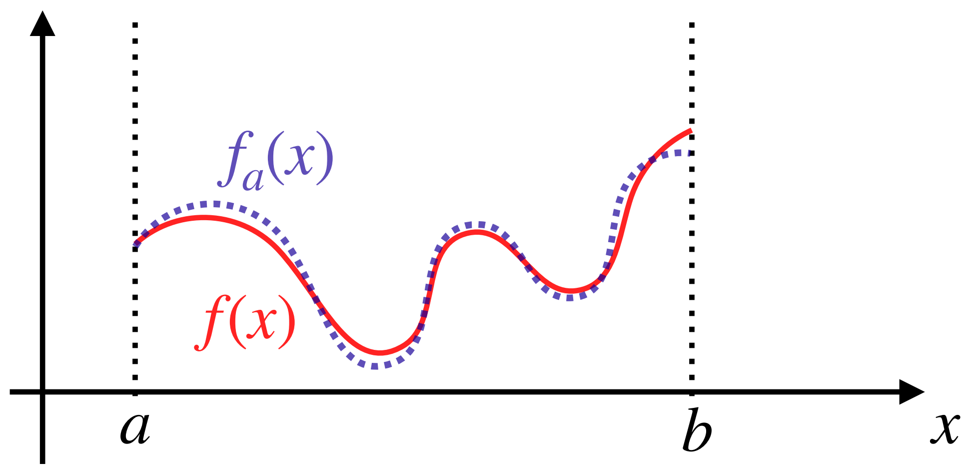
Lecture 4: Basis Functions

Today:

- More key concepts — ***basis***
- ***Interpolation*** — our first method for function approximation $f_a(x)$!

Reminder from last week:
Where are we with approximating functions?

Goal: find a function, $f_a(x)$, that approximates a given function, $f(x)$, *accurately* on $x \in [a, b]$



Where are we in solving this goal?

(A) We found that $f(x)$ belongs to a ***vector space,*** and that by selecting a ***subspace*** we can create a protocol for computing $f_a(x)$ by solving for a finite number of scalar coefficients

(B) We developed the ***norm*** to measure the error in our approximation

What do we still need to do?

(A) We didn't give many details about how to go from a ***subspace*** to these scalar coefficients. How does that work?

We will use the concept of a ***basis*** to explain this

(B) How do we solve for the coefficients?

We will cover multiple approaches. Today: ***interpolation***

$\Longrightarrow$ require f_a match f at a specific set of points

Basis

What is a basis?

Definition: A basis $\mathscr{B}$ for a vector space $\mathscr{V}$ is a set $\{b_0, b_1, \ldots, b_n\}$ that satisfies

(A) $b_0, b_1, \ldots, b_n \in \mathscr{V}$

(B) *Any* other $u \in \mathscr{V}$ can be written as a linear combination of the basis elements:

$$u = c_0 b_0 + c_1 b_1 + \cdots + c_n b_n$$

(C) *None* of the basis elements can be written as a linear combination of the other basis elements:

$$b_j = \sum_{\substack{k=0 \\ k \neq j}}^{n} c_k b_k \text{ if and only if } c_k = 0 \quad \forall k \neq j$$

A vector space can have multiple bases, but all of its bases have the same number of elements.

As with many mathematical definitions, examples are helpful to understanding. Let's consider a few…

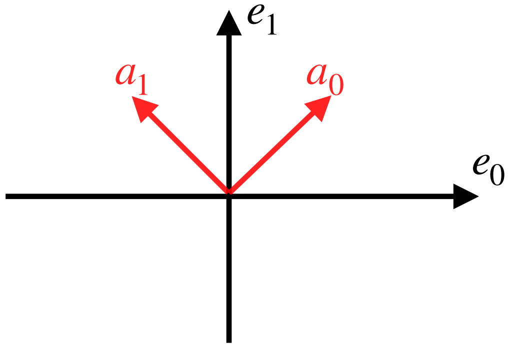
Example: bases in $\mathbb{R}^2$

Consider the vector space $\mathscr{V} = \mathbb{R}^2$

The set of all 2x1 real-valued vectors

Example: $\begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} -13 \\ 212 \end{bmatrix} \in \mathbb{R}^2$

Is the set $\mathscr{A} = \left\{ \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}_{a_0}, \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{a_1} \right\}$ a basis for $\mathscr{V}$? Justify with sketches and words.



