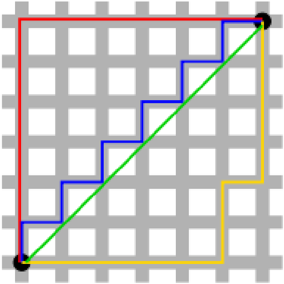
COMS20017 – Algorithms & Data



January 2025

Majid Mirmehdi

with a few slides from Rui Ponte Costa & Dima Damen

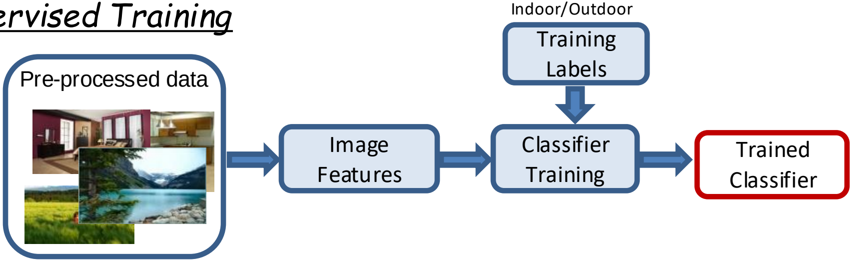
Image from Wikipedia

Lecture MM-02

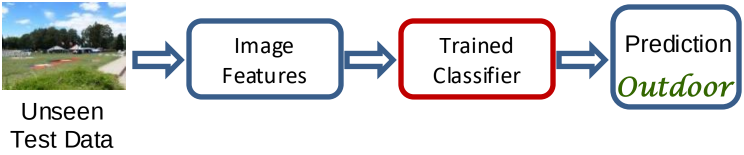
Last lecture: Typical Data Analysis Problem

1. Pre-processing
2. Feature Selection
3. Classification

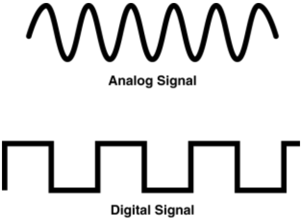
Supervised Training



Testing

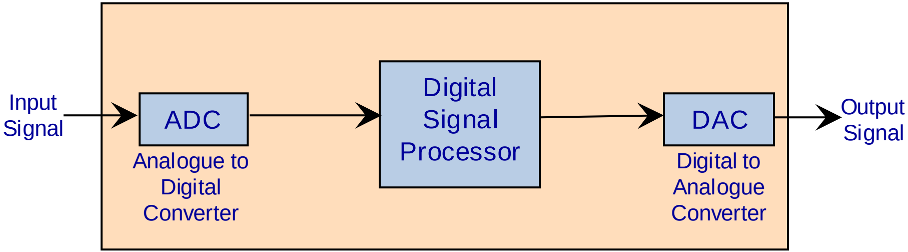


This lecture



- **Data acquisition**
- **Data characteristics: distance measures**
- Data characteristics: summary statistics [*reminder*]
- Data normalisation and outliers

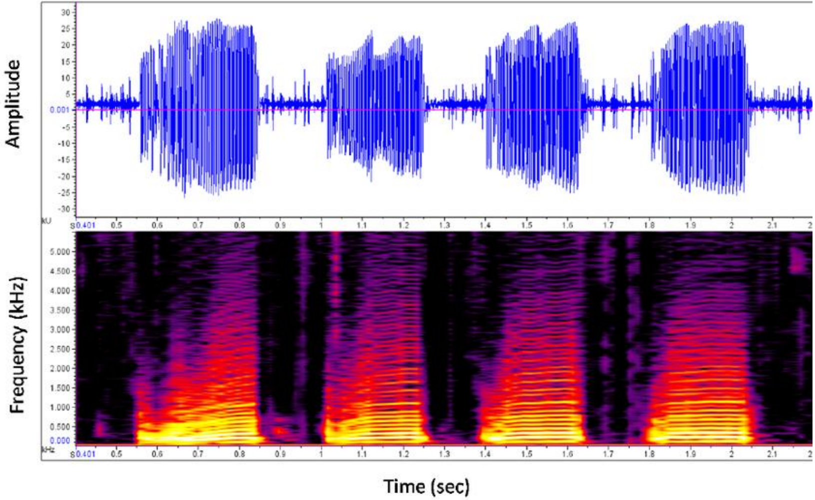
Data Acquisition – Example Data Journey



Data Acquisition - Analogue to Digital Conversion

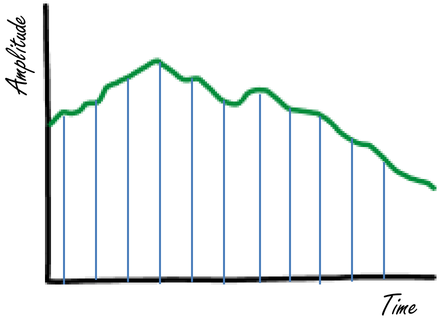
Analogue to Digital conversion involves *Sampling & Quantisation*

