Near-optimal sample compression for nearest neighbors

Authored by:

Aryeh Kontorovich
Lee-Ad Gottlieb
Pinhas Nisnevitch

Abstract

We present the first sample compression algorithm for nearest neighbors with non-trivial performance guarantees. We complement these guarantees by demonstrating almost matching hardness lower bounds, which show that our bound is nearly optimal. Our result yields new insight into margin-based nearest neighbor classification in metric spaces and allows us to significantly sharpen and simplify existing bounds. Some encouraging empirical results are also presented.

1 Paper Body

The nearest neighbor classifier for non-parametric classification is perhaps the most intuitive learning algorithm. It is apparently the earliest, having been introduced by Fix and Hodges in 1951 (technical report reprinted in [1]). In this model, the learner observes a sample S of labeled points (X, Y) = (Xi , Yi)i?[n] , where Xi is a point in some metric space X and Yi ? {1, ?1} is its label. Being a metric space, X is equipped with a distance function d : X ? X ? R. Given a new unlabeled point x ? X to be classified, x is assigned the same label as its nearest neighbor in S, which is argminYi ?Y d(x, Xi). Under mild regularity assumptions, the nearest neighbor classifier?s expected error is asymptotically bounded by twice the Bayesian error, when the sample size tends to infinity [2].1 These results have inspired a vast body of research on proximity-based classification (see [4, 5] for extensive background and [6] for a recent refinement of classic results). More recently, strong margin-dependent generalization bounds were obtained in [7], where the margin is the minimum distance between opposite labeled points in S. In addition to provable generalization bounds, nearest neighbor (NN) classification enjoys several other advantages. These include simple evaluation on new data, immediate extension to multiclass labels, and minimal structural assumptions ? it does not assume a Hilbertian or even a

Banach space. However, the naive NN approach also has disadvantages. In particular, it requires storing the entire sample, which may be memory-intensive. Further, information-theoretic considerations show that exact NN evaluation requires ?(—S—) time in high-dimensional metric spaces [8] (and possibly Euclidean space as well [9]) ? a phenomenon known as the algorithmic curse of dimensionality. Lastly, the NN classifier has infinite VC-dimension [5], implying that it tends to overfit the data. 1

A Bayes-consistent modification of the 1-NN classifier was recently proposed in [3].

1

This last problem can be mitigated by taking the majority vote among k ¿ 1 nearest neighbors [10, 11, 5], or by deleting some sample points so as to attain a larger margin [12]. Shortcomings in the NN classifier led Hart [13] to pose the problem of sample compression. Indeed, significant compression of the sample has the potential to simultaneously address the issues of memory usage, NN search time, and overfitting. Hart considered the minimum Consistent Subset problem ? elsewhere called the Nearest Neighbor Condensing problem ? which seeks to identify a minimal subset S ? ? S that is consistent with S, in the sense that the nearest neighbor in S ? of every x ? S possesses the same label as x. This problem is known to be NP-hard [14, 15], and Hart provided a heuristic with runtime O(n3). The runtime was recently improved by [16] to O(n2), but neither paper gave performance guarantees. The Nearest Neighbor Condensing problem has been the subject of extensive research since its introduction [17, 18, 19]. Yet surprisingly, there are no known approximation algorithms for it ? all previous results on this problem are heuristics that lack any non-trivial approximation guarantees. Conversely, no strong hardness-of-approximation results for this problem are known, which indicates a gap in the current state of knowledge. Main results. Our contribution aims at closing the existing gap in solutions to the Nearest Neighbor Condensing problem. We present a simple near-optimal approximation algorithm for this problem, where our only structural assumption is that the points lie in some metric space. Define the scaled margin ? ¡ 1 of a sample S as the ratio of the minimum distance between opposite labeled points in S to the diameter of S. Our algorithm produces a consistent set S 0 ? S of size d1/?eddim(S)+1 (Theorem 1), where ddim(S) is the doubling dimension of the space S. This result can significantly speed up evaluation on test points, and also yields sharper and simpler generalization bounds than were previously known (Theorem 3). To establish optimality, we complement the approximation result with an almost matching hardness-of-approximation lower-bound. Using a reduction from the Label Cover problem, we show that the Nearest Neighbor Condensing problem is NP-hard to approximate within factor 1?o(1) 2(ddim(S) log(1/?)) (Theorem 2). Note that the above upper-bound is an absolute size guarantee, and stronger than an approximation guarantee. Additionally, we present a simple heuristic to be applied in conjunction with the algorithm of Theorem 1, that achieves further sample compression. The empirical performances of both our algorithm and heuristic seem encouraging (see Section 4). Related work. A well-studied problem related

