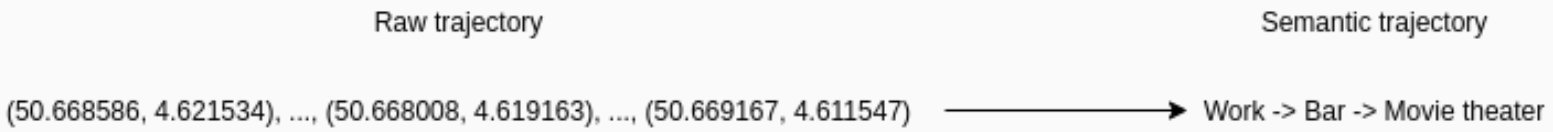
Mining Constrained Regions of Interest: An optimization approach

Alexandre Dubray Guillaume Derval Siegfried Nijssen Pierre Schaus

Introduction

Motivations

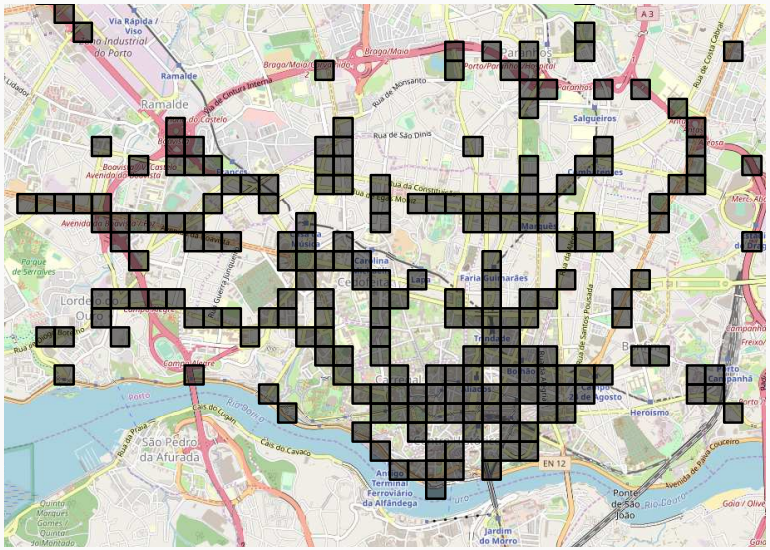
- The amount of spatiotemporal data is exploding (smartphone applications, sports devices, fleet management, etc.)
- There is a need to process more efficiently these data
- We can do that with *Semantic Trajectories*
- We can reason about semantic trajectories, and thus more easily extract knowledge



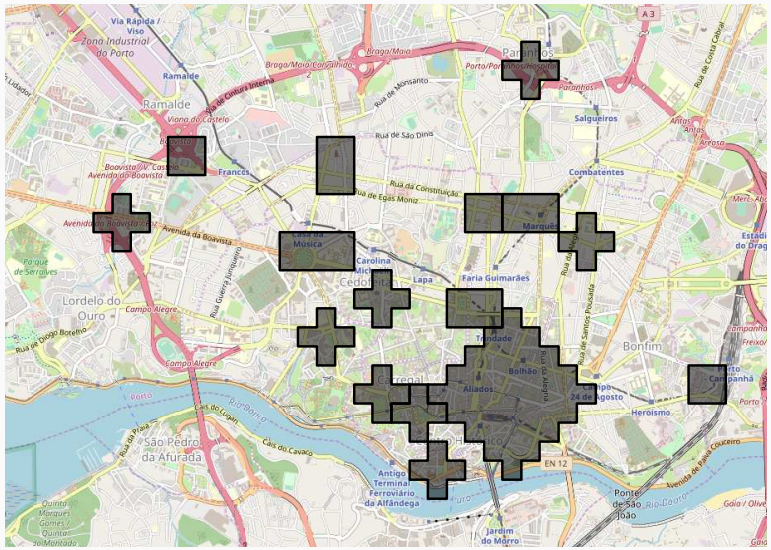
The general approach

1. Divide the map with a $N \times N$ grid.
2. Assign a density value to each cell. A cell is **dense** if its density is above a **threshold**. Typical density function is the number of crossing trajectories.
3. Express the ROI as an aggregation of dense cells

