# Fundamentals of Computer Graphics

## Metric geometry

Emanuele Rodolà
rodola@di.uniroma1.it

SAPIENZA
UNIVERSITÀ DI ROMA

1st semester a.y. 2018/2019 · September 27, 2018

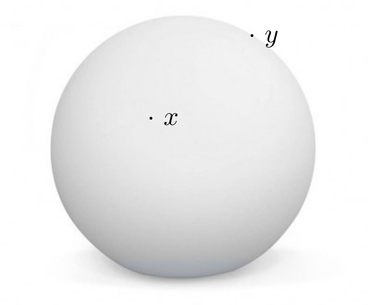
# Exercises

- Triangle mesh data structure
- Point cloud data structure

# Measuring distance

Working with curved surfaces rather than flat domains requires us to reconsider all the basic notions that we took for granted in high school geometry.

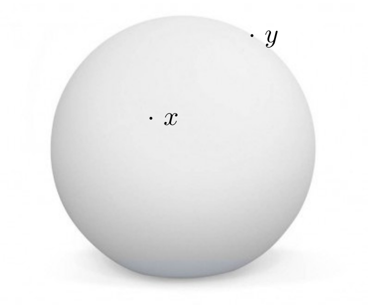
How do you measure distance between $x$ and $y$ in this picture?



# Measuring distance

Working with curved surfaces rather than flat domains requires us to reconsider all the basic notions that we took for granted in high school geometry.

How do you measure distance between $x$ and $y$ in this picture?

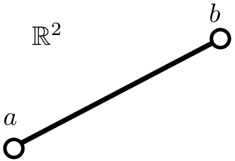



There is not a unique way!

- You can pass through the sphere with a straight line (Euclidean)
- You can walk on the surface in a "straight" path (non-Euclidean)

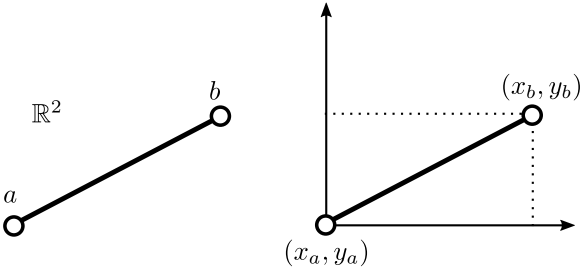
# Euclidean distance

The Euclidean distance measures the length of a straight line connecting two points:



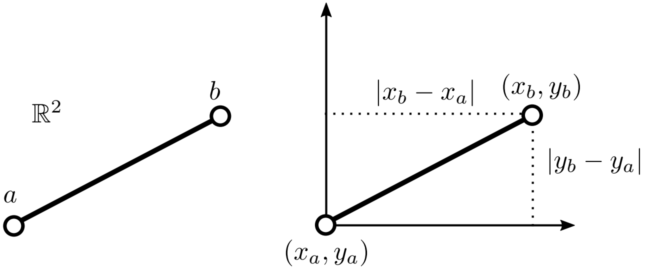
# Euclidean distance

The Euclidean distance measures the length of a straight line connecting two points:



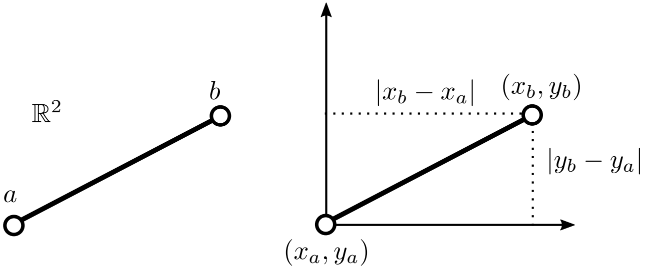
# Euclidean distance

The Euclidean distance measures the length of a straight line connecting two points:



# Euclidean distance

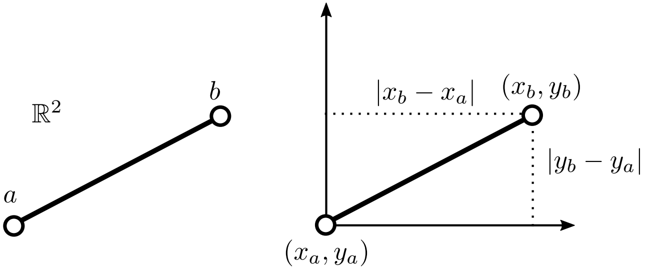
The Euclidean distance measures the length of a straight line connecting two points:




Apply Pythagoras' theorem: $d(a,b) = (|x_b - x_a|^2 + |y_b - y_a|^2)^{\frac{1}{2}}$

# Euclidean distance

The Euclidean distance measures the length of a straight line connecting two points:




Apply Pythagoras' theorem: $d(a,b) = (|x_b - x_a|^2 + |y_b - y_a|^2)^{\frac{1}{2}}$

In vector notation:

$$d(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2$$

where $\mathbf{a} = \begin{pmatrix} x_a \\ y_a \end{pmatrix}$ and $\mathbf{b} = \begin{pmatrix} x_b \\ y_b \end{pmatrix}$

# $L_p$ distance in $\mathbb{R}^k$

One can generalize to different power coefficients $p \geq 1$:

$$\|\mathbf{x} - \mathbf{y}\|_2 = (|x_1 - y_1|^2 + |x_2 - y_2|^2)^{\frac{1}{2}}$$
$$\Downarrow$$
$$\|\mathbf{x} - \mathbf{y}\|_p = (|x_1 - y_1|^p + |x_2 - y_2|^p)^{\frac{1}{p}}$$

