Introduction to topological data analysis for lattice field theory

Jeffrey Giansiracusa
August 2024

Durham University

Collaboration with:
Biagio Lucini, Tin Sulejmanpasic, Xavier Crean, Nick Sale

1. Homology

Counts holes / voids

1. Homology

Counts holes / voids

2. Persistent homology

Shows how the counts evolve with a parameter

Counting holes

Counting holes

0

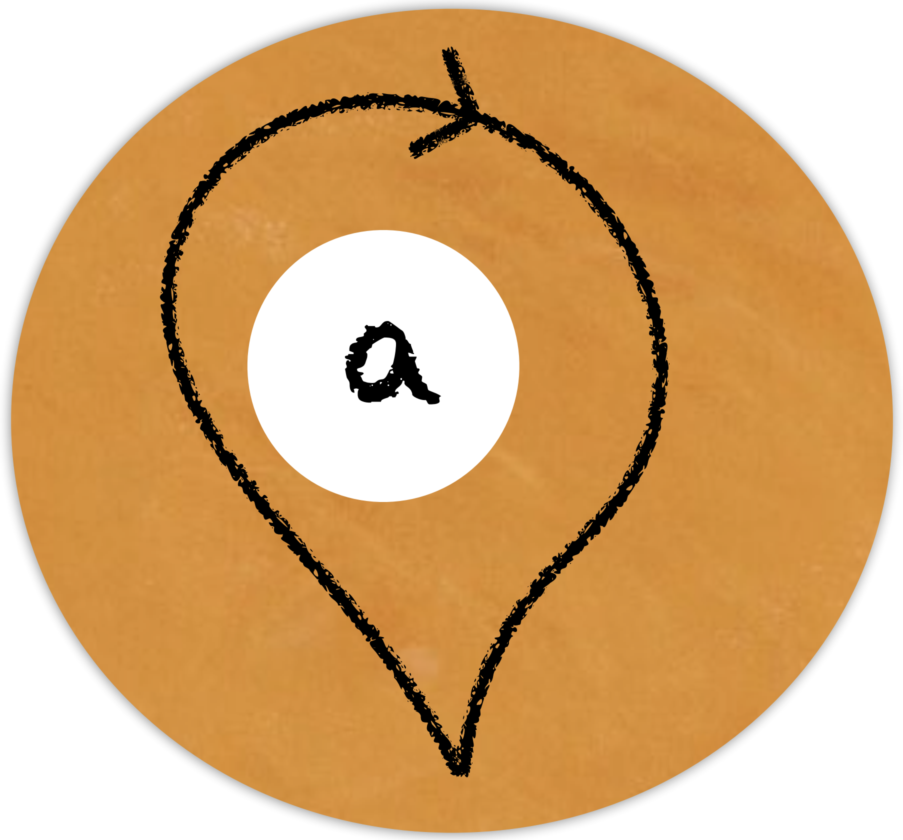
Counting holes

0

1

Counting holes

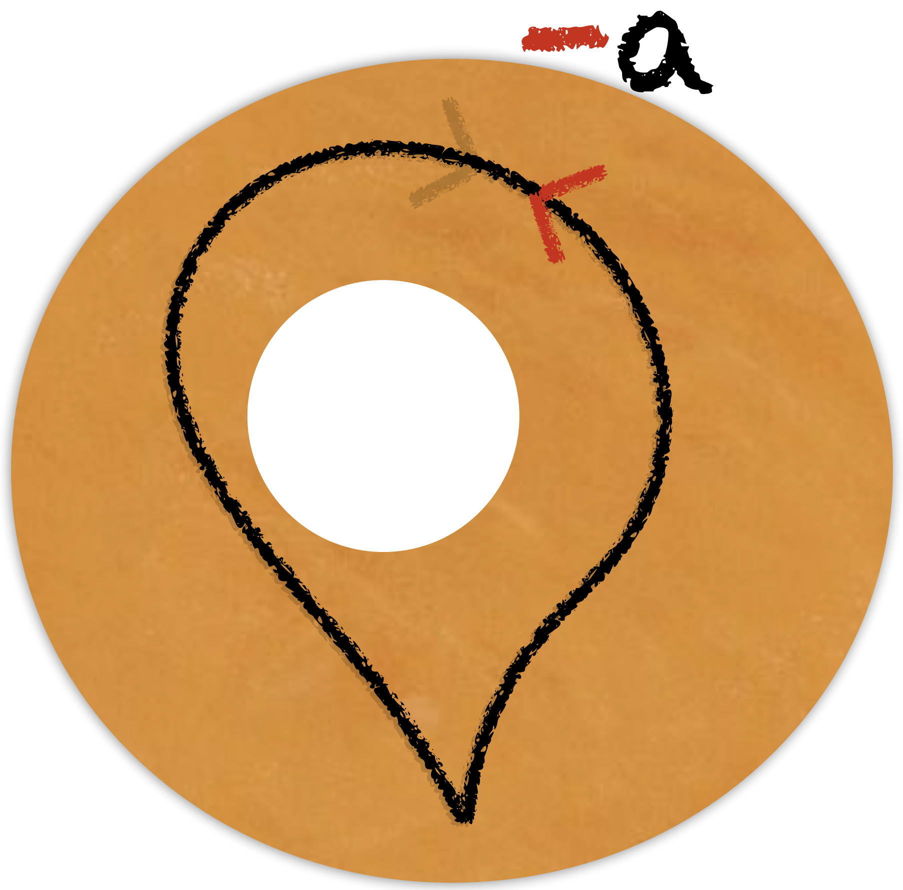
0



1

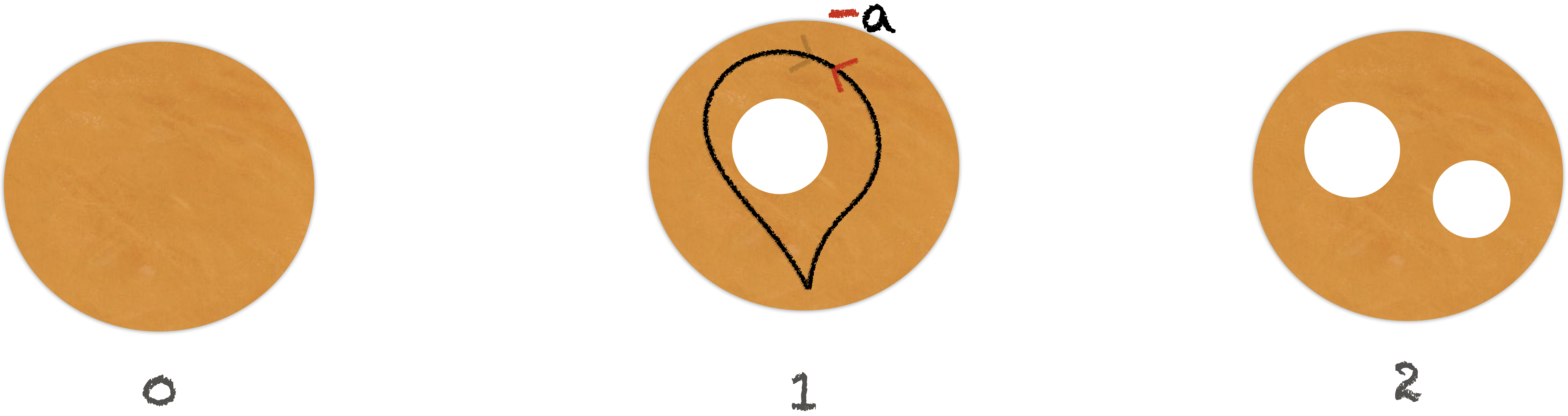
Counting holes

0

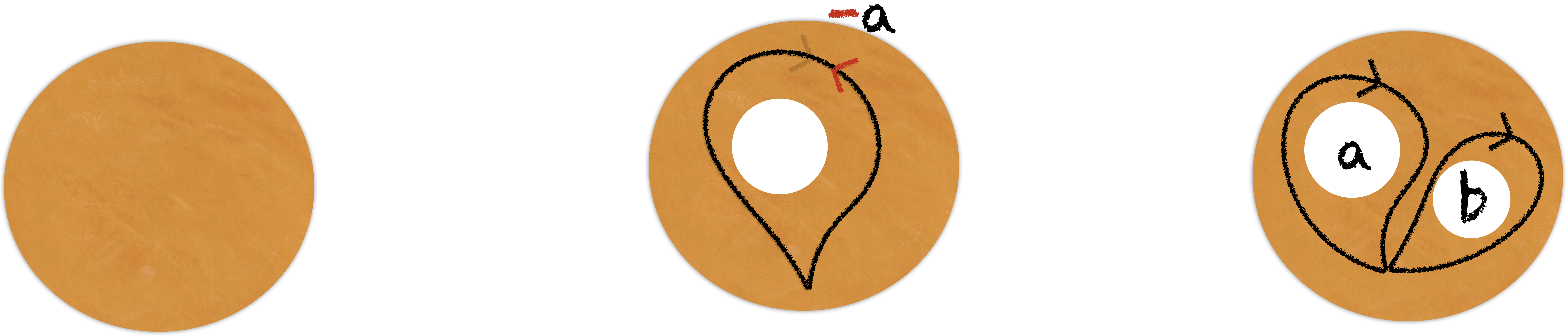


1

Counting holes



Counting holes

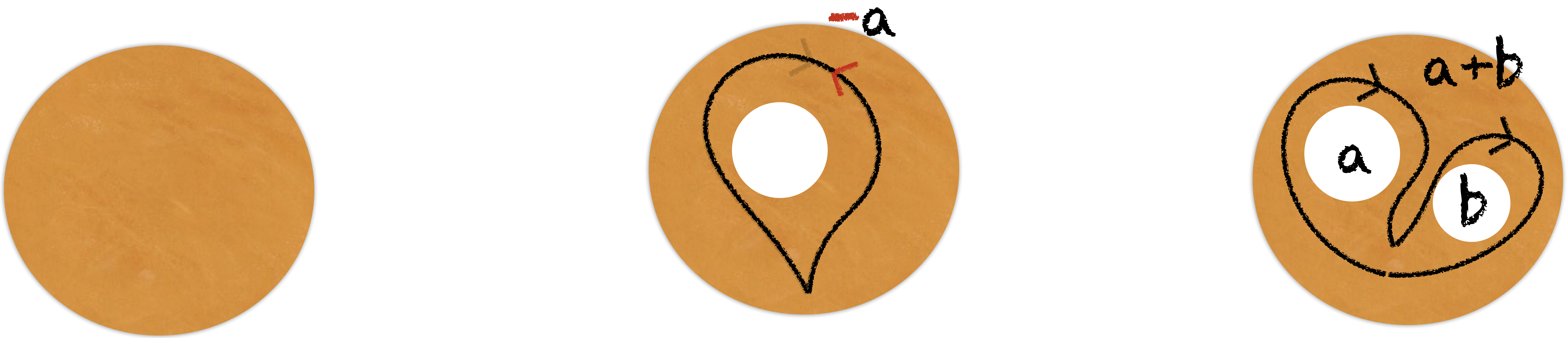


0

1

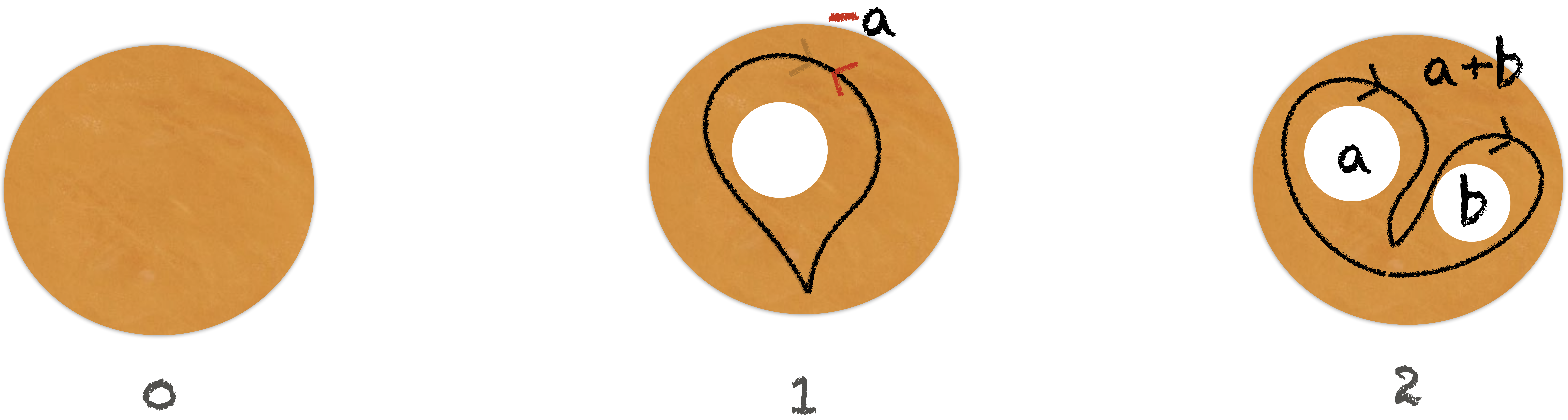
2

Counting holes



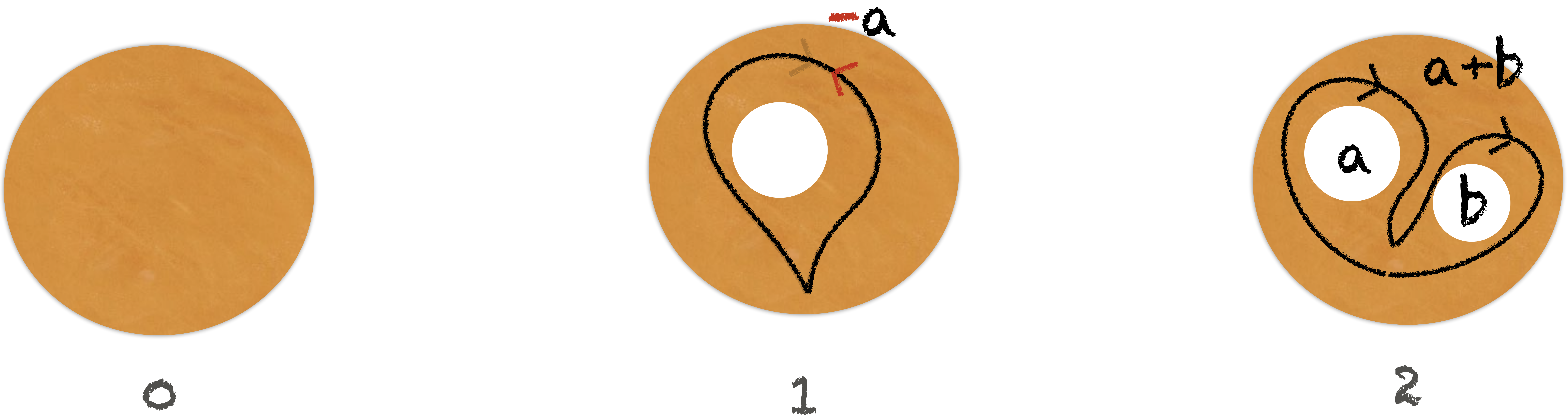
0 1 2

Counting holes



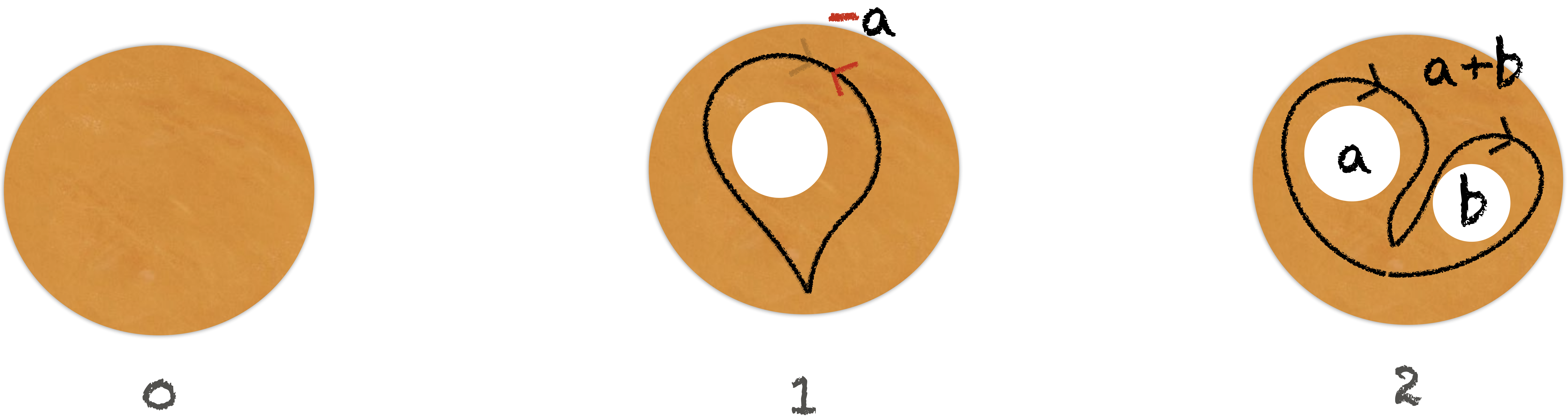
- Can do arithmetic with holes

Counting holes



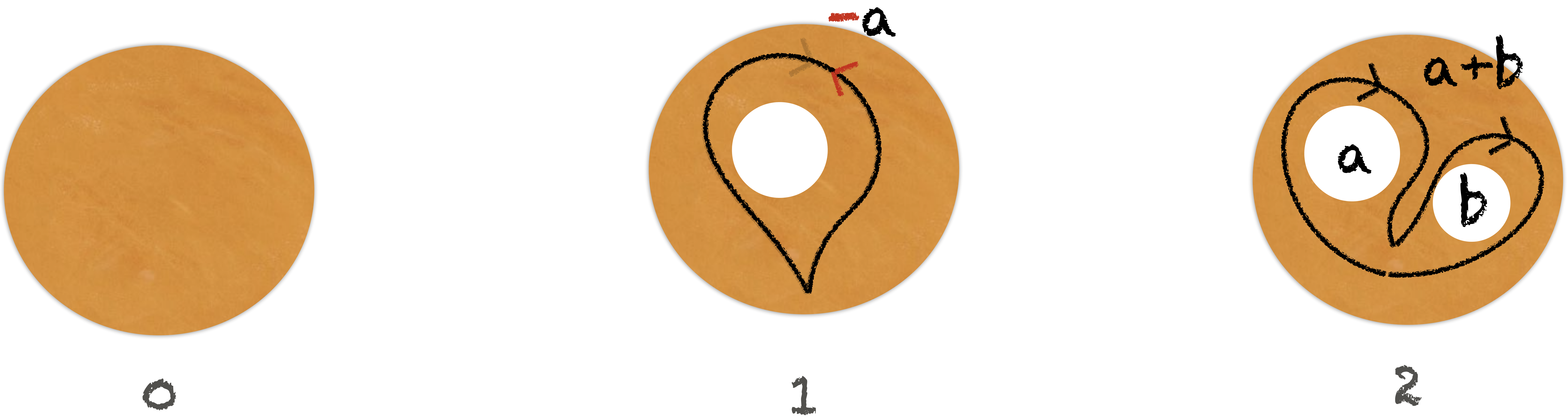
- Can do arithmetic with holes
  - There is a vector space H_1 of holes

Counting holes



- Can do arithmetic with holes
  - There is a vector space H_1 of holes
  - Number of holes = dim H_1

Counting holes



- Can do arithmetic with holes
  - There is a vector space H_1 of holes
  - Number of holes = dim H_1

Homology

Homology

Input: a geometric object X

Homology

Input: a geometric object X

Output: vector spaces

$$H_0(X), \quad H_1(X), \quad H_2(X), \quad H_3(X), \ \ldots$$

Homology

Input: a geometric object X

Output: vector spaces

$$H_0(X), \quad H_1(X), \quad H_2(X), \quad H_3(X), \quad \ldots$$

Connected components — Holes — Voids — Higher voids

Homology

Input: a geometric object X

Output: vector spaces

$$H_0(X), \quad H_1(X), \quad H_2(X), \quad H_3(X), \quad \ldots$$

Connected components | Holes | Voids | Higher voids

Dimension = number of holes/voids

Homology

Input: a geometric object X ← What is X?

