_E is the indicator function of the event E .

Some further notation is needed to prove Theorem 3. For $x \in \mathcal{R}^2$ and $r > 0$, let $S(x, r)$ denote the circle of radius r centered at x . Working in polar coordinates, let $W_i(x, r)$ be the ‘wedge’ defined as $S(x, r) \cap \{2\pi(i-1)/5 \leq \theta_i < 2\pi i/5\}$. Thus $S(x, r) = \cup_{i=1}^5 W_i(x, r)$, and $\mu(W_i(x, r)) \equiv 2\pi r^2/5 = \mu(S(x, r)/5)$. Clearly, if there exist points $z_i \in W_i(x, r)$, $i = 1, \dots, 5$, then $x \in \mathcal{C}(\{z_1, z_2, z_3, z_4, z_5\})$, that is, x is contained within the closure of the convex hull of $\{z_1, z_2, z_3, z_4, z_5\}$. By arguments using Fubini’s Theorem as in [37], it is enough to prove Theorem 3 for fixed $x \in \mathcal{R}^2$, where $x \in \text{supp}(f)$. Also, we will use the abbreviation “a.s. ultimately” to mean “almost surely ultimately.”

Now we begin the body of the proof. Note from [38, p. 39] that the circle $S(x, r)$ as x and r vary is a differentiation basis for $L^1(\mathcal{R}^2)$ (that is, for the Lebesgue-integrable real functions on \mathcal{R}^2). As a consequence, for Lebesgue almost all $x \in \text{supp}(f)^0$ and thus for almost all $x \in \text{supp}(f)$,

$$f(x) = \lim_{r \rightarrow \infty} \frac{\int_{S(x, r)} f(u) du}{\mu(S(x, r))}, \quad (37)$$

so for almost all $x \in \text{supp}(f)^0$,

$$\begin{aligned} \int_{S(x, r)} f(u) du &= P(x \in S(x, r)) \\ &= 2\pi r^2 f(x) + o(r^2) \\ &= 2\pi r^2 f(x) + o(\mu(S(x, r))). \end{aligned}$$

From the discussion prior to Theorem 1.1.1. on p. 3 of [39], it follows that (37) is equivalent to

$$\text{supp}_{\text{rational } r} \frac{\int_{S(x, r)} f(u) du}{\mu(S(x, r))} < \infty \quad (38)$$

for Lebesgue almost all x . From (38) and the discussion prior to Theorem 1.1.1 of [39] it follows that

$$\sup_{\text{rational } r} \frac{\int_{W_i(x,r)} f(u) du}{\mu(W_i(x,r))} < \infty \quad (39)$$

for $i = 1, \dots, 5$, and for almost all x . Therefore, for almost all $x \in \text{supp}(f)^0$, the analogue of (??) holds, that is, for all i ,

$$\begin{aligned} \int_{W_i(x,r)} f(u) du &= P(x \in W_i(x,r)) \\ &= \frac{2}{5} \pi r^2 f(x) + o(r^2) \\ &= \frac{2}{5} \pi r^2 f(x) + o(\mu(W_i(x,r))). \end{aligned} \quad (40)$$

Recall from pages 18 and 30 of [40] that $\{S(x,r)\} \cup \{W_i(x,r)\}$ as x , r , and i vary is a Vapnik-Chervonenkis class of sets. Let E_1^C denote the event E_1 not occurring, then for events E_1, E_2, \dots , write “ E_n wp1ofo” to mean “the event E_n with probability 1 only finitely often” if $P(\cup_{N=1}^{\infty} (\cap_{n=N}^{\infty} E_n^C)) = 1$. Now

$$|\hat{F}_n(W_i(x,r)) - \hat{F}_n(W_j(x,r))| \leq |\hat{F}_n(W_i(x,r)) - F(W_i(x,r))| + |\hat{F}_n(W_j(x,r)) - F(W_j(x,r))|.$$

Pick $1 > \epsilon > 0$. From Theorem 4 and the Borel-Cantelli lemma, it follows that there is a finite $c < \infty$ for which

$$|\hat{F}_n(W_i(x,r)) - F(W_i(x,r))| \leq \epsilon F(W_i(x,r)) + c \frac{\log n}{n} \text{ wp1ofo.} \quad (41)$$

