Dynamic programming example
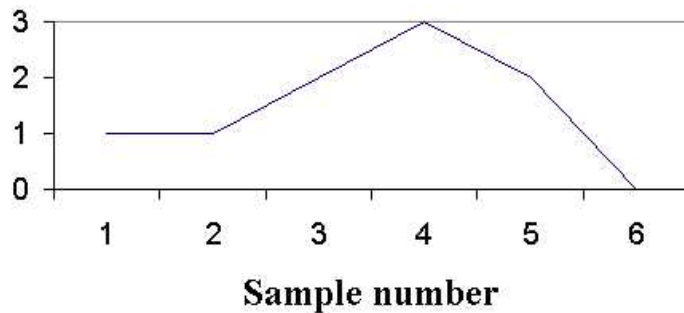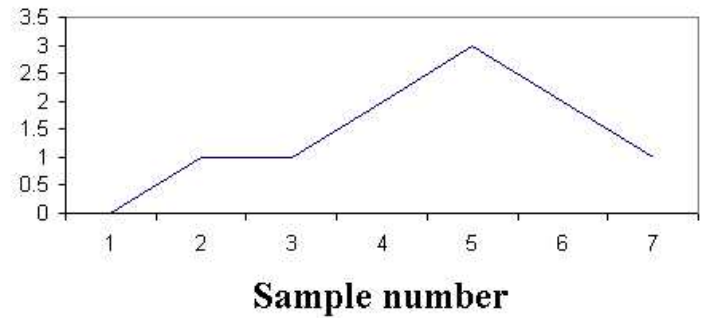# (dynamic time warping)

Suppose we wish to compare and evaluate the difference between the following two signals:

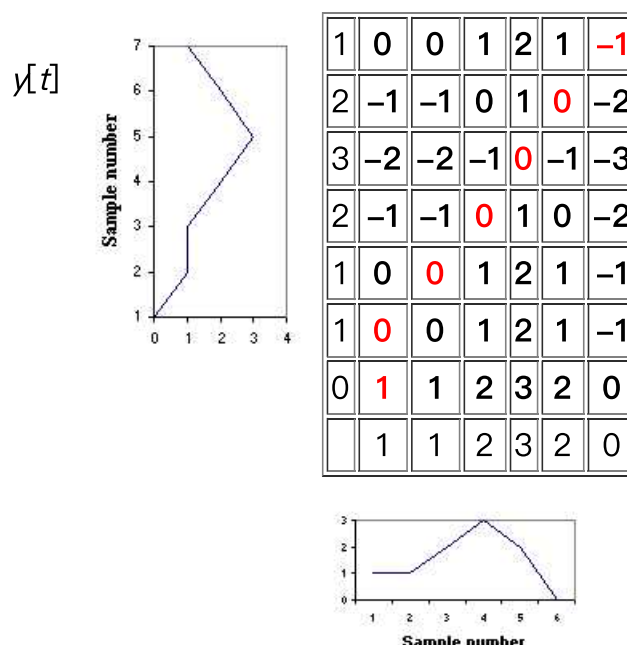### Test pattern x[t]

Sample number

Reference pattern y[t]

Sample number

a) (Input) test signal, $x[t]$:   1 1 2 3 2 0

b) (Stored) reference signal, $y[t]$:   0 1 1 2 3 2 1

Sample–by–sample difference $x[t] - y[t]$: 1 0 1 1 −1 −2 undefined

Both signals are similar in that they are single–peaked. However, the stored reference signal is longer than the test signal, and the peak is later. In other words, the two signals are not synchronized in time.

To calculate the difference between them, consider a matrix of the distance between every sample of $x[t]$ and each sample of $y[t]$. The distance matrix D is:

$y[t]$

1	0	0	1	2	1	−1
2	−1	−1	0	1	0	−2
3	−2	−2	−1	0	−1	−3
2	−1	−1	0	1	0	−2
1	0	0	1	2	1	−1
1	0	0	1	2	1	−1
0	1	1	2	3	2	0
	1	1	2	3	2	0

Sample number

$x[t]$

There is a sequence of low numbers, close to the diagonal, indicating which samples of $x[t]$ are closest in value to those of $y[t]$. These are marked in red.

Instead of a simple subtraction, it is customary to use a symmetrical distance measure, such as $(x[t] - y[t])^2$:

$y[t]$

1	0	0	1	4	1	1
2	1	1	0	1	0	4
3	4	4	1	0	1	9
2	1	1	0	1	0	4
1	0	0	1	4	1	1
1	0	0	1	4	1	1
0	1	1	4	9	4	0
	1	1	2	3	2	0

$x[t]$

Next: <u>Finding the best path through the matrix</u>