16th Conference on Learning Theory (COLT) 2003, Washington DC, August 2003.

Learning with Equivalence Constraints, and the Relation to Multiclass Learning

Aharon Bar-Hillel and Daphna Weinshall

School of Computer Sci. and Eng. & Center for Neural Computation
Hebrew University, Jerusalem 91904, Israel
{aharonbh,daphna}@cs.huji.ac.il
WWW home page: http://www.ca.huji.ac.il/~daphna

Abstract. We study the problem of learning partitions using equivalence constraints as input. This is a binary classification problem in the product space of pairs of datapoints. The training data includes pairs of datapoints which are labeled as coming from the same class or not. This kind of data appears naturally in applications where explicit labeling of datapoints is hard to get, but relations between datapoints can be more easily obtained, using, for example, Markovian dependency (as in video clips).

Our problem is an unlabeled partition problem, and is therefore tightly related to multiclass classification. We show that the solutions of the two problems are related, in the sense that a good solution to the binary classification problem entails the existence of a good solution to the multiclass problem, and vice versa. We also show that bounds on the sample complexity of the two problems are similar, by showing that their relevant 'dimensions' (VC dimension for the binary problem, Natarajan dimension for the multiclass problem) bound each other. Finally, we show the feasibility of solving multiclass learning efficiently by using a solution of the equivalent binary classification problem. In this way advanced techniques developed for binary classification, such as SVM and boosting, can be used directly to enhance multiclass learning.

1 Introduction

Multiclass learning is about learning a concept over some input space, which takes a discrete set of values $\{0, 1, \ldots, \mathbf{M}-1\}$. A tightly related problem is data partitioning, which is about learning a partitioning of data to $\mathbf{M}$ discrete sets. The latter problem is equivalent to unlabelled multiclass learning, namely, all the multiclass concepts which produce the same partitioning but with a different permutation of labels are considered the same concept.

Most of the work on multiclass partitioning of data has focused on the first variant, namely, the learning of an explicit mapping from datapoints to $\mathbf{M}$ discrete labels. It is assumed that the training data is obtained in the same form, namely, it is a set of datapoints with attached labels taken from the set $\{0, 1, \ldots, \mathbf{M}-1\}$. On the other hand, unlabeled data partitioning requires as training data only equivalence relations between pairs of datapoints; namely, for each pair of datapoints a label is assigned to indicate whether the pair originates from the same class or not. While it is straightforward to generate such binary labels on pairs of points from multiclass labels on individual points, the other direction is not as simple.

It is therefore interesting to note that equivalence constraints between pairs of datapoints may be easier to obtain in many real-life applications. More specifically, in data with natural Markovian dependency between successive datapoints (e.g., a video clip), there are automatic means to determine whether two successive datapoints (e.g., frames) come from the same class or not. In other applications, such as distributed learning where labels are obtained from many uncoordinated teachers, the subjective labels are meaningless, and the major information lies in the equivalence constraints which the subjective labels impose on the data. More details are given in [12].

Multiclass classification appears like a straightforward generalization of the binary classification problem, where the concept takes only two values $\{0, 1\}$. But while there is a lot of work on binary classification, both theoretical and algorithmic, the problem of multiclass learning is less understood. The VC dimension, for example, can only be used to characterize the learnability and sample complexity of binary functions. Generalizing this notion to multiclass classification has not been straightforward; see [4] for the details regarding a number of such generalizations and the relations between them.

On a more practical level, most of the algorithms available are best suitable (or only work for) the learning of binary functions. Support vector machines (SVM) [14] and boosting techniques [13] are two important examples. A possible solution is to reduce the problem to the learning of a number of binary classifiers ($O(\mathbf{M})$ or $O(\mathbf{M}^2)$), and then combine the classifiers using for example a winner-takes-all strategy [7]. The use of error correcting code to combine the binary classifiers was first suggested in [5]. Such codes were used in several successful generalizations to existing techniques, such as multiclass SVM and multiclass boosting [6, 1]. These solutions are hard to analyze, however, and only recently have we started to understand the properties of these algorithms, such as their sample complexity [7]. Another possible solution is to assume that the data distribution is known and construct a generative model, e.g., a Gaussian mixture model. The main drawback of this approach is the strong dependence on the assuption that the distribution is known.

In this paper we propose a different approach to multiclass learning. For each multiclass learning problem, define an equivalent binary classification problem. Specifically, if the original problem is to learn a multiclass classifier over data space X, define a binary classification problem over the product space $X \times X$, which includes all pairs of datapoints. In the binary classification problem, each pair is assigned the value 1 if the two datapoints come from the same class, and 0 otherwise. Hence the problem is reduced to the learning of a *single* binary classifier, and any existing tool can be used. Note that we have eliminated the problem of combining $\mathbf{M}$ binary classifiers. We need to address, however, the problems of how to generate the training sample for the equivalent binary problem, and how to obtain a partition of X from the learned concept in the product space.

A related idea was explored algorithmically in [11], where multiclass learning was translated to a binary classification problem over the *same* space X, using the difference between datapoints as input. This embedding in rather problematic, however, since the binary classification problem is ill-defined; it is quite likely that the same value would correspond to the difference between two vectors from the same class, and the difference between two other vectors from two different classes.

In the rest of this paper we study the properties of the binary classification problem in the product space, and their relation to the properties of the equivalent multiclass problem. Specifically, in Section 2.1 we define, given a multiclass problem, the equivalent binary classification problem, and state its sample complexity using the usual PAC framework. In Section 2.2 we show that for any solution of the product space problem with error e_{pr}, there is a solution of the multiclass problem with error e_o, such that

$$\frac{e_{pr}}{2} < e_o < \sqrt{2\mathbf{M}e_{pr}}$$

However, under mild assumptions, a stronger version for the right inequality exists, showing that the errors in the original and the product space are lineary related:

$$e_o < (\frac{e_{pr}}{K})$$

where K is the frequency of the smallest class. Finally, in Section 2.3 we show that the sample complexity of the two problems is similar in the following sense: for S_N the Natarajan dimension of the the multiclass problem, S_{VC} the VC-dimension of the equivalent binary problem, and $\mathbf{M}$ the number of classes, the following relation holds

$$\frac{S_N}{f_1(\mathbf{M})} - 1 \leq S_{VC} \leq f_2(\mathbf{M})S_N$$

where $f_1(\mathbf{M})$ is $O(\mathbf{M}^2)$ and $f_2(\mathbf{M})$ is $O(log\mathbf{M})$.

In order to solve a multiclass learning problem by solving the equivalent binary classification problem in the product space, we need to address two problems. First, a sample of independent points in X does not generate an independent sample in the product space. We note, however, that every n independent points in X trivially give $\frac{n}{2}$ independent pairs in the product space, and therefore the bounds above still apply up to a factor of $\frac{1}{2}$. We believe that the bounds are actually better, since a sample of n independent labels gives an order of $\mathbf{M}n$ non-trivial labels on pairs. By non-trivial we mean that given less than $\mathbf{M}n$ labels on pairs of points from $\mathbf{M}$ classes of the same size, we cannot deterministically derive the labels of the remaining pairs. This problem is more acute in the other direction, namely, it is actually not possible to generate a set of labels on individual points from a set of equivalence constraints on pairs of points.

Second, and more importantly, the approximation we learn in the product space may not represent any partition. A binary product space function f represents a partition only if it is an indicator of an equivalence relation, i.e. the relation $f(x1, x2) = 1$ is reflexive, symetric and transitive. It can be readily shown that f represents a partition, i.e., $\exists g, s.t. f = U(g))$ iff the function $1 - f$ is a binary metric. While this condition holds for our target concept, it doesn't hold for its approximation in the general case, and so an approximation will not induce any obvious partition on the original space.

