

Topological data analysis

CS594: Big Data Visualization & Analytics

Fabio Miranda

<https://fmiranda.me>

Topological data analysis

- What is topology?
- Why topological data analysis?
- What is topological data analysis?
- Mathematical background
- Applications
- Scikit-TDA

What is topology?

- Mathematical field that studies properties and relationships that are preserved under continuous transformations.
 - Stretching, twisting: continuous changes.
 - Tearing, gluing: discontinuous changes.
- Study the connectedness of a space.



Topological data analysis

- Approach that uses topology for the analysis of datasets.
- General framework to analyze data that can be:
 - High dimensional
 - Incomplete
 - Noisy
- Study the shape of the data.
 - Helps understand relationships between how data varies and its domain.
- General framework for feature extraction from raw data.

Topological data analysis

- Why topological data analysis?
 - Growing interest in academia.
 - 10% of SciVis papers since 2008.
 - ~20% in 2018.
 - Growing interest in companies.

The New York Times [SUBSCRIBE NOW](#) [LOG IN](#)

Ayasdi: A Big Data Start-Up With a Long History

BY STEVE LOHR JANUARY 16, 2013 7:00 AM [1](#)

Email Share Tweet Save More

Ayasdi is a data analysis start-up built on the career of one man: Gunnar Carlsson, a professor of mathematics at Stanford.

Government spending for science has helped a lot too. Dr. Carlsson was the principal investigator on research projects that were financed, from 2000 to 2008, with \$10 million from the National Science Foundation and the Defense Advanced Research Projects Agency, or Darpa.

In 2008, when Dr. Carlsson and his co-founders wanted to try to commercialize the research, it was Darpa and another government organization, IARPA, or Intelligence Advanced Research Projects Activity, that put up \$1.25 million in the form of a Small Business Innovation Research grant, for "high-

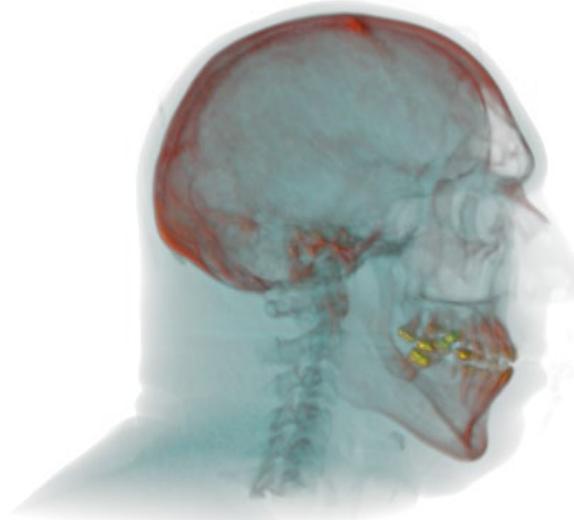
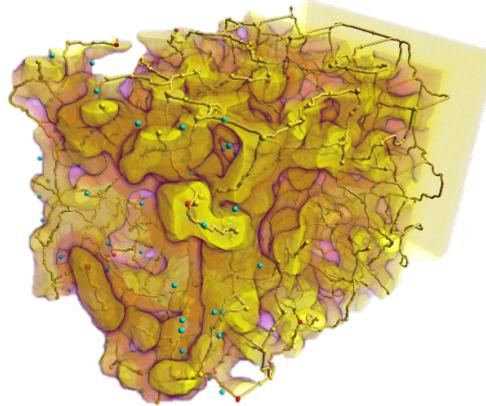


The three founders of Ayasdi, from left, Gurjeet Singh, Gunnar Carlsson and Harlan Sexton. Mark Leet Photography

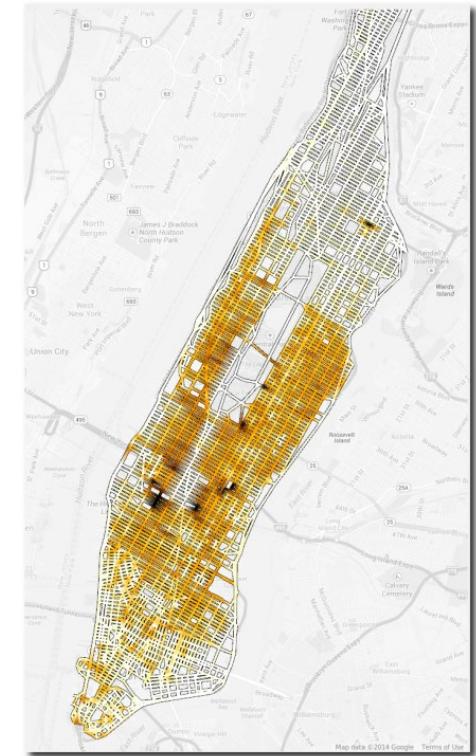
Topological data analysis



Simulation data



Medical data



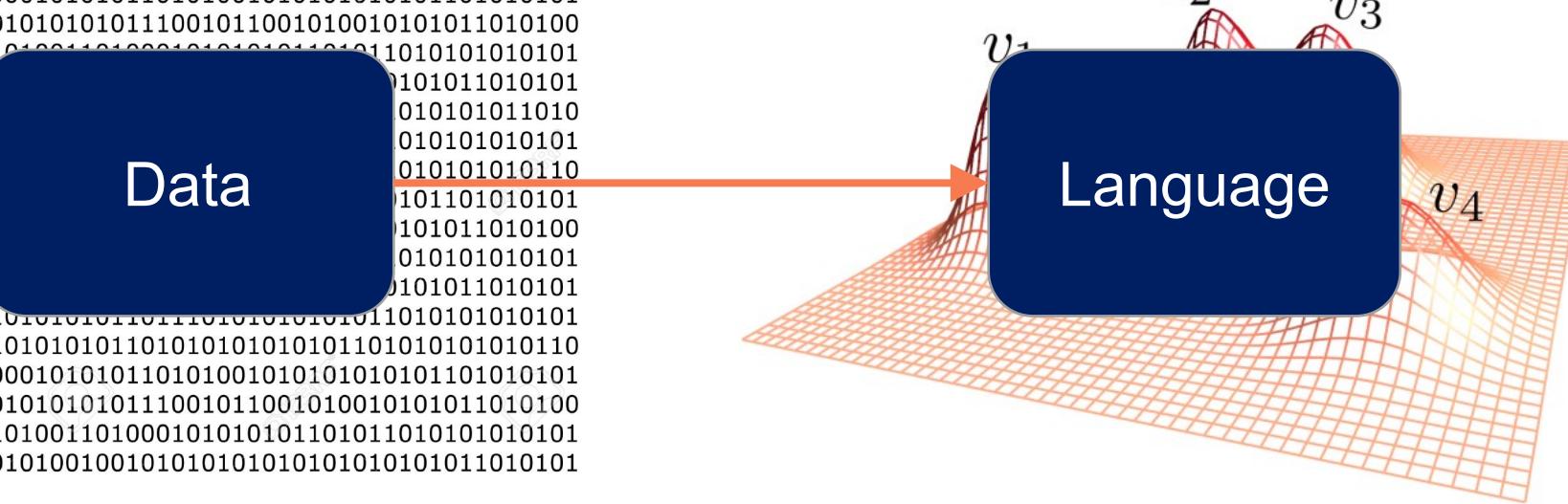
Urban data

Topological data analysis

- Data is complex
 - Size
 - Rich features
- Topological data analysis is a framework that summarizes irrelevant stories to get at something interesting.

Topological data analysis

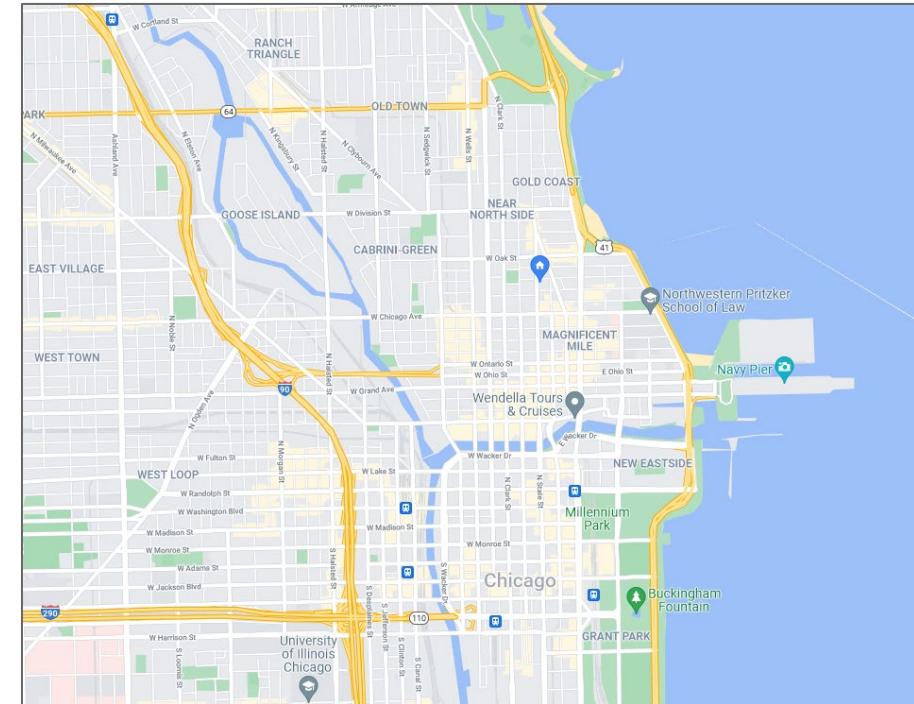
The image features a large, solid blue rounded rectangle centered on the page. Inside this blue box, the word "Data" is written in a large, white, sans-serif font. The background of the slide consists of a grid of binary digits (0s and 1s). A single bit, located in the middle column of the fourth row from the top, is highlighted with a thick red line, drawing attention to it.



Topological representations



Scalar function



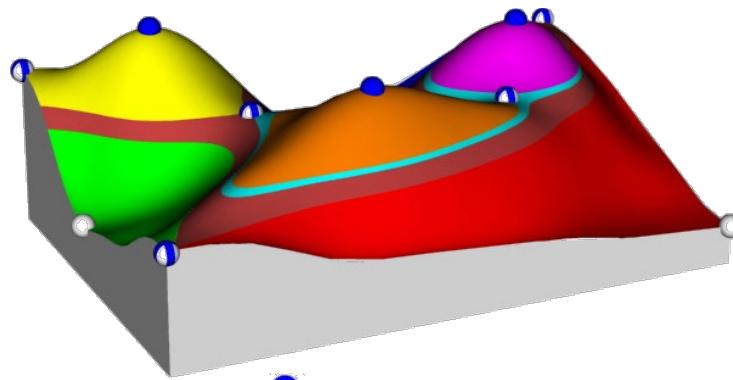
Structural representation

Topological representations

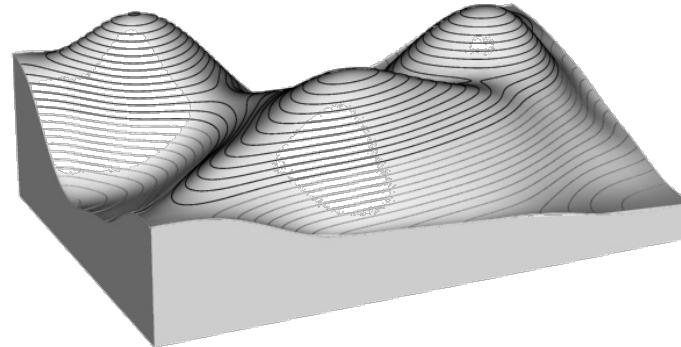
Contour trees

Reeb graphs

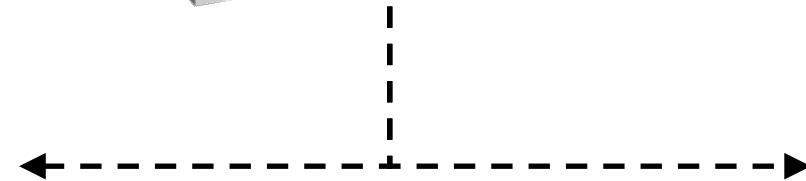
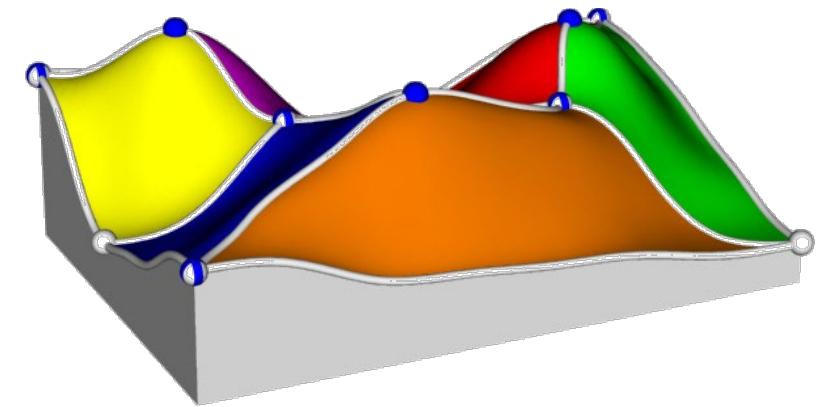
Merge trees



Scalar function



Morse-Smale complexes



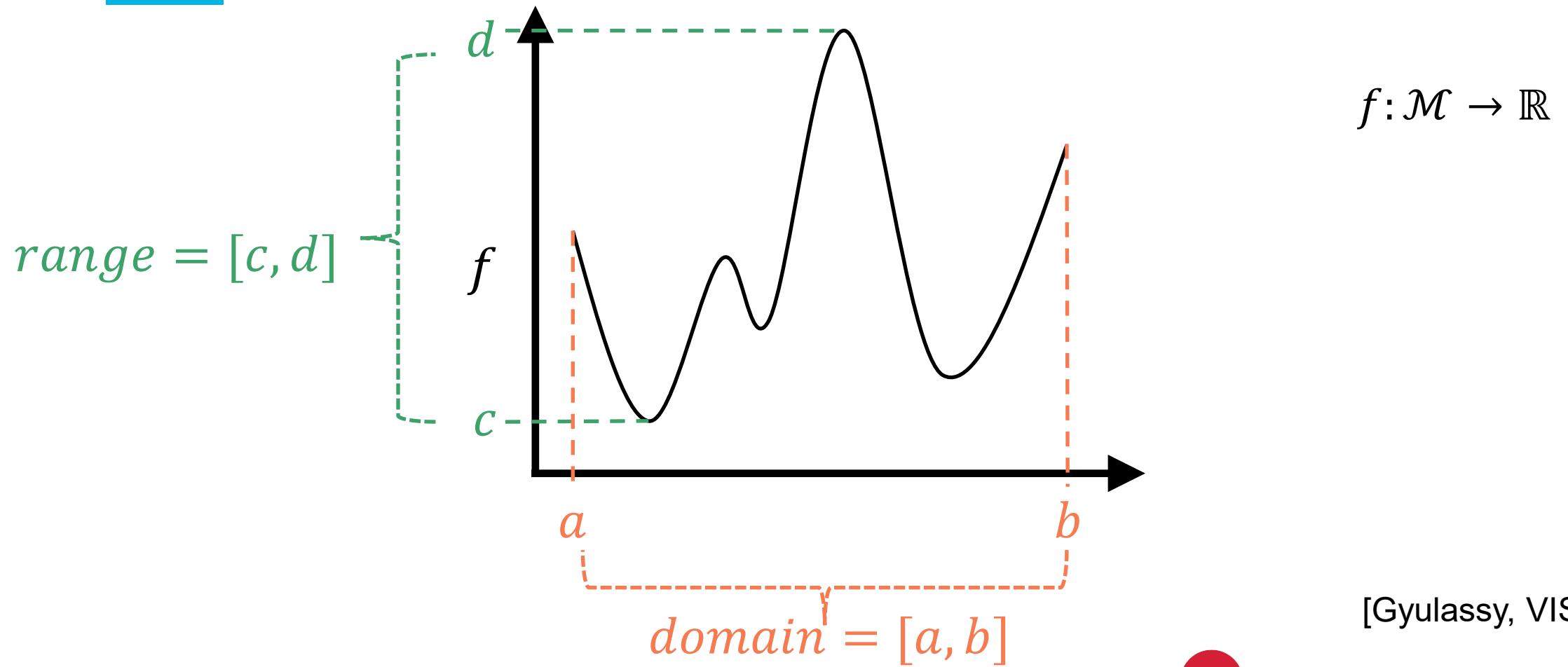
[Gyulassy, VIS 2018]

Topological data analysis

Data has shape, shape has meaning

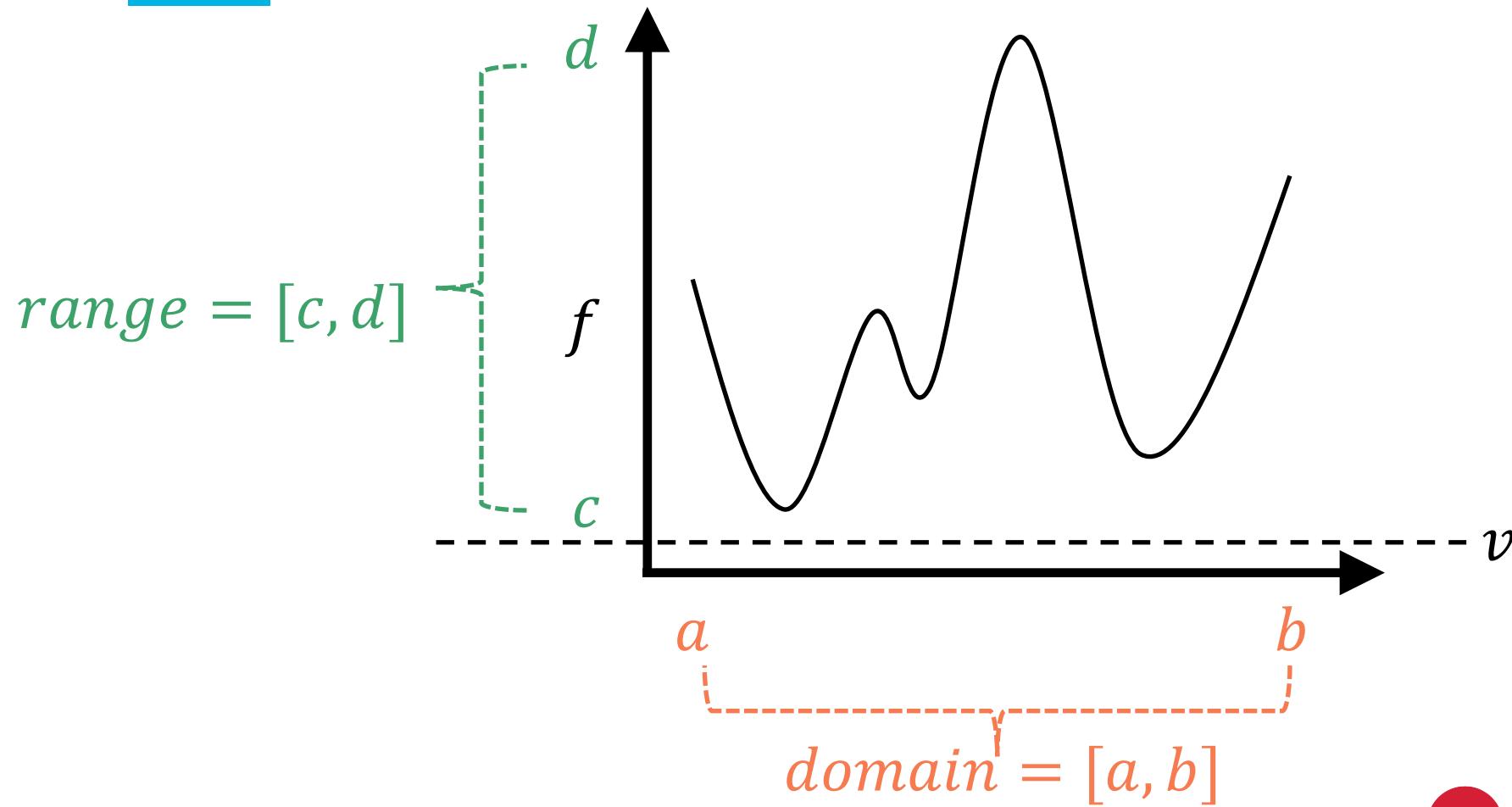
- How to extract meaning through the shape of the data?
- Shape of the data is related to the topological changes of the data.
- How to extract topological changes?

