A first look at basis expansions

In the previous section we looked at the sampling theorem from the point of view of frequency-domain transformations. This is a very good way to understand it in the context of classical signal processing. We can, however, think of it in another way: as an orthobasis expansion in a Hilbert space. This is a powerful viewpoint, as it will allow us to generalize what it means to **discretize** a continuous time signal. It will also give us a unified framework for treating all different types of signals (continuous, discrete, infinite length, time-limited, etc.)[1]

We will start with some concrete examples, and then develop the general framework. Essentially, we will be concerned with extending and abstracting the key concepts from **linear algebra**:

- linear subspaces
- norms
- bases / change of bases
- inner products / orthogonality / projections
- linear operators (matrices in finite dimensions)
- eigenvalues / singular values

[1] Good sources for the material in the rest of Section I include:

- Moon and Stirling, Chapter 2
- G. Strang, "Linear Algebra and its Applications"
- N. Young, "An Introduction to Hilbert Space"
- Naylor and Sell, "Linear Operator Theory in Engineering and Science"

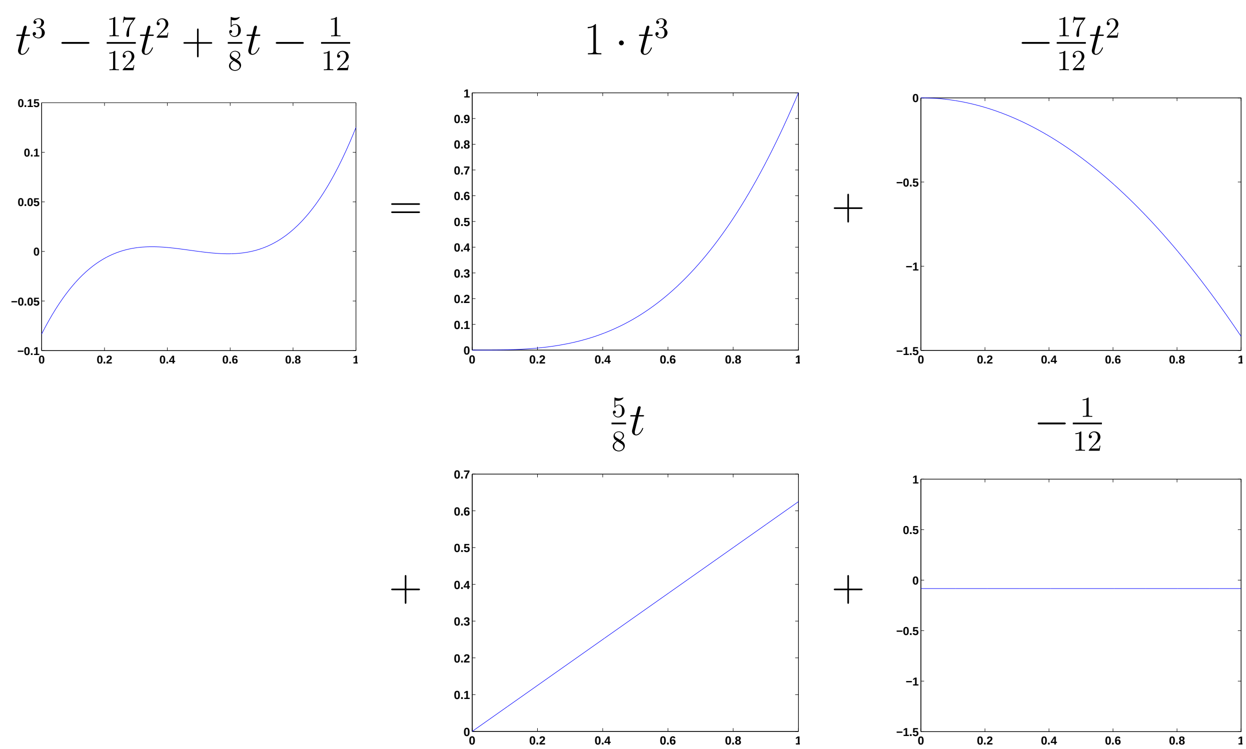
Example: Fourier series and bandlimited signals

We have already seen one of the canonical examples of a basis expansion. The Fourier series representation allows us to represent any periodic function as a superposition of harmonic sinusoids.

The sampling theorem, which as we argued earlier is mathematically equivalent to Fourier series, allows us to represent any bandlimited function as a superposition of suitably shifted sinc functions.

