

## Toán học tính toán: Python buổi 7/10

### 6 Lý thuyết xấp xỉ

```

1 X = [1, 1.3, 1.7, 2.]
2 Y = [3.5, 4., 4.6, 5.2]

3 from sympy import *
4 x = symbols('x')
5 cs = [1 + 0*x, x, log(x)] # biểu thức xuất hiện biến thì mới thể biến được

6 V = [ [cs_i.subs(x, X_k) for X_k in X] for cs_i in cs ]

7 import numpy as np
8 A = [ [np.dot(V_i, V_j) for V_j in V] for V_i in V ]
9 b = [np.dot(Y, V_i) for V_i in V]

10 hs = np.linalg.solve(
11     np.array(A).astype(float),
12     np.array(b).astype(float) )

13 P = hs.dot(cs)

14 [P.subs(x, X_k) for X_k in X]
15 errors = [Y_k - P.subs(x, X_k) for (X_k, Y_k) in zip(X, Y)]
16 np.linalg.norm(np.array(errors).astype(float))

```

Mã 43: Phương pháp bình phương tối thiểu: trường hợp rời rạc, 1 biến

```

1 X = [-0.7, 1.7, -4.9, 3.1, -1.3]
2 Y = [-2.9, -1.1, -2.9, 1.5, 0.8]
3 Z = [7.1, 5.8, -3.1, -1, -8.7]

4 from sympy import *
5 x, y = symbols('x y')
6 cs = [1 + 0*x, x, y]

7 V = [ [cs_i.subs({x: X_k, y: Y_k}) for (X_k, Y_k) in zip(X, Y)] for cs_i in cs ]

8 import numpy as np
9 A = [ [np.dot(V_i, V_j) for V_j in V] for V_i in V ]
10 b = [np.dot(Z, V_i) for V_i in V]

11 hs = np.linalg.solve(
12     np.array(A, dtype=float),
13     np.array(b, dtype=float) )

14 P = hs.dot(cs)

15 [P.subs({x: X_k, y: Y_k}) for (X_k, Y_k) in zip(X, Y)]
16 errors = [Z_k - P.subs({x: X_k, y: Y_k}) for (X_k, Y_k, Z_k) in zip(X, Y, Z)]
17 np.linalg.norm( np.array(errors, dtype=float) )

```

Mã 44: Phương pháp bình phương tối thiểu: trường hợp rời rạc, 2 biến