# $L_p$ distance in $\mathbb{R}^k$

One can generalize to different power coefficients $p \geq 1$:

$$\|\mathbf{x} - \mathbf{y}\|_2 = \left(|x_1 - y_1|^2 + |x_2 - y_2|^2\right)^{\frac{1}{2}}$$

$$\Downarrow$$

$$\|\mathbf{x} - \mathbf{y}\|_p = \left(|x_1 - y_1|^p + |x_2 - y_2|^p\right)^{\frac{1}{p}}$$

As well as generalize from $\mathbb{R}^2$ to $\mathbb{R}^k$:

$$\|\mathbf{x} - \mathbf{y}\|_p = \left(\sum_{i=1}^{k} |x_i - y_i|^p\right)^{\frac{1}{p}}$$

# $L_p$ distance in $\mathbb{R}^k$

One can generalize to different power coefficients $p \geq 1$:

$$\|\mathbf{x} - \mathbf{y}\|_2 = \left(|x_1 - y_1|^2 + |x_2 - y_2|^2\right)^{\frac{1}{2}}$$

$$\Downarrow$$

$$\|\mathbf{x} - \mathbf{y}\|_p = \left(|x_1 - y_1|^p + |x_2 - y_2|^p\right)^{\frac{1}{p}}$$

As well as generalize from $\mathbb{R}^2$ to $\mathbb{R}^k$:

$$\|\mathbf{x} - \mathbf{y}\|_p = \left(\sum_{i=1}^{k} |x_i - y_i|^p\right)^{\frac{1}{p}}$$

This definition gives us the $L_p$ distance between vectors in $\mathbb{R}^k$

Examples:

- Euclidean ($L_2$) distance between 3D points
- Manhattan ($L_1$) distance between cities in a map

# Exercise: $L_2$ distance in $\mathbb{R}^3$

Let us be given a triangle mesh $M$ and the associated point cloud $P$ obtained by removing the mesh connectivity from $M$.

# Exercise: $L_2$ distance in $\mathbb{R}^3$

Let us be given a triangle mesh $M$ and the associated point cloud $P$ obtained by removing the mesh connectivity from $M$.

- For a given vertex $v$ of $M$, compute $d_{L_2}(v, x)$, i.e., the $L_2$ distance from $v$ to all other vertices $x$ in $M$

# Exercise: $L_2$ distance in $\mathbb{R}^3$

Let us be given a triangle mesh $M$ and the associated point cloud $P$ obtained by removing the mesh connectivity from $M$.

- For a given vertex $v$ of $M$, compute $d_{L_2}(v, x)$, i.e., the $L_2$ distance from $v$ to all other vertices $x$ in $M$
- Make sure you understand that, for a fixed $v$, the distance $d_{L_2}(v, x)$ can be seen as a scalar function $f_v : M \to \mathbb{R}$ such that $f_v(x) = d_{L_2}(v, x)$

# Exercise: $L_2$ distance in $\mathbb{R}^3$

Let us be given a triangle mesh $M$ and the associated point cloud $P$ obtained by removing the mesh connectivity from $M$.

- For a given vertex $v$ of $M$, compute $d_{L_2}(v, x)$, i.e., the $L_2$ distance from $v$ to all other vertices $x$ in $M$
- Make sure you understand that, for a fixed $v$, the distance $d_{L_2}(v, x)$ can be seen as a scalar function $f_v : M \to \mathbb{R}$ such that $f_v(x) = d_{L_2}(v, x)$
- Visualize $f_v$ by coloring the mesh, by assigning a different vertex color to each value of $f_v$; in Matlab, you can use the trisurf () function

# Exercise: $L_2$ distance in $\mathbb{R}^3$

Let us be given a triangle mesh $M$ and the associated point cloud $P$ obtained by removing the mesh connectivity from $M$.

- For a given vertex $v$ of $M$, compute $d_{L_2}(v, x)$, i.e., the $L_2$ distance from $v$ to all other vertices $x$ in $M$
- Make sure you understand that, for a fixed $v$, the distance $d_{L_2}(v, x)$ can be seen as a scalar function $f_v : M \to \mathbb{R}$ such that $f_v(x) = d_{L_2}(v, x)$
- Visualize $f_v$ by coloring the mesh, by assigning a different vertex color to each value of $f_v$; in Matlab, you can use the trisurf () function
- Compute and visualize the average $L_2$ distance function, defined as:

$$\text{avg}(v) = \sum_{x \in M} f_v(x)$$

# Exercise: $L_2$ distance in $\mathbb{R}^3$

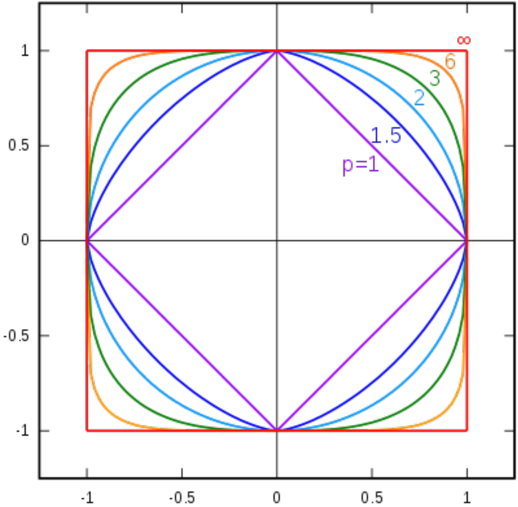
Let us be given a triangle mesh $M$ and the associated point cloud $P$ obtained by removing the mesh connectivity from $M$.

