

Der unbekannte Heizöltank

Projekt zum Seminar Angewandte Mathematik

Prof. Dr. J. Hertle
jochen.hertle@hm.edu

Sommersemester 2021

Einführung

Ein zylinderförmiger Heizöltank befindet sich in der Erde vergraben, jedoch kennt man den Radius r und die Länge l des Zylinders nicht.

Es wurde bereits zweimal getankt und dabei vor und nach dem Tanken die Höhe h des Ölstands im Tank mit einem Stab gemessen:

h vorher (in m)	h nachher (in m)	getankt (in Liter)
0.3	1.0	4500
0.2	0.6	2600

Ziel soll es sein, mit verschiedenen Methoden die unbekannten Größen r und l zu bestimmen.

Aufgabenstellung

1. Erstellen Sie eine Volumenfunktion $V(r, l, h)$ für einen zylindrischen, waagrecht liegenden Tank. Plotten Sie die Funktion $h \mapsto V(r, l, h)$, wobei die Grafik interaktive Schieberegler für die zwei Variablen r, l haben soll.
2. Versuchen Sie zunächst, das Problem mit der Methode `solve()` zu lösen. Hierzu erstellen Sie ein Gleichungssystem aus zwei Gleichungen in den beiden Variablen r und l , wobei jede Gleichung einen der Tankvorgänge darstellt (d.h. für h die konkreten Werte aus der Tabelle enthält). Versuchen Sie zunächst, `solve()` auf das Gleichungssystem mit zwei Gleichungen anzuwenden. Was passiert? Lösen Sie sodann eine der Gleichungen mit Hilfe

von `solve()` nach der Variablen l auf. Die Lösung setzen Sie in die andere Gleichung ein (mit `substitute()`), so dass Sie eine Gleichung mit einer Variablen r erhalten. Diese Gleichung lösen Sie mit `find_root()` und bestimmen anschließend auch die Lösung für l . Dokumentieren Sie Ihre Vermutungen, warum `solve()` das Gleichungssystem nicht ohne Ihre Hilfe lösen kann.

3. Recherchieren Sie in der mathematischen Literatur, welche Methoden es für das sogenannte Fitting, d.h. das Anpassen von Funktionen an Datenpunkte gibt. Diese Methoden unterscheiden sich hinsichtlich der verwendeten Modellklasse von Funktionen, der Art der Fehlerfunktion und des Verfahrens, wie man die Fehlerfunktion minimiert. Klassifizieren Sie das vorliegende Problem hinsichtlich der benötigten Modellklasse und der grundsätzlich geeigneten Fitting Methoden.
4. Lösen Sie das Problem mit Hilfe der Funktion `find_fit()`.
5. Fitting-Methoden minimieren den Fehler, d.h. die Summe der Abweichungen zwischen Datenpunkten und einer Funktion. Erstellen Sie eine solche Fehlerfunktion, die die Summe der Quadrate der Abweichungen (*=Residuen*) zwischen den berechneten Volumendifferenzen (als Funktion der Variablen r und l) und den tatsächlich getankten Volumen repräsentiert. Minimieren Sie diese Fehlerfunktion mit Hilfe der Funktion `minimize(..., algorithm = 'simplex')`. Stichwort für die Literatur dazu: *Methode der kleinsten Quadrate, Least Squares*.
6. Fitting-Methoden können auch Lösungen bei überbestimmten Problemen finden. Haben Sie Daten zu mehr als zwei Tankvorgängen zur Verfügung, so ist es immer noch möglich, eine optimale Lösung zu finden. Erzeugen Sie eine Liste von plausible Daten für mehrere Tankvorgänge bei einem angenommenen Tank mit den Parametern $r = 0.7$ und $l = 5.0$. Diese Daten sollen aufgrund von Messungenauigkeiten fehlerbehaftet sein. Testen Sie die untersuchten Methoden mit diesen Daten.
7. Strukturieren Sie den Code mit Hilfe einer Klasse, die alle benötigten Methoden enthält.

Arbeitsauftrag

- Tauschen Sie im Team Ihre Kontaktdaten aus.
- Zerlegen Sie die Arbeit in einzelne Komponenten und Schritte.

- Dokumentieren Sie diese Zerlegung als Teil der Abgabe und berichtigen Sie die Dokumentation, wenn sich im Laufe der Arbeit Änderungen als nötig erweisen.
- Schätzen Sie Schwierigkeitsgrad und Aufwand der einzelnen Teile und finden Sie eine gerechte und praktikable Aufteilung auf die Mitglieder des Teams.
- Erstellen Sie einen Software Entwurf mit Klassen und Schnittstellen, so dass jede(r) Teilnehmer(in) unabhängig implementieren kann.
- Vereinbaren Sie mindestens ein besser zwei Treffen unter der Woche um den Fortgang der Arbeit und Änderungen am Design oder der Arbeitsteilung zu besprechen.
- Die Dokumentation ist wesentlicher Teil des Notebooks.
- Der Beitrag jedes Team-Mitglieds muss im Notebook klar ersichtlich sein.
- Abgabe des SageMath Notebooks über Moodle.
- Verspätete Abgaben werden nicht akzeptiert.
- Es darf pro Gruppe nur ein Notebook abgegeben werden.
- Bewertungskriterien sind: Lesbarkeit, Korrektheit, Vollständigkeit, Design, Python PEP8 Style-Konformität, Softwarearchitektur / Strukturierung, Engagement, Kreativität.
- Es ist untersagt Code aus anderen Quellen zu verwenden. Kopieren führt zu 0 Punkten, in schweren Fällen zum Nichtbestehen der Teilleistung.