Data analysis and modeling: the tools of the trade

Patrice Koehl
Department of Biological Sciences
National University of Singapore

http://www.cs.ucdavis.edu/~koehl/Teaching/BL5229
koehl@cs.ucdavis.edu

Tools of the trade

- Set of numbers
- Binary representation of numbers
- Floating points
- Digital sound
- Vectors and matrices

Tools of the trade

- Set of numbers
- Binary representation of numbers
- Floating points
- Digital sound
- Vectors and matrices

The different set of numbers

$\mathbb{N}$	Natural numbers	$1, 2, 3, 4 \ldots,$
$\mathbb{Z}$	Integers	$\ldots, -4, -3, -2, -1, 0, 1, 2, 3, 4, \ldots$
$\mathbb{Q}$	Rational numbers	$\frac{a}{b}$ where a and b are integers and b is not zero
$\mathbb{R}$	Real numbers	The limit of a convergent sequence of rational numbers
$\mathbb{C}$	Complex numbers	$a + ib$ where a and b are real numbers and i is the square root of -1

Tools of the trade

- Set of numbers
- Binary representation of numbers
- Floating points
- Vectors and matrices

Number representation

We are used to counting in base 10:

1000	100	10	1
10^3	10^2	10^1	10^0

..... thousands hundreds tens units

Example:

1	7	3	2
1000	100	10	1

← ***digits***

1x1000+7x100+3x10+2x1 = 1732

Number representation

Computers use a different system: base 2:

1024	512	256	128	64	32	16	8	4	2	1
2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Example:

1	1	0	1	1	0	0	0	1	0	0 ← *bits*
1024	512	256	128	64	32	16	8	4	2	1

1x1024+1x512+0x256+1x128+1x64+0x32+ 0x16+ 0x8 + 1x4 + 0x2 + 0x1 = 1732

Number representation

Base 10	Base 2
0	0
1	1
2	10
3	11
4	100
5	101
6	110
…	…
253	11111101
254	11111110
255	11111111
…	…

Conversion

From base 2 to base 10:

1	1	1	0	1	0	1	0	1	0	0
1024	512	256	128	64	32	16	8	4	2	1

1x1024+1x512+1x256+0x128+1x64+0x32+ 1x16+ 0x8 + 1x4 + 0x2 + 0x1 = 1876

From base 10 to base 2:

1877 %2 = 938 Remainder 1
938 %2 = 469 Remainder 0
469 %2 = 234 Remainder 1
234 %2 = 117 Remainder 0
117 %2 = 58 Remainder 1
58 %2 = 29 Remainder 0
29 %2 = 14 Remainder 1
14 %2 = 7 Remainder 0
7 %2 = 3 Remainder 1
3 %2 = 1 Remainder 1
1 %2 = 0 Remainder 1

1877 (base10) = 11101010101 (base 2)

Tools of the trade

- Set of numbers
- Binary representation of numbers
- Floating points
- Digital sound
- Vectors and matrices

IEEE Floating Point

- **IEEE Standard 754**
  - Established in 1985 as uniform standard for floating point arithmetic
    Before that, many idiosyncratic formats
  - Supported by all major CPUs
- **Driven by Numerical Concerns**
  - Nice standards for rounding, overflow, underflow
  - Hard to make go fast
    - Numerical analysts predominated over hardware types in defining standard

Floating Point Representation

- Numerical Form

  $-1^s\ M\ \ 2^E$
  - Sign bit ***s*** determines whether number is negative or positive
  - Significand ***M*** normally a fractional value in range [1.0,2.0).
  - Exponent ***E*** weights value by power of two
- Encoding

s	exp	frac

  - MSB is sign bit
  - `exp` field encodes *E*
  - `frac` field encodes *M*

Floating Point Precisions

s	exp	frac

- Encoding
  - MSB is sign bit
  - `exp` field encodes *E*
  - `frac` field encodes *M*
- Sizes
  - Single precision: 8 `exp` bits, 23 `frac` bits
    (32 bits total)
  - Double precision: 11 `exp` bits, 52 `frac` bits
    (64 bits total)
  - Extended precision: 15 `exp` bits, 63 `frac` bits
    - Only found in Intel-compatible machines
    - Stored in 80 bits (1 bit wasted)