- For a given vertex $v$ of $M$, compute $d_{L_2}(v, x)$, i.e., the $L_2$ distance from $v$ to all other vertices $x$ in $M$
- Make sure you understand that, for a fixed $v$, the distance $d_{L_2}(v, x)$ can be seen as a scalar function $f_v : M \to \mathbb{R}$ such that $f_v(x) = d_{L_2}(v, x)$
- Visualize $f_v$ by coloring the mesh, by assigning a different vertex color to each value of $f_v$; in Matlab, you can use the trisurf () function
- Compute and visualize the average $L_2$ distance function, defined as:

$$\mathrm{avg}(v) = \sum_{x \in M} f_v(x)$$

- Do all the above for points on the point cloud $P$

# $L_p$ unit balls



# Metric spaces

The pair (object, distance) forms a metric space.

# Metric spaces

The pair (object, distance) forms a metric space. More formally:

A set $\mathcal{M}$ is a metric space if for every pair of points $x, y \in \mathcal{M}$ there is a metric (or distance) function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_+$ such that:

# Metric spaces

The pair (object, distance) forms a metric space. More formally:

A set $\mathcal{M}$ is a metric space if for every pair of points $x, y \in \mathcal{M}$ there is a metric (or distance) function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}_+$ such that:

- $d_{\mathcal{M}}(x, y) = 0 \Leftrightarrow x = y$ (identity of indiscernibles)

# Metric spaces

The pair (object, distance) forms a metric space. More formally:

A set $\mathcal{M}$ is a metric space if for every pair of points $x, y \in \mathcal{M}$ there is a metric (or distance) function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_+$ such that:

- $d_{\mathcal{M}}(x, y) = 0 \Leftrightarrow x = y$ (identity of indiscernibles)
- $d_{\mathcal{M}}(x, y) = d_{\mathcal{M}}(y, x)$ (symmetry)

# Metric spaces

The pair (object, distance) forms a metric space. More formally:

A set $\mathcal{M}$ is a metric space if for every pair of points $x, y \in \mathcal{M}$ there is a metric (or distance) function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_+$ such that:

- $d_{\mathcal{M}}(x, y) = 0 \Leftrightarrow x = y$ (identity of indiscernibles)
- $d_{\mathcal{M}}(x, y) = d_{\mathcal{M}}(y, x)$ (symmetry)
- $d_{\mathcal{M}}(x, y) \leq d_{\mathcal{M}}(y, z) + d_{\mathcal{M}}(z, x)$ for any $x, y, z \in \mathcal{M}$ (triangle inequality)

# Metric spaces

The pair (object, distance) forms a metric space. More formally:

A set $\mathcal{M}$ is a metric space if for every pair of points $x, y \in \mathcal{M}$ there is a metric (or distance) function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_+$ such that:

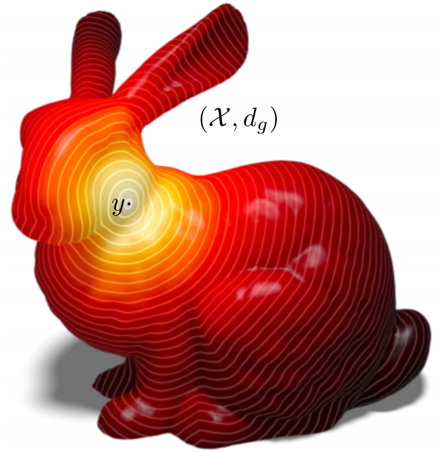
- $d_{\mathcal{M}}(x, y) = 0 \Leftrightarrow x = y$ (identity of indiscernibles)
- $d_{\mathcal{M}}(x, y) = d_{\mathcal{M}}(y, x)$ (symmetry)
- $d_{\mathcal{M}}(x, y) \leq d_{\mathcal{M}}(y, z) + d_{\mathcal{M}}(z, x)$ for any $x, y, z \in \mathcal{M}$ (triangle inequality)

We will specify a metric space as the pair $(\mathcal{M}, d_{\mathcal{M}})$

Example:

- The sphere with Euclidean distance is $(\mathbb{S}^2, d_{L_2})$
- The sphere with geodesic distance is $(\mathbb{S}^2, d_g)$

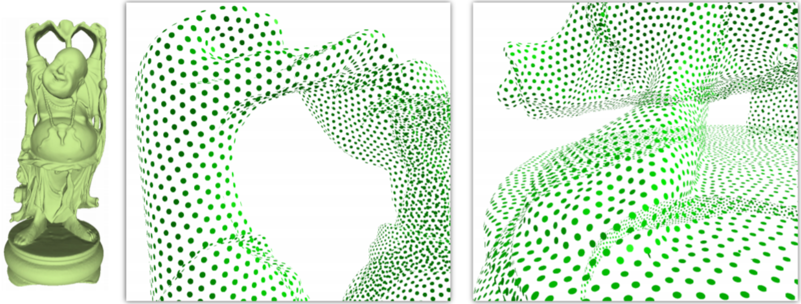
# Example: Geodesic isolines




Each isoline identifies a set of points $x \in \mathcal{X}$ at the same distance (according to $d_g$) from some reference $y \in \mathcal{X}$

# Exercise: Farthest point sampling

Implement a farthest point sampling (FPS) scheme using this algorithm:



# Exercise: Farthest point sampling

Implement a farthest point sampling (FPS) scheme using this algorithm:

- Fix $n$ and let $\mathcal{S}^{(0)} = \{y\}$ for some $y \in \mathcal{X}$
- Proceed recursively:
  - At step $k$, given $\mathcal{S}^{(k-1)}$, select $x \in (\mathcal{X}, d_{\mathcal{X}})$ such that

$$x = \arg\max_{x \in \mathcal{X}} d_{\mathcal{X}}(x, \mathcal{S}^{(k-1)})$$

  - Set $\mathcal{S}^{(k)} = \mathcal{S}^{(k-1)} \cup x$
  - Repeat until $k = n$
- Test with different starting points $y$
- Test with a fixed starting point and gradually increasing $n$

Use the Euclidean distance for the definition of $d_{\mathcal{X}}$.