To address this problem, we show in section 3 how an ε-good hypothesis f in the product space can be used to build an original space classifier with error linear in ε. First we show how f enables us, under certain conditions, to partition data in the original space with error linear in ε. Given the partitioning, we claim that a classifier can be built by using f to compare new presented data points to the partitioned sample. A similar problem was studied in [2], using the same kind of approximation. However, different criteria are optimized in the two papers: in [2] $e_{pr}(\bar{g}, f)$ is minimized (i.e., the product space error), while in our work a partition g is sought which minimizes $e_o(g, c)$ (i.e., the error in the original space of g w.r.t. the original concept).

2 From M-partitions to binary classifiers

In this section we show that multiclass classification can be translated to binary classification, and that the two problems are equivalent in many ways. First, in section 2.1 we formulate the binary classification problem whose solution is equivalent to a given multiclass problem. In section 2.2 we show that the solutions of the two problems are closely related: a good hypothesis for the multiclass problem provides a good hypothesis for the equivalent binary problem, and vice versa. Finally, in section 2.3 we show that the sample complexity of the two problems is similar.

2.1 PAC framework in the product space

Let us introduce the following notations:

- X: the input space.
- $\mathbf{M}$: the number of classes.
- D: the sampling distribution (measure) over X.
- c: a target concept over X; it is a labeled partition of X, $c : X \to \{0, \ldots, \mathbf{M}-1\}$. For each such concept, $c^{-1}(j) \in X$ denotes the cluster of points labeled j by c.
- $\mathcal{H}$: a family of hypotheses; each hypothesis is a function $h : X \to \{0, \ldots, \mathbf{M}-1\}$.

– $e(h, c)$: the error in X of a hypothesis $h \in \mathcal{H}$ with respect to c, defined as

$$e(h, c) = D(c(x) \neq h(x))$$

Given an unknown target concept c, the learning task is to find a hypothesis $h \in \mathcal{H}$ with low error $e(h, c)$. Usually it is assumed that a set of labeled datapoints is given during training. In this paper we do not assume to have access to such training data, but instead have access to a set of labeled equivalence constraints on pairs of datapoints. The label tells us whether the two points come from the same (unknown) class, or not. Therefore the learning problem is transformed as follows:

For any hypothesis h (or c), define a functor U which takes the hypothesis as an argument and outputs a function $\bar{h}, \bar{h} : X \times X \to \{0, 1\}$. Specifically:

$$\bar{h}(x, y) = 1_{h(x)=h(y)}$$

Thus $\bar{h}$ expresses the implicit equivalence relations induced by the concept h on pairs of datapoints.

The functor U is not injective: two different hypotheses h_1 and h_2 may result in the same $\bar{h}$. This, however, happens only when h_1 and h_2 differ only by a permutation of their corresponding labels, while representing the same partition; $\bar{h}$ therefore represents an unlabeled partition.

We can now define a second notion of error between unlabeled partitions over the product space $X \times X$:

$$e(\bar{h}, \bar{c}) = D \times D(\bar{h}(x, y) \neq \bar{c}(x, y))$$

where $\bar{c}$ is obtained from c by the functor U. This error measures the probability of disagreement between $\bar{h}$ and $\bar{c}$ with regard to equivalence queries. It is a rather intuitive measure for the comparison of unlabeled partitions. The problem of learning a partition can now be cast as a regular PAC learning problem, since $\bar{h}$ and $\bar{c}$ are binary hypotheses. Specifically:

> Let $X \times X$ denote the input space, $D \times D$ denote the sampling probability over the input space, and $\bar{c}$ denote the target concept. Let the hypotheses family be the family $\bar{\mathcal{H}} = \{\bar{h} : h \in \mathcal{H}\}$.[1]

Now we can use the VC dimension and PAC-learning theory on sample complexity, in order to characterize the sample complexity of learning the binary hypothesis $\bar{h}$. More interestingly, we can then compare our results with results on sample complexity obtained directly for the multiclass problem.

2.2 The Connection between solution quality of the two problems

In this section we show that a good (binary) hypothesis in the product space can be used to find a good hypothesis (partition) in the original space, and

[1] Note that $\bar{\mathcal{H}}$ is of the same size as $\mathcal{H}$ only when $\mathcal{H}$ does not contain hypotheses which are identical with respect to the partition of X.

vice versa. Note that the functor U, which connects hypotheses in the original space to hypotheses in the product space, is not injective, and therefore it has no inverse. Therefore, in order to asses the difference between two hypotheses $\bar{h}$ and $\bar{c}$, we must choose h and c such that $\bar{h} = U(h)$ and $\bar{c} = U(c)$, and subsequently compute $e(h, c)$.

We proceed by showing three results: Thm. 1 shows that in general, if we have a hypothesis in product space with some error ε, there is a hypothesis in the original space with error $O(\sqrt{M\varepsilon})$. However, if ε is small with respect to the smallest class probability K, Thm. 2 shows that the bound is linear, namely, there is a hypothesis in the original space with error $O(\frac{\varepsilon}{K})$. In most cases, this is the range of interest. Finally, Thm. 3 shows the other direction: if we have a hypothesis in the original space with some error ε, its product space hypothesis $U(h) = \bar{h}$ has an error smaller than 2ε.

Before proceeding we need to introduce some more notations: Let c and h denote two partitions of X into $\mathbf{M}$ classes. Define the joint distribution matrix $P = \{p_{ij}\}_{i,j=0}^{\mathbf{M}-1}$ as follows:

$$p_{ij} \stackrel{\Delta}{=} D(c(x) = i, h(x) = j)$$

Using this matrix we can express the probability of error in the original space and the product space.

1. The error in X is

$$e(h, c) = D(c(x) \neq h(x)) = \sum_{i=0}^{\mathbf{M}-1} \sum_{j \neq i} p_{ij}$$

2. The error in the product space is

$$\begin{aligned}
e(\bar{h}, \bar{c}) &= D \times D([\bar{c}(x, y) = 1 \wedge \bar{h}(x, y) = 0] \vee [\bar{c}(x, y) = 0 \wedge \bar{h}(x, y) = 1]) \\
&= \sum_{i=0}^{\mathbf{M}-1} \sum_{j=0}^{\mathbf{M}-1} D(c(x) = i, h(x) = j) \cdot (D(\{y | c(y) = i, h(y) \neq j\}) \\
&\quad + D(\{y | c(y) \neq i, h(y) = j\})) \\
&= \sum_{i=0}^{\mathbf{M}-1} \sum_{j=0}^{\mathbf{M}-1} p_{ij} \left(\sum_{k \neq i} p_{kj} + \sum_{k \neq j} p_{ik} \right)
\end{aligned}$$

Theorem 1. *For any two product space hypotheses $\bar{h}, \bar{c}$, there are h, c such that $\bar{h} = U(h), \bar{c} = U(c)$ and*

$$e(h, c) \leq \sqrt{2\mathbf{M}e(\bar{h}, \bar{c})}$$

where $\mathbf{M}$ is the number of equivalence classes of $\bar{h}, \bar{c}$.

The proof appears in Appendix A. We note that the bound is tight as a function of ε since there are indeed cases where $e(c,h) = O(\sqrt{(}e(\bar{h},\bar{c})))$. A simple example of such 3-class problem occurs when the matrix of joint distribution is the following:

$$P = \begin{pmatrix} 1-3q & 0 & 0 \\ 0 & q & q \\ 0 & 0 & q \end{pmatrix}$$

Here $e(c,h) = q$ and $e(\bar{c},\bar{h}) = 4q^2$. The next theorem shows, however, that this situation cannot occur if $e(\bar{c},\bar{h})$ is small compared to the smallest class frequency.