to the Nearest Neighbor Condensing problem is that of extracting a small set of simple conjunctions consistent with much of the sample, introduced by [20] and shown by [21] to be equivalent to minimum Set Cover (see [22, 23] for further extensions). This problem is monotone in the sense that adding a conjunction to the solution set can only increase the sample accuracy of the solution. In contrast, in our problem the addition of a point of S to S ? can cause S ? to be inconsistent ? and this distinction is critical to the hardness of our problem. Removal of points from the sample can also yield lower dimensionality, which itself implies faster nearest neighbor evaluation and better generalization bounds. For metric spaces, [24] and [25] gave algorithms for dimensionality reduction via point removal (irrespective of margin size). The use of doubling dimension as a tool to characterize metric learning has appeared several times in the literature, initially by [26] in the context of nearest neighbor classification, and then in [27] and [28]. A series of papers by Gottlieb, Kontorovich and Krauthgamer investigate doubling spaces for classification [12], regression [29], and dimension reduction [25]. k-nearest neighbor. A natural question is whether the Nearest Neighbor Condensing problem of [13] has a direct analogue when the 1-nearest neighbor rule is replaced by a (k ¿ 1)-nearest neighbor ? that is, when the label of a point is determined by the majority vote among its k nearest neighbors. A simple argument shows that the analogy breaks down. Indeed, a minimal requirement for the condensing problem to be meaningful is that the full (uncondensed) set S is feasible, i.e. consistent with itself. Yet even for k = 3 there exist self-inconsistent sets. Take for example the set S consisting of two positive points at (0, 1) and (0, ?1) and two negative points at (1, 0) and (?1, 0). Then the 3-nearest neighbor rule misclassifies every point in S, hence S itself is inconsistent.

2

Paper outline. This paper is organized as follows. In Section 2, we present our algorithm and prove its performance bound, as well as the reduction implying its near optimality (Theorem 2). We then highlight the implications of this algorithm for learning in Section 3. In Section 4 we describe a heuristic which refines our algorithm, and present empirical results. 1.1 Preliminaries Metric spaces. A metric d on a set X is a positive symmetric function satisfying the triangle inequality d(x, y) ? d(x, z) + d(z, y); together the two comprise the metric space (X , d). The diameter of a set A ? X , is defined by diam(A) = supx,y?A d(x, y). Throughout this paper we will assume that diam(S) = 1; this can always be achieved by scaling. Doubling dimension. For a metric (X , d), let ? be the smallest value such that every ball in X of radius r (for any r) can be covered by ? balls of radius 2r . The doubling dimension of X is ddim(X) = log2 ?. A metric is doubling when its doubling dimension is bounded. Note that while a low Euclidean dimension implies a low doubling dimension (Euclidean metrics of dimension d have doubling dimension O(d) [30]), low doubling dimension is strictly more general than low Euclidean dimension. The following packing property can be demonstrated via a repetitive application of the doubling property: For set S with doubling dimension ddim(X) and diam(S) ? ?, if the minimum interpoint distance in S is at least ? ¡ ? then —S— ?

d?/?eddim(X)+1

(1)

(see, for example [8]). The above bound is tight up to constant factors, meaning there exist sets of size (?/?)?(ddim(X)) . Nearest Neighbor Condensing. Formally, we define the Nearest Neighbor Condensing (NNC) problem as follows: We are given a set S = S? ? S+ of points, and distance metric d : S ? S ? R. We must compute a minimal cardinality subset S 0 ? S with the property that for any p ? S, the nearest neighbor of p in S 0 comes from the same subset {S+ , S? } as does p. If p has multiple exact nearest neighbors in S 0 , then they must all be of the same subset. Label Cover. The Label Cover problem was first introduced by [31] in a seminal paper on the hardness of computation. Several formulations of this problem have appeared the literature, and we give the description forwarded by [32]: The input is a bipartite graph G = (U, V, E), with two sets of labels: A for U and B for V . For each edge (u, v) ? E (where u ? U , v ? V), we are given a relation ?u,v ? A ? B consisting of admissible label pairs for that edge. A labeling (f, g) is a pair of functions f : U ? 2A and g : V ? 2B {?} assigning a set of labels to each vertex. A labeling covers an edge (u, v) if for every label b ? g(v) there is some label a ? f (u) such that (a, b) ? ?u,v . The goal is to find a labeling that covers Pall edges, and which minimizes the sum of the number of labels assigned to each u ? U , that is u?U —f (u)—. It was shown in [32] that it is 1?o(1) NP-hard to approximate Label Cover to within a factor 2(log n) , where n is the total size of the input. Learning. We work in the agnostic learning model [33, 5]. The learner receives n labeled examples (Xi , Yi) ? X ?{?1, 1} drawn iid according to some unknown probability distribution P. Associated P to any hypothesis h : X ? {?1, 1} is its empirical error err(h) c = n?1 i?[n] 1{h(Xi)6=Yi } and generalization error err(h) = P(h(X) 6= Y).

