

## HW2: Data Geometry

## 1 Written Exercises

Answer the following questions in 25-100 words each:

1. The “standard” notion of distance that we’ve been using is Euclidean distance. In particular, we measure the distance between two vectors  $\mathbf{x}$  and  $\mathbf{y}$  by the *Euclidean norm* of the vector  $\mathbf{z}$  defined by  $\mathbf{z} = \mathbf{x} - \mathbf{y}$ . The Euclidean norm is, of course, defined by  $\|\mathbf{z}\| = (\sum_d z_d^2)^{1/2}$ . There are other norms that one can define. In fact, there’s a whole class of them, called the  $\ell_p$  norms. The  $\ell_p$  norm,  $\|\cdot\|_p$  is defined below (for  $p > 0$ ):

$$\|\mathbf{z}\|_p = \left( \sum_d |z_d|^p \right)^{\frac{1}{p}}$$

Here,  $|a|$  means the absolute value of  $a$ . It’s easy to see that the Euclidean norm is exactly the  $\ell_2$  norm. The  $\ell_1$  norm is just  $\|\mathbf{z}\|_1 = \sum_d |z_d|$ . This is also known as the Manhattan norm because it measures distances by the number of “blocks” that one would have to walk to get between two points, when roads only run along axes.

Consider the case of using a  $k$ NN classifier, but with the  $\ell_1$  norm to measure distances rather than the  $\ell_2$  (Euclidean) norm. Draw (in two dimensions) a simple case of a binary classification problem for which the  $\ell_1$  classifier would return a different class for a test point than an  $\ell_2$  classifier. In particular, draw  $\geq 1$  training points (one for each class) and a test point that would be classified differently according to the two distance metrics.

What properties of a data set do you imagine would influence whether the  $\ell_1$  distance would work better or worse than the  $\ell_2$  distance?

2. One of the biggest problems with  $k$ NN classifiers is that they are very expensive to apply at test time, even if you use clever data structures and clever applications of the triangle inequality. One way to speed up a  $k$ NN classifier would just be to have fewer training points. Suppose you’re given a training set of  $N$  labeled pairs  $(x_n, y_n)_{n=1}^N$ . Suppose that we want to *throw out* some subset of these points. What criteria would you use for deciding which points to throw out? Sketch an algorithm for doing so.
3. [6350 only] An important notion for a classifier is that of *consistency*. Roughly, a classification algorithm is *consistent* if, whenever it has access to *infinite* amounts of training data, its error rate approaches the optimal error rate (aka, Bayes optimal). Consider the noise-free setting. Here, the Bayes optimal error rate is zero. Is the one-nearest-neighbor algorithm consistent in this setting?
4. [Bonus for all] Continuing on the previous question. Consider the case where there *is* noise. Now, the Bayes optimal error rate *is not* zero. Is the one-nearest-neighbor algorithm consistent in this setting? (Hint: you may have to define what the algorithm should do if there are multiple training points at exactly the same distance from the test point.)