Numerische Flächenberechnung

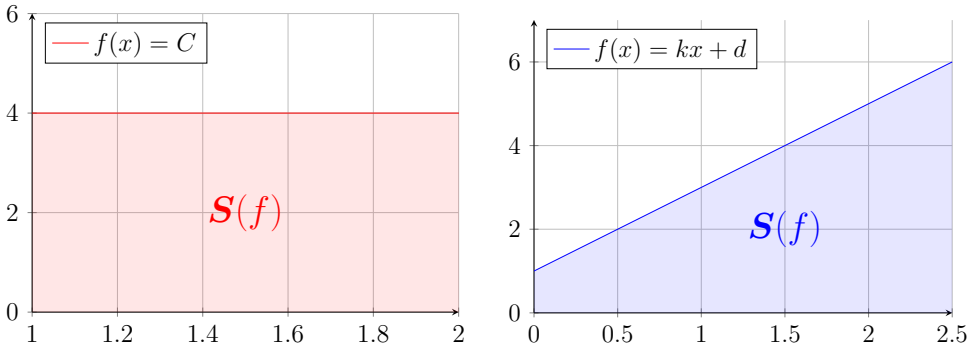
Miriam Schönauer, MSc

Unterlagen für Lehrende

23. April 2025

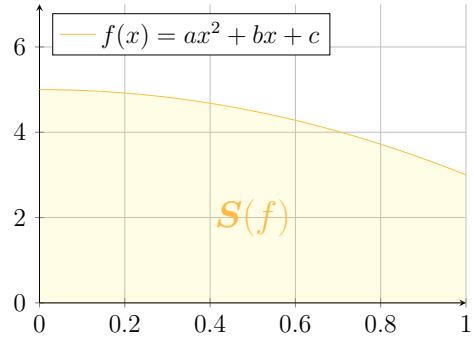
Grundsätzlich: Wir beschäftigen uns mit der Fläche unter einer Funktionskurve $S(f)$. Dabei geht es uns vor allem um die Berechnung von Näherungswerten $S_h(f)$ solcher Flächen. Ausgehend von der Polynominterpolation werden Formeln für solche Näherungswerte hergeleitet.
Technologieeinsatz mit Python:
Link zu Replit: `https://replit.com/`
Nützliche Python-Pakete: `numpy`, `matplotlib.pyplot`
Programm in Stichworten:
Beispiele aus der Antike, Exakte Flächenberechnung, Interpolatorische Quadraturformeln (Rechteck-, Mittelpunkt-, Trapez- und Simpsonregel), Beispiel Runge Funktion, Summierte Quadraturformeln (Rechteck-, Mittelpunkt-, Trapez- und Simpsonregel)
Dauer: 2 Unterrichtseinheiten; Voraussetzung: Flächenberechnung, Polynom(interpolation); Niveau: ★★

Die Berechnung von Flächen unter Funktionskurven ist eine Fragestellung, die Mathematiker bereits in der Antike beschäftigt hat. Beispielsweise berechnete Archimedes die Fläche unter einer Parabel mithilfe eines Dreiecks, oder es gelang den Babyloniern, die Position von Jupiter mithilfe der Fläche unter einer Zeit-Geschwindigkeitsfunktion zu bestimmen [1,2]. Da solche Flächen nützliche Anwendungen auch in heutiger Zeit haben, spielen deren Berechnung eine wichtige Rolle.
Aber wie lässt sich die Fläche $S(f)$ für eine Funktion f auf dem Intervall $[a, b]$ berechnen? Für einige Funktionen lassen sich sehr leicht Flächenformeln herleiten. Für eine konstante Funktionen $f(x) = C$ auf dem Intervall $[a, b]$ ergibt sich durch die Rechtecksflächenformel die Formel $S(f) = f(a) \cdot (b - a)$. Geht man auf affin lineare Funktionen $f(x) = kx + d$ über, so erhält man über die Trapezflächenformel die Formel $S(f) = (f(a) + f(b)) \frac{b-a}{2}$.

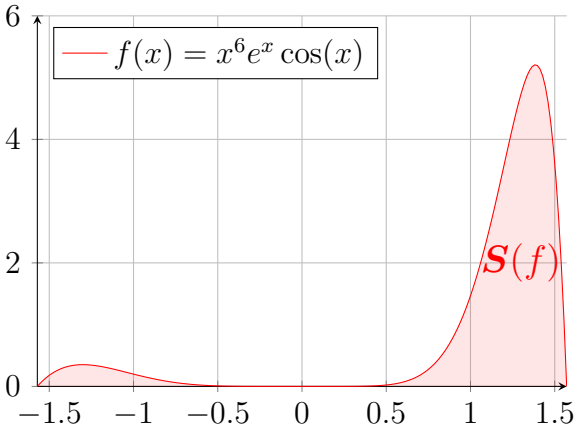


Für quadratische Funktionen $f(x) = ax^2 + bx + c$ kann mithilfe der Simpsonregel bzw. der Kepler'schen Fassregel die Fläche mithilfe der Formel $S(f) = \frac{(b-a)}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right)$ berechnet werden.

