Übungsblatt 9

Aufgabe 1 – 20%

1. Gegeben sei der Datensatz A bestehend aus den folgenden Punkten:

   $A = \{(1,1),(2,1),(4,3),(5,4),(6,5),(7,6),(9,8),(10,7)\}$

   Wenden Sie (manuell) den k-means-Algorithmus mit $k = 2$ an. Initialisieren sie den Algorithmus mit den Cluster-Zentren $C_1 = (2,1)$ und $C_2 = (10,7)$. Führen sie zwei Iterationen des Algorithmus durch, zeigen sie dabei jeweils, welcher Datenpunkt zu welchem Cluster gehört, und wie die neuen Cluster-Zentren nach jeder Iteration aussehen.

2. In einem Folgeversuch wiederholen Sie den Versuch auf dem gleichen Datenset mit $k = 3$ und den initialen Cluster-Zentren $C_1 = (1,1)$, $C_2 = (5,4)$, und $C_3 = (9,8)$.

Aufgabe 2 – 30%

Für den k-Means-Algorithmus ist ein Cluster gegeben als C_j. Zeigen Sie, dass das Cluster-Zentrum μ_j für den Cluster C_j genau dann folgende Gleichung minimiert

$$\min_{\mu_j} \sum_{x^{(i)} \in C_j} \left\| x^{(i)} - \mu_j \right\|^2$$

wenn μ_j der Centroid des Clusters ist:

$$\mu_j = \frac{1}{|C_j|} \sum_{x^{(i)} \in C_j} x^{(i)}.$$

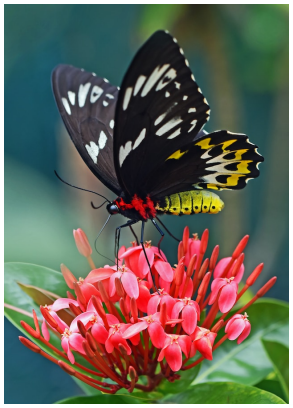
Führen Sie den Beweis wenn möglich mathematisch. Alternativ- erklären sie genau, was die Intuition hinter dem Beweis ist.

Aufgabe 3 – 50%

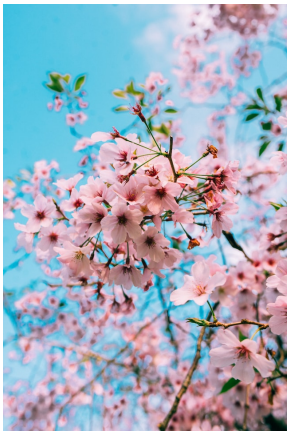
In dieser Übung sollen Sie sich mit den zwei populären Methoden k-means Clustering und Principal Component Analysis (PCA) auseinandersetzen. Sie sollen diese Methoden nutzen, um die drei Bilder `butterfly.jpg`, `flower.jpg` und `nasa.jpg` zu komprimieren. Ihnen steht ein Jupyter Notebook `kmeans_pca.ipynb` zur Verfügung, das ein grobes Gerüst für die Aufgabe bereitstellt. Sie sind nicht verpflichtet das Notebook zu nutzen, es dient nur zur Orientierung und Hilfestellung.

Part 1 (25%)

Führen sie k-means Clustering auf den drei Bildern aus. Behandeln Sie jeden Pixel im Bild als einen Datenpunkt im 3-dimensionalen Raum (Dimensionen: Rot, Grün, Blau). Ersetzen Sie jeden Pixel mit dem nächstgelegenen Cluster-Zentrum. Entscheiden Sie selbst, welche Anzahl an Clustern einen guten Kompromiss zwischen Kompressionsrate und Qualität der Bilder darstellt. Visualisieren Sie ihre Ergebnisse. Sie müssen die Kompressionsrate nicht berechnen, begründen Sie jedoch kurz, wieso diese Methode die Menge an Daten reduziert, die pro Bild gespeichert werden muss.



(a) Photo by David Clode on Unsplash



(b) Photo by Y S on Unsplash

(c) Photo by NASA on Unsplash

Abbildung 1: `butterfly.jpg`, `flower.jpg` und `nasa.jpg`

Part 2 (25%)

Nutzen Sie nun PCA um die drei Bilder zu komprimieren. Transformieren Sie hierzu jedes $n \times m$ Bild mit 3 Farbkanälen zu einer $n \times 3m$ 2D-Matrix, bevor Sie die PCA durchführen. Nehmen sie für die PCA an, dass die Datenmatrix n Datenpunkte mit jeweils $3m$ Dimensionen darstellt. Bestimmen Sie zunächst die l Hauptkomponenten mit höchster Varianz und berechnen Sie die dimensionsreduzierte Darstellung der Datenmatrizen. Führen Sie dann die inverse Transformation zurück zu einem Bild mit $n \times m$ Pixeln und 3 Farbkanälen durch (die Dokumentation des `PCA` Objekt in sklearn kann hier helfen). Wieviele Hauptkomponenten sind nötig, dass das menschliche Auge keinen Unterschied mehr sehen kann (entscheiden Sie selbt, das Experiment muss nicht quantitativ durchgeführt werden). Visualisieren Sie ihre Ergebnisse. Begründen Sie kurz, wieso die Methode die Menge an Daten reduziert, die pro Bild gespeichert werden muss.

Bitte lösen Sie Aufgaben bis zum **15. Januar 2024**. Ihren Python-Code, incl. Visualisierungen können Sie in Form eines Jupyter Notebooks oder PDFs hochladen.