

Übungsblatt 1

solution(s) due:

19. November 2020 um 23.59h via ecampus

Aufgabe 1.1: Die Datei

whData.dat

enthält Daten zu Körpergewicht (*weight*) und Körpergröße (*height*), die in einer Umfrage unter 39 Studierenden eines MSc Kurses erhoben wurden. Mit *python* / *numpy* können Sie diese Daten z.B. folgendermaßen einlesen

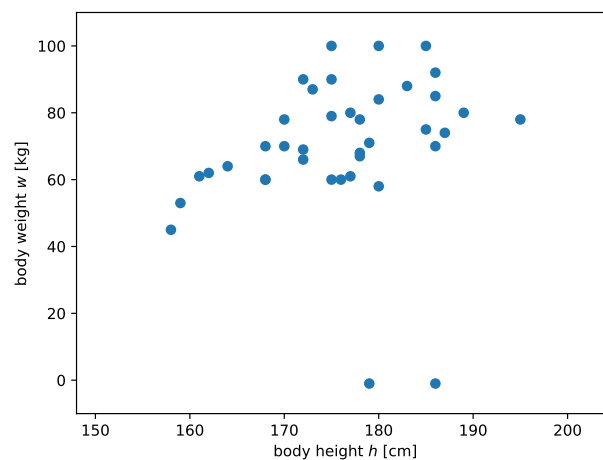
```
import numpy as np

name = ... # e.g. "./Data/whData.dat"
data = np.loadtxt(name, dtype=np.object, comments='#', delimiter=None)
```

Um die *w* und *h* Daten aus dem array *data* auszulesen, können Sie so vorgehen

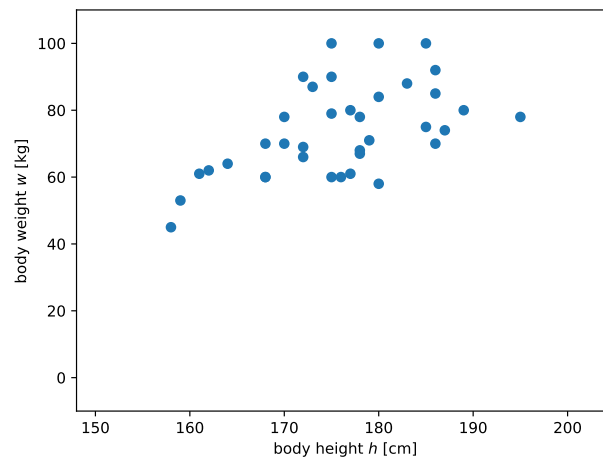
```
w = data[:,0].astype(float)
h = data[:,1].astype(float)
```

Hier ist Ihre Aufgabe: Nutzen sie *matplotlib*, um die Körpergewichte gegen die Körpergrößen zu plotten. Ihr Ergebnis sollte etwa so aussehen



Aufgabe 1.2: Bei der Umfrage haben zwei Studierende ihr Körpergewicht nicht angegeben. Dies gibt uns die Gelegenheit, mit *missing data* bzw. mit *Ausreißern* zu arbeiten.

Die Ausreißer in den Daten sind in der obigen Abbildung gut zu erkennen, ihre Gewichtswerte sind -1 . Schreiben Sie ein *python / numpy* Script, das solche Datenpunkte automatisch ausfiltert und plotten Sie die bereinigten Daten. Das Ergebnis sollte etwa so aussehen



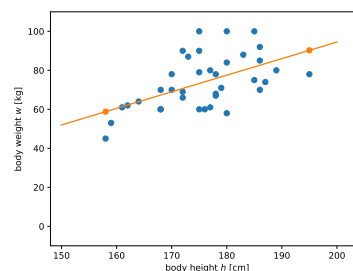
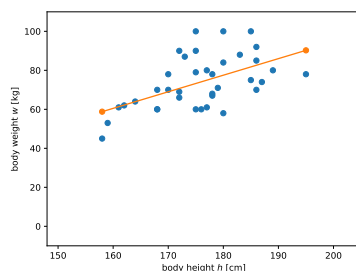
Aufgabe 1.3: Plotten Sie die bereinigten Daten erneut zusammen mit (in einer anderen Farbe) den beiden Punkten

$$\mathbf{x}_1 = \begin{bmatrix} 158.00 \\ 58.79 \end{bmatrix} \quad \text{und} \quad \mathbf{x}_2 = \begin{bmatrix} 195.00 \\ 90.25 \end{bmatrix}$$

und zeichnen Sie ein Liniensegment zwischen diesen Punkten. Nutzen Sie dann die Information, die Ihnen diese Punkte bieten, um die Parameter b und m der Funktion

$$f(h) = b + m \cdot h$$

zu bestimmen, die eine Gerade durch die Punkte \mathbf{x}_1 und \mathbf{x}_2 beschreibt. Plotten Sie diese Gerade. Ihre Ergebnisse könnten etwa so aussehen



Aufgabe 1.4: Berechnen Sie Mittelwert, Varianz und Standardabweichung der bereinigten Gewichts- und Größen-Daten. Nutzen Sie dazu z.B. die *numpy* Funktionen `mean()`, `var()` und `std()`.

Aufgabe 1.5: Auf der Webseite

data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases

finden Sie regelmäßige Updates zu weltweiten COVID-19 Infektionen. Laden Sie dort das File

`time_series_covid19_confirmed_global.csv`

herunter und schreiben Sie ein *python / numpy* Script, das diese Daten in den Speicher lädt und die Daten für Deutschland herausfiltert. Inspiration dazu, wie dies gehen könnte, finden Sie hier

C. Bauckhage, "[The Math of Epidemic Outbreaks and Spread \(Part 1\) Exponential Growth versus Logistic Growth](#)", researchgate.net, March 2020