Structural health monitoring meets data mining

Miao, S.

Author: Miao, Shengfa
Title: Structural health monitoring meets data mining
Issue Date: 2014-12-16

Chapter 2

Preliminaries

2.1 Background

In this chapter, we will review and explain a number of concepts to help better understand subsequent chapters. We begin by giving definitions related to the sensors involved in the sensor network, and then introduce concepts related to datasets collected with sensors.

In the InfraWatch project, a sensor network consisting of 145 sensors is employed. These sensors are placed along three cross-sections of a single span of the bridge[1]. Each of the sensors is either embedded in the concrete, or attached to the outside of the deck and girders. To measure the forces in different directions on the bridge, we utilize sensors of different types: *vibration sensors* measure vertical motion of the bridge, and *strain gauges* measure horizontal strain caused by deflection of the bridge. To measure the temperature of different parts of the bridge, we also employ multiple *temperature sensors*. To formalise this placement, we define each sensor as follows:

Definition 1 (Sensor) *A sensor is a tuple* $(type, x, y, e, o)$*, where* $type \in \{St, Vi, Te\}$ *indicates the sensor type (strain, vibration, and temperature, respectively),*

[1] The bridge has multiple spans, but they are identical in design, and independently constructed.

x and y are its coordinates on the bridge, $e \in \{embed, attach\}$ indicates whether the sensor is embedded or attached to the concrete, and $o \in \{X\text{-axis}, Y\text{-axis}\}$ indicates the orientation of the sensor.

The sensor network is collecting data at 100 Hz, and each sensor in the network produces a sequence of data-points, which forms a *time series* of measurements. We define a time series as:

Definition 2 (Time Series) *A time series $\boldsymbol{T}$ is an ordered sequence of n real values*

$$\boldsymbol{T} = (t_1, t_2, \ldots, t_n), \quad t_i \in \mathbb{R}$$

in which $t_i \in \mathbb{R}$ stands for the i^{th} item in the sequence collected by a sensor. In this thesis, we will often also refer to data produced by a single sensor over time as a *signal.* When we speak of a signal, we are typically interested in general aspects of the data (such as the main frequency), whereas when we speak of a time series, it typically refers to a specific sequence of data measured over a specific interval of time.

Instead of the whole time series, in some cases, we are more interested in part of a time series. For example, given a time series recording traffic events of one whole day, we may just want to know the traffic situation during rush hour. Here we define a piece of a given time series as a *subsequence*, and define it formally as:

Definition 3 (Subsequence) *Given a time series $\boldsymbol{T} = (t_1, \ldots, t_n)$ of length n, a subsequence $\boldsymbol{S}$ of $\boldsymbol{T}$ is a series of length $m \leq n$ consisting of contiguous data points from $\boldsymbol{T}$*

$$\boldsymbol{S} = (t_k, t_{k+1}, \ldots, t_{k+m-1}), \quad 1 \leq k \leq n - m + 1$$

In the following sections, we will introduce some operations related to the concepts mentioned above.

