UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR ROBOTIK
UND KOGNITIVE SYSTEME

Künstliche Intelligenz
Sommersemester 2022
30. Mai 2022

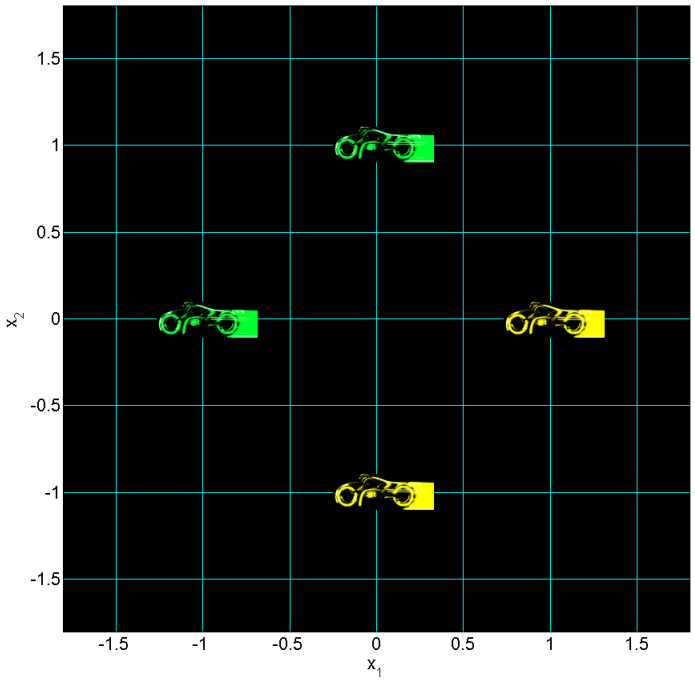
Niclas Bockelmann
bockelmann@rob.uni-luebeck.de

Übungsblatt 7

Abgabe bis zum 14. Juni 2022, 10 Uhr, im Moodle

Aufgabe 1: PerzepTRON (10 Punkte)

Auf dem *Raster* befinden sich zwei rivalisierende Gangs: die *Heaven's Angels* (grüne Bikes) und die *Sheriffidos* (gelbe Bikes). Das *MCP* möchte nun wissen, welche Bereiche von welcher Gang kontrolliert werden. Dazu bittet es *PerzepTRON* um Hilfe. Dieser entdeckt zur Zeit genau vier Bikes auf dem *Raster*.



PerzepTRON klassifiziert die grünen Bikes mit 1 und die gelben Bikes mit -1. Seine Klassifikationsfunktion lautet:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{falls } \mathbf{w}^T\mathbf{x} \geq 0 \\ -1 & \text{sonst} \end{cases}$$

Beachten Sie bitte die Erläuterungen zu Notation und Berechnungsvorschriften auf der nächsten Seite. Sie unterscheiden sich teilweise von denen aus der Vorlesung.

a) Notieren Sie die Beobachtungspaare (x_1, x_2) und die zugehörigen Sollausgaben in einer Tabelle. Wie groß ist R? (2 Punkte)

b) Lernen Sie die Perzeptron-Klassifikation mittels *pattern-by-pattern learning*. Für die Lernrate gelte $\eta = 0.5$, der initiale Gewichtsvektor sei $\mathbf{w} = (0, -1)^T$. (3 Punkte)

c) Lernen Sie die Perzeptron-Klassifikation mittels *batch learning*. Für die Lernrate gelte $\eta = 2$, der initiale Gewichtsvektor sei $\mathbf{w} = (0, -1)^T$. (3 Punkte)

d) Ermitteln Sie für beide Verfahren die Klassifikationsgerade. Zeichnen Sie die entsprechenden Geraden ein. (2 Punkte)

Geben Sie bei allen Aufgaben den Rechenweg mit an!