Theorem 2. *Let c denote a target partition and h a hypothesis, and let $\bar{c},\bar{h}$ denote the corresponding hypotheses in the product space. Denote the size of the minimal class of c by $K = \min_{i\in\{0,\ldots,\mathbf{M}-1\}} D(c^{-1}(i))$, and the product space error $\varepsilon = e(\bar{c},\bar{h})$.*

$$\varepsilon < \frac{K^2}{2} \quad \Longrightarrow \quad e(f\circ h, c) < \frac{\varepsilon}{K} \tag{1}$$

where $f : \{0,\ldots,\mathbf{M}-1\} \to \{0,\ldots,\mathbf{M}-1\}$ is a bijection matching the labels of h and c.

Proof. We start by showing that if the theorem's condition holds, then there is a natural correspondence between the classes of c and h:

Lemma 1. *If the condition in (1) holds, then there exists a bijection $J : \{0,\ldots,\mathbf{M}-1\} \to \{0,\ldots,\mathbf{M}-1\}$ such that*

- $p_{i,J(i)} > \sqrt{\frac{\varepsilon}{2}}$
- $p_{i,l} < \sqrt{\frac{\varepsilon}{2}}$ *for all* $l \neq J(i)$
- $p_{l,J(i)} < \sqrt{\frac{\varepsilon}{2}}$ *for all* $l \neq i$

Proof. Denote the class probabilities as $p_i^c = D(c^{-1}(i))$; clearly

$$p_i^c = \sum_{j=0}^{\mathbf{M}-1} p_{ij}$$

We further define for each class i of c its internal error $\varepsilon_i = \sum_{j=0}^{\mathbf{M}-1} p_{ij}(p_i^c - p_{ij})$. The rationale for this definition follows from the following inequality:

$$\varepsilon = \sum_{i=0}^{\mathbf{M}-1}\sum_{j=0}^{\mathbf{M}-1} p_{ij}\Big(\sum_{\substack{k=0\\k\neq i}}^{\mathbf{M}-1} p_{kj} + \sum_{\substack{k=0\\k\neq j}}^{\mathbf{M}-1} p_{ik}\Big) \geq \sum_{i=0}^{\mathbf{M}-1}\sum_{j=0}^{\mathbf{M}-1} p_{ij}(p_i^c - p_{ij}) = \sum_{i=0}^{\mathbf{M}-1} \varepsilon_i$$

We first observe that each row in matrix P contains at least one element bigger than $\sqrt{\frac{\varepsilon}{2}}$. Assume to the contrary that no such element exists in class i; then

$$\varepsilon \geq \varepsilon_i = \sum_{j=0}^{\mathbf{M}-1} p_{ij}(p_i^c - p_{ij}) > \sum_{j=0}^{\mathbf{M}-1} p_{ij}\Big(\sqrt{2\varepsilon} - \sqrt{\frac{\varepsilon}{2}}\Big) = \sqrt{\frac{\varepsilon}{2}}\sum_{j=0}^{\mathbf{M}-1} p_{ij} \geq \sqrt{\frac{\varepsilon}{2}}\cdot\sqrt{2\varepsilon} = \varepsilon$$

in contradiction.

Second, we observe that the row element bigger than $\sqrt{\frac{\varepsilon}{2}}$ is unique. This follows from the following argument: for any two elements p_{ij_1}, p_{ij_2} in the same row:

$$\varepsilon \geq \sum_{j=0}^{\mathbf{M}-1} p_{ij}(\sum_{\substack{k=0 \\ k\neq i}}^{\mathbf{M}-1} p_{kj} + \sum_{\substack{k=0 \\ k\neq j}}^{\mathbf{M}-1} p_{ik}) \geq \sum_{j=0}^{\mathbf{M}-1} p_{ij} \sum_{\substack{k=0 \\ k\neq j}}^{\mathbf{M}-1} p_{ik} \geq 2p_{ij_1}p_{ij_2}$$

Hence it is not possible that both the elements p_{ij_1} and p_{ij_2} are bigger than $\sqrt{\frac{\varepsilon}{2}}$. The uniqueness of an element bigger than $\sqrt{\frac{\varepsilon}{2}}$ in a column follows from an analogous argument with regard to the columns, which completes the proof of the lemma. □

Denote $f = J^{-1}$, and let us show that $\sum_{i=0}^{\mathbf{M}-1} p_{i,f(i)} > 1 - \frac{\varepsilon}{K}$. We start by showing that $p_{i,f(i)}$ cannot be 'too small':

$$\varepsilon_i = \sum_{j=0}^{\mathbf{M}-1} p_{ij}(p_i^c - p_{ij}) = p_{i,f(i)}(p_i^c - p_{i,f(i)}) + \sum_{\substack{j=0 \\ j\neq f(i)}}^{\mathbf{M}-1} p_{ij}(p_i^c - p_{ij})$$

$$\geq p_{i,f(i)}(p_i^c - p_{i,f(i)}) + \sum_{\substack{j=0 \\ j\neq f(i)}}^{\mathbf{M}-1} p_{ij}p_{i,f(i)} = 2p_{i,f(i)}(p_i^c - p_{i,f(i)})$$

This gives a quadratic inequality

$$p_{i,f(i)}^2 - p_i^c p_{i,f(i)} + \frac{\varepsilon_i}{2} \geq 0$$

which holds for $p_{i,f(i)} \geq \frac{p_i^c + \sqrt{(p_i^c)^2 - 2\varepsilon_i}}{2}$ or for $p_{i,f(i)} \leq \frac{p_i^c - \sqrt{(p_i^c)^2 - 2\varepsilon_i}}{2}$. Since

$$\sqrt{(p_i^c)^2 - 2\varepsilon_i} = \sqrt{(p_i^c)^2(1 - \frac{2\varepsilon_i}{(p_i^c)^2})} = p_i^c\sqrt{1 - \frac{2\varepsilon_i}{(p_i^c)^2}} > p_i^c(1 - \frac{2\varepsilon_i}{(p_i^c)^2}) = p_i^c - \frac{2\varepsilon_i}{p_i^c}$$

it must hold that either $p_{i,f(i)} > p_i^c - \frac{\varepsilon_i}{p_i^c}$ or $p_{i,f(i)} < \frac{\varepsilon_i}{p_i^c}$. But the second possiblity that $p_{i,f(i)} < \frac{\varepsilon_i}{p_i^c}$ leads to contradiction with condition (1) since

$$\sqrt{\frac{\varepsilon}{2}} < p_{i,f(i)} < \frac{\varepsilon_i}{p_i^c} \leq \frac{\varepsilon}{K} \implies K < \sqrt{2\varepsilon}$$

Therefore $p_{i,f(i)} > p_i^c - \frac{\varepsilon_i}{p_i^c}$.