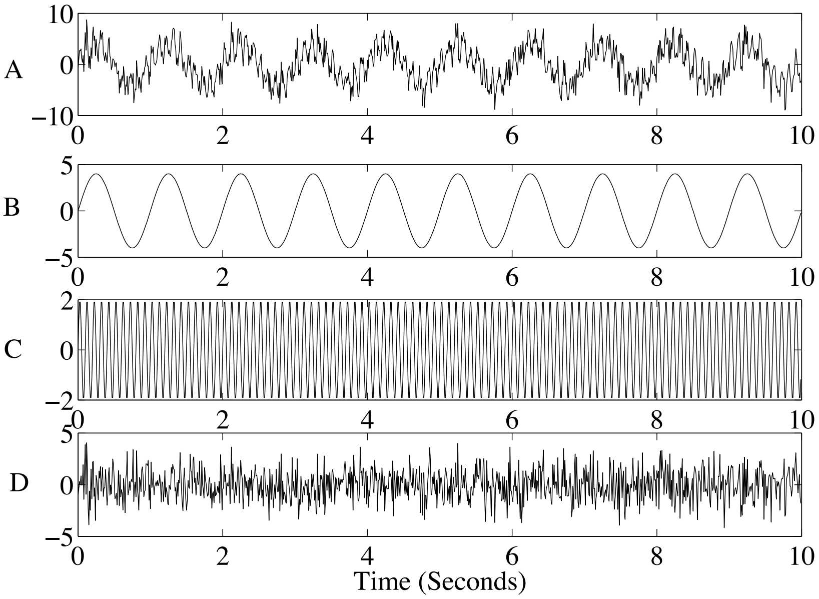

Figure 2.1: A signal in the time domain - The pictures illustrate a simulated signal (A) and its components (B, C, D) in the time domain.

2.2 Fourier Transform

The top picture (A) in Fig. 2.1 shows a signal in the time domain, which is sampled at 100 Hz (100 data points each second). As mentioned in the previous chapter, the signal can be generally decomposed into three components: a low frequency component (B), a high frequency component (C) and random noise (D). One method that is used to convert a signal from the time domain to the frequency domain, essentially extracting spectral information from the signal, is the Discrete Fourier Transform (DFT) [19]. The transform is defined as:

Definition 4 (Discrete Fourier Transform (DFT)) *Given a sequence of N samples* $\{x_0, x_1, \dots, x_{N-1}\}$*, the Discrete Fourier Transform is defined as:*

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N} \quad k \in \mathbb{Z}$$

in which $\mathbb{Z}$ are integers [20].

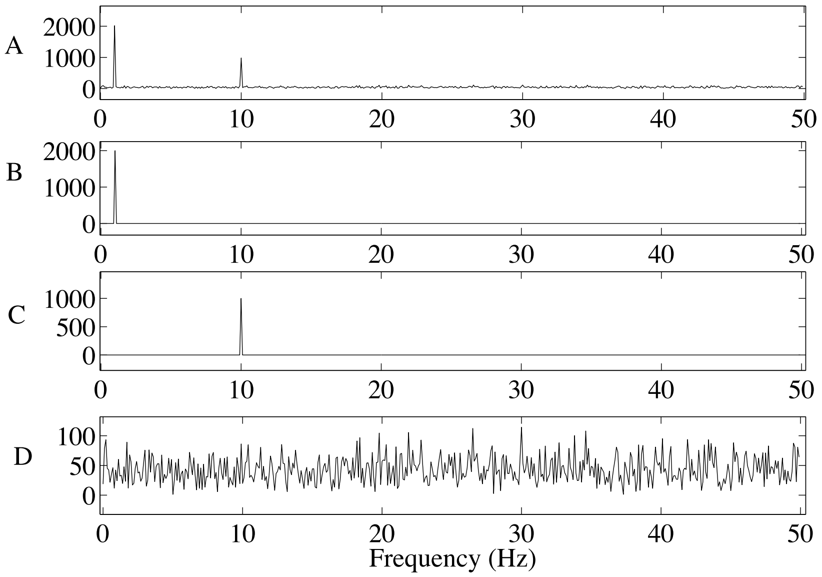

Figure 2.2: A signal in the frequency domain - The pictures illustrate a simulated signal (A) and its components (B, C, D) in the frequency domain.

An efficient way to implement DFT is the Fast Fourier Transform (FFT) [21]. With the FFT, the signals in Fig. 2.1 can be transformed into four spectra, as shown in Fig. 2.2. Spectrum A corresponds to signal A, which consists of two dominant peaks, corresponding to the two components present. The components B and C are noise-free signals, so the corresponding spectra consist of single peaks. Component D is a non-periodic signal, so there are no clear peaks in its spectrum D.

