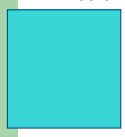
Vector model

- Images:
  - color histogram, shape, edges, etc.



2D Matrix

Vector model

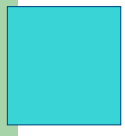
- Images:
  - color histogram, shape, edges, etc.

1D feature vector

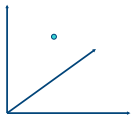
2D Matrix

Vector model

- Images:
  - color histogram, shape, edges, etc.



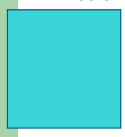
1D feature vector



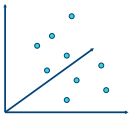
2D Matrix

Vector model

- Images:
  - color histogram, shape, edges, etc.



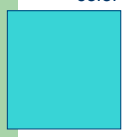
1D feature vector



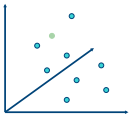
2D Matrix

Vector model

- Images:
  - color histogram, shape, edges, etc.



1D feature vector

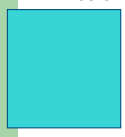


2D Matrix

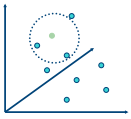
Maria Luisa Sapino (BDM 2018)

Vector model

- Images:
  - color histogram, shape, edges, etc.



1D feature vector

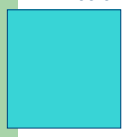


2D Matrix

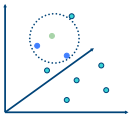
Maria Luisa Sapino (BDM 2018)

Vector model

- Images:
  - color histogram, shape, edges, etc.



1D feature vector



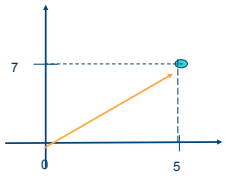
2D Matrix

Vectors…what are they???

- Image with 1 pixel <5>

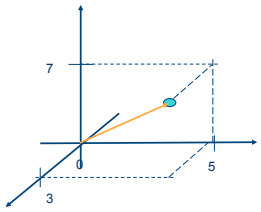
Vectors…what are they???

- Image with 2 pixels <5,7>



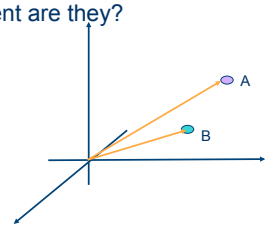
Vectors…what are they???

- Image with 3 pixels <5,7,3>



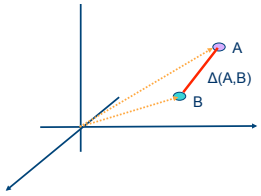
Distance between two images???

- Given A<a1,a2,a3> and B<b1,b2,b3>, how different are they?

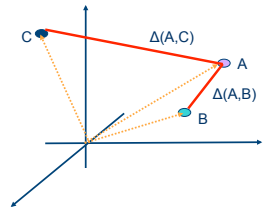


Euclidean distance

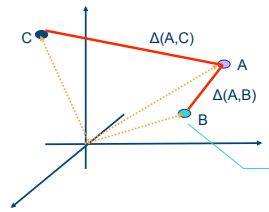
$$\Delta(A,B) = \sqrt{(a1-b1)^2 + (a2-b2)^2 + (a3-b3)^2}$$



Which image is more similar to A?

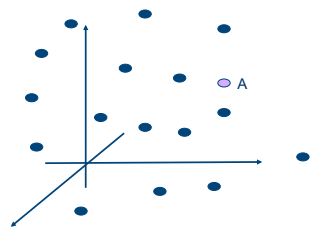


Which image is more similar to A?