Summing the inequalities over i, we get

$$\sum_{i=0}^{\mathbf{M}-1} p_{i,f(i)} > \sum_{i=0}^{\mathbf{M}-1} p_i^c - \frac{\varepsilon_i}{p_i^c} \geq 1 - \sum_{i=0}^{\mathbf{M}-1} \frac{\varepsilon_i}{K} \geq 1 - \frac{\varepsilon}{K}$$

This completes the proof of the theorem since

$$\begin{aligned} e(J\circ h,c) &= p(J\circ h(x)\neq c(x)) = 1-p(J\circ h(x)=c(x)) \\ &= 1-\sum_{i=0}^{\mathbf{M}-1} p(\{c(x)=i, h(x)=J^{-1}(i)) = 1-\sum_{i=0}^{\mathbf{M}-1} p_{i,f(i)} < \frac{\varepsilon}{K} = \frac{e(\bar{c},\bar{h})}{K} \end{aligned}$$

□

Corollary 1. *If the classes are equiprobable, namely $\frac{1}{K}=\mathbf{M}$, we get a bound of $\mathbf{M}\varepsilon$ on the error in the original space.*

Corollary 2. *As $K\to\sqrt{2\varepsilon}$, the lowest allowed value according to the theorem condition, we get an error bound approaching $\frac{\varepsilon}{\sqrt{2\varepsilon}}=\sqrt{\frac{\varepsilon}{2}}$. Hence the linear behavior of the bound on the original space error is lost near this limit, in accordance with Thm. 1.*

A bound in the other direction is much simpler to achieve:

Theorem 3. *For every two labeled partitions h,c: if $e(h,c)<\varepsilon$ then $e(\bar{h},\bar{c})<2\varepsilon$.*

Proof.

$$\begin{aligned} e(\bar{h},\bar{c}) &= \sum_{i=0}^{\mathbf{M}-1}\sum_{j=0}^{\mathbf{M}-1} p_{ij}\left[\sum_{k\neq i} p_{kj} + \sum_{k\neq j} p_{ik}\right] \\ &= \sum_{i=0}^{\mathbf{M}-1} p_{ii}[\sum_{k\neq i} p_{ki} + \sum_{k\neq i} p_{ik}] + \sum_{i=0}^{\mathbf{M}-1}\sum_{j\neq i} p_{ij}[\sum_{k\neq i} p_{kj} + \sum_{k\neq j} p_{ik}] \\ &\leq \sum_{i=0}^{\mathbf{M}-1} p_{ii}\cdot\varepsilon + \sum_{i=0}^{\mathbf{M}-1}\sum_{j\neq i} p_{ij} \leq \varepsilon+\varepsilon = 2\varepsilon \end{aligned}$$

□

2.3 The connection between sample size complexity

Several dimension-like measures of the sample complexity exist for multiclass probelms. However, these measures can be shown to be closely related [4]. We use here the Natarajan dimension, denoted as $S_N(\mathcal{H})$, to characterize the sample size complexity of the hypotheses family $\mathcal{H}$ [10, 4]. Since $\bar{\mathcal{H}}$ is binary, its sample size is characterized by its VC dimension $S_{VC}(\bar{\mathcal{H}})$ [14]. We will now show that each of these dimensions bounds the other up to a scaling factor which depends on $\mathbf{M}$. Specifically, we will prove the following double inequality:

$$\frac{S_N(\mathcal{H})}{f_1(\mathbf{M})} - 1 \leq S_{VC}(\bar{\mathcal{H}}) \leq f_2(\mathbf{M}) S_N(\mathcal{H}) \tag{2}$$

where $f_1(\mathbf{M}) = O(\mathbf{M}^2)$ and $f_2(\mathbf{M}) = O(log\mathbf{M})$.

Theorem 4. *Let $S_N^U(\mathcal{H})$ denote the uniform Natarajan dimension of $\mathcal{H}$ as defined by Ben-David et al. [4]; then*

$$S_N^U(\mathcal{H}) - 1 \leq S_{VC}(\bar{\mathcal{H}})$$

Proof. Let d denote the uniform Natarajan dimension, $d = S_N^U(\mathcal{H})$. It follows that there are $k, l \in \{0, \ldots, \mathbf{M}-1\}$ and $\{x_i\}_{i=1}^d$ points in X such that

$$\{0,1\}^d \subseteq \{(\psi_{k,l} \circ h(x_1), \ldots, \psi_{k,l} \circ h(x_d)) | h \in \mathcal{H}\}$$

where $\psi_{k,l} : \{0, \ldots, \mathbf{M}-1\} \rightarrow \{0, 1, *\}$, $\psi_{k,l}(k) = 1$, $\psi_{k,l}(l) = 0$, and $\psi_{k,l}(u) = *$ for every $u \neq k, l$.

Next we show that the set of product space points $\{\bar{x}_i = (x_i, x_{i+1})\}_{i=1}^{d-1}$ is VC-shattered by $\bar{\mathcal{H}}$. Assume an arbitrary $\bar{b} \in \{0,1\}^{d-1}$. Since by definition $\{x_i\}_{i=1}^d$ is $\psi_{k,l}$-shattered by $\mathcal{H}$, we can find $h \in \mathcal{H}$ which assigns $h(x_1) = l$ and gives the following assignments over the points $\{x_i\}_{i=2}^d$:

$$h(x_i) = \begin{pmatrix} k \ if \ \ h(x_{i-1}) = l \ and \ \bar{b}(i-1) = 0 \\ l \ if \ \ h(x_{i-1}) = l \ and \ \bar{b}(i-1) = 1 \\ l \ if \ h(x_{i-1}) = k \ and \ \bar{b}(i-1) = 0 \\ k \ if \ h(x_{i-1}) = k \ and \ \bar{b}(i-1) = 1 \end{pmatrix}$$

By construction $(\bar{h}(\bar{x}_1), \ldots, \bar{h}(\bar{x}_{d-1})) = \bar{b}$. Since $\bar{b}$ is arbitrary, $\{\bar{x}_i\}_{i=1}^{d-1}$ is shattered by $\bar{\mathcal{H}}$, and hence $S_{vc}(\bar{\mathcal{H}}) \geq d-1$. □

The relation between the uniform Natarajan dimension and the Natarajan dimension is given by theorem 7 in [4]. In our case it is

$$S_N(H) \leq \frac{M(M-1)}{2} S_N^U(H)$$

Hence the proof of theorem 4 gives us the left bound of inequality 2.

Theorem 5. *Let $d_{pr} = S_N(\mathcal{H})$ denote the Natarajan dimension of $\mathcal{H}$, and $d_o = S_{VC}(\bar{\mathcal{H}})$ denote the VC dimension of $\bar{\mathcal{H}}$. Then*

$$S_{VC}(\bar{\mathcal{H}}) \leq 4.87 S_N(\mathcal{H}) \log(\mathbf{M}+1)$$

Proof. Let $X_{pr} = \{\bar{x}_i = (x_i^1, x_i^2)\}_{i=1}^{d_{pr}}$ denote a set of points in the product space which are shattered by $\bar{\mathcal{H}}$. Let $X_o = \{x_1^1, x_1^2, x_2^1, \ldots, x_{d_{pr}}^1, x_{d_{pr}}^2\}$ denote the corresponding set of points in the original space.

There is a set $Y_{pr} = \{\bar{h}_j\}_{j=1}^{2^{d_{pr}}}$ of $2^{d_{pr}}$ hypotheses in $\bar{\mathcal{H}}$, which are different from each other on X_{pr}. For each hypothesis $\bar{h}_j \in Y_{pr}$ there is a hypothesis $h \in \mathcal{H}$ such that $\bar{h} = U(h)$. If $\bar{h}_1 \neq \bar{h}_2 \in Y_{pr}$ then the corresponding h_1, h_2 are different on X_o. To see this, note that $\bar{h}_1 \neq \bar{h}_2$ implies the existence of $\bar{x}_i = (x_i^1, x_i^2) \in X_{pr}$ on which $\bar{h}_1(\bar{x}_i) \neq \bar{h}_2(\bar{x}_i)$. It is not possible in this case that both $h_1(x_i^1) = h_2(x_i^1)$ and $h_1(x_i^2) = h_2(x_i^2)$. Hence there are $2^{d_{pr}}$ hypotheses in $\mathcal{H}$ which are different on X_o, from which it follows that

$$|\{(h(x_1^1), h(x_1^2), \ldots, h(x_{d_{pr}}^1), h(x_{d_{pr}}^2)) | h \in \mathcal{H}\}| \geq 2^{d_{pr}} \tag{3}$$

The existence of an exponential number of assignments of $\mathcal{H}$ on the set X_o is not possible if $|X_o|$ is much larger than the Natarajan dimension of $\mathcal{H}$. We use Thm. 9 in [4] (proved in [8]) to argue that if the Natarajan dimension of $\mathcal{H}$ is d_o, then

$$|\{(h(x_1^1), h(x_1^2), \ldots, h(x_{d_{pr}}^1), h(x_{d_{pr}}^2))|h \in \mathcal{H}\}| \leq (\frac{2d_{pr}e(\mathbf{M}+1)^2}{2d_o})^{d_o} \tag{4}$$

where $\mathbf{M}$ is the number of classes. Combining (3) and (4) we get

$$2^{d_{pr}} \leq (\frac{2d_{pr}e(\mathbf{M}+1)^2}{2d_o})^{d_o}$$

Here the term on left side is exponential in d_{pr}, and the term on the right side is polynomial. Hence the inequality cannot be true asymptotically and d_{pr} is bounded.