2.3 Convolution

Apart from the subtle degradation of the structure which we ignore for the moment, a bridge can be viewed as a *Linear Time-Invariant* (LTI) system [22]. Here, *time-invariant* indicates that the nature of the response of the system does not change over time. LTI systems are *linear* because their ‘output’ is a linear combination of the ‘inputs’. The behaviour of an LTI system with single input signal $x(n)$ and single output signal $y(n)$ can be represented as a discrete convolution

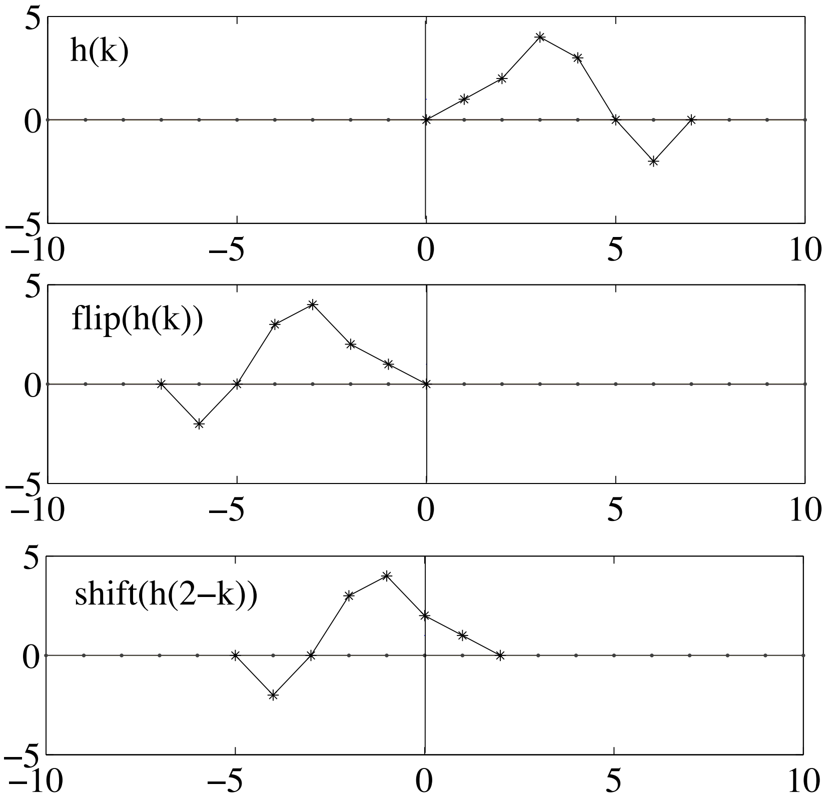

Figure 2.3: Graphical illustration of convolution - The top picture is the impulse response signal. The middle picture illustrates the flip operation. The bottom picture demonstrates the shift operation, which is obtained by shifting the flipped impulse response signal by 2 to the right.

integral [23], which is defined as:

$$y(n) = (x * h)(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{2.1}$$

in which $h(n-k)$ is impulse response signal. The convolution summation can be graphically interpreted as a combination of two operations: a flip and shift. For a given impulse response signal $h(k)$, shown as the top picture in Fig. 2.3, we first flip the impulse response signal (the middle picture in Fig. 2.3), then shift the flipped impulse response signal with n data points (the bottom picture in Fig. 2.3 is obtained by shifting the flipped $h(k)$ by 2 to the right).