Output: vector spaces

$$H_0(X), \quad H_1(X), \quad H_2(X), \quad H_3(X), \quad \ldots$$

Connected components — Holes — Voids — Higher voids

Dimension = number of holes/voids

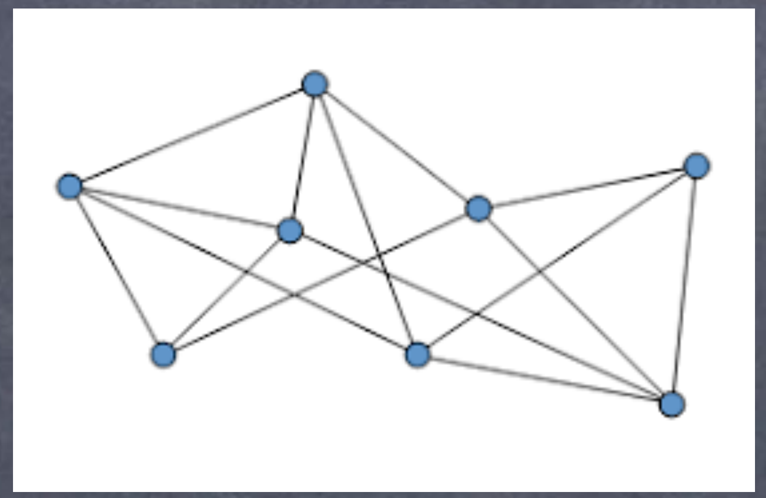
Examples

Dim H_0	
Dim H_1	
Dim H_2	
Dim H_3	

Examples

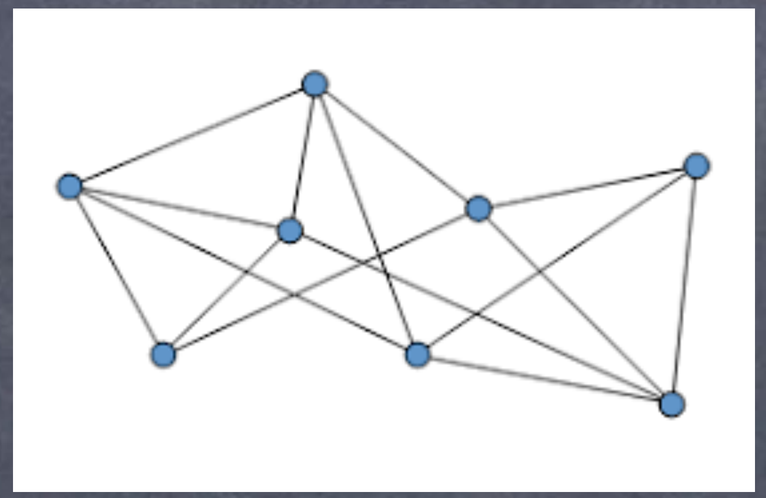
	Circle
	
Dim H_0	1
Dim H_1	1
Dim H_2	0
Dim H_3	$\vdots$

Examples

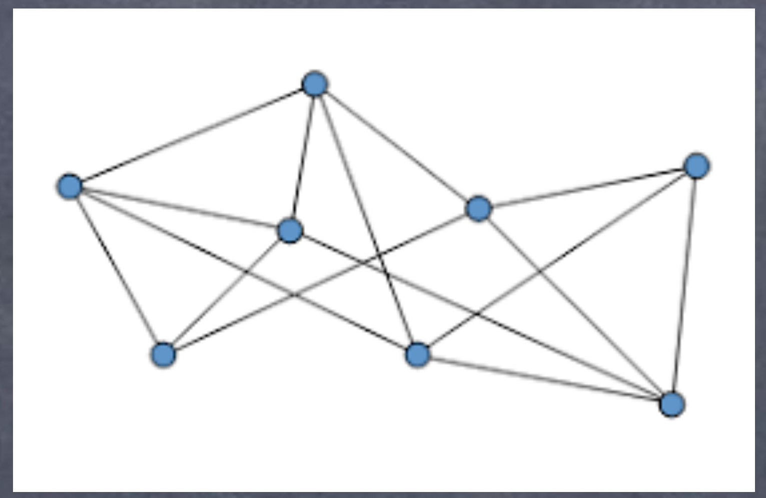


	Circle	Graph
Dim H_0	1	Number of components
Dim H_1	1	Number of loops
Dim H_2	0	0
Dim H_3	$\vdots$	$\vdots$

Examples

	Circle	Graph	Sphere
			
Dim H_0	1	Number of components	1
Dim H_1	1	Number of loops	0
Dim H_2	0	0	1
Dim H_3	$\vdots$	$\vdots$	0

Examples

	Circle	Graph	Sphere	surface of genus g
				
Dim H_0	1	Number of components	1	1
Dim H_1	1	Number of loops	0	2g
Dim H_2	0	0	1	1
Dim H_3	⋮	⋮	0	0

Given a statistical system on a lattice

Given a statistical system on a lattice

Boltzmann distribution
+ Monte Carlo
→ configurations $\phi_1, \phi_2, \phi_3, \ldots$

Given a statistical system on a lattice

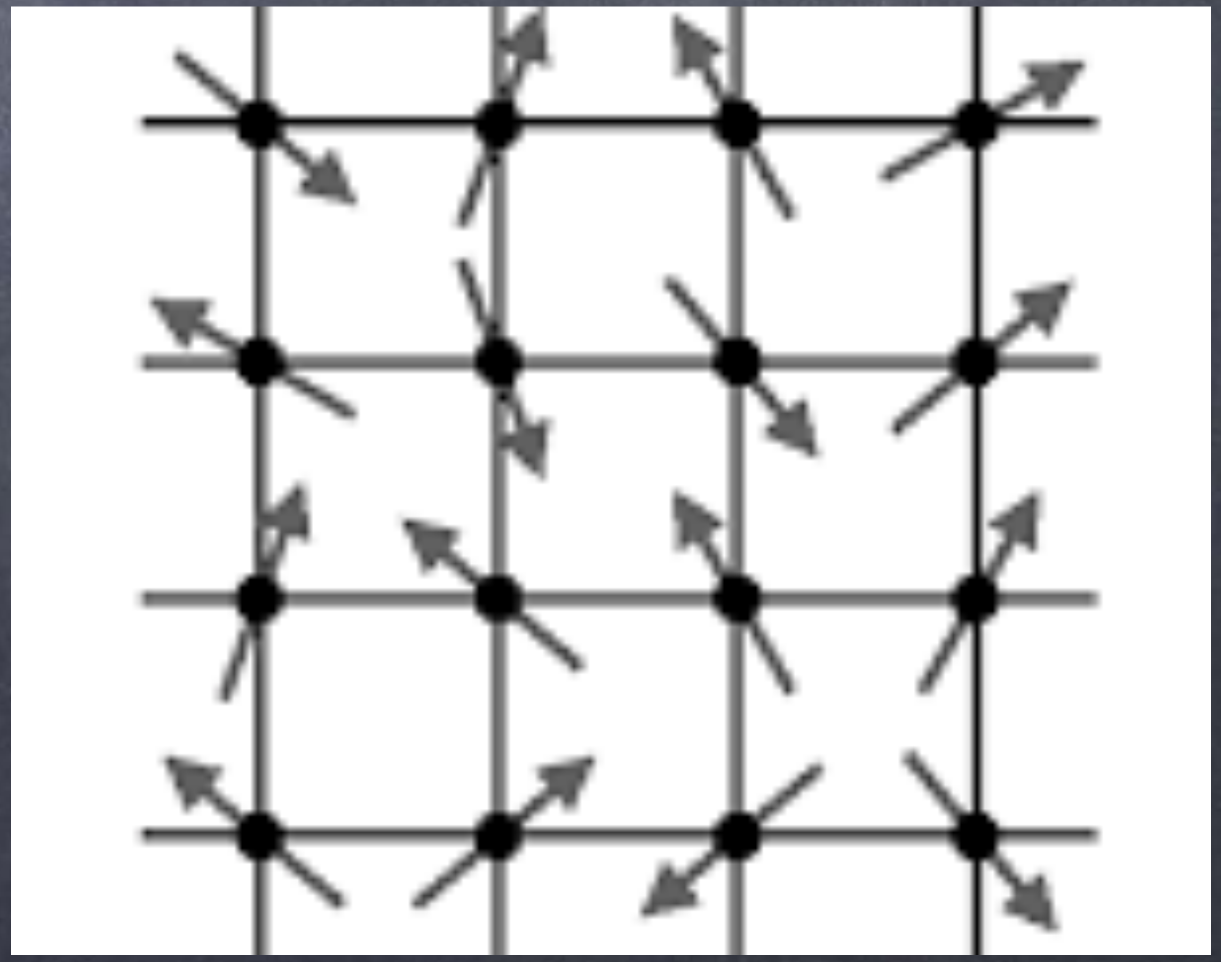
Boltzmann distribution
+ Monte Carlo
-> configurations $\phi_1,\ \phi_2,\ \phi_3,\ \ldots$

Given a statistical system on a lattice

Boltzmann distribution
+ Monte Carlo
-> configurations $\phi_1,\ \phi_2,\ \phi_3,\ \ldots$

E.g., 2d XY-model on an $L \times L$ lattice

$\phi_i \in U(1)^{L^2}$



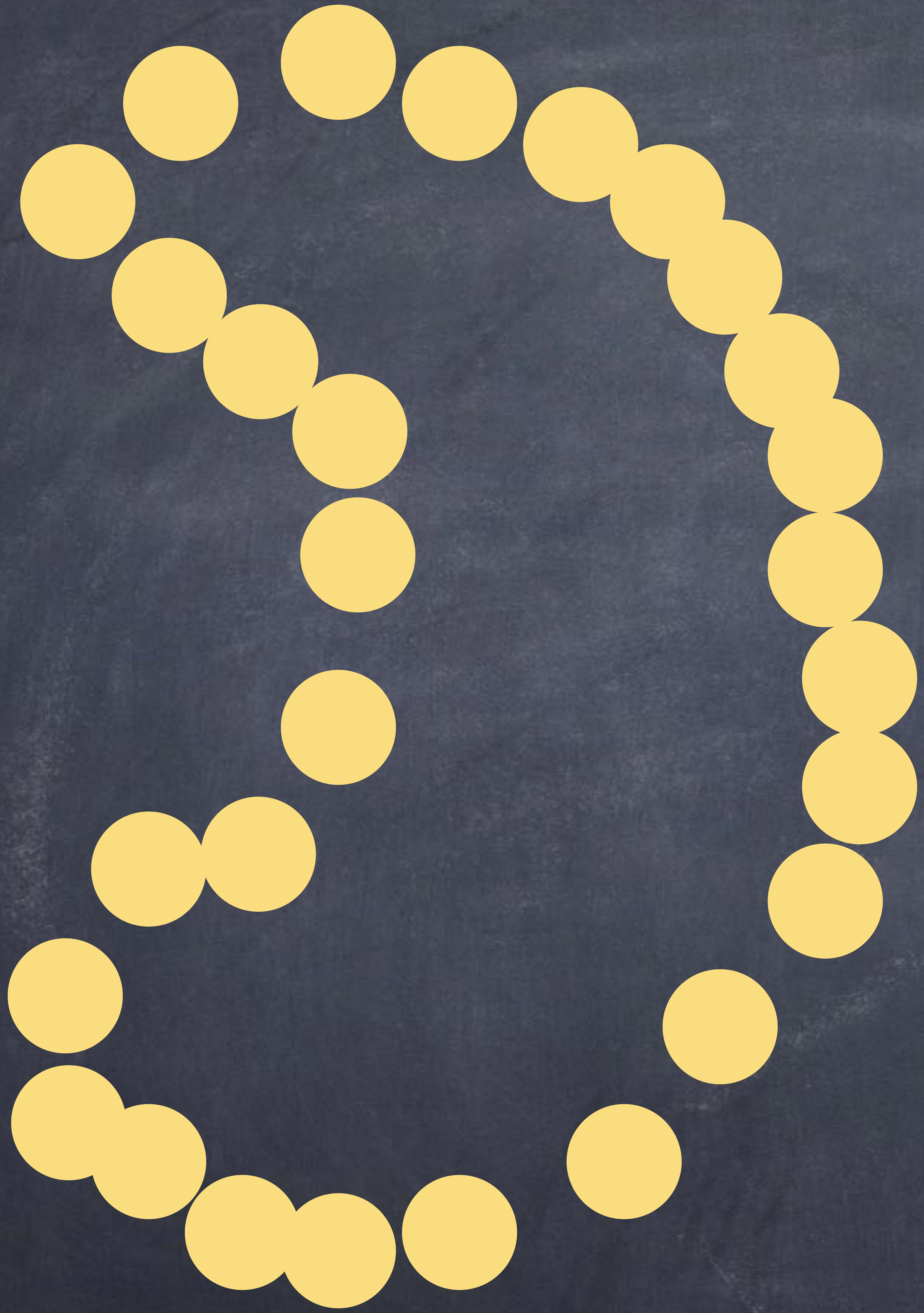
Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Choose a scale and compute homology $H_*(X)$



Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Choose a scale and compute homology $H_*(X)$

Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Choose a scale and compute homology $H_*(X)$

Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Choose a scale and compute homology $H_*(X)$

Which scale?

Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Choose a scale and compute homology $H_*(X)$

Which scale?
Use them all!

Methodology 1:

Quantify the topology of the (the dense part of) Boltzmann distribution on configuration space.

Choose a scale and compute homology $H_*(X)$

Which scale?
Use them all!
Persistent homology

Methodology 2:

Use homology as an observable.

Methodology 2:

Use homology as an observable.

Configuration ϕ

Methodology 2:

Use homology as an observable.

Configuration ϕ

$$\downarrow$$

Geometric object $X(\phi)$

Methodology 2:

Use homology as an observable.

Configuration ϕ

$$\downarrow$$

Geometric object $X(\phi)$

$$\downarrow$$

Homology $H_*(X(\phi))$

Methodology 2:

Use homology as an observable.

Configuration ϕ

$\downarrow$

Geometric object $X(\phi)$

$\downarrow$

Homology $H_*(X(\phi))$

Then do statistical analysis on the homology

Methodology 2:

Use homology as an observable.

Configuration ϕ

$\downarrow$

Geometric object $X(\phi)$

$\downarrow$

Homology $H_*(X(\phi))$

Then do statistical analysis on the homology

Configuration ϕ

$\downarrow$

Geometric object $X(\phi)$

Configuration ϕ

$\downarrow$

Geometric object $X(\phi)$

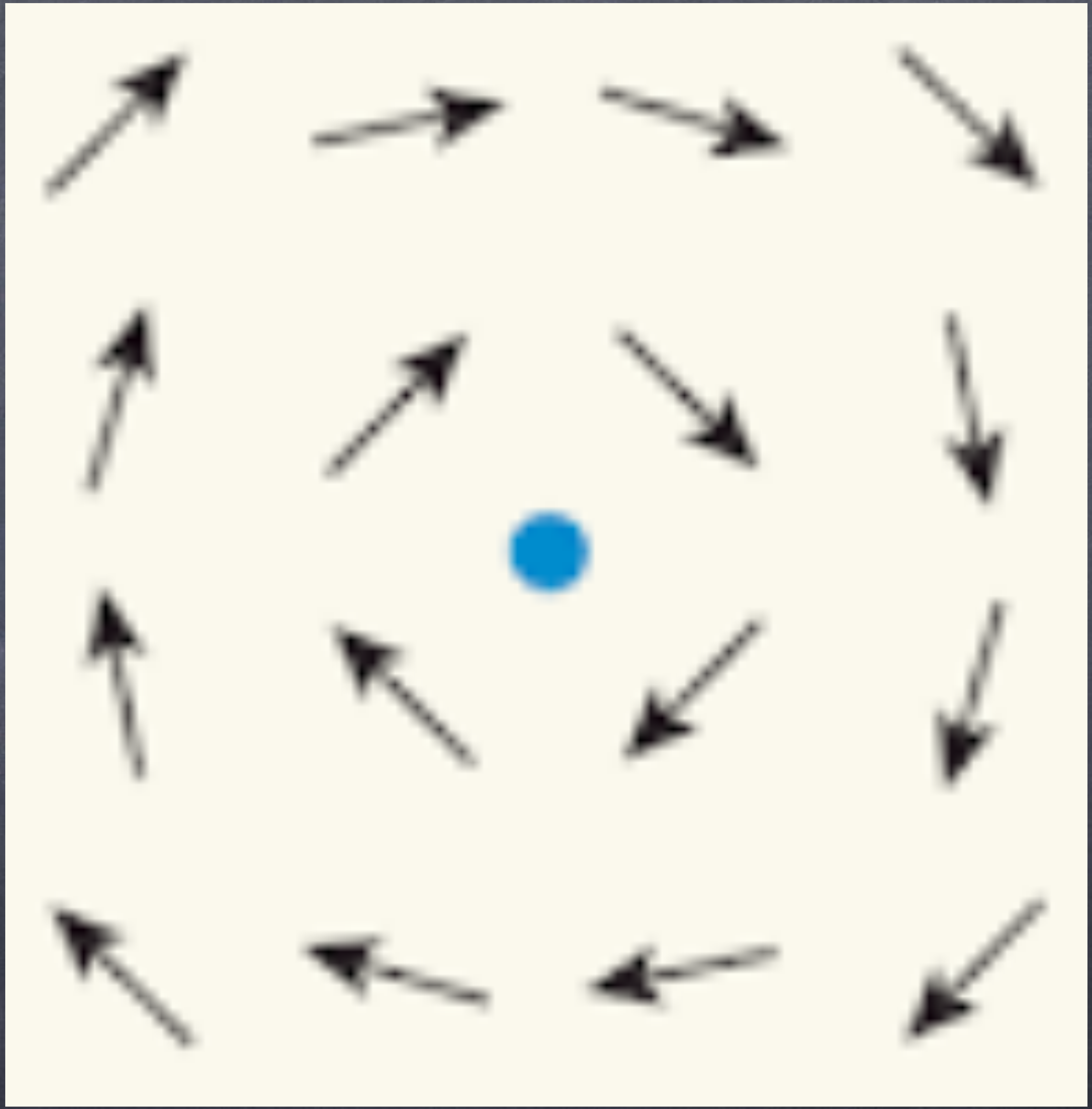
XY-model example

Configuration ϕ

$\downarrow$

Geometric object $X(\phi)$

XY-model example



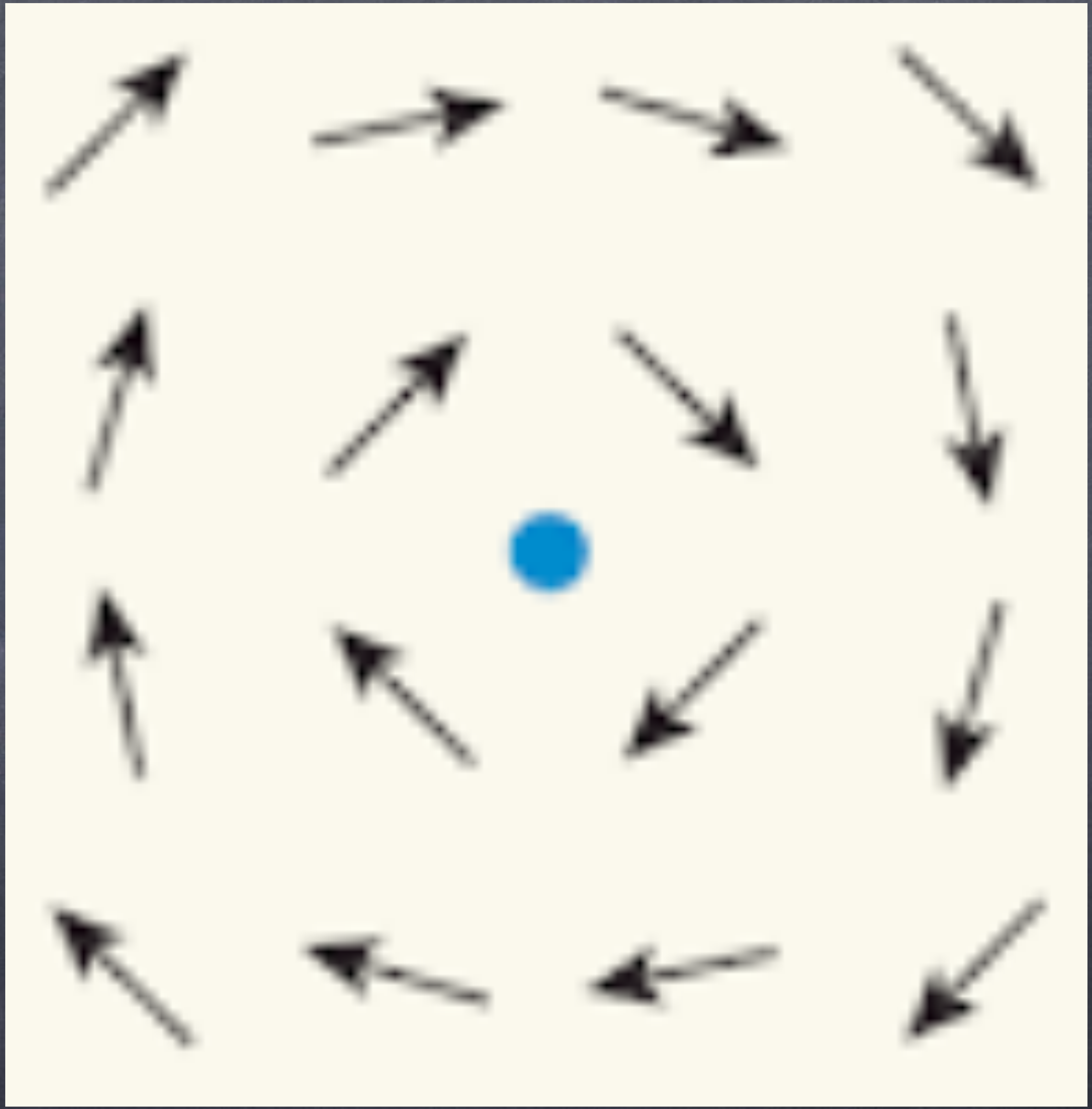
Configuration ϕ

$\downarrow$

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.