2 Near-optimal approximation algorithm In this section, we describe a simple approximation algorithm for the Nearest Neighbor Condensing problem. In Section 2.1 we provide almost tight hardness-of-approximation bounds. We have the following theorem: Theorem 1. Given a point set S and its scaled margin ? ¡ 1, there exists an algorithm that in time min{n2 , 2O(ddim(S)) n log(1/?)} computes a consistent set S 0 ? S of size at most d1/?eddim(S)+1 . Recall that an ?-net of point set S is a subset S? ? S with two properties: 3

(i) Packing. The minimum interpoint distance in S? is at least ?. (ii) Covering. Every point p ? S has a nearest neighbor in S? strictly within distance ?. We make the following observation: Since the margin of the point set is ?, a ?-net of S is consistent with S. That is, every point p ? S has a neighbor in S? strictly within distance ?, and since the margin of S is ?, this neighbor must be of the same label set as p. By the packing property of doubling spaces (Equation 1), the size of S? is at most d1/?eddim(S)+1 . The solution returned by our algorithm is S? , and satisfies the guarantees claimed in Theorem 1. It remains only to compute the net S? . A brute-force greedy algorithm can accomplish this in time O(n2): For every point p ? S, we add p to S? if the distance from p to all points currently in S? is ? or greater, d(p, S?) ? ?. See Algorithm 1. Algorithm 1 Brute-force net construction Require: S 1: S? ? arbitrary point of

S 2: for all p ? S do 3: if d(p, S?) ? ? then 4: S? = S? ? {p} 5: end if 6: end for The construction time can be improved by building a net hierarchy, similar to the one employed by [8], in total time 2O(ddim(S)) n log(1/?). (See also [34, 35, 36].) A hierarchy consists of all nets S2i for i = 0, ?1, . . . , blog ?c, where S2i ? S2i?1 for all i ¿ blog ?c. Two points p, q ? S2i are neighbors if d(p, q) ¡ 4 ? 2i . Further, each point q ? S is a child of a single nearby parent point p ? S2i satisfying d(p, q) ¡ 2i . By the definition of a net, a parent point must exist. If two points p, q ? S2i are neighbors (d(p, q) ¡ 4 ? 2i) then their respective parents p0 , q 0 ? S2i+1 are necessarily neighbors as well: d(p0 , q 0) ? d(p0 , p) + d(p, q) + d(q, q 0) ¡ 2i+1 + 4 ? 2i + 2i+1 = 4 ? 2i+1 . The net S20 = S1 consists of a single arbitrary point. Having constructed S2i , it is an easy matter to construct S2i?1 : Since we require S2i?1 ? S2i , we will initialize S2i?1 = S2i . For each q ? S, we need only to determine whether d(q, S2i?1) ? 2i?1 , and if so add q to S2i?1 . Crucially, we need not compare q to all points of S2i?1 : If there exists a point p ? S2i with d(q, p) ¡ 2i , then the respective parents p0 , q 0 ? S2i of p, q must be neighbors. Let set T include only the children of q 0 and of q 0 ?s neighbors. To determine the inclusion of every q ? S in S2i?1 , it suffices to compute whether d(q, T) ? 2i?1 , and so n such queries are sufficient to construct S2i?1 . The points of T have minimum distance 2i?1 and are all contained in a ball of radius 4 ? 2i + 2i?1 centered at T , so by the packing property (Equation 1) —T — = 2O(ddim(S)) . It follows that the above query d(q, T) can be answered in time 2O(ddim(S)) . For each point in S we execute O(log(1/?)) queries, for a total runtime of 2O(ddim(S)) n log(1/?). The above procedure is illustrated in the Appendix. 2.1 Hardness of approximation of NNC In this section, we prove almost matching hardness results for the NNC problem. Theorem 2. Given a set S of labeled points with scaled margin ?, it is NP-hard to approximate the solution to the Nearest Neighbor Condensing problem on S to within a factor 1?o(1) 2(ddim(S) log(1/?)) . To simplify the proof, we introduce an easier version of NNC called Weighted Nearest Neighbor Condensing (WNNC). In this problem, the input is augmented with a function assigning weight to each point of S, and the goal is to find a subset S 0 ? S of minimum total weight. We will reduce Label Cover to WNNC and then reduce WNNC to NNC (with some mild assumptions on the admissible range of weights), all while preserving hardness of approximation. The theorem will follow from the hardness of Label Cover [32]. P First reduction. Given a Label Cover instance of size m = —U —+—V —+—A—+—B—+—E—+ e?E —?E —, fix large value c to be specified later, and an infinitesimally small constant ?. We create an instance of WNNC as follows (see Figure 1). 1. We first create a point p+ ? S+ of weight 1. 4