Example of ROIs



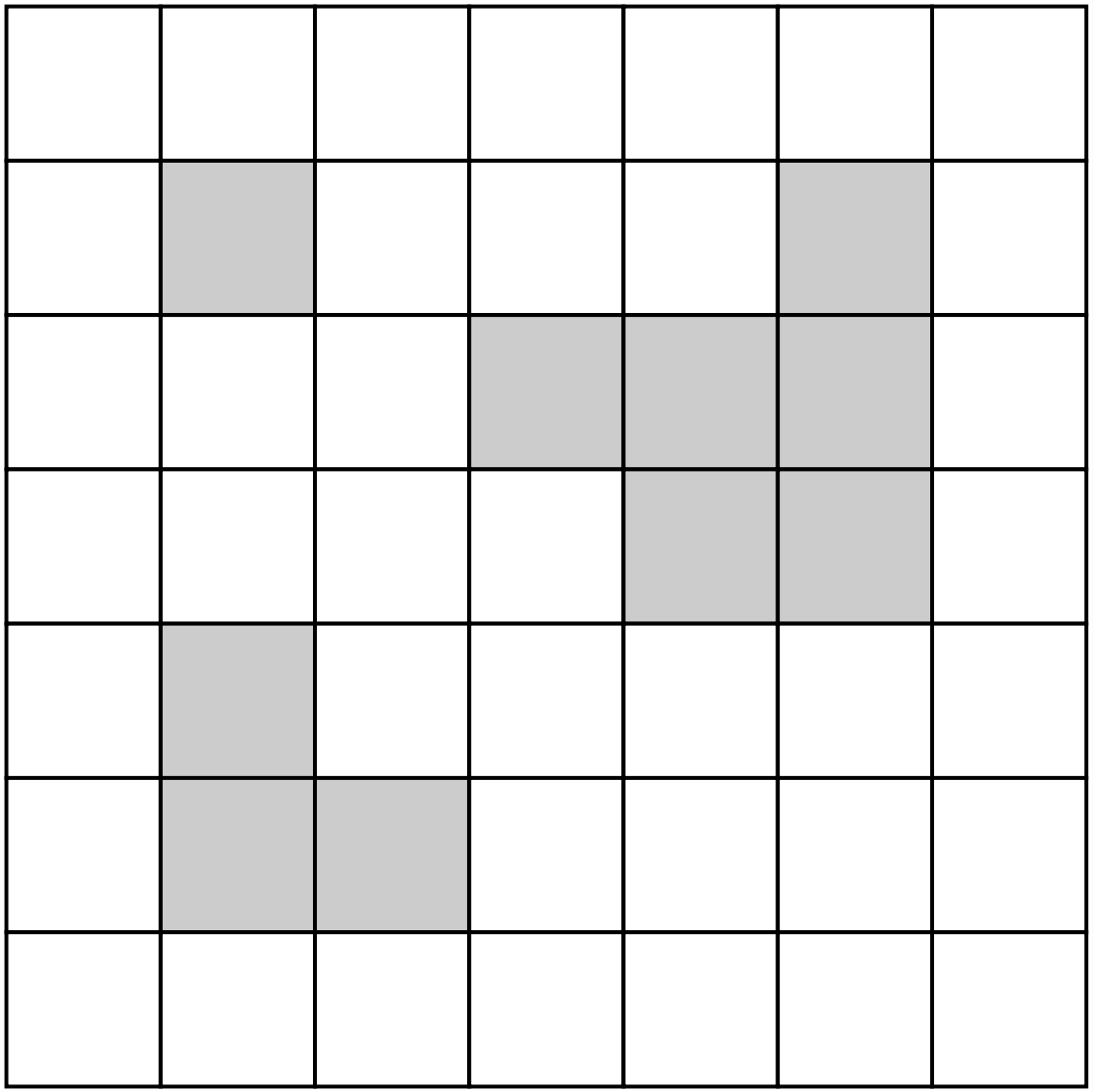
(a) Initial set of dense cells



