```
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as ss
import scipy.optimize as scp
import os
import csv

exec(open(’Kalibrering/kalibrering.py’).read())
exec(open(’../Scripts/Statistik.py’).read())
exec(open(’../Scripts/data_renser.py’).read())

fig, ax = plt.subplots(1,2,figsize = (16,8))

def sinus(t, *p):
    A = p[0]
    w = p[1]
    k = p[2]
    b = p[3]
    d = p[4]
    return (A*np.cos(w*t+k)*np.exp(-b*t)+d)

datases = [[’30grader’, ’30vind’], [’20grader’, ’20vind’],
           [’10grader’, ’10vind’],[’40grader’, ’40vind’]]

def plotter_fitter(data, i):

    sol1 = Data(’Kalibrering/’ + data)
    spÃ¦nding = sol1.points
    ts = sol1.t*1000

    mask = sol1.rinse2(0.15, 0.02)
    vink = vinkel(spÃ¦nding, *kali)*(360/(2*np.pi))

    ax[i].scatter(ts[˜mask], vink[˜mask], color = ’blue’, alpha = 0.2)
    ax[i].plot(ts[mask], vink[mask], ’ro’, alpha = 0.4, markersize = 4)

    error = propagation_function(spÃ¦nding[mask], vinkel, list(kali), pcov)

    guess = [22,-5,2,0,1]

    popt, pcov2 = scp.curve_fit(sinus, ts[mask], vink[mask], guess,
                                sigma = error, absolute_sigma = True)


    error1 = propagation_function(ts[mask], sinus, list(popt), pcov2)
    ax[i].fill_between(ts[mask],
                    sinus(ts[mask], *popt)-error1,
                    sinus(ts[mask], *popt)+error1,
                    alpha = 0.3)

    ax[i].plot(ts[mask], sinus(ts[mask], *popt), ’k’, linewidth = 2)

    sigma_b = round(np.sqrt(np.diag(pcov2))[3], 3)
    ax[i].set_xlabel(’t’)
    ax[i].set_ylabel(’vinkel’)
    ax[i].set_title(data + ’   b = {}   ’.format(round(popt[3], 3))
                    + ’$\sigma_b\sim$ {}’.format(sigma_b), fontsize = 18)


def save_plots(data, name):

    plotter_fitter(data[0], 0)
    plotter_fitter(data[1], 1)
    fig.savefig(’Plots/’ + name)

    for a in ax:
        a.clear()

save_plots(datases[0], ’30vind’)
save_plots(datases[1], ’20vind’)
save_plots(datases[2], ’10vind’)
save_plots(datases[3], ’40vind’)

fig, ax = plt.subplots(figsize = (16,8))

vinkler = [10, 20, 30, 40]
b1 = [0.119, 0.053, 0.059, 0.052]
b2 = [0.122, 0.056, 0.062, 0.06]

ax.plot(vinkler, b1, ’ro’, label = "Uden sejl")
ax.plot(vinkler, b2, ’ko’, label = "Med sejl")

ax.set_title(’DÃ¦mpningens vÃ¦rdi for forskellige vinkler’, fontsize = 18)
ax.set_xlabel(’Vinkel’, fontsize = 16)
ax.set_ylabel(’DÃ¦mpning’, fontsize = 16)
ax.legend()
plt.show()

fig.savefig(’Plots/Res_Vind’)
# plt.tight_layout()
# plt.show()
```