# Weighted Distance Nearest Neighbor Condensing

**Lee-Ad Gottlieb** [1] **Timor Sharabi** [1] **Roi Weiss** [1]

## Abstract

The problem of nearest neighbor condensing has enjoyed a long history of study, both in its theoretical and practical aspects. In this paper, we introduce the problem of weighted distance nearest neighbor condensing, where one assigns weights to each point of the condensed set, and then new points are labeled based on their weighted distance nearest neighbor in the condensed set. We study the theoretical properties of this new model, and show that it can produce dramatically better condensing than the standard nearest neighbor rule, yet is characterized by generalization bounds almost identical to the latter. We then suggest a condensing heuristic for our new problem. We demonstrate Bayes consistency for this heuristic, and also show promising empirical results.

## 1. Introduction

The nearest neighbor (NN) classifier, introduced by (Fix & Hodges, 1951), is an intuitive and popular learning tool. In this model, a learner observes a sample of labeled points, and given a new point to be classified, assigns the new point the same label as its nearest neighbor in the sample. It was subsequently demonstrated by Cover & Hart (1967) that when no label noise is present, the nearest neighbor classifier's expected error converges to zero as the sample size grows. This and other results helped spawn a deep body of research into proximity-based classification (Devroye et al., 1996; Shalev-Shwartz & Ben-David, 2014).

Nearest neighbor classifiers enjoy other advantages as well. They require only a distance function on the points, and do not even require that the host space be metric. They also extend naturally to the multi-class setting. Yet they are not without their disadvantages: For example, a naive nearest neighbor approach may require storing the entire sample. To address the disadvantages of the NN classifier, (Hart, 1968) introduced the technique of sample compression for the NN classifier. This work defined the minimum consistent subset problem (also called the nearest neighbor condensing problem): Given a sample, find a minimal subset of the sample that is *consistent* with it, meaning that for every point of the sample, its nearest neighbor in the subset (that is the condensed set) has the same label. (Hart, 1968) further suggested a heuristic for the NN condensing problem. Analysis of this problem, as well as the creation of new heuristics for it, has been the subject of extensive research since its introduction (Gates, 1972; Ritter et al., 1975; Devroye et al., 1996; Wilson & Martinez, 2000).

**Our Contribution** In this paper, we introduce a novel modification of the NN condensing problem. In our new condensing problem (presented formally in Section 2), each point of the condensed set is also assigned a weight. We modify the distance function to consider *weighted distance*, meaning that the distance from a new point to a point in the condensed set is their original distance divided by the weight of the point in the condensed set. It follows that assigning high weight to a point in the condensed set increases its influence on the labeling of new points. We call the new problem of choosing a condensed set and assigning its weights the *weighted distance nearest neighbor condensing problem*.

We proceed with an in-depth study of the statistical properties of weighted distance condensing (Section 3). Crucially, we find that our model allows dramatically better condensing than what is possible under standard (that is, unweighted) NN condensing: There are families of instances wherein the optimal condensing under unweighted NN rule is of size $\Theta(n)$, while condensing under the weighted NN (WNN) rule yields a condensed set of size exactly 2 (Theorem 3.1). At the same time, we demonstrate generalization bounds for condensing under the WNN rule which are almost identical to those previously known for the NN rule (Theorem 3.4). This means that the much more powerful weighted rule can be adopted with only negligible increase in the variance of the model, so that the more powerful rule does not contribute to overfitting.

[1] Department of Computer Science, Ariel University, Ariel, Israel. Correspondence to: Lee-Ad Gottlieb <leead@ariel.ac.il>, Timor Sharabi <timorsharabi@gmail.com>, Roi Weiss <roiw@ariel.ac.il>.

*Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

Having established these statistical properties of WNN condensing, we suggest a greedy-based heuristic for this problem, that is the identification of a condensed set and the assignment of weights to members of this set (Section 4). We further demonstrate that the suggested heuristic is a member of a broad set of classifiers which are Bayes consistent, thereby lending statistical support to its use.

After deciding on a heuristic for our condensing problem, we compare its empirical condensing abilities to those of popular heuristics for the unweighted NN problem (Section 5). We find that that the condensing bounds of our heuristic compare favorably to the others, which additionally suggests that further research on heuristics for WNN condensing is a promising direction.

### 1.1. Related Work

The nearest neighbor condensing problem is known to be NP-hard (Wilfong, 1991; Zukhba, 2010), and (Hart, 1968) provided the first heuristic for it. Many other heuristics have been suggested since, and we mention only a few of them here: (Gates, 1972) introduced the reduced nearest neighbor (RNN) rule heuristic to iteratively contract the sample set. (Ritter et al., 1975) introduced the selective subset heuristic, which additionally guarantees that for any sample point, the distance to its nearest neighbor in the compressed set is less than the distance to any sample point with opposite label. (Barandela et al., 2005) subsequently suggested a modification to this algorithm, which they called modified selected subset (MSS). Angiulli (2005) introduced the fast nearest neighbor condensing (FCNN) heuristic, while Flores-Velazco (2020) introduced the RSS and VSS heuristics, which modify the FCNN algorithm to improve its behavior in cases where the points are too close to each other. Another popular heuristic, modified condensed nearest neighbor (MCNN), was introduced by (Devi & Murty, 2002).

No concrete algorithmic condensing bounds were known for NN condensing until the work of Gottlieb et al. (2018): They derived hardness-of-approximation results, and designed an algorithm called NET, which computes in polynomial time an approximation to the minimum subset almost matching the hardness bounds. This approach was later extended by Gottlieb & Ozeri (2019) to asymmetric distance function. Also related to this is the result of Gottlieb & Kontorovich (2022), which presented non-uniform packing, that is using balls with different radii.

As for the statistical properties of NN condensing rules, Devroye et al. (1996, Chapter 19) established Bayes consistency for an intractable rule that searches for a condensed set of fixed, data-independent size, minimizing the empirical error on the entire sample. They showed that universal Bayes consistency is achieved in finite-dimensional spaces provided the size of the condensed set is sublinear in the sample size. Hanneke et al. (2021) introduced a computationally-efficient data-dependent sample-compression NN rule, termed OptiNet, that computes a net of the samples and associates each point of the condensed set with the majority vote label in its Voronoi cell. They showed that OptiNet is universally Bayes consistency in any separable metric space. A simpler sub-sampling NN rule achieving universal Bayes consistency in separable metric spaces was demonstrated by Györfi & Weiss (2021), establishing also error rates. Xue & Kpotufe (2018) studied the error rates achieved by more complex sub-sampling NN rules. Lastly, Kerem & Weiss (2023) studied jointly-achievable error and compression rates for OptiNet under a margin condition.

Our main statistical contribution is a compression bound (Theorem 3.4). Similar finite-sample bounds (up to constants) for standard NN rules were established in Gottlieb et al. (2014); Kontorovich et al. (2017); Hanneke et al. (2021); in particular, leverage the fact that such bounds are dimension free. See also (Cohen & Kontorovich, 2022), who gave compression bounds for learning mappings between two metric spaces.

We note that our weighted approach to the NN condensing is not related to the weighted KNN model (Dudani, 1976). Weighted KNN is a classification model (not a compression optimization problem), a variant of KNN which classifies using the $k$ closest neighbors while giving additional preference to some of them. Its local use of the underlying geometry of the $k$ neighbors is unrelated to our assignment of weights to points of a condensed set in order to create a new global distance function. Our weights are chosen for condensing a realizable sample in a pre-processing stage, not for local denoising among the neighbors in the search stage. Likewise, our condensing approach places emphasis on farther points over closer ones, which is the opposite of what is done for denoising in the KNN model.

### 1.2. Preliminaries

**Metric Space.** A *metric* $d$ defined on a point set $\mathcal{X}$ is a positive symmetric function satisfying the triangle inequality, i.e. $d(x, y) \leq d(x, z) + d(z, y)$. The set $\mathcal{X}$ and metric $d$ together define the metric space $(\mathcal{X}, d)$. Let $B(x, r)$ denote the (open) ball centered at $x$ with radius $r$; a point $y \in \mathcal{X}$ is in $B(x, r)$ if $d(x, y) < r$.