Topological changes



[Gyulassy, VIS 2018]

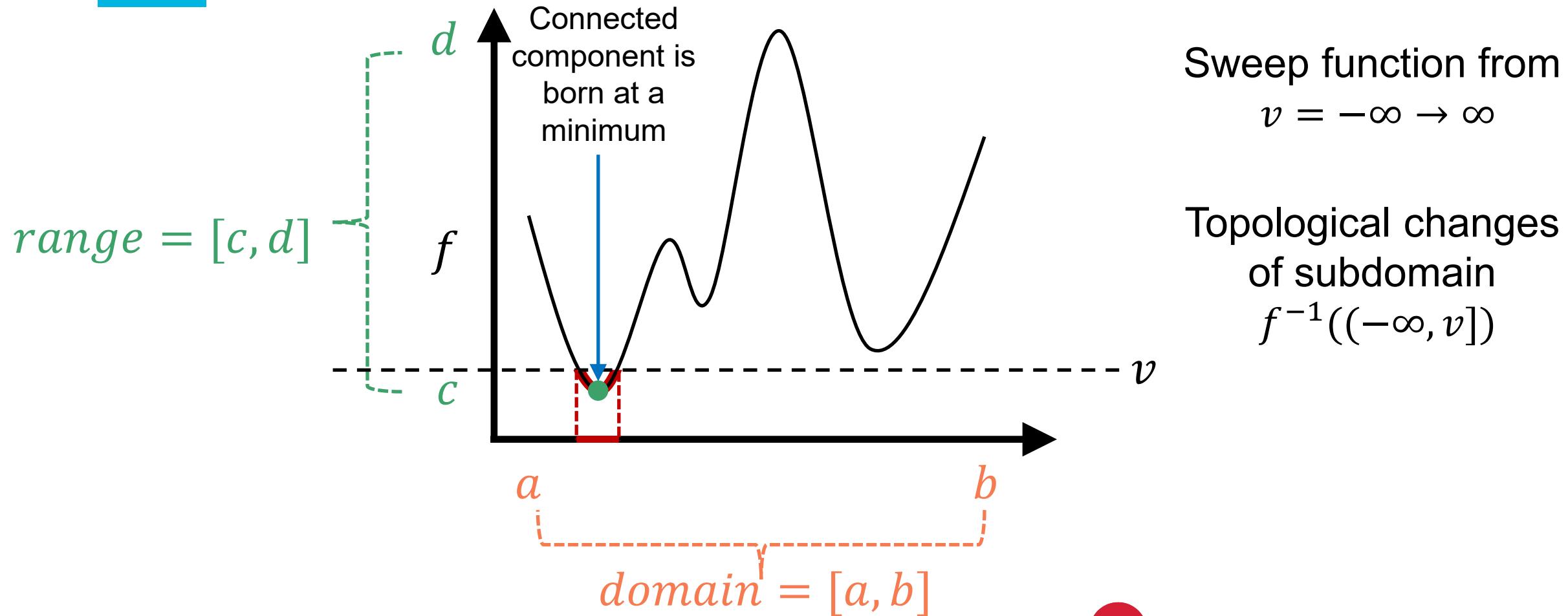
Topological changes



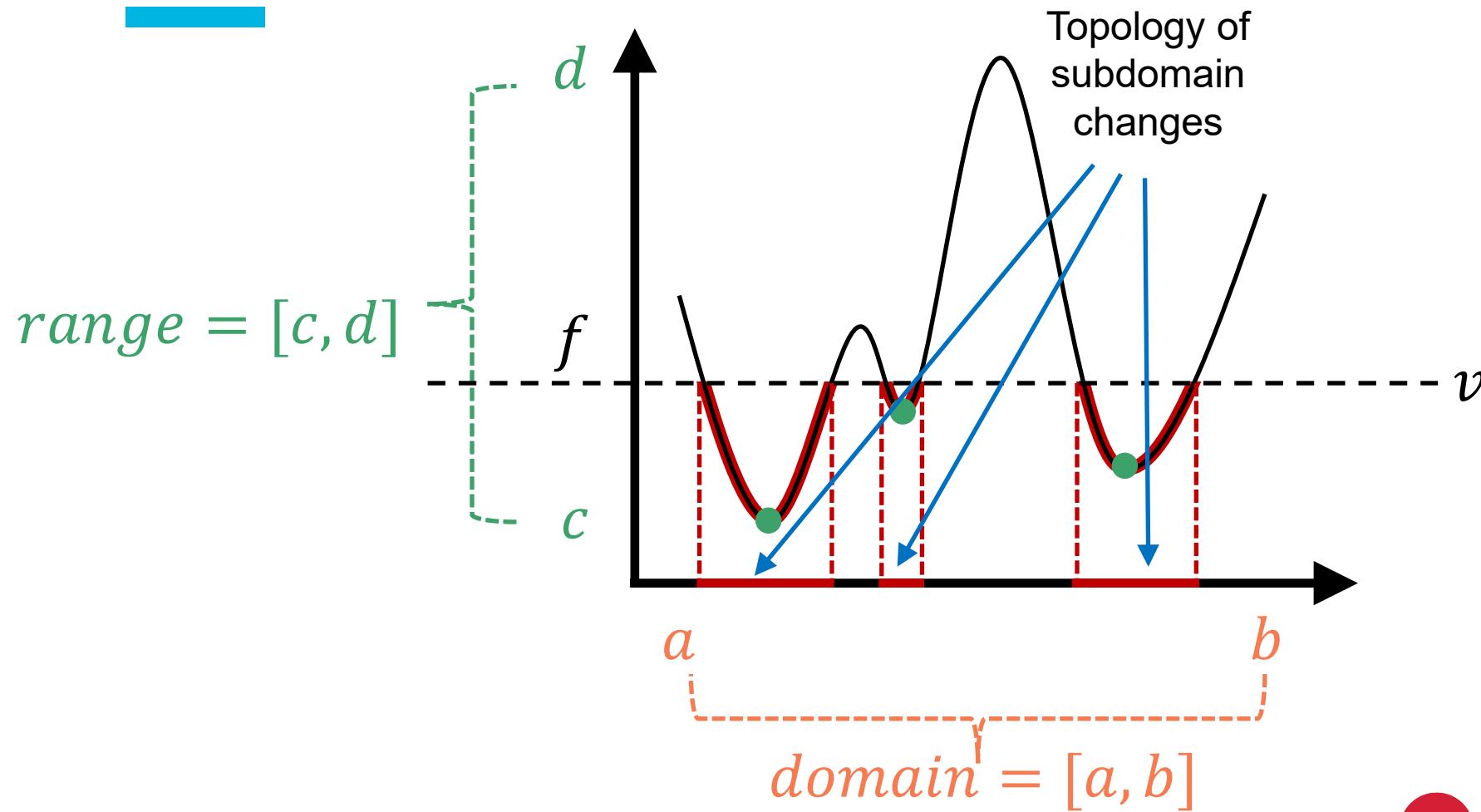
Sweep function from
 $v = -\infty \rightarrow \infty$

Topological changes
of subdomain
 $f^{-1}((-\infty, v])$

Topological changes



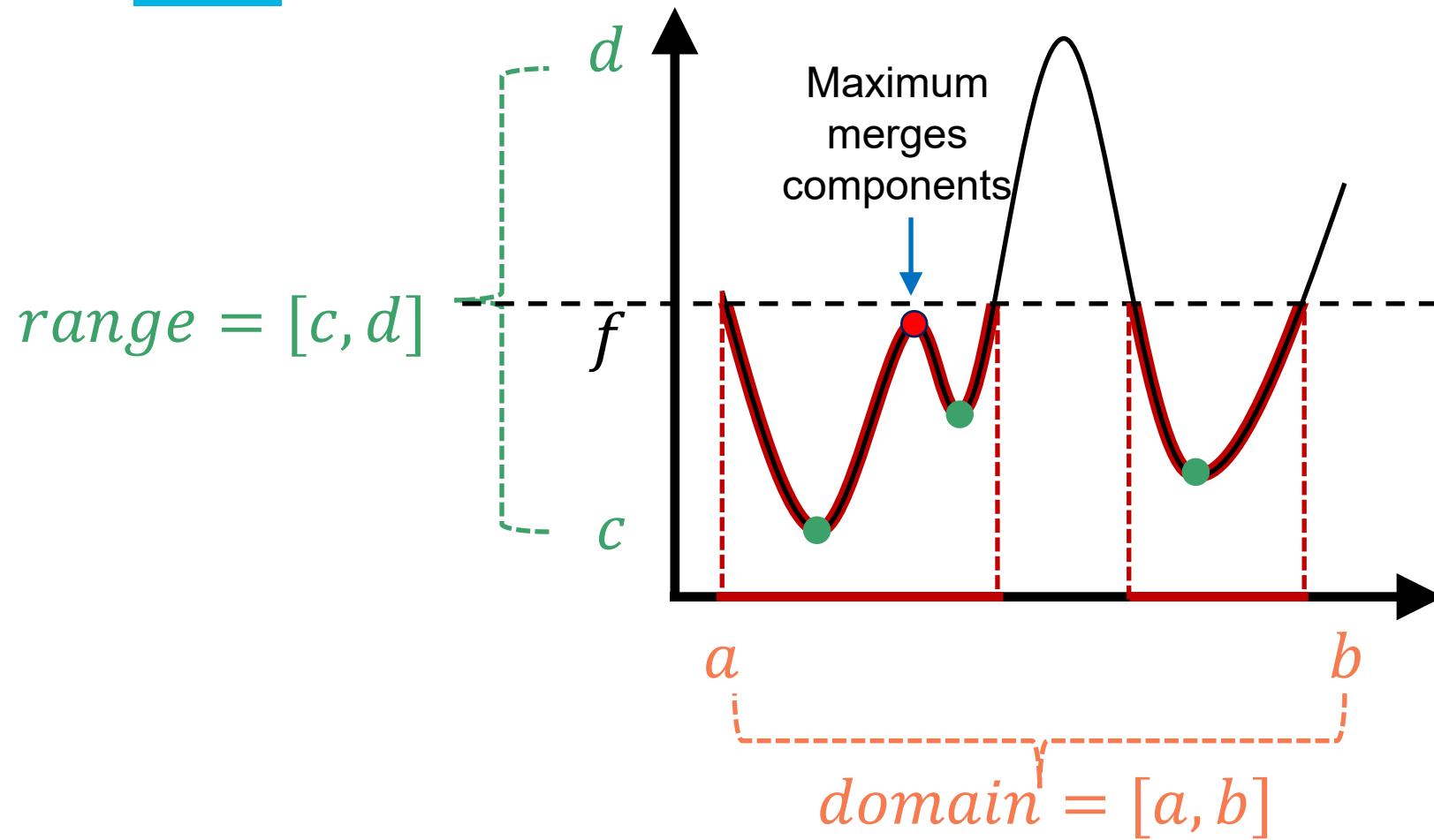
Topological changes



Sweep function from
 $v = -\infty \rightarrow \infty$

Topological changes
of subdomain
 $f^{-1}((-\infty, v])$

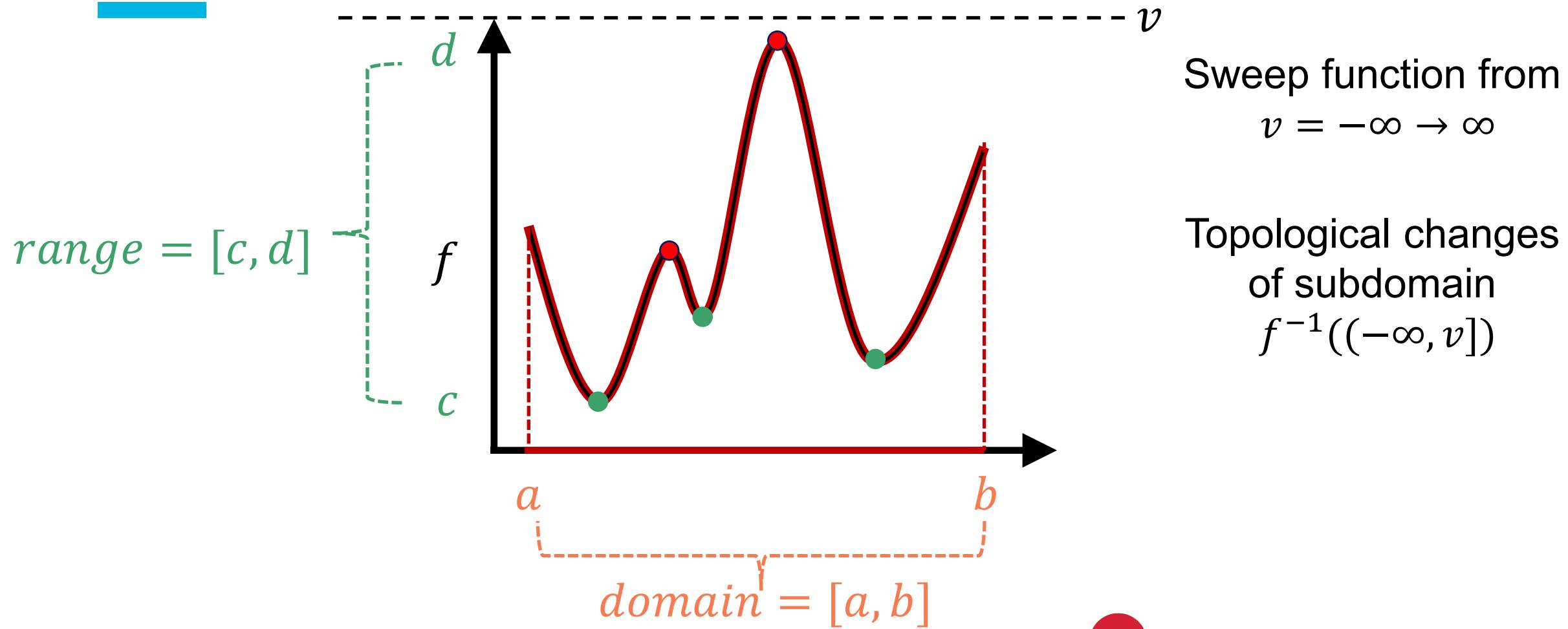
Topological changes



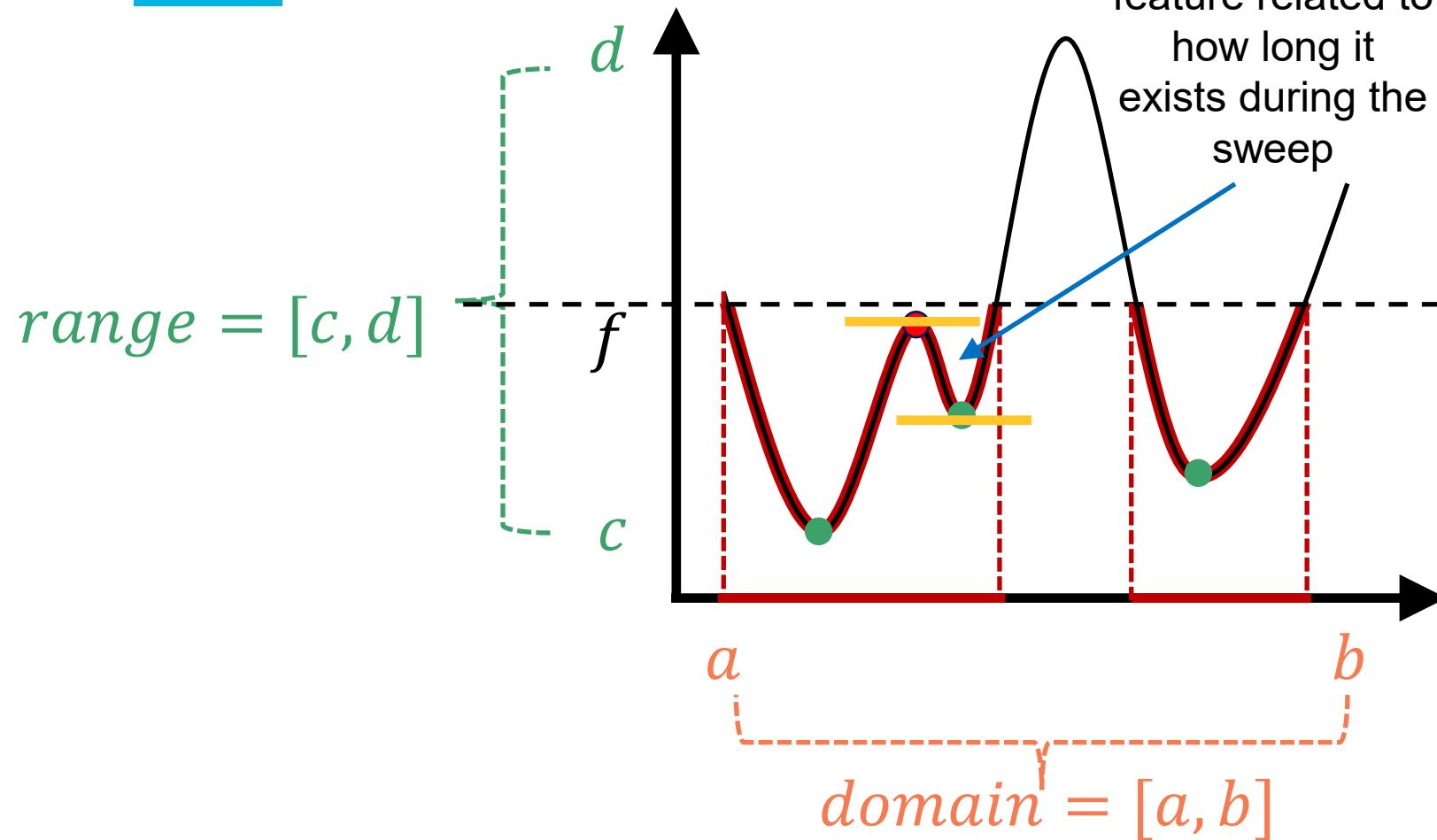
Sweep function from
 $\nu = -\infty \rightarrow \infty$

