CS 247 – Scientific Visualization
Lecture 5: Data Representation, Pt. 2

Markus Hadwiger, KAUST

Reading Assignment #3 (until Feb 12)

Read (required):

- Data Visualization book, finish Chapter 3 (read starting with 3.6)
- Data Visualization book, Chapter 5 until 5.3 (inclusive)

Data Representation

Data Space (Domain) vs. Data Type (Codomain)

	1D	2D	3D
1D	$y=f(x)$		spatial curve $\mathbf{x}(t)$
2D		2D flow $\mathbf{v}(\mathbf{x})$	
3D	CT data $s(\mathbf{x})$		3D flow $\mathbf{v}(\mathbf{x})$

examples

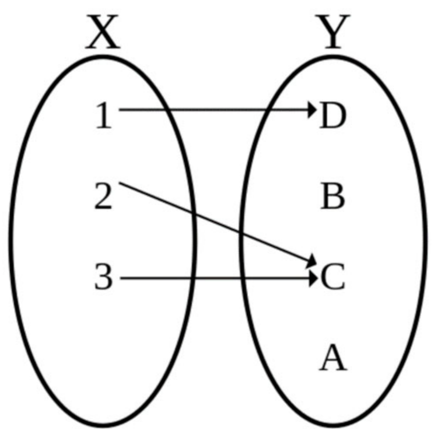
Data == Functions

Mathematical Functions

Associates every element of a set (e.g., X) with *exactly one* element of another set (e.g., Y)

Maps from *domain* (X) to *codomain* (Y)

$$\begin{aligned} f : X &\to Y \\ x &\mapsto f(x) \end{aligned}$$



Also important: *range/image*; *preimage*; continuity, differentiability, dimensionality, ...

Graph of a function (mathematical definition):

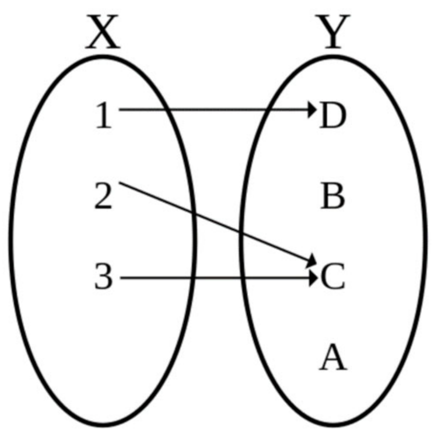
$$G(f) := \{(x, f(x)) | x \in X\} \subset X \times Y$$

Mathematical Functions

Associates every element of a set (e.g., X) with *exactly one* element of another set (e.g., Y)

Maps from *domain* (X) to *codomain* (Y)

$$
\begin{aligned}
f \colon \mathbb{R}^n &\to \mathbb{R}^m \\
x &\mapsto f(x)
\end{aligned}
$$



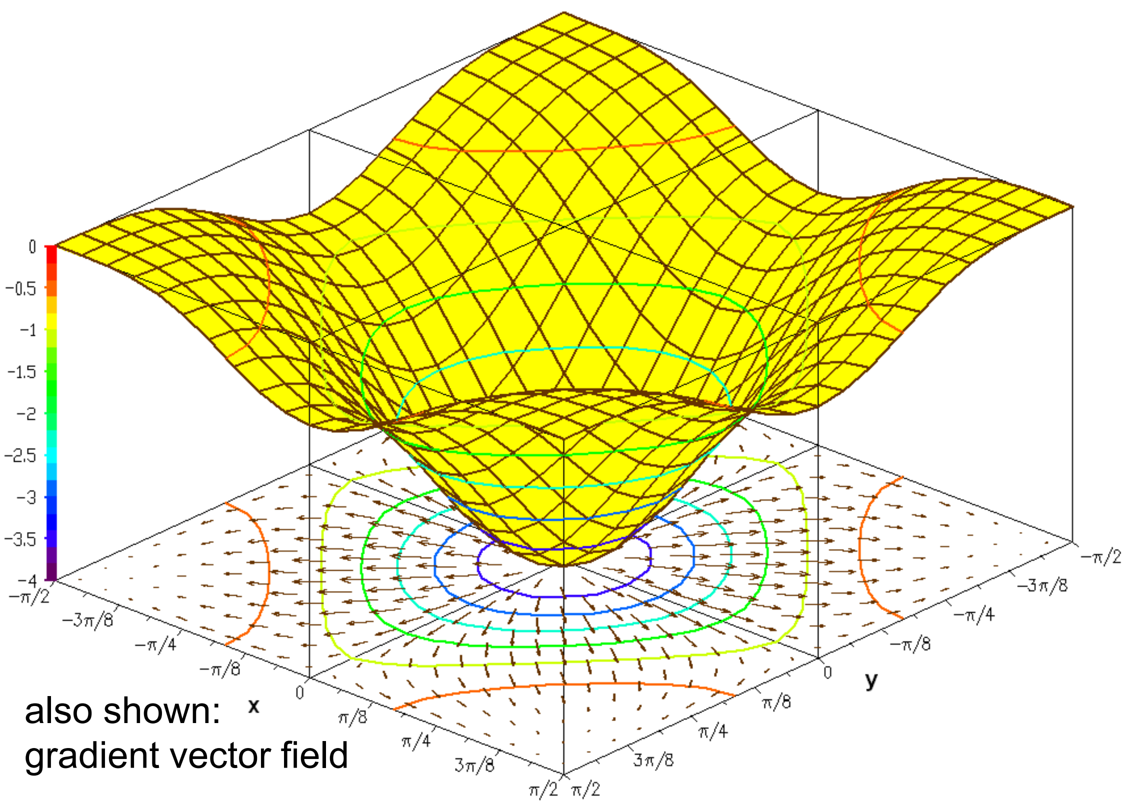
Also important: *range/image*; *preimage*; continuity, differentiability, dimensionality, ...