Special Values

Condition
`exp = 111…1`

- `exp = 111…1, frac = 000…0`
  - Represents value ∞ (infinity)
  - Operation that overflows
  - Both positive and negative
  - E.g., 1.0/0.0 = -1.0/-0.0 = +∞, 1.0/-0.0 = -∞
- `exp = 111…1, frac ≠ 000…0`
  - Not-a-Number (NaN)
  - Represents case when no numeric value can be determined
  - E.g., sqrt(–1), ∞ - ∞

Floating Point Operations

- Conceptual View
  - First compute exact result
  - Make it fit into desired precision
    - Possibly overflow if exponent too large
    - Possibly round to fit into `frac`
- Rounding Modes (illustrate with $ rounding)

	\$1.40	\$1.60	\$1.50	\$2.50	–\$1.50
Round down (-∞)	\$1	\$1	\$1	\$2	–\$2
Round up (+∞)	\$2	\$2	\$2	\$3	–\$1
Nearest Even	\$1	\$2	\$2	\$2	–\$2

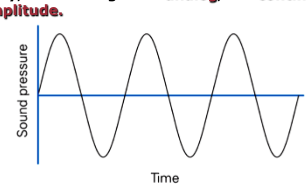
Note:
1. Round down: rounded result is close to but no greater than true result.
2. Round up: rounded result is close to but no less than true result.

Tools of the trade

- Set of numbers
- Binary representation of numbers
- Floating points
- Digital sound
- Vectors and matrices

Digital Sound

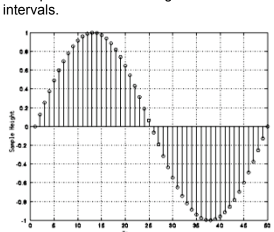
Sound is produced by the vibration of a media like air or water. Audio refers to the sound within the range of human hearing. Naturally, a sound signal is analog, i.e. continuous in both time and amplitude.



To store and process sound information in a computer or to transmit it through a computer network, we must first convert the analog signal to digital form using an analog-to-digital converter (ADC); the conversion involves two steps: (1) sampling, and (2) quantization.

Sampling

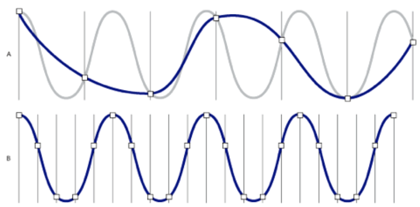
Sampling is the process of examining the value of a continuous function at regular intervals.



Sampling usually occurs at uniform intervals, which are referred to as sampling intervals. The reciprocal of sampling interval is referred to as the sampling frequency or sampling rate.
If the sampling is done in time domain, the unit of sampling interval is second and the unit of sampling rate is Hz, which means cycles per second.

Sampling

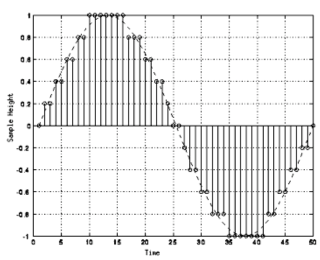
Note that choosing the sampling rate is not innocent:



A higher sampling rate usually allows for a better representation of the original sound wave. However, when the sampling rate is set to twice the highest frequency in the signal, the original sound wave can be reconstructed without loss from the samples. This is known as the Nyquist theorem.

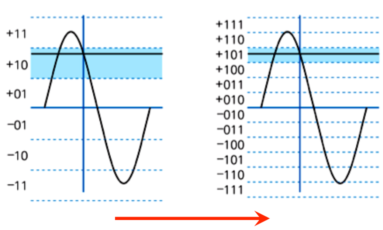
Quantization

Quantization is the process of limiting the value of a sample of a continuous function to one of a predetermined number of allowed values, which can then be represented by a finite number of bits.