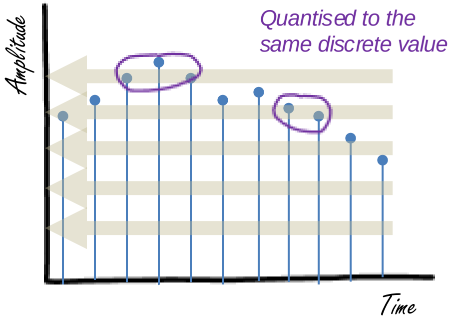
e.g. a 1D Audio Signal



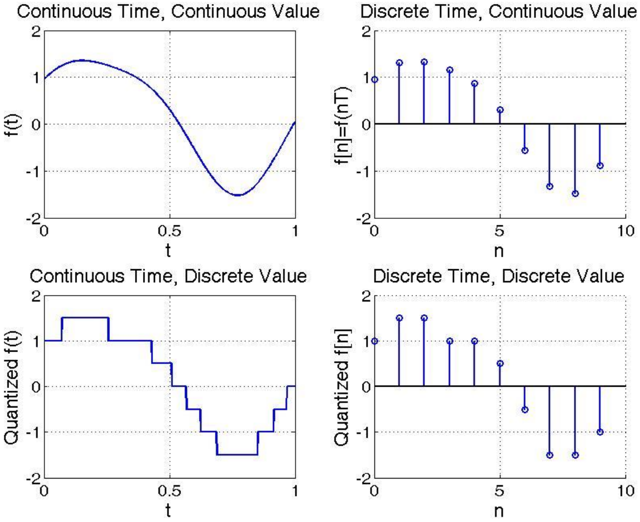
Data Acquisition - Analogue to Digital Conversion

Analogue to Digital conversion involves *Sampling & Quantisation*

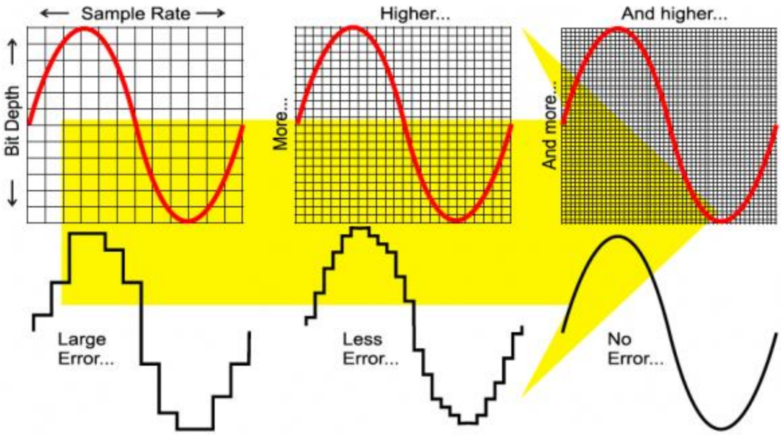




Sample and Quantise



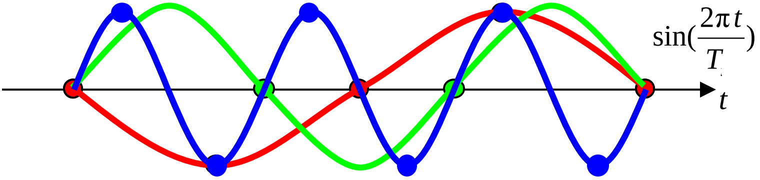
Sample and Quantise



Nyquist-Shannon Sampling Theory

"An analogue signal containing components up to some maximum frequency ***u*** *(Hz) may be completely reconstructed by regularly spread samples, provided the sampling rate is at least* ***2u*** *samples per second"*

Also referred to as the Nyquist-Shannon criterion: sampling rate s should be at least twice the highest spatial frequency u.