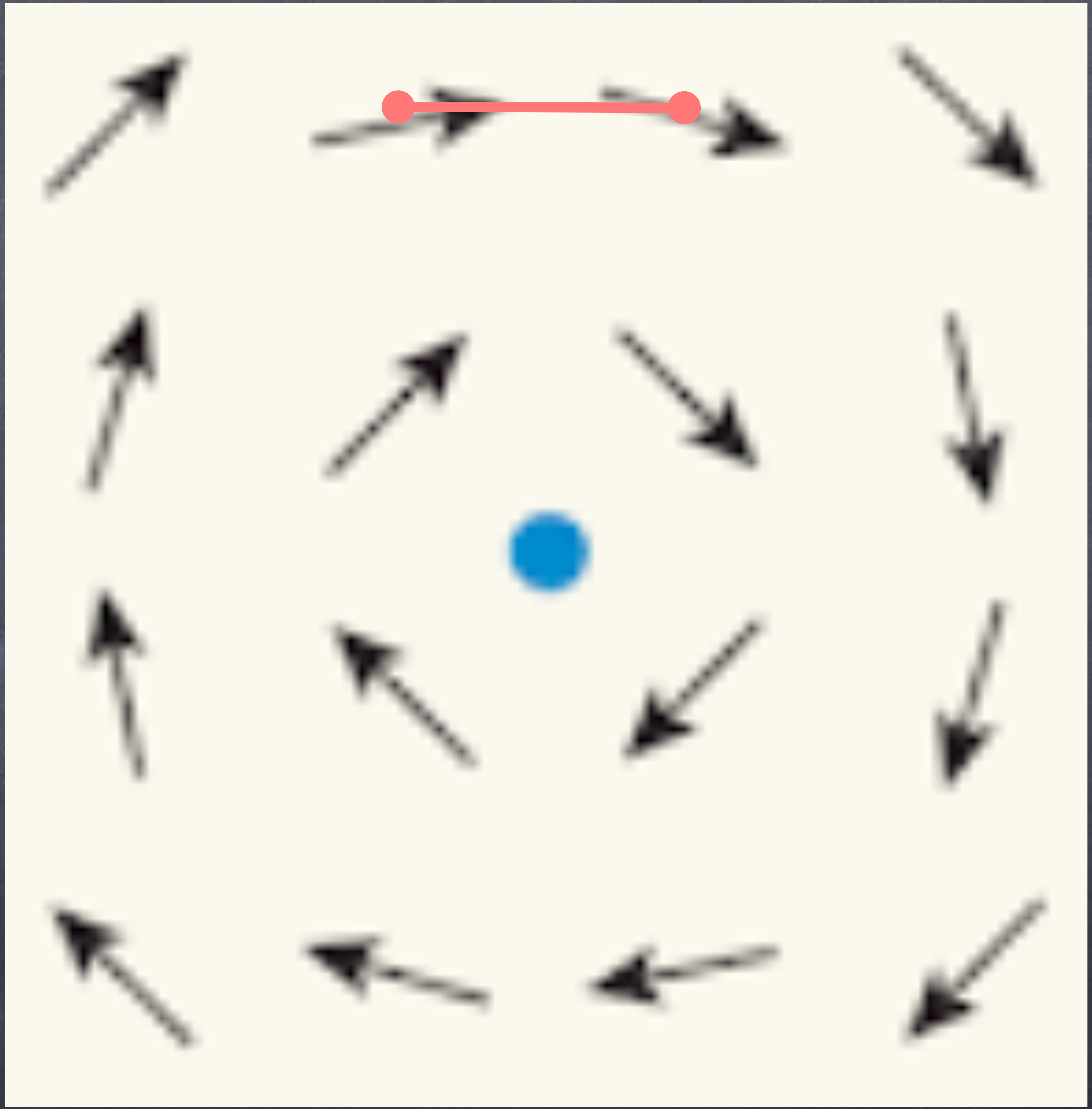
Configuration ϕ

↓

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.



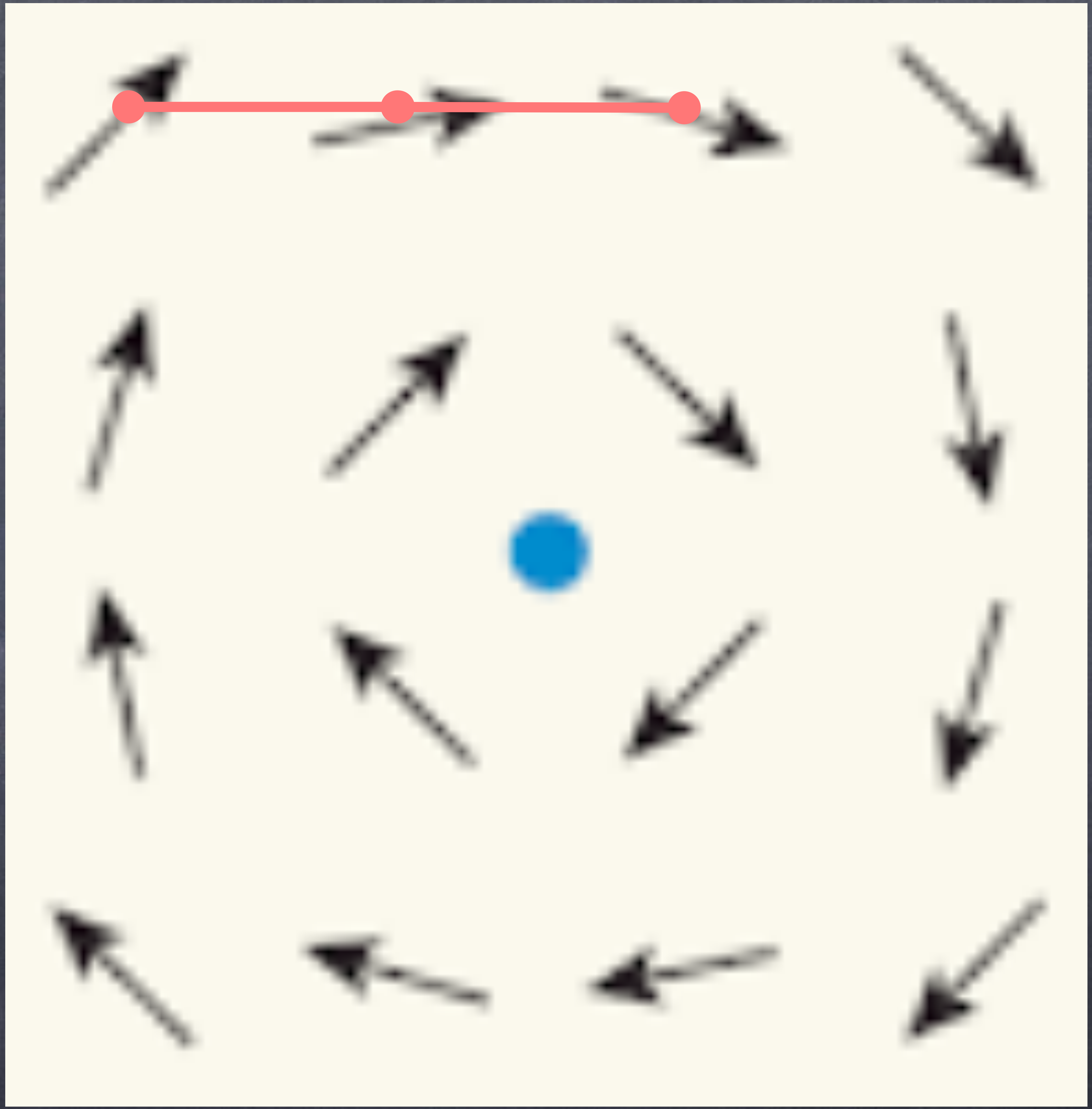
Configuration ϕ

↓

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.



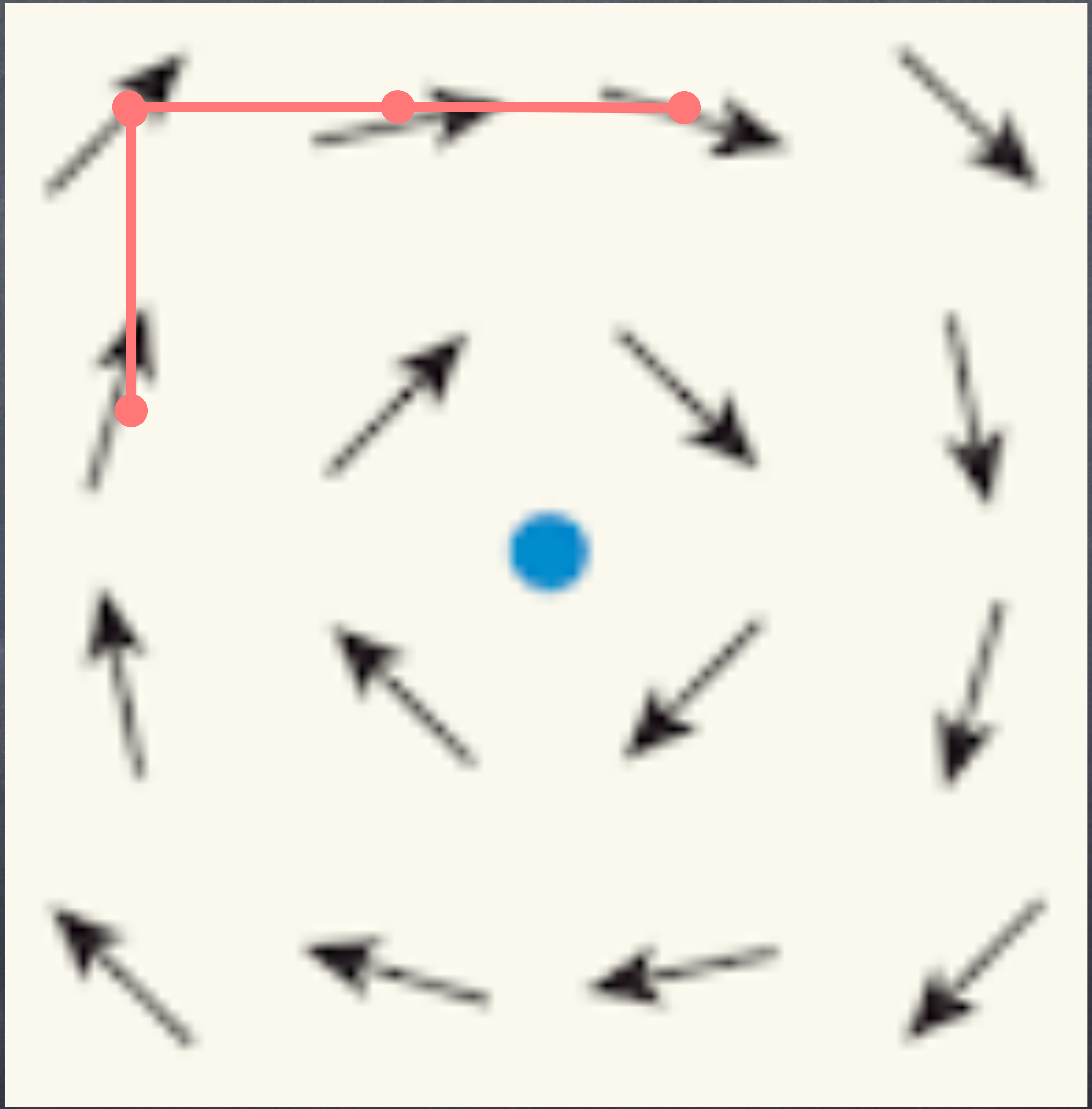
Configuration ϕ

↓

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.



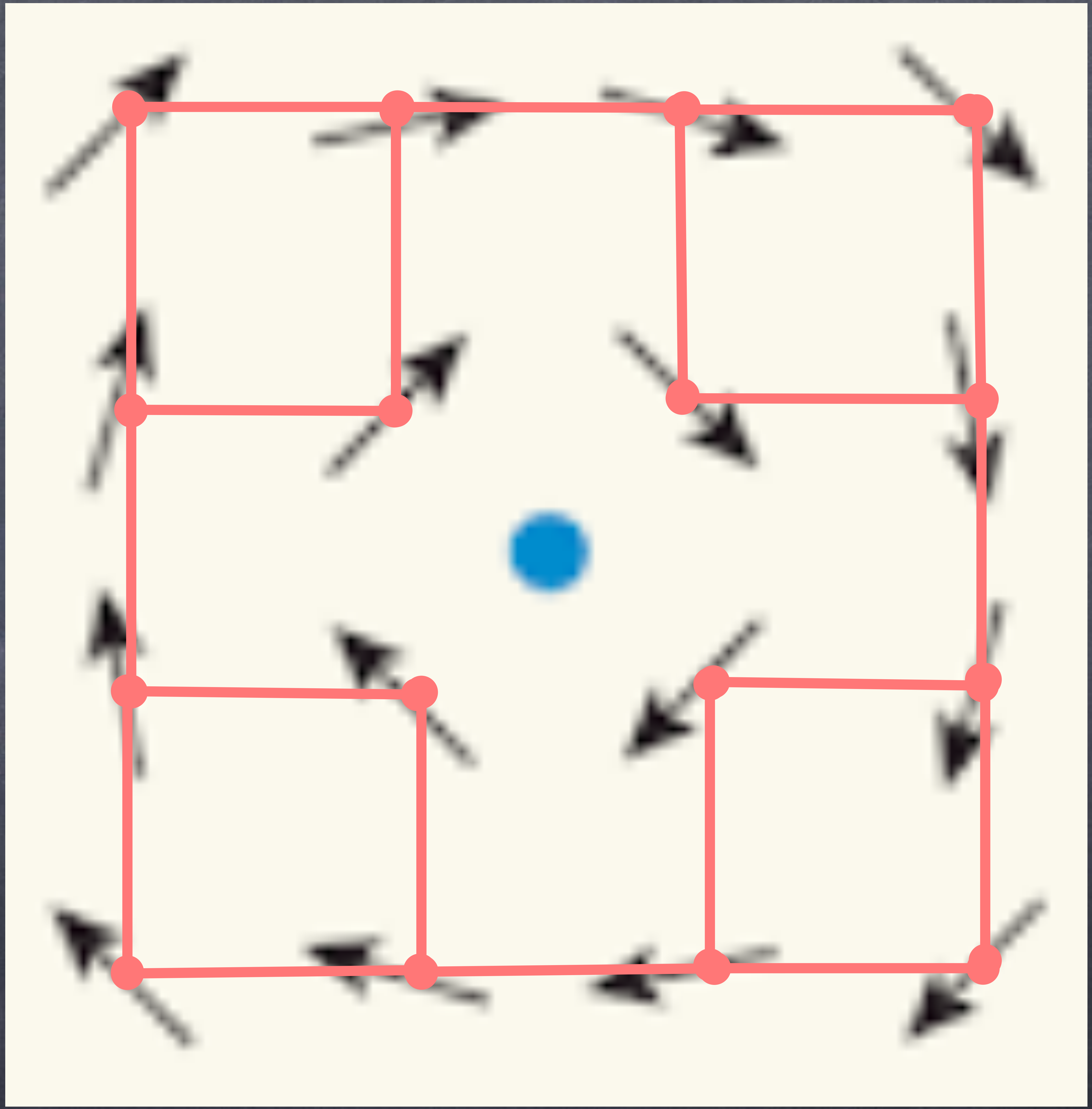
Configuration ϕ

↓

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.



Configuration ϕ

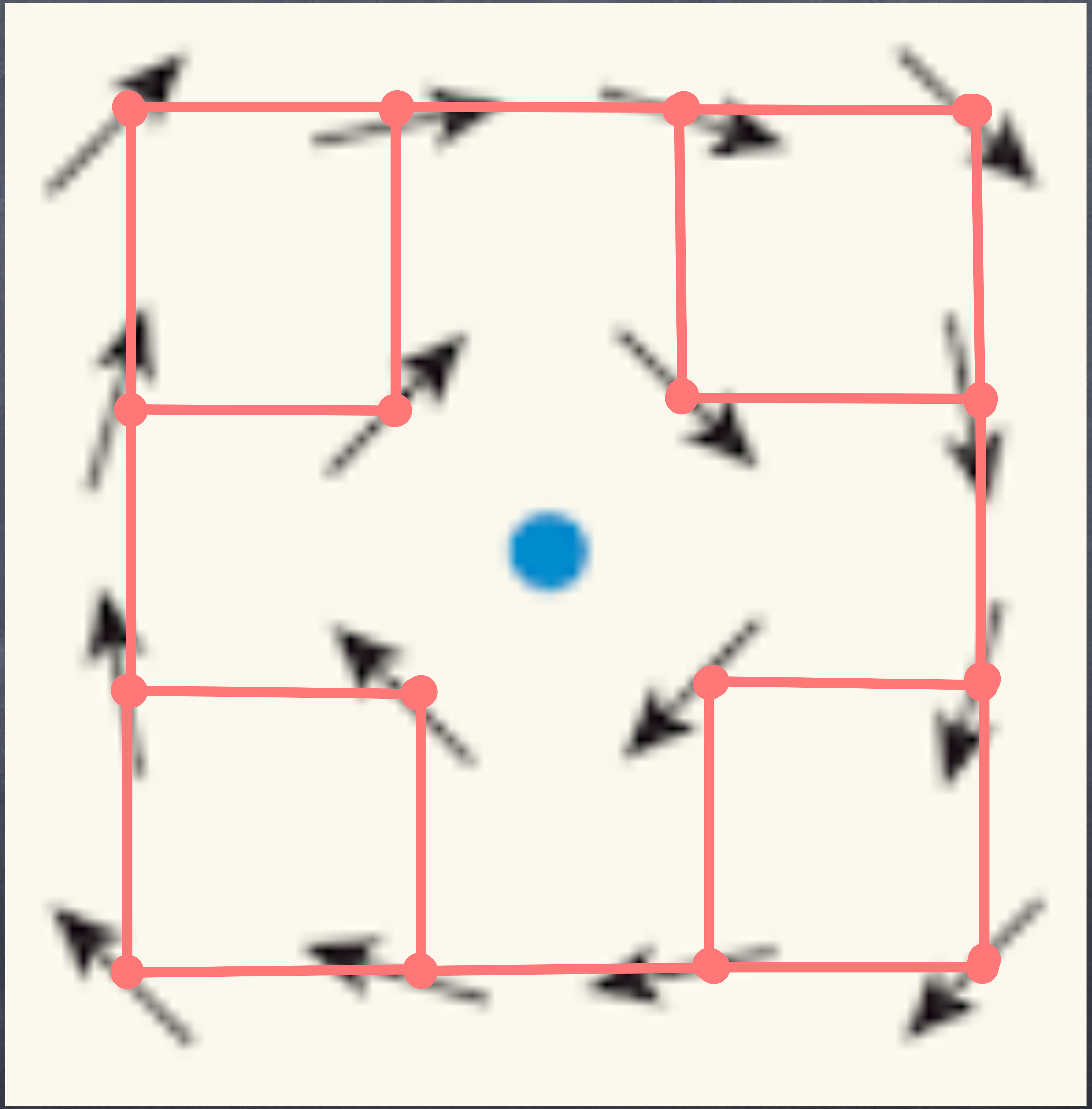
$\downarrow$

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.