**Notation.** We use $[k]$ to denote $\{1, \ldots, k\}$. We define the distance between a point $y$ and set $S$ as the distance of $y$ to the closest point in $S$, that is $d(x, S) = \min_{y \in S} d(x, y)$. For a set $S$, we denote its cardinality by $|S|$.

## 2. Nearest Neighbor Rules

Given a metric space $(\mathcal{X}, d)$ and a labeled sample $S \subset \mathcal{X}$ (where $l(x)$ is the label of point $x \in \mathcal{X}$), the nearest neighbor condensing problem is to find a subset $\tilde{S} \subset S$ of minimal cardinality, such that the nearest neighbor rule on $\tilde{S}$ classifies all sample points in $S$ correctly; that is, for any point $x \in S$, $l(x) = l(\text{argmin}_{y \in \tilde{S}} (d(x, y)))$.

Now let $w : \mathcal{X} \to (0, \infty)$ be a positive weight function, and define the weighted distance

$$\tilde{d}(x, x') = \frac{d(x, x')}{w(x) \cdot w(x')}, \qquad x, x' \in \mathcal{X}.$$

This is our weighted distance nearest neighbor (WNN) rule, under which we may define a WNN classifier:

$$h_{(\tilde{S}, w)}(x) = l(\arg\min_{y \in \tilde{S}} \tilde{d}(y, x)), \qquad x \in \mathcal{X}.$$

Note that the weighted distance may not satisfy the triangle inequality. In the weighted condensing problem, we seek a subset $\tilde{S} \subset S$ and a weight assignment $w$ for $\tilde{S}$ (with $w(x) = 1$ for all $x \in \mathcal{X} \setminus \tilde{S}$), such that for each point $x \in S$, the weighted distance nearest neighbor of $x$ in $\tilde{S}$ has the same label as $x$, $h_{(\tilde{S}, w)}(x) = l(x)$.

It is easy to see that WNN is a generalization of the NN rule, as the latter can be recovered by simply taking all weights to be equal to 1. Let us motivate our weighted distance function by illustrating the effect of weighting on the Euclidean decision boundary between two points. Consider two points in the Euclidean plane, $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$. If $w(p_1) = w(p_2)$, then the WNN decision boundary is equivalent to the NN decision boundary, defined by

$$\sqrt{(x_1 - x)^2 + (y_1 - y)^2} = \sqrt{(x_2 - x)^2 + (y_2 - y)^2},$$

which can be simplified to a line in slope-intercept form:

$$y = x \cdot \frac{x_1 - x_2}{y_2 - y_1} + \frac{-x_1^2 + x_2^2 - y_1^2 + y_2^2}{2(y_2 - y_1)}.$$

In contrast, when $w(p_1) \neq w(p_2)$, the decision boundary is defined by

$$\frac{\sqrt{(x_1 - x)^2 + (y_1 - y)^2}}{w_1} = \frac{\sqrt{(x_2 - x)^2 + (y_2 - y)^2}}{w_2}.$$

This can be simplified to the standard equation of a circle

$$\begin{aligned}
&\left(x + \frac{w_1^2 x_2 - w_2^2 x_1}{w_2^2 - w_1^2}\right)^2 + \left(y + \frac{w_1^2 y_2 - w_2^2 y_1}{w_2^2 - w_1^2}\right)^2 \\
&= \frac{w_1^2 y_2^2 + w_1^2 x_2^2 - w_2^2 y_1^2 - w_2^2 x_1^2}{w_2^2 - w_1^2} \\
&\quad + \left(\frac{w_1^2 x_2 - w_2^2 x_1}{w_2^2 - w_1^2}\right)^2 + \left(\frac{w_1^2 y_2 - w_2^2 y_1}{w_2^2 - w_1^2}\right)^2.
\end{aligned}$$

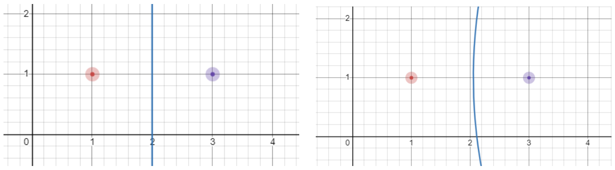

Figure 1: Illustration of decision boundary for equal (left) and unequal weights (right). In the right figure, $w(p_1) = 1.5$ and $w(p_2) = 1$, where $p_1$ is its left point and $p_2$ is its right point.

See Figure 1. The fact that WNN induces a circular boundary (and in higher dimension, a ball separator) will be useful for our proofs and constructions. The separator induced by the standard unweighted NN rule is simply the Voronoi diagram.

While the distance function for nearest neighbor condensing rules is often taken to be the Euclidean norm, all our results below hold for all metric distance functions.

## 3. Properties of Weighted Distance Nearest Neighbor

In this section, we establish condensing properties under WNN, especially in comparison to condensing properties already known for the regular unweighted nearest neighbor (NN) rule. We show that the former rule is strictly more powerful: It can always yield condensing as good as the latter, and in some cases better by a factor of $\Theta(n)$. At the same time, we derive generalization bounds for WNN condensing that are essentially the same as those known for NN condensing, so that the utilization of a more powerful tool does not lead to an increase in generalization error.

### 3.1. Condensing Bounds

As weighted nearest neighbor generalizes unweighted nearest neighbor, it can only improve the condensing. We can in fact show the following:

**Theorem 3.1.** *For any $n$, there exists an $n$-point data-set that can be condensed to 2 candidates under the WNN rule, but requires $\Theta(n)$ candidates under the NN rule.*

In place of simply presenting a construction achieving the bound of Theorem 3.1, we will instead present a new condensing rule we call the ball cover (BC) rule, and show that WNN generalizes this rule as well. We then demonstrate that condensing under NN and BC is incomparable, in that each rule admits $n$-point sets that it can condense to a constant size, but which the other cannot condense below $\Theta(n)$ points. This expanded explanation will yield a better understanding of the power of WNN, and motivate the heuristic of Section 4 below.

**Ball Cover Rule.** We define another rule for condensing, the ball cover (BC) rule: Given the input points $S \subset \mathcal{X}$, we must produce a subset $\tilde{S} \subset \mathcal{X}$ of minimum cardinality, and also assign each point $x_i \in \tilde{S}$ a radius $r_i$. Let $B_i = B(x_i, r_i)$. We require for all $x_i \in \tilde{S}$ and $x_j \in S$ with $l(x_i) \neq l(x_j)$, that $r_i < d(x_i, x_j)$. It follows that no ball contains sample points of the opposite label. The decision rule simply assigns a point $x \in S$ the same label as the center of a ball containing $x$. A valid BC condensing satisfies that every $x \in S \setminus \tilde{S}$ is found in a ball $B_i$ satisfying $l(x) = l(x_i)$. (There is however a caveat relating to the labeling of points not in the sample: These points can fall into multiple balls, or no ball at all. In these cases, one may impose some arbitrary decision rule, such as a priority over the balls.)

The BC rule is motivated by the NET algorithm of Gottlieb et al. (2018). This algorithm can be viewed as covering the space with balls of equal radius. The BC rule is an extension of the NET approach to balls of different radii.

It is easy to show that WNN generalizes the BC rule: Given a condensed set $\tilde{S}$ with assigned radii $r_i$, a WNN classifier can be produced by taking the same points of $\tilde{S}$, and assigning weight $w(x_i) = r_i$ for each $x_i \in \tilde{S}$. Consider any point $x \in S \setminus \tilde{S}$ falling in $B_i$ but not in $B_j$, and we have that $\tilde{d}(x, x_i) = \frac{d(x,x_i)}{w(x_i)} < \frac{r_i}{r_i} = 1$, while $\tilde{d}(x, x_j) = \frac{d(x,x_j)}{w(x_j)} \geq \frac{r_j}{r_j} = 1$, so that the WNN rule is indeed consistent with the BC rule on the sample.

**Comparison Between NN and BC.** It remains to prove the following lemma, concerning condensing under the NN and BC rules:

**Lemma 3.2.** *For any $n$, there exists an $n$-point data-set that can be condensed to 2 candidates under the NN rule, but requires $\Theta(n)$ candidates under the BC rule.*

*Likewise, for any $n$, there exists an $n$-point data-set that can be condensed to 2 candidates under the BC rule, but requires $\Theta(n)$ candidates under the NN rule.*

As WNN generalizes BC, Theorem 3.1 follows immediately from Lemma 3.2. It remains to prove the lemma.