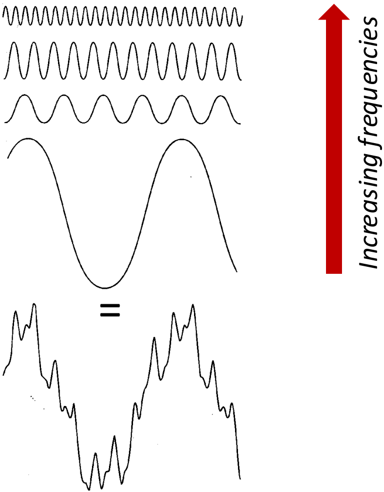


sampling period $T \leq \frac{1}{2u}$

equivalent to *sampling rate* $s \geq 2u$

Nyquist-Shannon Sampling Theory

"An analogue signal containing components up to some maximum frequency ***u*** *(Hz) may be completely reconstructed by regularly spread samples, provided the sampling rate is at least* ***2u*** *samples per second"*



Data Acquisition - Analogue to Digital Conversion

Examples of sampling and quantisation of standard audio formats:

- Speech (e.g. phone call)
  - Sampling: 8 KHz samples
  - Quantisation: 8 bits / sample
- Audio CD and Streaming
  - Sampling: 44 KHz samples
  - Quantisation: 16 bits / sample
  - Stereo (2 channels)

Higher sampling and quantisation levels achieves better signal quality, but at the expense of larger memory and storage.

Data Acquisition - Analogue to Digital Conversion

Examples of sampling and quantisation of Images (multi-dimensional):

- Sampling: Resolution in digital photography
- Quantisation: Representation of each pixel in the image
  - 8 Mega Pixel Camera: 3264 x 2448 pixels
  - Colour images: 3 channels - Red, Green, Blue (8 bits per colour)
  - Greyscale images: 1 channel - intensity $= aR+bG+cB$ *where $a+b+c=1.0$*
  - Binary images: Black/White - 1 bit per pixel

Sampling – visual example

The effect of sparser sampling…is ALIASING

256x256

64x64

32x32

Anti-aliasing is achieved by filtering to remove frequencies above the Nyquist limit.

Quantisation – visual example

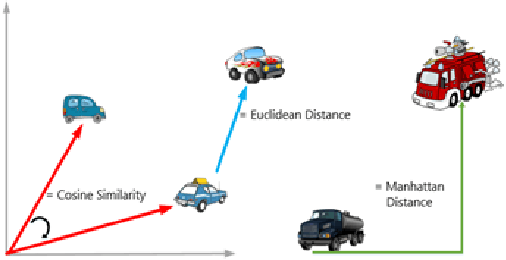
This results from representing a continuously varying function *f(x)* with a discrete one using quantisation levels

16 levels

6 levels

2 levels

Next…



- Data acquisition
- **Data characteristics: distance measures**
- Data characteristics: summary statistics [*reminder*]
- Data normalisation and outliers

Data Characteristics: Distance Measures

- Distance is measure of separation between data.
- Distance is important as it:
  - enables data to be ordered
  - allows numeric calculations
  - enables measuring similarity and dissimilarity
- Without defining a distance measure, almost all statistical and machine learning algorithms will not function!
- Can be defined between single-dimensional data, multi-dimensional data or data sequences.

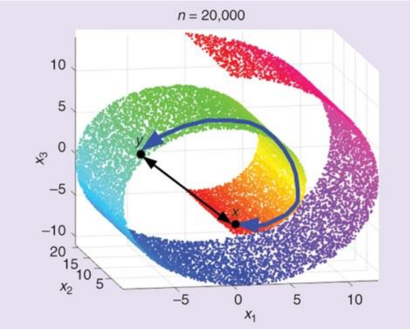
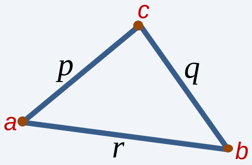


Image from https://www.embs.org/pulse/articles/what-is-the-distance-between-objects-in-a-data-set/

Properties for a Distance Measure

A valid distance measure $D(a,b)$ between two components a and b has the following properties

- non-negative: $D(a,b) \geq 0$
- symmetric: $D(a,b) = D(b,a)$
- reflexive: $D(a,b) = 0 \Longleftrightarrow a = b$
- satisfies triangular inequality: $D(a,b) \leq D(a,c) + D(c,b)$



$$p + q \geq r$$
$$p + r \geq q$$
$$q + r \geq p$$

Distance (Numerical)

To find the distance between numerical data points $\mathbf{x} = (x_1, x_2, .., x_n)$ and $\mathbf{y} = (y_1, y_2, .., y_n)$ in Euclidean space $\mathbb{R}^n$, the **Minkowski Distance** of order p (p-$norm$ distance) is defined as:

$$D(\mathbf{x}, \mathbf{y}) = (\sum_{i=1}^{n} |x_i - y_i|^p)^{\frac{1}{p}}$$

- $p = 1$
- 1-norm distance (L_1)

$$D(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i|$$

- Also known as the *Manhattan Distance*
- *Not the shortest path possible…*

Distance (Numerical)