Quantization

The number of bits used to store each intensity defines the accuracy of the digital sound:



Adding one bit makes the sample twice as accurate

Audio Sound

Sampling:

The human ear can hear sound up to 20,000 Hz: a sampling rate of 40,000 Hz is therefore sufficient. The standard for digital audio is 44,100 Hz.

Quantization:

The current standard for the digital representation of audio sound is to use 16 bits (i.e 65536 levels, half positive and half negative)

How much space do we need to store one minute of music?

- 60 seconds
- 44,100 samples
- 16 bits (2 bytes) per sample
- 2 channels (stereo)

S = 60x44100x2x2 = 10,534,000 bytes ≈ 10 MB !!

1 hour of music would be more than 600 MB !

Tools of the trade

- Set of numbers
- Binary representation of numbers
- Floating points
- Digital sound
- Vectors and matrices

Vectors

- Set of numbers organized in an array

$$v = (x_1, x_2, \ldots\ldots, x_n)$$

- Norm of a vector: size

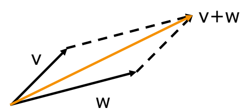
$$\|v\| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

If $\|v\| = 1$, v is a unit vector

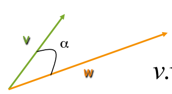
Example: (x,y,z), coordinates of a point in space.

Vector Addition

$$\mathbf{v}+\mathbf{w}=(x_1,x_2)+(y_1,y_2)=(x_1+y_1,x_2+y_2)$$



Inner (dot) Product



$$v.w=(x_1,x_2).(y_1,y_2)=x_1y_1+x_2.y_2$$

The inner product is a **SCALAR!**

$$v.w=(x_1,x_2).(y_1,y_2)=\|v\|\cdot\|w\|\cos\alpha$$

$$v.w=0\Leftrightarrow v\perp w$$

What is a matrix?

- MATRIX: A rectangular arrangement of numbers in rows and columns.
- The ORDER of a matrix is the number of the rows and columns.
- The ENTRIES are the numbers in the matrix.
- The order of this matrix is a 2 x 3.

columns ↓

rows →

$$\begin{bmatrix} 6 & 2 & -1 \\ -2 & 0 & 5 \end{bmatrix}$$

Matrix operations

- To add two matrices, they must have the same order. To add, you simply add corresponding entries.
- To subtract two matrices, they must have the same order. You simply subtract corresponding entries.
- To multiply a matrix by a scalar, you multiply each entry in the matrix by that scalar.
- To multiply two matrices A and B, A must have as many columns as B has rows.

Matrix operations

Matrix Addition:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} + \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} (a+e) & (b+f) \\ (c+g) & (d+h) \end{bmatrix}$$

Matrix Multiplication:

An (m x n) matrix A and an (n x p) matrix B, can be multiplied since the number of columns of A is equal to the number of rows of B.

Non-Commutative Multiplication

AB is NOT equal to BA

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} * \begin{bmatrix} e & f \\ g & h \end{bmatrix} = \begin{bmatrix} (ae+bg) & (af+bh) \\ (ce+dg) & (cf+dh) \end{bmatrix}$$

Matrices

Transpose:

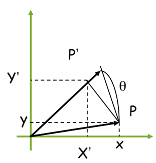
$$C_{m\times n} = A^{T}{}_{n\times m} \qquad (A+B)^T = A^T + B^T$$

$$c_{ij} = a_{ji} \qquad (AB)^T = B^T A^T$$

If $A^T = A$ A is symmetric

Applications of matrices: rotations

Counter-clockwise rotation by an angle θ



$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\mathbf{P' = R.P}$$

Applications of matrices: systems of equation

Let us consider the system:

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1$$
$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2$$
$$a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3$$

If we define:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

and

$$b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

The system becomes:

$$Ax = b$$

which is solved as:

$$x = A^{-1}b$$