Label Cover
U u1
u2
Nearest Neighbor Condensing
V e1 e2 e3
SU,A ? S+
SL ? S+

SV,B ? S-
SE ? S-
v1
v2
l1: (a1,b1) ? ?e1 l2: (a2,b2) ? ?e1
u1a1
l1
v1b1
u1a2
l2
v1b2
u2a1
l3,,l4
v2b1
u2a2
2
l5
2+2{
l3: (a1,b1) ? ?e2 l4: (a2,b1) ? ?e2
?[+
2+{
v2b2
2
p-
e1
e2 e3 3
3+{
p+
l5: (a1,b2) ? ?e3

Figure 1: Reduction from Label Cover to Nearest Neighbor Condensing. We introduce set SE ? S? representing edges in E: For each edge e ? E, create point pe of weight ?. The distance from pe to p+ is 3 + ?. 2. We introduce set SV,B ? S? representing pairs in V ? B: For each vertex v ? V and label b ? B, create point pv,b of weight 1. If edge e is incident to v and there exists a label (a, b) ? ?e for any a ? A, then the distance from pv,b to pe is 3. Further add a point p? ? S? of weight 1, at distance 2 from all points in SV,B . 3. We introduce set SL ? S+ representing labels in ?e . For each edge e = (u, v) and label b ? B for which (a, b) ? ?e (for any a ? A), we create point pe,b ? SL of weight ?. pe,b represents the set of labels (a, b) ? ?e over all a ? A. pe,b is at distance 2 + ? from pv,b . Further add a point p0+ ? S+ of weight 1, at distance 2 + 2? from all points in SL . 4. We introduce set SU,A ? S+ representing pairs in U ? A: For each vertex u ? U and label a ? A, create point pu,a of weight c. For any edge e = (u, v) and label b ? B, if (a, b) ? ?e then the distance from pe,b ? SL to pu,a is 2. The points of each set SE , SV,B , SL and SU,A are packed into respective balls of diameter 1. Fixing any target doubling dimension D = ?(1) and recalling that the cardinality of each of these sets is less than m2 ,

we conclude that the minimum interpoint distance in each ball is m?O(1/D) . All interpoint distances not yet specified are set to their maximum possible value. The diameter of the resulting set is constant, so its scaled margin is ? = m?O(1/D) . We claim that a solution of WNNC on the constructed instance implies some solution of the Label Cover Instance: 1. p+ must appear in any solution: The nearest neighbors of p+ are the negative points of SE , so if p+ is not included the nearest neighbor of set SE is necessarily the nearest neighbor of p+ , which is not consistent. 2. Points in SE have infinite weight, so no points of SE appear in the solution. All points of SE are at distance exactly 3 + ? from p+ , hence each point of SE must be covered by some point of SV,B to which it is connected ? other points in SV,B are farther than 3 + ?. (Note that SV,B itself can be covered by including the single point p? .) Choosing covering points in SV,B corresponds to assigning labels in B to vertices of V in the Label Cover instance. 3. Points in SL have infinite weight, so no points of SL appear in the solution. Hence, either p0+ or some points of SU,A must be used to cover points of SL . Specifically, a point in SL ? S+ incident on an included point of SV,B ? S? is at distance exactly 2 + ? from this point, and so it must be covered by some point of SU,A to which it is connected, at distance 2 ? other points in SU,A are farther than 2 + ?. Points of SL not incident on an included point of SV,B can be covered by p0+ , which at distance 2 + 2? is still closer than any point in SV,B . (Note that SU,A itself can be covered by including a single arbitrary point of SU,A , which at distance 1 is closer than all other point sets.) Choosing the covering point in SU,A corresponds to assigning labels in A to vertices of U in the Label Cover instance, thereby inducing a valid labeling for some edge and solving the Label Cover problem. 5