Yes. It is clear from the figure that:

- Any vector in $\mathscr{V}$ can be written as a linear combo of a_0, a_1
- a_0 and a_1 are not co-linear

Activity: more examples with bases in $\mathbb{R}^2$

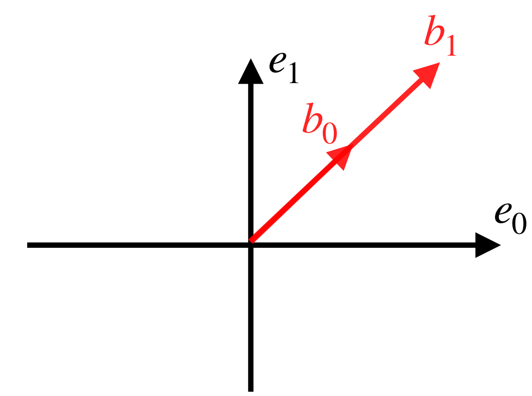
(A) Is the set $\mathscr{B} = \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \end{bmatrix} \right\}$ a basis for $\mathscr{V}$? Justify with sketches and words.

(B) Provide one other basis for $\mathscr{V}$. Justify that it is a basis using sketches and words.

(A) Is the set $\mathscr{B} = \left\{ \overset{b_0}{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}, \overset{b_1}{\begin{bmatrix} 2 \\ 2 \end{bmatrix}} \right\}$ a basis for $\mathscr{V}$? Justify with sketches and words.

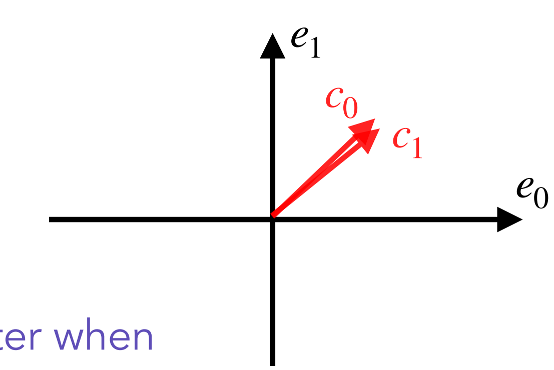
No. It is clear from the figure that:

- Not all vectors in $\mathscr{V}$ can be written as a linear combo of b_0, b_1
    - (e.g., consider a_1)
- b_0 and b_1 are co-linear



(B) Find another basis for $\mathscr{V}$? Justify with sketches and words.

Let's be nefarious and choose $\mathscr{C} = \left\{ \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0.99999999 \\ 1 \end{bmatrix} \right\}$



- Any vector in $\mathscr{V}$ can ***technically*** be written as a linear combo of c_0, c_1
- c_0 and c_1 are ***technically*** not co-linear $\longleftarrow$ But they nearly are $\Longrightarrow$ recipe for disaster when performing computations (HW 1)

Example: bases for function spaces

Consider the vector space $\mathscr{V} = \mathscr{P}^n[a, b]$

The set of all polynomials of degree n or less

Is $\mathscr{A} = \{1, x, x^2, \ldots, x^n\}$ a basis for $\mathscr{V}$? Give some intuition behind this answer.

Monomial basis

Yes.

- Each element in $\mathscr{A}$ belongs to $\mathscr{V} = \mathscr{P}^n[a, b]$
- We learned in algebra that any degree-n polynomial can be written as $c_0 + c_1 x + c_2 x^2 + \cdots + c_n x^n$
- The basis functions are also clearly linearly independent (e.g., no way I can write x as a linear combo of the other basis functions!)

Activity: more with bases for function spaces

$\mathscr{V} = \mathscr{P}^n[a, b]$

(A) Is $\mathscr{B} = \{1, x, x^2, \ldots, x^{n-1}\}$ a basis for $\mathscr{V}$? Give some intuition behind this answer.