2. Fill in a plaquette if all the edges are present.



Configuration ϕ

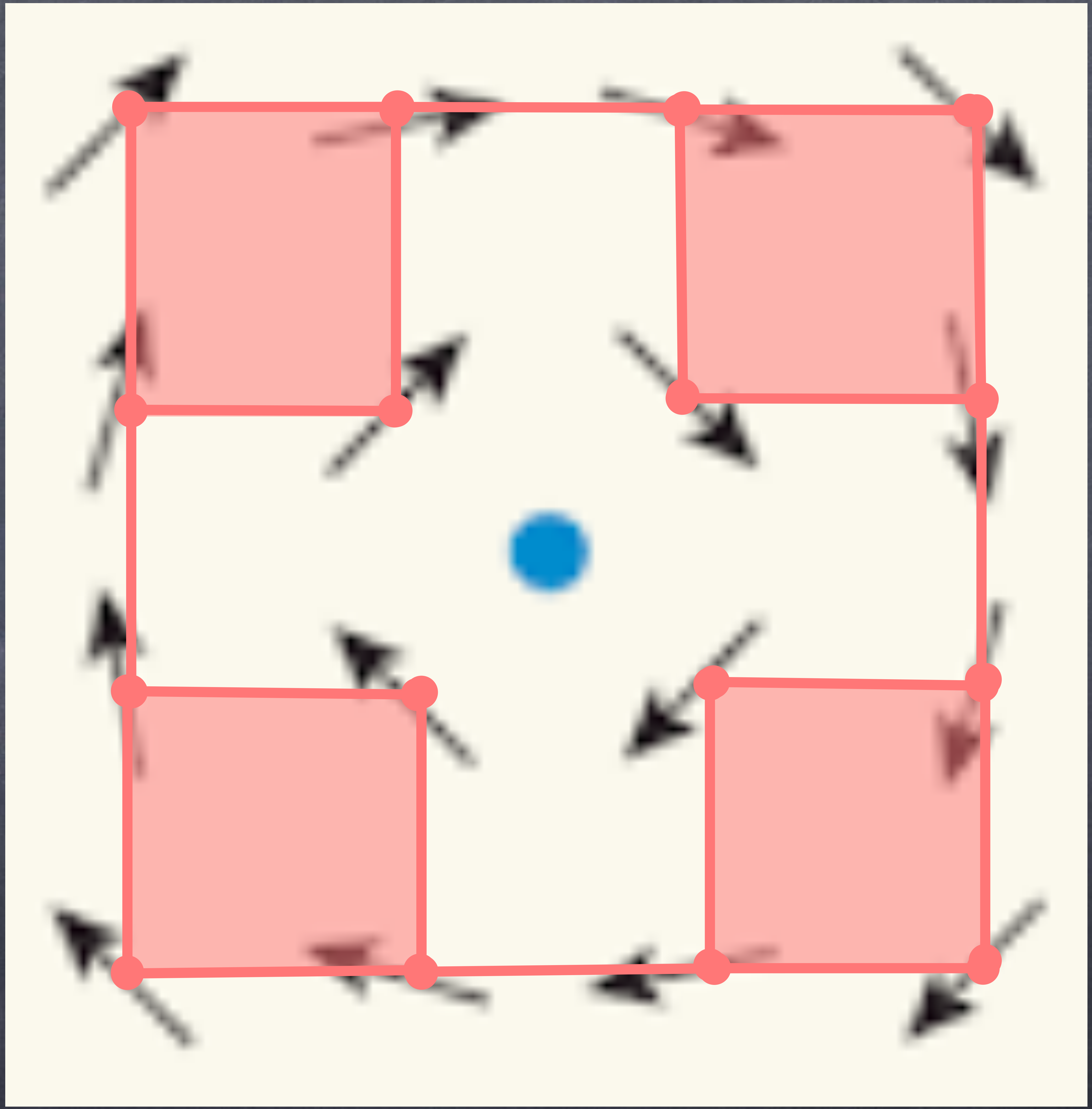
$\downarrow$

Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.

2. Fill in a plaquette if all the edges are present.



Configuration ϕ

↓

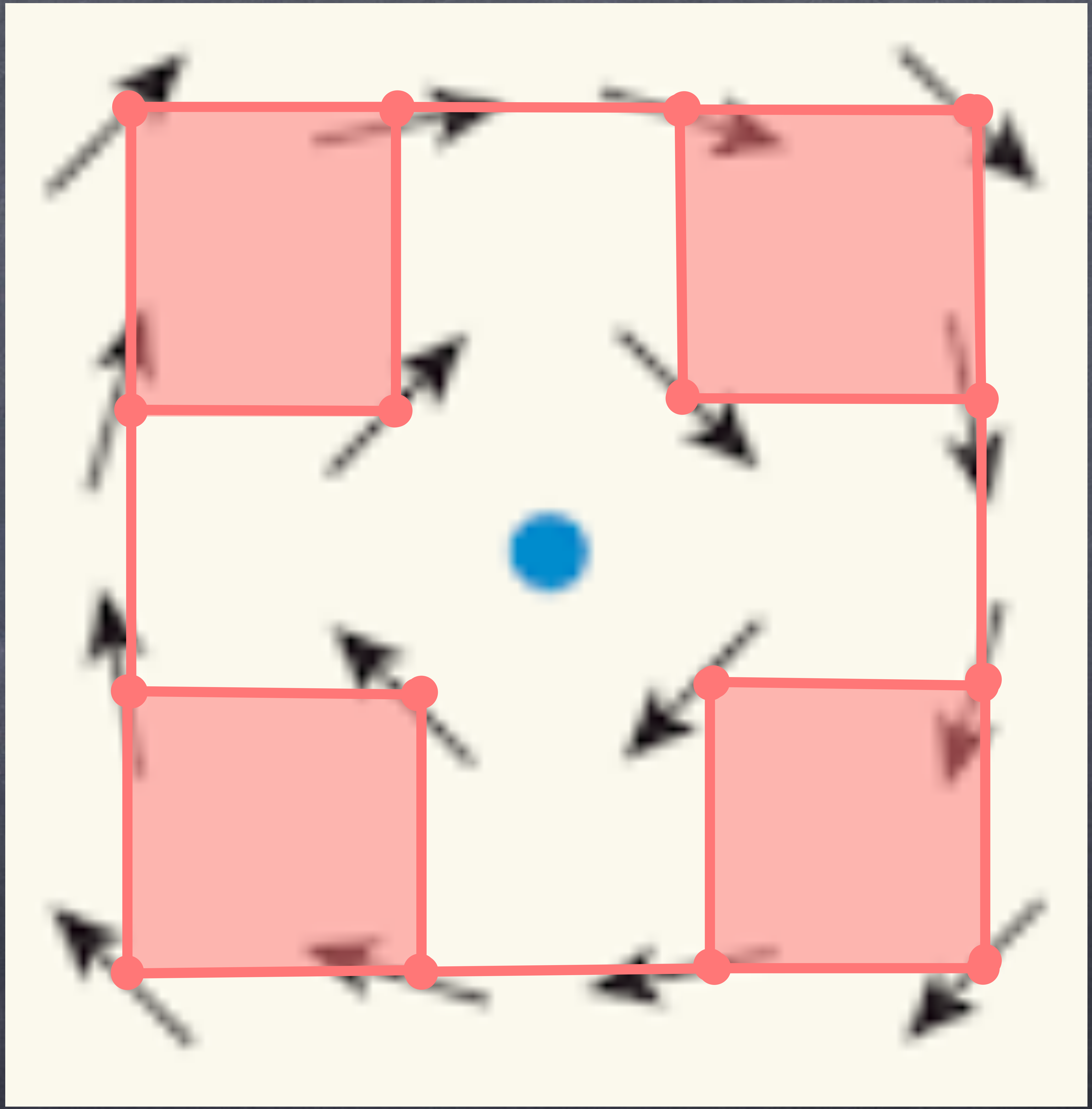
Geometric object $X(\phi)$

XY-model example

1. Fill in an edge if the spins are close.

threshold

2. Fill in a plaquette if all the edges are present.



A continuous map of geometric objects

$$X \to Y$$

induces a linear map $H_*(X) \to H_*(Y)$

A continuous map of geometric objects

$$X \to Y$$

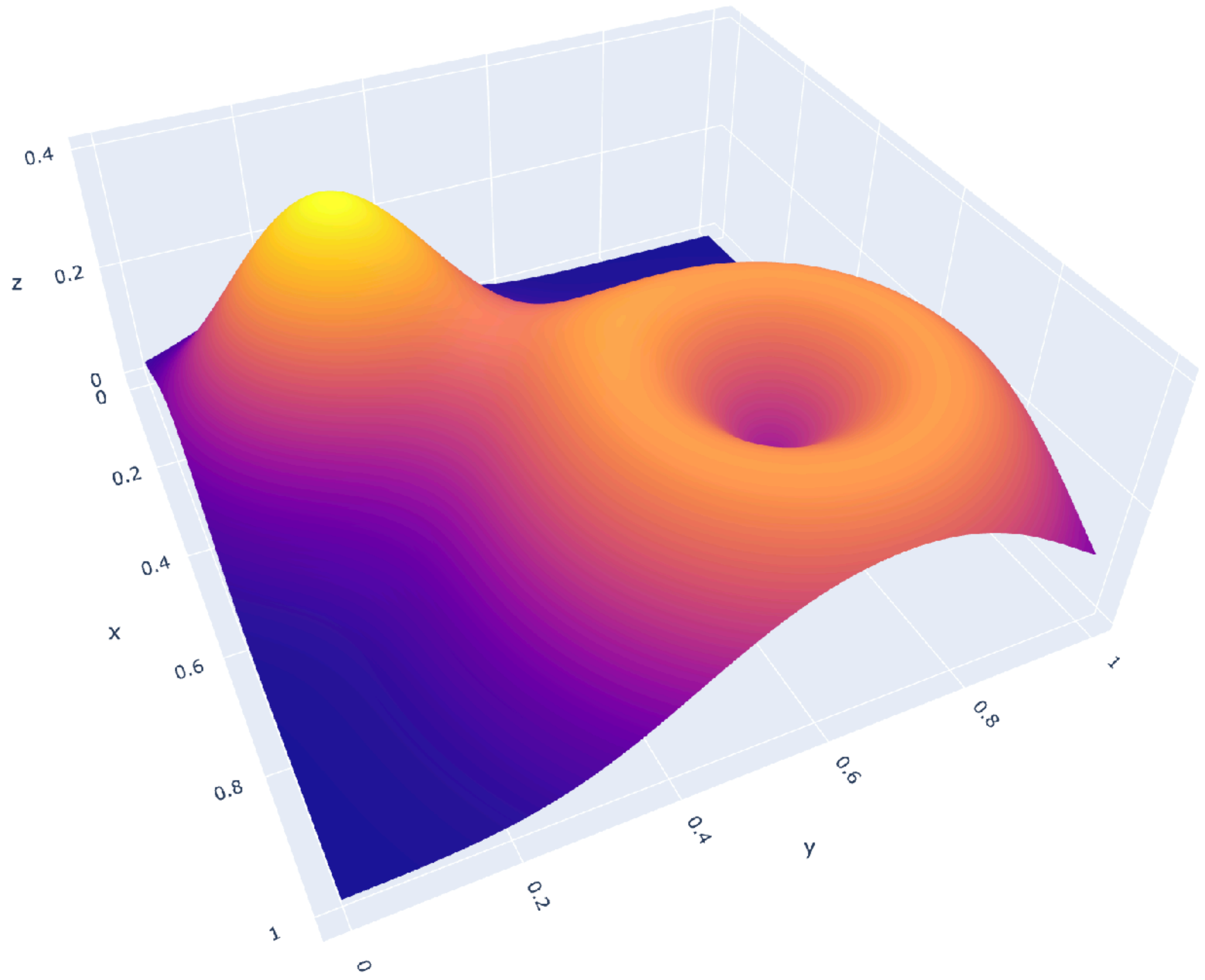
induces a linear map $H_*(X) \to H_*(Y)$

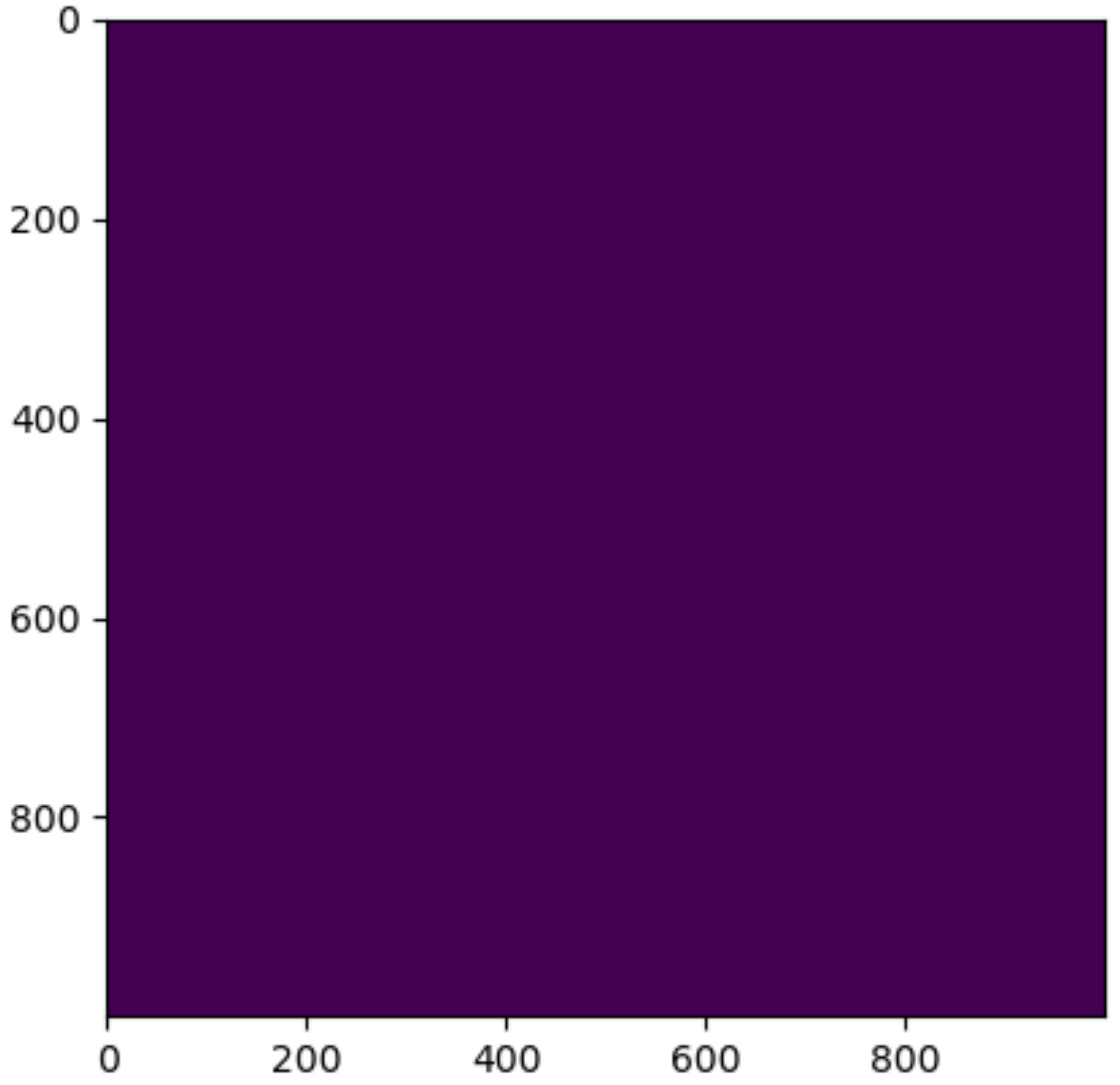
A sequence of geometric objects

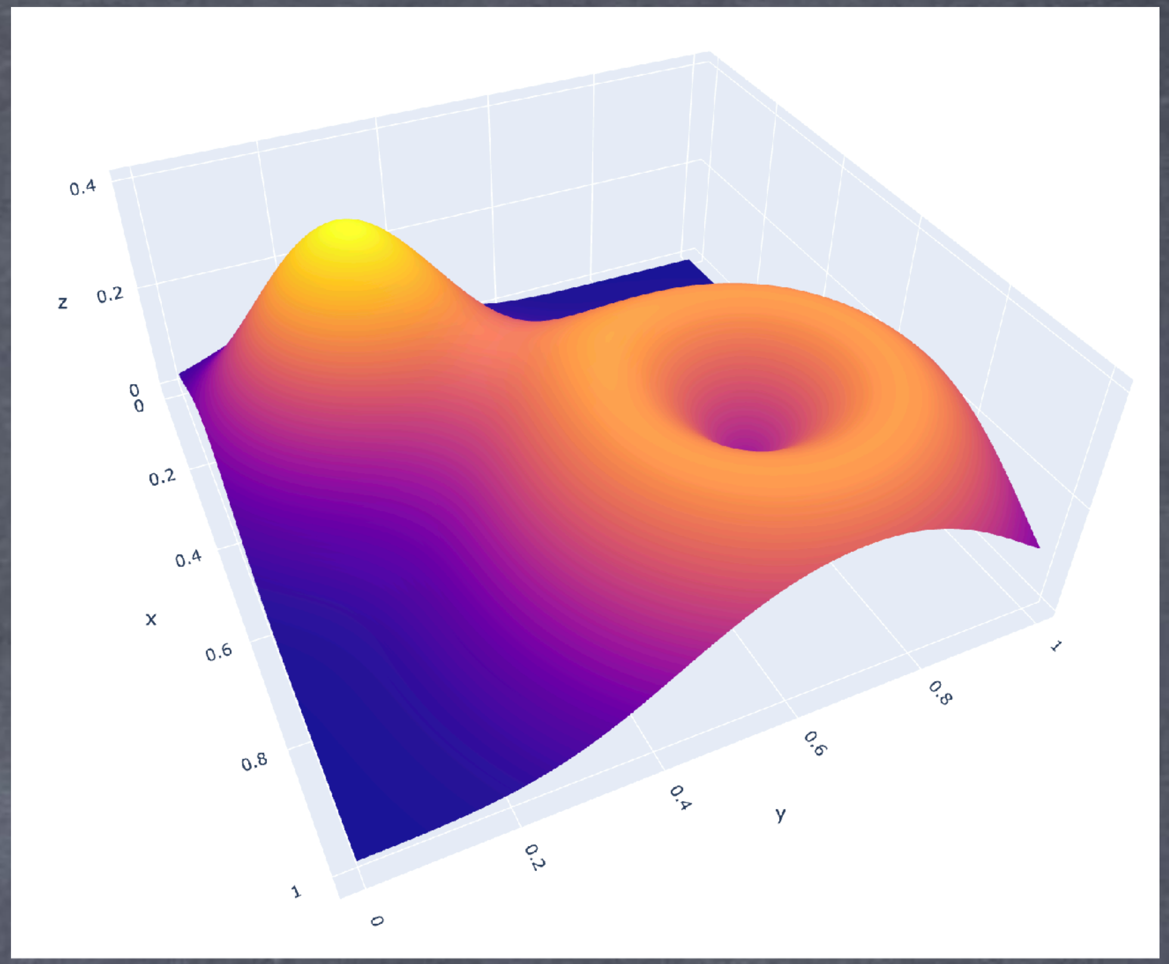
$$X_1 \to X_2 \to \cdots$$

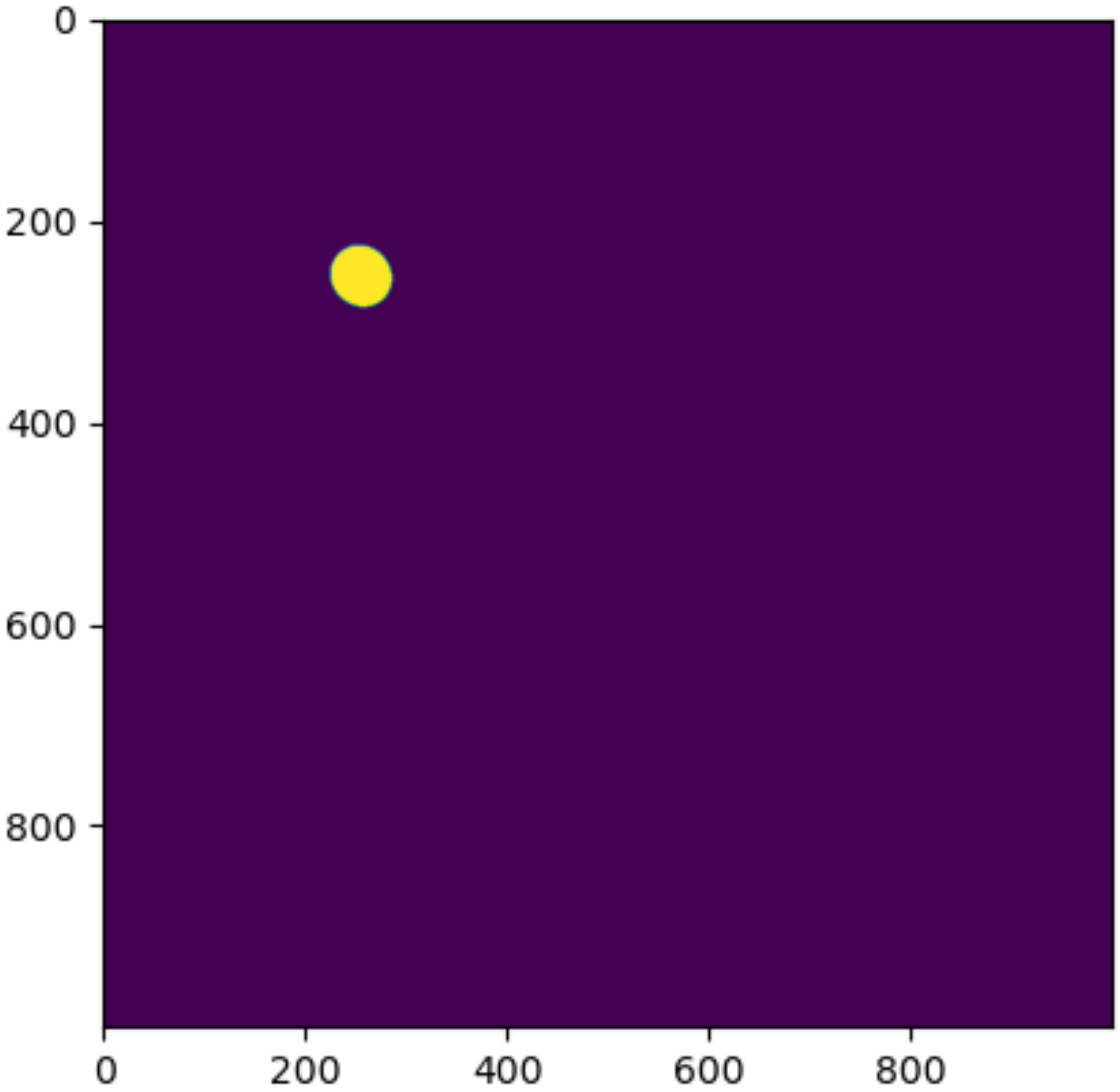
induces a sequence of vector spaces and linear maps