Graph of a function (mathematical definition):

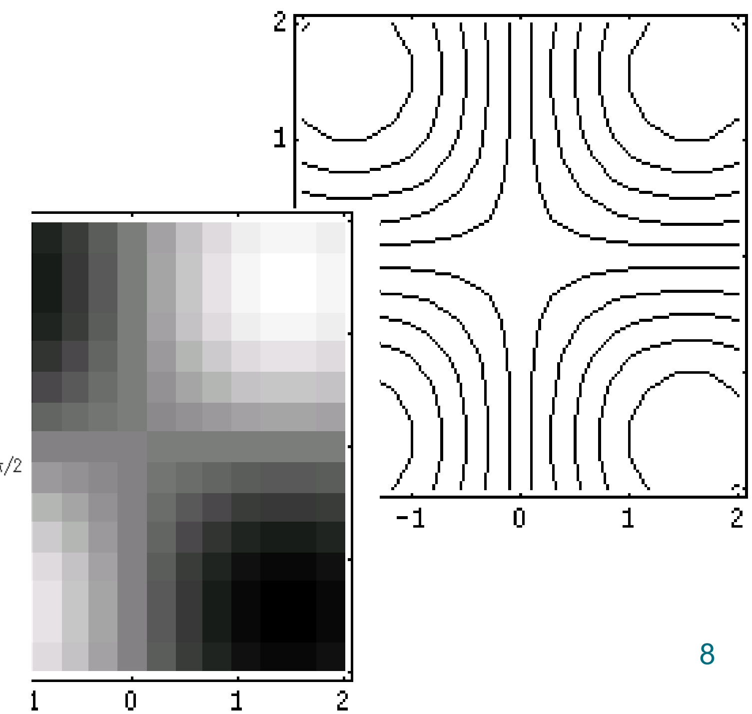
$$G(f) := \{(x, f(x)) | x \in \mathbb{R}^n\} \subset \mathbb{R}^n \times \mathbb{R}^m \simeq \mathbb{R}^{n+m}$$

Visualization Examples

data	description	visualization example
$R^2 \rightarrow R^1$	function over R^2	2D-height map in 3D, contour lines in 2D, false colors (heat map)