Example: Taylor series

It is almost too obvious that any m^{th} order polynomial can be written as a super position of the $m+1$ functions $1, t, t^2, \ldots, t^m$. (Indeed, this is pretty much the definition of polynomial.) For example:



More generally, well-defined "infinite degree" polynomials are called **analytic functions**[2]. For these functions, there is a systematic way of computing the expansion coefficients to represent $x(t)$ on some interval (say $[-1/2, 1/2]$ again),

$$x(t) = \sum_{k=0}^{\infty} \alpha_k \cdot t^k, \quad \text{where} \quad \alpha_k = \frac{x^{(k)}(0)}{k!},$$

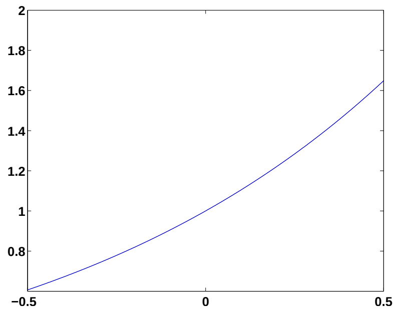
where $x^{(k)}(t)$ is the k^{th} derivative of $x(t)$. You may recall this as the Taylor series expansion. For instance, on the interval $[-1/2, 1/2]$, we can write

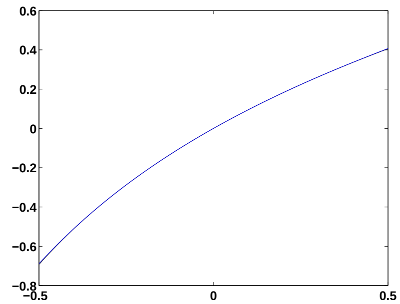
$$\begin{aligned} e^t &= \sum_{k=0}^{\infty} \alpha_k \cdot t^k, \quad \text{where } \alpha_k = \frac{1}{k!}, \\ \log(1+t) &= \sum_{k=0}^{\infty} \alpha_k \cdot t^k \quad \text{where } \alpha_k = \frac{(-1)^{k+1}}{k}, \\ \sin(2\pi t) &= \sum_{k=0}^{\infty} \alpha_k \cdot t^k \quad \text{where } \alpha_k = \begin{cases} \frac{(-1)^{(k+3)/2}(2\pi)^{k+1}}{(k+1)!} & k \text{ odd} \\ 0 & k \text{ even} \end{cases} \end{aligned}$$

Here are the three examples above with the series truncated to the first six terms:

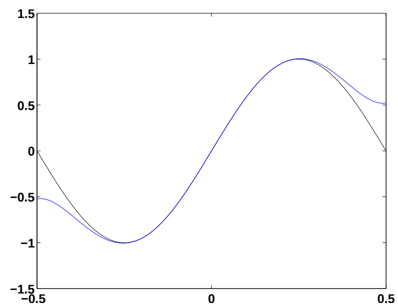
[2]More technically, a function $x(t)$ is called analytic on an interval $[a, b]$ if for any $t_0 \in [a, b]$, the infinite sum $\sum_{k \geq 0} a_k (t - t_0)^k$ converges to $x(t)$ for some choice of $\{a_k\}$.

$$e^t \approx \sum_{k=1}^{6} \frac{1}{k!}\, t^k \qquad \log(1+t) \approx \sum_{k=1}^{6} \frac{(-1)^{k+1}}{k}\, t^k$$





$$\sin(2\pi t) \approx t - \frac{2\pi}{6} t^3 + \frac{8\pi^3}{120} t^5$$



The exp and log examples are pretty much spot-on with only six terms, while the sin example is still suffering a little on the edges.

It is important to realize that Taylor series is not the only way to build up a function as a sum of polynomials, and despite its convenience, it has a few unsatisfying properties (e.g. there are infinitely differentiable functions whose Taylor series converges, but does not equal the original function anywhere). Moreover, it is unclear how to use Taylor series for signals that only have a small number of derivatives.

Example: Lagrange polynomials

The sampling theorem from the last set of notes showed that we can build-up a bandlimited signal from a (possibly infinite) superposition of sinc functions; for example, if $x(t)$ is bandlimited to $B = \pi$, then

$$x(t) = \sum_{k=-\infty}^{\infty} \alpha_k g(t-k), \quad g(t) = \frac{\sin(\pi t)}{\pi t},$$

and the expansion coefficient are simply samples of $x(t)$: $\alpha_k = x(k)$. We are able to reproduce $x(t)$ in this example because it adheres to known model: its Fourier transform is zero outside of $[-\pi, \pi]$.