*Proof of Lemma 3.2.* For the first item of the lemma, assume without loss of generality that $n$ is even, and set $\gamma = n/2$. Our example set $\mathcal{X}$ will have $\gamma$ red points and $\gamma$ blue points, with a diameter $\Theta(n)$ and minimum distance of 1 between the points. Let $c = \frac{1}{2}$, and add to the set $\gamma$ red points at positions $\{(i, c) : i \in [\gamma]\}$, and $\gamma$ blue points at positions $\{(i, -c) : i \in [\gamma]\}$. See Figure 2.

Now under the NN rule, there exists a solution which uses only two points: This solution $S$ takes the red point at $(1, c)$ together with the blue point at $(1, -c)$: Take any red point at $(i, c)$, and its distance from the red candidate is $i - 1$, while its distance from the blue candidate is exactly $\sqrt{(i-1)^2 + 4c^2} > i - 1$. A similar argument applies to the blue points, and so we conclude that $S$ is a consistent condensing of $\mathcal{X}$.

Turning to the BC rule, we show that any solution under this rule must have all $n$ points: Assume by contradiction that we can consistently cover all the points of $\mathcal{X}$ with only $k$ balls, where $k < n$. This implies that there exists some solution ball which covers 2 or more points of the same color; assume without loss of generality that this ball is red, and its center is $p_i = (i, c)$. As this ball covers more than a single red point, its radius must be greater than 1, but then it contains the blue point $(i, -c)$, which is forbidden. We conclude that no ball contains more than one point, and so the optimal solution under BC must contain exactly $n$ points.

For the second item of the lemma, assume without loss of generality that $n$ is odd, set $\gamma = \frac{n-1}{2}$, and further assume that $\gamma$ is odd. Our example has $\gamma$ blue points and $\gamma + 1$ red points: Let the points $\{(0, 2i) : i \in [\gamma]\}$ be red, and the points $\{(1, 2i+1) : i \in [\gamma - 1]\}$ be blue. Set $t = \frac{(\gamma+1)^2}{2}$, and add an additional red point $r = (-t, \gamma + 1)$, and an additional blue point $b = (2t, \gamma + 1)$. See Figure 2.

We show that under the BC rule, two balls are sufficient to correctly label all points: First take the red ball centered at $r$ with radius $\sqrt{t^2 + (\gamma+1)^2} = \sqrt{t^2 + 2t} < t + 1$. Clearly this ball contains no blue points, since the line containing the blue points is at distance exactly $t + 1$ from $r$. But the ball contains all the red points, as the farthest red points from $r$ are at $(0, 2)$ and $(0, 2\gamma)$, both of which are at distance $\sqrt{t^2 + (\gamma - 1)^2}$ from the ball center, and hence inside the ball. Now take the blue ball centered at $b$ with radius $\sqrt{4t^2 + (\gamma+1)^2} = \sqrt{(2t)^2 + 2t} < 2t + 1$. A similar argument to that above shows that this ball contains all blue points, but none of the red.

We show that under the NN rule, at least $n - 2$ points must be found in the condensed set. First take the blue points on the line $x = 1$, and at least one of these must be in the condensed set: If only $b$ were in the condensed set, then the other blue points would be closer to every red point in the condensed set. Now take a blue point $(1, 2i + 1)$ in the condensed set, and it must be that the red points $(0, 2i), (0, 2i + 2))$ are both in the condensed set, since the blue point is their nearest neighbor. It similarly follows that blue points $(1, 2i-1), (1, 2i+3)$ are both in the condensed, and in fact that all red points on the line $x = 0$ and all blue points on the line $x = 1$ are in the condensed set. □

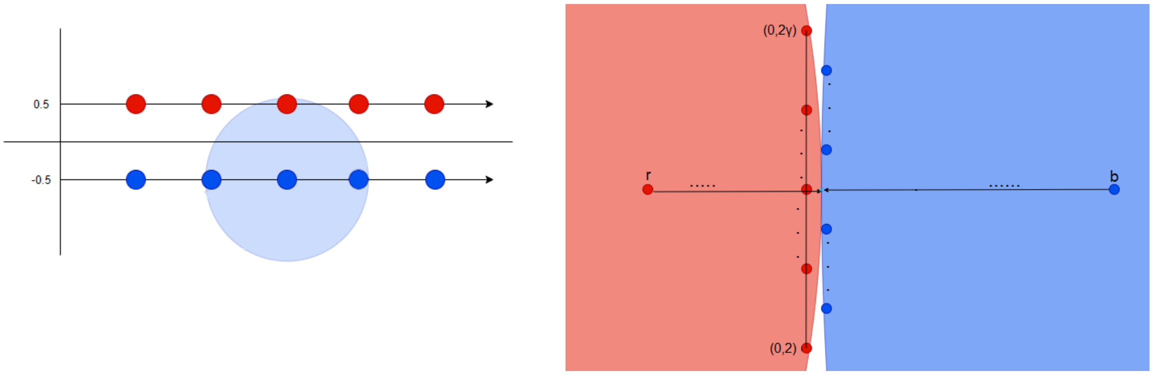

Figure 2: Left: A set that admits good condensing under the NN rule, but not under the BC rule. Right: A set that admits good condensing under the BC rule, but not under the NN rule.

### 3.2. Learning Bounds

Here we establish uniform generalization bounds for WNN rules. By definition, a general WNN classifier is uniquely determined by a subset of labeled samples $\tilde{S}$ and a weight function $w$. In addition, since $w(x) = 1$ for all $x \notin \tilde{S}$, $w$ is uniquely determined by its values on $\tilde{S}$. The following representation lemma establishes a $2|\tilde{S}|$-sample compression scheme for the class of WNN rules, which will allow us to derive compression-based error bounds (Littlestone & Warmuth, 1986; Floyd & Warmuth, 1995; Graepel et al., 2005).

**Lemma 3.3.** *There exist an encoding function $\mathcal{C}$ and a reconstruction function $\mathcal{R}$ that satisfy the following: For any finite labeled set $S$, any subset $\tilde{S} \subset S$, and any weight assignment $w$ for $\tilde{S}$, if the WNN classifier corresponding to the pair $(\tilde{S}, w)$ is consistent on $S$, then $\mathcal{C}(S, \tilde{S}, w)$ returns two ordered labeled subsets $\tilde{S}', \tilde{\mathcal{W}}' \subset S$, each of size $|\tilde{S}|$, such that the WNN classifier $h_{\tilde{S}',\tilde{\mathcal{W}}'} = \mathcal{R}(\tilde{S}', \tilde{\mathcal{W}}')$ is consistent on $S$.*

*Proof.* We first describe the encoding function $\mathcal{C}$ that given $S$ and the pair $(\tilde{S}, w)$, selects $\tilde{S}', \tilde{\mathcal{W}}'$. Then we will describe the reconstruction function $\mathcal{R}$ and conclude that $\mathcal{R}(\tilde{S}', \tilde{\mathcal{W}}')$ is indeed consistent on $S$.

Given $S$ and $(\tilde{S}, w)$, the encoding function $\mathcal{C}$ returns two ordered lists $\tilde{S}', \tilde{\mathcal{W}}' \subset S$ such that $\tilde{S}'$ has the same sample points as $\tilde{S}$ but in a specific order, and $\tilde{\mathcal{W}}'$ consists of sample points from $S$ from which appropriate weights for $\tilde{S}'$ can be deduced via $\mathcal{R}$. The first sample in the list $\tilde{S}'$ corresponds to $x \in \tilde{S}$ having the maximal weight, and the first sample in $\tilde{\mathcal{W}}'$ is also taken as $x$. Subsequent points are added to $\tilde{S}'$ and $\tilde{\mathcal{W}}'$ by the following procedure: Starting with initial weights as determined by $w$, multiplicatively increase the weights of all points of $\tilde{S}$ not yet placed in $\tilde{S}'$, until one of the following occurs:

(i) A point $x \in \tilde{S} \setminus \tilde{S}'$ has weight equal to that of a point already in $\tilde{S}'$, say $x_1$. Then $x$ is added to $\tilde{S}'$ and $x_1$ is added to $\tilde{\mathcal{W}}'$.