# Exercise: Voronoi decomposition

For a given sampling $\mathcal{S}$, the associated Voronoi regions are defined as:

$$V_i(\mathcal{S}) = \{x \in \mathcal{X} \ : \ d_{\mathcal{X}}(x, x_i) < d_{\mathcal{X}}(x, x_j), x_{j \neq i} \in \mathcal{S}\}$$

- How do these regions look like?
- Implement Voronoi decomposition for meshes and point clouds using the Euclidean metric and using farthest point sampling for $\mathcal{S}$
- Visualize the Voronoi regions by assigning to each of them a random color

In order to color point clouds in Matlab, you can use the scatter3 () function

# Examples: Metric spaces

- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = |x - y|$

# Examples: Metric spaces

- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = |x - y|$
- $\mathcal{X} = \mathcal{A} \subset \mathbb{R}^k\,, \quad d_{\mathcal{X}}(x, y) = \|x - y\|_2$

# Examples: Metric spaces

- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = |x - y|$
- $\mathcal{X} = \mathcal{A} \subset \mathbb{R}^k\,, \quad d_{\mathcal{X}}(x, y) = \|x - y\|_2$
- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = \log(|x - y| + 1)$

# Examples: Metric spaces

- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = |x - y|$
- $\mathcal{X} = \mathcal{A} \subset \mathbb{R}^k\,, \quad d_{\mathcal{X}}(x, y) = \|x - y\|_2$
- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = \log(|x - y| + 1)$
- $\mathcal{X} = \text{any set}\,, \quad d_{\mathcal{X}}(x, y) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$

# Examples: Metric spaces

- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = |x - y|$
- $\mathcal{X} = \mathcal{A} \subset \mathbb{R}^k\,, \quad d_{\mathcal{X}}(x, y) = \|x - y\|_2$
- $\mathcal{X} = \mathbb{R}\,, \quad d_{\mathcal{X}}(x, y) = \log(|x - y| + 1)$
- $\mathcal{X} = \text{any set}\,, \quad d_{\mathcal{X}}(x, y) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$
- $\mathcal{X} = \mathcal{A} \times \mathcal{B}\,, \quad d_{\mathcal{X}}((a_1, b_1), (a_2, b_2)) = \sqrt{d_{\mathcal{A}}(a_1, a_2)^2 + d_{\mathcal{B}}(b_1, b_2)^2}$

# Ambient space and restriction

If $\mathcal{A}$ is a metric space and $\mathcal{X} \subset \mathcal{A}$, then $\mathcal{A}$ is called ambient space for $\mathcal{X}$.


$\mathcal{A} = (\mathbb{R}^3, d_{L_2})$

# Ambient space and restriction

If $\mathcal{A}$ is a metric space and $\mathcal{X} \subset \mathcal{A}$, then $\mathcal{A}$ is called ambient space for $\mathcal{X}$.


$$\mathcal{A} = (\mathbb{R}^3, d_{L_2})$$

A metric on $\mathcal{X}$ can be obtained by the restriction $d_{\mathcal{X}} = d_{\mathcal{A}|\mathcal{X}}$, such that:

$$d_{\mathcal{X}}(x, y) = d_{\mathcal{A}}(x, y)$$

for all $x, y \in \mathcal{X}$

# Isometries

Let $(\mathcal{M}, d_{\mathcal{M}})$ and $(\mathcal{N}, d_{\mathcal{N}})$ be two metric spaces.

A bijective map $f : \mathcal{M} \to \mathcal{N}$ is called an isometry if:

$$d_{\mathcal{M}}(x, y) = d_{\mathcal{N}}(f(x), f(y))$$

for any $x, y \in \mathcal{M}$.

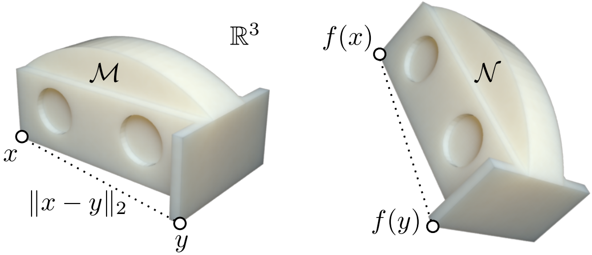
# Isometries

Let $(\mathcal{M}, d_{\mathcal{M}})$ and $(\mathcal{N}, d_{\mathcal{N}})$ be two metric spaces.

A bijective map $f : \mathcal{M} \to \mathcal{N}$ is called an isometry if:

$$d_{\mathcal{M}}(x, y) = d_{\mathcal{N}}(f(x), f(y))$$

for any $x, y \in \mathcal{M}$.



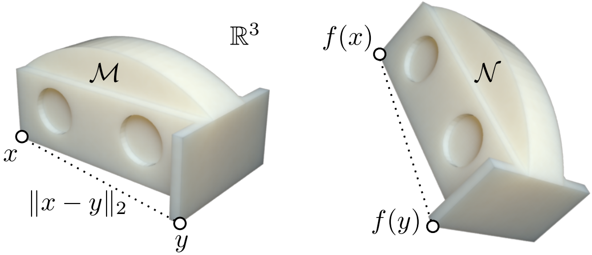
# Isometries

Let $(\mathcal{M}, d_{\mathcal{M}})$ and $(\mathcal{N}, d_{\mathcal{N}})$ be two metric spaces.

