Given are the two distance functions, which shall be discussed in this short note. Let (X, d_X) be a metric space with $A, B \in \mathcal{P}(X)$ and $x \in X$, then we will discuss the following distance functions. 1. $d(x, A) = \inf_{a \in A} d_X(x, a)$ (Point Set Distance) 2. $h(A, B) = \max\{\sup_{a \in A} \inf_{b \in B} d_X(a, b), \sup_{b \in B} \inf_{a \in A} d_X(a, b)\}$

Definitions

Point Set Distance

Definition Point Set Distance Let - (X, d_X) be a metric space - $\mathcal{P}(X)$ the power set of X - $x \in X$
- $A \in \mathcal{P}(X) \setminus \emptyset$

then the distance between x and A can be given as

$$d : X \times \mathcal{P}(X) \setminus \emptyset \to \mathbb{R}$$

with

$$d(x, A) = \inf_{a \in A} d_X(x, a)$$

Remark Metric Axioms

$$\begin{aligned} (i) \quad & d(x, y) = d(y, x) \\ (ii) \quad & d(x, y) \geq 0 \text{ and } d(x, x) = 0 \\ (iii) \quad & d(x, y) = 0 \Rightarrow x = y \\ (iv) \quad & d(x, y) \leq d(x, z) + d(z, y) \end{aligned}$$

- Symmetry

$$d(x, A) = \inf_{a \in A} d_X(a, x) \neq d(A, x)$$

Note, the function is defined such that the first argument is a point and the second argument is a set

- Non-Negativeness

$$d(A, x) = \inf_{a \in A} \underbrace{d_X(a, x)}_{\geq 0}$$

  Since d_X is a metric and is guaranteed to be non-negative, the infimum must be also non-negative

- Positiveness Clearly $x \neq A$ since $x \in X$ and $A \in \mathcal{P}(X)$. Further, there are potentially infinitely many $A \in \mathcal{P}(X)$ that hold $d(x, A) = 0$, i.e. let $A_1 = [0, 3]$, $A_2 = [1, 2]$ and $x = 1$, then

$$\begin{aligned} d(x, A_1) &= \inf_{a \in A_1} d(a, x) = 0 \\ d(x, A_2) &= \inf_{a \in A_2} d(a, x) = 0 \end{aligned}$$

- Triangle Inequality Again, since $x \in X$ and $A \in \mathcal{P}(X)$, we have the problem of properly setting up the triangle inequality, i.e. let $b \in X$ and $B \in \mathcal{P}(X)$, then we have

$$d(x, A) \leq d(x, B) + \underbrace{d(B, A)}_{\text{not defined}}$$
$$d(A, x) \leq d(A, b) + \underbrace{d(b, x)}_{\text{not defined}}$$

We can extend the concept of the **point set distance** to a **ordinary set to set distance** on the same concept

Definition Ordinary Set-to-Set Distance Let - (X, d_X) be a metric space - $\mathcal{P}(X)$ the power set of X - $A, B \in \mathcal{P}(X) \setminus \emptyset$

then the distance between A and B can be given as

$$d : \mathcal{P}(X) \setminus \emptyset \times \mathcal{P}(X) \setminus \emptyset \to \mathbb{R}$$

with

$$d(A, B) = \inf_{a \in A, b \in B} d_X(a, b)$$

Remark Metric Axioms

$$\begin{aligned} (i) \quad & d(x, y) = d(y, x) \\ (ii) \quad & d(x, y) \geq 0 \text{ and } d(x, x) = 0 \\ (iii) \quad & d(x, y) = 0 \Rightarrow x = y \\ (iv) \quad & d(x, y) \leq d(x, z) + d(z, y) \end{aligned}$$

- Symmetry

$$d(A, B) = \inf_{a \in A, b \in B} d_X(a, b) = d(B, A)$$

- Non-Negativeness

$$d(A, B) = \inf_{a \in A, b \in B} \underbrace{d_X(a, b)}_{\geq 0}$$

  Since d_X is a metric and is guaranteed to be non-negative, the infimum must be also non-negative

- Positiveness For a similar reason as in the [[#ˆfb2589| point to set distance]], we can easily construct cases, such that $d \equiv 0$ for distinct sets A, B. More precisely, any sets A, B with $A \cap B \neq \emptyset$ have

$$d(A, B) = 0$$

  The following example illustrates the problem. Let $A = [0, 2]$ and $B = [1, 3]$, then note that $A \cap B = [1, 2]$. For the infimum in $\inf_{a \in A, b \in B} d(a, b)$ we can choose any $a, b \in A \cap B$ and get

$$d(A, B) = 0$$

with $A \neq B$

- Triangle Inequality The following example will illustrate the issue with this property. Let again $A = [0, 2]$, $B = [1, 5]$, $C = [4, 6]$ and d_X the $p = 1$ Minkowski distance, we have

$$\begin{aligned} d(A, C) &= \inf_{a \in A, c \in C} d_X(a, c) &&= d_X(2, 4) = 2 \\ d(A, B) &= \inf_{a \in A, b \in B} d_X(a, b) &&= d_X(1, 1) = 0 \\ d(B, C) &= \inf_{b \in B, c \in C} d_X(b, c) &&= d_X(4, 4) = 0 \end{aligned}$$