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:



All diese Flächenformeln $S(f)$ sind exakt. Generell lassen sich die Flächen unter Polynomen exakt berechnen. Für nicht polynomielle Funktionen wie etwa $f(x) = x^6 e^x \cos(x)$ gibt es ebenfalls Techniken (partielle Integration, Substitution), um die Fläche exakt zu berechnen.



So ergibt sich für die Fläche $S(f)$ auf dem Intervall $[-\pi/2, \pi/2]$ der Wert

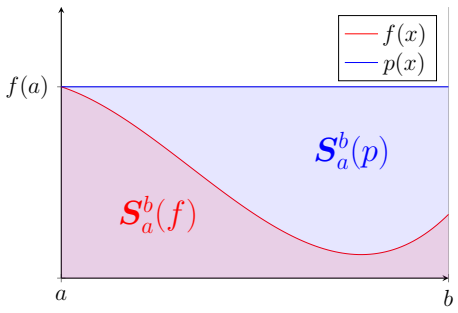
$$\boldsymbol{S}(f) = \frac{e^{\pi/2}(\pi^6 - 12\pi^5 + 60\pi^4 - 1440\pi^2 + 5760\pi - 5760)}{128} + \frac{e^{-\pi/2}(\pi^6 + 12\pi^5 + 60\pi^4 - 1440\pi^2 - 5760\pi - 5760)}{128}.$$

Allerdings kann mit diesem Wert nur näherungsweise gerechnet werden, da sowohl π als auch e irrationale Zahlen sind, deren unendlich lange, nicht-periodische Nachkommaentwicklung sich nicht auf dem Computer exakt berechnen lassen. Es gibt allerdings auch Funktionen wie zum Beispiel $f(x) = \sin(x^2)$, bei denen diese Standardtechniken gar kein Ergebnis liefern.
Da es nun in vielen Fällen dazu kommt, dass das Ergebnis einer solchen Flächenberechnung nur näherungsweise oder gar nicht berechnet werden kann, greift man auf numerische Methoden zurück. Diese Methoden liefern zwar ebenfalls nur Näherungswerte, sind aber in der Regel viel einfacher auszuwerten, und sie sind in der Lage, auch Werte für Funktionen zu berechnen, wo dies sonst nicht möglich ist.
Als Grundlage für unser numerisches Verfahren dient folgende Idee: Die Funktion, deren Fläche wir näherungsweise berechnen wollen, wird durch eine Funktion approximiert, deren Fläche wir exakt berechnen

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

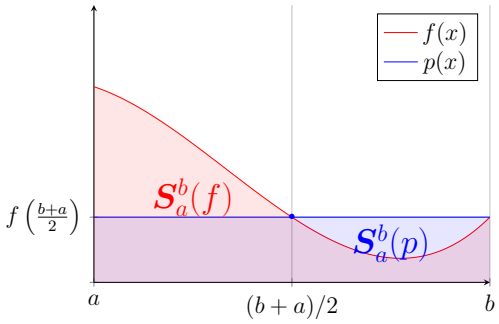
können. Um so eine Approximation der Funktion zu erhalten, verwenden wir die Polynominterpolation, da wir für Polynome die Fläche $S(f)$ exakt berechnen können. Für eine Polynominterpolation vom Grad 0 erhalten wir die Rechteck- und Mittelpunktregel:

Rechtecktregel



$$S(p) = (b-a) \cdot f(a)$$

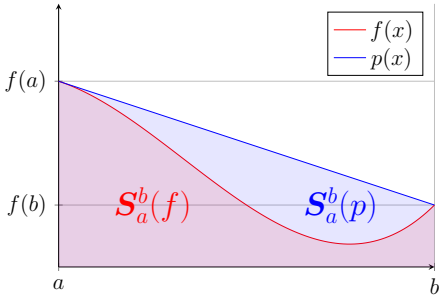
Mittelpunktregel



$$S(p) = (b-a) \cdot f\left(\frac{b+a}{2}\right)$$

Für die Interpolation vom Grad 1 die Trapezregel:

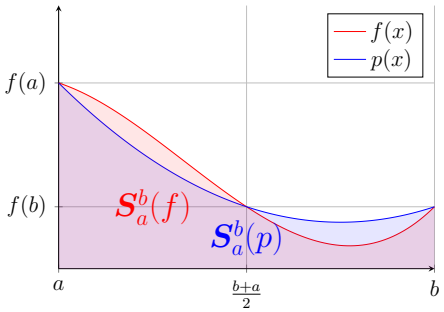
Trapezregel



$$S(p) = \frac{b-a}{2} \cdot (f(a) + f(b))$$

und für die vom Grad 2 die Simpsonregel:

Simpsonregel



$$S(p) = \frac{b-a}{6} \cdot \left(f(a) + 4f\left(\frac{b+a}{2}\right) + f(b)\right)$$

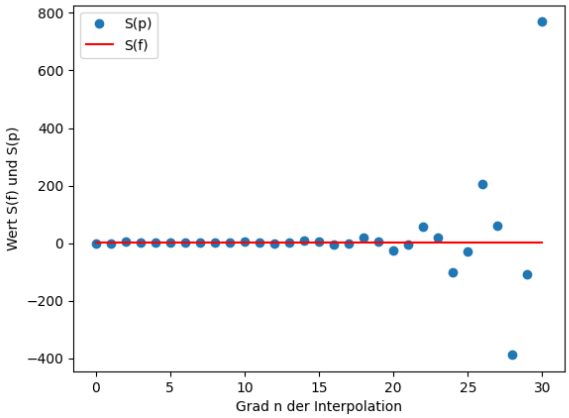
Diese Formeln werden auch interpolatorische Quadraturformeln genannt. Hierbei stammt der Name Quadratur von den Überlegungen der Griechen, allgemeine Flächen (Rechtecke, Polygone, Kreise) in ein flächengleiches Quadrat mit Lineal und Zirkel überzuführen (vgl. die Quadratur des Kreises). Mit diesen Quadraturformeln lassen sich nun die ersten Näherungswerte berechnen.

Beispiel: $f(x) = 2x(x-2)(x-3)+2$ auf dem Intervall $[1,3]$

1. Rechteckregel: $S(p) = (3-1)\cdot f(1) = 12$
2. Mittelpunktregel: $S(p) = (3-1)\cdot f(2) = 4$
3. Trapezregel: $S(p) = \frac{3-1}{2}\cdot(f(1)+f(3)) = 8$
4. Simpsonregel: $S(p) = \frac{3-1}{6}\cdot(f(1)+4f(2)+f(3)) = \frac{16}{3}$

Exakter Wert: $S(f) = \frac{16}{3} = 5.\dot{3}$

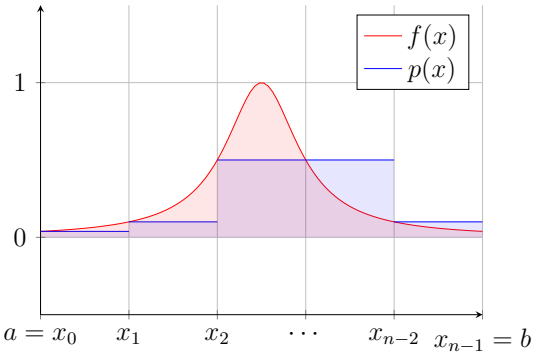
Eine nützliche Eigenschaft ist, dass die Mittelpunktregel Flächen von Funktionen bis zum Grad 1 und die Simpsonregel die Flächen von Funktionen bis zum Grad 3 exakt berechnen kann.
Es wäre nun zu vermuten, dass mit anwachsendem Grad der Polynominterpolation auch die Näherungswerte der Fläche besser werden, d.h. das Verfahren konvergiert. Leider kann man dies im Allgemeinen nicht erwarten. Anhand des Beispiels der Rungefunktion sieht man, dass sich hier die Näherungswerte immer weiter vom exakten Wert entfernen.



Wir müssen daher unseren Ansatz abändern, um zu gewährleisten, dass der Näherungswert immer gegen die tatsächliche Lösung konvergiert.
Da wir durch Erhöhung des Interpolationsgrades nicht den gewünschten Effekt erzielen, verfolgen wir eine andere Herangehensweise: Wir wenden unsere Polynominterpolation stückweise auf Teilintervallen an und lassen die Länge der Intervalle gegen 0 gehen. Dafür wählen wir äquidistante Stützstellen x_i in $[a,b]$ durch folgende Vorschrift aus: Es sei $h := \frac{b-a}{n-1}$ und $x_i := a + i\cdot h$ für $0 \leq i \leq n-1$. Durch die stückweise Anwendung der interpolatorischen Quadraturformeln ergeben sich die summierten Quadaraturformeln:

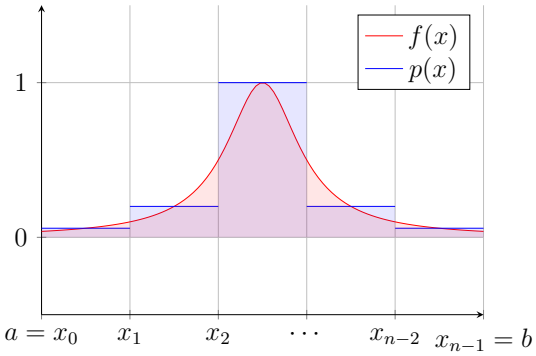
MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