Topological changes
of subdomain
 $f^{-1}((-\infty, \nu])$

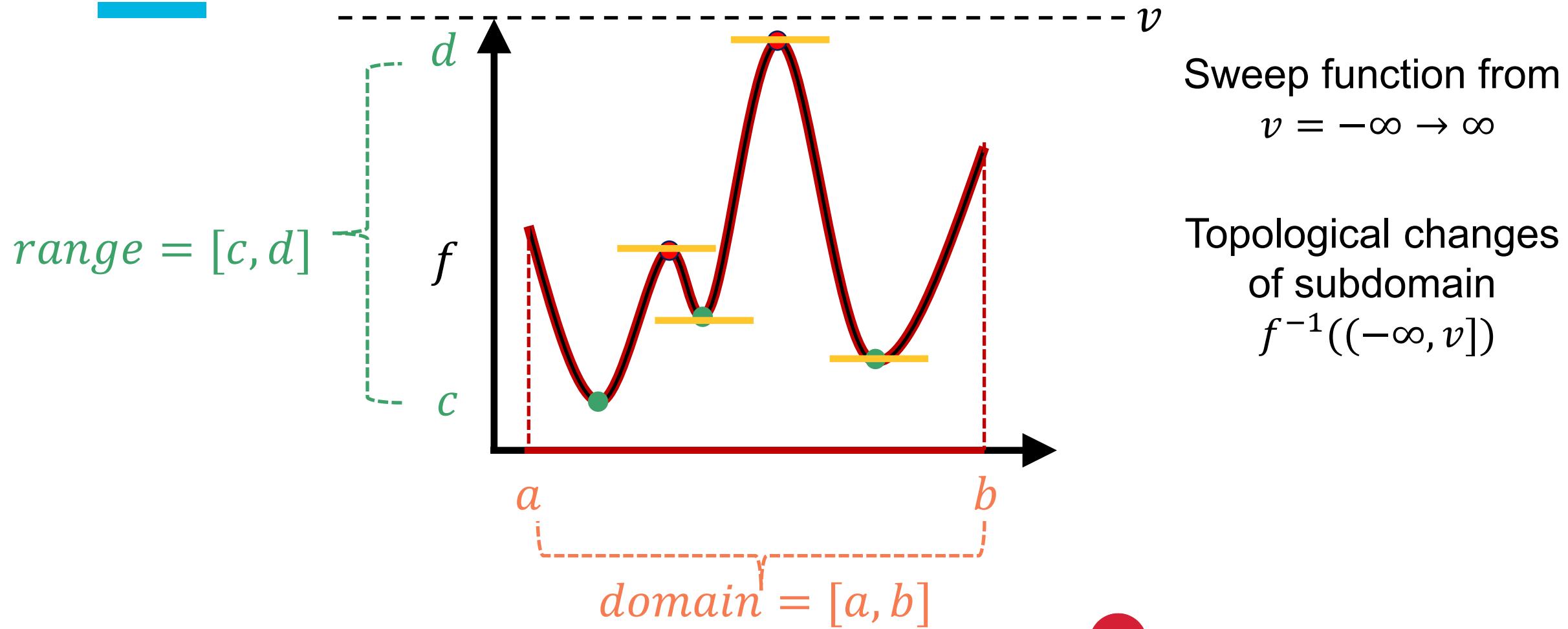
Topological changes



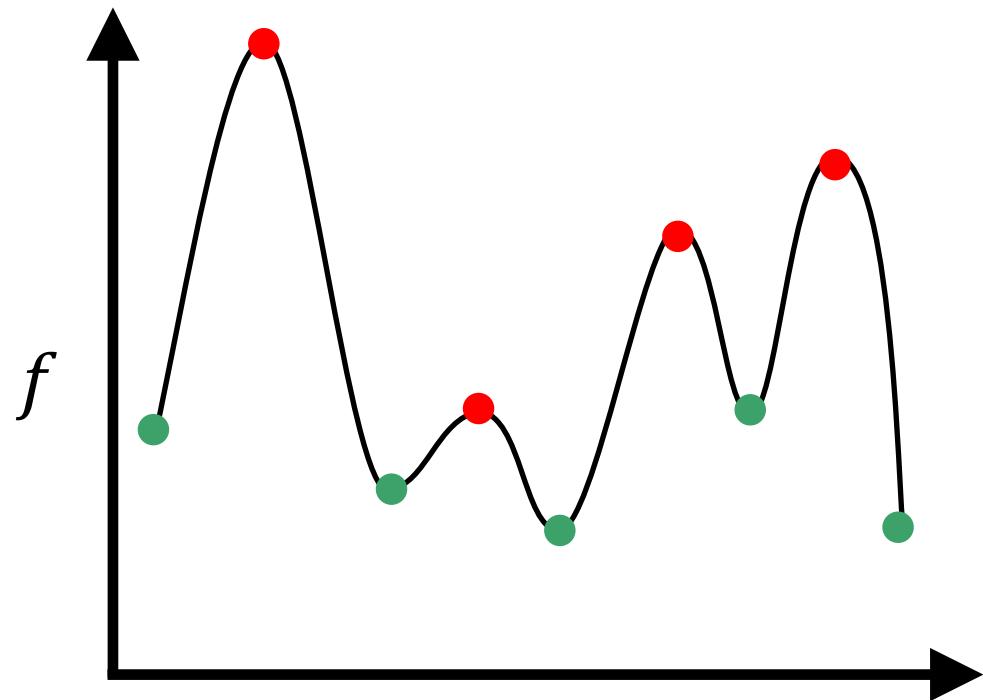
Topological changes



Topological changes

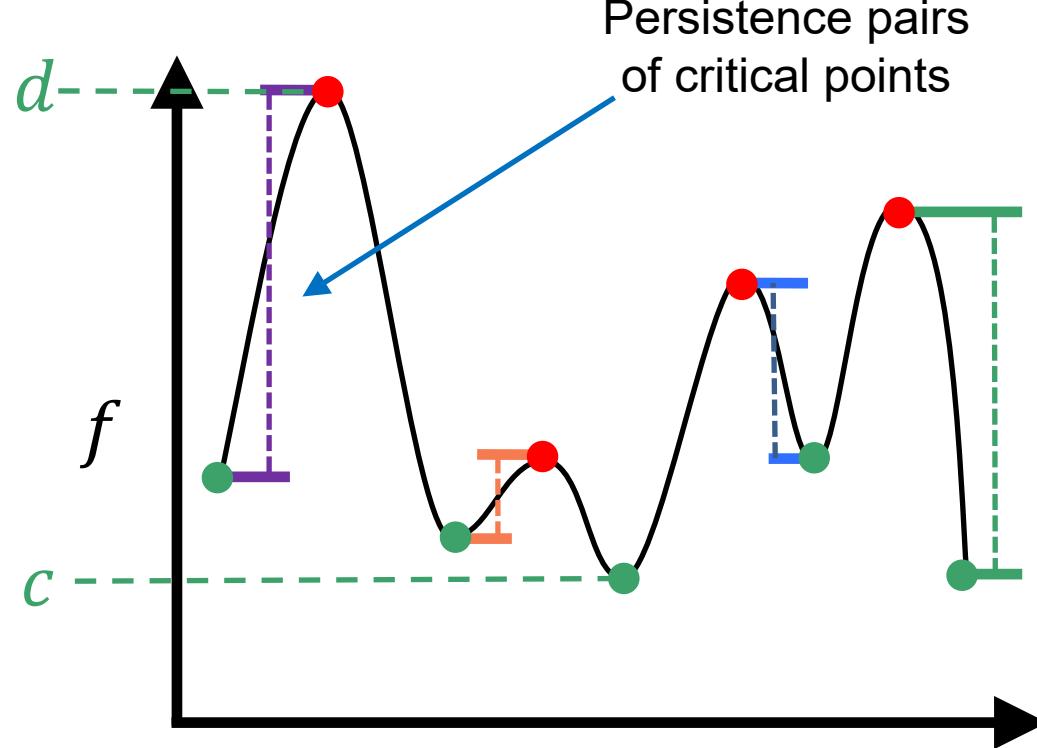


Features of a 1-dimensional function

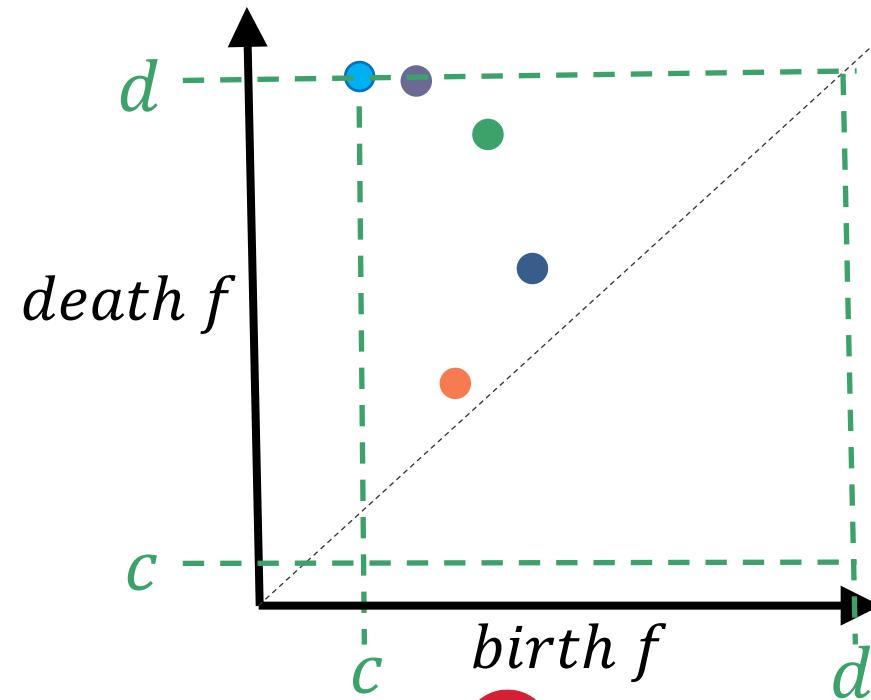


- Critical points where $\nabla f = 0$

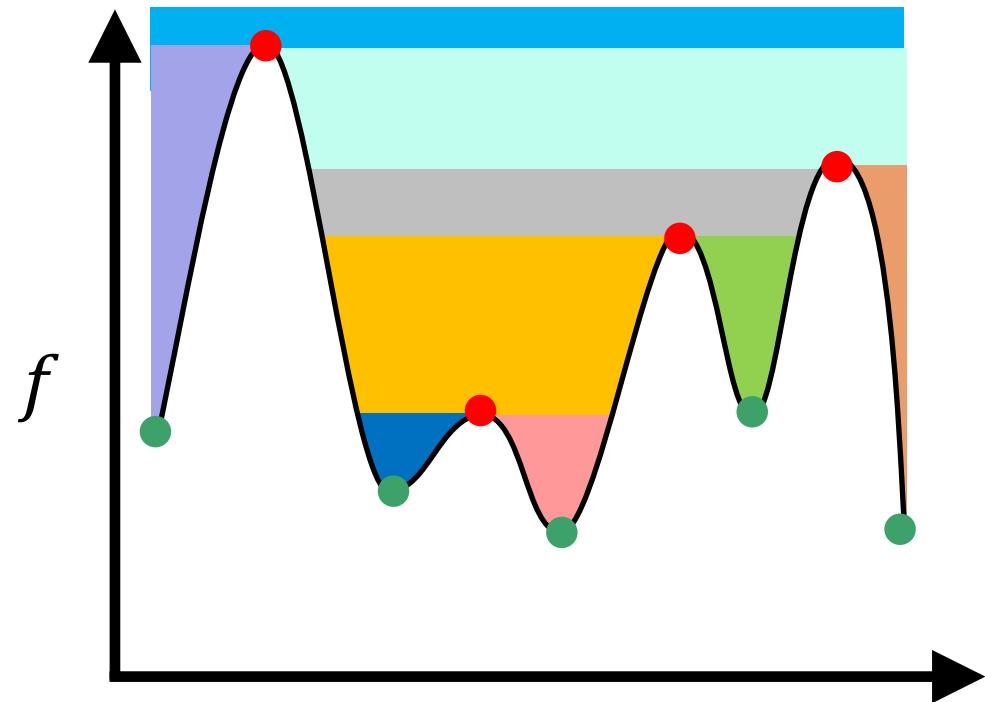
Features of a 1-dimensional function



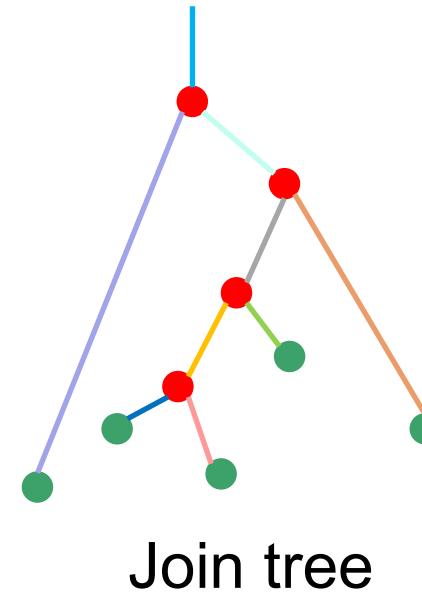
- Critical points where $\nabla f = 0$
- Persistence diagram



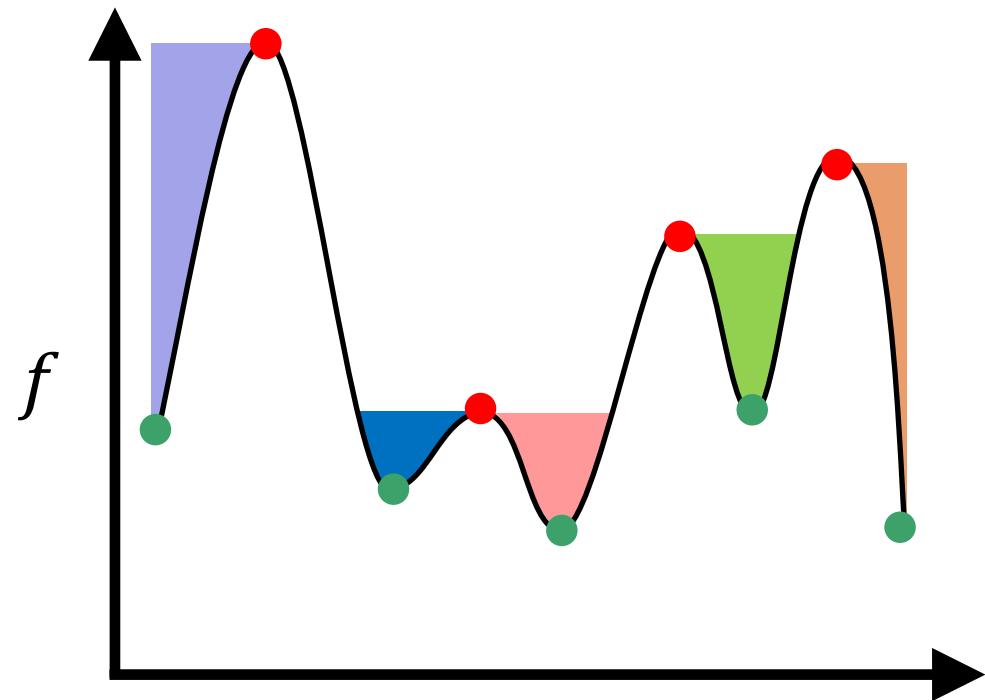
Features of a 1-dimensional function



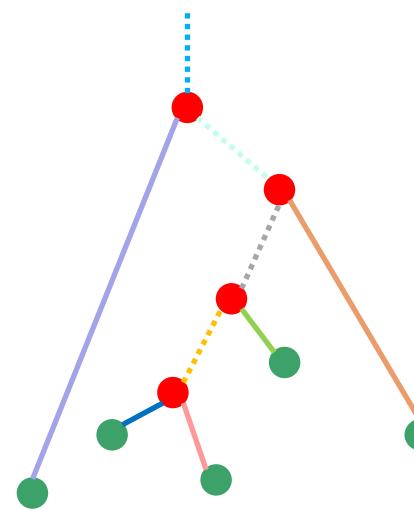
- Critical points where $\nabla f = 0$
- Components through sweep