It might be that we have a different model for the continuous-time signal $x(t)$. One alternative model might be that $x(t)$ is a polynomial. In fact, given a finite number $M+1$ of samples of $x(t)$, there is always an M^{th} order polynomial that passes through all of them — another way of saying this is that an M^{th} order polynomial can be reproduced from $M+1$ samples. If $x(t)$ is an M^{th} order polynomial, then using samples $x(0), x(T), \ldots, x(MT)$, we can write

$$x(t) = \sum_{k=0}^{M} \alpha_k \, p_k(t)$$

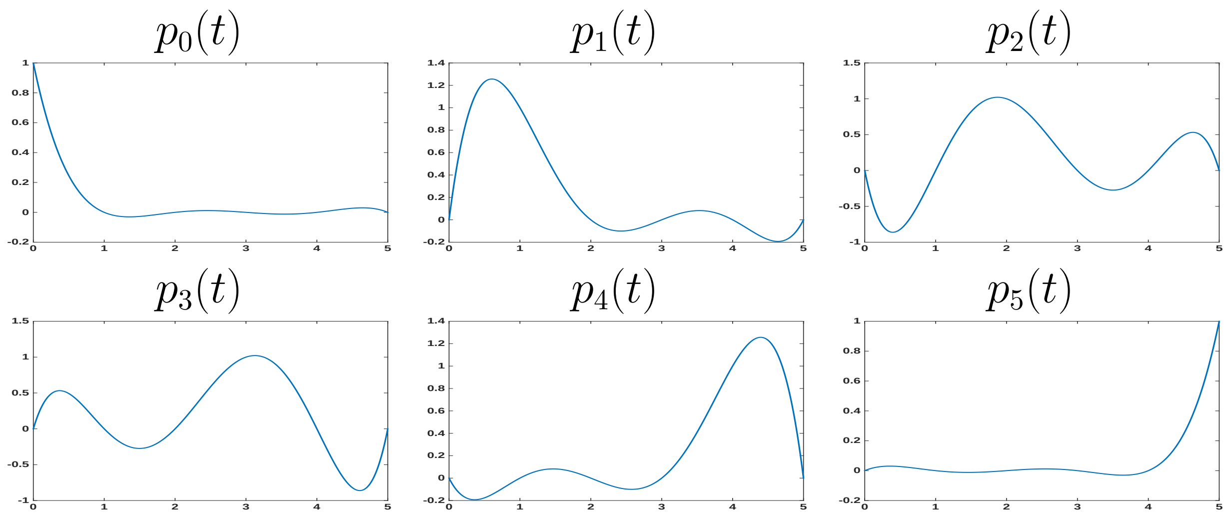
where $\alpha_k = x(kT)$, and

$$p_k(t) = \prod_{\substack{0 \leq m \leq M \\ m \neq k}} \frac{t - mT}{(k-m)T}.$$

It should be clear from the expression above that each $p_k(t)$ is a different M^{th} order polynomial, and

$$p_k(nT) = \begin{cases} 1, & n = k \\ 0, & n \neq k. \end{cases}$$

Given the samples $x(kT)$, moving to continuous-time signal $x(t)$ is called **Lagrange interpolation**. Here are the 6 basis functions $p_k(t)$ for $M = 5$:



One problem with Lagrange polynomials is that they are extremely unstable outside of the interval $[0, M]$ — they diverge very quickly to either ∞ or $-\infty$ (as all polynomials must do).

Example: Splines

A more stable way to interpolate between a sequence of discrete points is by using a **polynomial spline**. Given a sequence of locations $t_1, \ldots, t_K$ and function values at those locations $v_{t_1}, \ldots, v_{t_K}$, the ℓ^{th} order polynomial spline is the function $x(t)$ which obeys:

$$x(t_k) = v_{t_k}, \text{ for } k = 1, \ldots, K,$$

and

$$x(t) \text{ is an } \ell^{\text{th}} \text{ order polynomial between the } t_k.$$

For $\ell \geq 1$, the spline function is continuous and will have $\ell - 1$ derivatives which are continuous at the t_k.

