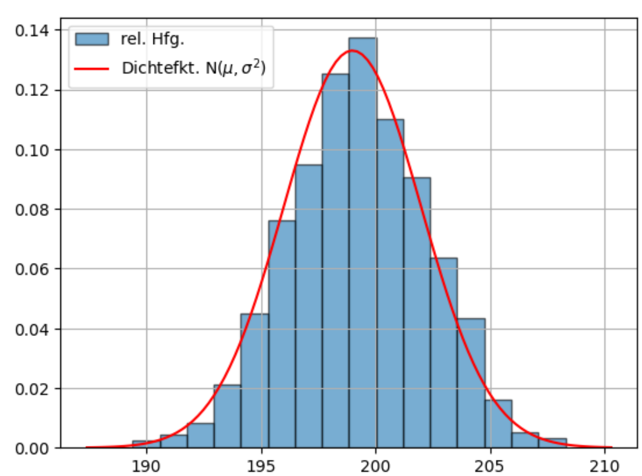
Labor 5

A1. Teepackungen, die von einer bestimmten Firma abgefüllt werden, sollten mit jeweils 200 g Inhalt abgefüllt werden. Die abgefüllte Menge Tee X in einer Packung ist normal verteilt $X \sim N(\mu, \sigma^2)$; die dafür zuständige Abfüllmaschine hat eine Standardabweichung von $\sigma = 3$ g und ist auf einen Erwartungswert $\mu = 199$ g eingestellt.
a) Anhand 1000 simulierten Daten, welche ist *im Mittel* die abgefüllte Menge Tee in einer Packung?
Hinweis: Man benutze `scipy.stats.norm.rvs(`$\mu, \sigma, 1000$`)` für die Generierung von Daten und danach `numpy.mean`.

```
#Beispiel Generieren von 1000 Daten
import matplotlib.pyplot as plt
from scipy.stats import norm
import numpy
mu=199
sigma=3
N=1000
Daten = norm.rvs(mu,sigma,N)
```

b) Mit welcher Wahrscheinlichkeit werden in einer Packung weniger als 195 g Tee abgefüllt? Mit welcher Wahrscheinlichkeit werden in einer Packung zwischen 195 g und 198 g Tee abgefüllt? Mit welcher Wahrscheinlichkeit werden in einer Packung mehr als 195 g Tee abgefüllt? Man vergleiche das Ergebnis mit den theoretischen Wahrscheinlichkeiten.
Hinweis: Für die theoretischen Wahrscheinlichkeiten benutze man `scipy.stats.norm.cdf(`x, μ, σ`)`.
Es gilt: $P(X \leq x) = F(x)$ und $P(X \in [a, b]) = F(b) - F(a)$ für die stetige ZG $X \sim N(\mu, \sigma^2)$, F ist die Verteilungsfunktion (`norm.cdf`) von X.
c) Die generierten Daten der Stichprobe sollen in 16 Klassen (Intervallen) eingeteilt. Man zähle mit `Hfg, Klasse=numpy.histogram(Daten, bins=16)` und gebe an (mit `print`) wie viele Daten in jeder Klasse sind. Man zeichne das entsprechende Histogramm der *relativen Häufigkeiten* mit
`matplotlib.pyplot.hist(Daten,bins=16,density=True,edgecolor="black",label="rel.  Hfg.")`
Auf demselben Bild zeichne man auch die Dichtefunktion der $N(\mu, \sigma^2)$ Verteilung ($\mu = 199, \sigma = 3$).
Hinweis: Man benutze `scipy.stats.norm.pdf(`x, μ, σ`))` und `plot`.



Aufgabe A1 - Histogramm und Dichtefunktion $N(\mu, \sigma^2)$

```
# Antwort A1 - (d) absolute Hfg. der Klassen anhand Simulationen
(  1) absolute Hfg.   3 der Klasse [ 189.4102,190.5905]
(  2) absolute Hfg.   5 der Klasse [ 190.5905,191.7709]
(  3) absolute Hfg.  10 der Klasse [ 191.7709,192.9512]
(  4) absolute Hfg.  25 der Klasse [ 192.9512,194.1315]
(  5) absolute Hfg.  53 der Klasse [ 194.1315,195.3118]
(  6) absolute Hfg.  90 der Klasse [ 195.3118,196.4921]
(  7) absolute Hfg. 112 der Klasse [ 196.4921,197.6724]
(  8) absolute Hfg. 148 der Klasse [ 197.6724,198.8527]
(  9) absolute Hfg. 162 der Klasse [ 198.8527,200.0330]
( 10) absolute Hfg. 130 der Klasse [ 200.0330,201.2133]
( 11) absolute Hfg. 107 der Klasse [ 201.2133,202.3936]
( 12) absolute Hfg.  75 der Klasse [ 202.3936,203.5739]
( 13) absolute Hfg.  51 der Klasse [ 203.5739,204.7542]
( 14) absolute Hfg.  19 der Klasse [ 204.7542,205.9345]
( 15) absolute Hfg.   6 der Klasse [ 205.9345,207.1148]
( 16) absolute Hfg.   4 der Klasse [ 207.1148,208.2951]
```