(b) Solution found by our method

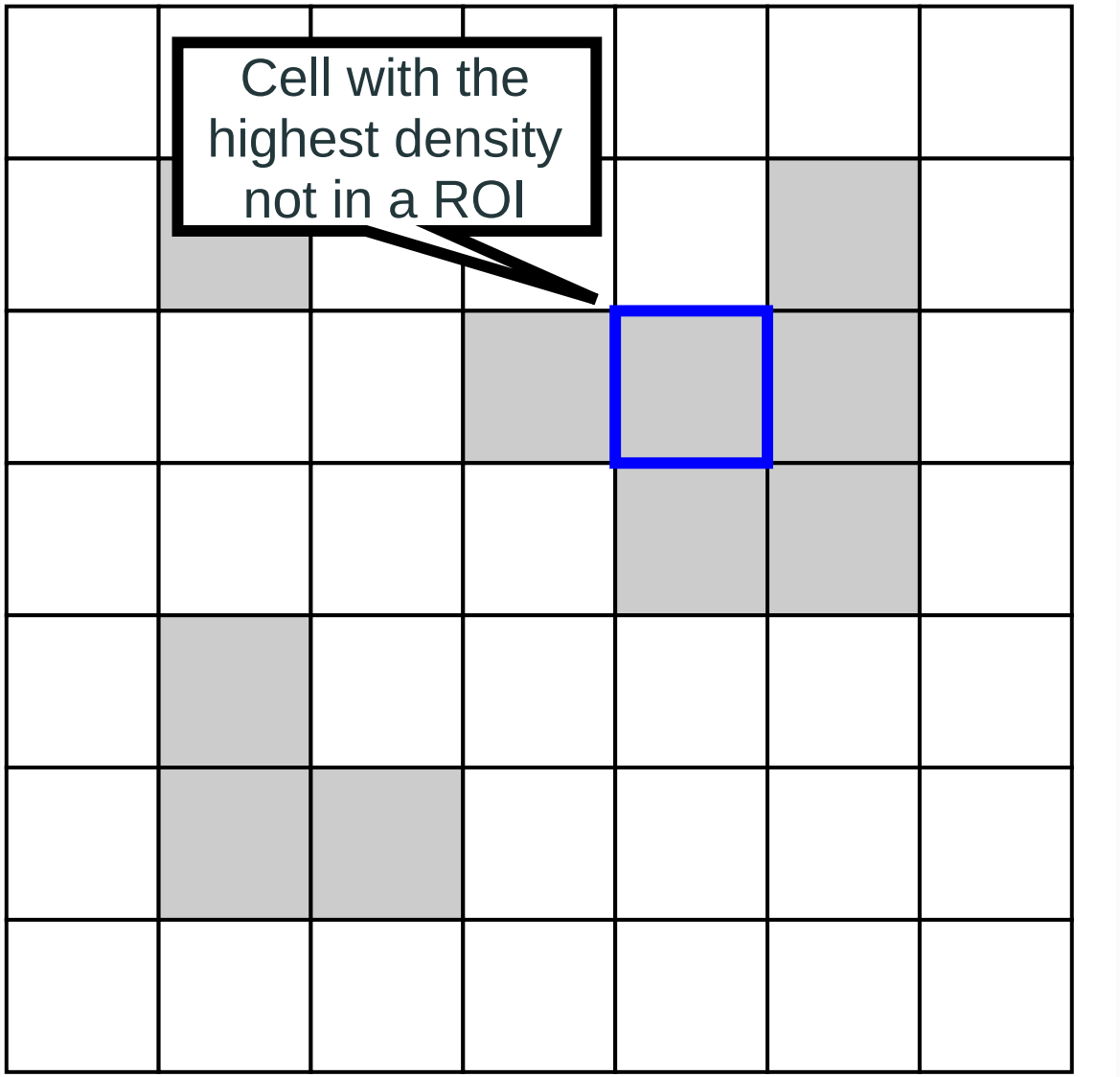
PopularRegion

Execution of the algorithm



