Solution Review: Fitting a Wave

This lesson discusses the solution to the fitting a wave problem.

Solution

Inserting the data below:

$$(t_0, y_0) = (0.25, 3)$$

$$(t_1, y_1) = (0.5, 2)$$

$$(t_2, y_2) = (0.75, -3)$$

$$(t_3, y_3) = (1, 0)$$

in the equation

$$y = a\ sin(\pi t) + b\ sin(2\pi t) + c\ sin(3\pi t) + d\ sin(4\pi t)$$

gives the following system of equations

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & 1 & \frac{1}{\sqrt{2}} & 0 \\ 1 & 0 & -1 & 0 \\ \frac{1}{\sqrt{2}} & -1 & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ -3 \\ 0 \end{bmatrix}$$

Now let's implement it in Python and solve it.

```
import numpy as np
import matplotlib.pyplot as plt

# setting up the equation
tp = np.array([0.25, 0.5, 0.75, 1])
yp = np.array([3, 2, -3, 0])
A = np.zeros((4, 4))
rhs = np.zeros(4)
for i in range(4):
  A[i] = np.sin(1 * np.pi * tp[i]), np.sin(2 * np.pi * tp[i]), \
      np.sin(3 * np.pi * tp[i]), np.sin(4 * np.pi * tp[i])  # Store one row at a time
  rhs[i] = yp[i]

# Solving the equation
sol = np.linalg.solve(A, rhs)
print('a, b, c, d: ', sol)

# plotting the wave
t = np.linspace(0, 1, 100)
y = sol[0] * np.sin(1 * np.pi * t) + sol[1] * np.sin(2 * np.pi * t) + \
    sol[2] * np.sin(3 * np.pi * t) + sol[3] * np.sin(4 * np.pi * t)
plt.plot(t, y, 'b', label='wave')
plt.xlabel('t')
plt.ylabel('y')

# plotting the initial points
plt.plot(tp, yp, 'ro', label='data')
plt.legend(loc='best');

# saving figure
plt.savefig('output/wave.png')
```

Explanation

- In lines 9 - 12, we are creating a 2-D matrix to solve the system of 4 linear equations. This matrix is formed using:
  - the equation given above,
  - the independent variable t, denoted as `tp`,
  - corresponding values of the dependent variable y, denoted as `yp`.
- We are solving the equation in line 15 using the `solve` method from the `linalg` submodule in `numpy`.
- In line 19, we create an equally spaced array for `t` ranging from 0 to 1, with length 100.
- In lines 20 - 21, we use `t` and the solution of the equation `sol` to plot the wave.

wave.

- In lines 27 - 28, we are plotting the initial points, `tp` and `yp`.

That is all for systems of linear equations for now. Let's move on to symbolic computation in Python.