Summierte Rechteckregel



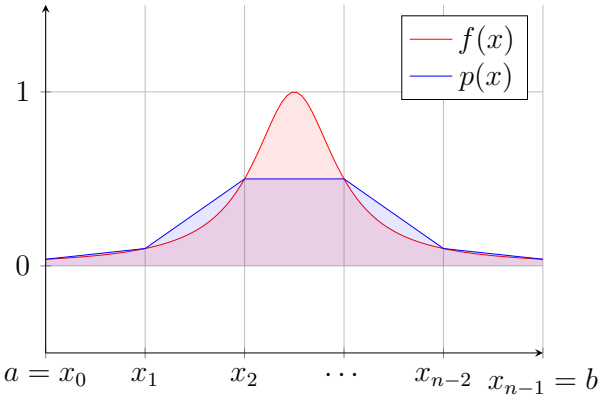
$$S_a^b(p) = h \cdot \sum_{i=0}^{n-2} f(x_i)$$

Summierte Mittelpunktregel



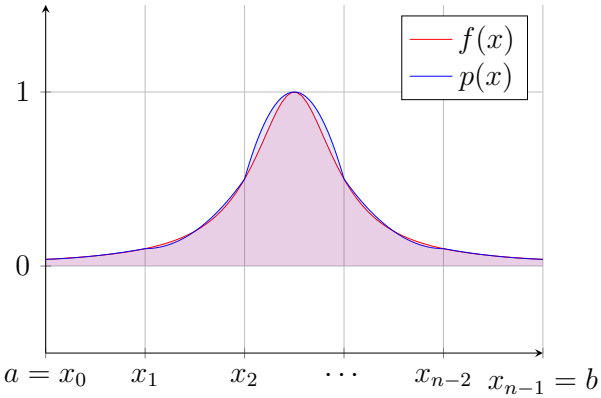
$$S_a^b(p) = h \cdot \sum_{i=0}^{n-2} f\left(\tfrac{x_i+x_{i+1}}{2}\right)$$

Summierte Trapezregel



$$S_a^b(p) = \tfrac{h}{2} \cdot \left(f(x_0) + 2\sum_{i=1}^{n-2} f(x_i) + f(x_{n-1})\right)$$

Summierte Simpsonregel

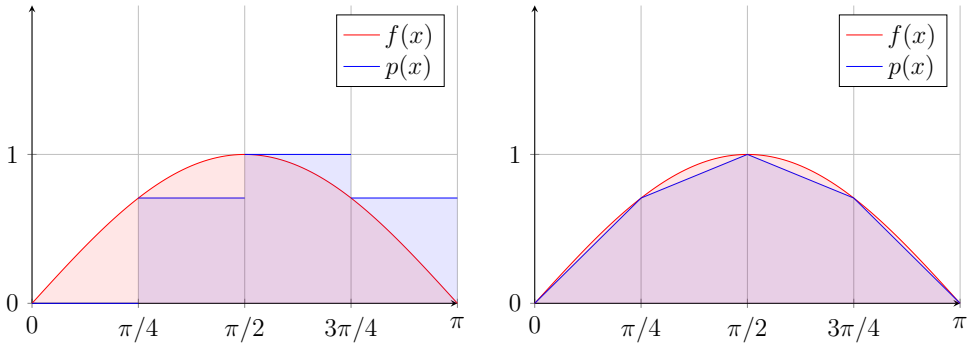


$$S_a^b(p) = \tfrac{h}{6} \cdot \left(f(x_0) + 2\sum_{i=1}^{n-2} f(x_i) + 4\sum_{i=0}^{n-2} f\left(\tfrac{x_i+x_{i+1}}{2}\right) + f(x_{n-1})\right)$$

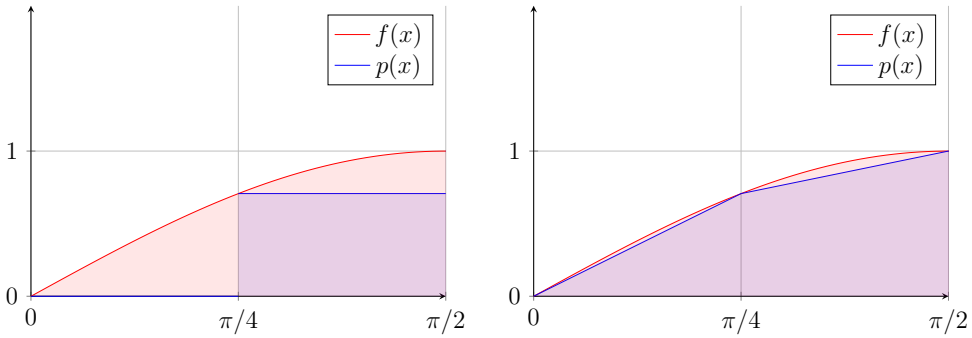
MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