To find the distance between numerical data points $\mathbf{x} = (x_1, x_2, .., x_n)$ and $\mathbf{y} = (y_1, y_2, .., y_n)$ in Euclidean space $\mathbb{R}^n$, the **Minkowski Distance** of order p (p-$norm$ distance) is defined as:

$$D(\mathbf{x},\mathbf{y}) = (\sum_{i=1}^{n} |x_i - y_i|^p)^{\frac{1}{p}}$$

- $p = 2$
- 2-norm distance (L_2)
- Also known as the *Euclidean Distance*

$$D(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2}$$

Can be expressed in vector form:

$$D(\mathbf{x},\mathbf{y}) = \| \mathbf{x} - \mathbf{y} \|$$
$$= \sqrt{(\mathbf{x} - \mathbf{y})^T(\mathbf{x} - \mathbf{y})}$$

Distance (Numerical)

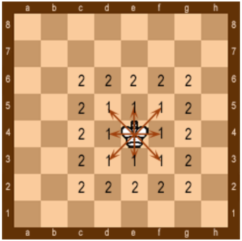
To find the distance between numerical data points $\mathbf{x} = (x_1, x_2, .., x_n)$ and $\mathbf{y} = (y_1, y_2, .., y_n)$ in Euclidean space $\mathbb{R}^n$, the **Minkowski Distance** of order p (p-$norm$ distance) is defined as:

$$D(\mathbf{x}, \mathbf{y}) = (\sum_{i=1}^{n} |x_i - y_i|^p)^{\frac{1}{p}}$$

- $p = \infty$
- ∞-norm distance (L_∞)
- Also known as the *Chebyshev Distance*

$$D(\mathbf{x}, \mathbf{y}) = \lim_{p \to \infty} (\sum_{i=1}^{n} |x_i - y_i|^p)^{\frac{1}{p}}$$

$$= max\,(|x_1 - y_1|, |x_2 - y_2|, \dots, |x_n - y_n|)$$



If 2 dimensions where two points have cartesian coordinates, then $D = max\,(|x_2 - x_1|, |y_2 - y_1|)$

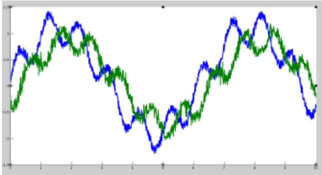
Image from https://www.translatorscafe.com/

Distance (Numerical Series)

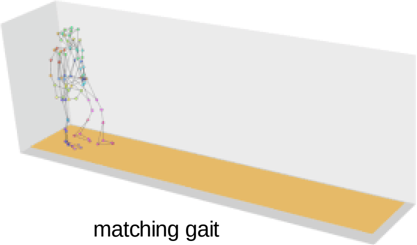
- Time Series: successive measurements made over a time interval

p-norm distances can only

- compare time series of the same length
- very sensitive to signal transformations:
  - shifting
  - amplitude scaling
  - uniform time scaling

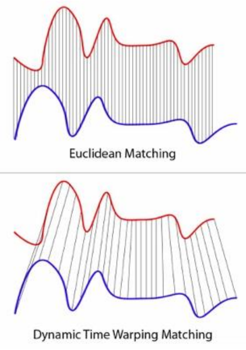


matching audio signal of two people saying the same word



matching gait

Distance (Numerical Time Series)



Dynamic Time Warping (Berndt and Clifford, 1994)

- Replaces Euclidean one-to-one comparison with many-to-one
- Recognises similar shapes even in the presence of shifting, length, and scaling
- Dynamic Time Warping (DTW) can be defined ***recursively*** to tell us how two signals align with each other:

For two time series $\boldsymbol{X} = (x_1, .., x_n)$ and $\boldsymbol{Y} = (y_1, .., y_m)$

$$DTW(\boldsymbol{X}, \boldsymbol{Y}) = D(x_1, y_1) + \min\{DTW(\boldsymbol{X}, REST(\boldsymbol{Y})), DTW(REST(\boldsymbol{X}), \boldsymbol{Y}), DTW(REST(\boldsymbol{X}), REST(\boldsymbol{Y}))\}$$

where $REST(\boldsymbol{X}) = (x_2, .., x_n)$

DTW builds a distance matrix between two time series and then finds the minimum path (alignment cost) for an optimal match.
OPTIONAL: for more details, watch: https://www.youtube.com/watch?v=ERKDHZyZDwA (2 parts)

Distance (Symbolic)

- Distance is not always between numerical data
- Distance between symbolic data is less well-defined (e.g. text data)
- Distance in text could be:
  - syntactic
  - semantic