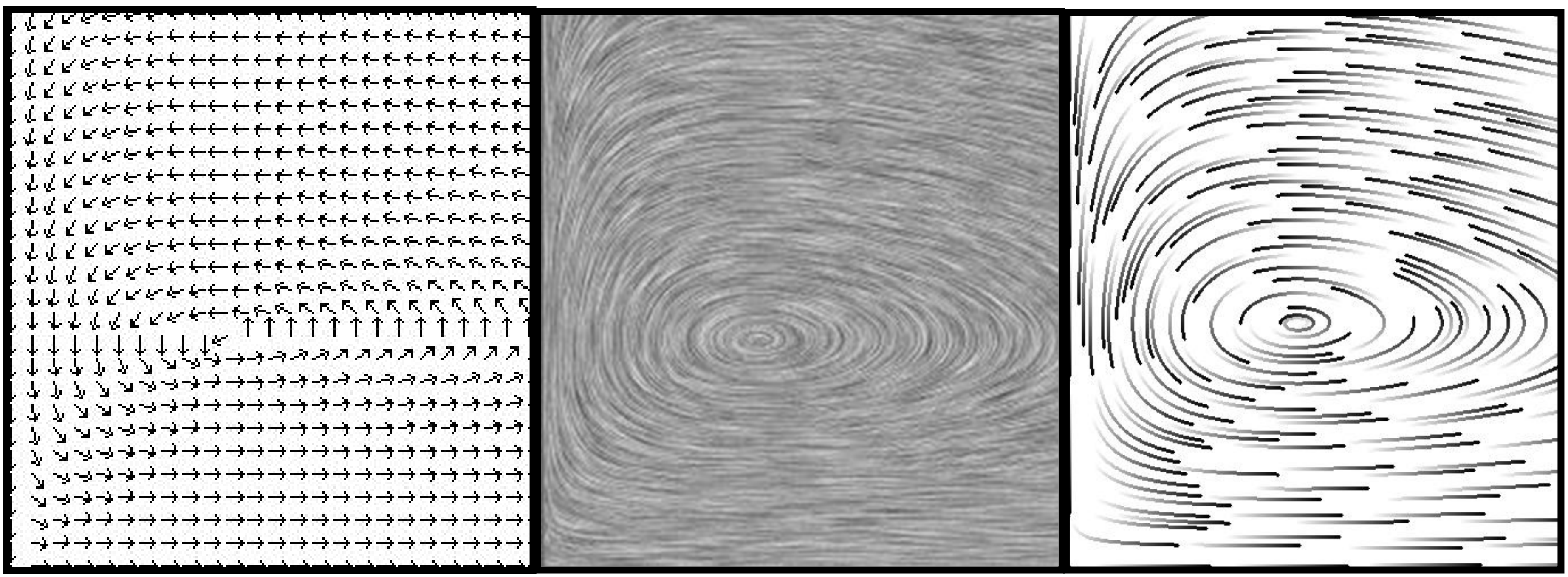


also shown:
gradient vector field



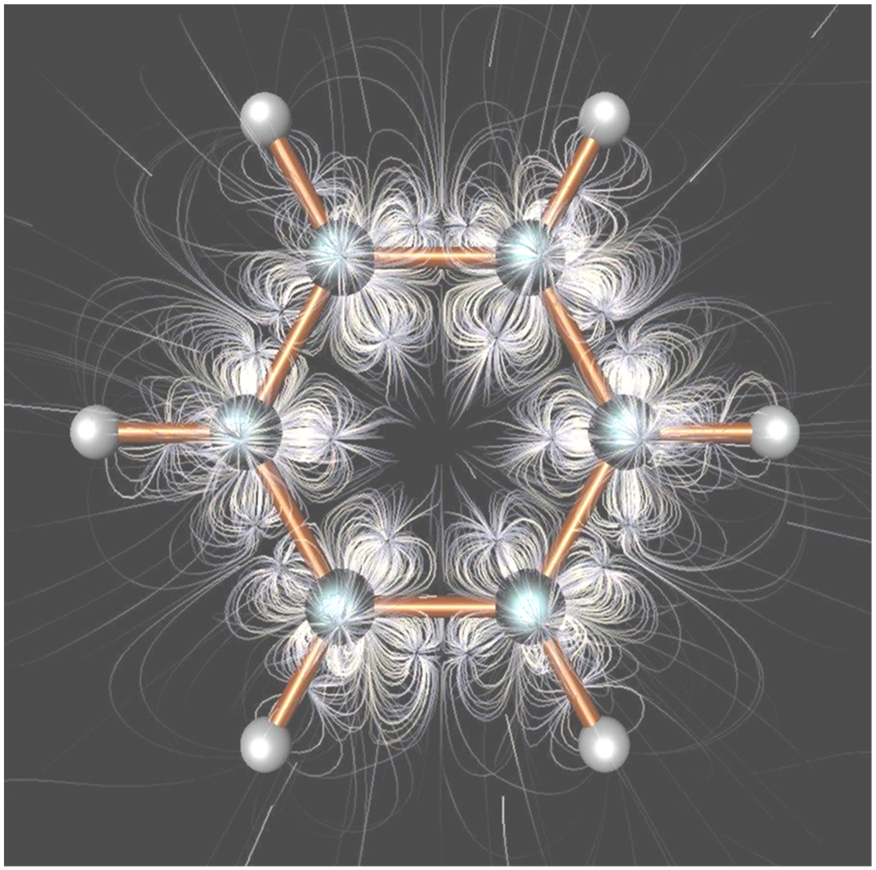
Visualization Examples

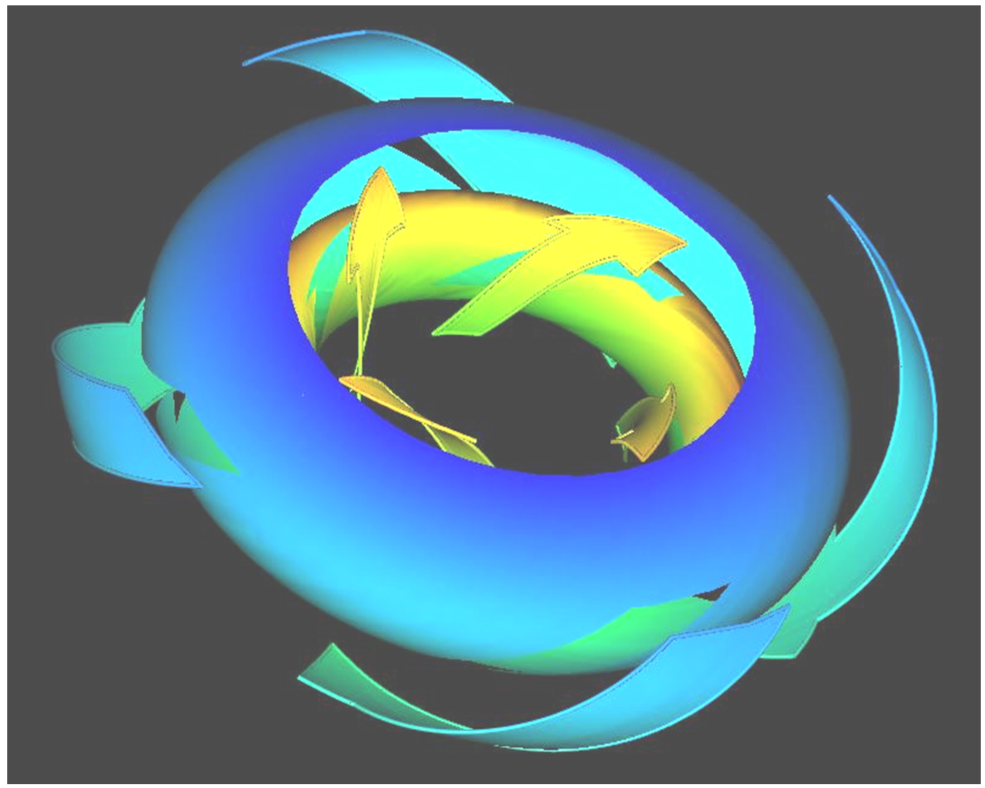
data	description	visualization example
$R^2 \rightarrow R^2$	2D-vector field	hedgehog plot, LIC, streamlets, etc



