Python – Übung 10

1 Schwingkreis

Die natürliche Schwingung eines gedämpften Schwingkreises kann mit folgender Funktion beschrieben werden

$$a(t) = a_0\,\mathrm{e}^{-t/\tau}\cos(2\pi f_\mathrm{n} t + \varphi)\ , \tag{1}$$

wobei $a_0 = 5$ die maximale Amplitude ist, $\tau = 0.2\,\mathrm{s}$ die Zeitkonstante ist, $f_\mathrm{n} = 10\,\mathrm{Hz}$ die natürliche Frequenz ist und $\varphi = -\pi/2$ die Phasenverschiebung ist.

☞ Schreiben Sie ein Programm, welches die beschriebene Schwingung ähnlich wie in Abb. 1 darstellt.

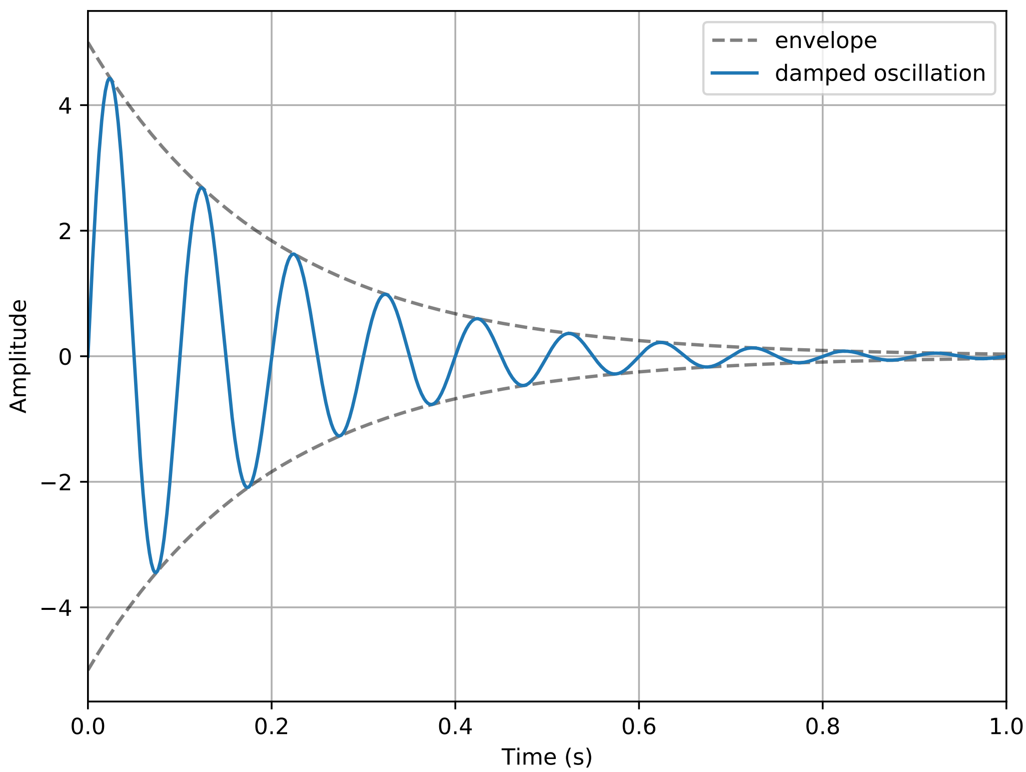

Abbildung 1: Gedämpfte Schwingung.

2 Methode der kleinsten Fehlerquadrate

Hier geht es wieder um eine Parabel. Die Punkte, die für die Koeffizienten-Identifikation zur Verfügung stehen, sind nicht mehr genau, sondern haben einen gewissen Messfehler. Dafür gibt es davon nicht nur drei, sondern ganze $N \gg 3$ Stück, d.h. das Gleichungssystem ist überbestimmt. Im Regelfall existiert kein Lösungsvektor $\boldsymbol{x}$, mit dem alle Gleichungen erfüllt werden können:

$$\boldsymbol{M}\boldsymbol{x} \approx \boldsymbol{y} \ , \tag{2}$$

$$\begin{bmatrix} x_1^2 & x_1 & 1 \\ \vdots & \vdots & \vdots \\ x_N^2 & x_N & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \approx \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \ . \tag{3}$$

Beim Einsetzen einer beliebigen Lösung wird es bei jeder Gleichung immer einen Rest geben:

$$\boldsymbol{r} = \boldsymbol{M}\boldsymbol{x} - \boldsymbol{y} \ . \tag{4}$$

In diesem Fall wird diejenige Lösung gesucht, bei der die "Summe der Quadrate der Fehler" minimal wird, d.h.

$$\boldsymbol{r}^T\boldsymbol{r} \to \text{Minimum} \ . \tag{5}$$

Die Lösung wurde schon von Gauss[1] vorgeschlagen:

$$\boldsymbol{x} = (\boldsymbol{M}^T\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{y} \ . \tag{6}$$

☞ Schreiben Sie ein Programm, welches die gemessenen Punkte (x_i, y_i) aus der mitgelieferten Datei **measurements.txt** einliest und die optimalen Koeffizienten (a, b, c) nach der obigen Methode bestimmt. Das Resultat soll ähnlich wie in Abb. 2 dargestellt werden.

