

Maschinellen Lernens

Aufgabenblatt 05

Prof. Dr. David Spieler
Hochschule München

14. Oktober 2019

Aufgabe 1 (KNN) Gegeben seien die Trainingsdaten Tabelle 1 und der zu klassifizierende Punkt $\mathbf{x} = (0, 0, 0)^T$.

	$\mathbf{x}_1^{(i)}$	$\mathbf{x}_2^{(i)}$	$\mathbf{x}_3^{(i)}$	$y^{(i)}$
$\mathbf{x}^{(1)}$	0	3	0	0
$\mathbf{x}^{(2)}$	2	0	0	0
$\mathbf{x}^{(3)}$	0	1	3	0
$\mathbf{x}^{(4)}$	0	1	2	1
$\mathbf{x}^{(5)}$	-1	0	1	1
$\mathbf{x}^{(6)}$	1	1	1	0

Tabelle 1: Trainingsdaten für KNN.

1. Berechnen Sie die euklidische Distanz zwischen \mathbf{x} und jedem der sechs Trainingsdatenpunkt.
2. Was ist die Vorhersage y für \mathbf{x} wenn $K = 1$?
3. Was ist die Vorhersage y für \mathbf{x} wenn $K = 3$?
4. Wenn die Entscheidungsgrenze des optimalen Bayes Klassifikators sehr stark nichtlinear wäre, wäre ein hoher oder niedriger Wert für K angebracht?

Aufgabe 2 (Praxisbeispiel KNN, Rastersuche und Kreuzvalidierung)
In dieser Aufgabe lernen Sie den Iris Datensatz kennen, trainieren eine Reihe von KNN-Modellen mit variierender Nachbarschaftsgröße und setzen dabei Kreuzvalidierung ein.

1. Laden Sie den Iris-Datensatz mit Hilfe `sklearn.datasets.load_iris` in die Variablen \mathbf{X} (Features) und \mathbf{y} (Zielgröße, Target).
2. Was sind die einzelnen Features und was sind die Target-Klassen?

3. Schreiben Sie eine Schleife, in welcher Sie über die Werte für K von 1 bis 100 iterieren (Rastersuche). Im Schleifenkörper erstellen Sie jeweils ein `sklearn.neighbors.KNeighborsClassifier` Modell und berechnen mit Hilfe von `sklearn.model_selection.cross_val_score` die Genauigkeit mit Hilfe von 5-facher Kreuzvalidierung. Nehmen Sie hierfür den Mittelwert. Speichern Sie diese Werte für jede Belegung von K in eine Liste.
4. Plotten Sie mit Hilfe von `matplotlib` die scores auf die K -Werte.
5. Für welche(n) Wert(e) von K ist die Genauigkeit maximal?
6. Interpretieren Sie den Plot hinsichtlich möglicher Über- und Unteranpassung.