→Rechenaufgaben

Diese summierten Quadraturregeln konvergieren nun gegen den exakten Wert.
Interessanterweise liefern die Rechteck- und Trapezregel dieselben Näherungswerte für $f(x) = sin(x)$ auf dem Intervall $[0, \pi]$, obwohl auf den ersten Blick die Trapezregel wie eine deutlich genauere Annäherung wirkt. Berechnet man die Fläche aber auf dem Intervall $[0, \pi/2]$, so liefert die Trapezregel bessere Werte.



Es liegt die Vermutung nahe, dass für symmetrische Funktionen diese beide Regeln zusammenfallen.

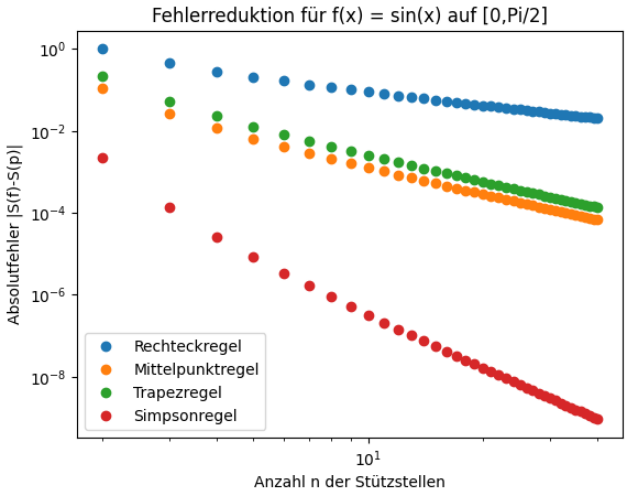


Man kann allerdings noch weniger voraussetzen:

$$
\begin{aligned}
h \cdot \sum_{i=0}^{n-2} f\left(x_i\right) &= \frac{h}{2} \cdot \left(f(x_0) + 2 \sum_{i=1}^{n-2} f(x_i) + f(x_{n-1}) \right) \\
\Leftrightarrow 2 \cdot \sum_{i=0}^{n-2} f\left(x_i\right) &= f(x_0) + 2 \sum_{i=1}^{n-2} f(x_i) + f(x_{n-1}) \\
\Leftrightarrow 2f(x_0) &= f(x_0) + f(x_{n-1}) \\
\Leftrightarrow f(x_0) &= f(x_{n-1})
\end{aligned}
$$

Es reicht also aus, dass die Auswertung der ersten Stützstelle mit der Auswertung der letzten übereinstimmt.
Ist das nicht der Fall, so ist die Trapezregel deutlich schneller als die Rechteckregel.

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:



Literatur:

1. Cusick, Larry W. "Archimedean Quadrature Redux." Mathematics Magazine, vol. 81, no. 2, 2008, pp. 83–95. JSTOR, http://www.jstor.org/stable/27643090. Accessed 16 Apr. 2025.
2. Ossendrijver, Mathieu. "Ancient Babylonian Astronomers Calculated Jupiter's Position from the Area under a Time-Velocity Graph." Science, vol. 351, no. 6272, 2016, pp. 482–84. JSTOR, http://www.jstor.org/stable/24741542. Accessed 16 Apr. 2025.
3. Rannacher, Rolf: Analysis 2: Differential- und Integralrechnung für Funktionen mehrerer reeller Veränderlichen, Heidelberg: Heidelberg University Publishing, 2018 (Lecture Notes). https://doi.org/10.178
4. Hämmerlin, Günther and Hoffmann, Karl-Heinz: Numerische Mathematik, Springer Berlin, Heidelberg, (1994) https://doi.org/10.1007/978-3-642-57894-6
5. Rannacher, Rolf: Numerik 0: Einführung in die Numerische Mathematik, Heidelberg: Heidelberg University Publishing, 2017 (Lecture Notes). https://doi.org/10.17885/heiup.206.281