We can find a convenient bound by following the proof of Thm. 10 in [4]. The algebraic details completing the proof are left to Appendix B. □

Corollary 3. *$\bar{\mathcal{H}}$ is learnable iff $\mathcal{H}$ is learnable.*

3 From product space approximations to original space classifiers

In section 3.1 we present an algorithm to partition a data set Y using a product space function which is ε-good over $Y \times Y$. f should only satisfy $e(f, \bar{c}) < \varepsilon$, but it doesn't have to be an equivalence relation indicator, and so in general there is no h such that $f = U(h)$. The partition generated is shown to have an error linear in ε. Then in section 3.2 we briefly discuss (without proof) how an ε-good product space hypothesis can be used to build a classifier with error $O(\varepsilon)$.

3.1 Partitioning using a product space hypothesis

Assume we are given a data set $Y = \{x_i\}_{i=1}^N$ of points drawn independently from the distribution over X. Let f denote a learned hypothesis from $\bar{\mathcal{H}}$, and denote the error of f over the product space $Y \times Y$ by

$$\varepsilon = e(\bar{c}, f) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} 1_{\bar{c}(x_i, y_j) \neq f(x_i, y_j)}$$

Denote by K the frequency $\frac{classsize}{N}$ of the smallest class in Y.

Note that since no explicit labels are given, we can only hope to find an approximation to c over Y up to a permutation of the labels. The following theorem shows that if ε is small enough compared to K and given f, there is a simple algorithm which is guaranteed to achieve an approximation to the partition represented by concept c with error linear in ε.

Theorem 6. *Using the notation defined above, if the following condition hold*

$$\varepsilon < \frac{K^2}{6}, \tag{5}$$

then we can find a partition g of Y with a simple procedure, such that $e(c, J \circ g) < \frac{6\varepsilon}{K}$. J here denotes a permutation $J : \{0, \ldots, \mathbf{M}-1\} \rightarrow \{0, \ldots, \mathbf{M}-1\}$ matching the labels of c and g.

In order to present the algorithm and prove the error bound as stated above, we first define several simple concepts.

Define the 'fiber' of a point $x \in Y$ under a function $h : X \times X \rightarrow \{0, 1\}$ as the following restriction of h:

$$fiber^h(x) : Y \rightarrow \{0, 1\}, \quad [fiber^h(x)](y) = h(x, y)$$

$fiber^h(x)$ is an indicator function of the points in Y which are in the same class with x according to h.

Let us now define the distance between two fibers. For two indicator functions $I_1, I_2 : Y \rightarrow \{0, 1\}$ let us measure the distance between them using the $L1$ metric over Y:

$$d(I_1, I_2) = Prob(I_1(x) \neq I_2(x)) = \frac{1}{N} \sum_{i=1}^{N} 1_{I_1(x_i) \neq I_2(x_i)} = \frac{1}{N} \sum_{i=1}^{N} |I_1(x_i) - I_2(x_i)|$$

Given two fibers $fiber^h(x), fiber^h(z)$ of a product space hypothesis, the L_1 distance between them has the form of

$$d(fiber^h(x), fiber^h(z)) = \frac{\#(Nei^h(x) \Delta Nei^h(z))}{N}$$

where $Nei^h(x) = \{y | h(x, y) = 1\}$. This gives us an intuitive meaning to the inter-fiber distance, namely, it is the frequency of sample points which are neighbors of x and not of z or vice versa.

The operator taking a point $x \in Y$ to $fiber^h(x)$ is therefore an embedding of Y in the metric space $L_1(Y)$. In the next lemma we see that if the conditions of Thm. 6 hold, most of the data set is well separated under this embedding, in the sense that points from the same class are near while points from different classes are far. This allows us to define a simple algorithm which uses this separability to find a good partitioning of Y, and prove that its error is bounded as required.

Lemma 2. *There is a set of 'good' points $\mathcal{G} \in Y$ such that $|Y \backslash \mathcal{G}| \leq \frac{3\varepsilon}{K} N$ (i.e., the set is large), and for every two points $x, y \in \mathcal{G}$:*

$$c(x) = c(y) \implies d(fiber^f(x), fiber^f(y)) < \frac{2K}{3}$$
$$c(x) \neq c(y) \implies d(fiber^f(x), fiber^f(y)) \geq \frac{4K}{3}$$

Proof. Define the 'good' set $\mathcal{G}$ as

$$\mathcal{G} = \{x | d(fiber^f(x), fiber^c(x)) < \frac{K}{3}\}$$

We start by noting that the complement of $\mathcal{G}$, the set of 'bad' points $\mathcal{B} = \{x | d(fiber^f_x, fiber^c_x) \geq \frac{K}{3}\}$, is small as the lemma requires. The argument is the following

$$\begin{aligned}\varepsilon = e(\bar{c}, f) = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} 1_{\bar{c}(x_i,y_j)=f(x_i,y_y)} =& \frac{1}{N^2}[\sum_{x_i \in \mathcal{B}}\sum_{j=1}^{N} 1_{\bar{c}(x_i,y_j)=f(x_i,y_y)} \\ &+ \sum_{x_i \in \mathcal{G}}\sum_{j=1}^{N} 1_{\bar{c}(x_i,y_j)=f(x_i,y_y)}] \geq \frac{1}{N^2}\sum_{x_i \in \mathcal{B}} \frac{K}{3} N = \frac{K}{3N}|\mathcal{B}|\end{aligned}$$

Next, assume that $c(x) = c(y)$ holds for two points $x, y \in \mathcal{G}$. Since $fiber^c(x) = fiber^c(y)$ we get

$$\begin{aligned}d(fiber^f(x), fiber^f(y)) &\leq d(fiber^f(x), fiber^c(x)) + d(fiber^c(x), fiber^c(y)) \\ &+ d(fiber^c(y), fiber^f(y)) < \frac{K}{3} + 0 + \frac{K}{3} = \frac{2K}{3}\end{aligned}$$

Finally, if $c(x) \neq c(y)$ then $fiber^c(x)$ and $fiber^c(y)$ are indicators of disjoint sets, each bigger or equal to K. Hence $d(fiber^c(x), fiber^c(y)) \geq 2K$ and we get

$$\begin{aligned}2K &\leq d(fiber^c(x), fiber^c(y)) \\ &\leq d(fiber^c(x), fiber^f(x)) + d(fiber^f(x), fiber^f(y)) + d(fiber^f(y), fiber^c(y)) \\ &\leq \frac{K}{3} + d(fiber^f(x), fiber^f(y)) + \frac{K}{3} \\ &\Longrightarrow d(fiber^f(x), fiber^f(y)) \geq \frac{4K}{3}\end{aligned}$$

□

It follows from the lemma that over the 'good' set $\mathcal{G}$, which contains more than $(1 - \frac{3\varepsilon}{K})N$ points, the classes are very well separated. Each class is concentrated in a $\frac{K}{3}$-ball and the different balls are $\frac{4K}{3}$ distant from each other. Intuitively, under such conditions almost any reasonable clustering algorithm can find the correct partitioning over this set; since the size of the remaining set of 'bad' points $\mathcal{B}$ is linear in ε, the total error is expected to be linear in ε too.