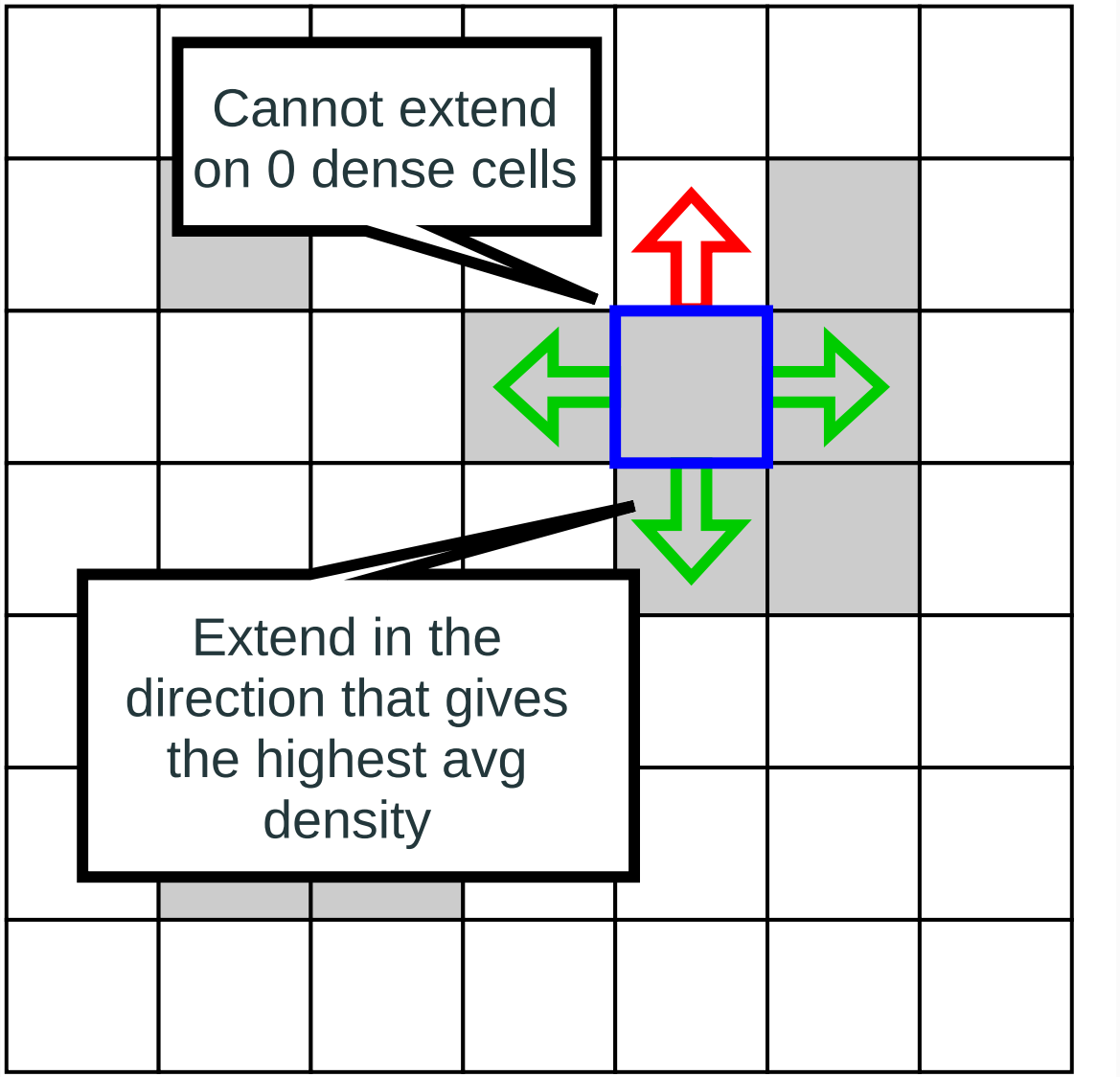
Fosca Giannotti et al. “Trajectory pattern mining”. In: *SIGKDD*. 2007