Notation

R	Anzahl Trainingsdatenpunkte
$\mathbf{x}_i$	Trainingsdatenpunkt i, $i = 1, 2, \ldots, R$
s_i	Sollausgabe für Trainingsdatenpunkt i, $i = 1, 2, \ldots, R$
$\mathbf{w}$	Gewichtsvektor
y_i	Ausgabe des Perzeptrons für Trainingsdatenpunkt i, $i = 1, 2, \ldots, R$
η	Lernschrittweite
d_i	Klassifikationsfehler (Abweichung) für Trainingsdatenpunkt i, $i = 1, 2, \ldots, R$
d	Gesamtklassifikationsfehler

Berechnungsvorschriften

$$y_i = f(\mathbf{x}_i) = \begin{cases} 1 & \text{falls } \mathbf{w}^T \mathbf{x}_i \geq 0 \\ -1 & \text{sonst} \end{cases}$$

$$d_i = (s_i - y_i)^2$$

$$d = \sum_{i=1}^{R} d_i = \sum_{i=1}^{R} (s_i - y_i)^2$$

Pattern-by-pattern learning

1. Setze $\mathbf{w}$ zufällig
2. Für $i = 1, 2, \ldots, R$:
   - Bestimme d_i
   - Falls $d_i > 0$, aktualisiere $\mathbf{w} = \mathbf{w} + \eta \cdot (s_i - y_i) \cdot \mathbf{x}_i$
3. Falls mind. eine Aktualisierung von $\mathbf{w}$, wiederhole Schritt 2 (ggf. zusätzliches Abbruchkriterium)

Es werden in einem Iterationsschritt immer alle $\mathbf{x}_i$ durchlaufen, $\mathbf{w}$ wird dabei kontinuierlich aktualisiert. Wurde $\mathbf{w}$ beispielsweise bei einem $\mathbf{x}_k$ aktualisiert, so wird dieser neue Gewichtsvektor $\mathbf{w}$ auch schon bei $\mathbf{x}_{k+1}$ angewendet.

Batch learning

1. Setze $\mathbf{w}$ zufällig
2. - Bestimme d
   - Falls $d > 0$, aktualisiere $\mathbf{w} = \mathbf{w} + \frac{\eta}{R} \cdot \sum_{i=1}^{R} ((s_i - y_i) \cdot \mathbf{x}_i)$
3. Falls Aktualisierung von $\mathbf{w}$, wiederhole Schritt 2 (ggf. zusätzliches Abbruchkriterium)

Klassifikationsgerade

Im zweidimensionalen Fall ist die Klassifikationsgerade $x_2 = -\frac{w_1}{w_2} \cdot x_1$.

Aufgabe 2: California-Dreamin' (10 Punkte)

Bei Ihrem wohlverdienten Urlaub in Kalifornien nach der anstrengenden KI-Klausur haben Sie sich in ein wunderschönes Haus verliebt. Sie haben Glück und es steht sogar zum Verkauf! Allerdings ist weit und breit kein Preisschild zu finden und das Maklerbüro hat auch schon geschlossen. Da Sie nicht länger warten können, überlegen Sie sich, wie Sie wohl am besten den Preis schätzen können. Online stoßen Sie auf einen Datensatz mit mehr als 20.000 Einträgen und jeweils 8 Featuren die über Blockgruppen (kleinste geografische Einheit mit ungefähr 600 bis 3000 Menschen) erhoben wurden:

1. median income in block group
2. median house age in block group
3. average number of rooms per household
4. average number of bedrooms per household
5. block group population
6. average number of household members
7. block group latitude
8. block group longitude

Zusätzlich stehen Ihnen auch die ensprechenden Hauswerte einer Blockgruppe als Median zur Verfügung. Dabei erinneren Sie sich an die Regressionsthematik in der KI-Vorlesung.

Laden Sie sich das juypter notebook aus dem Moodle herunter. Für diese Aufgabe benötigen Sie die Packages *scikit-learn*, *pandas*, *matplotlib* und *seaborn*.