No.

- While the functions belonging to $\mathscr{B}$ are linearly independent, not every degree-n polynomial can be written as a linear combination of those functions. Example: $f(x) = x^n$

(B) Provide a new basis for $\mathscr{V}$. Give some intuition for your answer.

- Another option is $\mathscr{C} = \left\{42, 3x, 52x^2, \ldots, \frac{1}{32}x^n\right\}$
- Rationale is similar to that from the last slide

Math appreciation break: Concepts like subspace, basis, and inner product can feel unnecessarily abstract. But they aren't! These concepts give us a way to treat vectors and functions using ***the same*** framework!

Interpolation

Why did we define a basis?

Return to the function approximation problem.

(A) We want to approximate a function that belongs to a ***vector space***, e.g., $f(x) \in C[a,b]$

(B) We can pick a ***subspace*** of that vector space that has a finite number of ***basis*** vectors, e.g., subspace $\mathscr{W} = \mathscr{P}^n[a,b]$ with basis $\mathscr{B} = \{1, x, x^2, \ldots, x^n\}$

More generally, $\mathscr{B} = \{b_0(x), b_2(x), \ldots, b_n(x)\}$

(C) Then if we require that our approximation function, $f_a(x)$, belong to the subspace, it *must* be able to be written as a linear combo of the basis vectors, e.g.,

$f_a(x) = c_0 + c_1 x + \cdots + c_n x^n$

More generally, $f_a(x) = c_0 b_0(x) + c_1 b_1(x) + \cdots + c_n b_n(x)$

(D) The name of the game: solving for a finite number of scalar numbers.

Computers can do that!

How do we solve for the coefficients?

Solving for the unknown coefficients

We have a functional form for $f_a(x)$: $f_a(x) = c_0 b_0(x) + c_1 b_1(x) + \cdots + c_n b_n(x)$

- We need $n+1$ equations for the $n+1$ unknowns.
- There are multiple ways to obtain these equations: ***interpolation*** and ***least squares***
- We will cover both. First: ***interpolation***.

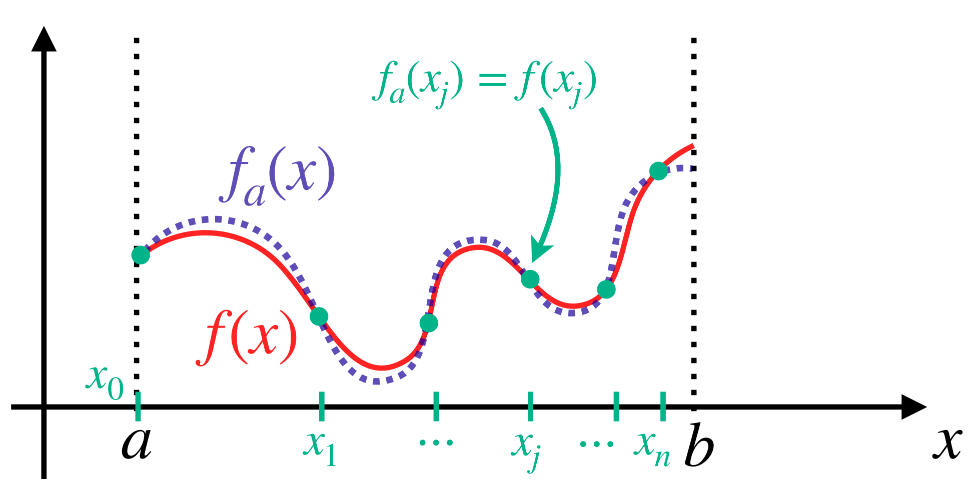
(A) Break domain $[a, b]$ up into $n+1$ interpolation points