Features of a 1-dimensional function

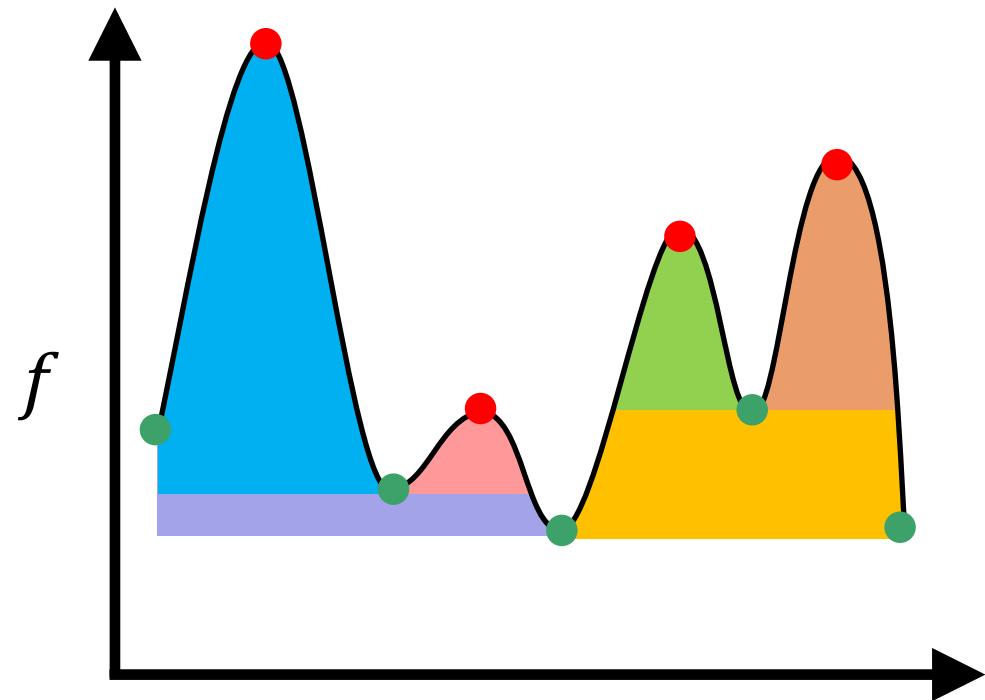


- Critical points where $\nabla f = 0$
- Components through sweep

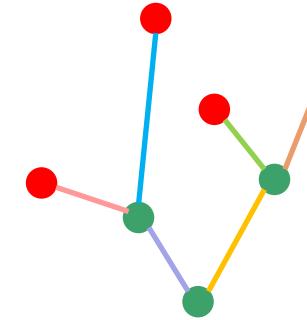


Join tree

Features of a 1-dimensional function

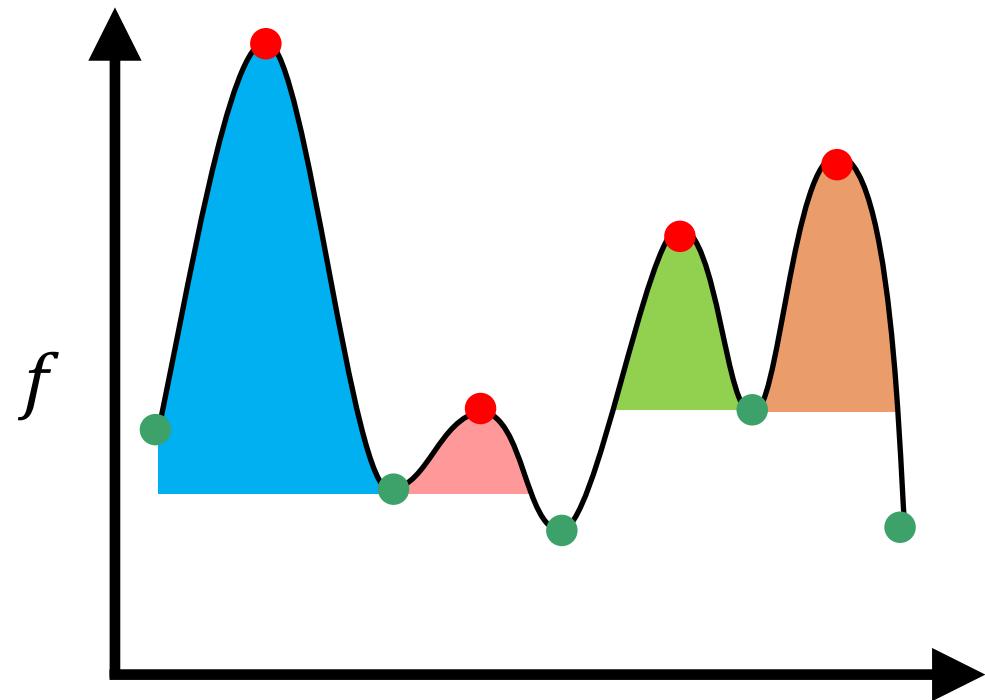


- Critical points where $\nabla f = 0$
- Components through sweep

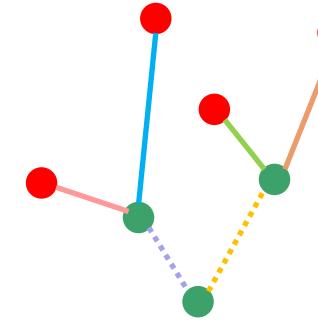


Merge tree

Features of a 1-dimensional function

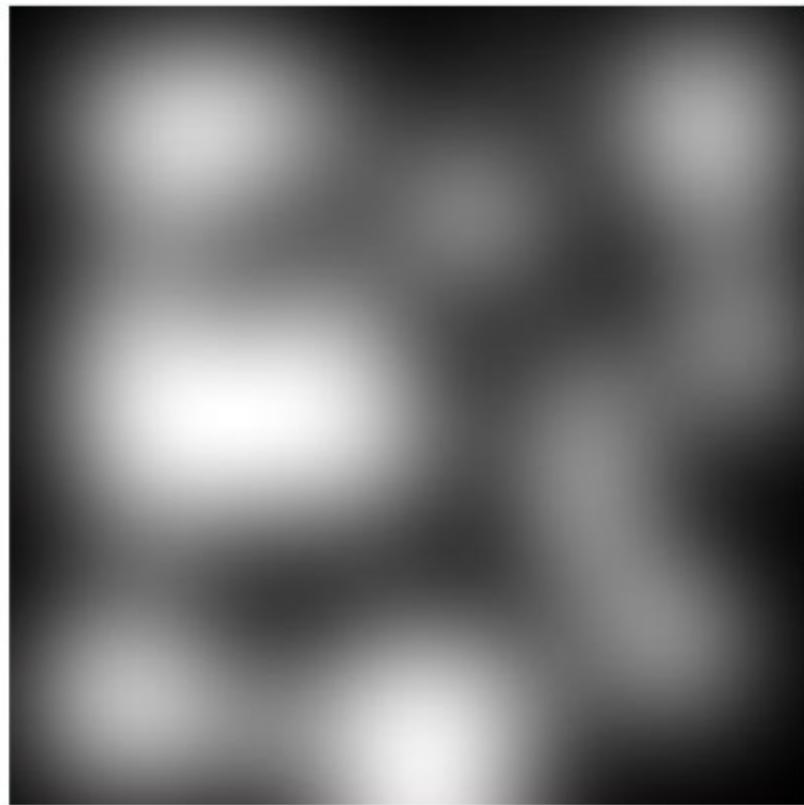


- Critical points where $\nabla f = 0$
- Components through sweep

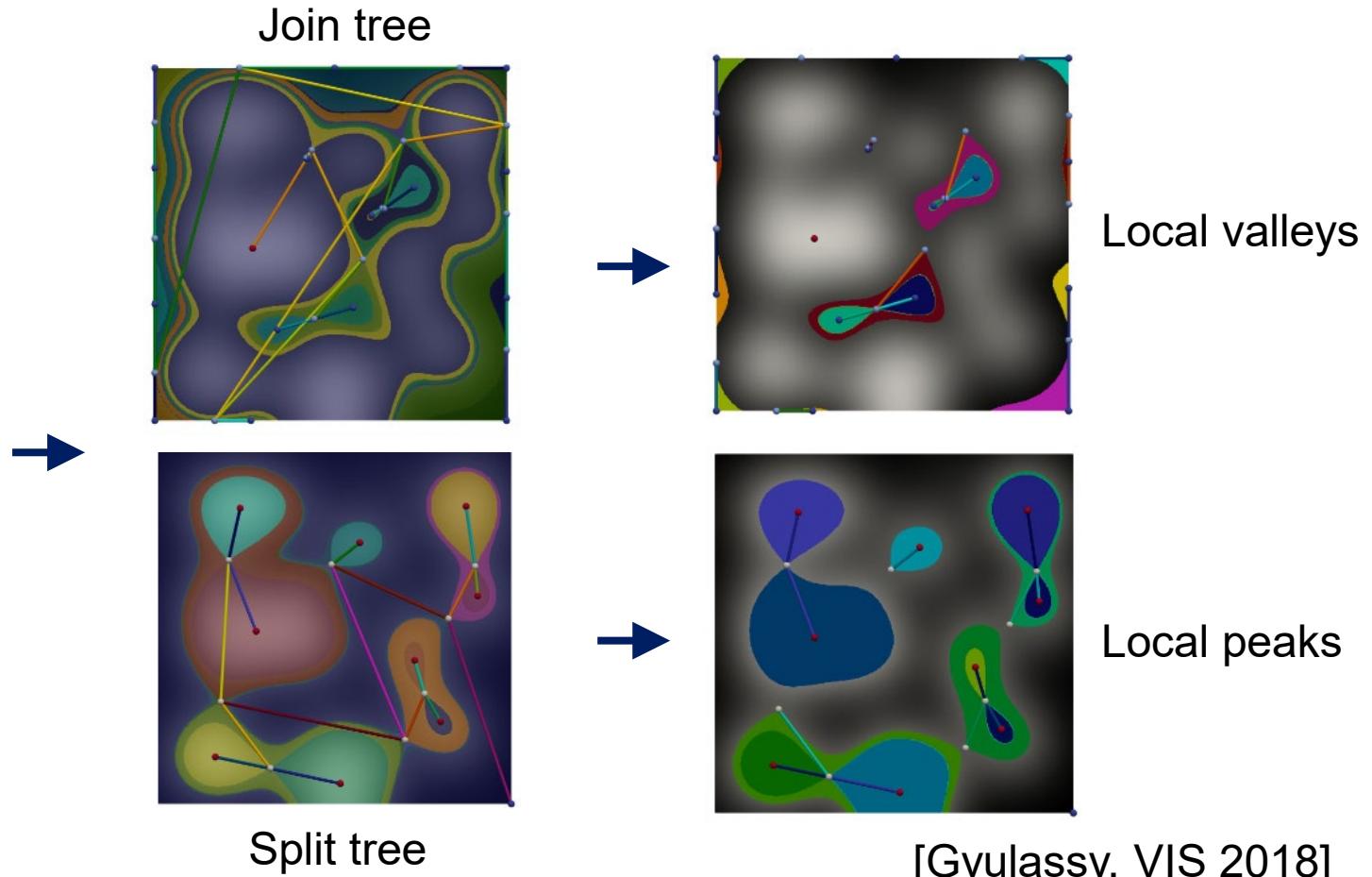


Merge tree

Features of a 2-dimensional function



Scalar function



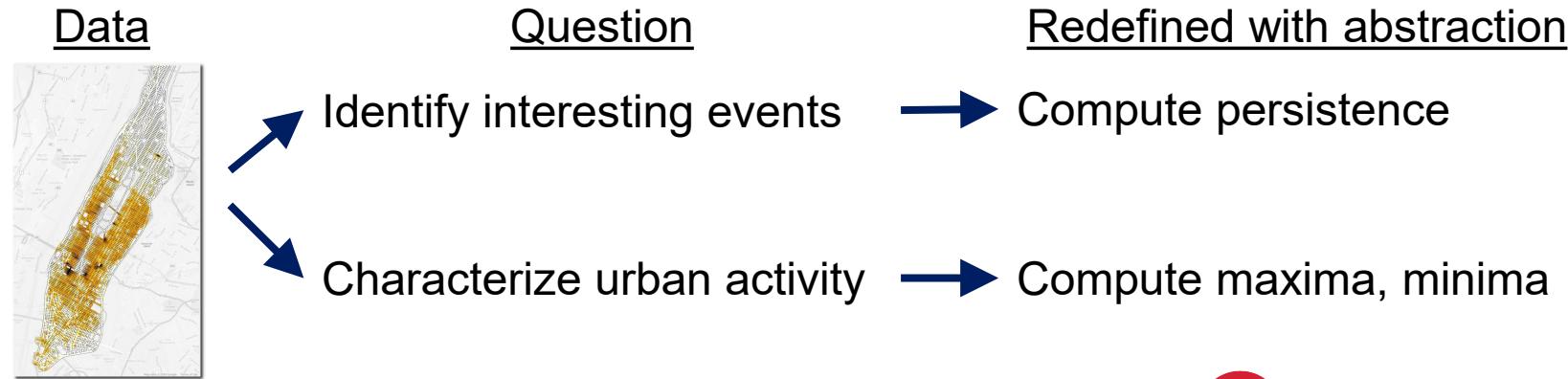
[Gyulassy, VIS 2018]



COMPUTER SCIENCE

Topological data analysis

- Analysis workflow:
 1. Generate data
 2. Pick an abstraction
 3. Define the features
 4. Vis: evaluate features
- Create hypotheses for topological features:



Applications

1. Taxi patterns: finding interesting events on spatiotemporal data.

- “Using Topological Analysis to Support Event-Guided Exploration in Urban Data” [TVCG 2014]

2. Urban Pulse: characterizing urban activity with spatiotemporal data.

- “*Urban Pulse: Capturing the Rhythm of Cities*” [TVCG 2016]

Using Topological Analysis to Support Event-Guided Exploration in Urban Data

Harish Doraiswamy *Member, IEEE*, Nivan Ferreira *Student Member, IEEE*, Theodoros Damoulas, Juliana Freire *Member, IEEE* and Cláudio T. Silva *Fellow, IEEE*

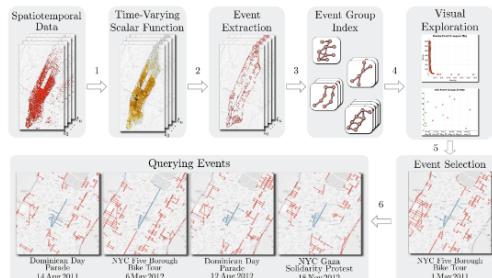


Fig. 1. Overview of the event guided exploration technique. First, (1) the input data is transformed into a time-varying scalar function. Topological features are computed from the scalar function to identify the set of events. (3) An event group index is then created on the identified events to support efficient querying over a large number of events. (4) A visual interface guides the user towards the identified events in an interactive fashion.

event analysis. The event analysis framework has been used to help both policy and administration and thereby help governments to monitor the level of their effects, increase the efficiency of their services, and reduce environmental harms of development. However, cities are complex systems and exploring the data they generate is challenging, as the interaction between the various components in a city creates complex dynamics where interesting facts occur at multiple scales, requiring users to inspect a large number of data slices over time and space. Manual exploration of these slices is ineffective, time-consuming, and can many times impede progress. In this paper, we propose a technique that supports event-guided exploration of large, complex, and dynamic data sets. Our approach is based on the concept of event analysis, which is a process of identifying, analyzing, and reporting events in a data set. We demonstrate the effectiveness of our technique on two different data sets from New York City (NYC): interesting taxi events and subway service. We also evaluate our technique on the feedback we received from

Keywords: Computational topology, event detection, cyclic temporal index, urban data, visual exploration

Urban Pulse: Capturing the Rhythm of Cities

Fabio Miranda, Harish Doraiswamy, *Member, IEEE*, Marcos Lage, Kai Zhao, Bruno Gonçalves, Luc Wilson, Mondrian Hsieh, and Cláudio T. Silva, *Fellow, IEEE*



Fig. 1. Comparing two popular tourist locations – Rockefeller Center in New York City (NYC) and Alcatraz Island in San Francisco (SF), using the pulse of these locations. The pulse, defined as a set of beats over multiple resolutions, captures the level of activity at a given location. In this example, the beats for the hourly and monthly resolutions are shown, based on Flickr activity. They are computed based on the topology of the time-varying scalar function that models the spatio-temporal distribution of the activity corresponding to a city. Dark green represents a significantly high activity at the location, while light green represent a relatively high activity compared to its neighboring locations. The similarity between the pulses of the two locations over the different resolution indicates that the level

The beats are then used to analyze and compare different pulses. We also design a visual exploration framework that allows users to explore the pulses within and across multiple cities under different conditions. Finally, we present three case studies carried out by experts from two different domains that demonstrate the utility of our framework.

INTRODUCTION

ties have been the loci of economic activity and human development through the ages, and will continue to be the source of many of the innovations and solutions to the challenges faced by society. In less than a century, the world has moved from less than 20% of the population residing in urban areas to over 50%. A number that is expected to rise

- F. Miranda, H. Douraiswamy, K. Zhao, B. Gonçalves, and C. Silva are with New York University. E-mail: {fmiranda, harishd, kai-zhao, bgoncalves, csilva}@nyu.edu*
M. Lugo is with Universidade Federal Fluminense. E-mail: mlagre@ic.uff.br
J. Wöhrel and M. Heizig are with Kuhn Pedersen, Inc., Associate PC.

cles [5, 22, 54], mobile devices [26], and social media [12, 29]. Cities all over the world are not only collecting these data, but they are also making the data available (see e.g., [43, 44, 45]). If properly analyzed, urban data can take us beyond today's imperfect and often anecdotal understanding of cities to enable better governance, informed planning, and improved policy. These data also offer a rich resource for social scientists to study the dynamics of urban society and behavior.

However, there are many challenges involved in enabling the effective analysis of urban data. They stem not only from the volume of data, but also from the inherent spatio-temporal complexity of the underlying processes in a city. The usual approach to analyze this kind of data is to use different types of aggregation and produce visual summaries [2, 35]. These lead to a trade-off between the level of aggregation

to efficiently transport, house, educate, employ and even entertain an ever increasing number of citizens on a daily basis. A better understanding of the structure of the population, and on how it is distributed and evolving can greatly help in this endeavour.

Technological innovations have enabled the automatic collection of a diverse set of data about our daily lives both as individuals and as a society. As a consequence, cities are not only collecting, but also making unique data sets available through open data portals and live streams [30, 31, 42]. An example that has quickly been followed by several other providers is the release of open data sets that reveal the potential that can be unlocked by combining diverse sources of data. This opens up new opportunities for city governments and social scientists to engage in new data-driven science to better understand cities, and improve the lives of their residents. Not surprisingly, there have been several research efforts in this direction, with the above goal (e.g., see [2, 3, 13, 19, 46]). However, most of these studies focus on identifying features or events in the data, or exploring the mobility of predefined entities. That is, the focus in these works

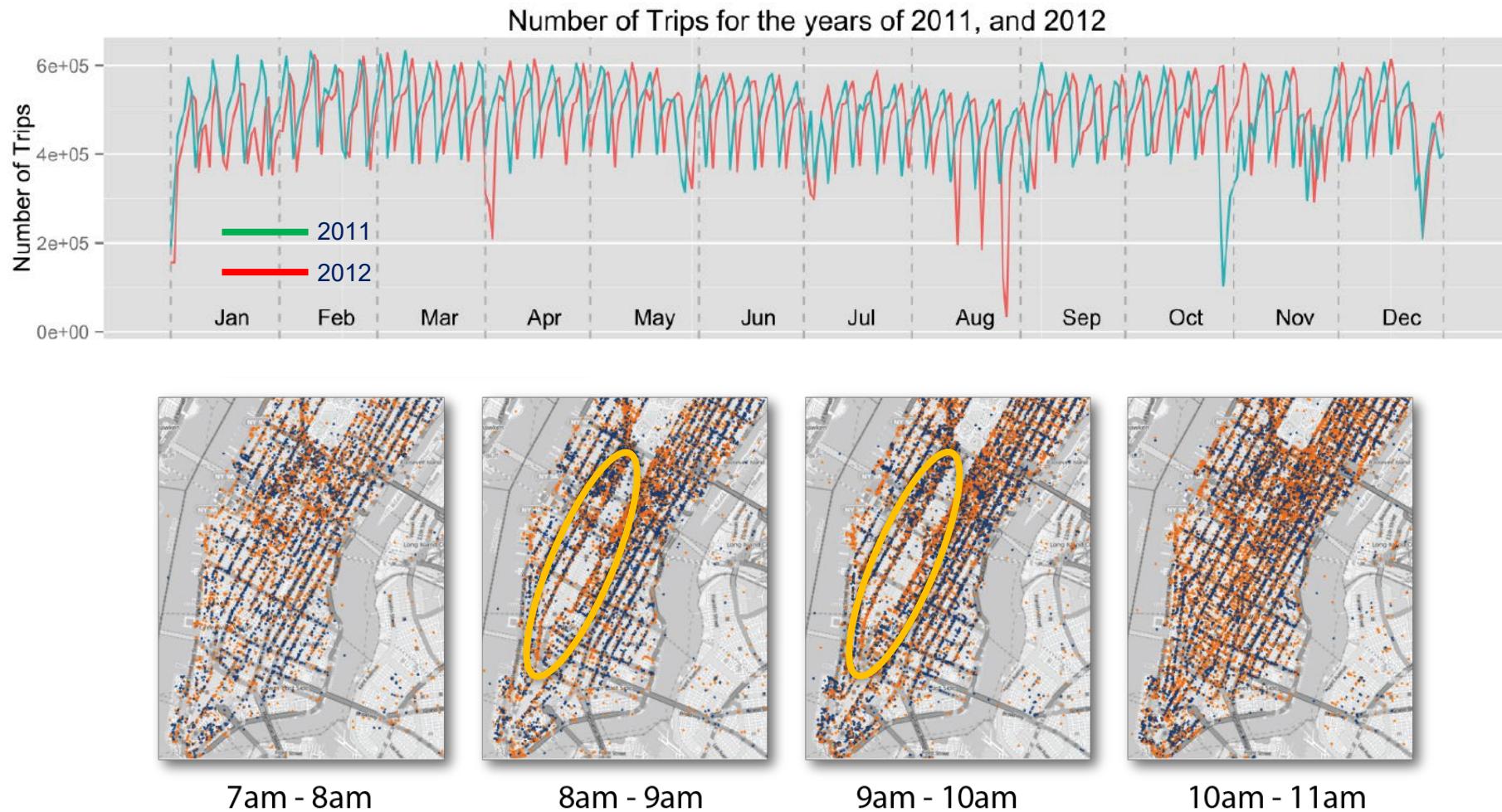
⁵ See also H. H. Salter, *Urbanization and the City* (London, 1963).

Applications: taxi patterns

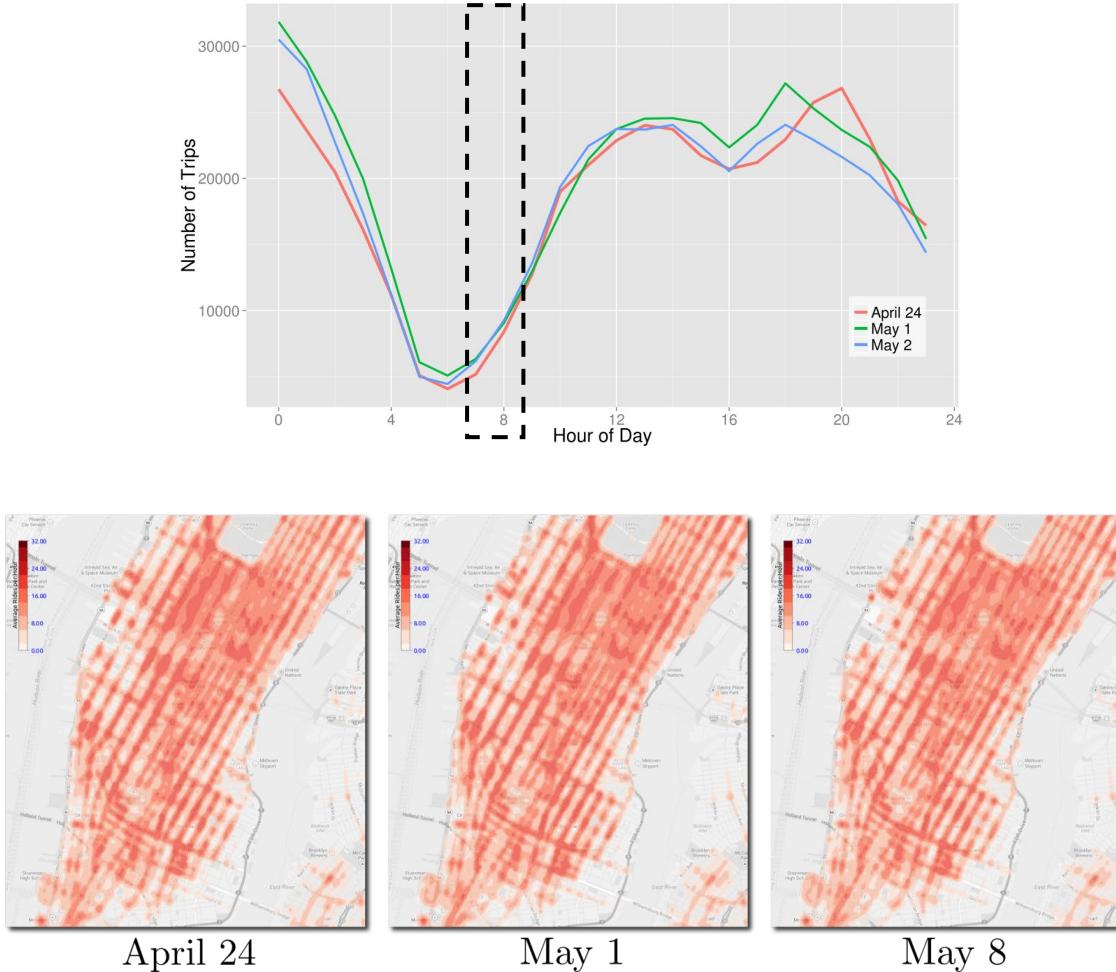
- NYC taxi data:
 - ~175 million trips / year
 - Spatiotemporal
 - Other attributes:
 - Fare, tip
 - Distance
 - Duration
 - ...