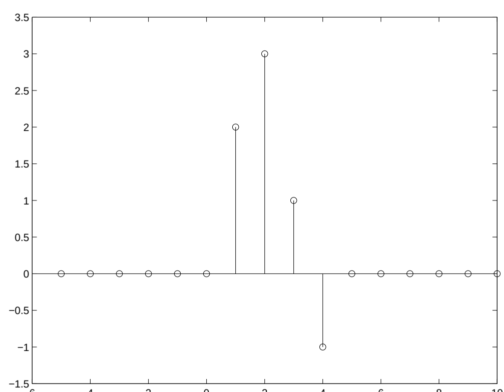
For example, suppose we have data points at the integers

$$t_1 = 1,\ t_2 = 2,\ t_3 = 3,\ t_4 = 4$$

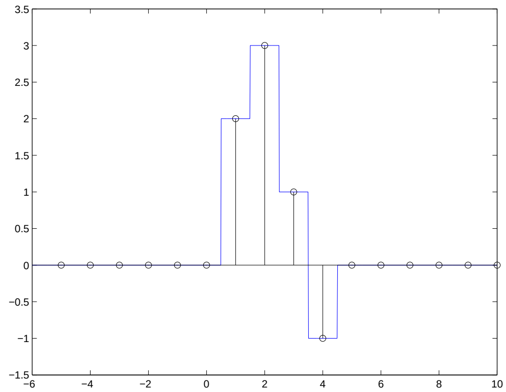
with values

$$v_1 = 2,\ v_2 = 3,\ v_3 = 1,\ v_4 = -1$$

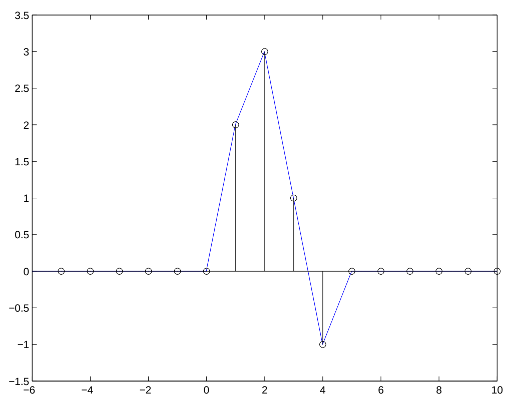
and values of zeros at the other integers. This is illustrated below.



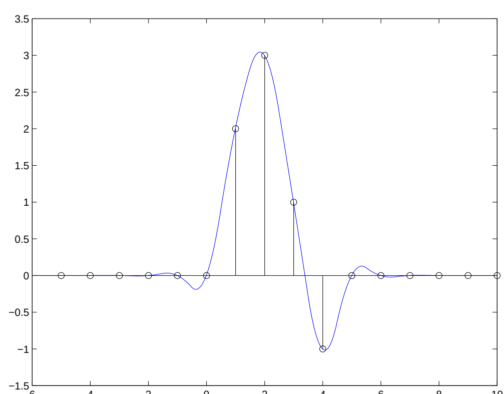
The zero-th order interpolation is:



The linear interpolation is:



The quadratic interpolation is:



Any ℓ^{th} order polynomial spline can be written as a superposition of B-spline functions (the 'B' is for basis!).

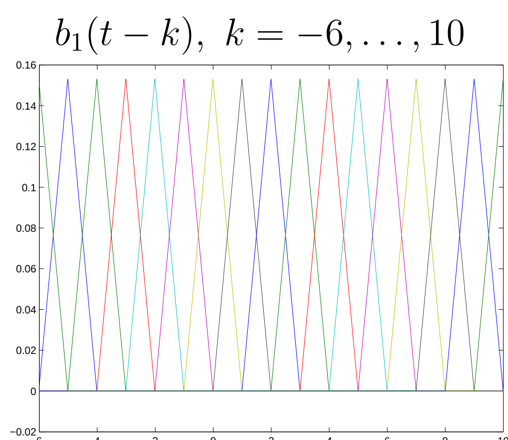
The piecewise constant function above can be written as

$$x(t) = \sum_{k=1}^{4} \alpha_k \, b_0(t-k), \quad b_0(t) = \begin{cases} 1 & -1/2 \le t < 1/2 \\ 0 & \text{else} \end{cases}$$

for $\alpha_1 = 2,\ \alpha_2 = 3,\ \alpha_3 = 1,\ \alpha_4 = -1$. The piecewise linear function above can be written as

$$x(t) = \sum_{k=1}^{4} \alpha_k \, b_1(t-k), \quad b_1(t) = \begin{cases} t+1 & -1 \le t \le 0 \\ 1-t & 0 \le t \le 1 \\ 0 & \text{else} \end{cases}$$

for $\alpha_1 = 2,\ \alpha_2 = 3,\ \alpha_3 = 1,\ \alpha_4 = -1$. In this case, the building blocks $b_1(t)$ are 'hat' functions:



For spline expansions using an order greater than 1, the expansion coefficients α_k will not be equal to the sample values. However, given a set of M samples values, the α_k that interpolate these samples can be found by solving a system of equations.