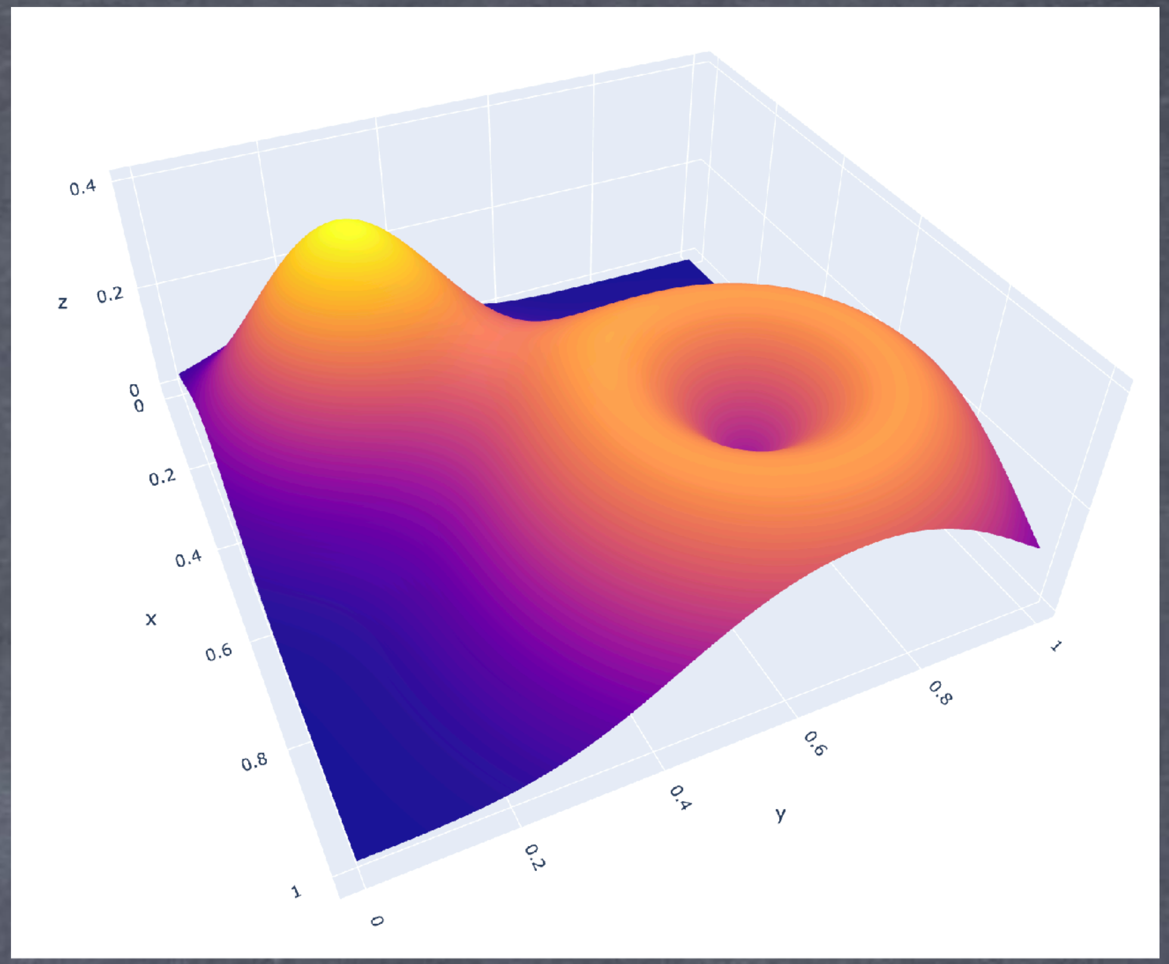
$$H_*(X_1) \to H_*(X_2) \to \cdots$$

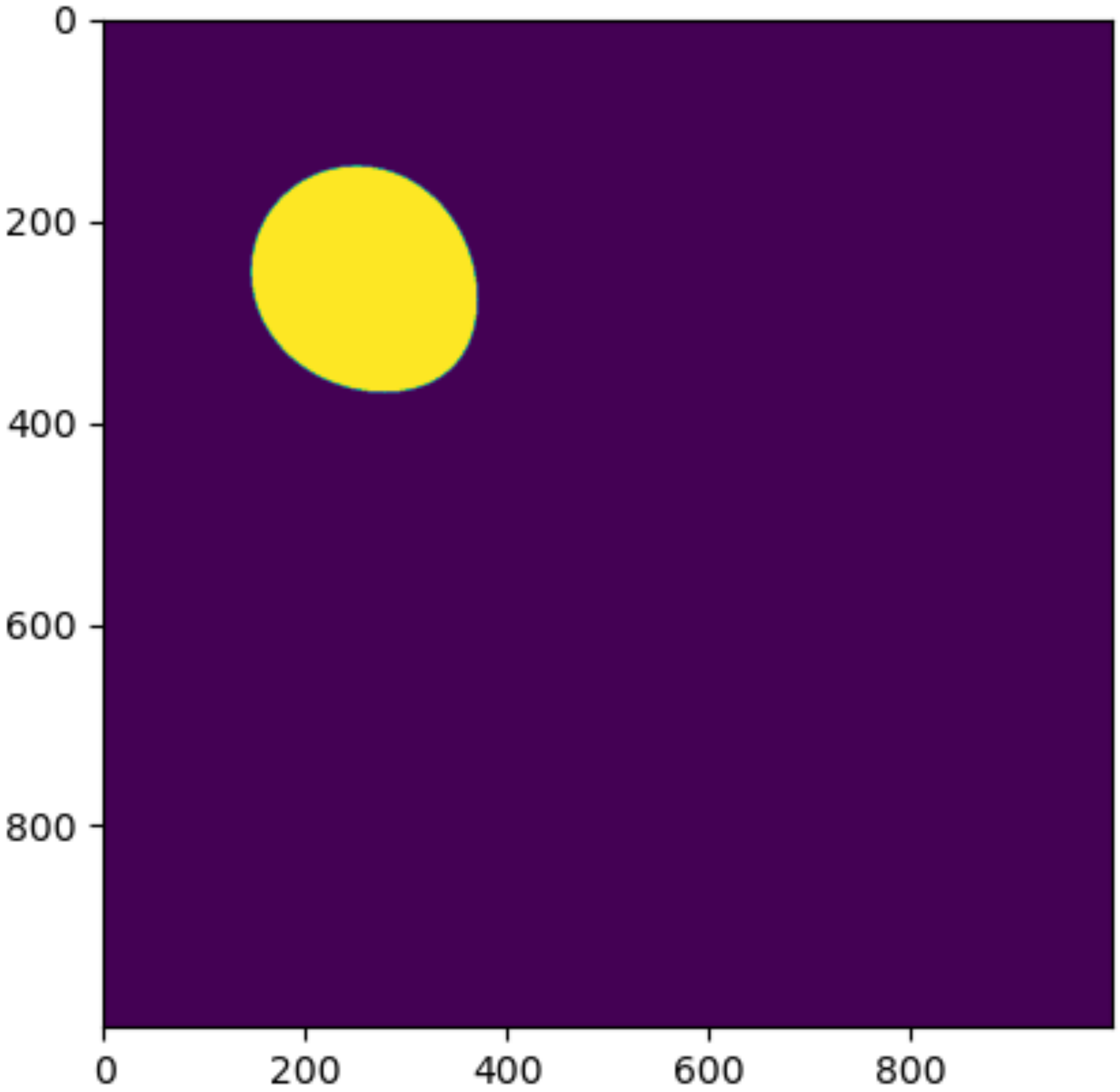

0.4
0.2
z
0
0
0.2
0.4
x
0.6
0.8
1
0
0.2
0.4
y
0.6
0.8
1

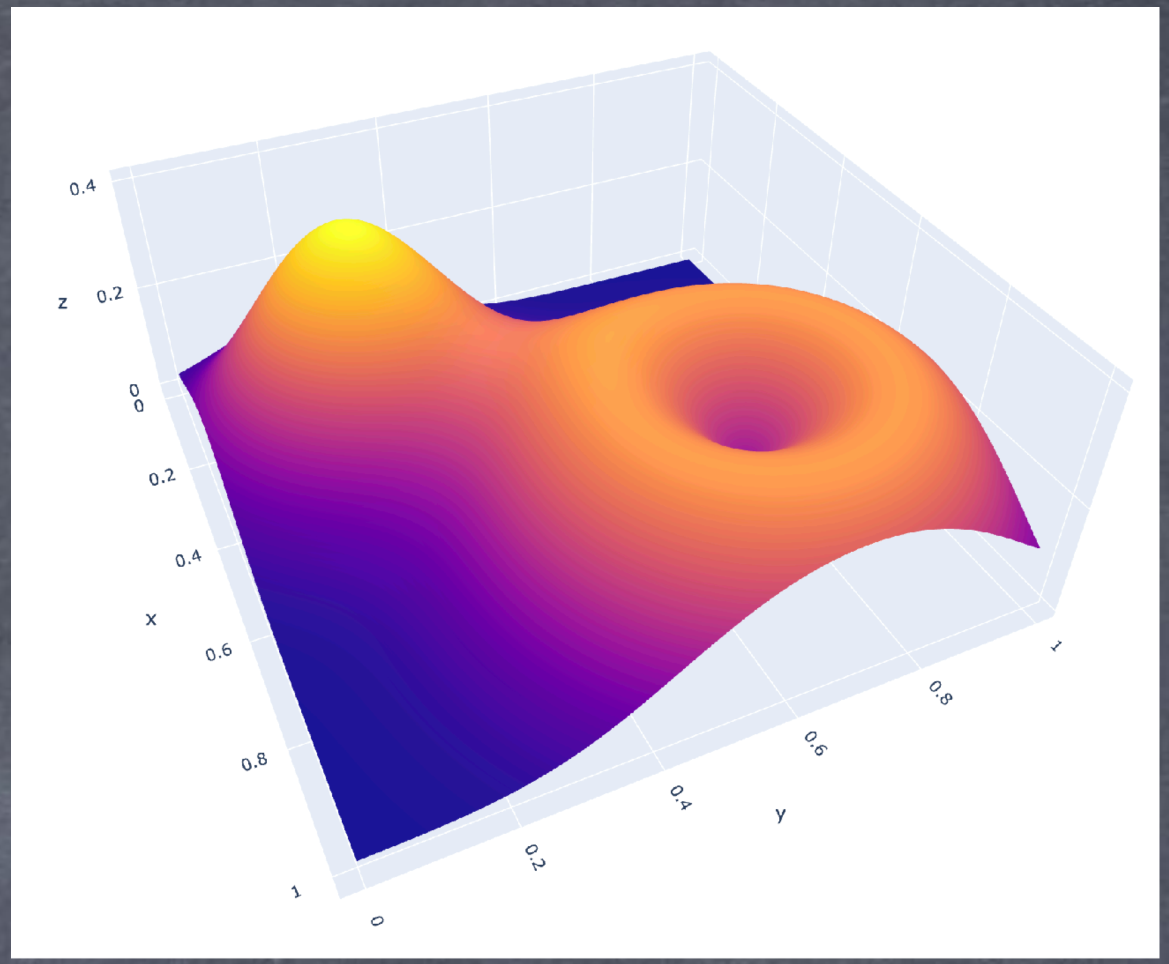

0
200
400
600
800
0
200
400
600
800

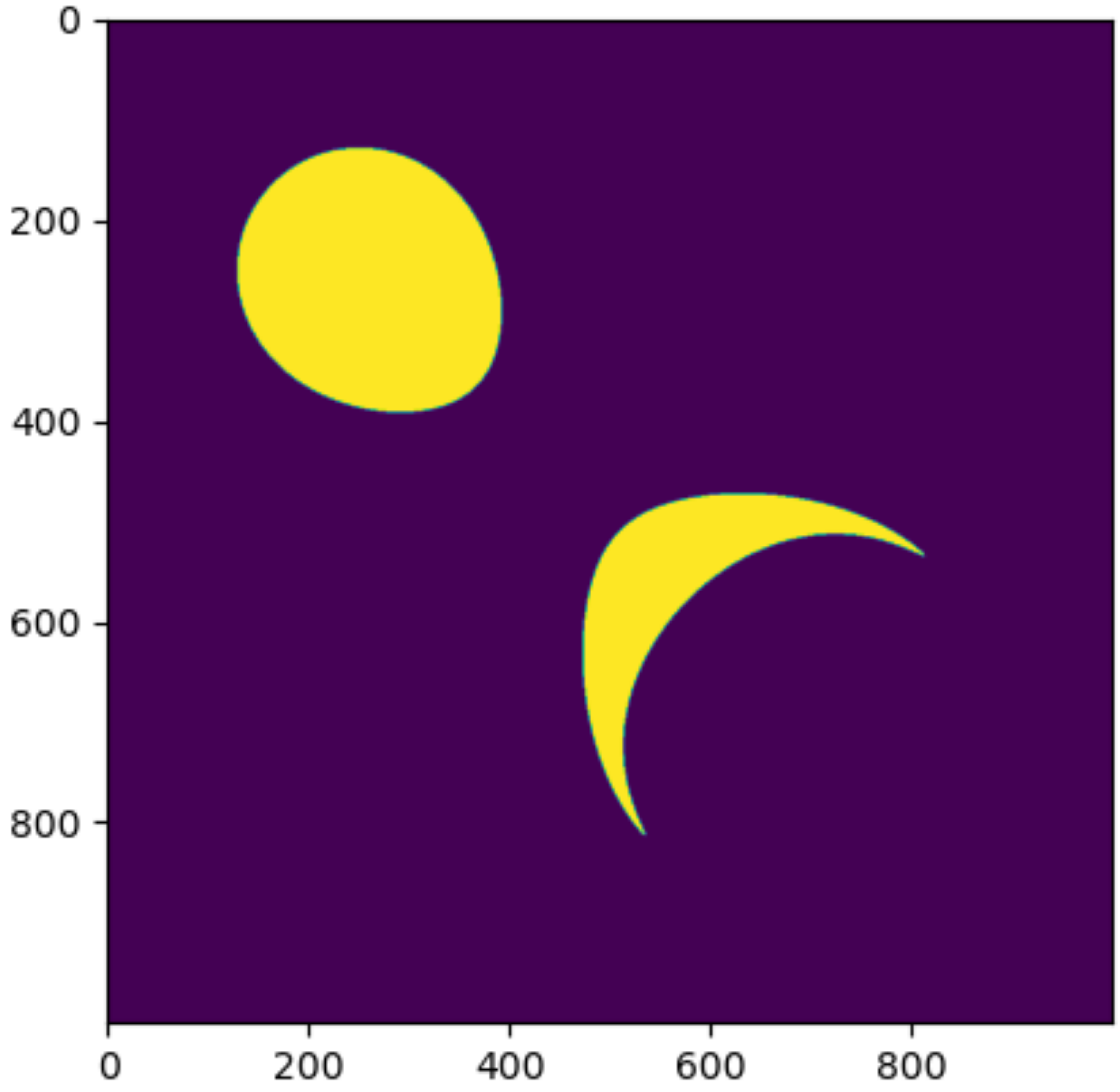



0
200
400
600
800
0
200
400
600
800

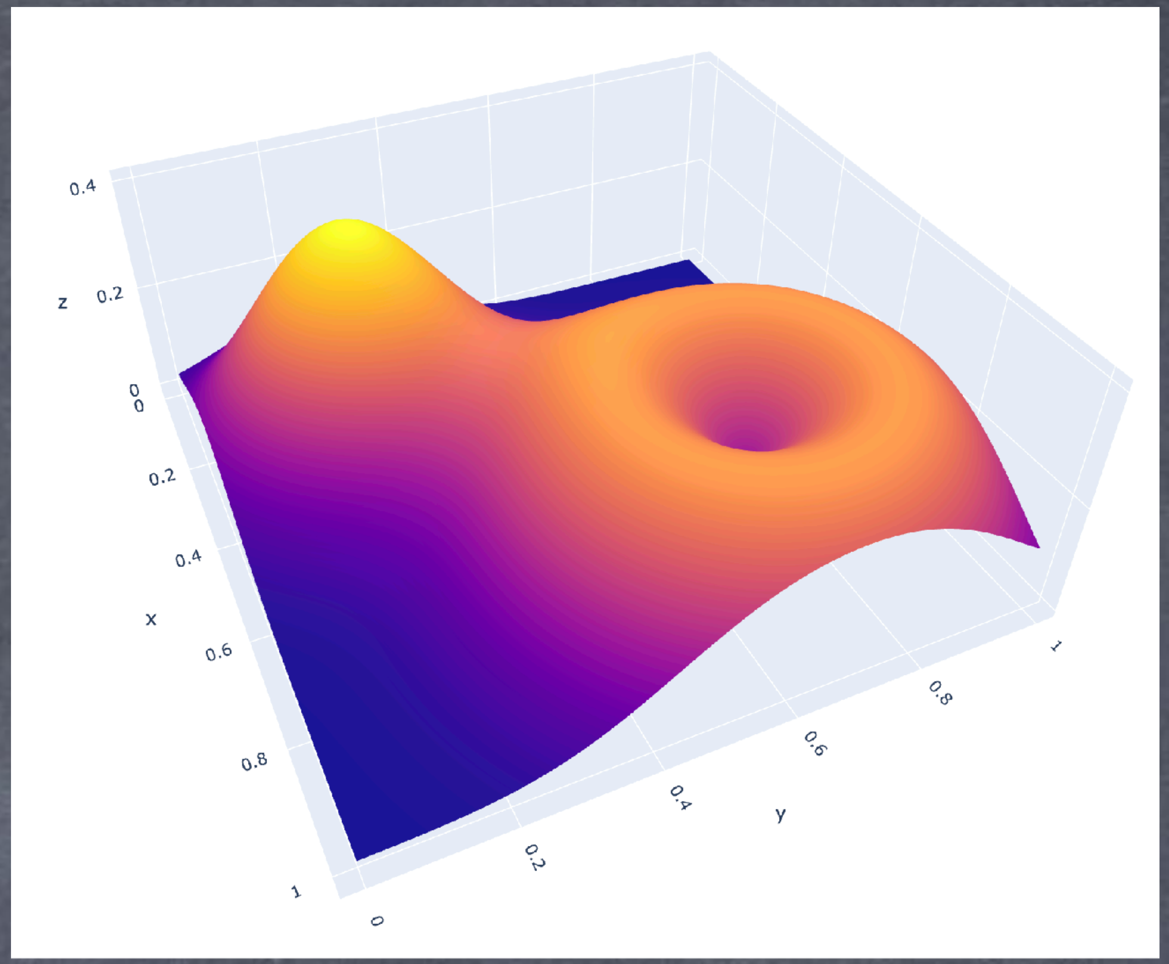


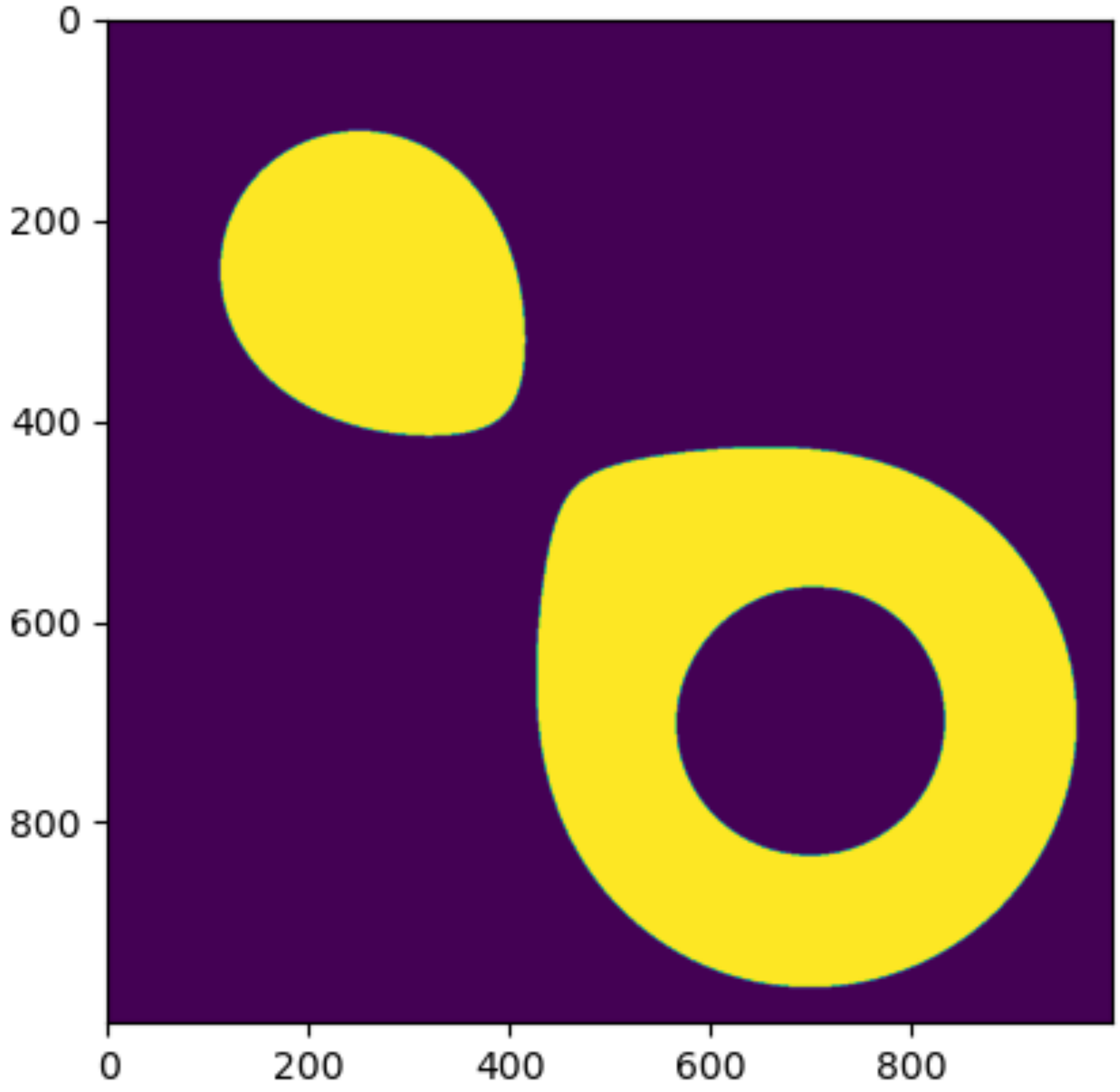

0
200
400
600
800
0
200
400
600
800

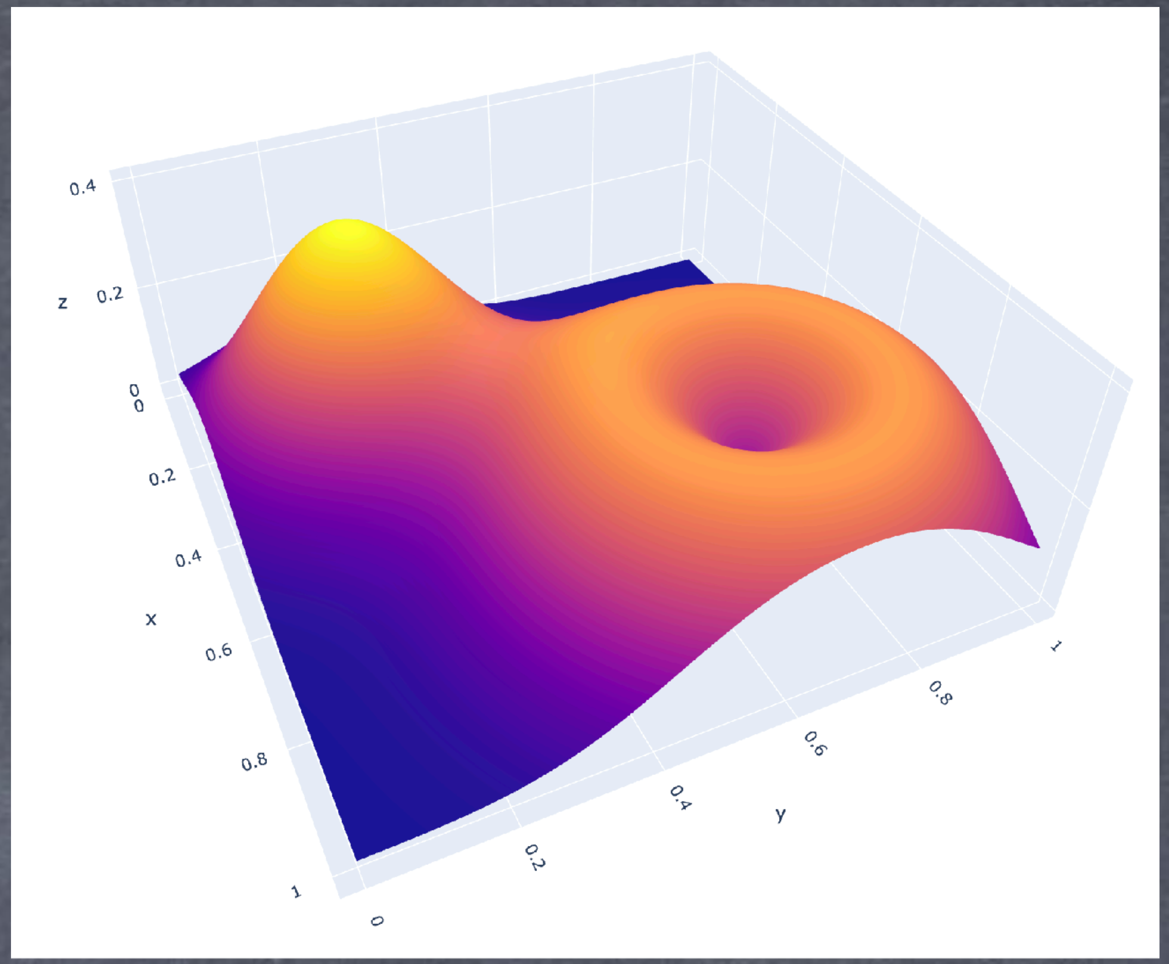


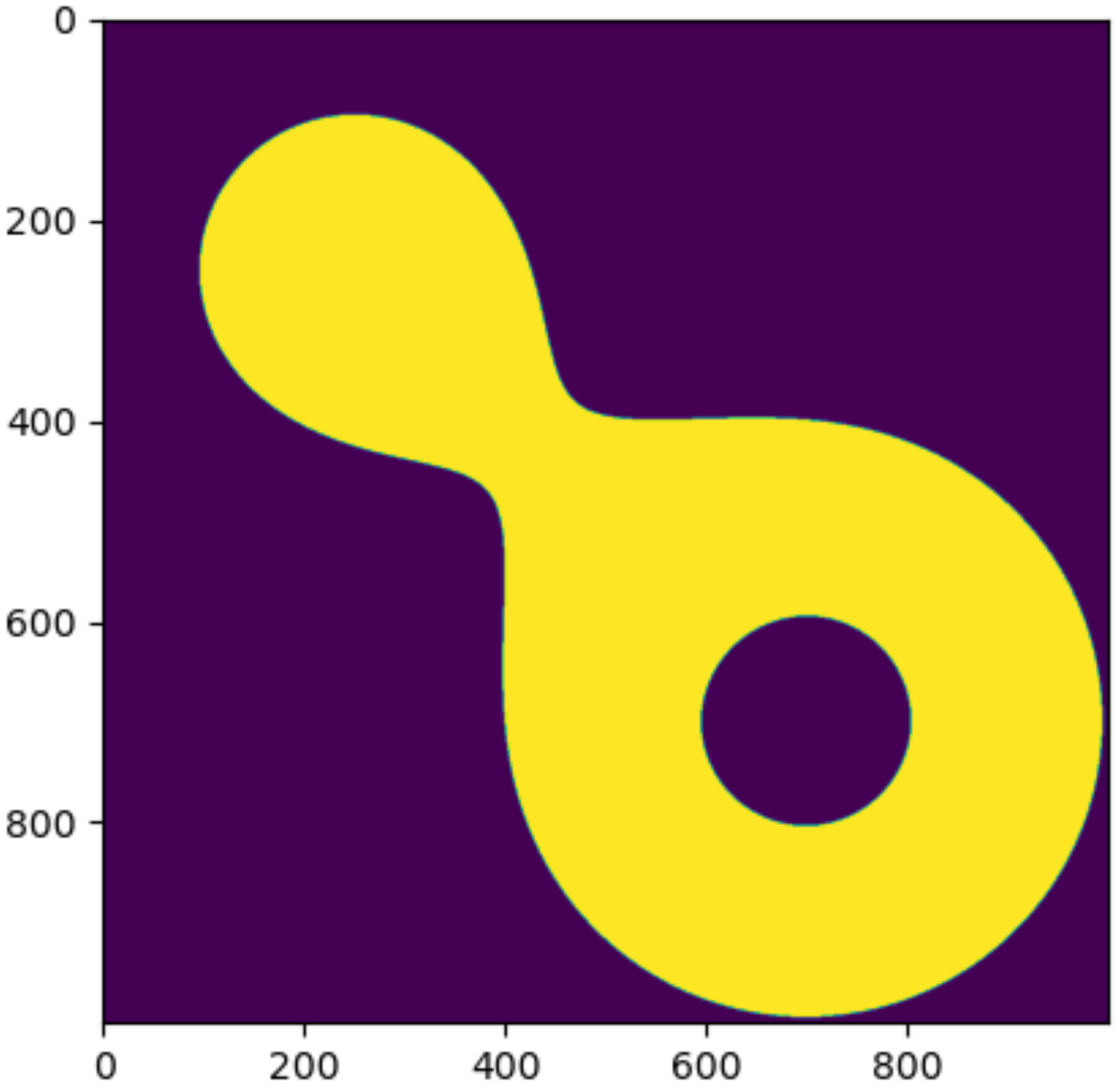


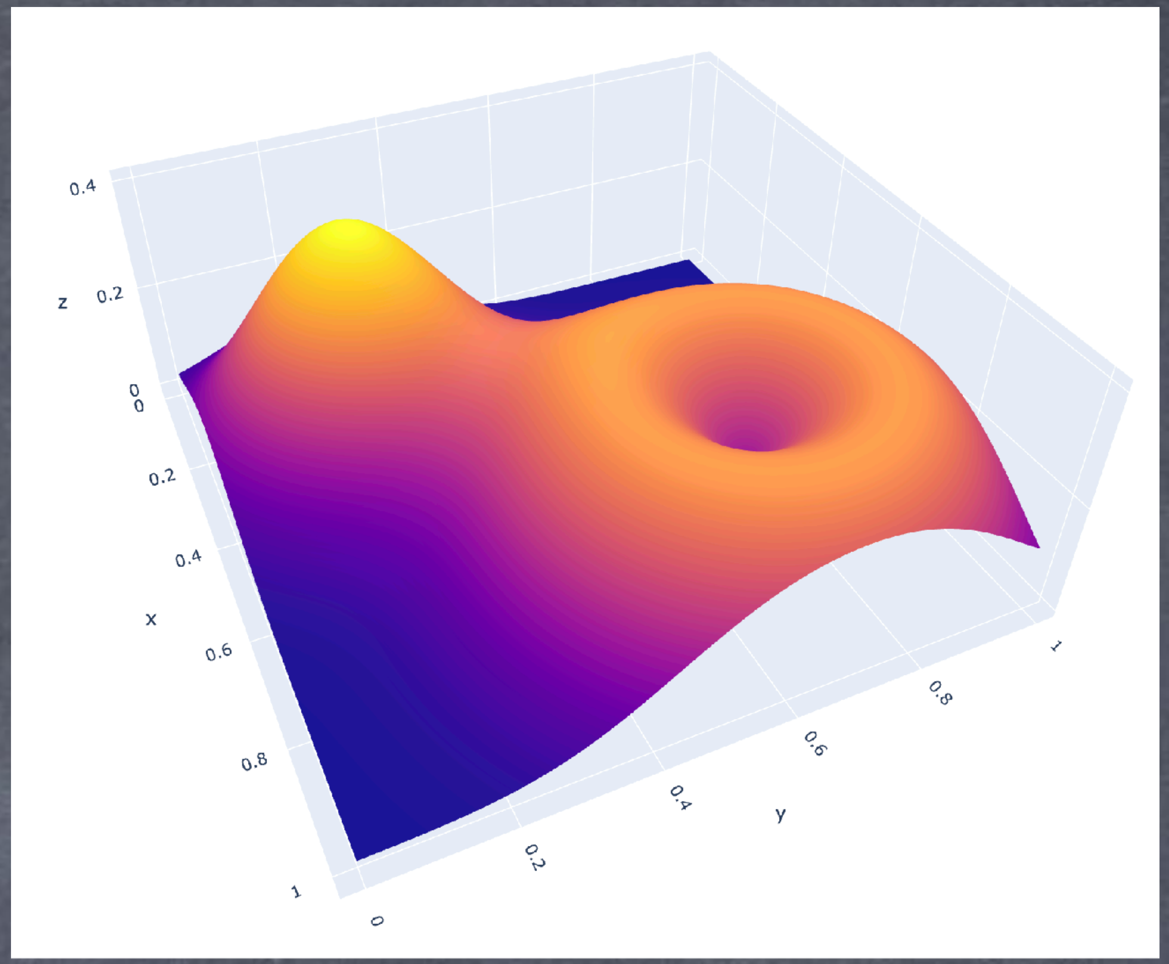
0
200
400
600
800
0
200
400
600
800

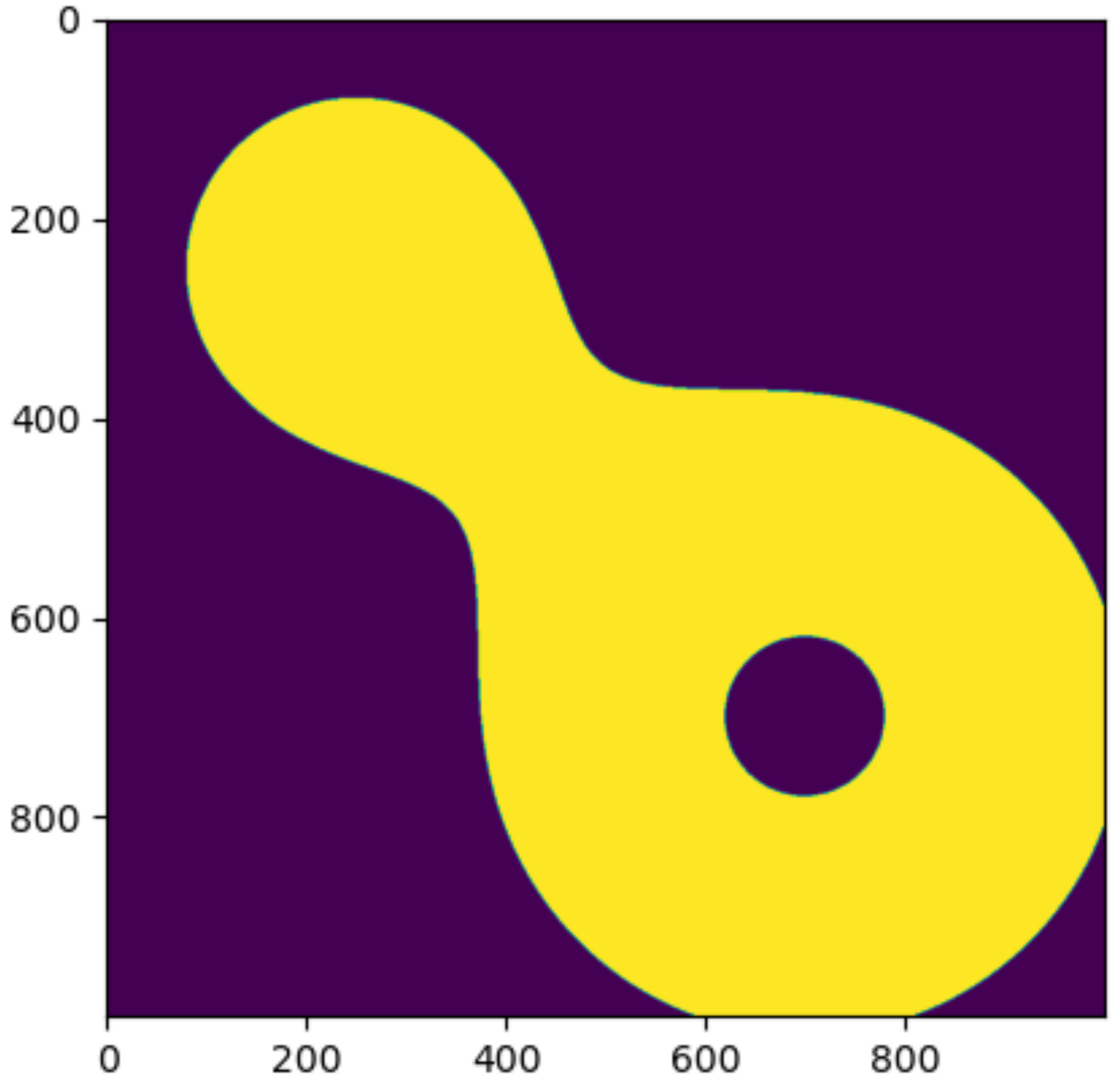



0
200
400
600
800
0
200
400
600
800

