

Computational Methods and Modelling

Antonio Attili & Edward McCarthy

antonio.attili@ed.ac.uk

ed.mccarthy@ed.ac.uk

*School of Engineering
University of Edinburgh
United Kingdom*

Tutorial 5

Fitting and interpolation



THE UNIVERSITY
of EDINBURGH

Exercise 1: Linear regression

- ▶ Consider the data in the file `data_reg.py` on Learn.
- ▶ These data are experimental measurements of normalized internal energy (variable y) vs a normalized temperature (variable x).
- ▶ Determine a model linking these two quantities by fitting the data with a straight line (linear regression). Do the fitting by applying the formulas presented in the lectures and by using intrinsic python functions (e.g., `np.polyfit` and `np.poly1d`).
- ▶ Plot the data before you start coding to help understanding the task.

Exercise 1: Linear regression

”

```
# importing modules
import numpy as np
import matplotlib.pyplot as plt
import math
import random

x = np.array([0.      , 0.06666667, 0.13333333, 0.2      , 0.26666667, 0.33333333,
             0.4      , 0.46666667, 0.53333333, 0.6      , 0.66666667, 0.73333333,
             0.8      , 0.86666667, 0.93333333, 1.      ,   ])

y = np.array([2.17312991, 2.19988829, 2.33988149, 2.33940595, 2.41968027, 2.99955891,
             3.04855788, 3.86631749, 3.66009775, 4.42305111, 4.22747852, 4.11717969,
             3.87539822, 4.53121841, 5.52211102, 5.30792203])
```

Exercise 2: Spline interpolation

- ▶ Consider the data in the file `data_int.py` on Learn.
- ▶ These data are experimental measurements with a rather coarse resolution (we have only 16 measurement points).
- ▶ We want to increase the resolution of the data by interpolating with splines to have 128 points in the same interval.
- ▶ Use `scipy.interpolate.splrep` and `scipy.interpolate.splev` for this task.
- ▶ Plot the data before you start coding to help understanding the task.

Exercise 2: Spline interpolation

”

```
# importing modules
import numpy as np
import matplotlib.pyplot as plt
import math
import random

x = np.array([0. , 0.06666667, 0.13333333, 0.2 , 0.26666667, 0.33333333,
0.4 , 0.46666667, 0.53333333, 0.6 , 0.66666667, 0.73333333,
0.8 , 0.86666667, 0.93333333, 1. , ])

y = np.array([ 0.00000000e+00, 7.78309056e-01, 1.24040577e+00, 1.24494914e+00,
8.90566050e-01, 4.33012702e-01, 1.12256994e-01, 4.54336928e-03,
-4.54336928e-03, -1.12256994e-01, -4.33012702e-01, -8.90566050e-01,
-1.24494914e+00, -1.24040577e+00, -7.78309056e-01, -4.89858720e-16])
```