(ii) There is some point $x \in S \setminus \tilde{S}$ that became equidistant (under weighted distance) from its closest point $x_1 \in \tilde{S}$ of the same label, and its closest point $x_2 \in \tilde{S}$ with opposite label. It must be that $x_1$ is already in $\tilde{S}'$, or else the weights of $x_1, x_2$ would increase in unison. Then $x_2$ is added to $\tilde{S}'$ and $x$ is added to $\tilde{\mathcal{W}}'$.

The weight increase procedure is carried on until all samples of $\tilde{S}$ have been placed in $\tilde{S}'$.

As for the reconstruction, given two lists of sample point-label pairs $\tilde{S}' = ((x_1, y_1), \ldots, (x_m, y_m))$ and $\tilde{\mathcal{W}}' = ((x'_1, y'_1), \ldots, (x'_m, y'_m))$ that have been computed by $\mathcal{C}$ as described above, the reconstruction function $\mathcal{R}$ construct a WNN classifier corresponding to the pair $(\tilde{S}', w')$, where the weight assignment $w'$ is computed as follows. The weight of the first point in $\tilde{S}'$ is set to $w'(x_1) = 1$. The weights of subsequent points in $\tilde{S}'$ are set depending on their corresponding points in $\tilde{\mathcal{W}}'$: For $k > 1$,

- If $x_k$ appears in $(x_1, \ldots, x_{k-1})$, say as $x_j$, then we are in case (i), and thus put $w'(x_k) = w'(x_j)$.
- If $x_k$ does not appear in $(x_1, \ldots, x_{k-1})$, then we are in case (ii), and $(x'_k, y'_k)$ corresponds to a witness for $(x_k, y_k)$ and some other point in $(x_1, \ldots, x_{k-1})$, say $(x_j, y_j)$. The identity of $x_j$ can be inferred by finding the point in $(x_1, \ldots, x_{k-1})$ with label opposite to $y_k$ and having the minimal weighted distance to $x'_k$,

$$x_j = \operatorname*{arg\,min}_{x_i \in \{x_1, \ldots, x_{k-1}\} : y_i \neq y_k} \left\{ \frac{d(x'_k, x_i)}{w'(x_i)} \right\},$$

breaking ties towards $x_i$ with smallest index $i$. Then

$w'(x_k)$ is set to satisfy the equation

$$\frac{d(x'_k, x_k)}{w'(x_k)} = \frac{d(x'_k, x_j)}{w'(x_j)}.$$

Since multiplying the weights of several points in unison do not change their pairwise weighted-distance boundaries, it is clear that during the whole construction process done in $\mathcal{C}$, the classifier remains consistent on $S$, provided ties in weighted distances are decided in favor of points in $\tilde{S}'$ of smaller index. Hence $\mathcal{R}(\tilde{S}', \tilde{\mathcal{W}}')$ is consistent on $S$. □

Lemma 3.3 can be used to derive generalization bounds, as we show in Theorem 3.4. But note first that the reconstruction function $\mathcal{R}$ of Lemma 3.3 heavily relies on that $\tilde{S}'$ and $\tilde{\mathcal{W}}'$ are ordered. Below in Section 4 we will consider a subclass of WNNs whose $\mathcal{R}$ assumes no such matching is given, and this matching needs to be deduced. In this case a tighter generalization bound holds, which we will leverage to establish Bayes consistency for the aforementioned subclass in Section 4.2. Formally, a reconstruction function $\mathcal{R}$ is said to be *permutation invariant* if for any two arbitrary permutations $\sigma_1, \sigma_2$ of the samples in $\tilde{S}'$ and $\tilde{\mathcal{W}}'$ respectively,

$$\mathcal{R}(\sigma_1(\tilde{S}'), \sigma_2(\tilde{\mathcal{W}}')) = \mathcal{R}(\tilde{S}', \tilde{\mathcal{W}}'). \quad (1)$$

In other words, a permutation invariant $\mathcal{R}$ is able to deduce the matching between the samples in $\tilde{S}'$ and their weights from the unordered elements of $\tilde{S}'$ and $\tilde{\mathcal{W}}'$ alone.

We can now present the generalization bounds. These essentially correspond to sample compression-based error bounds for the compression scheme established in Lemma 3.3. See Section 1.1 for previously known bounds related to these.

**Theorem 3.4.** *For any probability distribution of $x$, any labeling function $l : \mathcal{X} \to \{-1, 1\}$, and any $n \in \mathbb{N}$ and $0 < \delta < 1$, it holds that with probability $1 - \delta$ over the i.i.d. labeled sample $S = \{(x_1, l(x_1)), \dots, (x_n, l(x_n))\}$, for any $\tilde{S} \subset S$ and weight assignment $w$ for $\tilde{S}$, if the WNN classifier $h_{(\tilde{S},w)}$ corresponding to the pair $(\tilde{S}, w)$ correctly classifies all points in S, then*

$$\text{err}(h_{(\tilde{S},w)}) \le \frac{2}{n - |\tilde{S}|} \left( |\tilde{S}| \log 2n + \log \frac{n}{\delta} \right). \quad (2)$$

*If in addition the reconstruction function $\mathcal{R}$ is permutation invariant, then*

$$\text{err}(h_{(\tilde{S},w)}) \le \frac{2}{n - |\tilde{S}|} \left( |\tilde{S}| \log \left( \frac{2en}{|\tilde{S}|} \right) + \log \frac{n}{\delta} \right). \quad (3)$$

The proof of Theorem 3.4 is deferred to Appendix A.

**Algorithm 1** Greedy weighted heuristic

```
Input: Point set $S$
Initialize solution set $T \leftarrow \emptyset$, $S' \leftarrow S$, weight function $w : S \to \{1\}$
while $S' \neq \emptyset$ do
    $x \leftarrow \text{argmax}_{x \in S} |B(x, d_{\text{ne}}(x)) \cap S'|$
    $S' \leftarrow S' \setminus B(x, d_{\text{ne}}(x))$
    $T \leftarrow T \cup \{x\}$
    $w(x) \leftarrow d_{\text{ne}}(x)$
end while
return $T, w$.
```

## 4. Greedy Heuristic for WNN, and its Properties

In this section, we suggest a heuristic to produce a weighted condensed set. The heuristic is motivated by the ball cover rule introduced above. After presenting the heuristic, we establish that it is Bayes consistent under mild assumptions.

### 4.1. Heuristic

Our heuristic for WNN condensing is based on a greedy approach for the BC rule, meaning that at each step, we identify a ball covering the maximum number of uncovered points of the same label (and no points of the opposite label), and add it to our ball set. Similarly, in our WNN heuristic (see Algorithm 1), we iteratively add to the condensed set a point and weight which correspond to the center and radius of the ball covering the largest number of as of yet not covered points. (The equivalence of the radius in the BC rule to weight in the WNN rule was already established above in Section 3.1.) As in (Flores-Velazco & Mount, 2021), the notation $d_{\text{ne}}(x)$ denotes the distance from $x$ to its closest oppositely labeled point in $S$ (its 'nearest enemy').

### 4.2. Bayes Consistency

In the following, we consider a family of WNN classifiers that use a specific (data-dependent) weight function $w_{\text{ne}}$, that assigns to each data point $(\tilde{x}, \tilde{y}) \in \tilde{S} \subset S$ weight equal to the minimal distance from $\tilde{x}$ to the points in $S$ that have the opposite label from $\tilde{y}$, and for points $x \notin \tilde{S}$ assigns $w_{\text{ne}}(x) = 1$; that is, defining $S^+$ and $S^- = S \setminus S^+$ as the split of $S$ into positively and negatively labeled points respectively, $w_{\text{ne}}(\tilde{x}) = d_{\text{ne}}(\tilde{x})$, where

$$d_{\text{ne}}(\tilde{x}) = \begin{cases} d(\tilde{x}, S^-), & \text{if } \tilde{x} \in \tilde{S}^+, \\ d(\tilde{x}, S^+), & \text{if } \tilde{x} \in \tilde{S}^-. \end{cases}$$

Note that for this subclass of WNN classifiers there is a simpler compression scheme than that of Lemma 3.3;

in particular, the reconstruction function $\mathcal{R}$ can be made permutation invariant in the sense of (1). Indeed, the weight assignment $w_{\text{ne}}$ for $\tilde{S}$ can be encoded into a subset $\tilde{\mathcal{W}} \subset S \setminus \tilde{S}$ consisting of the nearest enemies of the samples in $\tilde{S}$. Then the weight for $(\tilde{x}, \tilde{y}) \in \tilde{S}$ can be uniquely determined by splitting $\tilde{\mathcal{W}}$ into its positively and negatively labeled samples $\tilde{\mathcal{W}}^{+}$ and $\tilde{\mathcal{W}}^{-}$ and putting $w_{\text{ne}}(\tilde{x}) = \min_{(x,y)\in\tilde{\mathcal{W}}^{-\tilde{y}}} d(\tilde{x}, x)$. With this rule, for any two permutations $\sigma_1, \sigma_2$, $\mathcal{R}(\sigma_1(\tilde{S}), \sigma_2(\tilde{\mathcal{W}})) = \mathcal{R}(\tilde{S}, \tilde{\mathcal{W}})$.