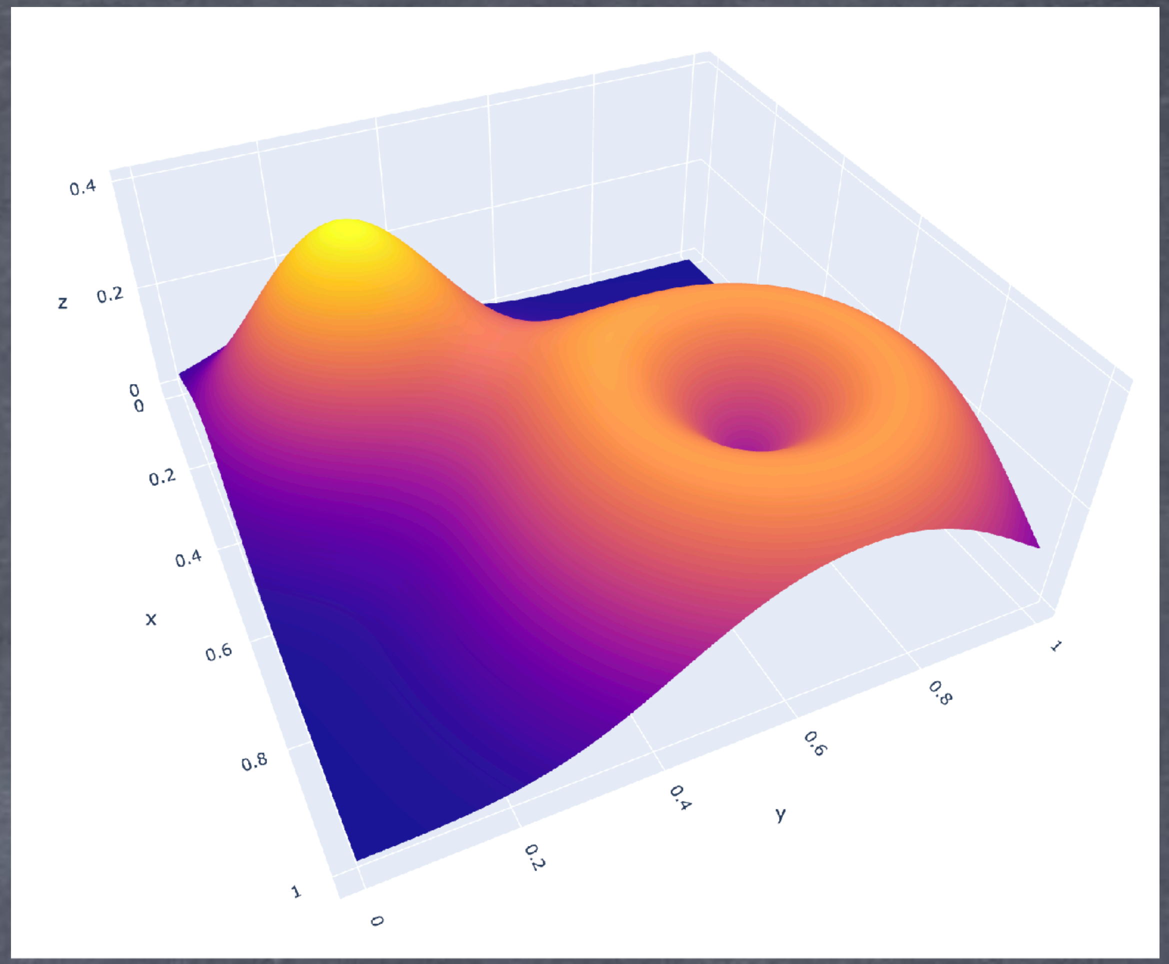



0
200
400
600
800
0
200
400
600
800

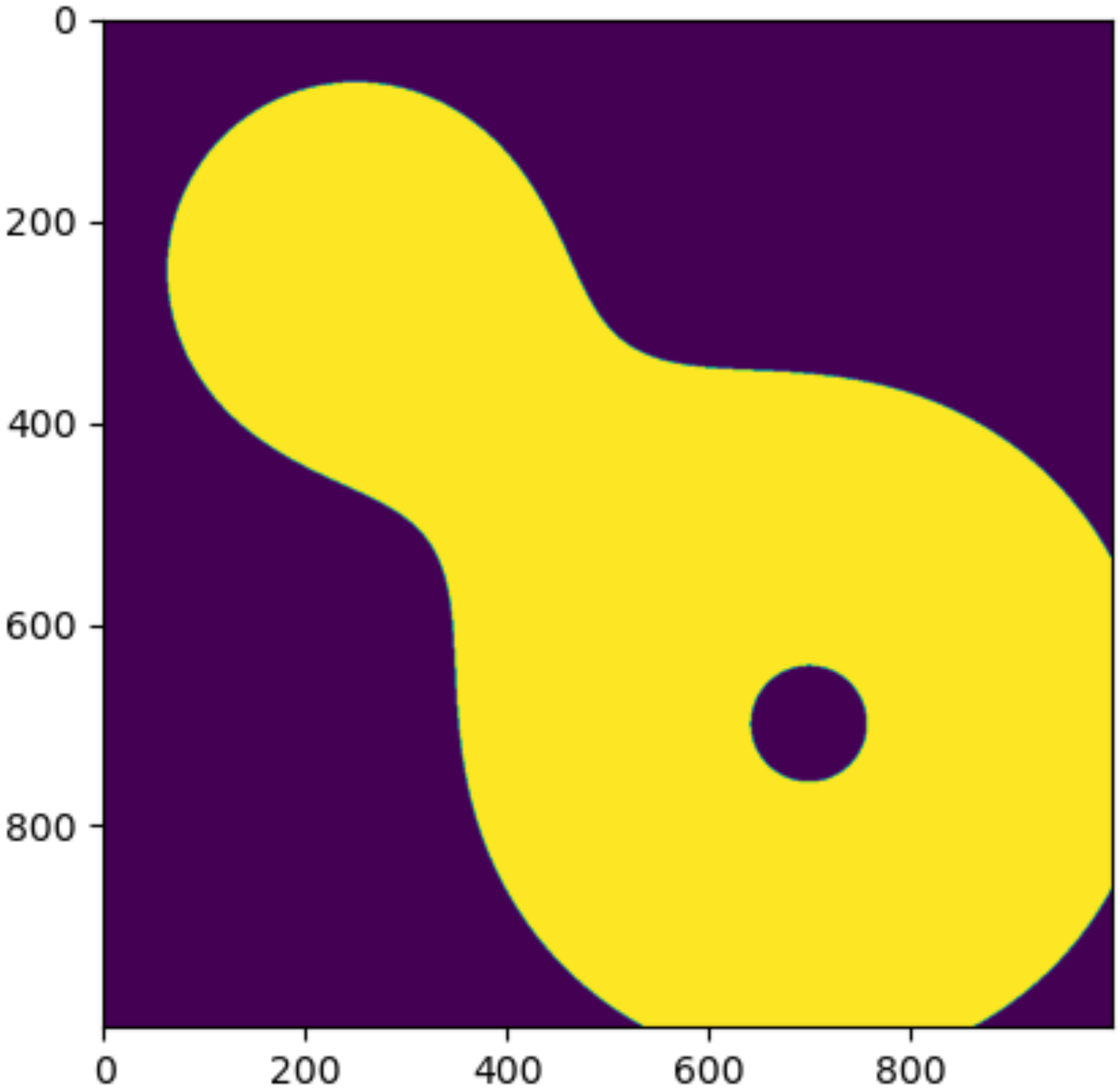


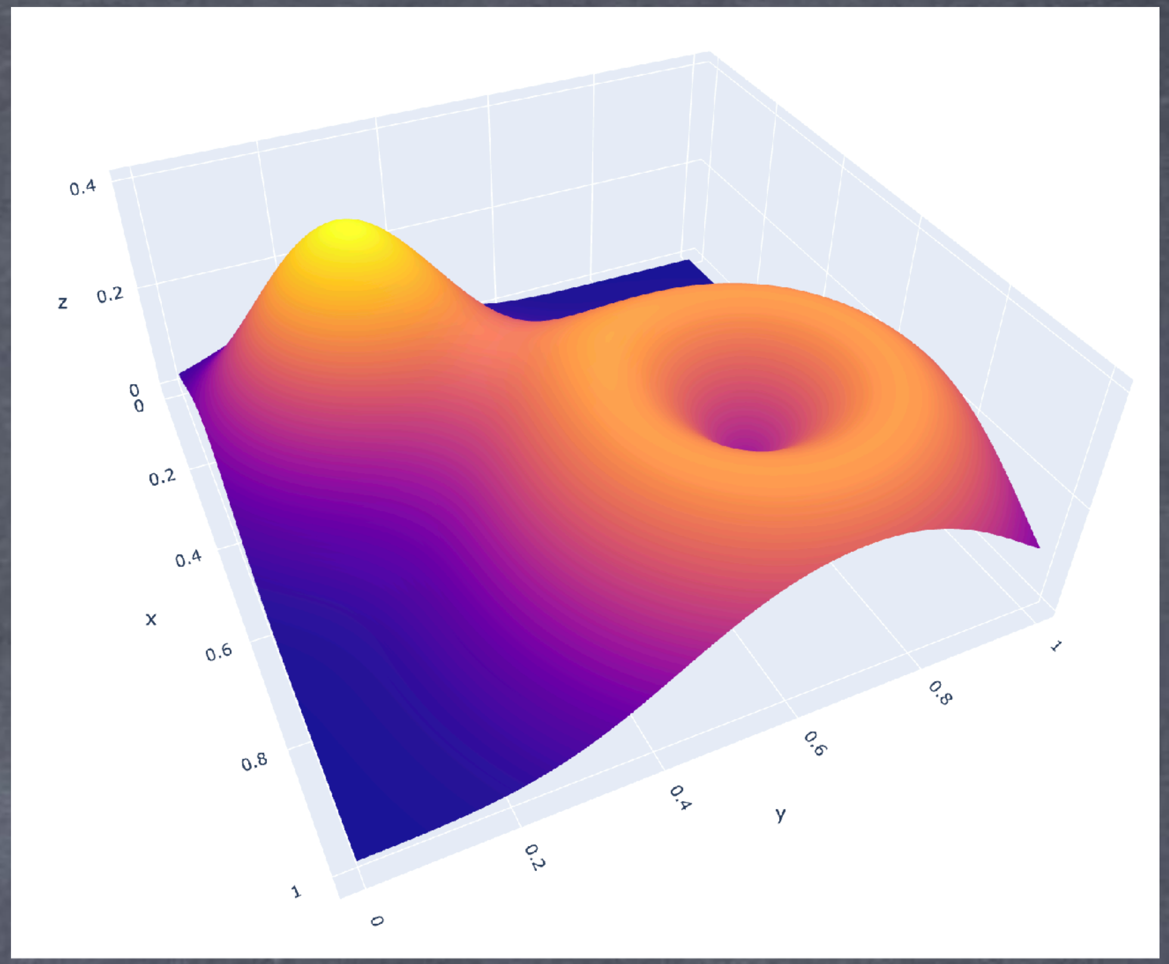


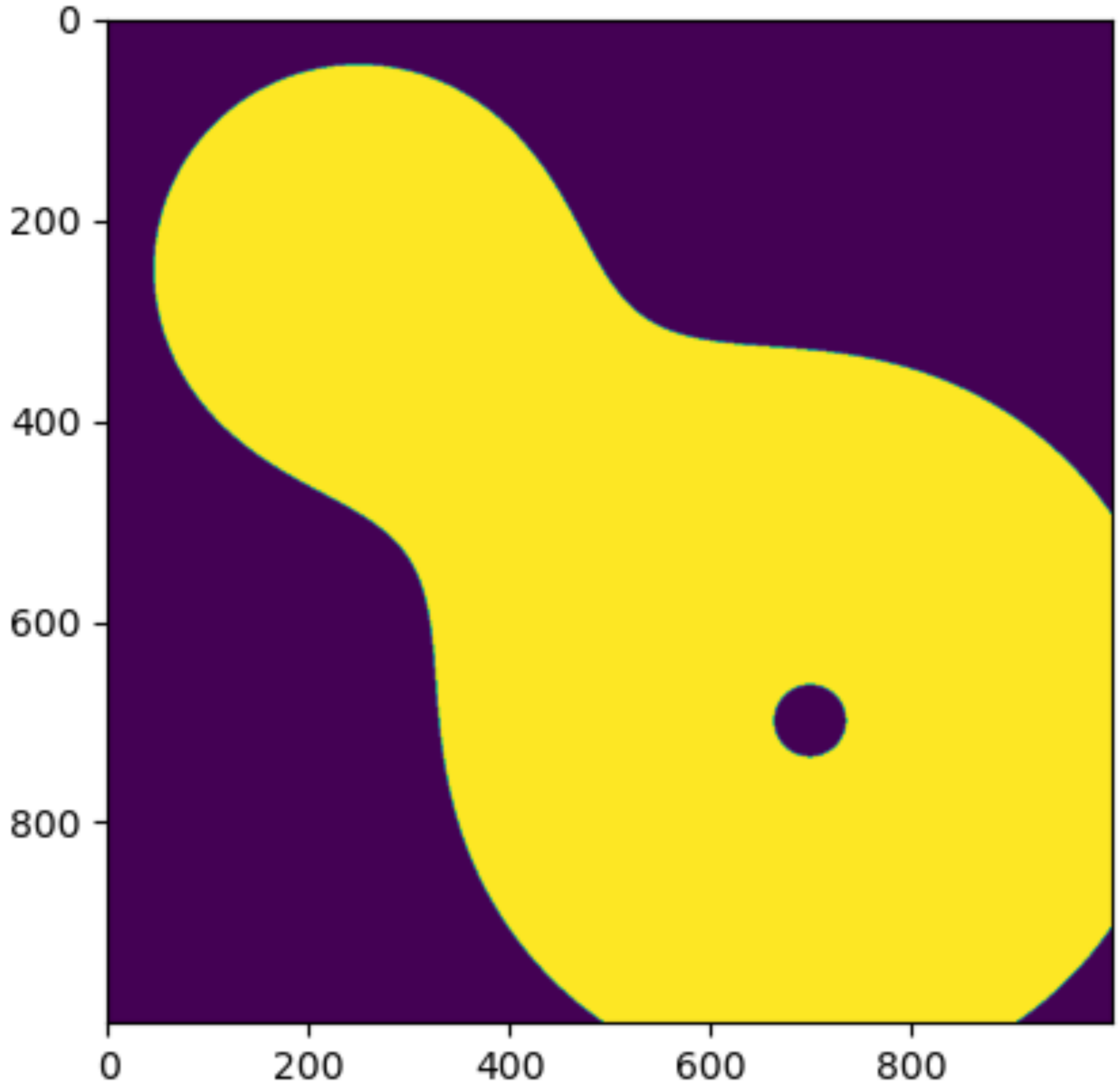
0
200
400
600
800
0
200
400
600
800

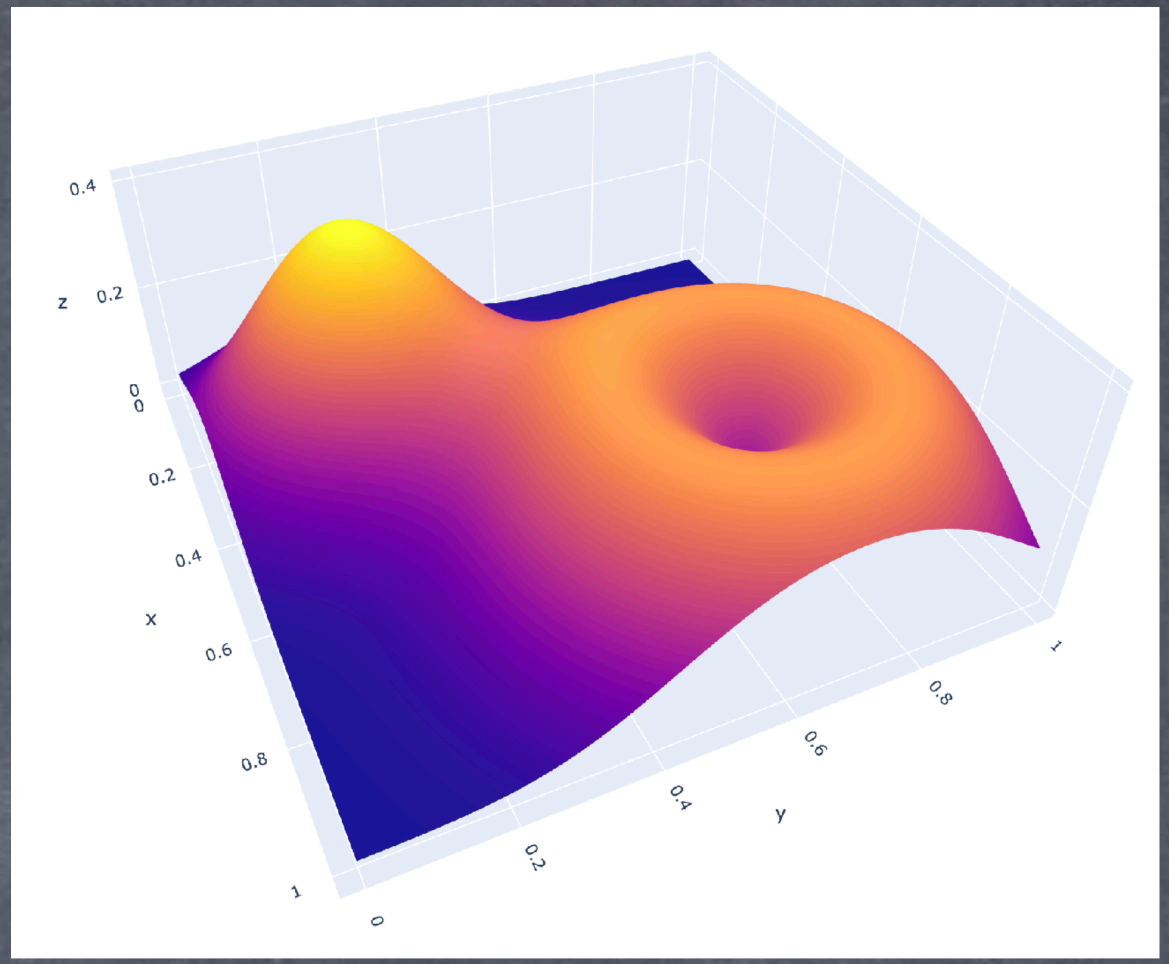


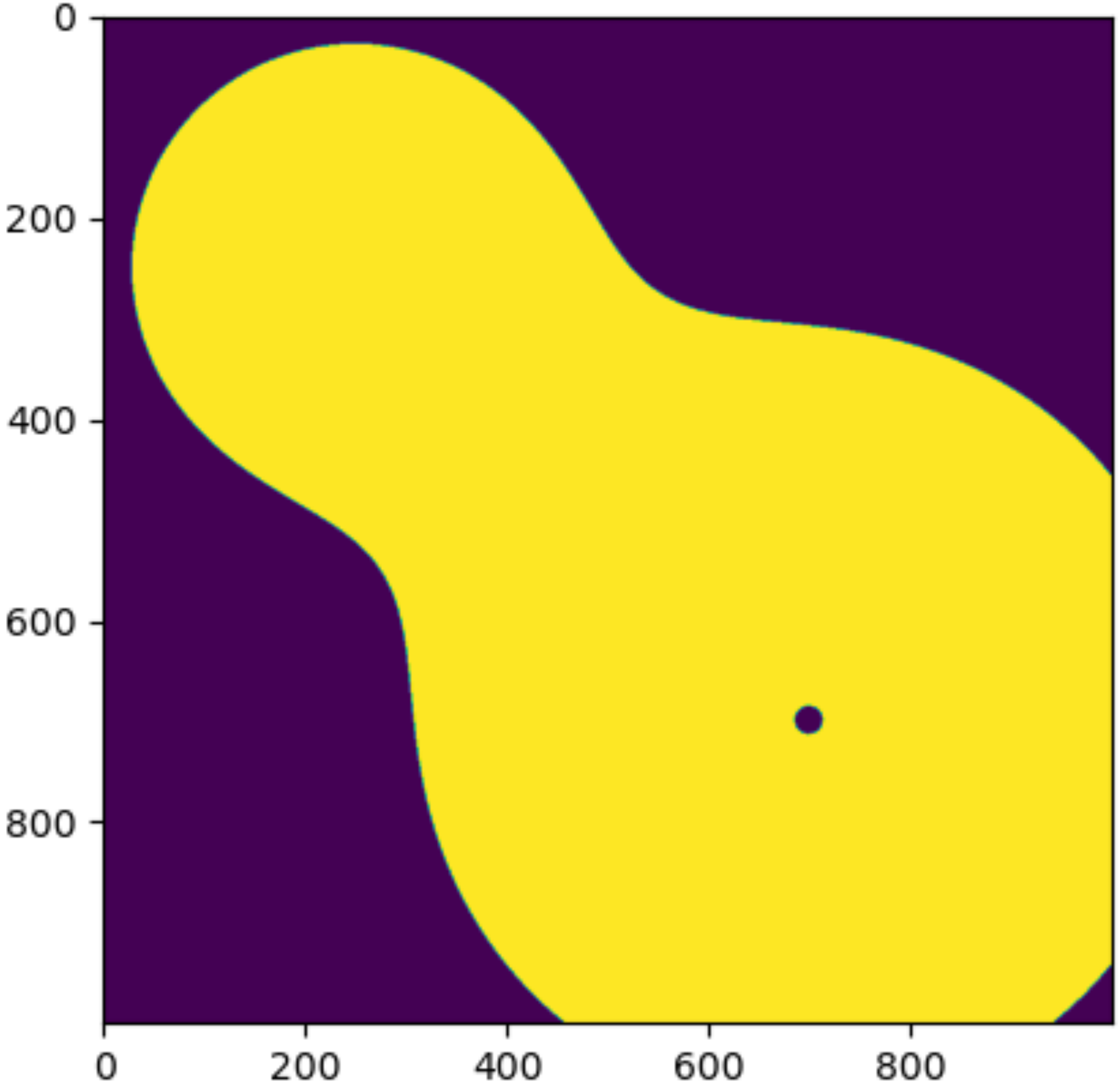
z
x
y

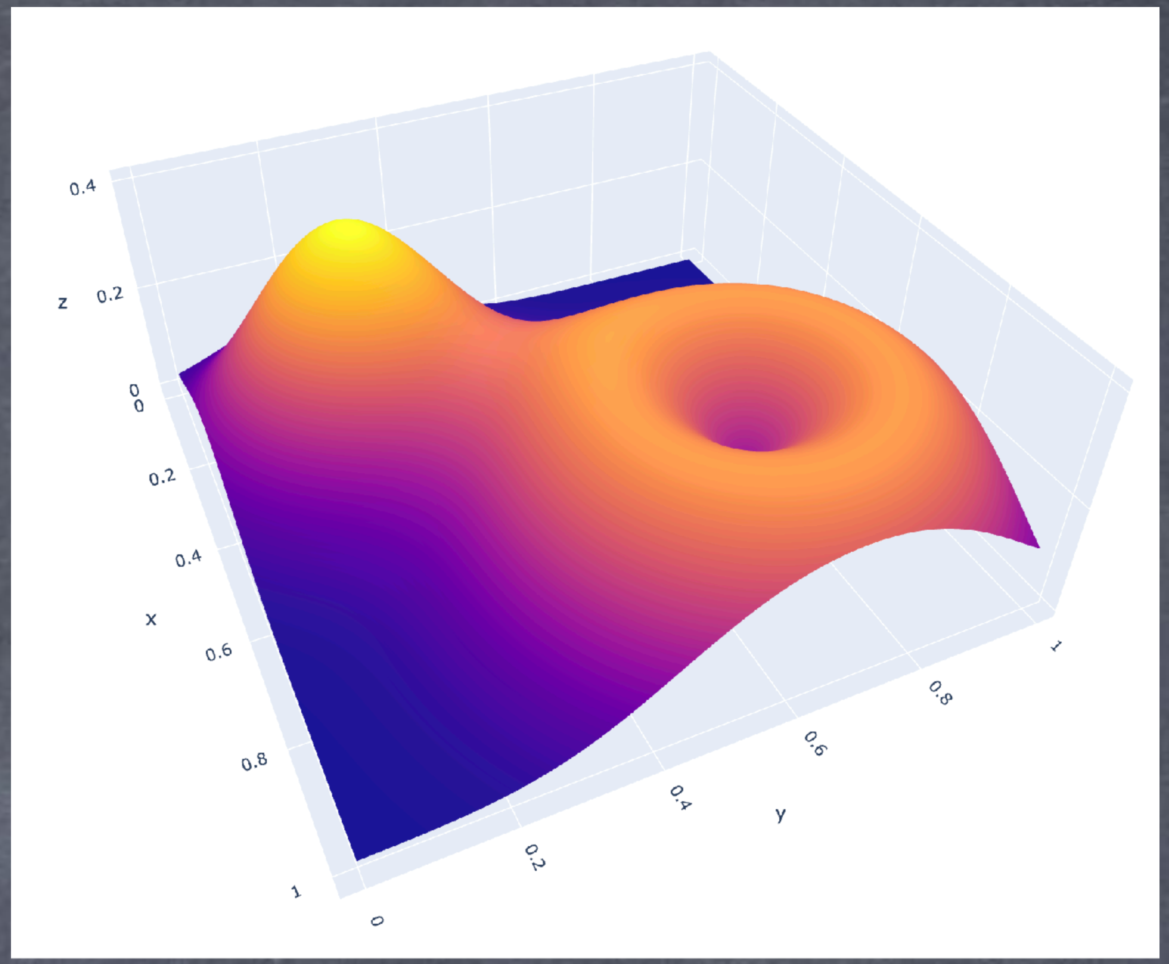

0
200
400
600
800
0
200
400
600
800

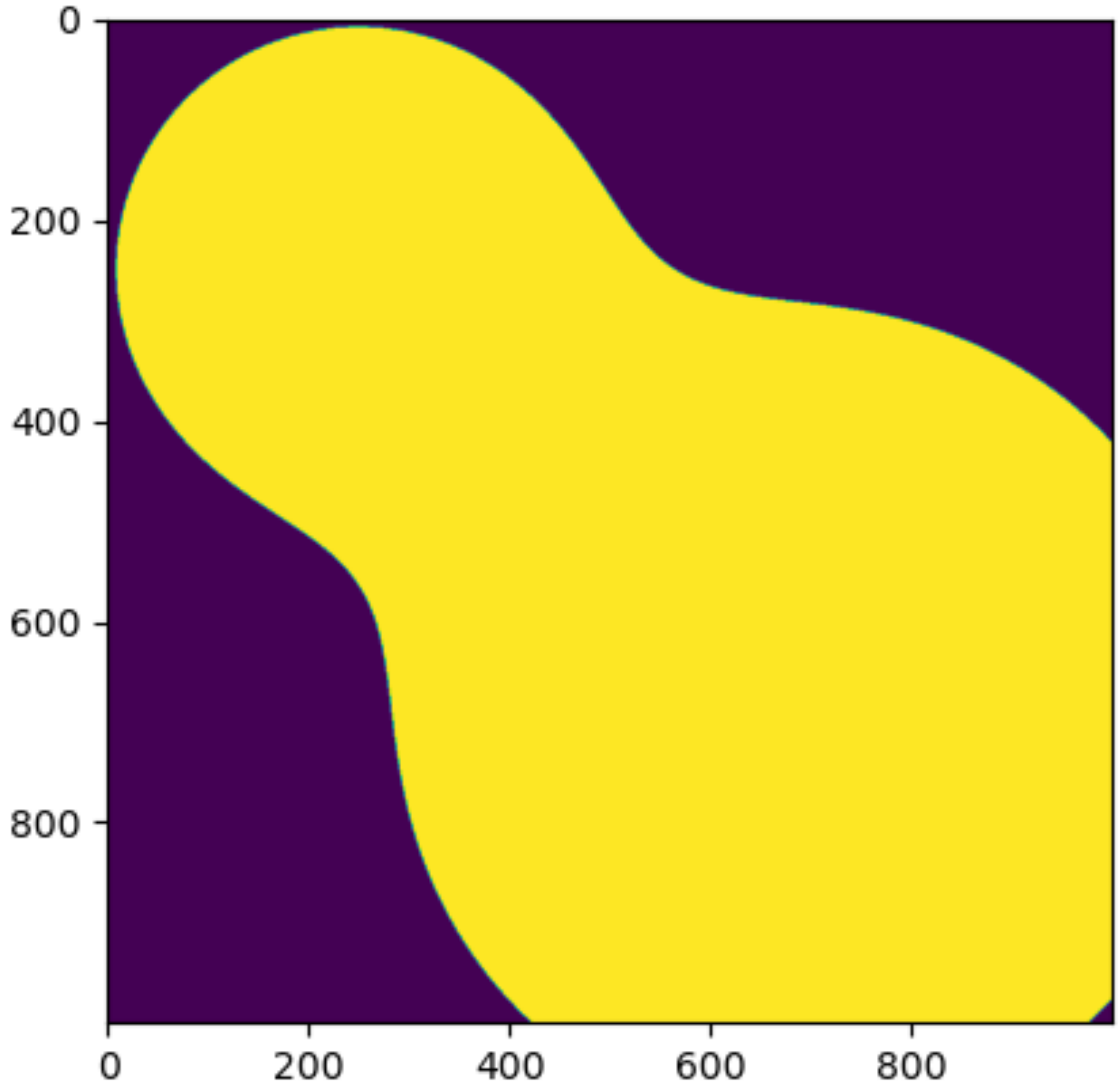



0
200
400
600
800
0
200
400
600
800

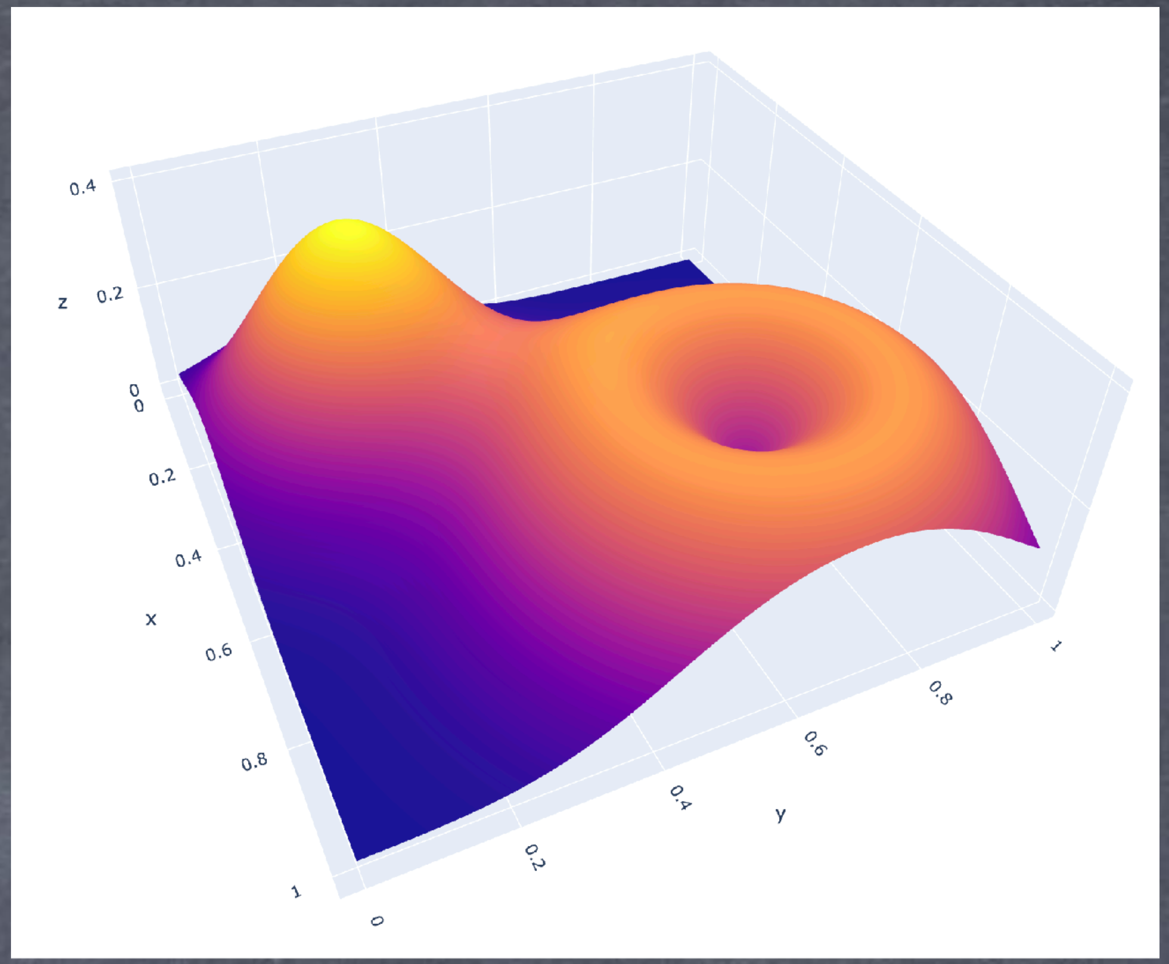



0
200
400
600
800
0
200
400
600
800





0
200
400
600
800
0
200
400
600
800



z
x
y

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

One can choose bases
compatible with the linear maps