However, in order to prove a worst case bound we still face a certain problem. Since we do not know how to tell $\mathcal{G}$ from $\mathcal{B}$, the 'bad' points might obscure the partition. We therefore suggest the following greedy procedure to define a partition g over Y:

– Compute the fibers $fiber^f(x)$ for all $x \in Y$.

– Let $i = 0$, $S_0 = Y$; while $|S_i| > \frac{KN}{2}$ do:
  - for each point $x \in S_i$, compute the set of all points lying inside a sphere of radius $\frac{2K}{3}$ around x:
$$B_{\frac{2k}{3}}(x) = \{y \in S_i : d(fiber^f(x), fiber^f(y)) < \frac{2K}{3}\}$$
  - find $z = \arg\max_{x \in S_i} |B_{\frac{2K}{3}}(x)|$ and define $g(y) = i$ for every $y \in B_{\frac{2k}{3}}(z)$;
  - remove the points of $B_{\frac{2k}{3}}(z)$ from S_i: let $S_{i+1} = S_i \backslash B_{\frac{2k}{3}}(z)$, and $i = i+1$.

– Let $\mathbf{M}_g$ denote the number of rounds completed. Denote the domain on which g has been defined so far as G_0. Define g for the remaining points in $Y \backslash G_0$ as follows:
$$g(x) = \arg\min_{i \in \{0,\ldots,\mathbf{M}_g - 1\}} d(fiber^f(x), I_{\{g^{-1}(i)\}})$$
where $I_{\{g^{-1}(i)\}}$ is the indicator function of cluster i of g. Note, however, that the way g is defined over this set is not really important since, as we shall see, the set is small.

The proof for the error bound of g starts with two lemmas:

1. The first lemma uses lemma 2 to show that each cluster defined by g intersects only a single set of the form $c^{-1}(i) \cap \mathcal{G}$.
2. The second lemma shows that due to the greedy nature of the algorithm, the sets $g^{-1}(i)$ chosen at each step are big enough so that each intersects at least one of the sets $\{c^{-1}(j) \cap \mathcal{G}\}_{j=1}^{\mathbf{M}-1}$.
   It immediately follows that each set $g^{-1}(i)$ intersects a single set $\{c^{-1}(j) \cap \mathcal{G}\}$, and a match between the clusters of g and the classes of c can be established, while $Y \backslash G_0$ can be shown to be $O(\varepsilon)$ small.
3. Finally, the error of g is bounded by showing that if $x \in G_0 \cap \mathcal{G}$ then x is classified correctly by g.

Details of the lemmas and proofs are given at Appendix C, which completes the proof of Thm. 6.

3.2 Classifing using a product space hypothesis

Given an ε good product space hypothesis f, we can build a multiclass classifier as follows: Sample N unlabeled data points $Y = \{x_i\}_{i=1}^N$ from X and partition them using the algorithm presented in the previous subsection. A new point Z is classified as a member of the class l where
$$l = \arg\min_{i \in \{0,..,M-1\}} d(fiber^f(z), I_{g^{-1}(i)})$$
The following theorem bounds the error of such a classifier

Theorem 7. *Assume the error probability of f over $X \times X$ is $e(f, \bar{c}) = \varepsilon < \frac{K^2}{8}$. For each $\delta > 0$, $l > 4$: if $N > \frac{3l}{K(\frac{l}{4}-1)^2} log(\frac{1}{\delta})$, then the error of the classifier proposed is lower than $\frac{l\varepsilon}{K} + \delta$*

The proof is omitted.

4 Concluding remarks

We showed in this paper that learning in the product space produces good multiclass classifiers of the original space, and that the sample complexity of learning in the product space is comparable to the complexity of learning in the original space. We see the significance of these results in two aspects: First, since learning in the product space always involves only binary functions, we can use the full power of binary classification theory and its many efficient algorithms to solve multiclass classification problems. In contrast, the learning toolbox for multiclass problems in the original space is relatively limited. Second, the automatic acquisition of product space labels is plausible in many domains in which the data is produced by some Markovian process. In such domains the learning of interesting concepts without any human supervision may be possible.

References

1. E. Allwein, R. Schapire, and Y. Singer. Reducing Multiclass to Binary: A Unifying Approach for Margin Classifiers. *Journal of Machine Learning Research*, 1:113-141, 2000.
2. N. Bansal, A. Blum, and S. Chawla. Correlation Clustering. In Proc. FOCS 2002, pages 238-247.
3. A. Bar-Hillel, and D. Weinshall. Learning with Equivalence Constraints. HU Technical Report 2003-38, in http://www.cs.huji.ac.il/˜daphna.
4. S. Ben-David, N. Cesa-Bianchi, D. Haussler, and P. H. Long. Characterizations of learnability for classes of 0, . . . , n-valued functions.
5. T. G. Dietterich, and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artifical Intelligence Research*, 2:263-286, 1995.
6. V. Guruswami and Amit Sahai. Multiclass learning, Boosting, and Error-Correcting codes. In Proc. COLT, 1999.
7. S. Har-Peled, D. Roth, D. Zimak. Constraints classification:A new approach to multiclass classification and ranking. In Proc. NIPS, 2002.
8. D. Haussler and P.M. Long. A generalization of Sauer's lemma. Technical Report UCSU-CRL-90-15. UC Santa Cruz, 1990.
9. M. J. Kearns and U. V. Vazirani. An Introduction to Computational Learning Theory. MIT Press, 1994.
10. B. K. Natarajan. On learning sets and functions. *Machine Learning*, 4:67-97, 1989.
11. P. J. Phillips. Support vector machines applied to face recognition. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 11*, page 803. MIT Press, 1998.
12. T. Hertz, N. Shental, A. Bar-Hillel, and D. Weinshall. Enhancing Image and Video Retrieval: Learning via Equivalence Constraints. In Proc. of IEEE Conference on Computer Vision and Pattern Recognition, 2003.
13. R. E. Schapire. A brief introduction to boosting. In Proc. of the Sixteenth International Joint Conference on Artificial Intelligence, 1999.
14. V. N. Vapnik. The Nature of Statistical Learning. Springer, 1995.

A Proof of Thm. 1

In order to prove this theorem, we first describe a procedure for finding c and h such that their labels are matched. We then look for a lower bound on the ratio

$$\frac{e(\bar{h},\bar{c})}{e(h,c)^2}=\frac{\sum_{i=0}^{\mathbf{M}-1}\sum_{j=0}^{\mathbf{M}-1}p_{ij}(\sum_{k\neq i}p_{kj}+\sum_{k\neq j}p_{ik})}{(1-\sum_{i=0}^{\mathbf{M}-1}p_{ii})^2} \tag{6}$$

for the c,h described, where the expressions for the errors are those presented earlier. Finally, we use the properties of the suggested match between c and h to bound the ratio. 1 Let c,h denote any two original space hypotheses such that $\bar{c}=U(c),\bar{h}=U(h)$. We wish to match the labels of h with the labels of c, using a permutation of the labels of h. If we look at the matrix P, such a permutation is a permutation of the columns, but since the order of the labels in the raws is arbitrary, we may permute the rows as well. Note that the product space error is invariant to permutations of the rows and columns, but the original space error is not: it only depends on the mass of the diagonal, and so we seek permutations which maximize this mass. We suggest an $\mathbf{M}$-step greedy procedure to construct the permuted matrix.