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

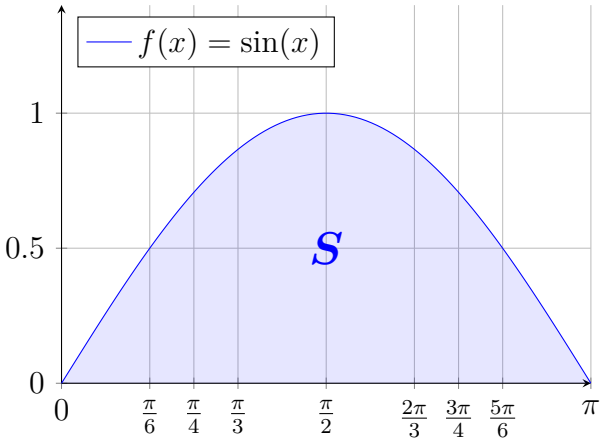
Numerische Flächenberechnung

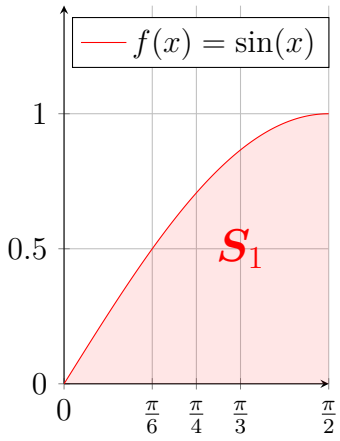
Miriam Schönauer, MSc

Handout

23. April 2025

$\boldsymbol{x}$	0	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$	$\frac{2\pi}{3}$	$\frac{3\pi}{4}$	$\frac{5\pi}{6}$	π
$\sin(\boldsymbol{x})$	0	$\frac{1}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{3}}{2}$	1	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{1}{2}$	0





Rechenaufgaben

Es sei $a = 0$, $b = \pi$ und $f(x) = \sin(x)$. Es gilt $\boldsymbol{S} := \boldsymbol{S}_a^b(f) = 2$.

1. Verwende die **summierte Rechteckregel** mit $h = \pi/2$, $\pi/3$, $\pi/6$, um die Fläche $\boldsymbol{S}$ näherungsweise zu berechnen.
   Fertige eine Skizze für $h = \pi/6$ an.

2. Verwende die **summierte Mittelpunktregel** mit $h = \pi/2$, $\pi/3$, um die Fläche $\boldsymbol{S}$ näherungsweise zu berechnen.
   Fertige eine Skizze für $h = \pi/3$ an.

3. Verwende die **summierte Trapezregel** mit $h = \pi/2$, $\pi/3$, $\pi/6$, um die Fläche $\boldsymbol{S}$ näherungsweise zu berechnen.
   Fertige eine Skizze für $h = \pi/6$ an.

4. Verwende die **summierte Simpsonregel** mit $h = \pi/2$, $\pi/3$, um die Fläche $\boldsymbol{S}$ näherungsweise zu berechnen.

5. Es sei nun $a = 0$, $b = \pi/2$ und $\boldsymbol{S}_1 := \boldsymbol{S}_a^b(f) = 1$. Verwende die **summierte Rechteckregel** und **summierte Trapezregel** mit $h = \pi/4$, $\pi/6$ um die Fläche $\boldsymbol{S}_1$ näherungsweise zu berechnen.
   Fällt dir im Vergleich zu **Aufgabe 1** und **Aufgabe 3** etwas auf?

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:
Bundesministerium Frauen, Wissenschaft und Forschung

Was ist eine Quadraturformel? Welche Quadraturformeln gibt es?

Was hast du heute sonst gelernt?

Das hat mir gefallen. Das hat mir nicht gefallen.

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

Handout

Miriam Schönauer, MSc
23. April 2025

Programmieraufgaben

1. Programmiere eine Funktion, die mithilfe der **summierten Rechteckregel** die Fläche $\boldsymbol{S}$ näherungsweise berechnet.

   Als Eingabewerte sollen die Intervallgrenzen $\boldsymbol{a}$ und $\boldsymbol{b}$, die Funktion $\boldsymbol{f}$ und die Anzahl der Stützstellen $\boldsymbol{n}$ verwendet werden, z.B. r_sum(a,b,f,n). Die Funktion soll als Ausgabewerte die Näherungslösung $\boldsymbol{S}_h$, die Liste der Stützstellen $\boldsymbol{x}_i$ und die Liste der Auswertungen $\boldsymbol{f}(\boldsymbol{x}_i)$ liefern.

   Berechne $\boldsymbol{S}_h$ für verschiedene $\boldsymbol{n}$ und vergleiche die Näherungswerte mit dem exakten Wert $\boldsymbol{S}$.

2. Programmiere eine Funktion, die mithilfe der **summierten Mittelpunktregel** die Fläche $\boldsymbol{S}$ näherungsweise berechnet.

   Als Eingabewerte sollen die Intervallgrenzen $\boldsymbol{a}$ und $\boldsymbol{b}$, die Funktion $\boldsymbol{f}$ und die Anzahl der Stützstellen $\boldsymbol{n}$ verwendet werden, z.B. m_sum(a,b,f,n). Die Funktion soll als Ausgabewerte die Näherungslösung $\boldsymbol{S}_h$, die Liste der Stützstellen $\boldsymbol{x}_i$ und die Liste der Auswertungen $\boldsymbol{f}(\boldsymbol{y}_i)$ liefern, wobei $\boldsymbol{y}_i = (\boldsymbol{x}_i + \boldsymbol{x}_{i+1})/2$ den Mittelpunkt zwischen $\boldsymbol{x}_i$ und $\boldsymbol{x}_{i+1}$ bezeichnet.