Execution of the algorithm



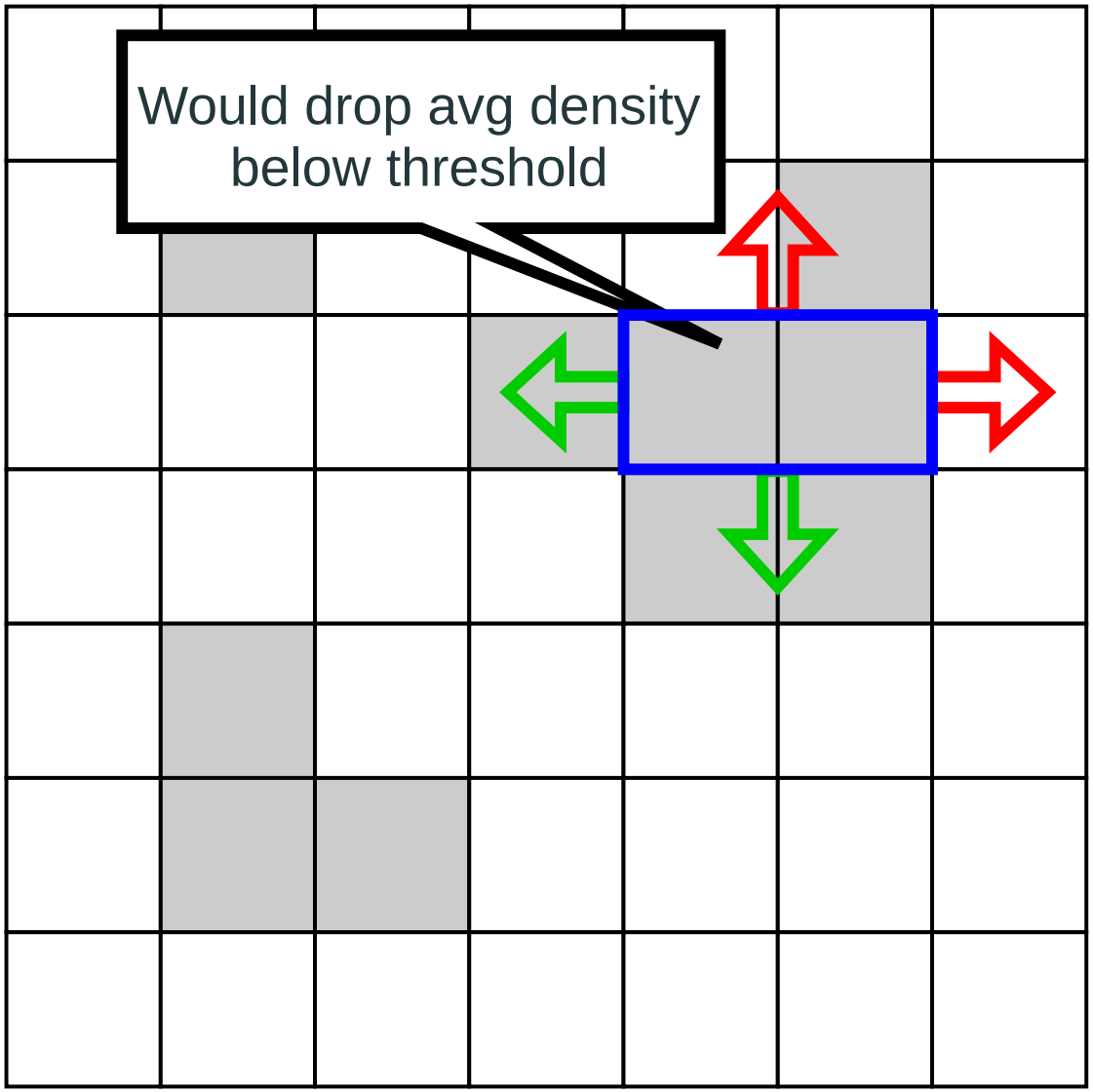
Fosca Giannotti et al. “Trajectory pattern mining”. In: *SIGKDD*. 2007