Then we have for the triangle inequality

$$\begin{aligned} d(A, C) &\leq d(A, B) &&+ d(B, C) \\ 2 &\not\leq 0 &&+ 0 \end{aligned}$$

General Hausdorff Function

Definition The Hausdorff Family

Let - (X, d_X) be a metric space - $\mathcal{P}(X)$ the power set of X

The Hausdorff function

$$h : \mathcal{P}(X) \setminus \emptyset \times \mathcal{P}(X) \setminus \emptyset \to \overline{\mathbb{R}}$$

is given $\forall A, B \in \mathcal{P}(X) \setminus \emptyset$

$$\begin{aligned} h(A, B) &= \max \{ \sup_{a \in A} d_X(a, B) &&, \sup_{b \in B} d_X(b, A) \} \\ &= \max \{ \sup_{a \in A} \inf_{b \in B} d_X(a, b) &&, \sup_{b \in B} \inf_{a \in A} d_X(a, b) \} \end{aligned}$$

Remark Axiom Metrics

$$\begin{aligned} &(i) && d(x, y) = d(y, x) \\ &(ii) && d(x, y) \geq 0 \text{ and } d(x, x) = 0 \\ &(iii) && d(x, y) = 0 \Rightarrow x = y \\ &(iv) && d(x, y) \leq d(x, z) + d(z, y) \end{aligned}$$

- Symmetry The Hausdorff function satisfies the symmetry property, since

$$d(A, B) = \max\{ \sup_{a \in A} d_X(a, B), \sup_{b \in B} d_X(b, A) \} = d(B, A)$$

The suprema inside the maximum operator are interchangeable and the maximum value will be returned regardless of the order the arguments are passed into the distance function.

- Non-Negativeness Since the Hausdorff function relies on the underlying metric space X and its distance function, the values are guaranteed to be at least 0
- Remaining Ideas It is not a distance function, since it maps to the extended real numbers $\overline{\mathbb{R}}$ with

  $$h(\{0\}, \mathbb{R}) = +\infty$$

  It is not a pseudo metric, because it does not satisfy property

  $$(iii) \qquad d(x, y) = 0 \Longrightarrow x = y$$

  because let $A = (0, 1)$ and $B = [0, 1]$, we have

  $$h(A, B) = 0$$

Therefore, we have to require compact subsets as domain of the distance function, i.e.

$$h : \mathcal{B}(X) \setminus \emptyset \times \mathcal{B}(X) \setminus \emptyset \to \mathbb{R}$$

Conclusion

The point-set distance and the set-to-set distance don't seem to be stable options for a proper learning process, since they fail to hold many of the metric properties.

The Hausdorff metric on the other hand seems counter-intuitive for the objective of the learning process. Given the Hyperbox-LVQ method, we could summarize the objective for an optimal prototype placement and size as follows: 1. a box prototype should just be as large as needed to cover a cloud of data points 2. overlaps between box prototypes should be as small as possible, which should result from the 1. 3. the position of the center of the box should allow point 1. and 2. to hold

Since the Hausdorff distance describes the maximum distance between a closest point of a set, and the farthest point of the other set with respect to the closest point, it would incentivise the collapse of the boxes over time, such that they collapse on data points, because then the distance would be minimal, i.e. 0.

Since the set, i.e. the hyperbox, is described over its center and its dimensions, it seems natural to focus on designing a distance function involving these two parameters.

Ideas

Since there are two distinct objectives in optimizing prototypes, an alternating learning scheme seems an appropriate method. 1. Placement of the centers could follow the box-distance d_B (needs to be checked, only idea for now)

$$d_B(\mathbf{w}, \mathbf{x}) = d(\mathbf{w}, \mathbf{x}) \cdot \sigma(\text{Overlap-Factor}(\mathbf{w}, \mathbf{x}))$$

where the overlap factor is the magnitude of overlap between the two boxes of $\mathbf{w}, \mathbf{x}$.

$$\text{Overlap-Factor}(\mathbf{w}, \mathbf{x}) = \begin{cases} > 0 & \text{if boxes overlap} \\ = 0 & \text{if boxes touch} \\ < 0 & \text{else} \end{cases}$$

2. Optimizing the size of the boxes. After placing the boxes, the algorithm should check if there are data points $\mathbf{x}$ which are not yet covered by a box of same class. The suggestion follows - **If** there are points which are not covered by any box of the same class yet => the size of the box should be increased. - **If** there are all prototypes are covered by a box of the same class - **if** the farthest data point inside the box is exactly on the border of the prototype => the box should neither decrease or increase. - **If** the farthest data point inside the box is inside the box of the prototype => the box should decrease