

Spatial, Temporal and Multimedia Databases
 WS 2015/16

Übungsblatt 1: Distanzmaße, Dreiecksungleichung

Besprechung: 19./26.10.2015

Aufgabe 1-1 *Distanzmaße*

Distanzmaße können wir nach ihren Eigenschaften in folgende Kategorien einteilen:

| $d : S \times S \rightarrow \mathbb{R}$ $x, y, z \in S :$ | reflexiv $x = y \Rightarrow d(x, y) = 0$ | symmetrisch $d(x, y) = d(y, x)$ | strikt $d(x, y) = 0 \Rightarrow x = y$ | Dreiecks-Ungleichung $d(x, z) \leq d(x, y) + d(y, z)$ |
|--|---|------------------------------------|---|--|
| Unähnlichkeitsfunktion | × | | | |
| Distanzfunktion | × | × | | |
| Semi-Metrik | × | × | × | |
| Pseudo-Metrik | × | × | | × |
| Metrik | × | × | × | × |

Entscheiden Sie für die folgenden Funktionen $d(\mathbb{R}^n, \mathbb{R}^n)$ jeweils, ob es sich um ein Distanzmaß handelt, und wenn ja, in welche Kategorie es fällt.

(a) $d(x, y) = \sum_{i=1}^n (x_i - y_i)$

(b) $d(x, y) = \sum_{i=1}^n (x_i - y_i)^2$

(c) $d(x, y) = \sqrt{\sum_{i=1}^{n-1} (x_i - y_i)^2}$

(d) $d(x, y) = \sum_{i=1}^n \begin{cases} 1 & \text{falls } x_i = y_i \\ 0 & \text{falls } x_i \neq y_i \end{cases}$

(e) $d(x, y) = \sum_{i=1}^n \begin{cases} 1 & \text{falls } x_i \neq y_i \\ 0 & \text{falls } x_i = y_i \end{cases}$

Anmerkung: Die Begriffe *Semi-Metrik*, *Pseudo-Metrik* und *Distanzfunktion* werden in der Literatur zum Teil unterschiedlich verwendet. In der Mathematik ist z.B. “Distanzfunktion” ein Synonym für “Metrik”. In der Informatik gibt es jedoch viele “Distanzfunktionen” für die die Dreiecksungleichung nicht gilt. Für eine vollwertige Metrik verwenden wir daher bewusst den Begriff “Metrik”, wenn die Dreiecksungleichung benötigt wird. Wenn Sie in der Literatur den Begriff “distance” lesen, kann i.A. nicht davon ausgegangen werden, dass eine Metrik vorliegt.

Aufgabe 1-2 *Dreiecksungleichung*

Die Dreiecksungleichung wird häufig als Approximation genutzt, um unter Umständen teure Distanzberechnungen zu sparen. Veranschaulichen Sie sich diese elementargeometrische Eigenschaft in untenstehender Zeichnung: Wie kann die Länge der gestrichelten Linie approximiert werden? Welche weiteren Strecken (mit bereits eingezeichneten Punkten) würden die Approximation verbessern?