Taxi data slices

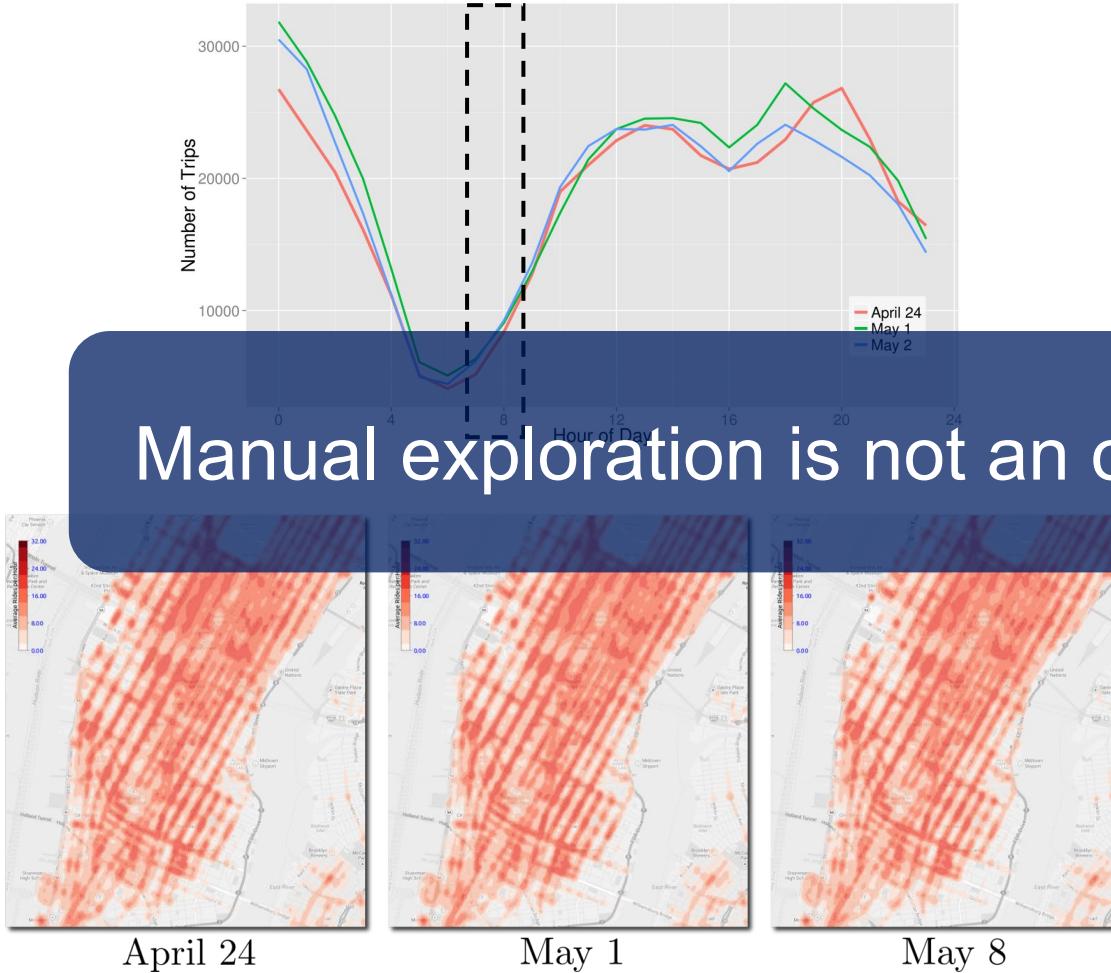


Taxi data slices



8am - 9am

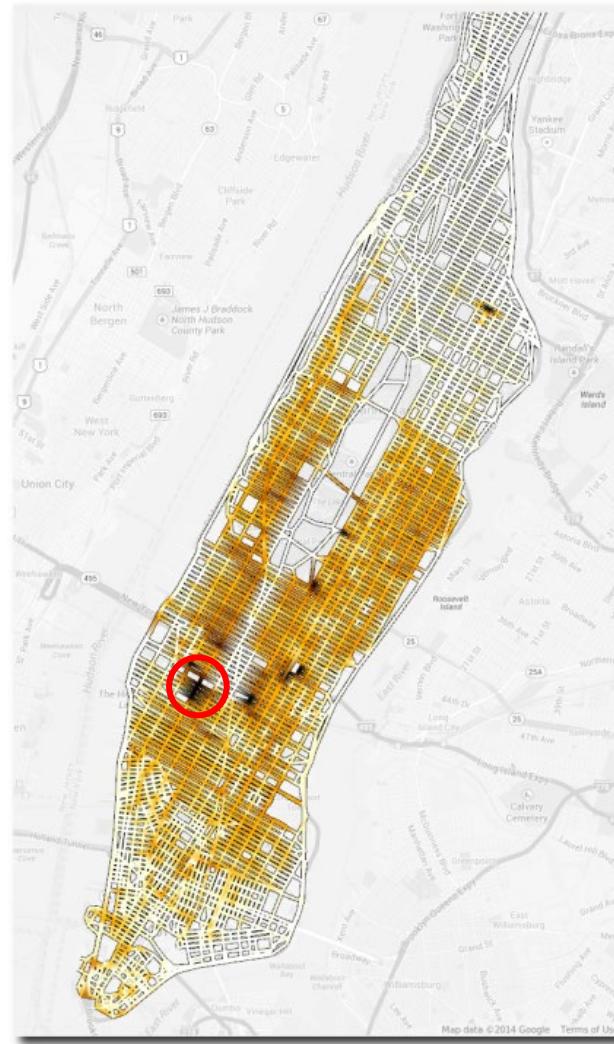
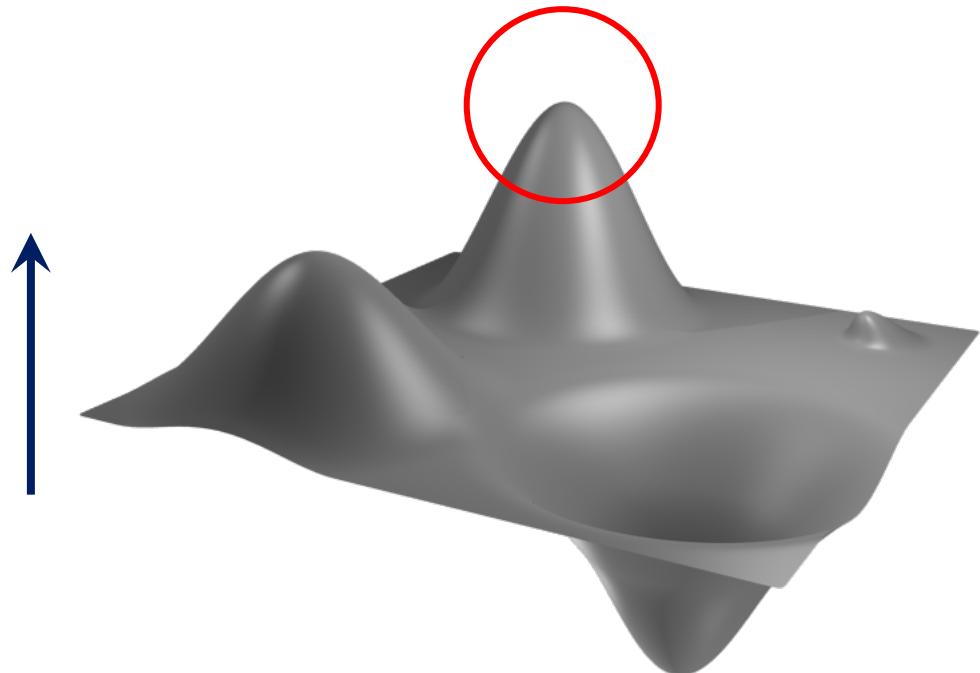
Taxi data slices



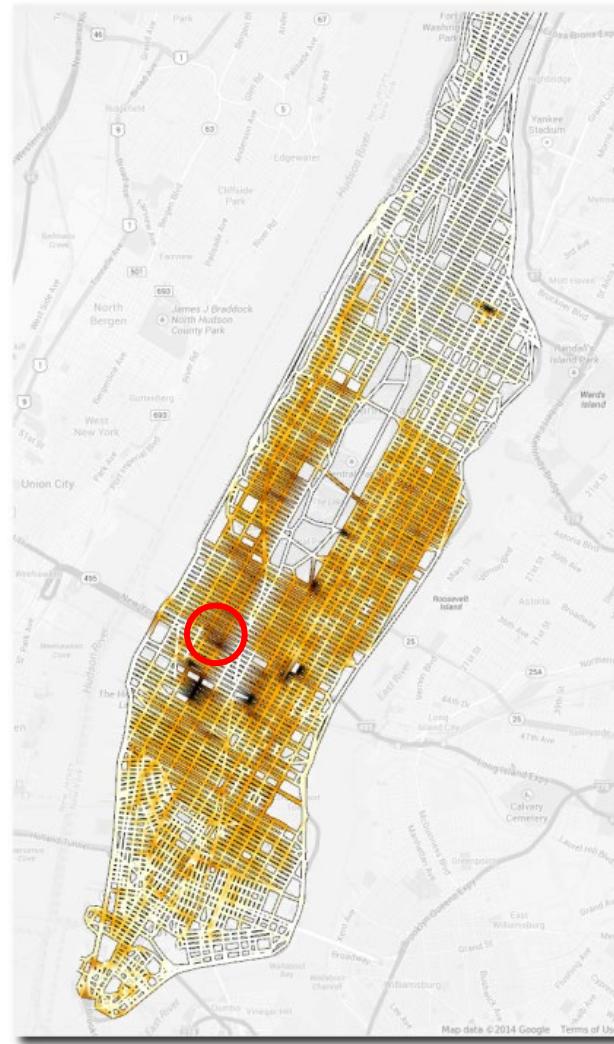
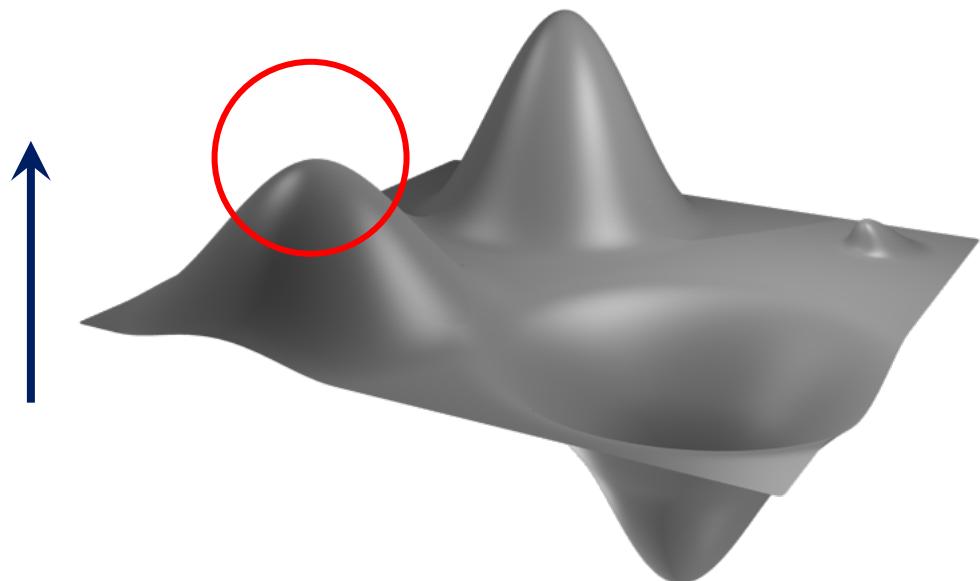
Taxi patterns: goals

- Guide users towards potentially interesting slices
- What is an interesting data slice?
 - Contains an “event”
- Flexible definition of events:
 - Arbitrary spatial structure.
 - Different types of events.
 - Multiple temporal scales.
- Efficient search for similar event patterns.

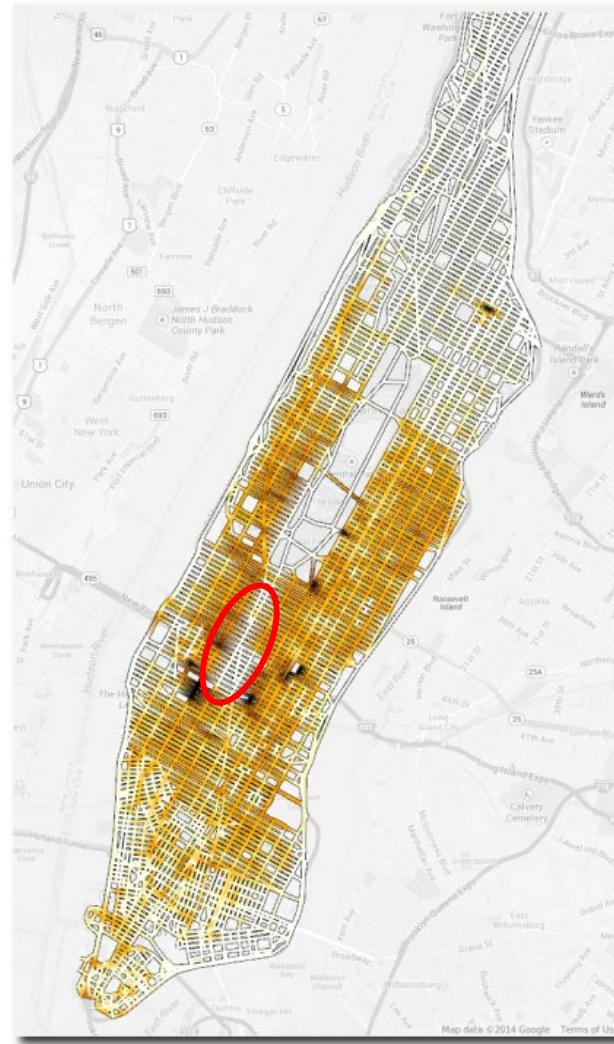
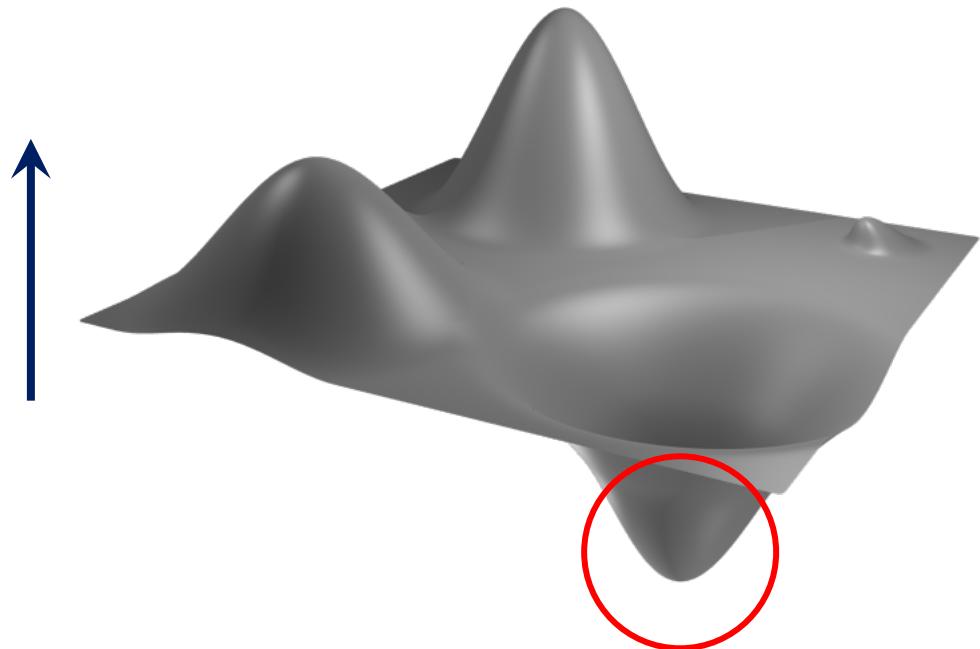
Topology of the data