We first consider the (computationally intractable) learning rule that finds the subset $\tilde{S}^* \subset S$ of minimal cardinality such that the classifier $h_{(\tilde{S}^*, w_{\text{ne}})}$ is consistent on $S$,

$$\tilde{S}^* = \underset{\tilde{S}\subset S}{\arg\min}\{|\tilde{S}| : h_{(\tilde{S}, w_{\text{ne}})}(x) = y, \forall (x, y) \in S\}. \quad (4)$$

The following theorem establishes the Bayes consistency[1] of $h_{(\tilde{S}^*, w_{\text{ne}})}$, meaning that its error on a new sample, drawn independently from the same probability distribution that generated the dataset, converges to zero as the dataset size increases, with probability one over the random dataset.

**Theorem 4.1.** *Let $(\mathcal{X}, d)$ be a separable metric space and assume $x$ has an atomless distribution and that the labeling function $l$ is countably piece-wise continuous. Then, almost surely,*

$$\text{err}(h_{(\tilde{S}^*, w_{\text{ne}})}) \xrightarrow[n\to\infty]{} 0. \quad (5)$$

The proof of Theorem 4.1 is given in Appendix A. The proof essentially establishes that $|\tilde{S}^*|$ is almost surely sub-linear in the sample size $n$. An application of the error bound (3) of Theorem 3.4 (corresponding to a permutation-invariant rule) then establishes (5). Note that a sub-linear $|\tilde{S}^*|$ in conjunction with the error bound (2) (corresponding to a non-permutation invariant rule) do not suffice to establish Bayes consistency with our proof technique: Without further assumptions on the tail of the distribution of $x$, the rate at which $|\tilde{S}^*|/n \xrightarrow[n\to\infty]{} 0$ can be arbitrarily slow.

As for our greedy heuristic in Algorithm 1, note that the intractable optimization problem (4) can be cast as a set cover problem. Algorithm 1 then corresponds to the standard greedy approximation for set cover (Chvatal, 1979). This approximation is guaranteed to compute $\tilde{S}$ of size at most $O(|\tilde{S}^*| \log |\tilde{S}^*|)$. Hence, if $|\tilde{S}^*| \log |\tilde{S}^*|$ is guaranteed to be almost surely sub-linear, an adaptation of our proof of Theorem 4.1 implies that the greedy heuristic is Bayes consistent. This is made formal in the following Corollary.

**Corollary 4.2.** *Under the conditions of Theorem 4.1 and an additional appropriate tail condition on the probability distribution of $x$, the greedy weighted heuristic of Algorithm 1 is Bayes consistent.*

*Proof.* In the proof of Theorem 4.1 we fetched a function $t_{r^*} : \mathbb{N} \to \mathbb{R}^+$ in $o(1)$ to establish that the size of a $r^*$-net (with $r^* > 0$) is sub-linear in $n$. Inspecting the proof of Theorem 4.1, to guarantee that $|\tilde{S}^*| \log |\tilde{S}^*|$ is almost surely sub-linear in $n$, it suffices to additionally assume that $t_{r^*} \in o(1/\log n)$. □

As two examples of the applicability of Corollary 4.2, if random variable $x$ is bounded then $|\tilde{S}^*| = O(1)$, and if $x$ has a normal distribution then $|\tilde{S}^*| = O(\log n)$. Hence in these examples Algorithm 1 is Bayes consistent.

## 5. Experimental Results

In this section we present promising experimental results for condensing under WNN, using the heuristic of Section 4. We ran two separate trials: The first experiment was a comparison of condensing achieved by our results to those achieved by other popular condensing algorithms. For these we used datasets already established as appropriate for condensing. The second experiment also compared condensing algorithms, but here we also computed the optimal unweighted compressed set and compared our results to these. This required the introduction of an exact integer program, and the use of very small datasets amenable to exact solutions.

### 5.1. Trial: Comparison Between Condensing Algorithms

As proof of concept, we selected representative datasets from the condensing experiments of (Garcia et al., 2012) (see Tables 2 and 7 there), appearing in Table 1. For each data set, we randomly split it into training samples (70%) and testing samples (30%). On the training sets, we ran the popular MSS (Barandela et al., 2005) and the recent successful RSS (Flores-Velazco & Mount, 2021) heuristics, as well as our greedy heuristic for WNN condensing, as presented above in Algorithm 1. We found that across all datasets considered, our weighted condensing heuristic achieved either superior or comparable compression of the training dataset, and either superior or comparable accuracy on the testing data set, when compared to the unweighted heuristics; see Table 1, which reports the size of the condensed set as a fraction of the size of the original input training dataset, and the test error as the fraction of wrongly classified samples from the test dataset.

[1] Not to be confused with consistency, which here means that $h_{(\tilde{S}^*, w_{\text{ne}})}$ correctly classifies all points in the uncondensed dataset.

| Dataset | Size | Classes | Fraction retained | | | Test error | | |
|---|---|---|---|---|---|---|---|---|
| | | | MSS | RSS | WNN | MSS | RSS | WNN |
| Magic | 19,020 | 2 | 0.29 | 0.37 | **0.26** | 0.22 | 0.26 | **0.21** |
| SatImage | 6,430 | 7 | 0.15 | 0.19 | **0.14** | 0.11 | 0.12 | **0.09** |
| Spambase | 4,560 | 2 | **0.27** | 0.33 | **0.27** | 0.21 | 0.21 | **0.18** |
| Twonorm | 7,400 | 2 | 0.15 | 0.16 | **0.06** | 0.06 | 0.11 | **0.03** |
| Phoneme | 5,404 | 2 | 0.19 | 0.22 | **0.16** | 0.13 | **0.12** | **0.12** |
| Segment | 2,310 | 7 | 0.13 | 0.14 | **0.10** | 0.07 | **0.05** | **0.05** |
| Shuttle | 43,498 | 7 | 0.030 | 0.008 | **0.005** | 0.004 | **0.002** | **0.002** |

Table 1: The fraction of training samples retained in the condensed subset and the error achieved on the testing samples as described in Section 5.1.

### 5.2. Trial: Comparison with Exact Solvers on Small Datasets

In this experiment, we compared our condensing algorithm with the optimal solution produced by a brute-force solver. This is instructive in understanding the quality of the tested heuristics. Due to the significant limitations inherent in producing an exact solution, our comparison necessitated the use of small dataset amenable to computing an optimal solution using an integer program (IP) solver.

**Integer Programming for NN Condensing.** We formalize an integer program for NN condensing. As this problem is NP-hard, we do not expect the algorithm to have a reasonable run time for large sets, but for smaller sets it successfully returns an optimal solution (after large run time).

To model the NN condensing problem as an integer program, we introduced constraints corresponding to the inclusion of a point in the condensed set. This allowed us to identify the minimal non-empty subset of examples that can recover all labels of the sample via the nearest neighbor classifier.

For each sample point $x$, we introduce a 0–1 variable $v(x)$, corresponding to whether $x$ will appear in the condensed set. For each ordered pair of points $x$ and $x'$ with opposite labels, we introduce the constraint $v(x') \leq \sum_{x'' \in C(x,x')} v(x'')$, where $C(x,x')$ is the set of points with the same label as $x$ which are all closer to $x$ than $x'$ is to $x$. (Note that $x \in C(x,x')$.) This constraint enforces that if $x'$ appears in the condensed set (meaning $v(x') = 1$), then there must be in the condensed set some point closer to $x$ with the same label as $x$. We also need the constraint $\sum_x v(x) \geq 1$, to disallow the empty set. Finally, the objective is to minimize $\sum_x v(x)$, which corresponds to minimizing the size of the condensed set. We implemented this program using the Python cvxpy library.