Visualization Examples

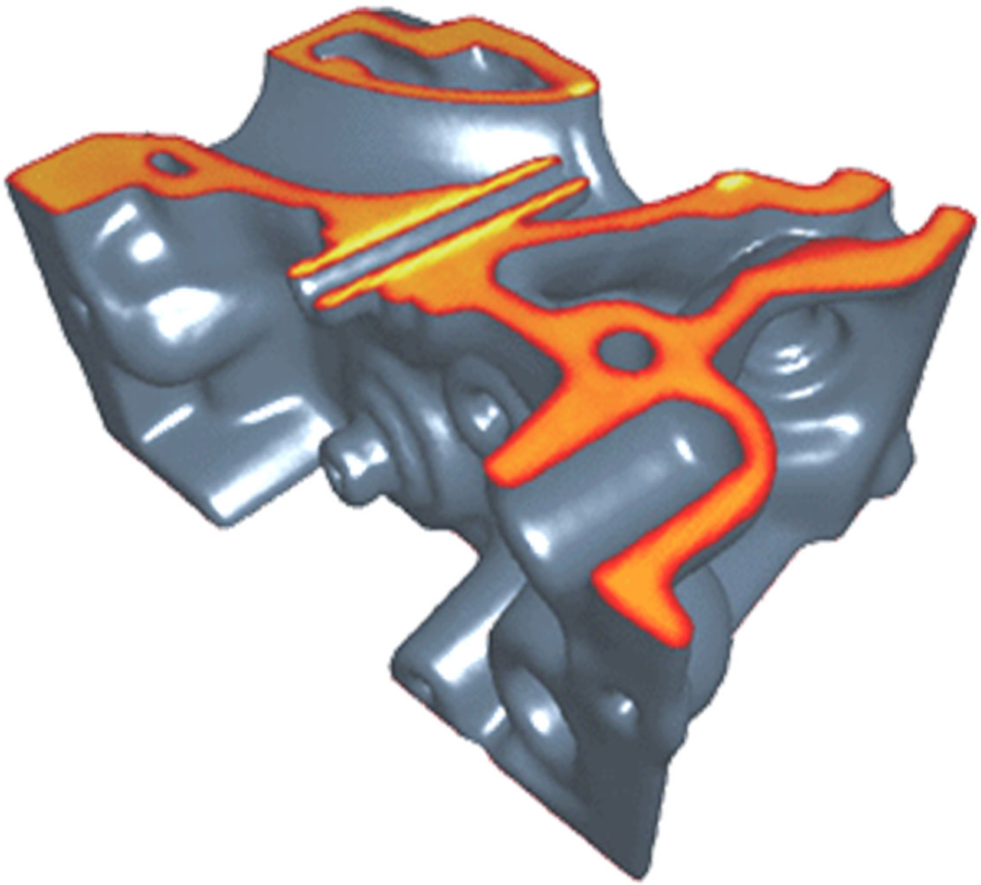
data	description	visualization example
$R^3 \rightarrow R^3$	3D-flow	streamlines, streamsurfaces

