

Computergestützte Mathematik zur linearen Algebra – 14. Übungsblatt

Aufgabe 50: (Vererbung)

- (a) Erstellen Sie eine PYTHON-Klasse `Person`, die zur Initialisierung folgende Parameter übergeben bekommt: *vorname*, *nachname*, *geschlecht*. Implementieren Sie die Klassenmethode `zusammenfassung()`, welche die Daten der Person per `print` folgendermaßen ausgibt:
Name: Max Mustermann (m)
- (b) Erstellen Sie eine Klasse `Dozent`, die von `Person` erbt und zusätzlich das Attribut *bueronummer* übergeben bekommt. Aus Vorname und Name soll das Attribut *emailadresse* erstellt werden (der Aufbau soll so sein wie Ihre Uni-Mailadresse). Überschreiben Sie `zusammenfassung()` so, dass zusätzlich auch Mailadresse und Büronummer ausgegeben werden (verwenden Sie hierbei `zusammenfassung()` von `Person`):
Name: Felix Klein (m)
Mail: Felix.Klein@hhu.de
Büro: 25.22.U3.42
- (c) Erstellen Sie eine Klasse `Student`, die von `Person` erbt und zusätzlich die Attribute *matrikelnummer* und *studienfach* übergeben bekommt. Aus Vorname und Name soll das Attribut *emailadresse* erstellt werden (analog zu b)). Die Methode `zusammenfassung()` soll die Daten auf folgende Weise ausgeben:
Donald Duck hat die Matrikelnummer 123456 und studiert Mathematik
Außerdem soll `Student` die Methode `lernen()` bekommen, die folgenden Text ausgibt (mit dem passenden Namen und einer ganzzahligen Zufallszahl *X* aus [2, 7]):
Donald lernt jetzt *X* Stunden
- (d) Erstellen Sie einige Personen, Dozenten und Studenten und testen Sie die Methoden.

Aufgabe 51: (Klassen zur Funktionsauswertung / Vererbung)

- (a) Erstellen Sie eine Klasse `Funktion_mit_Ableitung_und_Integral`, welche von `FuncWithDerivatives` (siehe Vorlesungsseite) erbt und analog zur Vorlesung zusätzlich folgende Approximationsmethode für das Integral in der Methode `integriere(a,b)` implementiert:

$$\int_a^b f(x) \approx h \cdot \sum_{k=1}^n f\left(a - \frac{h}{2} + k \cdot h\right), \text{ mit } h = \frac{b-a}{n}$$

n soll standardmäßig auf 100 gesetzt sein. Ihre Klasse soll OHNE das NumPy-Paket funktionieren.

- (b) Sei $f_\alpha(x) = 0,1x \sin(\alpha x) - 4,9$. Implementieren Sie, analog zur Vorlesung, jeweils einmal eine Klasse von $f_\alpha(x)$ mit den numerischen Ableitungen und dem Integral und einmal ein Klasse mit den analytischen Ableitungen und dem exakten Integral. Ihre Klassen sollen den Parameter α übergeben bekommen.
- (c) Testen Sie die beiden verschiedenen Klassen (beispielsweise mit Plots) mit $\alpha = 3$ und $\alpha = 200$.

Aufgabe 52: (Code verstehen / Slicing)

Gegeben ist folgende Methode (siehe Vorlesungsseite), die eine Matrix A (als numpy-Array) übergeben bekommt:

```
def unbekannt(A):
    m,n=A.shape
    for k in range(m):
        for l in range(int(n/2)):
            tmp=A[k,n-1-1]
            A[k,n-1-1]=A[k,l]
            A[k,l]=tmp
```

- (a) Was tut der Code?
- (b) Schreiben Sie eine Methode `passender_name()`, die mittels Slices die innere for-Schleife überflüssig macht.
- (c) Schreiben Sie eine Methode `noch_ein_passender_name()`, die komplett ohne for-Schleifen auskommt.

Aufgabe 53: (Matrix generieren / Singulärwertzerlegung, alte Klausuraufgabe)

Befehle: `np.linalg.svd`, `np.diag(v, k=0)`

Gegeben sei die Matrix

$$B = B(N) = \frac{1}{N^2} \begin{bmatrix} -2 & 1 & & & \\ 1 & \ddots & \ddots & & \\ & \ddots & \ddots & 1 & \\ & & & 1 & -2 \end{bmatrix} \in \mathbb{R}^{N \times N}$$

- (a) Schreiben Sie eine Funktion `B = MatrixB(N)`, die diese Matrix in Abhängigkeit von $N \in \mathbb{N}_{\neq 0}$ aufstellt. Geben Sie eine Warnung aus und `None` zurück, falls $N \leq 0$.
- (b) Approximieren Sie $B(10)$, indem Sie die 5 größten Singulärwerte und die passenden Singulärvektoren einer Singulärwertzerlegung von $B(10)$ verwenden.
- (c) Wie sieht die Bestapproximation von $B(20)$ vom Rang 10 aus, d.h. bestimmen Sie eine Matrix C , s.d.

$$\|B(20) - C\|_2 = \min_{\substack{D \in \mathbb{R}^{20 \times 20}, \\ \text{rank}(D)=10}} \|B(20) - D\|_2.$$