A2. Die Zeit T (in Sekunden), die ein Drucker benötigt, um ein Werbeplakat zu drucken, folgt einer Exponentialverteilung $Exp(\alpha)$ mit dem Parameter $\alpha = \dfrac{1}{12}$.
(a) Man simuliere $N = 1000$ Daten für eine Stichprobe.
Hinweis: `scipy.stats.expon.rvs(loc=0,scale=1/alpha,size=N)`
(a) Welche ist *die durchschnittliche* Druckzeit für das Drucken eines Plakats?
(b) Man zeichne ein Histogramm mit 12 Klassen für die simulierten Daten und auf demselben Bild zeichne man die Dichtefunktion `scipy.stats.expon.pdf(x,loc=0,scale=1/alpha)`.
(c) Man schätze danach die Wahrscheinlichkeiten $P(T < 20)$, $P(T > 10)$, $P(10 < T < 30)$.
Man vergleiche das Ergebnis mit den theoretischen Wahrscheinlichkeiten, welche man mit `scipy.stats.expon.cdf(x,loc=0,scale=1/alpha)` berechnet.
(d) Die generierten Daten der Stichprobe wurden in 12 Klassen (Intervallen) eingeteilt. Man zähle und gebe an wie viele Daten in jeder Klasse sind, und man zeichne auf einem neuen Bild das entsprechende Histogramm der *absoluten Häufigkeiten*.
`matplotlib.pyplot.hist(Daten,bins=12,density=False,edgecolor="black",label="absolute Hfg.")`
(e) *Auf einem anderen Bild* zeichne man auf dem Intervall $[0, 10]$ die Dichtefunktion der $Exp(1)$ Verteilung.

A3. Jedesmal, wenn Professor X eine Gruppe von 6 Personen trifft, wettet er 6 €, dass mindestens zwei von diesen 6 Personen im gleichen Monat Geburtstag haben. Anhand Simulationen schätze man: den *durchschnittlichen Gewinn oder Verlust* bei dieser Wette, bzw. die Wahrscheinlichkeit p, mit welcher Professor X eine Wette gewinnt.
Hinweis:

$$W(\text{Gewinn, bzw. Verlust, bei einer Wette}) \sim \begin{pmatrix} -6 & 6 \\ 1-p & p \end{pmatrix}.$$

Es gilt $p = P($"mindestens 2 Personen von 6 haben in demselben Monat Geburtstag"$)$.

A4. In einer Urne sind 4 weiße und 6 schwarze Kugeln. Ein Spieler zieht nacheinander eine Kugel ohne Zurücklegen. Das Spiel ist aus, wenn er eine weisse Kugel zieht oder wenn er dreimal gezogen hat. Die Zufallsvariable X zeigt die Anzahl der gezogenen schwarzen Kugeln.
a) Welches ist die (theoretische) Wahrscheinlichkeitsverteilung von X und simuliere zufällige Werte für X.
b) Der Spieler erhält 30 Punkte, wenn er drei schwarze Kugeln gezogen hat. Er erhält 25 Punkte, wenn er zwei schwarze Kugeln zieht. In allen anderen Fällen verliert er 5 Punkte. Anhand Simulationen schätze man die mittlere Punktezahl des Spielers. Man vergleiche das Ergebnis mit dem theoretischen Ergebnis.
Hinweise: a) Die Zufallsvariable X ist die Anzahl der gezogenen schwarzen Kugeln. Die (theoretische) Wahrscheinlichkeitsverteilung von X ist:

$P(X=0) = P(\text{erste Kugel ist weiß}) = \dfrac{4}{10}$.
$P(X=1) = P(\text{erste Kugel schwarz, zweite Kugel weiß}) = ...$
$P(X=2) = P(\text{erste und zweite Kugel schwarz, dritte Kugel weiß}) = ...$
$P(X=3) = P(\text{erste und zweite und dritte Kugel schwarz}) = ...$

b) Die Zufallsvariable Y zeigt die Punktezahl des Spielers an; die (theoretische) Wahrscheinlichkeitsverteilung von Y ist:
$P(Y=30) = ...$
$P(Y=25) = ...$
$P(Y=-5) = ...$
$\Longrightarrow$ der theoretische Erwartungswert von Y ist $E(Y) = ...$ (Punkte).

A5. Man simuliere mit Hilfe von `scipy.stats.uniform.rvs` 1000 zufällige Punkte aus dem Quader

$$[-1,\,1] \times [-1,\,1] \times [-1,\,1] \subset \mathbb{R}^3.$$

Sei D die ZG welche Distanz dieser Punkte zum Ursprung (0,0,0) darstellt. Man schätze a) den Erwartungswert (`numpy.mean`) und die Varianz von D (`numpy.var`).