Therefore,

$$\hat{F}_n(W_i(x,r)) < F(W_i(x,r)) - \epsilon F(W_i(x,r)) - c \frac{\log n}{n} \text{ wp1ofo,} \quad (42)$$

and

$$\hat{F}_n(W_i(x,r)) \geq (1 - \epsilon) F(W_i(x,r)) - c \frac{\log n}{n} \text{ a.s. ultimately.} \quad (43)$$

It follows that for Lebesgue almost all fixed $x_0 \in \text{supp}(f)^0$,

$$\hat{F}_n(W_i(x_0,r)) \geq (1 - \epsilon) \left(\frac{2\pi f(x_0)}{5} r^2 \right) + o(r^2) - c \frac{\log n}{n} \text{ a.s. ultimately.} \quad (44)$$

Let $r_n = \|X_{k,n}(x_0) - x_0\|$, where $X_{k,n}(x_0)$ is the k th nearest neighbor out of n neighbors to x_0 . If $\frac{\log n}{n} = o(r_n^2)$, then a.s. ultimately $\hat{F}_n(W_i(x_0, r)) > 0$, $i = 1, \dots, 5$; so $x_0 \in \mathcal{C}(\{X_{1,n}, X_{2,n}, \dots, X_{k,n}\})$. Therefore, what remains is to prove that the probability is 1 that $\log(n)/n = o(r_n^2)$.

Let a_n be defined as in the theorem's statement; then Theorem 4 implies that

$$\left| \frac{\hat{F}_n(S(x_0, r_n))}{F(S(x_0, r_n))} - 1 \right| > \varepsilon + \frac{c}{a_n} \quad \text{wp1ofo} \quad (45)$$

It follows that

$$\left| \frac{\hat{F}_n(S(x_0, r_n))}{2\pi f(x_0)r_n^2 + o(r_n^2)} - 1 \right| > \varepsilon + \frac{c}{a_n} \quad \text{wp1ofo} \quad (46)$$

Because $\hat{F}_n(S(x_0, r_n)) \sim k(n)/n$, where $k(n)$ is as described in the statement of the theorem, a.s. ultimately, $\log n/n = o(r_n^2)$. ■

Conjecture: [LIME weights consistent for unbounded X]

Here we discuss what we know about whether the LIME weights are L^p consistent in Stone's sense if X is unbounded but with $E[\|X\|^p] < \infty$.

Denote $E[Y|X]$ with $f(X)$. Let $D_n = D_n(X, X_1, \dots, X_n) = \{\|X - X_{k,n}(X)\| < \delta_n\}$, where δ_n is to be specified. Suppose that $p > 1$ and $E[|Y|^p] < \infty$. Let $1 < p' < p$, and $q = p/(p - p')$. Then

$$\begin{aligned} E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}\right] &= E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}(I_{D_n} + I_{D_n^c})\right] \\ &= E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}I_{D_n^c}\right] + E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}I_{D_n}\right]. \end{aligned} \quad (47)$$

We study the first term of (47),

$$\begin{aligned} 0 &\leq E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}I_{D_n^c}\right] \\ &\leq (E^{1/p}\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}\right])^p E^{1/q}[I_{D_n^c}^q] \\ &\leq \left(\sum_{i=1}^n E^{1/p}\left[\left|w_{i,n}Y_i - f(X)\right|^{p'}\right]\right)^p P^{1/q}(D_n^c). \end{aligned}$$