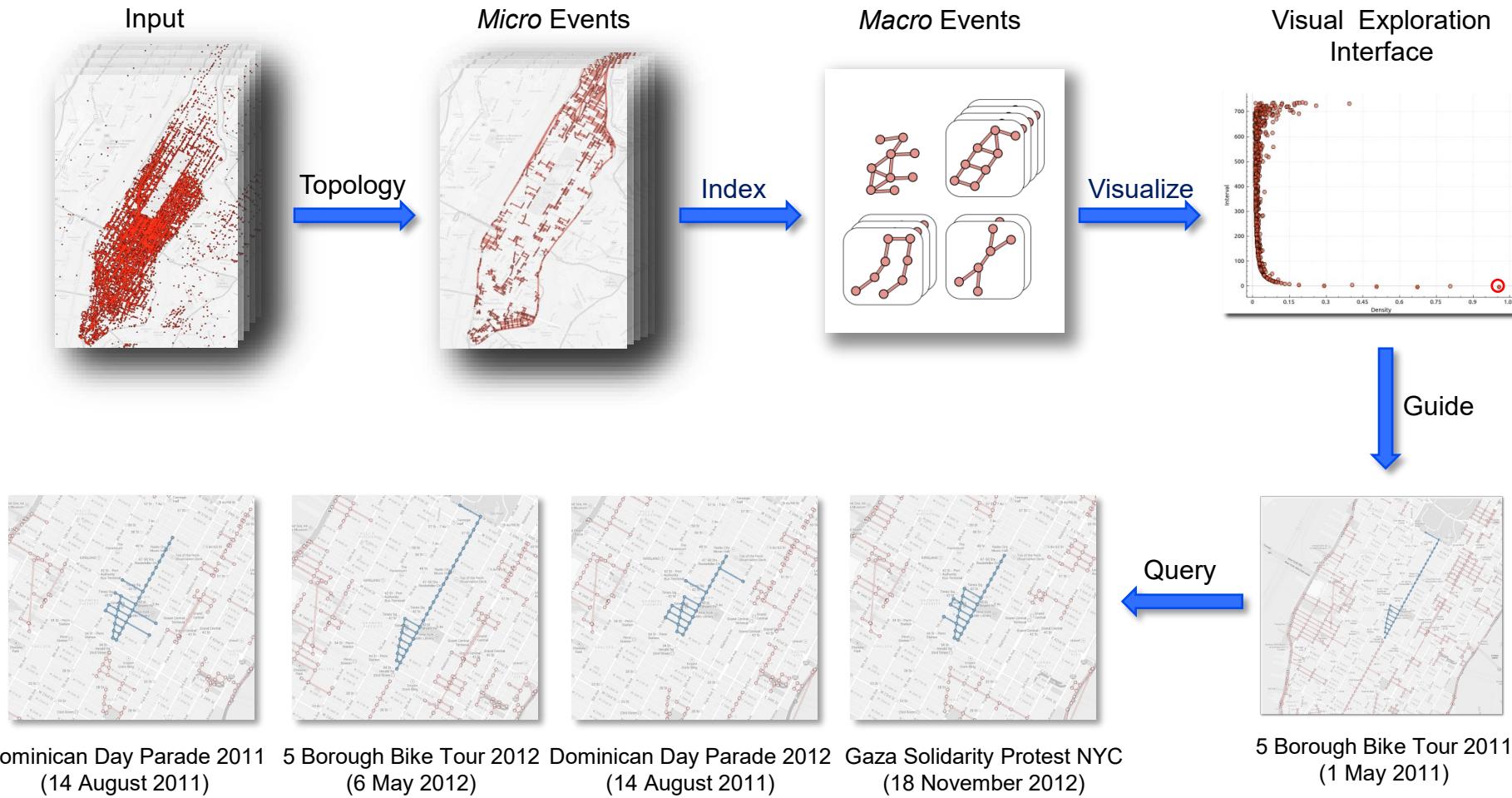
Topology of the data



Topology of the data

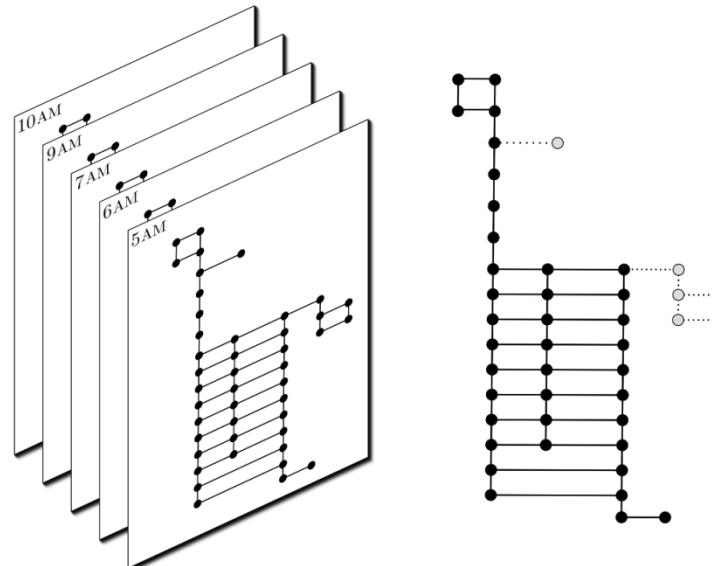


Topology framework



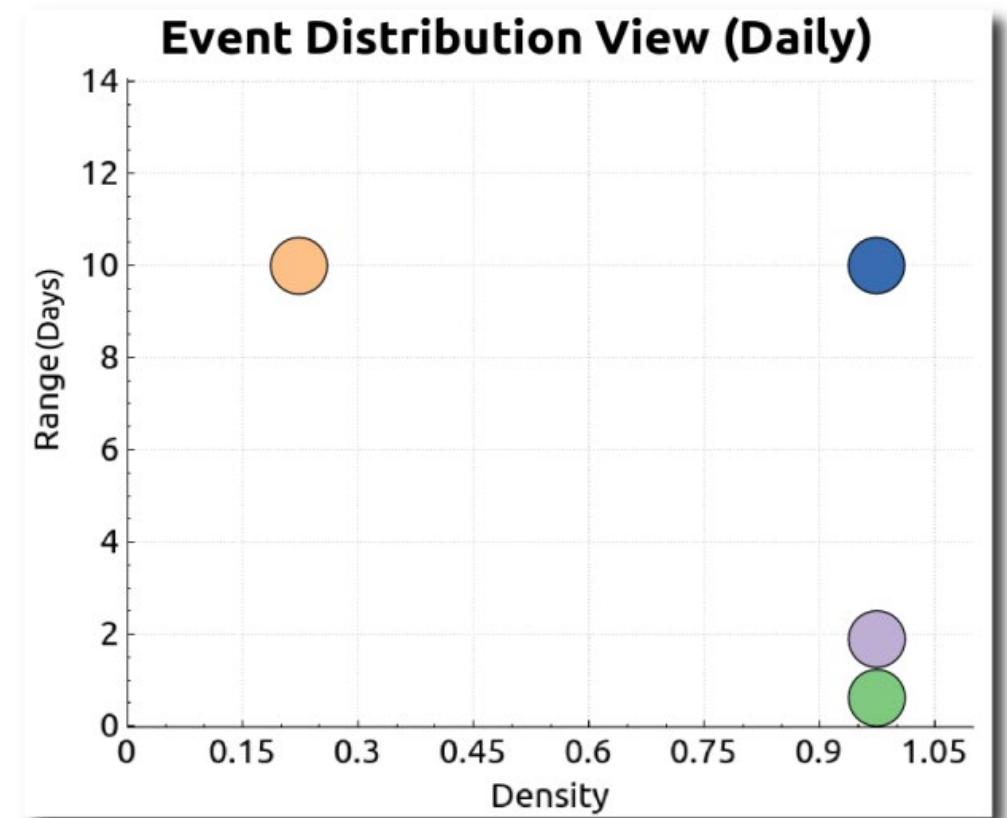
Exploring topological features

- Several features per time step.
- Group similar features within a larger time interval.
 - Represents “macro” events.

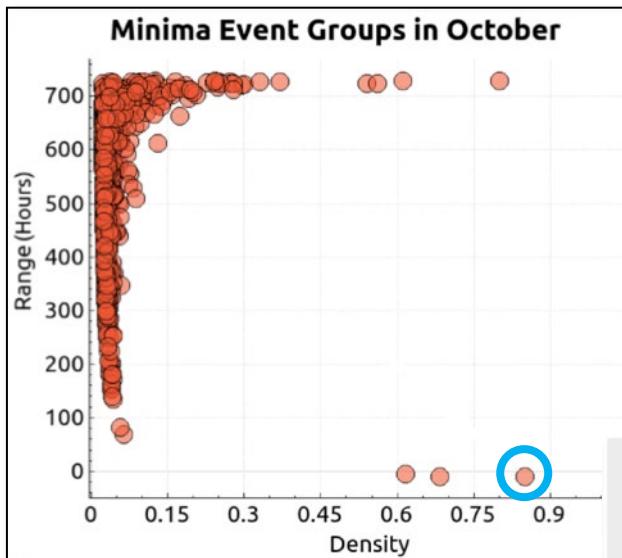


Event distribution view

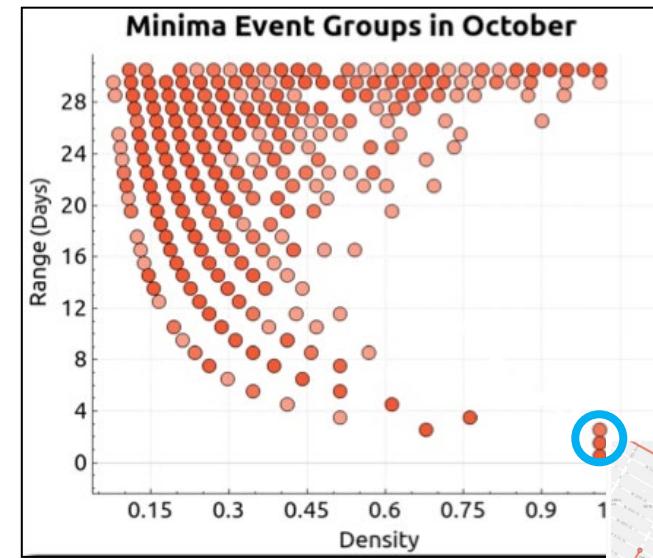
- Event group properties:
 - Range: time span of the group
 - Hours, days, weeks
 - Density: time span / no. of events
 - Frequency of events within a group



Rare events

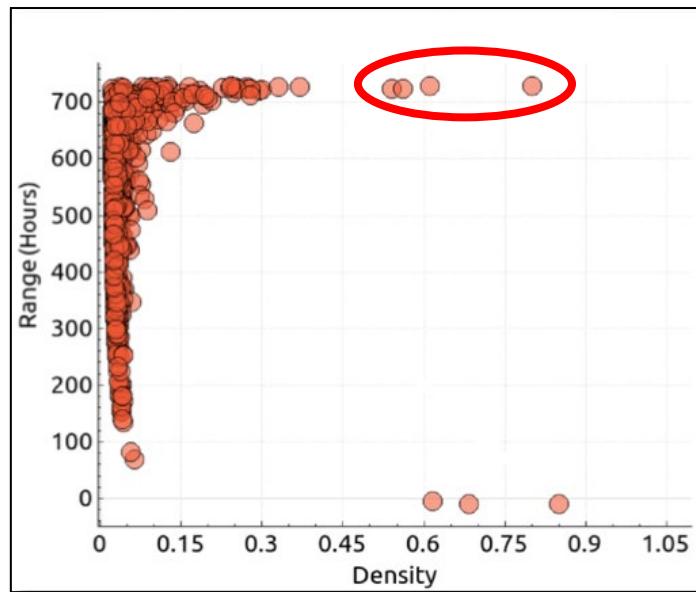


Hourly: Halloween parade



Frequent events

Maxima: taxi hotspots



Nighttime trends

Applications: Urban Pulse



Infrastructure

flickr

twitter

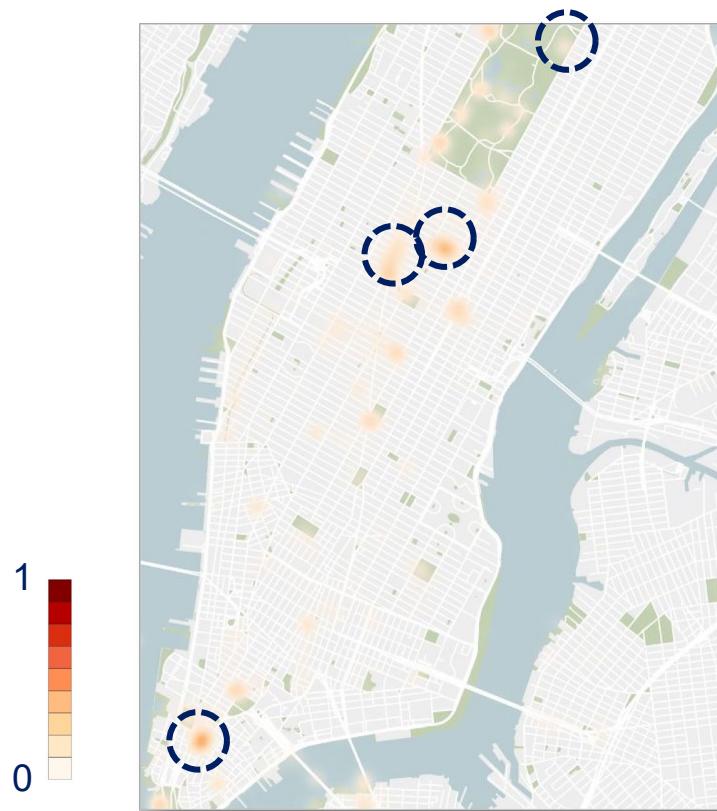


Environment

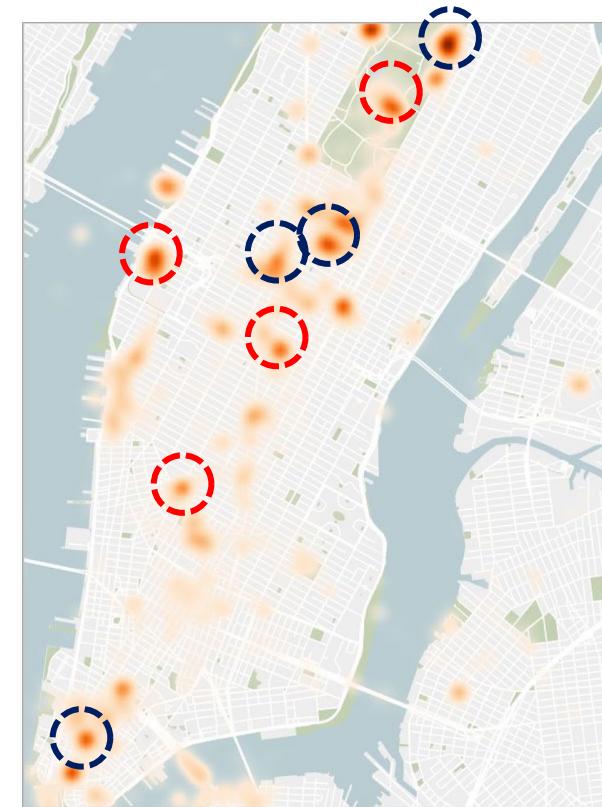
Social media

Urban Pulse

Flickr activity in New York City



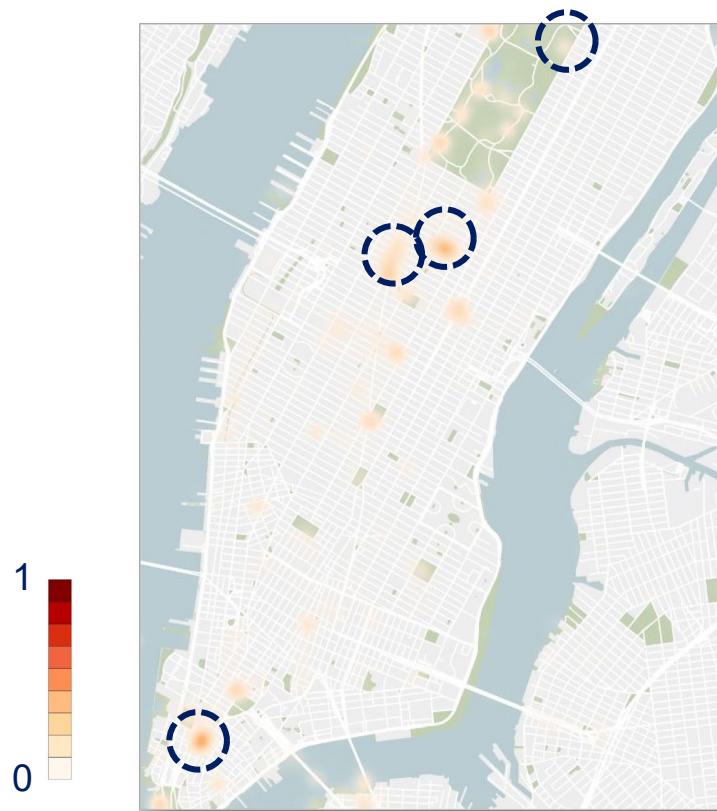
7:00 am



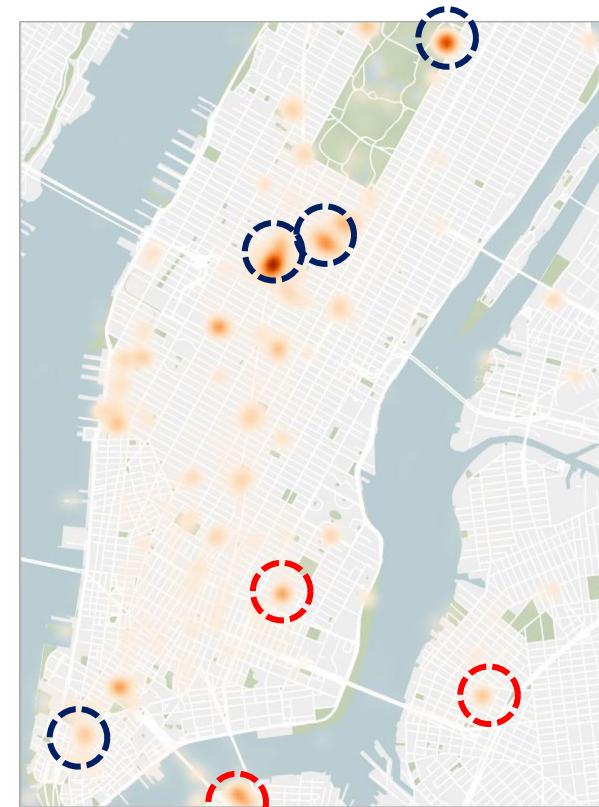
11:00 am

Urban Pulse

Flickr activity in New York City



7:00 am



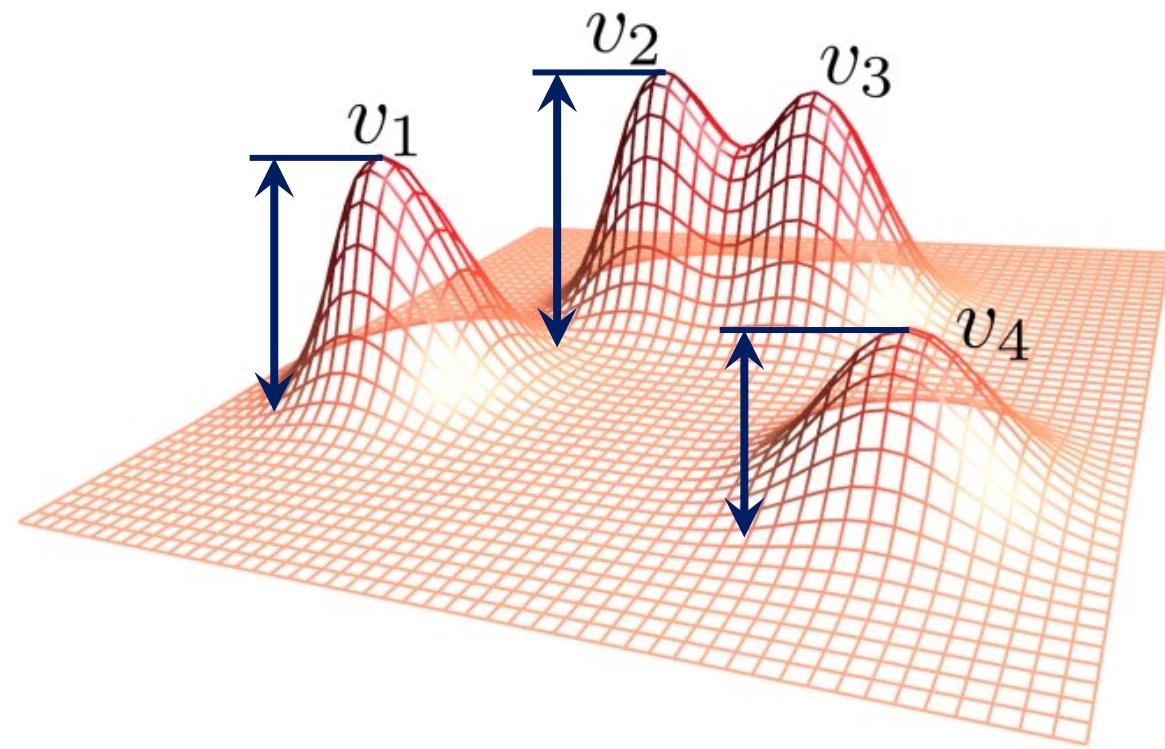
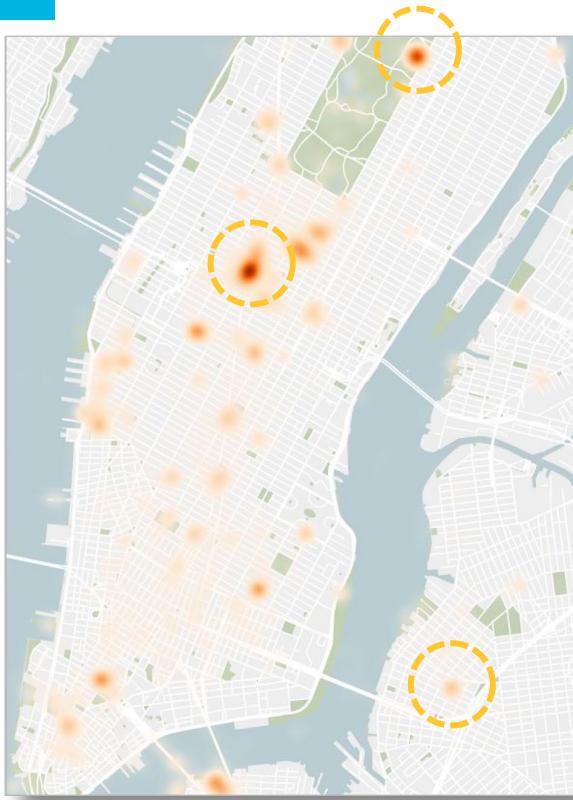
7:00 pm