a) Plotten Sie in einer Figure zunächst jedes Feature gegenüber der Hauswerte. Die Wahl der Plot-Methoden bleibt Ihnen überlassen. Speichern Sie die Figure als png-Datei. (1 Punkt)

b) Führen Sie einen 80-20 Train-Test-Split auf den Daten durch. Nutzen Sie entsprechende *scikit-learn* Funktionen. (1 Punkt)

c) Skalieren Sie die Trainings- und Testdaten. Es dürfen *scikit-learn* Funktion verwendet werden. Achten Sie drauf, auf welchen Daten die Skalierung basiert. (1 Punkt)

d) Trainieren Sie ein lineares Regressionsmodell `LinearRegression`. Das Training soll nur auf einem Feature Ihrer Wahl stattfinden. Begründen Sie die Wahl Ihres Features im Code. Geben Sie den mean squared error (MSE) und R2-Score auf den Testdaten aus. (2 Punkte)

e) Plotten Sie die gelernte Regressionsgerade in Ihrem entsprechenden Featureraum. Die Wahl der Plot-Methode bleibt Ihnen überlassen. Speichern Sie die Figure als png-Datei. (1 Punkt)

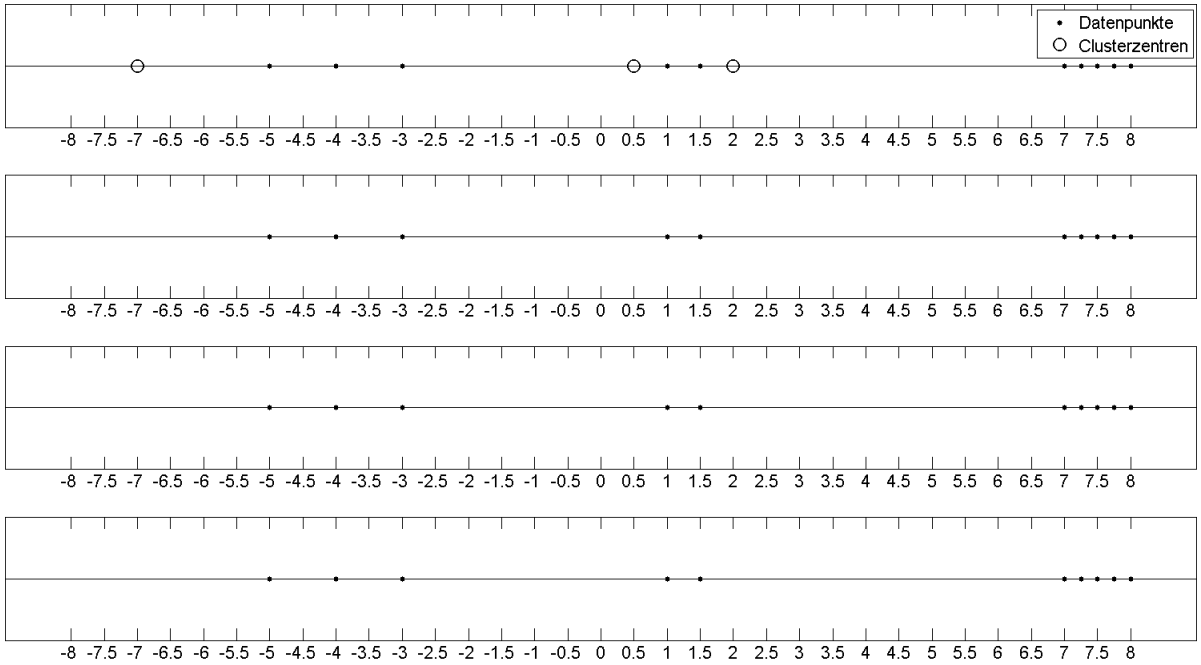
f) Trainieren Sie nun ein lineares Regressionsmodell auf allen Featuren. Geben Sie den MSE und R2-Score aus. (2 Punkte)

g) Trainieren Sie ein kNN-Regressor `KNeighborsRegressor` auf allen Featuren. Wählen Sie den Nachbarschaftsparameter entsprechend aus. Geben Sie den MSE und R2-Score aus. Diskutieren Sie die Ergebnisse Ihrer drei genutzten Methoden kurz im Code. (2 Punkte)

Geben Sie Ihre Implementierung und alle Figures als zip-Datei im Moodle ab.

Aufgabe 3: Sesamstraßen-Clustering (10 Punkte)

Die Bewohner der Sesamstraße wollen das *k-means clustering* ausprobieren. Graf Zahl hat dazu ein paar seiner geliebten Zahlen auf dem Zahlenstrahl als Datenpunkte markiert, Ernie hat daraufhin willkürlich ein paar Clusterzentren eingezeichnet und Bert darf jetzt die ganze, restliche Arbeit machen. Und Sie helfen ihm dabei!