A bijective map $f : \mathcal{M} \to \mathcal{N}$ is called an isometry if:

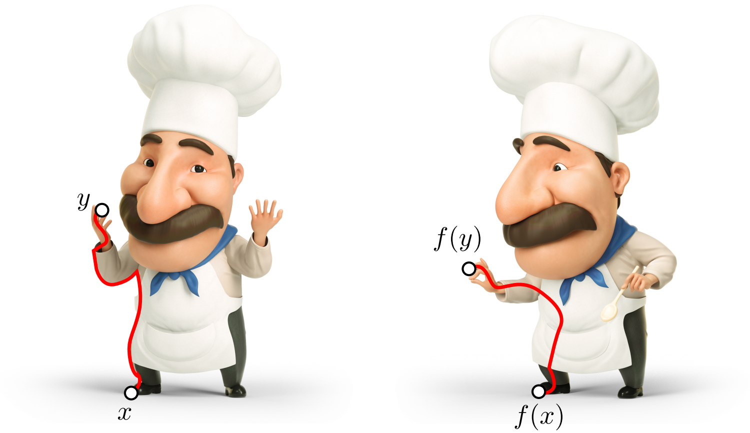
$$d_{\mathcal{M}}(x, y) = d_{\mathcal{N}}(f(x), f(y))$$

for any $x, y \in \mathcal{M}$.

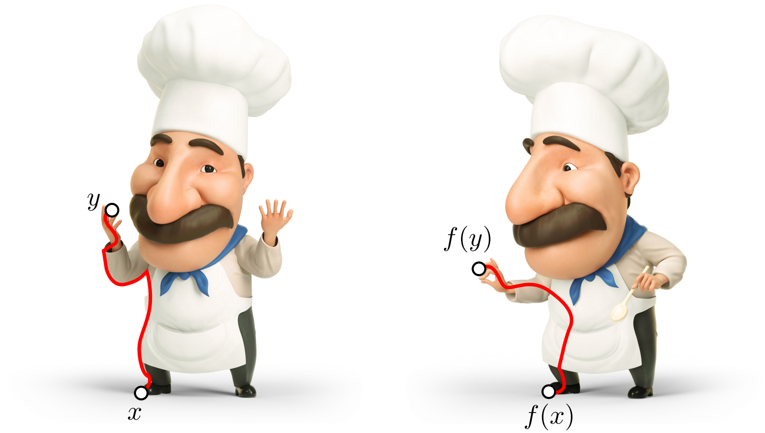



If $d_{\mathcal{M}} = \| \cdot \|_2$ and $d_{\mathcal{N}} = \| \cdot \|_2$ we say "rigid isometry"

# Example: Non-rigid “quasi”-isometries



# Example: Non-rigid “quasi”-isometries




$$d_{\mathcal{M}}(x, y) \approx d_{\mathcal{N}}(f(x), f(y))$$

(here $d_{\mathcal{M}}, d_{\mathcal{N}}$ are geodesic distance functions)

# Isometry as equivalence

"Being isometric" is an equivalence relation, since it is:

- reflective ($a = a$)
- symmetric ($a = b \Rightarrow b = a$)
- transitive ($a = b \wedge b = c \Rightarrow a = c$)

# Isometry as equivalence

"Being isometric" is an equivalence relation, since it is:

- reflective ($a = a$)
- symmetric ($a = b \Rightarrow b = a$)
- transitive ($a = b \land b = c \Rightarrow a = c$)

In this sense, we think of isometric shapes as being the same shape:

$=$

$=$

# Distance

We now have a notion of equivalence between shapes. Can we also establish a notion of distance between shapes?

There are many!

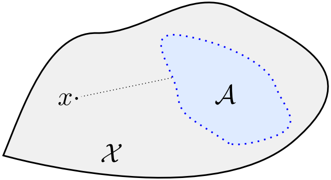
# Distance

We now have a notion of equivalence between shapes. Can we also establish a notion of distance between shapes?

There are many!

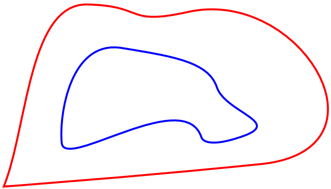
We start by defining the distance from a point $x$ to a set $\mathcal{A} \subseteq (\mathcal{X}, d_{\mathcal{X}})$:

$$\mathrm{dist}_{\mathcal{X}}(x, \mathcal{A}) = \min_{y \in \mathcal{A}} d_{\mathcal{X}}(x, y)$$



# Hausdorff distance

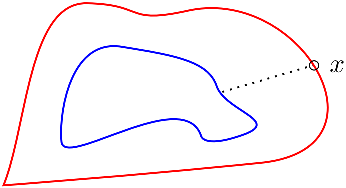
Consider two subsets $\mathcal{X}, \mathcal{Y} \subset (\mathcal{Z}, d_{\mathcal{Z}})$.



# Hausdorff distance

Consider two subsets $\mathcal{X}, \mathcal{Y} \subset (\mathcal{Z}, d_\mathcal{Z})$.

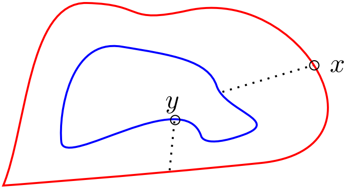
$$\max_{x \in \mathcal{X}} \mathrm{dist}_\mathcal{Z}(x, \mathcal{Y})$$



# Hausdorff distance

Consider two subsets $\mathcal{X}, \mathcal{Y} \subset (\mathcal{Z}, d_{\mathcal{Z}})$.