Urban Pulse: desiderate

- Capture locations where the pulse is “interesting”
- Quantify the pulse
 - Track “activity”
- Temporal resolutions

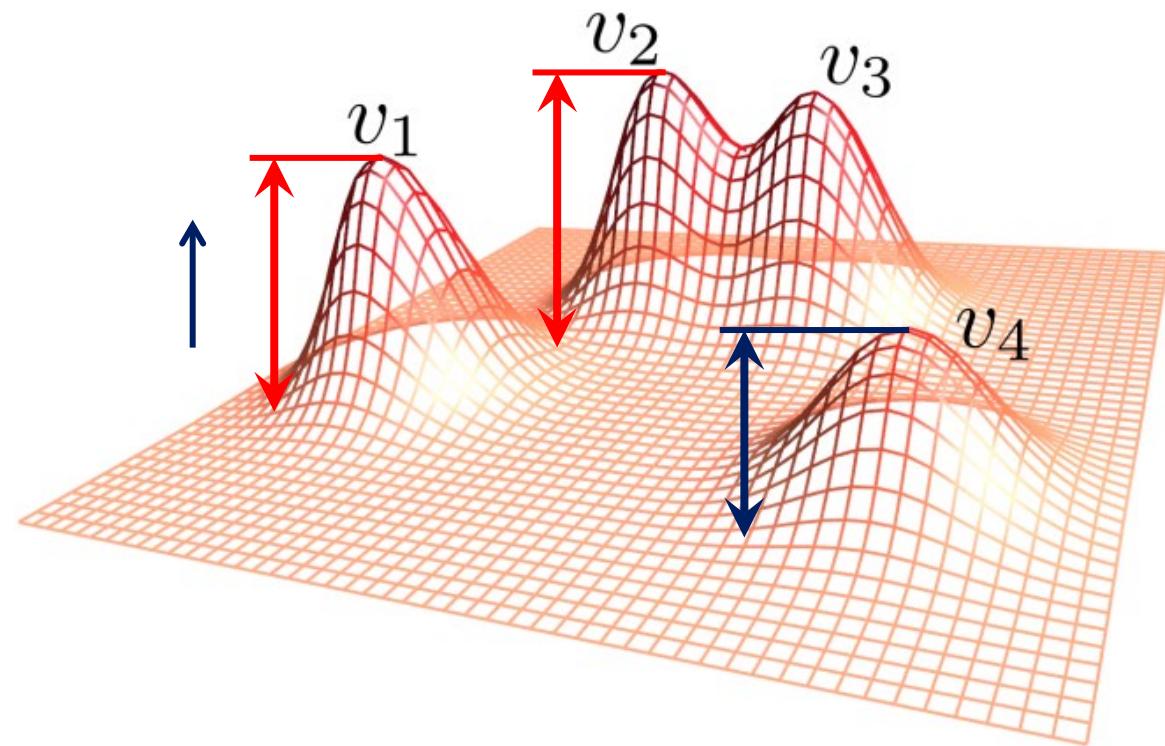
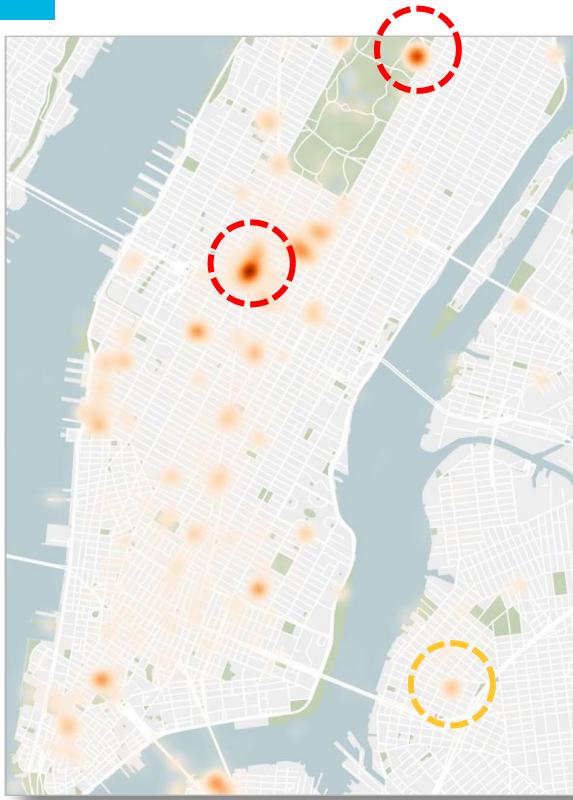
Step 1: identify pulse locations

1. Identify Locations
2. Quantify Pulse



Step 1: identify pulse locations

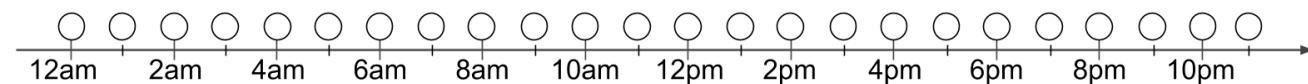
1. Identify Locations
2. Quantify Pulse



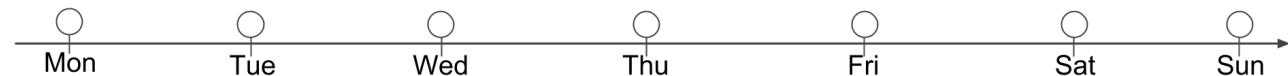
Handling temporal resolutions

Assume functions are defined along 3 resolutions (group by operations)

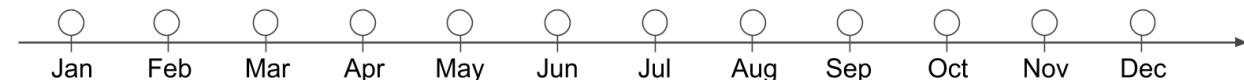
Time of Day



Day of Week



Month of Year



Step 1: identify pulse locations

- Set of scalar functions over time
 - Density functions as Gaussian weighted sum:

$$\sum_{x_i \in N(p)} e^{\frac{-d(p, x_i)^2}{\varepsilon^2}}$$

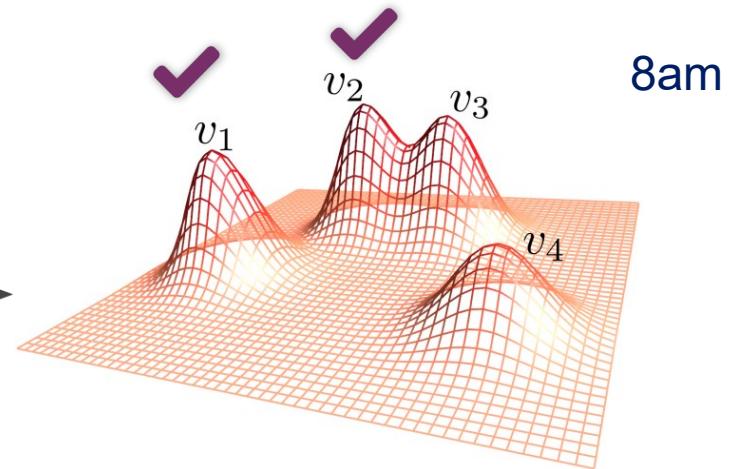
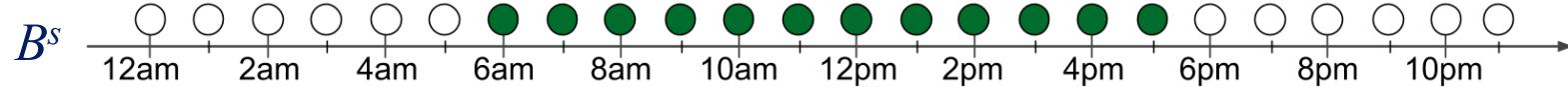
- Identify all maxima
- Location of **prominent** pulses:
 - is a high persistent maxima in at least one timestep
 - is a high persistent maxima in at least one resolution



Step 2: quantifying pulse

1. Identify Locations
2. Quantify Pulse

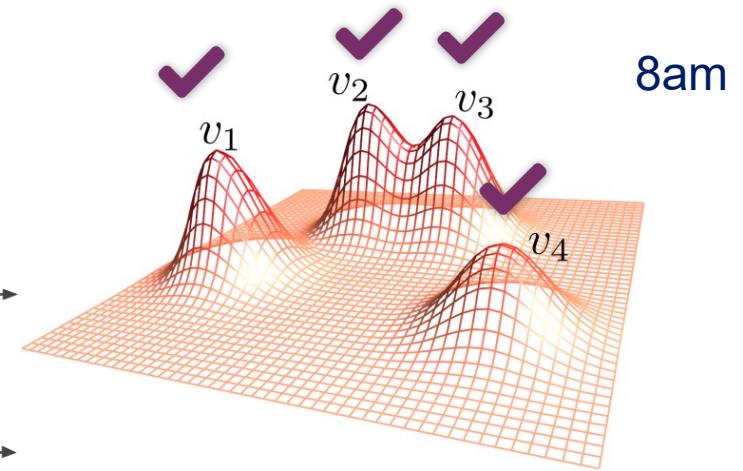
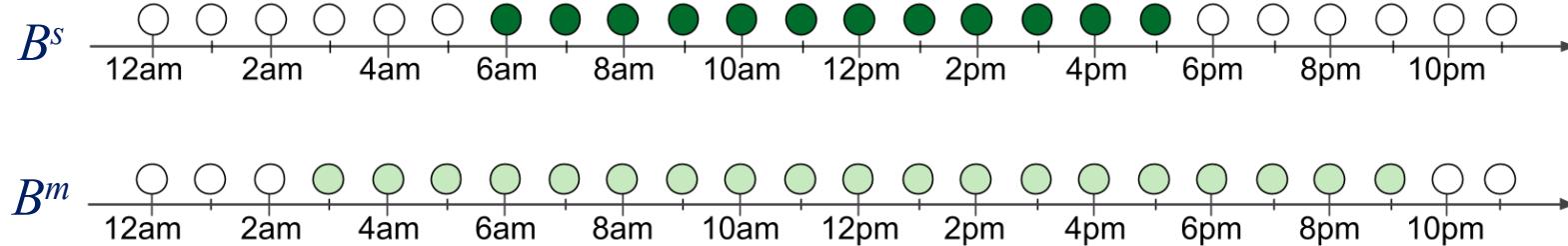
- 3 **Beats** to quantify the pulse at each location
- Significant Beats B^s
 - Pulse is a high persistent maximum



Step 2: quantifying pulse

1. Identify Locations
2. Quantify Pulse

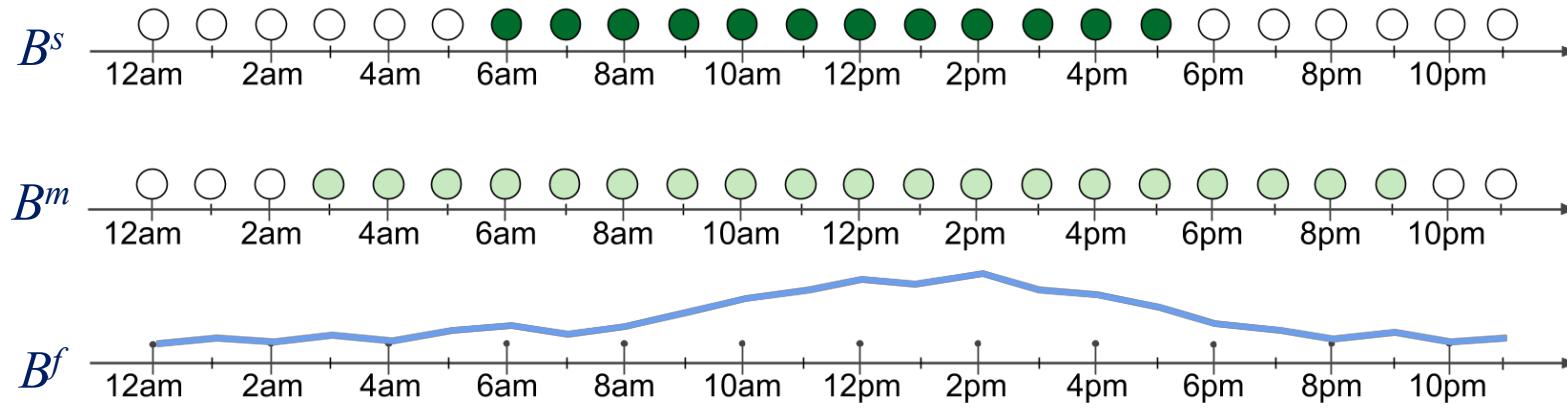
- 3 **Beats** to quantify the pulse at each location
- Maxima Beats B^m
 - Pulse is a maximum



Step 2: quantifying pulse

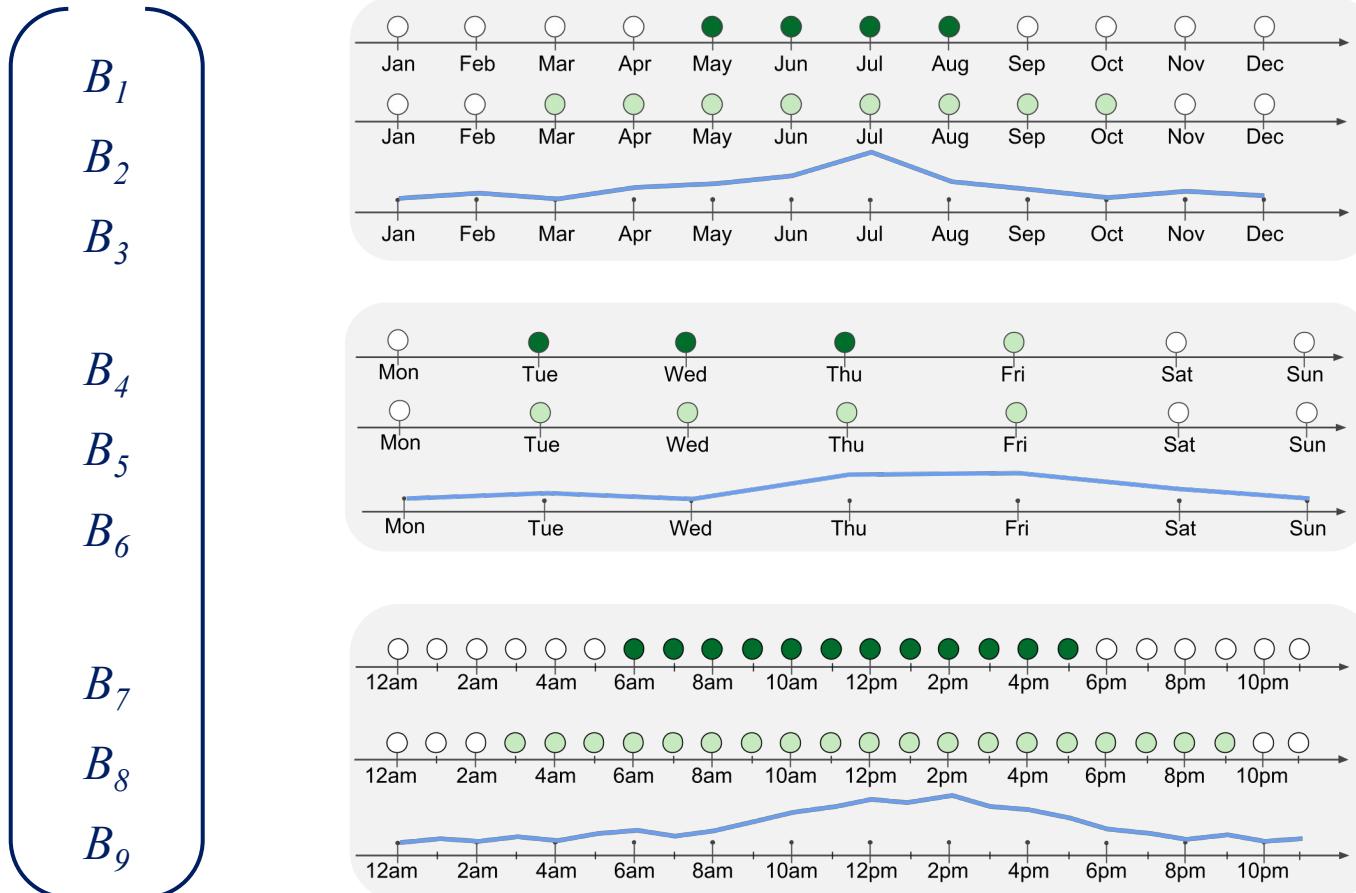
1. Identify Locations
2. Quantify Pulse

- 3 **Beats** to quantify the pulse at each location
- Function Beats B^f
 - Variation of the function values



Step 2: quantifying pulse

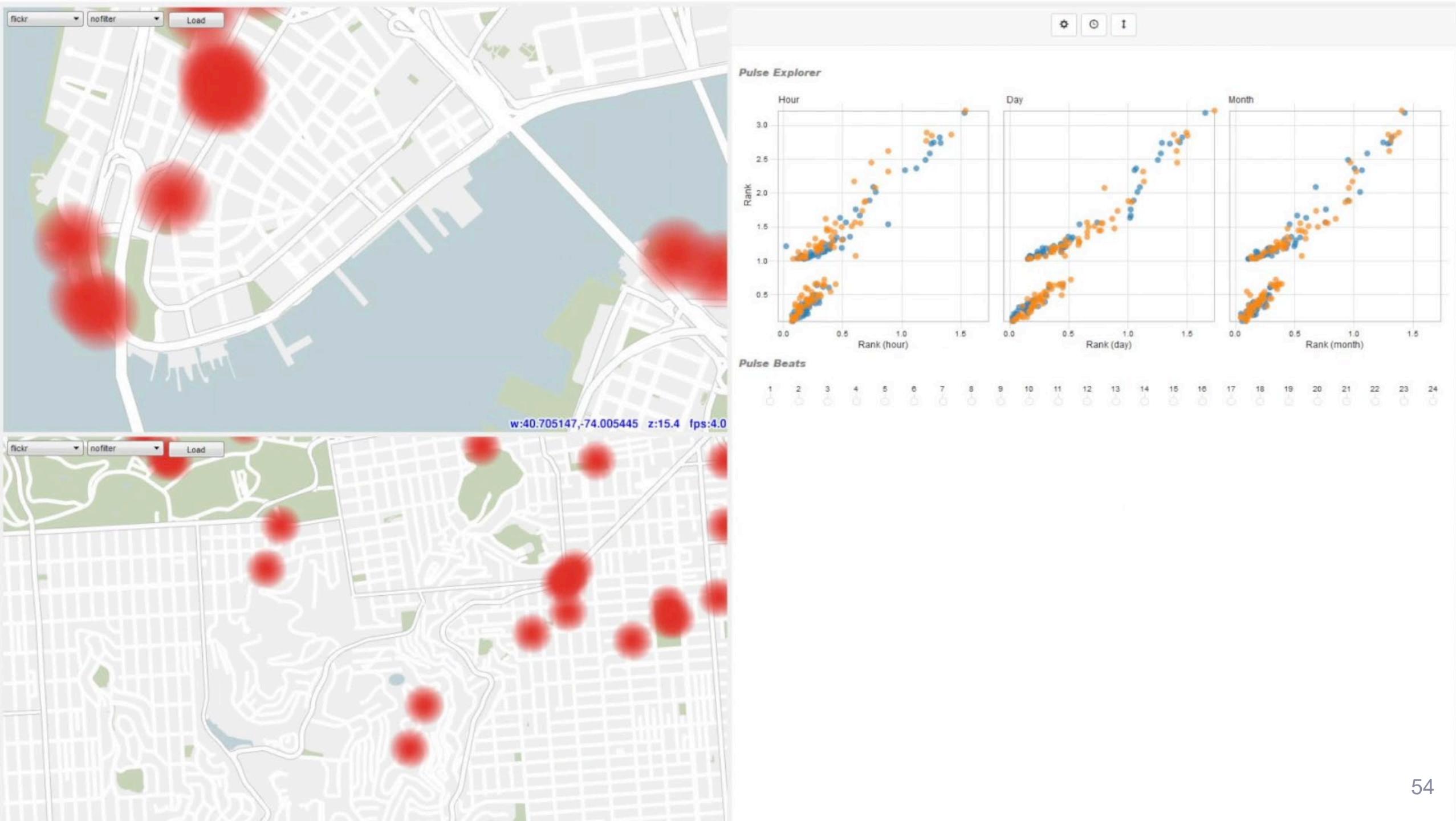
1. Identify Locations
2. Quantify Pulse



Signature

Data
Oblivious

Rank



Use cases

- Provided the interface to domain experts
- Architects from Kohn Pedersen Fox
 - Better understand design precedents
- Human behavioral expert
 - Try to understand the cohabitation between cultural communities
 - Twitter as proxy for cultural communities