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

One can choose bases compatible with the linear maps

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

One can choose bases compatible with the linear maps

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

One can choose bases compatible with the linear maps

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

Death time

One can choose bases compatible with the linear maps

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

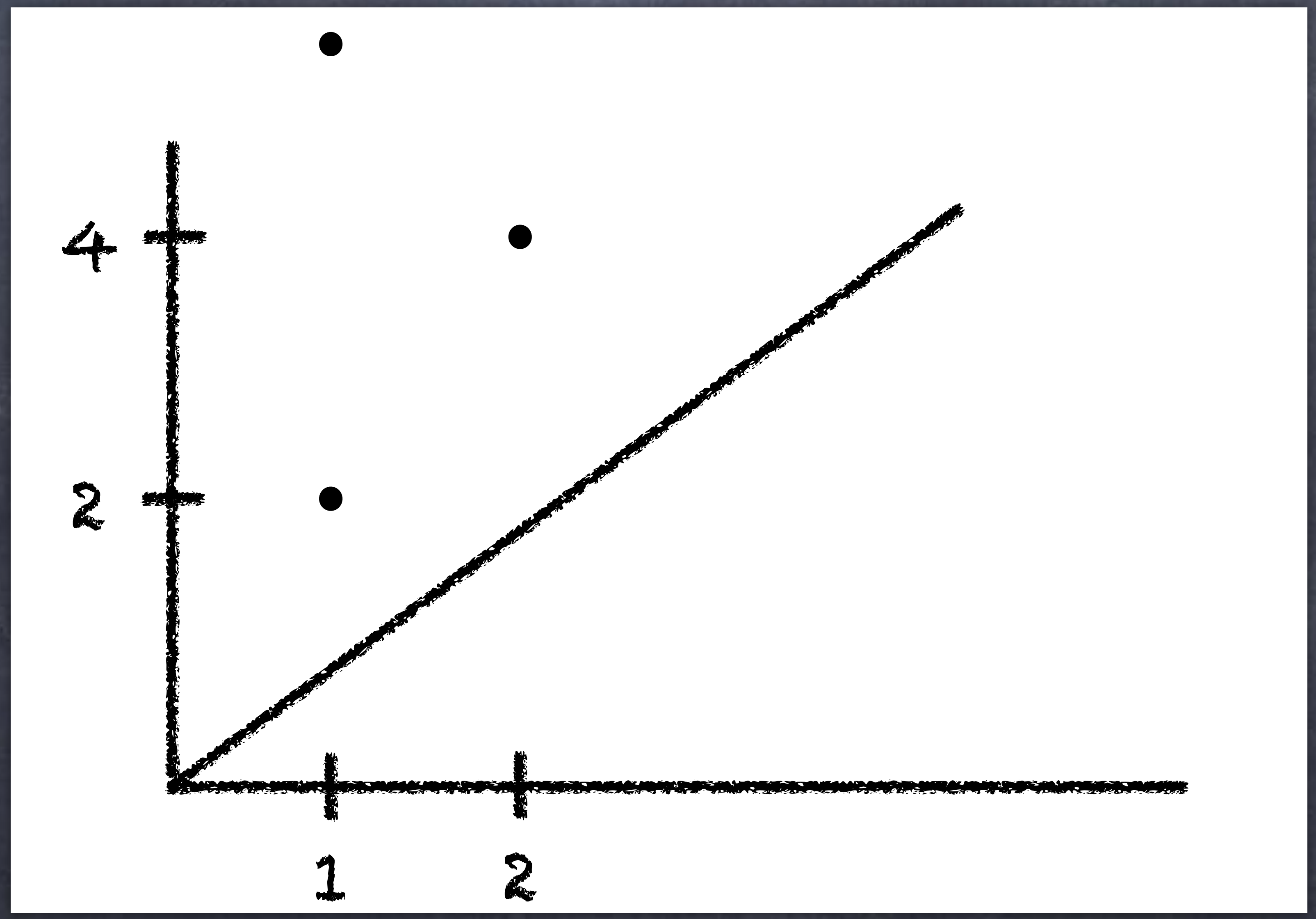
One can choose bases compatible with the linear maps

$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

Birth time

Death time

One can choose bases