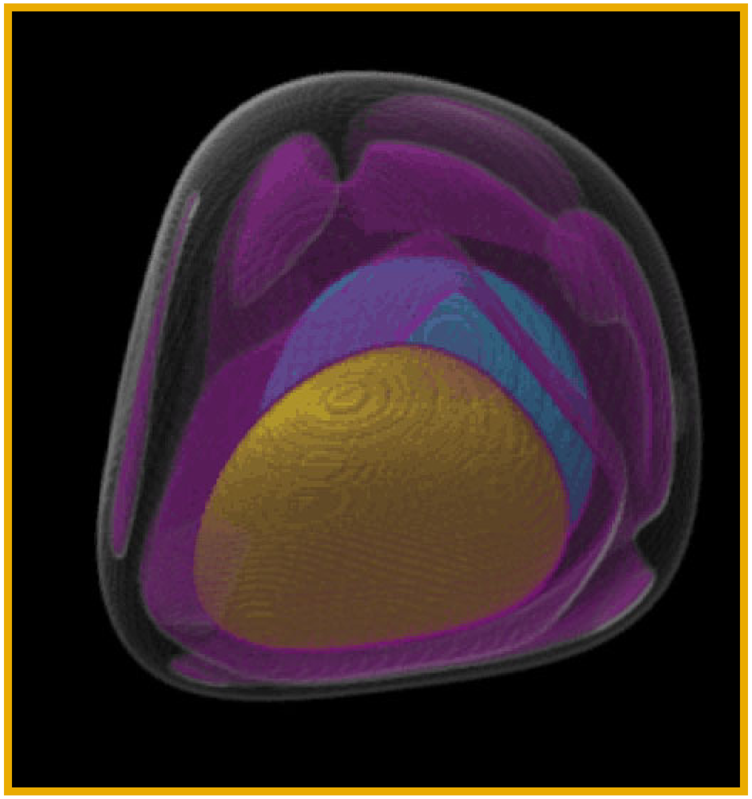




Visualization Examples

data	description	visualization example
$R^3 \to R^1$	3D-densities	iso-surfaces in 3D, volume rendering



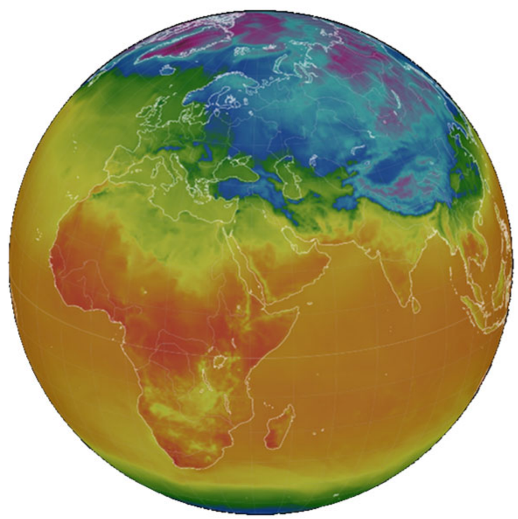


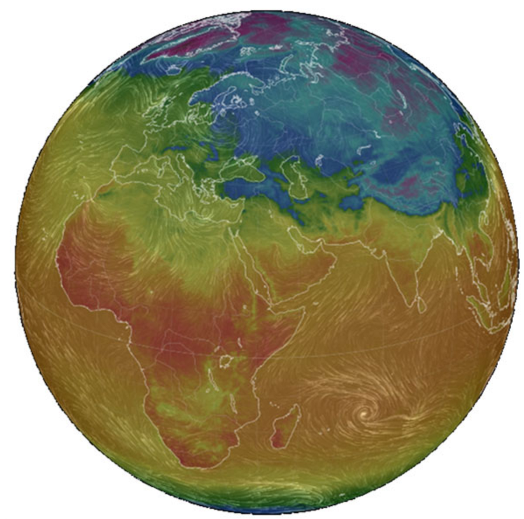
Domain is Not Always Euclidean

Manifolds

…

- Scalar, vector, tensor fields on manifolds

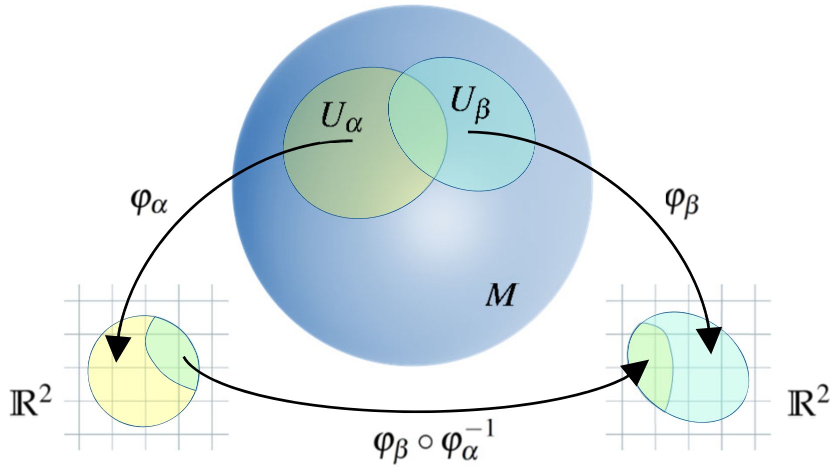


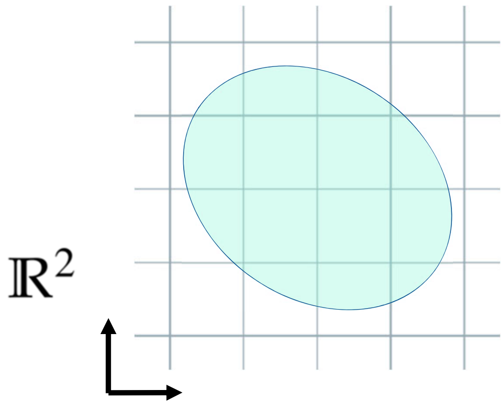


Topological Manifolds

Every point of an *n*-manifold is homeomorphic (topologically equivalent) to a region of $\mathbb{R}^n$