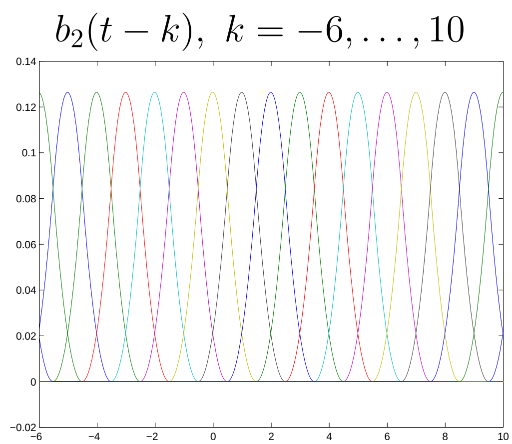
Notice that we can generate b_1 by convolving b_0 with itself:

$$b_1(t) = (b_0 * b_0)(t).$$

The expansion for the piecewise quadratic spline above is a little more complicated:

$$x(t) = \sum_{k=-\infty}^{\infty} \alpha_k \, b_2(t-k), \quad b_2(t) = \begin{cases} (t+3/2)^2/2 & -3/2 \le t \le -1/2 \\ -t^2 + 3/4 & -1/2 \le t \le 1/2 \\ (t-3/2)^2/2 & 1/2 \le t \le 3/2 \\ 0 & |t| \ge 3/2 \end{cases}$$

where the α_k are all non-zero. The expansion has an infinite number of terms because the basis functions overlap on the integers:



Just as before, we can generate the basis function $b_2(t)$ from the lower order ones:

$$b_2(t) = (b_1 * b_0)(t) = (b_0 * b_0 * b_0)(t).$$

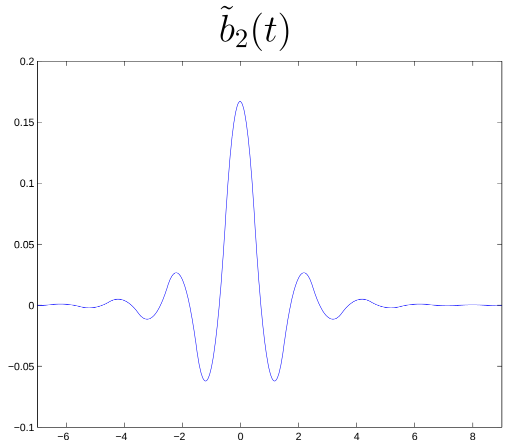
In general, any ℓ^{th} order polynomial spline $x(t)$ is uniquely represented by a list of numbers $\{\alpha_k, \; k \in \mathbb{Z}\}$, which correspond to the weights needed to re-synthesize the spline from the building blocks $\{b_\ell(t-k),\ k \in \mathbb{Z}\}$:

$$x(t) = \sum_{k=-\infty}^{\infty} \alpha_k b_\ell(t-k), \quad b_\ell(t) = \underbrace{b_0(t) * \cdots * b_0(t)}_{\ell \text{ times}}$$

If we are given an ℓ^{th} order spline $x(t)$, there is a systematic way to compute the corresponding α_k (and hence gives us another reproducing formula). Without getting too much into the details at this point, there is a complementary function $\tilde{b}_\ell(t)$ such that

$$\alpha_k = \int_{-\infty}^{\infty} x(t)\tilde{b}_\ell(t-k)\ \mathrm{d}t.$$

In the quadratic case $\ell = 2$, this function looks kind of like a sinc:



We know how to compute these complementary $\tilde{b}_\ell(t)$, but they are not easy to write down as nice expressions.

Bases and discretization

All of the examples above have a common theme: we take a signal in a certain class (bandlimited, zero outside of $[0, T]$, polynomial spline) and represent it using a discrete list of numbers $\{\alpha_k,\ k \in \mathbb{Z}\}$.

These numbers represent weights used to build up the signal out of a set of pre-determined building blocks ("basis functions"). This framework gives us a systematic way to manipulate continuous time signals by operating on discrete vectors. This allows us to unleash the power of **linear algebra**.

It often times also gives us a straightforward way to simply or compress signals. As you can see from the sawtooth Fourier series example, although it technically takes an infinity of sinusoids to build up the signals, we can get away with 50 if we are willing to suffer some loss. We will see some more examples of this later in this section.

In this set of notes, we have just gotten our first taste of basis expansions. What we will do next is develop a systematic method for taking a signal and breaking it down into a superposition of basis functions. We will also discuss how to optimally approximate a function using a fixed number of basis functions — this simple idea has an incredible number of applications.

To do these things correctly, we need to first build up some mathematical machinery so we can avoid talking in hazy terms. We start in the next section with precise (but abstract) definitions of **linear vector space**, **norm**, and **inner product**.