I will send you some **cash**

Distance (Symbolic)

Syntactic - e.g. Hamming Distance

- Defined over symbolic data of *the same* length
- Measures the number of substitutions required to change one string/number into another

  - *B r i s t o l*
    B u r t t o n D('Bristol', 'Burtton') = 4

  - 5 2 4 3
    6 2 1 3 D(5243, 6213) = 2

  - 1011101
    1001001 D(1011101, 1001001) = 2

- Used in coding theory and error correcting codes
- For binary strings, Hamming Distance is the same as taking L_1 absolute difference of bits and summing them

Distance (Symbolic)

Syntactic - e.g. Edit Distance

- Defined on text data of *any* length
- Measures the *minimum* number of ‘operations’ required to transform one sequence of characters into another
- ‘Operations’ can be: insertion, substitution, deletion

- e.g. D(‘fishing’, ‘first’) = 5

‘fishing’ $\xrightarrow{\textit{insertion}}$ ‘firshing’ $\xrightarrow{\textit{substitution}}$ ‘firsting’ $\xrightarrow{\textit{3x deletion}}$ ‘first’

- used in spelling correction, DNA string comparisons, etc.

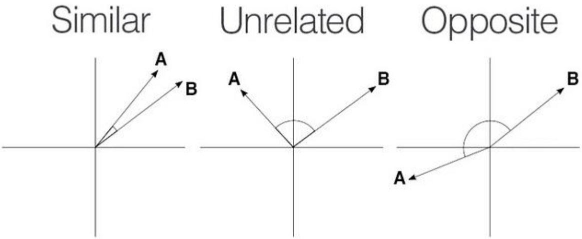
Distance

Cosine Similarity and Cosine Distance

- Cosine similarity is a metric that determines how **two vectors** (words, sentences, features) are **similar** to each other.
- Cosine distance measures how **different** two vectors are.

$$Similarity(A,B) = \cos(\theta) = \frac{A.B}{\parallel A \parallel \parallel B \parallel}$$

$$Distance(A,B) = 1 - Similarity(A,B)$$



Cosine Similarity/Distance – Example 1

Find Cosine Similarity and Distance between two vectors

x = { 4, 2, 0, 3 } and **y = { 1, 0, 1, 0 }**

$$\mathbf{x} \,.\, \mathbf{y} = 4 * 1 \; + \; 2 * 0 \; + \; 0 * 1 \; + \; 3 * 0$$

$$\| \, \mathbf{x} \, \| = \sqrt{16 + 4 + 0 + 9} = 5.385$$

$$\| \, \mathbf{y} \, \| = \sqrt{1 + 0 + 1 + 0} = 1.414$$

$$Similarity(\mathbf{x}, \mathbf{y}) = \frac{4}{5.385 * 1.414} = 0.525$$

$$Distance(\mathbf{x}, \mathbf{y}) = 1 - 0.525 = 0.475$$

Cosine Similarity/Distance – Example 2

Find Cosine Similarity and Distance between two documents that contain these words:

Doc1 = {Computer Science at Bristol is simple}

Doc2 ={Amongst all courses Computer Science isn't simple}

Generate vectorised representations of the texts:

Amongst	*all*	*courses*	*Computer*	*Science*	*at*	*Bristol*	*is*	*isn't*	*simple*

Doc1 = $\mathbf{x} = \{0,0,0,1,1,1,1,1,0,1\}$

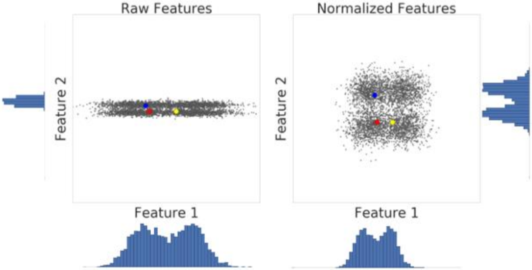
Doc2 = $\mathbf{y} = \{1,1,1,1,1,0,0,0,1,1\}$

$$\mathbf{x} \,.\, \mathbf{y} = 3$$
$$\| \mathbf{x} \| = \sqrt{6}$$
$$\| \mathbf{y} \| = \sqrt{7}$$

$$Similarity(\mathbf{x}, \mathbf{y}) = \frac{3}{\sqrt{42}} = 0.463$$

$$Distance(\mathbf{x}, \mathbf{y}) = 1 - 0.463 = 0.537$$

Next lecture



- Data acquisition
- Data characteristics: distance measures
- **Data characteristics: summary statistics [*reminder*]**
- **Data normalisation and outliers**