Table 2: Number of samples retained in condensed subset

| Dataset | Points | MSS | RSS | IP | WNN |
|---|---|---|---|---|---|
| Circle | 200 | 52 | 45 | 7 | 12 |
| Banana | 200 | 74 | 66 | 32 | 35 |
| Iris | 100 | 11 | 9 | 2 | 4 |

**Datasets.** While the condensing heuristics can handle larger datasets, we found that (not surprisingly) the exact IP algorithm failed for set sizes much larger than 200 points. Accordingly, we ran trials on the small banana, circle and iris data sets, which all have binary classes, see Figure 3.

- Banana. This data set is a synthetic collection, previously used by Flores-Velazco & Mount (2021) for NNC. It contains instances arranged in several banana-shaped clusters. (The $x$ and $y$ axes represent the respective properties At1 and At2 defined there.) For our experiment, we retained only 200 of more than 5000 original points.
- Circle. This is a synthetic randomized data set containing 200 points. It contains instances arranged in a circular cluster, surrounded by instances of the opposing class.
- Iris. This is the very popular data set of the UCI Machine Learning Repository. It consists of three classes, each containing 50 instances of a certain species of iris. For our experiments, we considered only two classes of the three (namely Setosa and Versicolour).

**Results.** We ran the above NN condensing heuristics, the exact IP algorithm, and also our greedy heuristic for WNN condensing on the small data sets. We again found that our weighted condensing heuristic achieved superior compression when compared to the unweighted heuristics; see Table 2 which reports the exact sizes of the condensed sets. Our heuristic also approached the optimal unweighted solution computed by the brute-force IP, an algorithm which (unlike ours) does not scale to larger datasets.

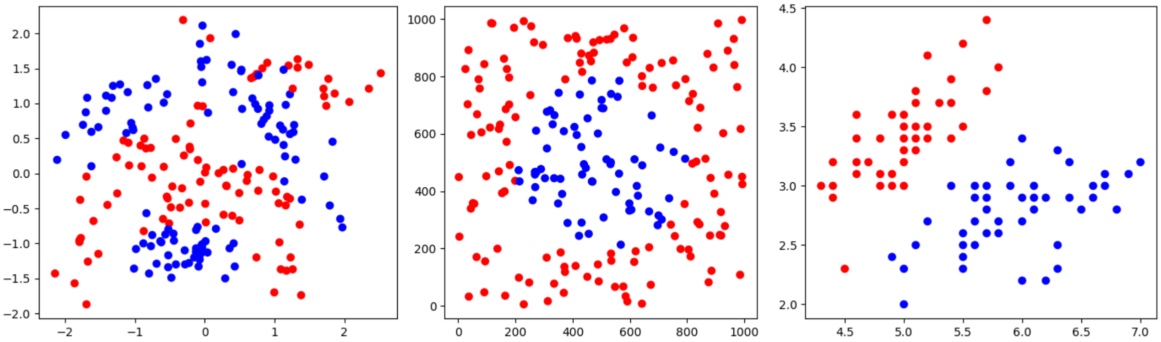

Figure 3: The banana, circle and iris data sets

## 6. Conclusions

We have demonstrated that WNN condensing is more powerful than standard NN condensing, yet is characterized by similar generalization bounds. Hence WNN can only improve the degree of compression, while maintaining the same theoretical guarantees. Our suggested heuristic is theoretically sound, and gave promising empirical results. This indicates that WNN condensing heuristics are deserving of further study.

## Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

## References

Angiulli, F. Fast condensed nearest neighbor rule. In *Proceedings of the 22nd international conference on Machine learning*, pp. 25–32, 2005.

Barandela, R., Ferri, F. J., and Sánchez, J. S. Decision boundary preserving prototype selection for nearest neighbor classification. *International Journal of Pattern Recognition and Artificial Intelligence*, 19(06):787–806, 2005.

Chvatal, V. A greedy heuristic for the set-covering problem. *Mathematics of operations research*, 4(3):233–235, 1979.

Cohen, D. T. and Kontorovich, A. Learning with metric losses. In Loh, P.-L. and Raginsky, M. (eds.), *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 662–700. PMLR, 02–05 Jul 2022.

Cover, T. and Hart, P. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1): 21–27, 1967. doi: 10.1109/TIT.1967.1053964.

Devi, V. S. and Murty, M. N. An incremental prototype set building technique. *Pattern Recognition*, 35(2):505–513, 2002.

Devroye, L., Györfi, L., and Lugosi, G. *A probabilistic theory of pattern recognition*. Springer-Verlag New York, Inc., 1996.

Dudani, S. A. The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, (4):325–327, 1976.

Fix, E. and Hodges, J. L. Discriminatory analysis, nonparametric discrimination: Consistency properties. 57 (3):238–247, 1951.

Flores-Velazco, A. Social distancing is good for points too! *arXiv preprint arXiv:2006.15650*, 2020.

Flores-Velazco, A. and Mount, D. Guarantees on nearest-neighbor condensation heuristics. *Computational Geometry*, 95:101732, 2021.

Floyd, S. and Warmuth, M. Sample compression, learnability, and the Vapnik-Chervonenkis dimension. *Machine learning*, 21(3):269–304, 1995.

Garcia, S., Derrac, J., Cano, J., and Herrera, F. Prototype selection for nearest neighbor classification: Taxonomy and empirical study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3):417–435, 2012.

Gates, W. The reduced nearest neighbor rule. *IEEE Transactions on Information Theory*, 18:431–433, 1972.

Gottlieb, L., Kontorovich, A., and Krauthgamer, R. Efficient classification for metric data (extended abstract COLT 2010). *IEEE Transactions on Information Theory*, 60(9):5750–5759, 2014. doi: 10.1109/TIT.2014.2339840. URL `http://dx.doi.org/10.1109/TIT.2014.2339840`.

Gottlieb, L.-A. and Kontorovich, A. Non-uniform packings. *Information Processing Letters*, 174:106179, 2022.

Gottlieb, L.-A. and Ozeri, S. Classification in asymmetric spaces via sample compression. *arXiv preprint arXiv:1909.09969*, 2019.

Gottlieb, L.-A., Kontorovich, A., and Nisnevitch, P. Near-optimal sample compression for nearest neighbors. *IEEE Transactions on Information Theory*, 64(6):4120–4128, 2018.

Graepel, T., Herbrich, R., and Shawe-Taylor, J. PAC-Bayesian compression bounds on the prediction error of learning algorithms for classification. *Machine Learning*, 59(1):55–76, 2005.

Györfi, L. and Weiss, R. Universal consistency and rates of convergence of multiclass prototype algorithms in metric spaces. *The Journal of Machine Learning Research*, 22 (1):6702–6726, 2021.

Hanneke, S., Kontorovich, A., Sabato, S., and Weiss, R. Universal Bayes consistency in metric spaces. *The Annals of Statistics*, 49(4):2129 – 2150, 2021. doi: 10.1214/20-AOS2029. URL `https://doi.org/10.1214/20-AOS2029`.

Hart, P. The condensed nearest neighbor rule (corresp.). *IEEE transactions on information theory*, 14(3):515–516, 1968.

Kerem, O. and Weiss, R. On error and compression rates for prototype rules. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 8228–8236, 2023.

Kontorovich, A., Sabato, S., and Weiss, R. Nearest-neighbor sample compression: Efficiency, consistency, infinite dimensions. In *Advances in Neural Information Processing Systems*, pp. 1573–1583, 2017.

Littlestone, N. and Warmuth, M. K. Relating data compression and learnability. unpublished, 1986.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.

Ritter, G., Woodruff, H., Lowry, S., and Isenhour, T. An algorithm for a selective nearest neighbor decision rule (corresp.). *IEEE Transactions on Information Theory*, 21(6):665–669, 1975.