Now, a trivial solution to this instance of WNNC is to take all points of SU,A , SV,B and the single point p+ : then SE and p? are covered by SV,B , and SL and p0+ by SU,A . The size of the resulting set is c—SU,A — + —SU,B — + 1, and this provides an upper bound on the optimal solution. By setting c = m4 m3 ¿ m(—SU,B — + 1), we ensure that the solution cost of WNNC is asymptotically equal to the number of points of SU,A included in its solution. This in turn is exactly the sum of labels of A assigned to each vertex of U in a solution to the Label Cover problem. Label Cover is hard 1?o(1) to approximate within a factor 2(log m) , implying that WNNC is hard to approximate within a 1?o(1) 1?o(1) factor of 2(log m) = 2(D log(1/?)) . Before proceeding to the next reduction, we note that to rule out the inclusion of points of SE , SL in the solution set, infinite weight is not necessary: It suffices to give each heavy point weight c2 , which is itself greater than the weight of the optimal solution by a factor of at least m2 . Hence, we may assume all weights are restricted to the range [1, mO(1)], and the hardness result for WNNC still holds. Second reduction. We now reduce WNNC to NNC, assuming that the weights of the n points are in the range [1, mO(1)]. Let ? be the scaled margin of the WNNC instance. To mimic the weight assignment of WNNC using the unweighted points of NNC, we introduce the following gadget graph G(w, D): Given parameter w and doubling dimension D, create a point set T of size w whose interpoint distances are the same as those realized by a set of contiguous

points on the D-dimensional ‘1 -grid of side-length dw1/D e. Now replace each point p ? T by twin positive and negative points at mutual distance ?2 , so that the distance from each twin replacing p to each twin replacing any q ? T is the same as the distance from p to q. G(w, D) consists of T , as well as a single positive point at distance dw1/D e from all positive points of T , and dw1/D e + ?2 from all negative points of T , and a single negative point at distance dw1/D e from all negative points of T , and dw1/D e + ?2 from all positive points of T . Clearly, the optimal solution to NNC on the gadget instance is to choose the two points not in T . Further, if any single point in T is included in the solution, then all of T must be included in the solution: First the twin of the included point must also be included in the solution. Then, any point at distance 1 from both twins must be included as well, along with its own twin. But then all points within distance 1 of the new twins must be included, etc., until all points of T are found in the solution. To effectively assign weight to a positive point of NNC, we add a gadget to the point set, and place all negative points of the gadget at distance dw1/D e from this point. If the point is not included in the NNC solution, then the cost of the gadget is only 2.2 But if this point is included in the NNC solution, then it is the nearest neighbor of the negative gadget points, and so all the gadget points must be included in the solution, incurring a cost of w. A similar argument allows us to assign weight to negative points of NNC. The scaled margin of the NNC instance is of size ?(?/w1/D) = ?(?m?O(1/D)), which completes the proof of Theorem 2.

3 Learning In this section, we apply Theorem 1 to obtain improved generalization bounds for binary classification in doubling spaces. Working in the standard agnostic PAC setting, we take the labeled sample S to be drawn iid from some unknown distribution over X ? {?1, 1}, with respect to which all of our probabilities will be defined. In a slight abuse of notation, we will blur the distinction between S ? X as a collection of points in a metric space and S ? (X ? {?1, 1})n as a sequence of pointlabel pairs. As mentioned in the preliminaries, there is no loss of generality in taking diam(S) = 1. Partitioning the sample S = S+ ? S? into its positively and negatively labeled subsets, the margin induced by the sample is given by ?(S) = d(S+ , S?), where d(A, B) := minx?A,x0 ?B d(x, x0) for A, B ? X . Any labeled sample S induces the nearest-neighbor classifier ?S : X ? {?1, 1} via

+1 if d(x, S+) ¡ d(x, S?) ?S (x) = ?1 else. 2 By scaling up all weights by a factor of n2 , we can ensure that the cost of all added gadgets (2n) is asymptotically negligible.

