

Übungsblatt Ana

Computational and Data Science BSc
FS 2023

Zusatz - Lösungen

Analysis und Lineare Algebra 2

1. Aussagen über die Berechnung von Integralen mit Python/Numpy

Welche der folgenden Aussagen sind wahr und welche falsch?	wahr	falsch
a) Mit Hilfe dem Befehl <code>trapz</code> können <i>bestimmte Integrale</i> berechnet werden.	●	○
b) Mit Hilfe dem Befehl <code>trapz</code> können <i>unbestimmte Integrale</i> berechnet werden.	○	●
c) Der Befehl <code>trapz</code> berechnet ein <i>bestimmtes Integral</i> durch Anwenden der NEWTON-LEIBNIZ-Formel.	○	●
d) Der Befehl <code>trapz</code> berechnet ein <i>bestimmtes Integral</i> näherungsweise durch die <i>Flächen</i> von <i>Trapezen</i> .	●	○
e) Um den Befehl <code>trapz</code> anwenden zu können, benötigt man in jedem Fall den <i>Funktionsterm</i> des <i>Integranden</i> .	○	●
f) Um den Befehl <code>trapz</code> anwenden zu können, benötigt man in jedem Fall Daten des <i>Integranden</i> (Wertepaare <i>Argument</i> und <i>Funktionswert</i>).	●	○

2. Integrale mit Python/Numpy berechnen

Wir berechnen jeweils das *Integral* mit Hilfe des Befehls `trapz` in Python/Numpy auf 3 *Dezimalstellen* genau. Dazu implementieren wir den folgenden Code, den wir für jede Teilaufgabe modifizieren.

```
# Python initialisieren:
import matplotlib.pyplot as plt;
import numpy as np;
# Parameter:
x_0=...; x_E=...; n=...; N=...; lw=3; fig=...;
# Funktionen:
def f(x): y=...; return y;
# Daten:
for k in range(0,n):
    x_data=np.linspace(x_0,x_E,N);
    f_data=f(x_data);
    # Integration:
    I=np.trapz(f_data,x_data);
    print('I =',I);
    N=2*N;
```

```
# Ausgabe:
print('Integral: I =',f"{I:.3}");
# Plot:
fh=plt.figure(fig);
plt.plot(x_data,f_data,linewidth=lw);
plt.xlabel(r'$x$'); plt.ylabel(r'$y$');
plt.grid('on'); plt.axis('image');
```

a) Wir modifizieren den Code.

```
# Parameter:
x_0=0; x_E=np.pi; n=7; N=10; lw=3; fig=1;
# Funktionen:
def f(x): y=np.sin(x); return y;
```

Gemäss Output erhalten wir für das *Integral* die *Näherung*

$$\underline{\underline{I}} = \int_0^{\pi} \sin(x) dx \approx \underline{\underline{2.00}}. \quad (1)$$

b) Wir modifizieren den Code.

```
# Parameter:
x_0=2; x_E=5; n=6; N=10; lw=3; fig=fig+1;
# Funktionen:
def f(x): y=(1+x)/(1-x); return y;
```

Gemäss Output erhalten wir für das *Integral* die *Näherung*

$$\underline{\underline{I}} = \int_2^5 \frac{1+x}{1-x} dx \approx \underline{\underline{-5.77}}. \quad (2)$$

c) Wir modifizieren den Code.

```
# Parameter:
x_0=-2; x_E=0; n=6; N=10; lw=3; fig=fig+1;
# Funktionen:
def f(x): y=3**x; return y;
```

Gemäss Output erhalten wir für das *Integral* die *Näherung*

$$\underline{\underline{I}} = \int_{-2}^0 3^x dx \approx \underline{\underline{0.809}}. \quad (3)$$

d) Wir modifizieren den Code.

```
# Parameter:
x_0=2; x_E=100; n=13; N=10; lw=3; fig=fig+1;
# Funktionen:
def f(x): y=np.sin(x)/(1+3*x); return y;
```

Gemäss Output erhalten wir für das *Integral* die *Näherung*

$$\underline{\underline{I}} = \int_2^{100} \frac{\sin(x)}{1+3x} dx \approx \underline{\underline{-0.0171}}. \quad (4)$$

e) Wir modifizieren den Code.

```
# Parameter:
x_0=0.01; x_E=1; n=7; N=10; lw=3; fig=fig+1;
# Funktionen:
def f(x): y=np.log(x)/np.log(2); return y;
```

Gemäss Output erhalten wir für das *Integral* die *Näherung*

$$\underline{\underline{I}} = \int_{0.01}^1 \log_2(x) dx \approx \underline{\underline{-1.36.}} \quad (5)$$

f) Wir modifizieren den Code.

```
# Parameter:
x_0=2; x_E=3; n=5; N=10; lw=3; fig=fig+1;
# Funktionen:
def f(x): y=np.log(10)/np.log(x); return y;
```

Gemäss Output erhalten wir für das *Integral* die *Näherung*

$$\underline{\underline{I}} = \int_2^3 \log_x(10) dx \approx \underline{\underline{2.58.}} \quad (6)$$

3. Aussagen über eine Integration mit Python/Numpy

Wir betrachten den folgenden Code für Python/Numpy.

```
# Python initialisieren:
import matplotlib.pyplot as pl;
import numpy as np;
# Parameter:
x_0=-2; x_E=2; N=3; lw=3; fig=1;
# Funktionen:
def f(x): y=x**2; return y;
# Daten:
x_data=np.linspace(x_0,x_E,N);
y_data=f(x_data);
# Berechnungen:
I=np.trapz(y_data,x_data);
# Plot:
fh=pl.figure(fig);
pl.plot(x_data,y_data,linewidth=lw);
pl.xlabel(r'$x$'); pl.ylabel(r'$y$');
pl.grid('on'); pl.axis('image');
```

Welche der folgenden Aussagen sind wahr und welche falsch?	wahr	falsch
a) Die Variable <code>y_data</code> ist ein <i>Array</i> mit 3 Werten.	<input checked="" type="radio"/>	<input type="radio"/>
b) Die Variable <code>I</code> hat den Wert 8.	<input checked="" type="radio"/>	<input type="radio"/>
c) Erhöht man den Wert der Variablen <code>N</code> , dann nähert sich der Wert der Variable <code>I</code> immer mehr der Zahl 5.	<input type="radio"/>	<input checked="" type="radio"/>
d) Der Python/Numpy-Befehl <code>trapz</code> berechnet das <i>bestimmte Integral</i> $\int_{-2}^2 x^2 dx$ näherungsweise mit Hilfe von Rechtecken.	<input type="radio"/>	<input checked="" type="radio"/>
e) Mit Hilfe des Python/Numpy-Befehls <code>trapz</code> könnte auch das <i>bestimmte Integral</i> $\int_{-1}^1 x^{-1} dx$ in guter Näherung berechnet werden.	<input type="radio"/>	<input checked="" type="radio"/>