```
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(figsize = (16,8))

# Data-funktionen lÃ¦ser .txt-filen gemmer i et data-objekt med .points og .t
# som attributer

sol1 = Data('Kalibrering/40grader')
spÃ¦nding = sol1.points
ts = sol1.t*1000

# Rinse2 er en metode defineret pÃ¥ data-objektet som fjerner den relevante del
# af dataet.
# Hvis du er interesseret er denne funktion defineret inde i
# Scripts/data_renser.py

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

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

# Fejlpropagering

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

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)

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.fill_between(ts[mask],
                sinus(ts[mask], *popt)-error1,
                sinus(ts[mask], *popt)+error1,
                alpha = 0.3)

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

ax.set_xlabel('t')
ax.set_ylabel('vinkel')

plt.show()
```