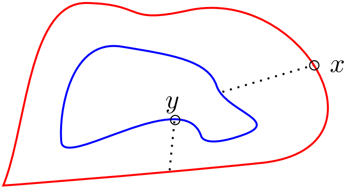
$$\max_{x \in \mathcal{X}} \operatorname{dist}_{\mathcal{Z}}(x, \mathcal{Y}), \max_{y \in \mathcal{Y}} \operatorname{dist}_{\mathcal{Z}}(y, \mathcal{X})$$



# Hausdorff distance

Consider two subsets $\mathcal{X}, \mathcal{Y} \subset (\mathcal{Z}, d_{\mathcal{Z}})$.

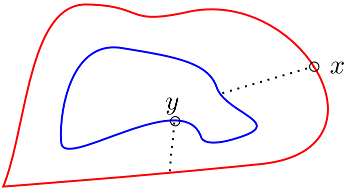
$$\max \left\{ \max_{x \in \mathcal{X}} \operatorname{dist}_{\mathcal{Z}}(x, \mathcal{Y}), \max_{y \in \mathcal{Y}} \operatorname{dist}_{\mathcal{Z}}(y, \mathcal{X}) \right\}$$



# Hausdorff distance

Consider two subsets $\mathcal{X}, \mathcal{Y} \subset (\mathcal{Z}, d_{\mathcal{Z}})$.

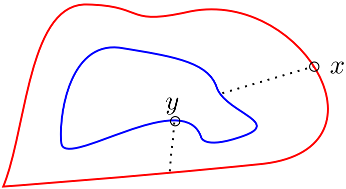
$$d_{\mathcal{H}}^{\mathcal{Z}}(\mathcal{X},\mathcal{Y}) = \max\left\{ \max_{x\in\mathcal{X}} \mathrm{dist}_{\mathcal{Z}}(x,\mathcal{Y}), \max_{y\in\mathcal{Y}} \mathrm{dist}_{\mathcal{Z}}(y,\mathcal{X}) \right\}$$



# Hausdorff distance

Consider two subsets $\mathcal{X}, \mathcal{Y} \subset (\mathcal{Z}, d_{\mathcal{Z}})$.

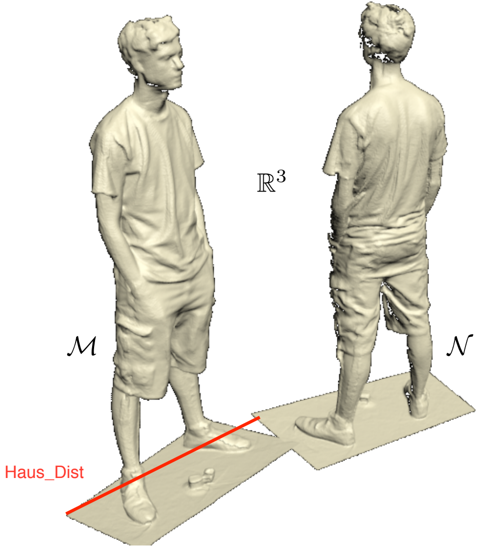
$$d_{\mathcal{H}}^{\mathcal{Z}}(\mathcal{X}, \mathcal{Y}) = \max \left\{ \max_{x \in \mathcal{X}} \operatorname{dist}_{\mathcal{Z}}(x, \mathcal{Y}), \max_{y \in \mathcal{Y}} \operatorname{dist}_{\mathcal{Z}}(y, \mathcal{X}) \right\}$$




The Hausdorff distance is defined between subsets of a metric space

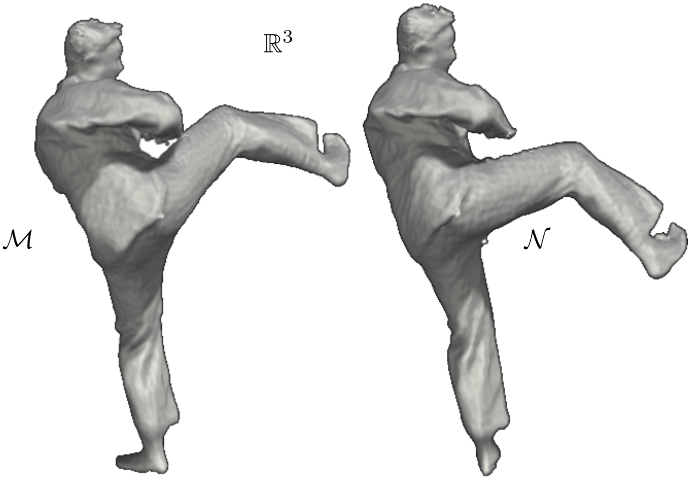
Note that perturbing one single point can make $d_{\mathcal{H}}^{\mathcal{Z}}$ arbitrarily large