The first inequality is a consequence of Hölder's inequality. The second follows from the norm inequality

and the fact that an indicator function to a positive real power is just itself. Now the last displayed product of two terms is at most

$$(nE^{1/p}[|w_{1,n}Y_1 - f(X)|^p])^{p'} P^{1/q}(D_n^c) \quad (48)$$

by the exchangeability of the learning sample. But (48) is bounded above by

$$\begin{aligned} n^{p'} P^{1/q}(D_n^c) (E^{1/p}[|w_{1,n}Y_1|^p] + E^{1/p}[|f(X)|^p])^{p'} &\leq n^{p'} P^{1/q}(D_n^c) (E^{1/p}[|Y|^p] + E^{1/p}[|f(X)|^p])^{p'} \\ &\leq n^{p'} P^{1/q}(D_n^c) (2E^{1/p}[|Y|^p])^{p'} \end{aligned}$$

because conditional expectations reduce norms. Showing that

$$E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^p I_{D_n^c}\right] \rightarrow 0 \quad (49)$$

is now seen to follow from showing that $n^{p'} P^{1/q}(D_n^c) \rightarrow 0$. There is a constant $K, 1 < K < \infty$ for which

$$K\|X - X_{k,n}(X)\|^d \inf_{S(X, X_{k,n}(X))} f \leq P(S(X, X_{k,n}(X)))$$

which is not trivial if there is a version of f and an $\eta > 0$ for which

$$\inf_{S(X, X_{k,n}(X))} f > \eta. \quad (50)$$

When this is the case, $\|X - X_{k,n}(X)\|$ and $P^{1/d}(S(X, X_{k,n}(X)))$ are of the same order of magnitude. It follows from Theorem 5 that when (50) holds, it is easy to choose δ_n so that (49) also holds. But now comes the difficulty. When the support of f has infinite Lebesgue measure, even when $f > 0$ everywhere on the non-empty interior of its support, there is no $\eta > 0$ for which (50) holds. Necessarily $\eta = \eta(X)$. We would like to argue that $P(S(X, X_{k,n}(X)))$ small ensures that $\|X - X_{k,n}(X)\|$ is small, except possibly on an event of ‘‘small enough’’ probability.

The discussion thus far has focused only on the first term of (47). That the second term tends to 0 follows from an argument like that for LIME consistency when $\|X\|$ is bounded. Though the proof remains in the details, based on the above arguments we conjecture that $E\left[\left|\sum_{i=1}^n w_{i,n}Y_i - f(X)\right|^{p'}\right] \rightarrow 0$.

ACKNOWLEDGMENTS

The authors thank the anonymous reviewers, L. Lorne Campbell, Michael P. Friedlander, Jon Kristian Hagene, Yitzhak Katznelson, Deirdre O'Brien, Santosh Srivastava, Charles Stone, and Robert Tibshirani for helpful comments.

REFERENCES

- [1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. New York: Springer-Verlag, 2001.
- [2] D. Loftsgaarden and C. Quesenberry, "A nonparametric estimate of a multivariate density function," *Ann. Math. Statistics*, vol. 36, pp. 1049–1051, 1965.
- [3] E. Fix and J. L. Hodges, "Discriminatory analysis, nonparametric discrimination: Consistency properties," *Technical Report 4*, 1951, USAF School of Aviation Medicine, TX.
- [4] Y. P. Mack and M. Rosenblatt, "Multivariate k-nearest neighbor density estimates," *Journal of Multivariate Analysis*, vol. 9, pp. 1–15, 1979.
- [5] D. W. Scott, *Multivariate Density Estimation: theory, practice, and visualization*. New York: Wiley, 1992.
- [6] C. J. Stone, "Consistent nonparametric regression," *The Annals of Statistics*, vol. 5, no. 4, pp. 595–645, 1977.
- [7] M. P. Friedlander and M. R. Gupta, "On minimizing distortion and relative entropy," *To appear, IEEE Trans. on Information Theory*, 2005, preprint available at www.ee.washington.edu/research/guptalab.
- [8] S. Kullback, *Information Theory and Statistics*. New York: Wiley, 1959.
- [9] "www.stanford.edu/dept/msande/faculty/saunders," 2002.
- [10] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Information Theory*, vol. 13, pp. 21–27, 1967.
- [11] T. M. Cover, "Estimation by the nearest-neighbor rule," *IEEE Trans. on Information Theory*, vol. 14, no. 1, pp. 50–55, 1968.
- [12] R. M. Gray, *Entropy and Information Theory*. New York: Springer-Verlag, 1990.
- [13] E. T. Jaynes, "On the rationale of maximum entropy methods," *Proc. of the IEEE*, vol. 70, no. 9, pp. 939–952, 1982.
- [14] T. Cover and J. Thomas, *Elements of Information Theory*. United States of America: John Wiley and Sons, 1991.
- [15] N. Wu, *The Maximum Entropy Method*. Berlin: Springer-Verlag, 1997.
- [16] T. C. Hesterberg, "The bootstrap and empirical likelihood," *Proc. of the Section on Statistical Computing, American Statistical Association*, pp. 34–36, 1997.
- [17] T. Kohonen, G. Barna, and R. Chrisley, "Statistical pattern recognition with neural networks: benchmarking studies," *IEEE Intl. Conf. on Neural Networks*, vol. 1, pp. 61–68, 1988.
- [18] R. M. Gray and R. A. Olshen, "Vector quantization and density estimation," *Proc. of the Compression and Complexity of Sequences Conference*, pp. 172–193, 1997.
- [19] L. Devroye, L. Györfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*. New York: Springer-Verlag Inc., 1996.
- [20] J. Rice, "Boundary modification for kernel regression," *Communications in Statistics, Theory and Methods*, vol. 13, pp. 893–900, 1984.
- [21] T. Hastie and C. Loader, "Local regression: automatic kernel carpentry," *Statistical Science*, vol. 8, no. 2, pp. 120–143, 1993.
- [22] B. Ripley, *Pattern recognition and neural nets*. Cambridge: Cambridge University Press, 2001.