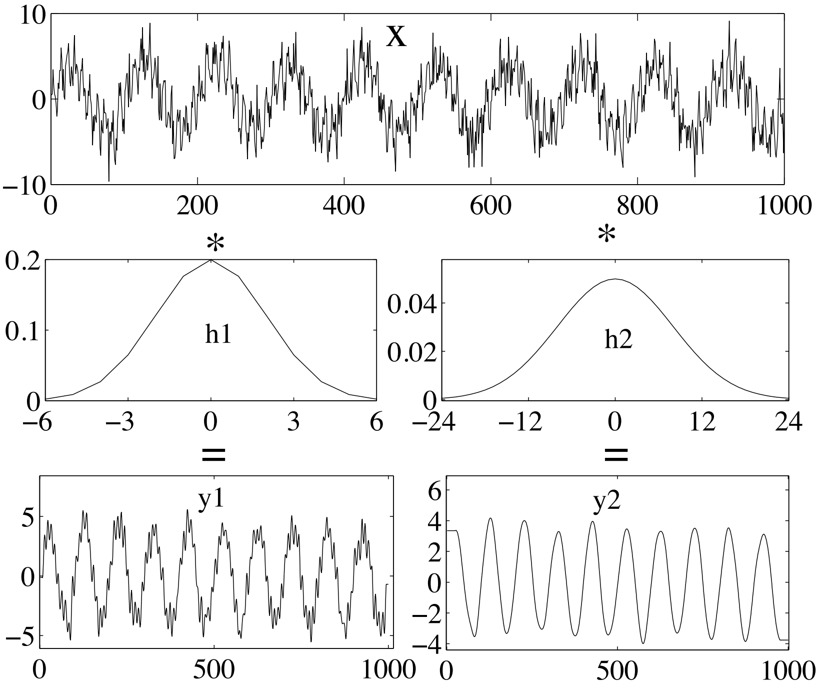

Figure 2.4: Convolution for low-pass filtering - The top picture is the input signal. The middle left curve is a small Gaussian kernel with $\sigma = 2$. The middle right curve is a big Gaussian kernel with $\sigma = 8$. The bottom graphs show the corresponding resulting signals using the small and big kernel. Note how the big kernel has a larger influence on the resulting signal, and most of the high-frequency component has been removed from the signal.

If the system is considered to be a filter, the impulse response signal is called a filter kernel. One of the widely used kernels is the *Gaussian* kernel, which resembles a "bell curve". The normalised *Gaussian* kernel is defined as:

$$\mathbf{G}(\sigma, x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \tag{2.2}$$

where the parameter σ controls the width of the "bell".

The Gaussian kernel can be used for low-pass filtering. Shown as the pictures in Fig. 2.4, the input signal X is the same signal as the top picture in Fig. 2.1. When the input signal X is convolved with a small Gaussian kernel $h1$ ($\sigma = 2$), the output signal $y1$ preserves both the low and middle frequency components, while suppressing the influence of high frequency noise. When the input signal X is convolved with a big Gaussian kernel $h2$ ($\sigma = 8$), the output signal $y2$ only preserves the low frequency component.

2.4 Similarity Measurement

Given two time series (or subsequences) of interest, we may want to know how similar they are. A number of similarity measurements have been proposed [24], of which the Euclidean Distance (ED) [25, 26] is the most common [27, 28]. Given two time series $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_n)$, their ED-based similarity can be obtained by comparing local point values. The ED between P and Q can be defined as:

$$D(P, Q) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

The advantage of the ED is that it is simple and efficient. When the given time series are well aligned, like the left picture in Fig. 2.5, the ED works well. However, the ED is sensitive to scaling, and when shifting and temporal distortions exist in the given time series, like the right picture in Fig. 2.5, it is proven to be ineffective [28] as a similarity measure.

Time series might still be intuitively similar even though each series might be subject to a certain scaling. In this situation, the similarity can still be captured with the so-called *correlation measures*. The most well-known, Pearson's correlation coefficient [29] is defined as:

$$r = \frac{n \sum p_i q_i - \sum p_i \sum q_i}{\sqrt{n \sum p_i^2 - (\sum p_i)^2} \sqrt{n \sum q_i^2 - (\sum q_i)^2}}$$

The correlation coefficient r is always between -1 and 1, where 1 means that the two series are a strict linear function of one another, 0 means that they are completely uncorrelated, and -1 means that they are perfect opposites. Although this correlation also suffers from temporal shifting and distortions, it is invariant to scaling and translation (in the domain of measurement).