Think about being able to assign coordinates to a region: coordinate chart; (collection of charts: atlas)



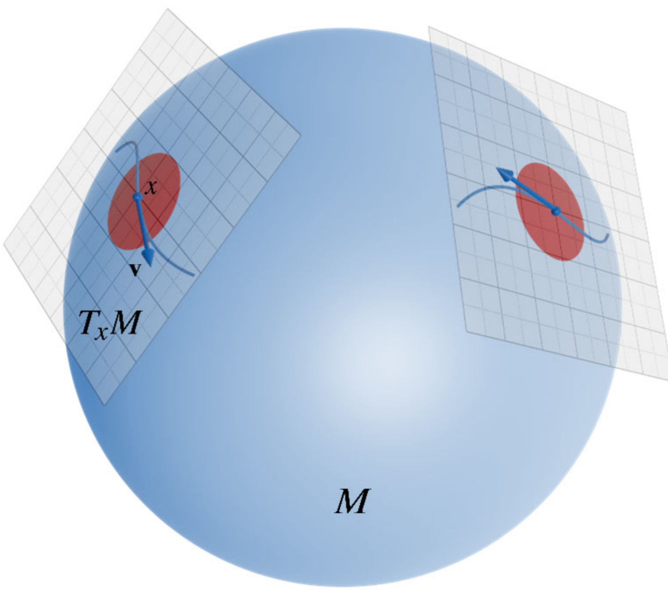


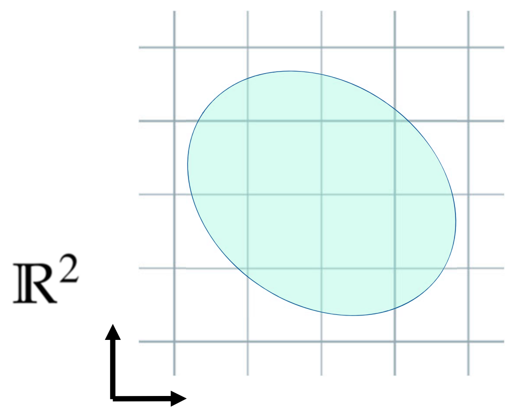
Smooth Manifolds

Well-defined tangent space at every point

- Dimensionality of each tangent space is the same as that of manifold

Enables calculus on manifolds (and vector fields, tensor fields, …)





Sampled Functions and Data Structures

Data Representation

- Discrete (sampled) representations
  - The objects we want to visualize are often ‘continuous’
  - But in most cases, the visualization data is given only at discrete locations in space and/or time
  - Discrete structures consist of samples, from which grids/meshes consisting of cells are generated
- Primitives in different dimensions

dimension	cell	mesh
0D	points	
1D	lines (edges)	polyline(–gon)
2D	triangles, quadrilaterals (rectangles)	2D mesh
3D	tetrahedra, prisms, hexahedra	3D mesh

Domain

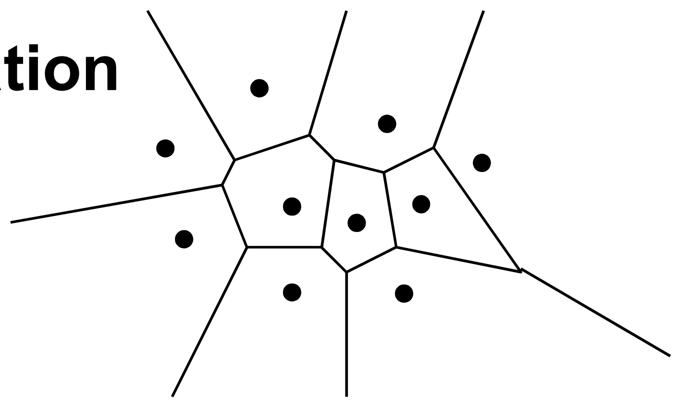
- The (geometric) shape of the domain is determined by the positions of sample points
- Domain is characterized by
  - Dimensionality: 0D, 1D, 2D, 3D, 4D, …
  - Influence: How does a data point influence its neighborhood?
  - Structure: Are data points connected? How? (Topology)

Domain

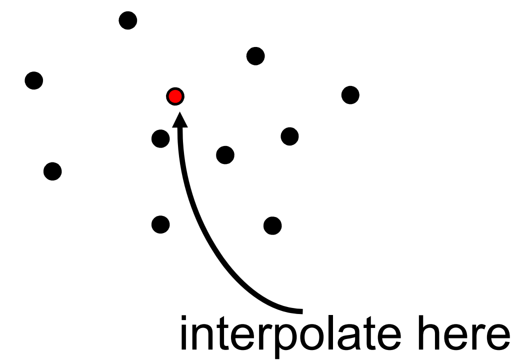
- Influence of data points
  - Values at sample points influence the data distribution in a certain region around these samples
  - To reconstruct the data at arbitrary points within the domain, the distribution of all samples has to be calculated
- Point influence
  - Only influence on point itself
- Local influence
  - Only within a certain region
    - Voronoi diagram
    - Cell-wise interpolation (see later in course)
- Global influence
  - Each sample might influence any other point within the domain
    - Material properties for whole object
    - Scattered data interpolation