(B) Require: approximate function equals the true function at the interpolation points

$$f_a(x_j) = f(x_j), \quad j = 0, \ldots, n$$

$$\implies \sum_{k=0}^{n} c_k b_k(x_j) = f(x_j), \quad j = 0, \ldots, n$$

$$\implies \begin{bmatrix} b_0(x_0) & b_1(x_0) & \cdots & b_{n-1}(x_0) & b_n(x_0) \\ b_0(x_1) & b_1(x_1) & \cdots & b_{n-1}(x_1) & b_n(x_1) \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ b_0(x_{n-1}) & b_1(x_{n-1}) & \cdots & b_{n-1}(x_{n-1}) & b_n(x_{n-1}) \\ b_0(x_n) & b_1(x_n) & \cdots & b_{n-1}(x_n) & b_n(x_n) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_{n-1}) \\ f(x_n) \end{bmatrix}$$

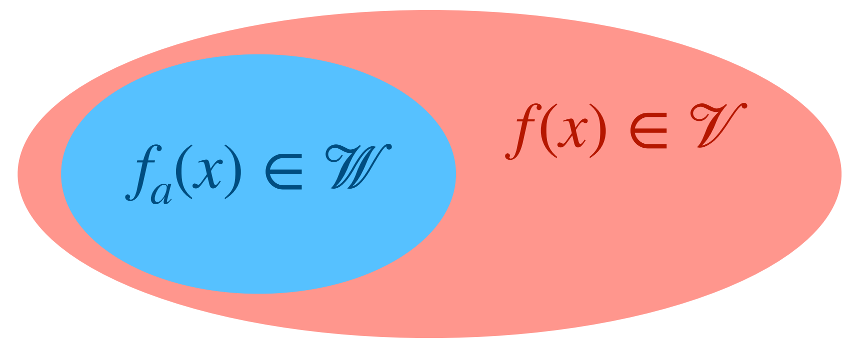


(C) Solve linear system for the coefficients $c_0, \ldots, c_n$

(D) We now have our interpolant, $f_a(x)$!

Summary of function approximation concepts

Function approximation concepts…



Vector space — used to represent $f(x)$

Subspace — used to restrict representation of $f_a(x)$

Inner product — used to define a ***norm*** — used to measure approximation error

Basis functions — used to represent $f_a(x)$ such that we can solve for coefficients

$$||e|| = \sqrt{(e, e)} \qquad f_a(x) = c_0 b_0(x) + c_1 b_1(x) + \cdots + c_n b_n(x)$$

Function approximation methods…

How do we choose $b_0(x), b_1(x), \ldots, b_n(x)$ and solve for $c_0, c_1, \ldots, c_n$?

$n+1$ equations and $n+1$ unknowns

Interpolation

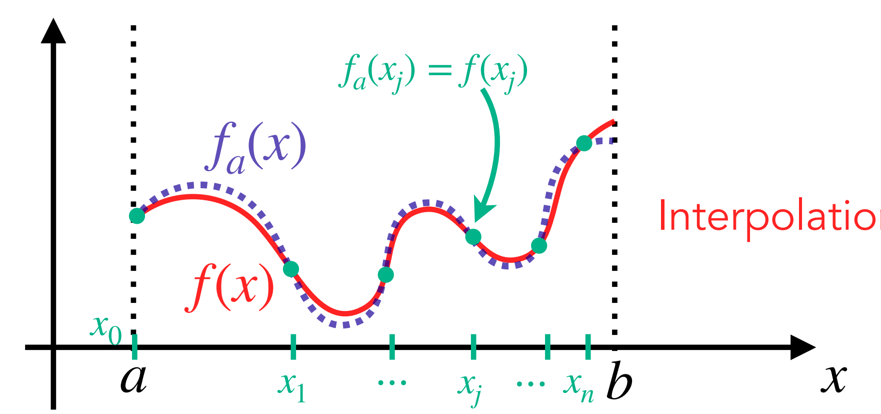
Polynomial interpolation, $f_a(x) \in \mathscr{P}^n[a, b]$

Global: monomial basis, Lagrange basis

Local: cubic spline

Trigonometric interpolation, $f_a(x) \in \mathscr{T}^n[a, b]$

Least Squares



$f(x) \in C[a, b] \qquad f_a(x) \in \mathscr{P}^n[a, b]$

$$f_a(x) = c_0 + c_1 x + \cdots + c_n x^n$$