Specifically, denote by $f_r:\{0,..,\mathbf{M}-1\}\rightarrow\{0,..,\mathbf{M}-1\}$ and $f_c:\{0,\ldots,\mathbf{M}-1\}\rightarrow\{0,\ldots,\mathbf{M}-1\}$ the permutations of the rows and the columns respectively. In step $0\leq k\leq\mathbf{M}-1$ we extend the definition of both permutations by finding the row and column to be mapped to row and column k. In the first step we find the largest element p_{ij} of the matrix P, and define $f_r(i)=0,f_c(j)=0$. In the k-th step we find the largest element of the sub matrix of P with rows $\{0,\ldots,\mathbf{M}-1\}\backslash f_r^{-1}(0,\ldots,k-1)$ and columns $\{0,\ldots,\mathbf{M}-1\}\backslash f_c^{-1}(0,\ldots,k-1)$ (rows and columns not already 'used'). Denoting this element as p_{ij}, we then define $f_r(i)=k$ and $f_c(j)=k$.

Without loss of generality, let P denote the joint distribution matrix after applying the permutations thus defined to the original P's rows and columns. By construction, P now has the property:

$$\forall 0\leq i\leq\mathbf{M}-1,\quad\forall j,k\geq i,\qquad p_{ii}\geq p_{jk} \tag{7}$$

In order to bound the ratio (6), we bound the nominator from below as follows

$$\begin{aligned}
e(\bar{h},\bar{c}) &= \sum_{j=0}^{\mathbf{M}-1}\sum_{i=0}^{\mathbf{M}-1}p_{ij}\sum_{\substack{k=0\\k\neq i}}^{\mathbf{M}-1}p_{kj}+\sum_{i=0}^{\mathbf{M}-1}\sum_{j=0}^{\mathbf{M}-1}p_{ij}\sum_{\substack{k=0\\k\neq j}}^{\mathbf{M}-1}p_{ik}\\
&\geq \sum_{j=0}^{\mathbf{M}-1}[\sum_{i\geq j}^{\mathbf{M}-1}p_{ij}\sum_{\substack{k\geq j\\k\neq i}}^{\mathbf{M}-1}p_{kj}]+\sum_{i=0}^{\mathbf{M}-1}[\sum_{j\geq i}^{\mathbf{M}-1}p_{ij}\sum_{\substack{k\geq i\\k\neq j}}^{\mathbf{M}-1}p_{ik}]\\
&= \sum_{j=0}^{\mathbf{M}-1}[\sum_{i\geq j}^{\mathbf{M}-1}p_{ij}[\sum_{k\geq j}^{\mathbf{M}-1}p_{kj}-p_{ij}]]+\sum_{i=0}^{\mathbf{M}-1}[\sum_{j\geq i}^{\mathbf{M}-1}p_{ij}[\sum_{k\geq i}^{\mathbf{M}-1}p_{ik}-p_{ij}]]
\end{aligned}$$

and then use constraint (7) to improve the bound

$$\geq \sum_{j=0}^{\mathbf{M}-1}[\sum_{i\geq j}^{\mathbf{M}-1} p_{ij}[\sum_{k\geq j}^{\mathbf{M}-1} p_{kj} - p_{jj}]] + \sum_{i=0}^{\mathbf{M}-1}[\sum_{j\geq i}^{\mathbf{M}-1} p_{ij}[\sum_{k\geq i}^{\mathbf{M}-1} p_{ik} - p_{ii}]]$$

The denominator in (6) is the square of $e(c,h)$, which can be written as

$$e(h,c) = 1 - \sum_{k=0}^{\mathbf{M}-1} p_{ii} = \sum_{k=0}^{\mathbf{M}-1}[\sum_{i>k} p_{ik} + \sum_{j>k} p_{kj}]$$

To simplify the notations, denote

$$\mathbf{M}_k^v = \sum_{j>k} p_{jk} \qquad 0 \leq k \leq \mathbf{M}-1$$

$$\mathbf{M}_k^h = \sum_{i>k} p_{ki} \qquad 0 \leq k \leq \mathbf{M}-1$$

Changing variables from $\{p_{ij}\}_{i,j=0}^{\mathbf{M}-1}$ to $\{\mathbf{M}_k^v, \mathbf{M}_k^h, p_{kk}\}_{k=0}^{\mathbf{M}-1}$, ratio (6) becomes

$$\frac{e(\bar{h},\bar{c})}{e(h,c)^2} = \frac{\sum_{k=0}^{\mathbf{M}-1}(\mathbf{M}_k^v + p_{kk})\mathbf{M}_k^v + (\mathbf{M}_k^h + p_{kk})\mathbf{M}_k^h}{(\sum_{k=0}^{\mathbf{M}-1}\mathbf{M}_k^v + \mathbf{M}_k^h)^2}$$

Now we use the inequality $\sum_{i=1}^{N} a_i^2 \geq \frac{1}{N}(\sum_{i=1}^{N} a_i)^2$ (for positive arguments) twice to get the required bound:

$$\frac{\sum_{k=0}^{\mathbf{M}-1}(\mathbf{M}_k^v + p_{kk})\mathbf{M}_k^v + (\mathbf{M}_k^h + p_{kk})\mathbf{M}_k^h}{(\sum_{k=0}^{\mathbf{M}-1}\mathbf{M}_k^v + \mathbf{M}_k^h)^2} \geq \frac{\sum_{k=0}^{\mathbf{M}-1}(\mathbf{M}_k^v)^2 + (\mathbf{M}_k^h)^2}{(\sum_{k=0}^{\mathbf{M}-1}\mathbf{M}_k^v + \mathbf{M}_k^h)^2} \geq \frac{\sum_{k=0}^{\mathbf{M}-1}\frac{1}{2}(\mathbf{M}_k^v + \mathbf{M}_k^h)^2}{(\sum_{k=0}^{\mathbf{M}-1}\mathbf{M}_k^v + \mathbf{M}_k^h)^2}$$

$$\geq \frac{\frac{1}{2\mathbf{M}}(\sum_{k=0}^{\mathbf{M}-1}\mathbf{M}_k^v + \mathbf{M}_k^h)^2}{(\sum_{k=0}^{\mathbf{M}-1}\mathbf{M}_k^v + \mathbf{M}_k^h)^2} = \frac{1}{2\mathbf{M}}$$

B Completion of the proof of Thm. 5

we have observed that

$$2^{d_{pr}} \leq (\frac{2d_{pr}e(\mathbf{M}+1)^2}{2d_o})^{d_o}$$

Following the proof of Thm. 10 in [4], let us write

$$d_{pr} \ln 2 \leq d_o[\ln \frac{d_{pr}}{d_o} + \ln(e(\mathbf{M}+1)^2)]$$

Using the inequality $\ln(x) \leq xy - \ln(ey)$ which is true for all $x, y \geq 0$, we get

$$\begin{aligned} d_{pr} \ln 2 &\leq d_o[\frac{d_{pr}}{d_o} y - \ln ey + \ln e(\mathbf{M}+1)^2] \\ &\leq d_{pr} y + d_o \ln \frac{(\mathbf{M}+1)^2}{y} \\ &\leq \frac{d_o}{ln(2)-y} ln \frac{(\mathbf{M}+1)^2}{y} = \frac{2d_o ln(\mathbf{M}+1) - d_o lny}{ln(2)-y} \\ &= \frac{2ln2 d_o log_2(\mathbf{M}+1) - d_o lny}{ln(2)-y} \end{aligned}$$

If we limit ourselves to $y < 1$ then $(-d_o \ln y) \geq 0$, and therefore we can multiply this expression by $\log_2(\mathbf{M}+1) > 1$ and keep the inequality. Hence

$$d_{pr} \leq \frac{(2\ln 2 - \ln y)}{\ln 2 - y} d_o \log_2(\mathbf{M}+1)$$

Finally we choose $y = 0.34$ to get the bound

$$d_{pr} \leq 4.87 d_o \log_2(\mathbf{M}+1)$$

C Completion of the proof of Thm. 6

Lemma 3. *Each cluster $g^{-1}(i)$ $i = 0, \ldots, \mathbf{M}_g$ intersects at most one of the sets $\{c^{-1}(j) \cap \mathcal{G}\}_{j=0}^{\mathbf{M}-1}$.*