6

P We say that S? ? S is ?-consistent with S if n1 x?S 1{?S (x)6=? ? (x)} ? ?. For ? = 0, an ?-consistent S S? is simply said to be consistent (which matches our previous notion of consistent subsets). A ? if there is an ?-consistent S? ? S with sample S is said to be (?, ?)-separable (with witness S) ? ?(S) ? ?. We begin by invoking a standard Occam-type argument to show that the existence of small ?consistent sets implies good generalization. The generalizing power of sample compression was independently discovered by [37, 38], and later elaborated upon by [39]. Theorem 3. For any distribution P, any n ? N and

any 0 ¡ ? ¡ 1, with probability at least 1 ? ? over the random sample S ? (X ? {?1, 1})n , the following holds:

1 ? log n + log n + log 1 . (i) If S? ? S is consistent with S, then err(?S?) ? —S— ? ? n ? —S— s ? log n + 2 log n + log 1 —S— ?n ? (ii) If S? ? S is ?-consistent with S, then err(?S?) ? + . ? ? n ? —S— 2(n ? —S—) Proof. Finding a consistent (resp., ?-consistent) S? ? S constitutes a sample compression scheme of ? as stipulated in [39]. Hence, the bounds in (i) and (ii) follow immediately from Theorems size —S—, 1 and 2 ibid. Corollary 1. With probability at least 1 ? ?, the following holds: If S is (?, ?)-separable with ? then witness S, s ‘ log n + 2 log n + log 1? ?n err(?S?) ? + , n¿ 2(n ? ‘) where ‘ = d1/?eddim(S)+1 . Proof. Follows immediately from Theorems 1 and 3(ii). Remark. It is instructive to compare the bound above to [12, Corollary 5]. Stated in the language of this paper, the latter upper-bounds the NN generalization error in terms of the sample margin ? and ddim(X) by r 2 ?+ (d? ln(34en/d?) log2 (578n) + ln(4/?)), (2) n ddim(X)+1

where d? = d16/?e and ? is the fraction of the points in S that violate the margin condition (i.e., opposite-labeled point pairs less than ? apart in d). Hence, Corollary 1 is a considerable improvement over (2) in at least three aspects. First, the data-dependent ddim(S) may be significantly smaller than the dimension of the ambient space, ddim(X).3 Secondly, the factor of 16ddim(X)+1 is shaved off. Finally, (2) relied on some fairly intricate fat-shattering arguments [40, 41], while Corollary 1 is an almost immediate consequence of much simpler Occam-type results. One limitation of Theorem 1 is that it requires the sample to be (0, ?)-separable. The form of the bound in Corollary 1 suggests a natural Structural Risk Minimization (SRM) procedure: minimize the right-hand size over (?, ?). A solution to this problem was (essentially) given in [12, Theorem 7]: Theorem 4. Let R(?, ?) denote the right-hand size of the inequality in Corollary 1 and put (?? , ? ?) = argmin?,? R(?, ?). Then (i) One may compute (?? , ? ?) in O(n4.376) randomized time. (ii) One may compute (? ?, ??) satisfying R(? ?, ??) ? 4R(?? , ? ?) in O(ddim(S)n2 log n) deterministic time. Both solutions yield a witness S? ? S of (?, ?)-separability as a by-product. ? we may Having thus computed the optimal (or near-optimal) ??, ?? with the corresponding witness S, ? now run the algorithm furnished by Theorem 1 on the sub-sample S and invoke the generalization bound in Corollary 1. The latter holds uniformly over all ??, ??. 3

In general, ddim(S) ? c ddim(X) for some universal constant c, as shown in [24].