# Example: Hausdorff distance, rigid case



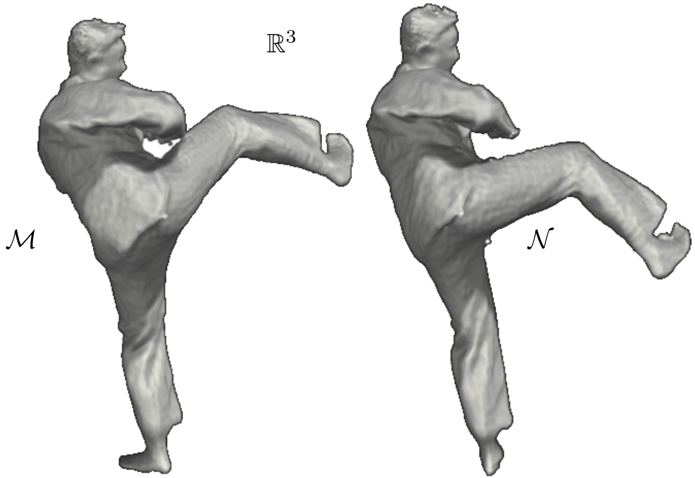

What can we do to minimize $d_{\mathcal{H}}^{\mathbb{R}^3}(\mathcal{M}, \mathcal{N})$? overlap them

# Example: Hausdorff distance, non-rigid case




What can we do to minimize $d_{\mathcal{H}}^{\mathbb{R}^3}(\mathcal{M}, \mathcal{N})$?

# Example: Hausdorff distance, non-rigid case




What can we do to minimize $d_{\mathcal{H}}^{\mathbb{R}^3}(\mathcal{M}, \mathcal{N})$?
The Hausdorff distance is better suited to compare rigid shapes

The Hausdorff distance can be used to compute the difference between meshes representing the same underlying surface (e.g. compare level-of-detail)

# Isometric embeddings

Hausdorff distance is between subsets of a common ambient space.

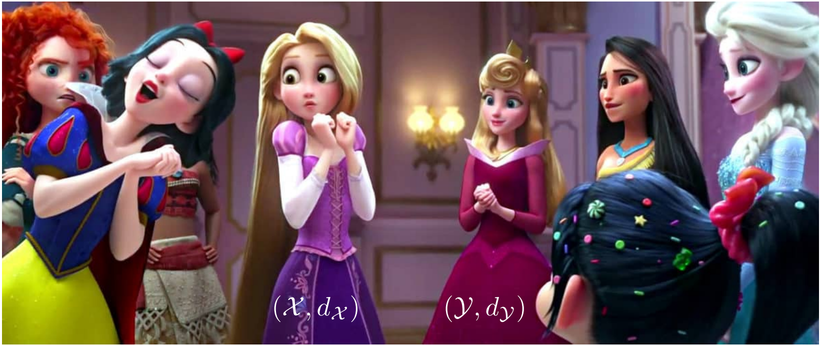
Can we extend it to general metric spaces, each with its own metric?

# Isometric embeddings

Hausdorff distance is between subsets of a common ambient space.

Can we extend it to general metric spaces, each with its own metric?

For example, we want to compare Disney princesses:



# Isometric embeddings

Hausdorff distance is between subsets of a common ambient space.

Can we extend it to general metric spaces, each with its own metric?

General idea: Embed $(\mathcal{X}, d_{\mathcal{X}})$ and $(\mathcal{Y}, d_{\mathcal{Y}})$ into a new metric space $(\mathcal{Z}, d_{\mathcal{Z}})$, and compute the classical Hausdorff distance there.

# Isometric embeddings

Hausdorff distance is between subsets of a common ambient space.

Can we extend it to general metric spaces, each with its own metric?

General idea: Embed $(\mathcal{X}, d_{\mathcal{X}})$ and $(\mathcal{Y}, d_{\mathcal{Y}})$ into a new metric space $(\mathcal{Z}, d_{\mathcal{Z}})$, and compute the classical Hausdorff distance there.

An isometric embedding is a transformation $f : \mathcal{X} \to \mathcal{Z}$ which preserves the metric for all pairs $x, y \in \mathcal{X}$, i.e.

$$d_{\mathcal{Z}}(f(x), f(y)) = d_{\mathcal{X}}(x, y)$$

# Isometric embeddings

Hausdorff distance is between subsets of a common ambient space.

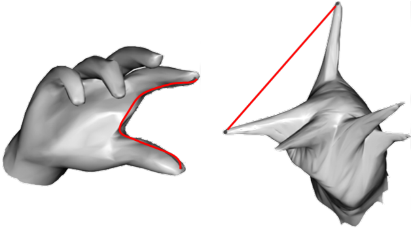
Can we extend it to general metric spaces, each with its own metric?

General idea: Embed $(\mathcal{X}, d_{\mathcal{X}})$ and $(\mathcal{Y}, d_{\mathcal{Y}})$ into a new metric space $(\mathcal{Z}, d_{\mathcal{Z}})$, and compute the classical Hausdorff distance there.

An isometric embedding is a transformation $f : \mathcal{X} \to \mathcal{Z}$ which preserves the metric for all pairs $x, y \in \mathcal{X}$, i.e.

$$d_{\mathcal{Z}}(f(x), f(y)) = d_{\mathcal{X}}(x, y)$$

For example, take $d_{\mathcal{X}} = d_g$ and $d_{\mathcal{Z}} = \| \cdot \|_2$:

$(\mathcal{X}, d_g)$  $(\mathbb{R}^3, \| \cdot \|_2)$

# Gromov-Hausdorff distance

The following questions arise:

- In which new metric space $(\mathcal{Z}, d_{\mathcal{Z}})$ should we embed?
- Using which isometric embedding?

# Gromov-Hausdorff distance

The following questions arise:

- In which new metric space $(\mathcal{Z}, d_{\mathcal{Z}})$ should we embed?
- Using which isometric embedding?