   Berechne $\boldsymbol{S}_h$ für verschiedene $\boldsymbol{n}$ und vergleiche die Näherungswerte mit dem exakten Wert $\boldsymbol{S}$.

3. Programmiere eine Funktion, die mithilfe der **summierten Trapezregel** die Fläche $\boldsymbol{S}$ näherungsweise berechnet.

   Als Eingabewerte sollen die Intervallgrenzen $\boldsymbol{a}$ und $\boldsymbol{b}$, die Funktion $\boldsymbol{f}$ und die Anzahl der Stützstellen $\boldsymbol{n}$ verwendet werden, z.B. t_sum(a,b,f,n). Die Funktion soll als Ausgabewerte die Näherungslösung $\boldsymbol{S}_h$, die Liste der Stützstellen $\boldsymbol{x}_i$ und die Liste der Auswertungen $\boldsymbol{f}(\boldsymbol{x}_i)$ liefern.

   Berechne $\boldsymbol{S}_h$ für verschiedene $\boldsymbol{n}$ und vergleiche die Näherungswerte mit dem exakten Wert $\boldsymbol{S}$.

4. Programmiere eine Funktion, die mithilfe der **summierten Simpsonregel** die Fläche $\boldsymbol{S}$ näherungsweise berechnet.

   Als Eingabewerte sollen die Intervallgrenzen $\boldsymbol{a}$ und $\boldsymbol{b}$, die Funktion $\boldsymbol{f}$ und die Anzahl der Stützstellen $\boldsymbol{n}$ verwendet werden, z.B. s_sum(a,b,f,n). Die Funktion soll als Ausgabewerte die Näherungslösung $\boldsymbol{S}_h$, die Liste **aller** Auswertungsstellen (d.h. die Stützstellen $\boldsymbol{x}_i$ und Mittelpunkte $\boldsymbol{y}_i$ zwischen $\boldsymbol{x}_i$ und $\boldsymbol{x}_{i+1}$) und die Liste der Auswertungen von $\boldsymbol{f}$ in **allen** Auswertungsstellen liefern.

   Berechne $\boldsymbol{S}_h$ für verschiedene $\boldsymbol{n}$ und vergleiche die Näherungswerte mit dem exakten Wert $\boldsymbol{S}$.

Hinweis

Mit den Näherungswerten aus den **Rechenaufgaben** und der Zeichenroutine in der Programmiervorlage kannst du deine Berechnungen kontrollieren.

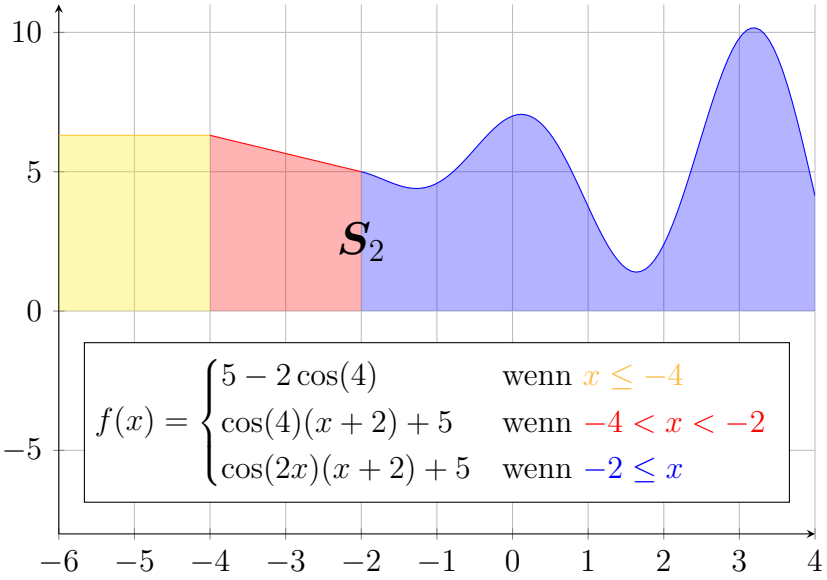
MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

OeAD Agentur für Bildung und Internationalisierung

Bundesministerium Frauen, Wissenschaft und Forschung

5. **Anwendung**

- Verwende die verschiedenen Regeln, um die Fläche unter der folgenden Funktion auf dem Intervall $[-6, 4]$ näherungsweise zu berechnen:



  Vergleiche deine Ergebnisse mit dem exakten Wert
  $\boldsymbol{S_2}$=$50 - 6.25\cos(4) + 0.25\cos(8) + 2\sin(8)$.
  Die Funktion $\boldsymbol{f}$ ist bereits implementiert in der Datei *Funktion.py*.