Understanding Public Spaces

Rockefeller Center



Union Square



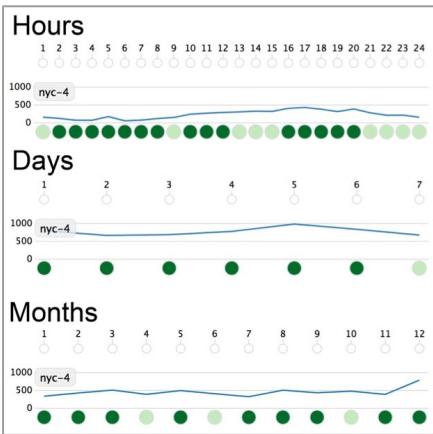
Bryant Park



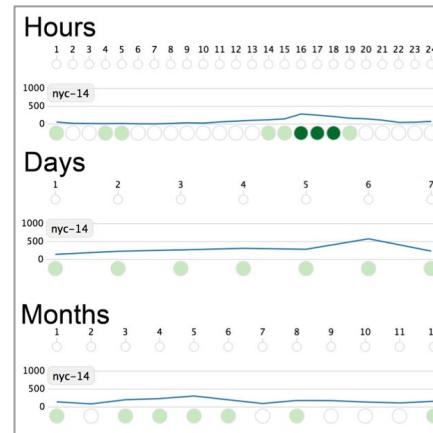
Typically classified together as being similar

Understanding public spaces

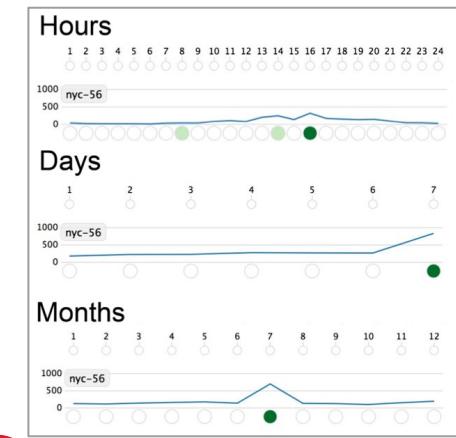
Rockefeller Center



Union Square

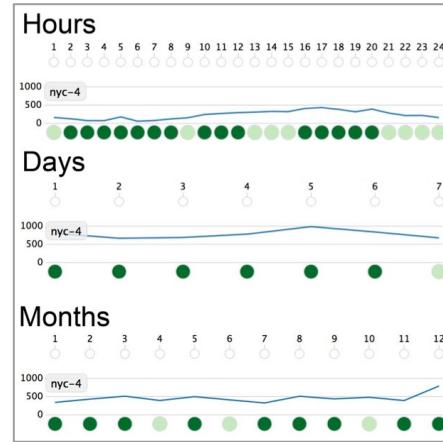


Bryant Park



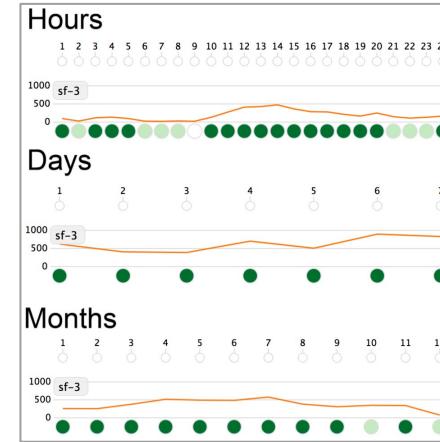
Understanding public spaces

Rockefeller Center



San Francisco

Alcatraz



<https://github.com/VIDA-NYU/urban-pulse>

Urban Pulse

Urban Pulse is a framework that uses computational topology techniques to capture the *pulse* of cities. This is accomplished by first modeling the urban data as a collection of time-varying scalar functions over different temporal resolutions, where the scalar function represents the distribution of the corresponding activity over the city. The topology of this collection is then used to identify the locations of prominent pulses in a city. The framework includes a visual interface that can be used to explore pulses within and across multiple cities.

The framework was first presented in the [paper](#):

Urban Pulse: Capturing the Rhythm of Cities
Fabio Miranda, Harish Doraiswamy, Marcos Lage, Kai Zhao, Bruno Gonçalves, Luc Wilson, Mondrian Hsieh and Cláudio T. Silva
IEEE Transactions on Visualization and Computer Graphics, 23 (1), 2017, 791-800.

The team includes:

- Fabio Miranda (New York University)
- Harish Doraiswamy (New York University)
- Marcos Lage (Fluminense Federal University)
- Bruno Gonçalves (New York University)
- Kai Zhao (Georgia State University)
- Luc Wilson, Mondrian Hsieh (Kohn Pedersen Fox)
- Cláudio T. Silva (New York University)

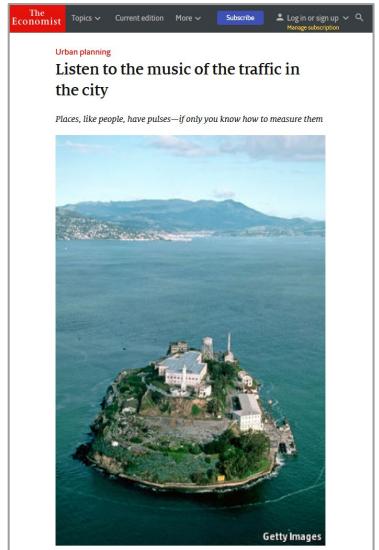
Urban Pulse has also been featured on [The Economist](#), [Architectural Digest](#), [Curbed](#) and [GCN](#).



Curbed



Architectural Digest



The Economist

Topological data analysis

- Naturally captures interesting features.
- Features can have arbitrary shapes.
- Very efficient.
- Robust to noise.
- Works on data in any dimension.

Scikit-TDA

- Python libraries for topological data analysis.

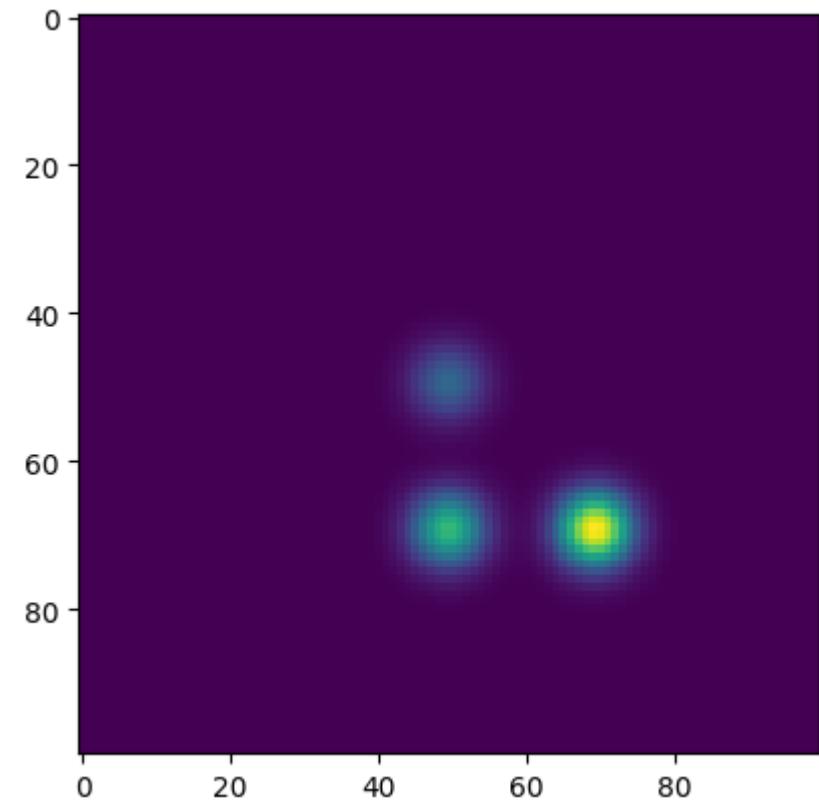
```
user@DESKTOP MINGW64 ~/  
$ pip install scikit-tda
```

<https://scikit-tda.org/>

Scikit-TDA: synthetic data example

Creating 2D synthetic data:

```
ts = np.linspace(-1, 1, 100)
x1 = np.exp(-ts**2/(0.1**2))
ts -= 0.4
x2 = np.exp(-ts**2/(0.1**2))
scalar = x1[None, :]*x1[:, None] + 2*x1[None, :]*x2[:, None] + 3*x2[None, :]*x2[:, None]
plt.imshow(scalar)
plt.show()
```



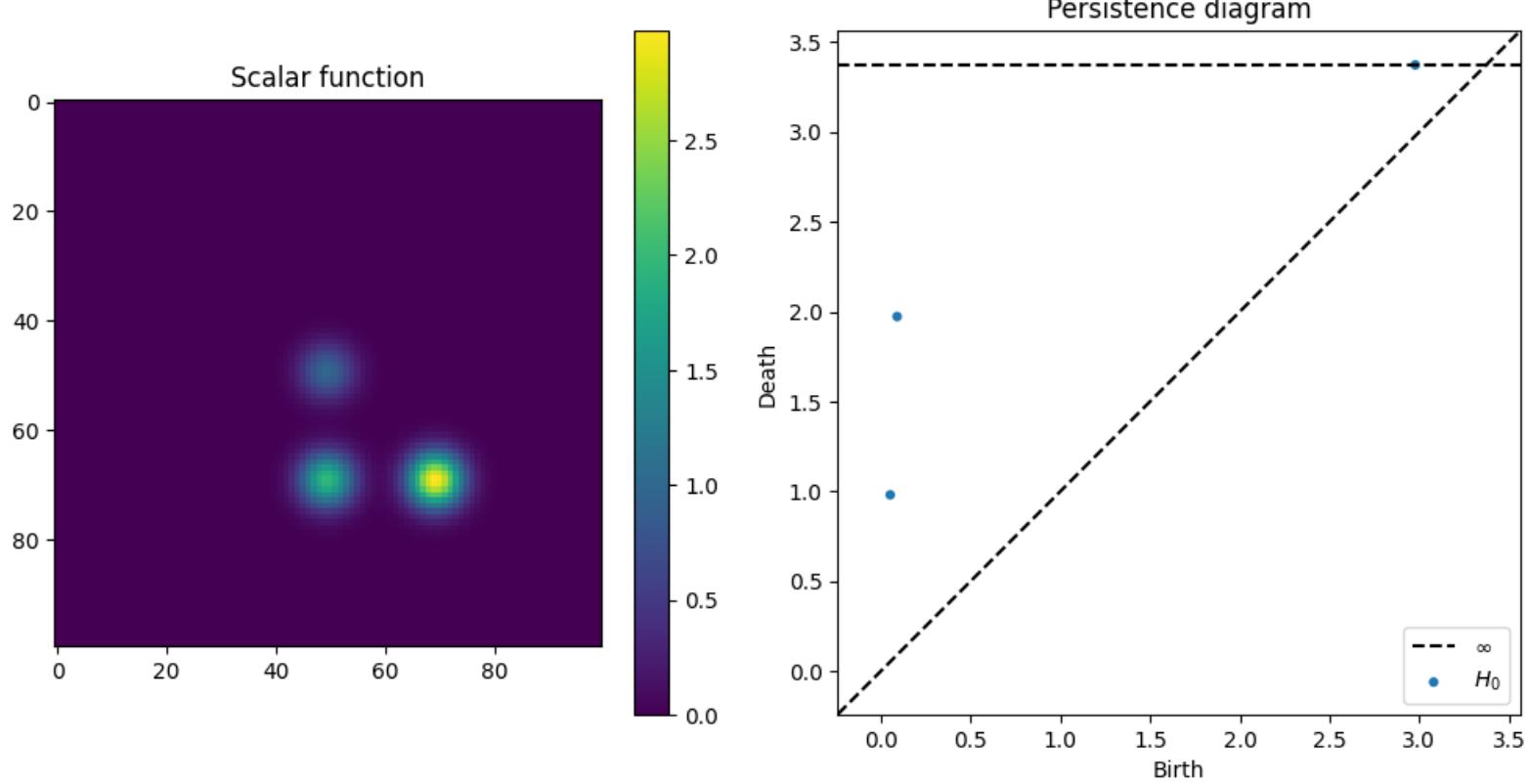
Scikit-TDA: synthetic data example

Computing persistence diagram:

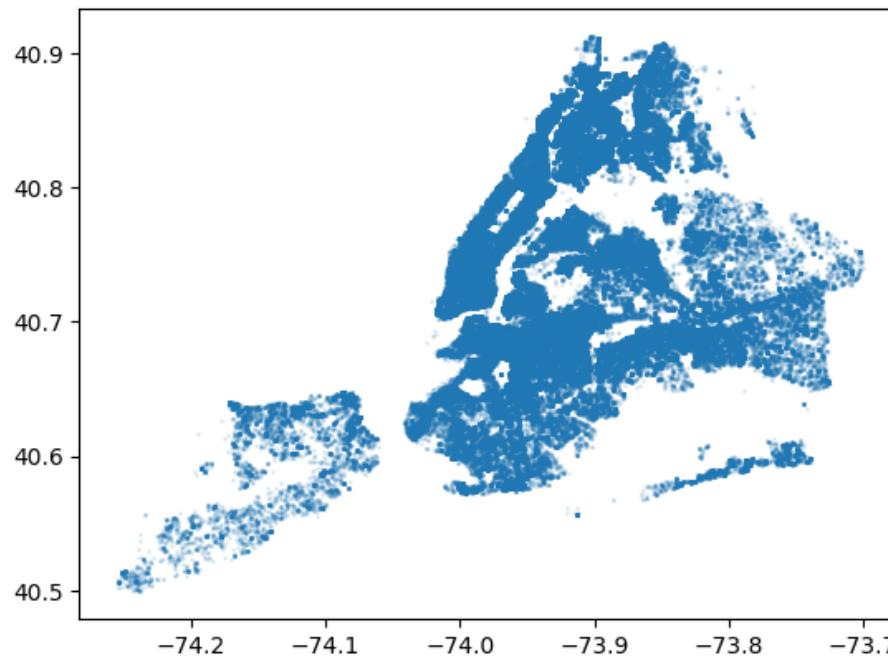
```
dgm = lower_star_img(-scalar)
dgm[~np.isinf(dgm)] = -dgm[~np.isinf(dgm)]
dgm[~np.isinf(dgm).any(axis=1)] = np.flip(dgm[~np.isinf(dgm).any(axis=1)])

plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(img)
plt.colorbar()
plt.title("Scalar function")
plt.subplot(122)
plot_diagrams(dgm)
plt.title("Persistence diagram")
plt.tight_layout()
plt.show()
```

Scikit-TDA: synthetic data example



Scikit-TDA: spatiotemporal data example



Spatiotemporal dataset with 311 noise complaints

Scikit-TDA: spatiotemporal data example

Creating 2D scalar function with KDE:

```
df = pd.read_pickle('data/311_2019.pkl.gz')

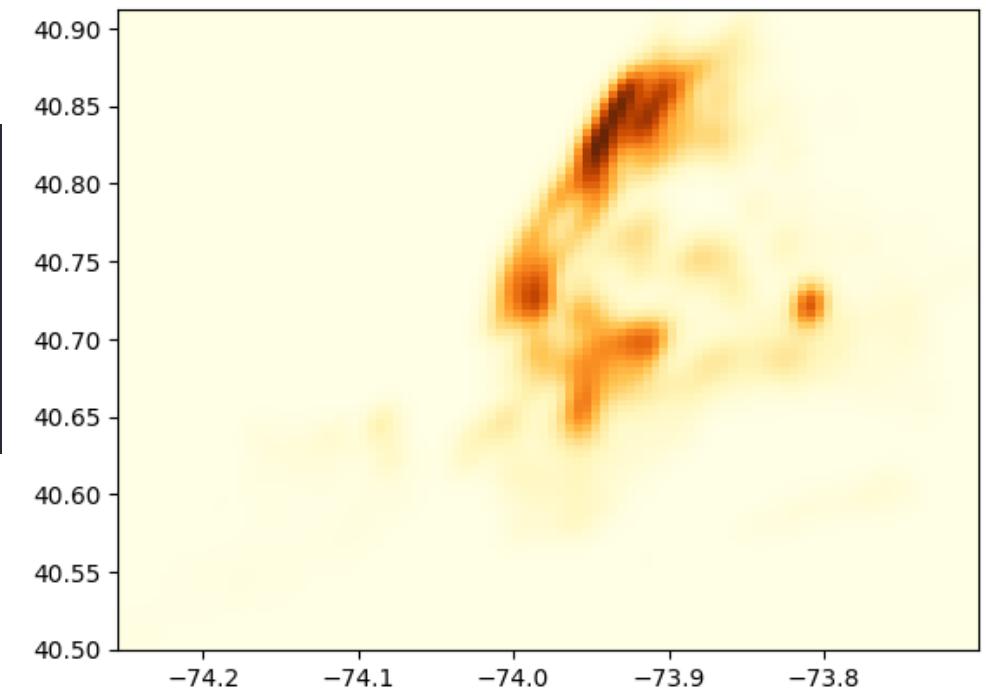
m1 = df[['latitude','longitude']].to_numpy()[:,1]
m2 = df[['latitude','longitude']].to_numpy()[:,0]
xmin = m1.min()
xmax = m1.max()
ymin = m2.min()
ymax = m2.max()

X, Y = np.mgrid[xmin:xmax:100j, ymin:ymax:100j]
positions = np.vstack([X.ravel(), Y.ravel()])
values = np.vstack([m1, m2])
kernel = stats.gaussian_kde(values)
Z = np.reshape(kernel(positions).T, X.shape)
```

Scikit-TDA: spatiotemporal data example

Plotting 2D scalar function:

```
fig, ax = plt.subplots()
ax.imshow(np.rot90(Z), cmap=plt.cm.YlOrBr, extent=[x
min, xmax, ymin, ymax])
ax.set_xlim([xmin, xmax])
ax.set_ylim([ymin, ymax])
plt.show()
```



Scikit-TDA: spatiotemporal data example

Computing persistence diagram:

```
dgm = lower_star_img(-Z)
dgm[~np.isinf(dgm)] = -dgm[~np.isinf(dgm)]
dgm[~np.isinf(dgm).any(axis=1)] = np.flip(dgm[~np.isinf(dgm).any(axis=1)])

plt.figure(figsize=(10, 5))
plt.subplot(121)
plt.imshow(np.rot90(Z), cmap=plt.cm.YlOrBr, extent=[xmin, xmax, ymin, ymax])
plt.colorbar()
plt.title("Scalar function")
plt.subplot(122)
plot_diagrams(dgm)
plt.title("Persistence diagram")
plt.tight_layout()
plt.show()
```

Scikit-TDA: spatiotemporal data example