Shalev-Shwartz, S. and Ben-David, S. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

Wilfong, G. Nearest neighbor problems. In *Proceedings of the Seventh Annual Symposium on Computational Geometry*, SCG '91, pp. 224–233, 1991.

Wilson, D. R. and Martinez, T. R. Reduction techniques for instance-based learning algorithms. *Machine Learning*, 38:257–286, 2000.

Xue, L. and Kpotufe, S. Achieving the time of 1-nn, but the accuracy of k-nn. In *International Conference on Artificial Intelligence and Statistics*, pp. 1628–1636. PMLR, 2018.

Zukhba, A. V. NP-completeness of the problem of prototype selection in the nearest neighbor method. *Pattern Recognit. Image Anal.*, 20(4):484–494, December 2010. ISSN 1054-6618.

## A. Deferred proofs

*Proof of Theorem 3.4.* Consider first the case of non-invariant $\mathcal{R}$. Set $m = |\tilde{S}|$ and let $\mathcal{I}_{n,m}$ denote the set of all (ordered) sequences of length $m$ of indices from $\{1, 2, \ldots, n\}$. For $\boldsymbol{i} = (i_1, \ldots, i_m) \in \mathcal{I}_{n,m}$ denote by $S(\boldsymbol{i})$ the subset of $S$ with indices in $\boldsymbol{i}$. For a weight assignment $w'(\boldsymbol{i})$ for $S(\boldsymbol{i})$, write $\widehat{\mathrm{err}}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))})$ for the empirical error of $h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}$ over the $n-m$ samples in $S \setminus S(\boldsymbol{i})$, that is

$$\frac{1}{n-m} \sum_{(x_j,y_j) \in S \setminus S(\boldsymbol{i})} \mathbb{1}_{\{h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}(x_j) \neq y_j\}},$$

and note that the samples in $S \setminus S(\boldsymbol{i})$ are i.i.d. and independent of $S(\boldsymbol{i})$. For $\varepsilon > 0$, we bound

$$\begin{aligned} &\mathbb{P}\Big\{\widehat{\mathrm{err}}(h_{(\tilde{S},w)}) = 0 \ \wedge\ |\tilde{S}| = m \ \wedge\ \mathrm{err}(h_{(\tilde{S},w)}) > \varepsilon\Big\} \\ &\leq \mathbb{P}\Big\{\exists \boldsymbol{i} \in \mathcal{I}_{n,m}, \exists w'(\boldsymbol{i}) : \widehat{\mathrm{err}}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}) = 0 \\ &\qquad\qquad \wedge\ \mathrm{err}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}) > \varepsilon\Big\} \\ &\leq \sum_{\boldsymbol{i} \in \mathcal{I}_{n,m}} \mathbb{E}\Big\{\mathbb{P}\big\{\exists w'(\boldsymbol{i}) : \widehat{\mathrm{err}}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}) = 0 \\ &\qquad\qquad \wedge\ \mathrm{err}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}) > \varepsilon \mid S(\boldsymbol{i})\big\}\Big\}. \end{aligned} \tag{6}$$

Fix $\boldsymbol{i} \in \mathcal{I}_{n,m}$ and $S(\boldsymbol{i})$, and consider the class of WNN classifiers given by

$$\mathcal{H}_{S(\boldsymbol{i})} = \{h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))} : w'(\boldsymbol{i}) \text{ weight assign. for } S(\boldsymbol{i})\}.$$

The growth function $\Pi_{\mathcal{H}_{S(i)}}(n)$ of $\mathcal{H}_{S(\boldsymbol{i})}$ counts the maximum number of different possible labelings of $n$ points from $\mathcal{X}$:

$$\begin{aligned} &\Pi_{\mathcal{H}_{S(i)}}(n) = \\ &\qquad \max_{\{z_1,\ldots,z_n\} \subset \mathcal{X}} \Big|\big\{(h(z_1), \ldots, h(z_n)) : h \in \mathcal{H}_{S(\boldsymbol{i})}\big\}\Big|. \end{aligned}$$

Then by a standard argument (Mohri et al., 2018),

$$\begin{aligned} &\mathbb{P}\big\{\exists w'(\boldsymbol{i}) : \widehat{\mathrm{err}}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}) = 0 \\ &\qquad\qquad \wedge\ \mathrm{err}(h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}) > \varepsilon \mid S(\boldsymbol{i})\big\} \\ &\leq 2\Pi_{\mathcal{H}_{S(i)}}(2(n-m)) \cdot e^{-(n-m)\varepsilon/2}. \end{aligned}$$

To bound $\Pi_{\mathcal{H}_{S(i)}}(2(n-m))$, note that by Lemma 3.3, for any weight assignment $w'(\boldsymbol{i})$ for $S(\boldsymbol{i})$ and any set $S'$ of $2(n-m)$ points from $\mathcal{X}$, there exists a subset $\tilde{\mathcal{W}}' \subset S' \cup \tilde{S}(\boldsymbol{i})$ of size $m$ such that the WNN classifier $h_{S(\boldsymbol{i}),\tilde{\mathcal{W}}'}$ gives the same labeling as $h_{(S(\boldsymbol{i}),w'(\boldsymbol{i}))}$ on the $2(n-m)$ points in $S'$. Since the number of different classifiers in

$$\begin{aligned} &\Big\{h_{S(\boldsymbol{i}),\tilde{\mathcal{W}}'} : \tilde{\mathcal{W}}' \text{ is a list of } m \text{ points} \\ &\qquad\qquad \text{from a sample of size } 2n-m\Big\} \end{aligned}$$

is at most $|\mathcal{I}_{2n-m,m}| \leq |\mathcal{I}_{2n,m}|$, it follows that

$$\Pi_{\mathcal{H}_{S(i)}}(2(n-m)) \leq |\mathcal{I}_{2n,m}|.$$

Hence, Eq. (6) is bounded from above by

$$\sum_{\boldsymbol{i} \in \mathcal{I}_{n,m}} |\mathcal{I}_{2n,m}| e^{-(n-m)\varepsilon/2} \leq |\mathcal{I}_{n,m}||\mathcal{I}_{2n,m}| e^{-(n-m)\varepsilon/2}. \tag{7}$$

Put

$$\begin{aligned}\varepsilon &= \frac{2}{n-m}\left(\log\left(|\mathcal{I}_{n,m}|\cdot|\mathcal{I}_{2n,m}|\right)+\log\frac{n}{\delta}\right) \\ &\le \frac{2}{n-m}\left(m\log 2n+\log\frac{n}{\delta}\right),\end{aligned} \tag{8}$$

where we used the bound $|\mathcal{I}_{n,m}| \le n^m$. Then the right hand side of (7) is $\delta/n$. Summing over the $n$ possible values of $m$ completes the proof for the case of non-invariant $\mathcal{R}$.

As for the case of permutation invariant $\mathcal{R}$, the only difference from the proof above is the definition of $\mathcal{I}_{n,m}$. For the permutation invariant case we take $\mathcal{I}_{n,m}$ to be the set of all (unordered) subsets of $\{1,2,\ldots,n\}$ of size $m$. Then $|\mathcal{I}_{n,m}| \le \binom{n}{m} \le (\frac{en}{m})^m$. Putting this into Eq. (8), we have

$$\varepsilon \le \frac{2}{n-m}\left(m\log\frac{2en}{m}+\log\frac{n}{\delta}\right),$$

in accordance with (3). □

*Proof of Theorem 4.1.* Denote by $\mu$ the probability distribution of $x$ and abbreviate $h_{\tilde{S}^*} = h_{(\tilde{S}^*, w_{\mathrm{ne}})}$. For $r > 0$ let

$$U_r = \left\{x \in \mathrm{supp}(\mu) : \frac{1}{\mu(B_r(x))}\int_{B_r(x)} l(x')\mu(dx') = l(x)\right\};$$

that is, $U_r$ is the set of all points in the support of $\mu$ where $l$ is essentially constant in the ball of radius $r$ around $x$. The assumptions that $x$ is atomless and that $l$ is piece-wise continuous imply that $\mu(U_r)$ is monotonic decreasing and continuous in $r$ and satisfies

$$\lim_{r\to 0}\mu(U_r) = 1. \tag{9}$$

Given $\varepsilon > 0$, let $r^* = r^*(\varepsilon) > 0$ be such that

$$\mu(U_{r^*}) \ge 1-\alpha,$$

where $\alpha = \alpha(\varepsilon) \in (0, 1/8)$ satisfies

$$\frac{8\alpha\log\left(\frac{e}{2\alpha}\right)}{1-4\alpha} \le \frac{\varepsilon}{2}. \tag{10}$$

Since the left hand side of (10) goes to 0 monotonically (and continuously) as $\alpha \to 0$, such an $\alpha$ always exists (this choice of $\alpha$ will be made clear below).