7

4 Experiments In this section we discuss experimental results. First, we will describe a simple heuristic built upon our algorithm. The theoretical guarantees in Theorem 1 feature a dependence on the scaled margin ?, and our heuristic aims to give an improved solution in the problematic case where ? is small. Consider the following procedure for obtaining a smaller consistent set. We first extract a net S? satisfying the guarantees of Theorem 1. We then remove points from S? using the following rule: for all i ? {0, . . . dlog ?e}, and for each p ? S? , if the distance from p to all opposite labeled points in S? is at least 2 ? 2i , then

remove from S? all points strictly within distance 2i ? ? of p (see Algorithm 2). We can show that the resulting set is consistent: Lemma 5. The above heuristic produces a consistent solution. Proof. Consider a point p ? S? , and assume without loss of generality that p is positive. If d(p, S??) ? 2 ? 2i , then the positive net-points strictly within distance 2i of p are closer to p than to any negative point in S? , and are ?covered? by p. The removed positive net-points at distance 2i ? ? themselves cover other positive points of S within distance ?, but p covers these points of S as well. Further, p cannot be removed at a later stage in the algorithm, since p?s distance from all remaining points is at least 2i ? ?. Algorithm 2 Consistent pruning heuristic 1: S? is produced by Algorithm 1 or its fast version (Appendix) 2: for all i ? {0, . . . , dlog ?e} do 3: for all p ? S? do 4: if p ? S?? and d(p, S??) ? 2 ? 2i then 5: for all q 6= p ? S? with d(p, q) ¡ 2i ? ? do 6: S? ? S? {q} 7: end for 8: end if 9: end for 10: end for As a proof of concept, we tested our sample compression algorithms on several data sets from the UCI Machine Learning Repository. These included the Skin Segmentation, Statlog Shuttle, and Covertype sets.4 The final dataset features 7 different label types, which we treated as 21 separate binary classification problems; we report results for labels 1 vs. 4, 4 vs. 6, and 4 vs. 7, and these typify the remaining pairs. We stress that the focus of our experiments is to demonstrate that (i) a significant amount of consistent sample compression is often possible and (ii) the compression does not adversely affect the generalization error. For each data set and experiment, we sampled equal sized learning and test sets, with equal representation of each label type. The L1 metric was used for all data sets. We report (i) the initial sample set size, (ii) the percentage of points retained after the net extraction procedure of Algorithm 1, (iii) the percentage retained after the pruning heuristic of Algorithm 2, and (iv) the change in prediction accuracy on test data, when comparing the heuristic to the uncompressed sample. The results, averaged over 500 trials, are summarized in Figure 2. data set Skin Segmentation Statlog Shuttle Covertype 1 vs. 4 Covertype 4 vs. 6 Covertype 4 vs. 7

original sample 10000 2000 2000 2000 2000

% after net 35.10 65.75 35.85 96.50 4.40

% after heuristic 4.78 29.65 17.70 69.00 3.40

?% accuracy -0.0010 +0.0080 +0.0200 -0.0300 0.0000

Figure 2: Summary of the performance of NN sample compression algorithms. 4 http://tinyurl.com/skin-data; http://tinyurl.com/cover-data

http://tinyurl.com/shuttle-data;

8

2 References

[1] E. Fix and J. L. Hodges, Discriminatory analysis. nonparametric discrimination: Consistency properties. International Statistical Review / Revue Internationale de Statistique, 57(3):pp. 238?247, 1989. [2] T. Cover, P. Hart. Nearest neighbor pattern classification. IEEE Trans. Info. Theo., 13:21?27, 1967. [3] A.