Domain

- Voronoi diagram
  - Construct a region around each sample point that covers all points that are closer to that sample than to every other sample
  - Each point within a certain region gets assigned the value of the sample point
  - **Nearest-neighbor interpolation**



Domain



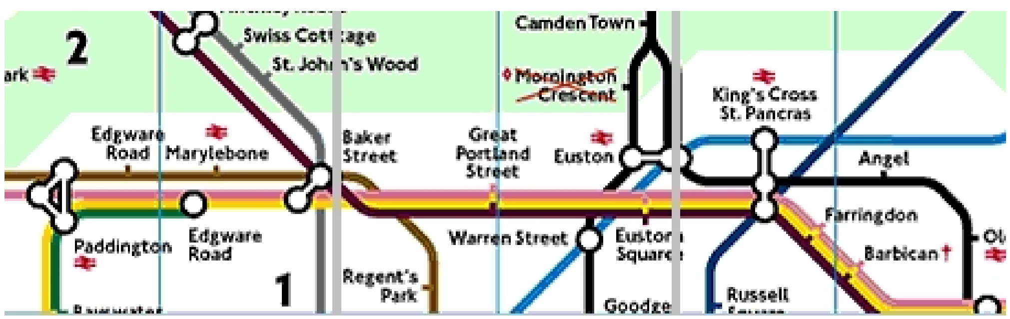
- Scattered data interpolation
  - At each point the weighted average of all sample points in the domain is computed
  - Weighting functions determine the support of each sample point
    - Radial basis functions simulate decreasing influence with increasing distance from samples
  - Schemes might be non-interpolating and expensive in terms of numerical operations

Data Structures

- Requirements:
  - Efficiency of accessing data
  - Space efficiency
  - Lossless vs. lossy
  - Portability
    - Binary – less portable, more space/time efficient
    - Text – human readable, portable, less space/time efficient
- Definition
  - If points are arbitrarily distributed and no connectivity exists between them, the data is called scattered
  - Otherwise, the data is composed of cells bounded by grid lines
  - **Topology** specifies the structure (**connectivity**) of the data
  - **Geometry** specifies the **position** of the data

Data Structures

- Some definitions concerning topology and geometry
  - In topology, qualitative questions about geometrical structures are the main concern
    - Does it have any holes in it?
    - Is it all connected together?
    - Can it be separated into parts?
- Underground map does not tell you how far one station is from the other, but rather how the lines are connected (topological map)



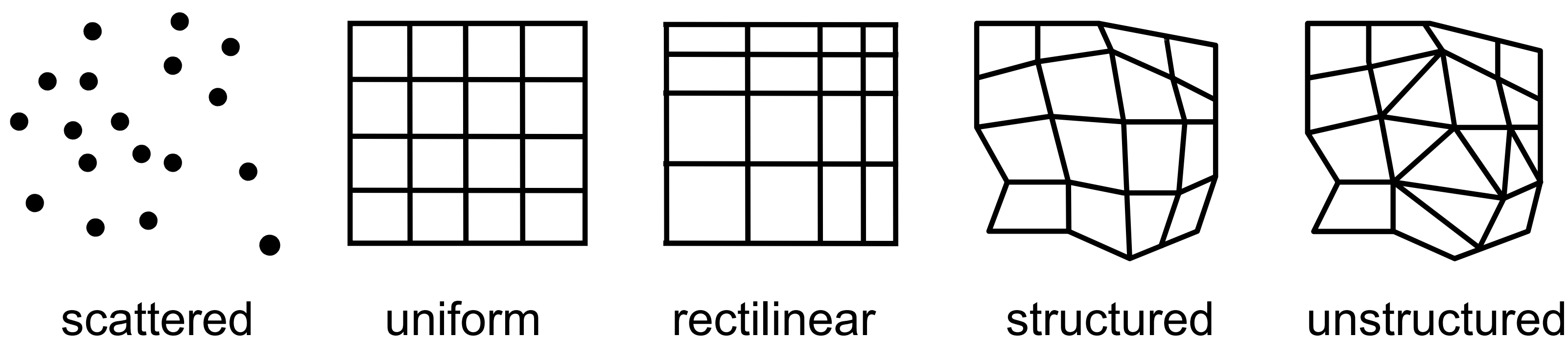
Grids – General Questions

Important questions:

- Which data organization is optimal?
- Where do the data come from?
- Is there a neighborhood relationship?
- How is the neighborhood info stored?
- How is navigation within the data possible?
- What calculations with the data are possible ?
- Are the data structured (regular/irregular topology)?

