

```

#Jakob H. Schauser, pwn274
#Hold 4, Pythonaflevering 1
#12/09-2018

# Importerer de forskellige pakker
import numpy as np
import matplotlib.pyplot as plt

#####Forsøg 1 - Variende Vægt:#####

#Jeg skriver nu vores data fra første forsøg ind
#fordelt på x- og y-aksen
vægte = np.array([20,100,200,500])

#Vi tog fem målinger per skiftende vægt så jeg skriver
#dem alle ind, for senere at kunne regne videre på dem
vægttider = np.array([[1.51,1.56,1.52,1.51,1.57],
                      [1.47,1.56,1.58,1.54,1.54],
                      [1.59,1.53,1.54,1.59,1.59],
                      [1.57,1.54,1.53,1.55,1.57]])

#Jeg beregner nu gennemsnittet af min y-akse-data
vt_avg = np.mean(vægttider, axis = 1)

#Og NumPy tillader os at beregne standardvigelsen således
vt_std = np.std(vægttider,axis=1,ddof=1)/np.sqrt(len(vægttider))

#Jeg starter et nyt plot
plt.figure(1)
#Indsætter mine x-værdier og de beregnede y-gennemsnit
plt.plot(vægte,vt_avg,'bo')
#Lægger også usikkerhederne ind i plottet
plt.errorbar(vægte,vt_avg,vt_std,ls='None')
#Vælger en god x- og y-akseskalering
plt.xlim(0,550)
plt.ylim(1.4,1.7)
#Vælger titlen, så vi kan genkende forsøget senere
plt.title("Gennemsnit af tider som funktion af vægt")
#Giver akserne labels
plt.xlabel(r"Vægt/$g$")
plt.ylabel(r"Tid/$s$")
#Gemmer vores plot
plt.savefig("VægtFigur.jpg")

# Jeg gør nu det samme for vores andet forsøg

#####Forsøg 2 - Skiftende Snorlængde:#####

#Jeg skriver nu vores data fra forsøg ind fordelt på x- og y-aksen
#Giver selvfølgelig et passende navn
længder = np.array([40,60,80,100])

#Vi tog fem målinger per skiftende vægt så jeg skriver
#dem alle ind, for senere at kunne regne videre på dem

```

```

længdetider = np.array([[1.69,1.60,1.69,1.59,1.68],
                        [1.68,1.74,1.74,1.82,1.72],
                        [1.81,1.83,1.86,1.82,1.84],
                        [2.06,2.18,2.10,2.13,2.16]])

#Jeg beregner igen gennemsnittet af min y-akse-data
ld_avg = np.mean(længdetider, axis = 1)

#Og NumPy tillader os igen at beregne standardafvigelsen således
ld_std = np.std(længdetider,axis=1,ddof=1)/np.sqrt(len(længdetider))

#Jeg starter et nyt plot
plt.figure(2)
#Indsætter mine x-værdier og de beregnede y-gennemsnit
plt.plot(længder,ld_avg,'bo')
#Lægger også usikkerhederne ind i plottet
plt.errorbar(længder,ld_avg,ld_std,ls='None')
#Vælger en god x- og y-akseskalering
plt.xlim(20,120)
plt.ylim(1.5,2.3)
#Skifter titlen, så vi kan genkende forsøget senere
plt.title("Gennemsnit af tider som funktion af snorlængde")
#Giver akserne labels
plt.xlabel(r"Længde/$cm$")
plt.ylabel(r"Tid/$s$")
#Gemmer vores plot
plt.savefig("LængdeFigur.jpg")

# Jeg gør nu det samme for vores tredje forsøg

#####Forsøg 3 - Vekslede Vinkel:#####

#Jeg skriver nu vores data fra forsøg ind fordelt på x- og y-aksen
#Giver selvfølgelig et passende navn
vinkler = np.array([0.317,0.522,0.671,0.906])

#Vi tog fem målinger per skiftende vægt så jeg skriver
#dem alle ind, for senere at kunne regne videre på dem
vinkeltider = np.array([[1.51,1.54,1.57,1.56,1.44],
                        [1.55,1.59,1.65,1.65,1.65],
                        [1.69,1.69,1.73,1.63,1.68],
                        [1.72,1.73,1.81,1.67,1.72]])

#Jeg beregner igen gennemsnittet af min y-akse-data
vk_avg = np.mean(vinkeltider, axis = 1)

#Og NumPy tillader os igen at beregne standardafvigelsen således
vk_std = np.std(vinkeltider,axis=1,ddof=1)/np.sqrt(len(vinkeltider))

#Jeg starter et nyt plot
plt.figure(3)
#Indsætter mine x-værdier og de beregnede y-gennemsnit
plt.plot(vinkler,vk_avg,'bo')

```

```
#Lægger også usikkerhederne ind i plottet
plt.errorbar(vinkler,vk_avg,vk_std,ls='None')
#Vælger en god x- og y-akseskalering
plt.xlim(0,1.2)
plt.ylim(1.4,1.9)
#Skifter titlen, så vi kan genkende forsøget senere
plt.title("Gennemsnit af tider som funktion af vinkler")
#Giver akserne labels
plt.xlabel(r"Vinkel/$radianer$")
plt.ylabel(r"Tid/$s$")
#Gemmer vores plot
plt.savefig("VinkelFigur.jpg")
```