Floating Point

sign bit   exp.   significand
"mantissa"

float: 8 exp, 23 sig

double: 11 exp, 52 sig

$4\cdot(2^{10})^5 = 4\cdot 1024^5 \approx 4\cdot 10^{15}$ | $2^{52} \approx 10^x$

$$.1 = \frac{1}{16} + \frac{1}{32} + \dots$$

$$.1 > \frac{3}{32}$$

$$1.\underbrace{000}1 \cdot 10^3$$

1000

$$1.\underbrace{\underbrace{\quad}_{9\text{ bits}}\underbrace{\quad}_{52-9=43}} \leftarrow \text{significand}$$

$0.0000000020\underbrace{xyz\ldots}_{43}$

$\underbrace{\text{accurate sig. is 43}} \longleftarrow$ 9 bits

$1000.1 = 1.\underbrace{\qquad\qquad}_{sig} \cdot 2^9$

exp	sign	signif.	type
0 ... 0	matters		subnormal
0 0	+/-	000	
111111	+/-	000	infinity
111111		$\neq 0$	NaN

$$\frac{0}{0} = \text{NaN} = 0 \cdot \infty$$

$$x.yz \cdot 10^{b} \cdot a.bc \cdot 10^{d}$$

$$\underbrace{(x.yz \cdot a.bc)}\cdot 10^{b+d}$$

$$\left|\left(\text{true answer} - x.yz \cdot a.bc\right)\right|$$

$$\leq 10^{-2} \cdot a.bc + 10^{-2} \cdot x.yz$$

$$\leq O\left(10^{-2}\right)$$

Outline

Outline

Some Perspective

- We have so far (mostly) looked at what we can do with single numbers (and functions that return single numbers).
- Things can get *much* more interesting once we allow not just one, but *many* numbers together.
- It is natural to view an *array of numbers* as one object with its own rules.

  The simplest such set of rules is that of a vector.
- A 2D array of numbers can also be looked at as a matrix.
- So it's natural to use the tools of computational linear algebra.
- 'Vector' and 'matrix' are just *representations* that come to life in many (*many!*) applications. The purpose of this section is to explore some of those applications.

Vectors

What's a vector?

$c = a + b$

vectors

$c = \alpha \cdot a$

scalar

$c = a \otimes b$

$c_i = a_i \cdot b_i$

$d = \sum_i a_i \cdot b_i$

properties

$(a+b)+c = a+(b+c)$

$a+b = b+a$

$\alpha(a+b)$
$= \alpha a + \alpha b$

$c_{ij} = a_i \cdot b_j$

Vectors from a CS Perspective

What would the concept of a vector look like in a programming language (e.g. Java)?

interface vector

op add → scalars

op mul

vector vadd(vector a, vector b)

vector vscale(scalar a, vector b)

Vectors in the ‘Real World’

Demo: Images as Vectors
Demo: Sound as Vectors
Demo: Shapes as Vectors

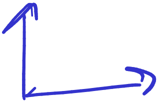
Outline

Matrices

What does a matrix do?

It represents a *linear function* between two vector spaces $f : U \to V$ in terms of bases $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n$ of U and $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$ of V. Let

$$\boldsymbol{u} = \alpha_1 \boldsymbol{u}_1 + \cdots + \alpha_n \boldsymbol{u}_n$$

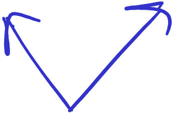


and

$$\boldsymbol{v} = \beta_1 \boldsymbol{v}_1 + \cdots + \beta_m \boldsymbol{v}_m.$$

Then f can *always* be represented as a matrix that obtains the βs from the αs:

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_m \end{pmatrix}.$$



Example: The 'Frequency Shift' Matrix

Assume both $\boldsymbol{u}$ and $\mathbf{v}$ are linear combination of sounds of different frequencies:

$$\boldsymbol{u} = \alpha_1 \boldsymbol{u}_{110\text{ Hz}} + \alpha_2 \boldsymbol{u}_{220\text{ Hz}} + \cdots + \alpha_4 \boldsymbol{u}_{880\text{ Hz}}$$

(analogously for $\boldsymbol{v}$, but with βs). What matrix realizes a 'frequency doubling' of a signal represented this way?

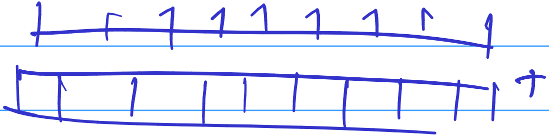
$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \begin{bmatrix} 0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}$$

Example: The 'Frequency Shift' Matrix

Assume both $\boldsymbol{u}$ and $\mathbf{v}$ are linear combination of sounds of different frequencies:

$$\boldsymbol{u} = \alpha_1 \boldsymbol{u}_{\text{110 Hz}} + \alpha_2 \boldsymbol{u}_{\text{220 Hz}} + \cdots + \alpha_4 \boldsymbol{u}_{\text{880 Hz}}$$

(analogously for $\boldsymbol{v}$, but with βs). What matrix realizes a 'frequency doubling' of a signal represented this way?

$V =$ 

$$\begin{bmatrix} -1 & 0 & 1 & 0 & 0 & \cdots \\ 0 & -1 & 0 & 1 & 0 & 0 \cdots \\ 0 & 0 & -1 & 0 & 1 & \cdots \\ & & & & & \end{bmatrix} V = V[2:] - V[:-2]$$