Kontorovich, R. Weiss. A Bayes consistent 1-NN classifier (arXiv:1407.0208), 2014. [4] G. Toussaint. Open problems in geometric methods for instance-based learning. In Discrete and computational geometry, volume 2866 of Lecture Notes in Comput. Sci., pp 273?283. 2003. [5] S. Shalev-Shwartz, S. Ben-David. Understanding Machine Learning. 2014. [6] K. Chaudhuri, S. Dasgupta. Rates of Convergence for Nearest Neighbor Classification. In NIPS, 2014. [7] U. von Luxburg, O. Bousquet. Distance-based classification with Lipschitz functions. JMLR, 2004. [8] R. Krauthgamer and J. R. Lee. Navigating nets: Simple algorithms for proximity search. In SODA, 2004. [9] K. L. Clarkson. An algorithm for approximate closest-point queries. In SCG, 1994 [10] L. Devroye, L. Gy?orfi, A. Krzy?zak, G. Lugosi. On the strong universal consistency of nearest neighbor regression function estimates. Ann. Statist., 22(3):1371?1385, 1994. [11] R. R. Snapp and S. S. Venkatesh. Asymptotic expansions of the k nearest neighbor risk. Ann. Statist., 26(3):850?878, 1998. [12] L. Gottlieb, A. Kontorovich, R. Krauthgamer. Efficient classification for metric data. In COLT, 2010. [13] P. E. Hart. The condensed nearest neighbor rule. IEEE Trans. Info. Theo., 14(3):515?516, 1968. [14] G. Wilfong. Nearest neighbor problems. In SCG, 1991. [15] A. V. Zukhba. NP-completeness of the problem of prototype selection in the nearest neighbor method. Pattern Recognit. Image Anal., 20(4):484?494, 2010. [16] F. Angiulli. Fast condensed nearest neighbor rule. In ICML, 2005. [17] W. Gates. The reduced nearest neighbor rule. IEEE Trans. Info. Theo., 18:431?433, 1972. [18] G. L. Ritter, H. B. Woodruff, S. R. Lowry, T. L. Isenhour. An algorithm for a selective nearest neighbor decision rule. IEEE Trans. Info. Theo., 21:665?669, 1975. [19] D. R. Wilson and T. R. Martinez. Reduction techniques for instance-based learning algorithms. Mach. Learn., 38:257?286, 2000. [20] L. G. Valiant. A theory of the learnable. Commun. ACM, 27(11):1134?1142, 1984. [21] D. Haussler. Quantifying inductive bias: AI learning algorithms and valiant?s learning framework. Artificial Intelligence, 36(2):177 ? 221, 1988. [22] F. Laviolette, M. Marchand, M. Shah, S. Shanian. Learning the set covering machine by bound minimization and margin-sparsity trade-off. Mach. Learn., 78(1-2):175?201, 2010. [23] M. Marchand and J. Shawe-Taylor. The set covering machine. JMLR, 3:723?746, 2002. [24] L. Gottlieb and R. Krauthgamer. Proximity algorithms for nearly doubling spaces. SIAM J. on Discr. Math., 27(4):1759?1769, 2013. [25] L. Gottlieb, A. Kontorovich, R. Krauthgamer. Adaptive metric dimensionality reduction. ALT, 2013. [26] A. Beygelzimer, S. Kakade, J. Langford. Cover trees for nearest neighbor. In ICML, 2006. [27] Y. Li and P. M. Long. Learnability and the doubling dimension. In NIPS, 2006. [28] N. H. Bshouty, Y. Li, P. M. Long. Using the doubling dimension to analyze the generalization of learning algorithms. J. Comp. Sys. Sci., 75(6):323 ? 335, 2009. [29] L. Gottlieb, A. Kontorovich, R. Krauthgamer. Efficient regression in metric spaces via approximate Lipschitz extension. In SIMBAD, 2013. [30] A. Gupta, R. Krauthgamer, J. R. Lee. Bounded geometries, fractals, and low-distortion embeddings. In FOCS, 2003. [31] S. Arora, L. Babai, J. Stern, Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. In FOCS, 1993. [32] I. Dinur, S. Safra. On the hardness of approximating label-

cover. Info. Proc. Lett., 2004. [33] M. Mohri, A. Rostamizadeh, A. Talwalkar. Foundations Of Machine Learning. 2012. [34] A. Beygelzimer, S. Kakade, J. Langford. Cover trees for nearest neighbor. In ICML 2006. [35] S. Har-Peled and M. Mendel. Fast construction of nets in low-dimensional metrics and their applications. SIAM J. on Comput., 35(5):1148?1184, 2006. [36] R. Cole, L. Gottlieb. Searching dynamic point sets in spaces with bounded doubling dimension. STOC, 2006. [37] N. Littlestone and M. K. Warmuth. Relating data compression and learnability, unpublished. 1986. [38] L. Devroye, L. Gy?orfi, G. Lugosi. A probabilistic theory of pattern recognition, 1996. [39] T. Graepel, R. Herbrich, J. Shawe-Taylor. Pac-bayesian compression bounds on the prediction error of learning algorithms for classification. Mach. Learn., 59(1-2):55?76, 2005. [40] N. Alon, S. Ben-David, N. Cesa-Bianchi, D. Haussler. Scale-sensitive dimensions, uniform convergence, and learnability. J. ACM, 44(4):615?631, 1997. [41] P. Bartlett and J. Shawe-Taylor. Generalization performance of support vector machines and other pattern classifiers, pages 43?54. 1999.

9