  Welche Regel funktioniert am besten? Welche am schlechtesten?

- Programmiere eine Funktion, die mithilfe der **summierten Trapezregel**, die Fläche $\boldsymbol{S}$ näherungsweise berechnet und deren Stützstellenanzahl sich in den Intervallen [-6,-2] und [-2,4] anpassen lässt. Als Eingabewerte sollen die Intervallgrenzen $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, die Funktion $\boldsymbol{f}$ und die Anzahl $\boldsymbol{n}$ der Stützstellen in [-6,-2] und $\boldsymbol{m}$ in [-2,4] verwendet werden, z.B. t_sum(a,b,c,f,n,m).

  Die Funktion soll als Ausgabewerte die Näherungslösung $\boldsymbol{S_h}$, die Liste der Stützstellen $\boldsymbol{x_i}$ (d.h. die Stützstellen in beiden Intervallen) und die Liste der Auswertungen $\boldsymbol{f(x_i)}$ liefern.

  Variiere die Stützstellenanzahl in den Intervallen und vergleiche die Ergebnisse mit den Ergebnissen der anderen Regeln. Was kannst du beobachten?

- Würde es hier Sinn machen, zwei verschiedene Regeln zu kombinieren? Falls ja, welche? Versuche so eine Kombination zu programmieren.

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:
Bundesministerium Frauen, Wissenschaft und Forschung

Python Cheat Sheet

Python Listenoperationen

```python
import numpy as np

# Erzeugt die Liste
meine_liste = [1, 4, 3, 2, -5, 7, 9]

# Gibt das Element an mit dem Index 1 zurueck
erstes_element = meine_liste[1]
print('erstes_element =',erstes_element)

# Aendert das erste Element zu 10
meine_liste[0] = 10
print('meine_liste =',meine_liste)

# Gibt die Anzahl der Elemente in der Liste zurueck
laenge = len(meine_liste)
print('laenge = ',laenge)

# Gibt die Elemente von Index 1 bis 3 zurueck
teil_liste = meine_liste[1:4]
print('teil_liste =',teil_liste)

# Gibt jedes zweite Element zurueck
jedes_zweite = meine_liste[::2]
print('jedes_zweite =',jedes_zweite)

# Entfernt alle Elemente aus der Liste
meine_liste.clear()
print('meine_liste =', meine_liste)

# Erzeugt eine Liste mit 4 Elementen zwischen 0 und 6
list1 = np.linspace(0,6,4)
print('list1 =',list1)
# Erzeugt eine Liste mit 5 Elementen zwischen -2 und 0
list2 = np.linspace(-2,0,5)
print('list2 =',list2)

# Verbindet list1 und list2
verknuepft = np.concatenate((list1, list2))
print('verknuepft = ', verknuepft)
```

```
erstes_element = 4
meine_liste = [10, 4, 3, 2, -5, 7, 9]
laenge =  7
teil_liste = [4, 3, 2]
jedes_zweite = [10, 3, -5, 9]
meine_liste = []
list1 = [0. 2. 4. 6.]
list2 = [-2.  -1.5 -1.  -0.5  0. ]
verknuepft =  [ 0.   2.   4.   6.  -2.  -1.5 -1.  -0.5  0. ]
```

MAJA wird gefördert vom Bundesministerium für Frauen, Wissenschaft und Forschung:

OeAD Agentur für Bildung und Internationalisierung

Bundesministerium Frauen, Wissenschaft und Forschung

Beispiel-Code

```
import numpy as np
import matplotlib.pyplot as plt

# Funktion zur Berechung der Werte 0, Pi/n, 2*Pi/n, ..., Pi*(n-1)/n, Pi
def pi(n):
    # Loesungsvektor der Vielfachen wird angelegt
    p_mult = np.zeros(n+1)
    # Berechnung der der Werte Pi*j/n
    for j in range (0,n+1):
        p_mult[j] = np.pi*j/n
    return p_mult

# Parameter auswaehlen
n = 5
# Werte berechnen
p_mult = pi(n)
# Werte anzeigen lassen
print(p_mult)
# Auswertungen von cos in den Werten Pi*j/n
print(np.cos(p_mult))

# Plotten die Auswertungen
plt.plot(p_mult, np.cos(p_mult), marker = 'o')
# Legende
plt.legend()
# Achsen-Label
plt.xlabel('x-Achse')
plt.ylabel('y-Achse')
# Plot anzeigen
plt.show()
```

```
[0.         0.62831853 1.25663706 1.88495559 2.51327412 3.14159265]
[ 1.          0.80901699  0.30901699 -0.30901699 -0.80901699 -1.        ]
```