Choose the ones resulting in minimum Hausdorff distance:

$$d_{\mathcal{GH}}(\mathcal{X}, \mathcal{Y}) = \min_{\mathcal{Z}, f, g} d_{\mathcal{H}}^{\mathcal{Z}}(f(\mathcal{X}), g(\mathcal{Y}))$$

where $f : \mathcal{X} \to \mathcal{Z}$ and $g : \mathcal{Y} \to \mathcal{Z}$ are isometric embeddings

The Gromov-Hausdorff distance is a metric on the space of isometry classes of metric spaces

# Gromov-Hausdorff distance

"The Gromov-Hausdorff distance is a metric on the space of isometry classes of metric spaces"

An isometry class is a set of shapes which are equal up to isometry. Therefore:

$d_{\mathcal{GH}}($  $,$  $) = d_{\mathcal{GH}}($  $,$  $)$

# Gromov-Hausdorff distance

"The Gromov-Hausdorff distance is a metric on the space of isometry classes of metric spaces"

An isometry class is a set of shapes which are equal up to isometry.
Therefore:

$d_{\mathcal{GH}}($  $,$  $) = d_{\mathcal{GH}}($  $,$  $)$

Question: What is the isometry class for the sphere $(\mathbb{S}^2, d_g)$?

# A cartographer's problem

Computing Gromov-Haudorff distances entails computing embeddings. Is this always possible?

Consider the following:

# A cartographer's problem

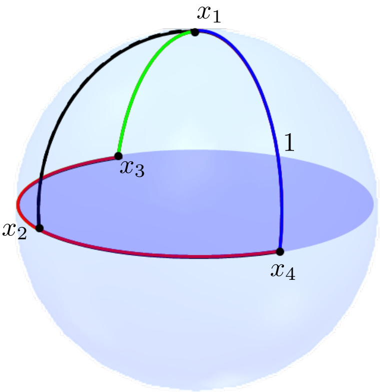
Computing Gromov-Haudorff distances entails computing embeddings. Is this always possible?

Consider the following:

An isometric embedding of $\mathbb{S}^2$ into $\mathbb{R}^2$ is not possible!

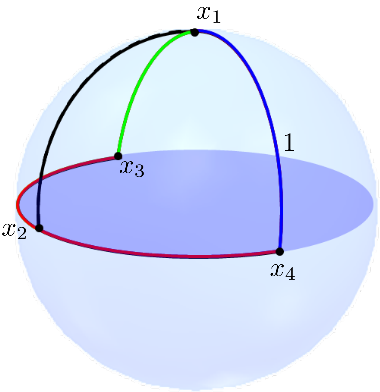
Any approximate solution introduces metric distortion

# Non-embeddability of the sphere



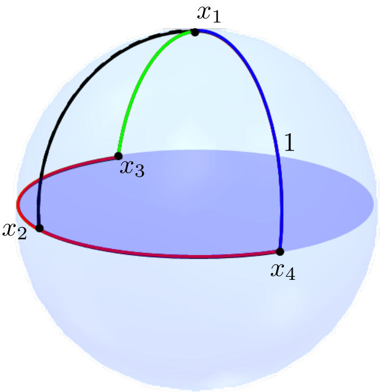

- Consider the triangle $\Delta(x_1, x_3, x_4)$

# Non-embeddability of the sphere



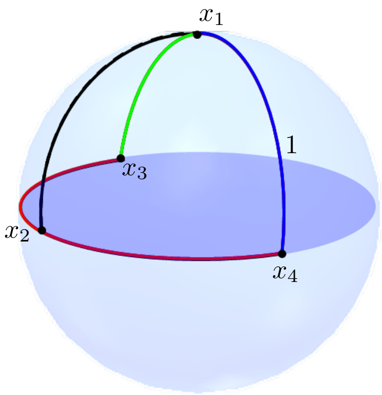

- Consider the triangle $\Delta(x_1, x_3, x_4) \Rightarrow$ collinear!

# Non-embeddability of the sphere



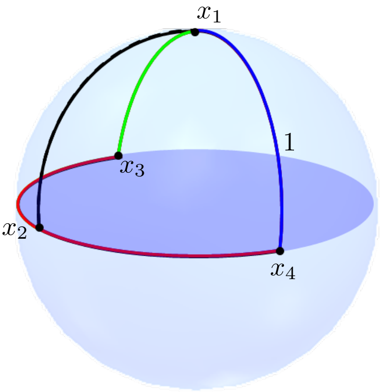

- Consider the triangle $\Delta(x_1, x_3, x_4) \Rightarrow$ collinear!
- Consider the triangle $\Delta(x_2, x_3, x_4)$

# Non-embeddability of the sphere




- Consider the triangle $\Delta(x_1, x_3, x_4) \Rightarrow$ collinear!
- Consider the triangle $\Delta(x_2, x_3, x_4) \Rightarrow$ collinear!

# Non-embeddability of the sphere




- Consider the triangle $\Delta(x_1, x_3, x_4) \Rightarrow$ collinear!
- Consider the triangle $\Delta(x_2, x_3, x_4) \Rightarrow$ collinear!
- Then $x_1 = x_2$, which contradicts $d_g(x_1, x_2) = 1$
  $\Rightarrow$ This metric space cannot be embedded into $\mathbb{R}^k$ for any $k$

# A cartographer’s solution


# Teaser exercise: Matrix calculus

Let matrix $\mathbf{X} \in \mathbb{R}^{n \times 3}$ contain the 3D coordinates of points $x_i$ as its rows.

Consider the following expression:

$$n \sum_{i=1}^{n} \langle x_i, x_i \rangle - \sum_{i,j} \langle x_i, x_j \rangle$$

How do you write the expression above in matrix notation?

Tip: Use the trace operation, defined as $\operatorname{tr}(\mathbf{X}) = \sum_{i=1}^{n} x_{ii}$