- [23] J. H. Friedman, "On bias, variance, 0/1 loss, and the curse-of-dimensionality," *Data mining and knowledge discovery*, vol. 1, no. 1, pp. 55–77, 1997.
- [24] D. B. O'Brien, M. R. Gupta, and R. M. Gray, "Analysis and classification of internal pipeline images," *Proc. of the IEEE Int. Conf. on Image Proc.*, 2003.
- [25] C. S. Peirce, *The philosophy of Peirce: selected writings*. Great Britain: Jarrold and Sons Limited, 1956.
- [26] W. Kneale, *Probability and Induction*. Oxford: Clarendon Press, 1949.
- [27] H. Kang, *Color Technology for Electronic Imaging Devices*. United States of America: SPIE Press, 1997.
- [28] "www.matlab.com," 2002, matlab version 6.1 by Mathworks.
- [29] W. H. Press, W. T. Vetterling, S. A. Teukolsky, and B. P. Flannery, *Numerical Recipes in C*, 2nd ed. Cambridge University Press, 1999.
- [30] G. Lugosi and K. Zeger, "Concept learning using complexity regularization," *IEEE Trans. on Information Theory*, vol. 42, pp. 48–54, 1996.
- [31] A. R. Barron and T. Cover, "Minimum complexity density estimation," *IEEE Trans. on Information Theory*, vol. 37, pp. 1034–1054, 1991.
- [32] A. Najmi, *Data compression, model selection and statistical inference*. Stanford, CA: Stanford University PhD Dissertation, 1999.
- [33] C. J. Stone, "Optimal global rates of convergence for nonparametric regression," *Annals of Statistics*, vol. 10, pp. 1040–1053, 1982.
- [34] P. J. Bickel and L. Breiman, "Sums of functions of nearest neighbor distances, moment bounds, limit theorems and a goodness of fit test," *Annals of Probability*, vol. 11, no. 1, pp. 185–214, 1983.
- [35] L. Breiman, J. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. United States of America: Chapman and Hall, 1984.
- [36] L. Gordon and R. A. Olshen, "Almost surely consistent nonparametric regression from recursive partitioning schemes," *J. of Multivariate Analysis*, vol. 15, pp. 146–163, 1984.
- [37] P. Bickel and R. R. Bahadur, "Substitution in conditional expectation," *Annals of Mathematical Statistics*, vol. 39, pp. 442 – 456, 1968.
- [38] M. de Guzmán, *Differentiation of Integrals in R^n* . Berlin: Springer Verlag, 1975.
- [39] A. Garsia, *Topics in Almost Everywhere Convergence*. Chicago: Markham, 1970.
- [40] D. Pollard, *Convergence of Stochastic Processes*. New York: Springer Verlag, 1984.