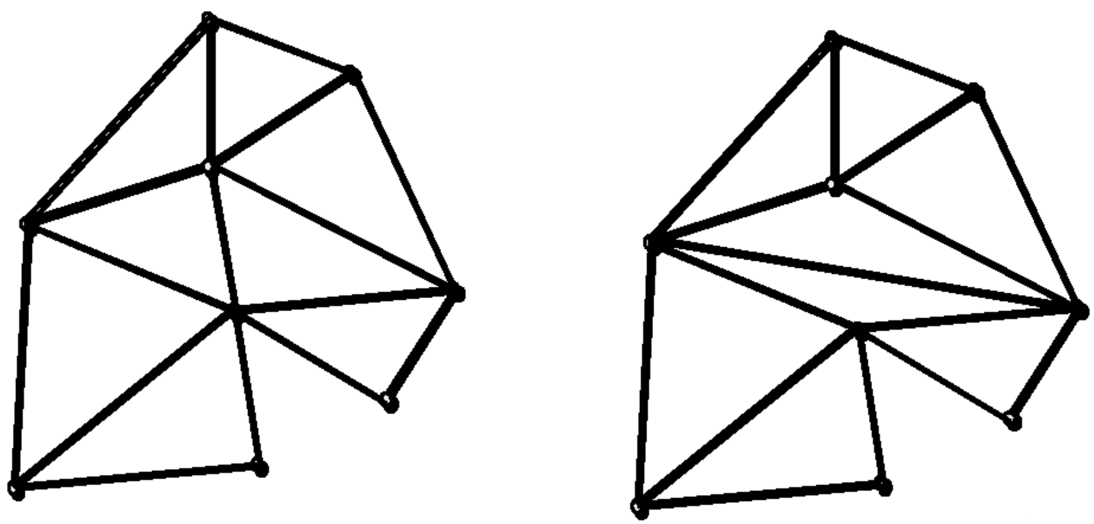
Data Structures

- Grid types
  - Grids differ substantially in the cells (basic building blocks) they are constructed from and in the way the topological information is given



Data Structures

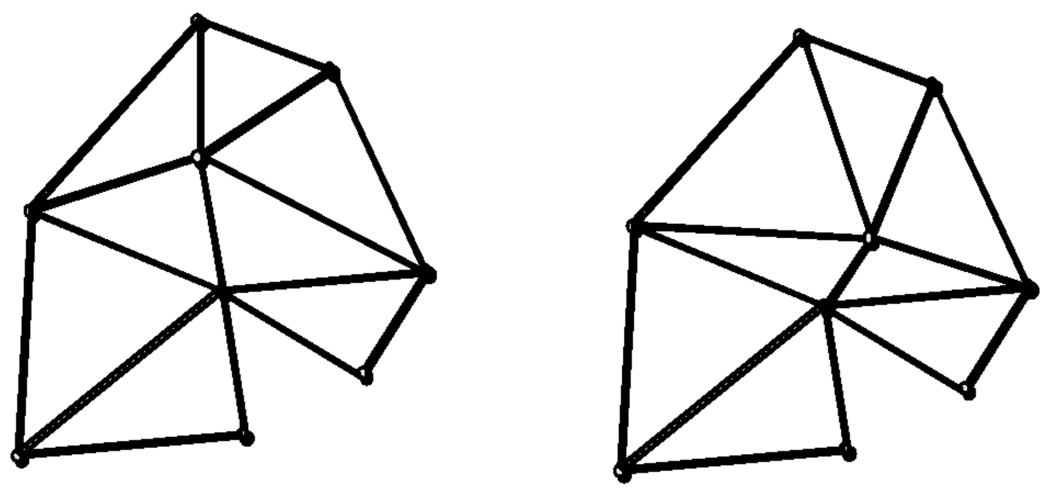
- Topology
  - Properties of geometric shapes that remain unchanged even when under distortion



Same geometry (vertex positions), different topology (connectivity)

Data Structures

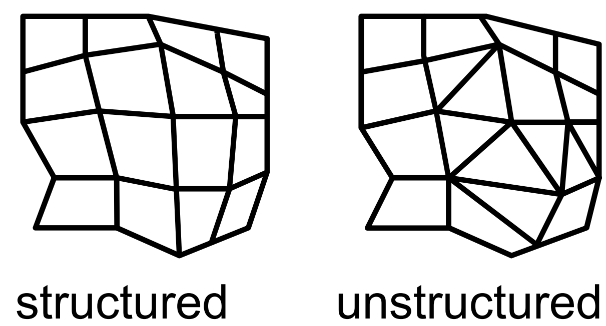
- Topologically equivalent
  - Things that can be transformed into each other by stretching and squeezing, without tearing or sticking together bits which were previously separated



topologically equivalent

Data Structures

- Structured and unstructured grids can be distinguished by the way the elements or cells meet
- Structured grids
  - Have a regular topology and regular / irregular geometry
- Unstructured grids
  - Have irregular topology and geometry



Thank you.

Thanks for material

- Helwig Hauser
- Eduard Gröller
- Daniel Weiskopf
- Torsten Möller
- Ronny Peikert
- Philipp Muigg
- Christof Rezk-Salama