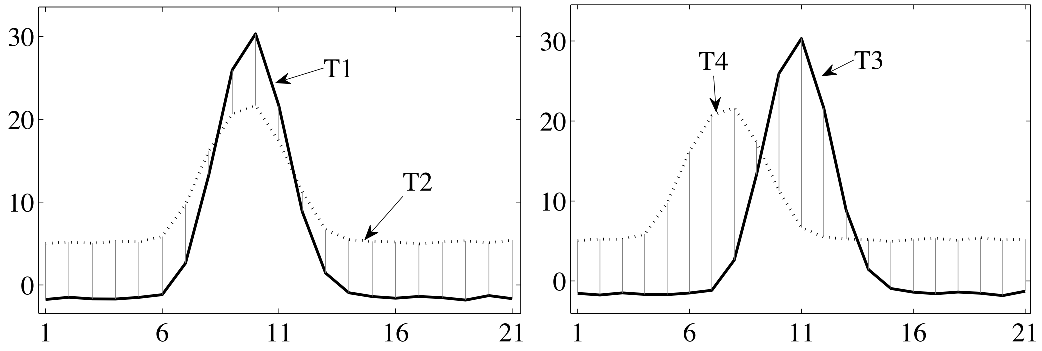

Figure 2.5: Euclidean Distance. - The left picture demonstrates the ED between two aligned time series T1 and T2; the right picture illustrates the ED between two shifted time series T3 and T4, where ED fails to capture the intuitive similarity.

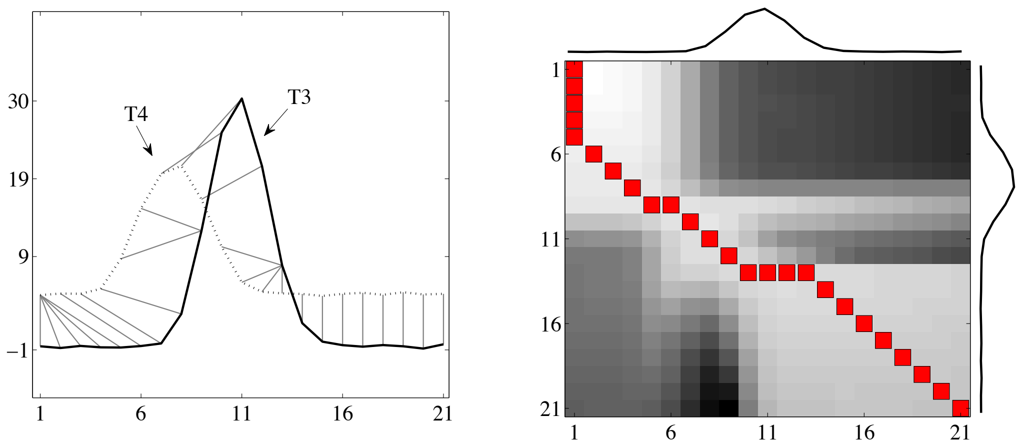

Figure 2.6: Dynamic Time Warping. - The left picture shows the DTW between two shifted time series T3 and T4; the right picture illustrates the accumulated distance matrix and the optimal matching path (a square-chain going through light grey cells), between time series T3 and T4.

To handle shifting, stretching and compression along the temporal dimension, Dynamic Time Warping (DTW) [30] was proposed, which achieves an optimal temporal alignment, shown as the left picture in Fig. 2.6, through detecting the shortest warping path in an accumulated distance matrix [24, 31, 32, 33], shown as the right picture in Fig. 2.6. Given two time series $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_m)$, the element $r(i, j)$ in the accumulated distance matrix [34] is defined as:

$$r(i,j) = \begin{cases} d(p_1, q_1) & i = 1, j = 1 \\ d(p_i, q_1) + r(i-1, 1) & j = 1, 1 < i \leq n \\ d(p_1, q_j) + r(1, j-1) & i = 1, 1 < j \leq m \\ d(p_i, q_j) + \min\{r(i-1, j-1), r(i-1, j), r(i, j-1)\} & 1 < i \leq n, 1 < j \leq m \end{cases}$$

where $d(i, j)$ is the distance found in the current cell, which can be chosen from several metrics, such as p-norms [35] (when p is 2, the distance becomes the Euclidean distance), and $r(i, j)$ is the cumulative distance of $d(i, j)$ and the minimum cumulative distances from three adjacent cells.

Finding the shortest warping path is a non-trivial problem, whose computation complexity is $O(n^2)$. To speed up the computation of DTW, some lower bounding constraints, such as LB_Keogh [32, 36] and the Ratanamahatana-Keogh Band [37], have been introduced to prune expensive computations, which can reduce the complexity to $O(n)$.

In Chapter 6, we propose a novel pattern detection method, based on landmarks and constraints. The method is capable of extracting predefined patterns efficiently, and is robust to temporal and magnitude deformations.