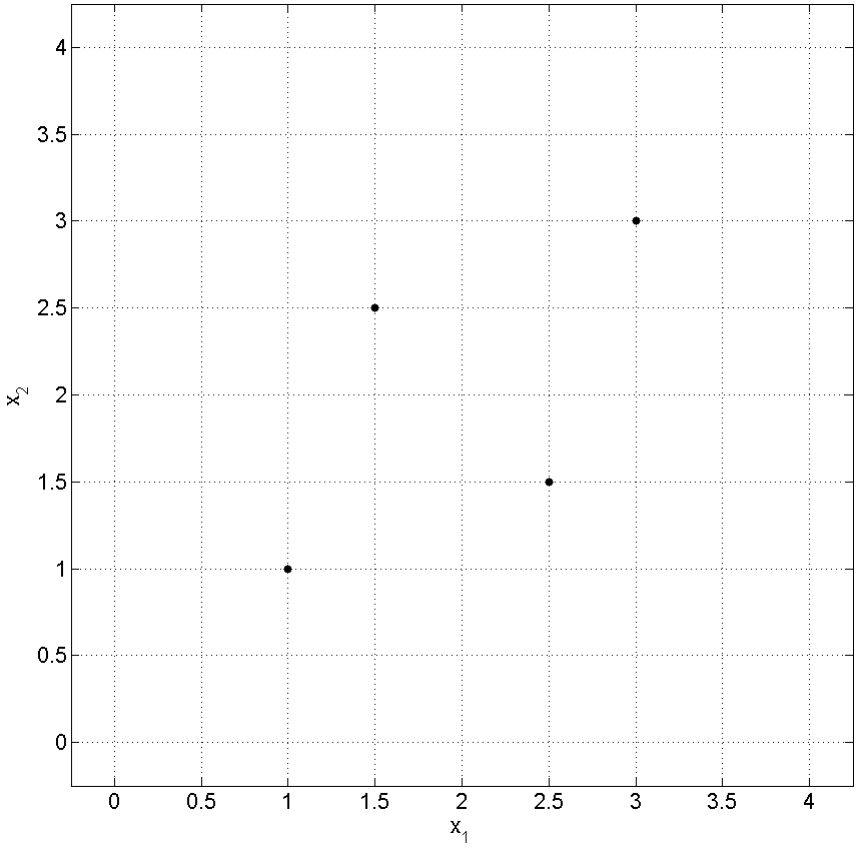


Berechnen Sie für jeden Iterationsschritt des *k-means clusterings* die Grenzen zwischen den derzeitigen Clustern sowie die neuen Schwerpunkte der Cluster und zeichnen Sie beides ein. Stoppen Sie, sobald sich die Schwerpunkte der Cluster nicht mehr verändern. (10 Punkte)

Geben Sie bei allen Iterationsschritten den Rechenweg mit an!

Aufgabe 4: Prinzipiell Ceine Ahnung (10 Punkte)

Nach sechs Übungszetteln schwindet die Kreativität. Zu dieser Aufgabe gibt es keine originelle Hintergrundgeschichte... Wenden Sie die Hauptkomponentenanalyse an!



a) Geben Sie die Matrix $\mathbf{X}$ der Datenpunkte an. Berechnen Sie die Matrix Ψ der mittelwertbefreiten Datenpunkte. (2 Punkte)

b) Berechnen Sie die Kovarianzmatrix Φ. (2 Punkte)

c) Berechnen Sie die Eigenwerte von Φ und die zu den Eigenwerten gehörigen Eigenvektoren. Normieren Sie die Eigenvektoren dabei auf Länge 1. (3 Punkte)

d) Die Daten sollen nun derart transformiert werden, dass nur noch die erste Hauptkomponente Anwendung findet. Bestimmen Sie die Transformationsmatrix $\hat{\mathbf{U}}$. Geben Sie die transformierten Datenpunkte an. Zeichnen Sie die erste Hauptkomponente und die transformierten Datenpunkte ein. (3 Punkte)

Geben Sie bei allen Teilaufgaben den Rechenweg mit an!