Hinweise: Benutzen Sie `np.loadtxt()`[2] mit dem passenden `delimiter`-Argument, um die Daten aus der Textdatei einzulesen. Benutzen Sie `np.linalg.solve()` oder `np.linalg.lstsq()`[3] für die Berechnung von (6). Sie können mathematische Ausdrücke in Textfeldern und Labels mit TEX-Ausdrücken[4] schreiben.

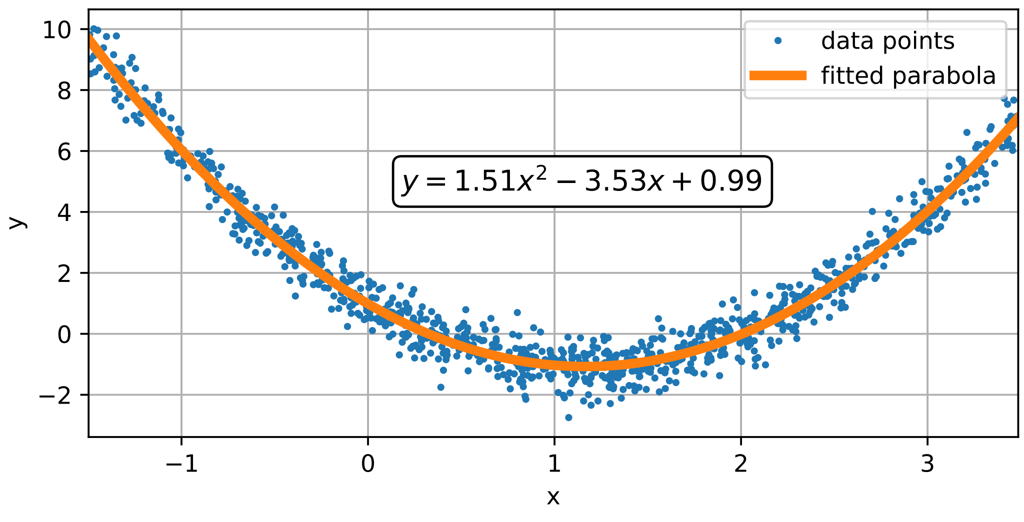

Abbildung 2: Gemessene Punkte und die optimale Parabel mit den geschätzten Koeffizienten.

[1] https://de.wikipedia.org/wiki/Methode_der_kleinsten_Quadrate
[2] https://docs.scipy.org/doc/numpy/reference/generated/numpy.loadtxt.html
[3] https://docs.scipy.org/doc/numpy/reference/generated/numpy.linalg.lstsq.html
[4] https://matplotlib.org/stable/tutorials/text/mathtext.html

3 Kreisfläche approximieren

Die Fläche eines Kreises (mit Radius r) soll mittels einem N-Eck approximiert werden. Wie in Abb. 3 dargestellt, werden dafür N Punkte, $\vec{P}_n$, im gleichen Abstand entlang des Kreises platziert.

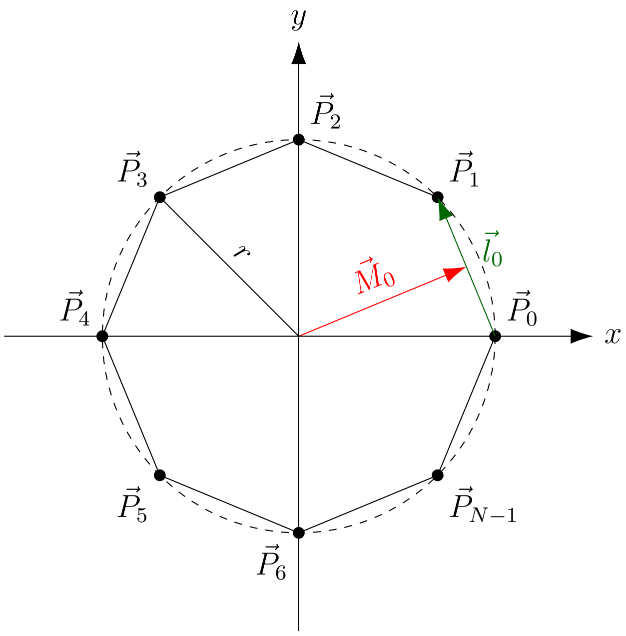

Abbildung 3: Approximation des Kreises mittels einem N-Eck.