Proof. According to Lemma 2, $c^{-1}(j_1) \cap \mathcal{G}$ and $c^{-1}(j_2) \cap \mathcal{G}$ for $j_1 \neq j_2$ are two $\frac{K}{3}$-balls that are separated by a distance $\frac{4K}{3}$ in the $L1$ metric space. By construction $g^{-1}(i)$ is an open ball with diameter $\frac{4K}{3}$. Hence it cannot intersect more than one of the sets $\{c^{-1}(j) \cap \mathcal{G}\}_{j=0}^{\mathbf{M}-1}$. $\square$

Lemma 4. *The labeling function g as defined above has the following properties:*

1. *g defines an $\mathbf{M}$-class partition, i.e., $\mathbf{M}_g = \mathbf{M}$.*
2. *There is a bijection $J : \{0, \ldots, \mathbf{M}-1\} \rightarrow \{0, \ldots, \mathbf{M}-1\}$ matching the sets $\{g^{-1}(i)\}_{i=0}^{\mathbf{M}-1}$ and $\{c^{-1}(i) \cap \mathcal{G}\}_{i=0}^{\mathbf{M}-1}$ such that*

$$\begin{aligned} g^{-1}(i) \cap (c^{-1}(J(i)) \cap \mathcal{G}) &\neq \phi \\ g^{-1}(i) \cap (c^{-1}(l) \cap \mathcal{G}) &= \phi \; for \;\; l \neq J(i) \end{aligned}$$

3. *$|Y \backslash G_0| < \frac{3\varepsilon}{K}$.*

Proof. Assume without loss of generality that the classes are ordered according to their size, i.e.

$$|c^{-1}(j) \cap \mathcal{G}| \geq |c^{-1}(j+1) \cap \mathcal{G}|, \quad j = 0, \ldots, \mathbf{M}-2$$

We claim that

$$|g^{-1}(i)| \geq |c^{-1}(i) \cap \mathcal{G}|, \quad i = 0, \ldots, \mathbf{M}-1$$

To see this, note that since each of the sets $\{g^{-1}(j)\}_{j=0}^{i-1}$ intersects at most one of the sets $\{c^{-1}(j) \cap \mathcal{G}\}_{j=0}^{i}$, there is at least one $l \in \{1, \ldots, i\}$ such that $c^{-1}(l) \cap \mathcal{G}$ has not been touched yet in the i-th step. This set is contained in a $\frac{k}{3}$-ball, and hence it is contained in $B_{\frac{2K}{3}}(x)$ for each of its members. Therefore, our greedy procedure must choose at this step a set of the form $B_{\frac{2K}{3}}(x)$ such that

$$|B_{\frac{2K}{3}}(x)| = \max_{y \in S_{i-1}} |B_{\frac{2K}{3}}(y)| \geq |c^{-1}(l) \cap \mathcal{G}| \geq |c^{-1}(i) \cap \mathcal{G}| \tag{8}$$

Using condition (5) in the theorem, we can see that the set is bigger than the algorithm's stopping condition:

$$|c^{-1}(i) \cap \mathcal{G}| \geq |c^{-1}(i)| - |\mathcal{B}| \geq KN - \frac{3\varepsilon N}{K} > KN - \frac{KN}{2} = \frac{KN}{2} \tag{9}$$

and therefore the algorithm cannot stop just yet. Furthermore, it follows from the same condition that $\frac{KN}{2} > \frac{3\varepsilon N}{K}$, and thus the chosen set is bigger than $\mathcal{B}$. This implies that it has non empty intersection with $c^{-1}(j)$ for some $j \in \{0, \ldots, \mathbf{M}-1\}$. We have already shown that this j is unique, and so we can define the matching $J(i) = j$.

After $\mathbf{M}$ steps we are left with the set $S_{\mathbf{M}}$ with size

$$\begin{aligned} |S_{\mathbf{M}}| &= N - \sum_{i=0}^{\mathbf{M}-1} |g^{-1}(i)| \leq N - \sum_{i=0}^{\mathbf{M}-1} |c^{-1}(i) \cap \mathcal{G}| \\ &= N - |\mathcal{G}| \leq N - (N - \frac{3\varepsilon N}{K}) = \frac{3\varepsilon N}{K} < \frac{KN}{2} \end{aligned}$$

where the last inequality once again follows from condition (5). The algorithm therefore stops at this step, which completes the proof of claims (1) and (3) of the lemma.

To prove claim (2) note that since $|S_{\mathbf{M}}| \leq \frac{3\varepsilon N}{K}$, it is too small to contain a whole set of the form $c^{-1}(i) \cap \mathcal{G}$. We know from Eq. (9) that for all i $|c^{-1}(i) \cap \mathcal{G}| > \frac{KN}{2} > \frac{3\varepsilon N}{K}$. Hence, for each $j \in 0, \ldots, \mathbf{M}-1$ there is an $i \in 0, \ldots, \mathbf{M}-1$ such that

$$(c^{-1}(j) \cap \mathcal{G}) \cap g^{-1}(i) \neq \phi$$

Therefore $J : \{0, \ldots, \mathbf{M}-1\} \rightarrow \{0, \ldots, \mathbf{M}-1\}$ which was defined above is a surjection, and since both its domain and range are of size $\mathbf{M}$, it is also a bijection. $\square$

We can now complete the proof of Thm. 6:

Proof. Let $\tilde{g}$ denote the composition $J \circ g$. We will show that for $x \in G_0$, $fiber^f(x)$ of a point x on which $\tilde{g}(x)$ makes an error is far from the 'true' fiber $fiber^c(x)$ and hence x is in $\mathcal{B}$. This will prove that $e(c, \tilde{g}) < \frac{3\varepsilon}{K}$ over G_0. Since the remaining domain $Y \backslash G_0$ is known from Lemma 4 to be smaller than $\frac{3\varepsilon}{K}N$, this concludes the proof.

Assume $c(x) = i, \tilde{g}(x) = j$ and $i \neq j$ hold for a certain point $x \in G_0$. x is in $\tilde{g}^{-1}(j) \cap G_0$ which is the set chosen by the algorithm at step $i = J^{-1}(j)$. This set is known to contain a point $z \in c^{-1}(j) \cap \mathcal{G}$. On the one hand, z is in $\tilde{g}^{-1}(j)$, which is a ball of radius $\frac{2K}{3}$. Thus $d(fiber^f(x), fiber^f(z)) < \frac{4K}{3}$. On the other hand, z is in $c^{-1}(j) \cap \mathcal{G}$ and hence $d(fiber^f(z), fiber^c(z)) < \frac{K}{3}$. We can use the triangle inequality to get

$$\begin{aligned} d(fiber^f(x), fiber^c(z)) &\leq d(fiber^f(x), fiber^f(z)) + d(fiber^f(z), fiber^c(z)) \\ &< \frac{4K}{3} + \frac{K}{3} = \frac{5K}{3} \end{aligned}$$

This inequality implies that $fiber^f(x)$ is far from the 'true' fiber $fiber^c(x)$:

$$\begin{aligned} 2K &\leq d(fiber^c(x), fiber^c(z)) \\ &\leq d(fiber^c(x), fiber^f(x)) + d(fiber^f(x), fiber^c(z)) \\ &< d(fiber^c(x), fiber^f(x) + \frac{5K}{3} \\ &\Longrightarrow d(fiber^c(x), fiber^f(x) > \frac{K}{3} \end{aligned}$$

and hence $x \in \mathcal{B}$. $\square$