Given a sample $S$ of size $n$, denote by $\boldsymbol{X}_n = (x_1, \ldots, x_n)$ the instances in $S$ and by $\boldsymbol{Y}_n = (y_1, \ldots, y_n)$ their corresponding labels. Let $\boldsymbol{X}_{r^*} \subseteq \boldsymbol{X}_n \cap U_{r^*}$ be an $r^*$-net of $\boldsymbol{X}_n \cap U_{r^*}$ (see (Gottlieb et al., 2018) for the formal definition of an $r$-net) and let $\boldsymbol{Y}_{r^*}$ be the corresponding labels in $\boldsymbol{Y}_n$, stacked into the labeled set $\tilde{S}_{(1)} = (\boldsymbol{X}_{r^*}, \boldsymbol{Y}_{r^*})$. Let $U_{r^*}^c = \mathcal{X} \setminus U_{r^*}$ denote the set complement of $U_{r^*}$ and let $\tilde{S}_{(2)} = S \cap (U_{r^*}^c \times \mathcal{Y})$. Define the labeled dataset

$$\tilde{S}_{r^*} = \tilde{S}_{(1)} \cup \tilde{S}_{(2)} \subset S.$$

Then $h_{\tilde{S}_{r^*}}$ is consistent on $S$. Indeed, any $(x_i, y_i) \in S \cap (U_{r^*}^c \times \mathcal{Y})$ is included in $\tilde{S}_{r^*}$ and is thus classified correctly by $h_{\tilde{S}_{r^*}}$. For any $(x_i, y_i) \in S \cap (U_{r^*} \times \mathcal{Y})$, since $\boldsymbol{X}_{r^*}$ is an $r^*$-net of $\boldsymbol{X}_n \cap U_{r^*}$, there is $\tilde{x} \in \boldsymbol{X}_{r^*}$ with

$$d(x_i, \tilde{x}) < r^*.$$

Since $\tilde{x} \in U_{r^*}$ and $d(x_i, \tilde{x}) < r^*$, we have $y_i = l(x_i) = l(\tilde{x}) = \tilde{y}$ with probability 1. In addition, any point $(x_j, y_j) \in S$ with an opposite label to $\tilde{y}$ satisfies $d(x_j, \tilde{x}) \ge r^*$, and so $w_{\mathrm{ne}}(\tilde{x}) \ge r^*$. Thus,

$$\tilde{d}(x_i, \tilde{x}) = \frac{d(x_i, \tilde{x})}{w_{\mathrm{ne}}(\tilde{x})} < 1.$$

Additionally, any $\tilde{x}' \in \tilde{S}_{r^*}$ with a different label from $y_i$ has weight

$$w_{\mathrm{ne}}(\tilde{x}') \leq d(x_i, \tilde{x}').$$

Thus,

$$\tilde{d}(x_i, \tilde{x}') = \frac{d(x_i, \tilde{x}')}{w_{\mathrm{ne}}(\tilde{x})} \geq 1.$$

So the WNN classifier $h_{\tilde{S}_{r^*}}$ classifies the point $(x_i, y_i)$ correctly in this case as well, and so $h_{\tilde{S}_{r^*}}$ is consistent on $S$. It follows that the subset $\tilde{S}^*$ in (4) computed by the learning rule satisfies

$$|\tilde{S}^*| \leq |\tilde{S}_{r^*}|. \tag{11}$$

We next bound $|\tilde{S}_{r^*}| = |\tilde{S}_{(1)}| + |\tilde{S}_{(2)}|$ with high probability. Since by construction $\mu(U_{r^*}^c) \leq \alpha$, Hoeffding's inequality implies that

$$\mathbb{P}\left\{|\tilde{S}_{(2)}| > 2n\alpha\right\} = \mathbb{P}\left\{|\boldsymbol{X}_n \cap U_{r^*}^c| > 2n\alpha\right\} \leq e^{-2n\alpha^2}.$$

As for $|S_{(1)}|$, by Hanneke et al. (2021, Lemma 3.7), there is $t_{r^*} : \mathbb{N} \to \mathbb{R}^+$ in $o(1)$ such that

$$\mathbb{P}\left\{|\boldsymbol{X}_{r^*}| \geq n t_{r^*}(n)\right\} \leq 1/n^2.$$

Since $t_{r^*} \in o(1)$, we may take $n$ sufficiently large so that $t_{r^*}(n) \leq 2\alpha$. So for all sufficiently large $n$,

$$\mathbb{P}\{|\tilde{S}_{r^*}| > 4\alpha n\} \leq \frac{1}{n^2} + e^{-2n\alpha^2}.$$

We bound

$$\begin{aligned}
&\mathbb{P}\left\{\text{err}(h_{\tilde{S}^*}) > \varepsilon\right\} \\
&\leq \mathbb{P}\left\{\text{err}(h_{\tilde{S}^*}) > \varepsilon \ \wedge\ |\tilde{S}_{r^*}| \leq 4\alpha n\right\} + \mathbb{P}\left\{|\tilde{S}_{r^*}| > 4\alpha n\right\} \\
&\leq \mathbb{P}\left\{\text{err}(h_{\tilde{S}^*}) > \varepsilon \ \wedge\ |\tilde{S}_{r^*}| \leq 4\alpha n\right\} + \frac{1}{n^2} + e^{-2n\alpha^2}.
\end{aligned} \tag{12}$$

To complete the proof we show below that for all sufficiently large $n$,

$$\mathbb{P}\left\{\text{err}(h_{\tilde{S}^*}) > \varepsilon \ \wedge\ |\tilde{S}_{r^*}| \leq 4\alpha n\right\} \leq \frac{1}{n^2}. \tag{13}$$

Since the right hand side of (12) is summable over $n$, the Borel-Cantelli Lemma implies that almost surely,

$$\text{err}(h_{\tilde{S}^*}) \xrightarrow[n\to\infty]{} 0,$$

concluding the proof of the Theorem.

To show (13), put $\delta = \delta_n = 1/n^2$ in (3) and observe that the right hand side of (3) is monotonic increasing with $|\tilde{S}|$. Thus, using (11), we have that under the event $\{|\tilde{S}_{r^*}| \leq 4\alpha n\}$,

$$\begin{aligned}
&\frac{2}{n - |\tilde{S}^*|}\left(|\tilde{S}^*| \log\left(\frac{2en}{|\tilde{S}^*|}\right) + 3\log n\right) \\
&\leq \frac{2}{n - |\tilde{S}_{r^*}|}\left(|\tilde{S}_{r^*}| \log\left(\frac{2en}{|\tilde{S}_{r^*}|}\right) + 3\log n\right) \\
&\leq \frac{2}{n - 4\alpha n}\left(4\alpha n \log\left(\frac{en}{2\alpha n}\right) + 3\log n\right) \\
&= \frac{8\alpha \log\left(\frac{e}{2\alpha}\right)}{1 - 4\alpha} + \frac{3\log n}{(1-4\alpha)n} \\
&\leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon,
\end{aligned}$$

where in the last inequality we used the choice of $\alpha$ in (10) and took $n$ sufficiently large so that $\frac{3\log n}{(1-4\alpha)n} \le \frac{\varepsilon}{2}$. Applying Lemma 3.4, it follows that

$$\begin{aligned}
&\mathbb{P}\left\{\text{err}(h_{\tilde{S}^*}) > \varepsilon \ \wedge \ |\tilde{S}_{r^*}| \le 4\alpha n\right\} \\
&\le \mathbb{P}\left\{\text{err}(h_{\tilde{S}^*}) > \frac{2}{n-|\tilde{S}^*|}\left(|\tilde{S}^*|\log\left(\frac{2en}{|\tilde{S}^*|}\right) + 3\log n\right)\right\} \\
&\le \frac{1}{n^2},
\end{aligned}$$

establishing (13). □