Die Koordinaten eines Punktes ist also wie folgt definiert:

$$\vec{P}_n = \begin{bmatrix} x_n \\ y_n \end{bmatrix} = r \begin{bmatrix} \cos(2\pi n/N) \\ \sin(2\pi n/N) \end{bmatrix} . \tag{7}$$

Der Vektor $\vec{l}_n$, der zwischen zwei benachbarte Punkte liegt, ist als Differenz der beiden Ortvektoren definiert:

$$\vec{l}_n = \vec{P}_{n+1} - \vec{P}_n . \tag{8}$$

Der Vektor $\vec{M}_n$ zeigt auf den Mittelpunkt des Vektors $\vec{l}_n$, d.h.

$$\vec{M}_n = \frac{\vec{P}_{n+1} + \vec{P}_n}{2} . \tag{9}$$

Die Fläche des N-Ecks besteht aus N gleichschenkligen Dreiecken, deren Fläche das halbe Produkt aus Grundseite (l_n) mal Höhe (M_n) ist, d.h.

$$A_N = \frac{1}{2} \sum_{n=0}^{N-1} \left\| \vec{l}_n \right\| \cdot \left\| \vec{M}_n \right\| . \tag{10}$$

☞ Berechnen Sie die Fläche des N-Ecks wie oben beschrieben, ohne jegliche `for`-Schleifen zu benutzen, ausser um den Wert von N zu iterieren. Setzen Sie für N die Werte 2^k, $\forall\, k = 2, \ldots, 10$ ein. Plotten Sie dann die Resultate, ähnlich wie in Abb. 4 dargestellt, und speichern Sie das Diagramm im PDF- und PNG-Format ab.

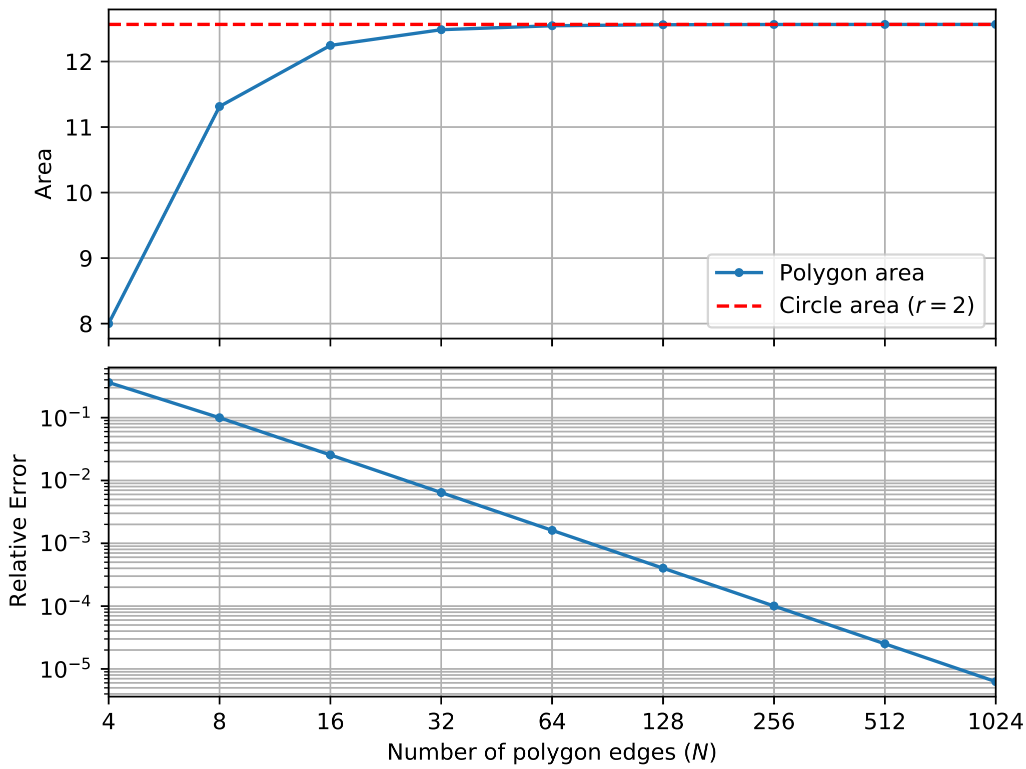

Abbildung 4: Approximation der Kreisfläche mittels einem N-Eck.

Hinweis: Benutzen Sie die Funktion[5] `plt.xticks()` oder die Methoden[6,7] `ax.set_xticks()` und `ax.set_xticklabels()` für die Nummerierung der x-Achse.

[5] `https://matplotlib.org/api/_as_gen/matplotlib.pyplot.xticks.html`
[6] `https://matplotlib.org/api/_as_gen/matplotlib.axes.Axes.set_xticks.html`
[7] `https://matplotlib.org/api/_as_gen/matplotlib.axes.Axes.set_xticklabels.html`