compatible with the linear maps

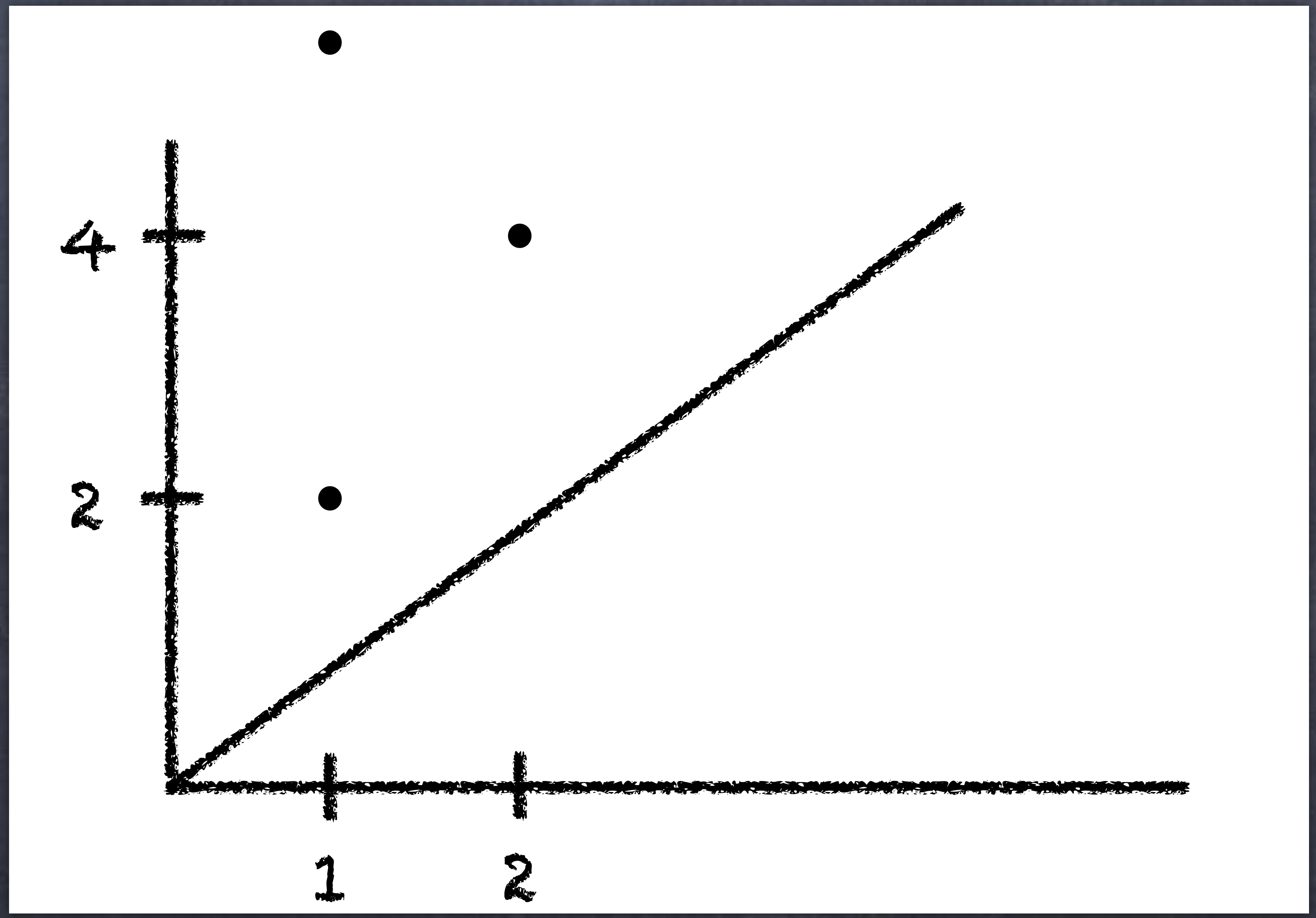


$$H_*(X_1) \to H_*(X_2) \to H_*(X_3) \to H_*(X_4) \to \cdots$$

Birth time

Death time

One can choose bases compatible with the linear maps



Persistence diagram

- Vectorise persistence diagrams

- Vectorise persistence diagrams
- Feed them into statistical analysis / ML

- Vectorise persistence diagrams
- Feed them into statistical analysis / ML
- There is good software for these computations:
  GUDHI, Ripser, Giotto-TDA, and more

Thank you

- Nick Sale used these tools to study vortices in SU(2) gauge theory.
- Later today:

  Xavier Crean on monopoles in U(1) gauge theory.

  Biagio Lucini on monopoles in SU(3) gauge theory.

Thank you