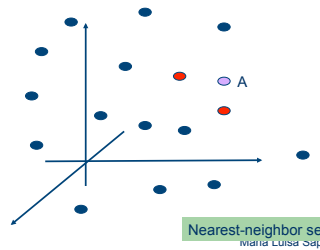


Closer to A
Similar to A

"Find 2 most similar images to A"

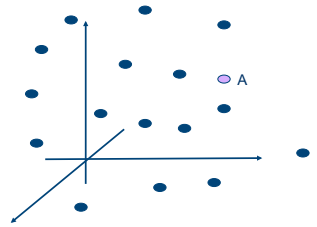


“Find 2 most similar images to A”

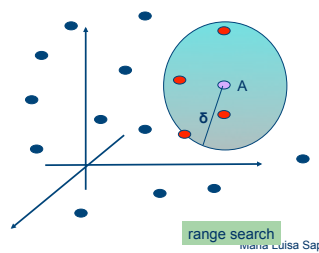


Nearest-neighbor search

“Find images at most δ different from A”

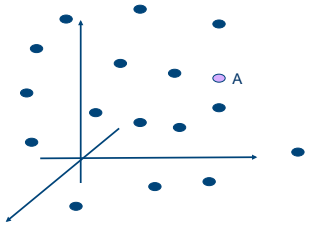


“Find images at most δ different from A”



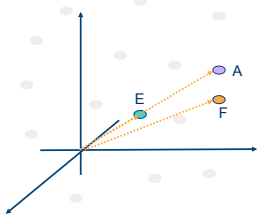
range search

Are there other similarity measures?



Let's try angles...

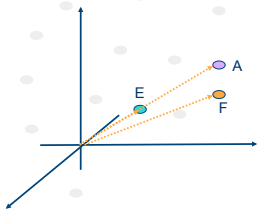
F	3	2	5
A	5	5	5
E	2	2	2



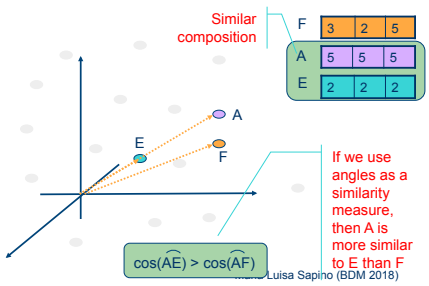
Let's try angles...

Similar composition

F	3	2	5
A	5	5	5
E	2	2	2



Let's try angles...



Angle-based measures

- Given

$$\vec{x} = \langle x_1, x_2, .., x_n \rangle \qquad \vec{y} = \langle y_1, y_2, .., y_n \rangle$$

  - Dot product

$$\vec{x}.\vec{y} = \sum_{i=1}^{n} x_i y_i$$

  - Cosine similarity

$$\cos(\vec{x}, \vec{y}) = \frac{\vec{x}.\vec{y}}{|\vec{x}||\vec{y}|}$$

What is a good measure then??

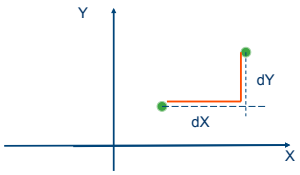
- Application dependent...
- ...but, distances in a metric space help indexing!

Metric model: axioms

- Any function *d* expressing a distance must satisfy the following axioms:
  - self-minimality: $d(s,s) = 0$
  - minimality $d(s_1,s_2) >= d(s_1,s_1)$
  - simmetry $d(s_1, s_2) = d(s_2, s_1)$
  - triangular inequality $d(s_1,s_2) + d(s_2,s_3) >= d(s_1,s_3)$
- Example: Euclidean distance

Metric distances (Minkowski metrics)

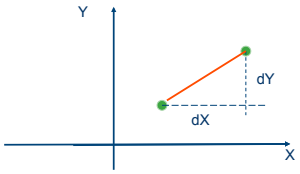
- L1-metric: d = (dX+dY)



Also called Manhattan Distance

Metric distances (Minkowski metrics)

- L2-metric: $d = (dX^2+dY^2)^{1/2}$



Also called Euclidean Distance

Metric distances (Minkowski metrics)

- L3-metric; d = $(dX^3+dY^3)^{1/3}$
- .....
- .....
- L(infinity): d = max{X,Y}

…metric model

- Well suited for certain kinds of similarity evaluation, such as color based comparisons
- Consistent with widely used approaces from computer vision and pattern recognition communities
  - results suggest that the L1 metric may better capture human notions of image similarity.
- Makes it relatively easy to index data, modeled as vectors of properties, in terms of classical multi-dimensional indexing techniques.