Execution of the algorithm



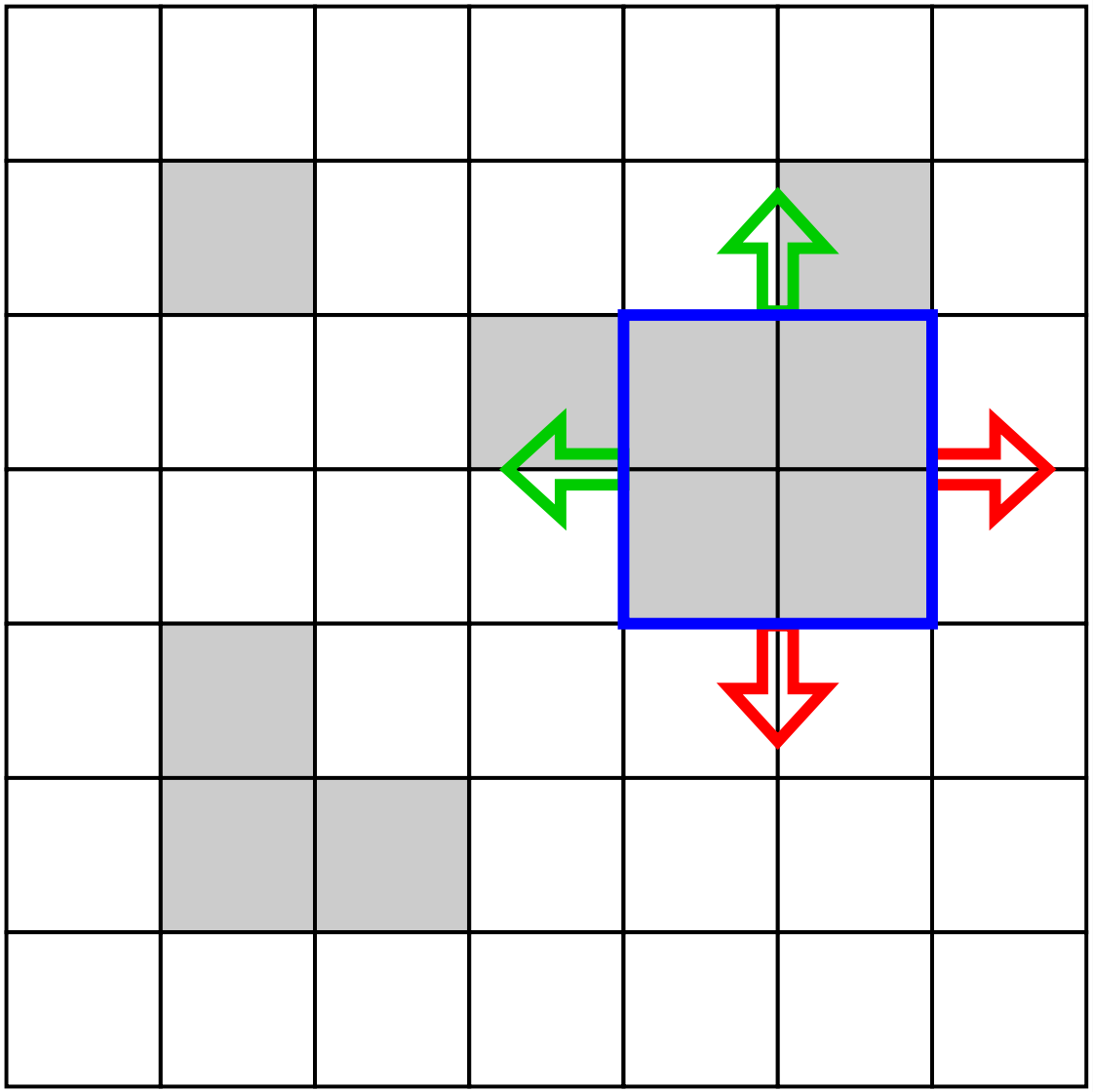
Fosca Giannotti et al. "Trajectory pattern mining". In: *SIGKDD*. 2007

Execution of the algorithm



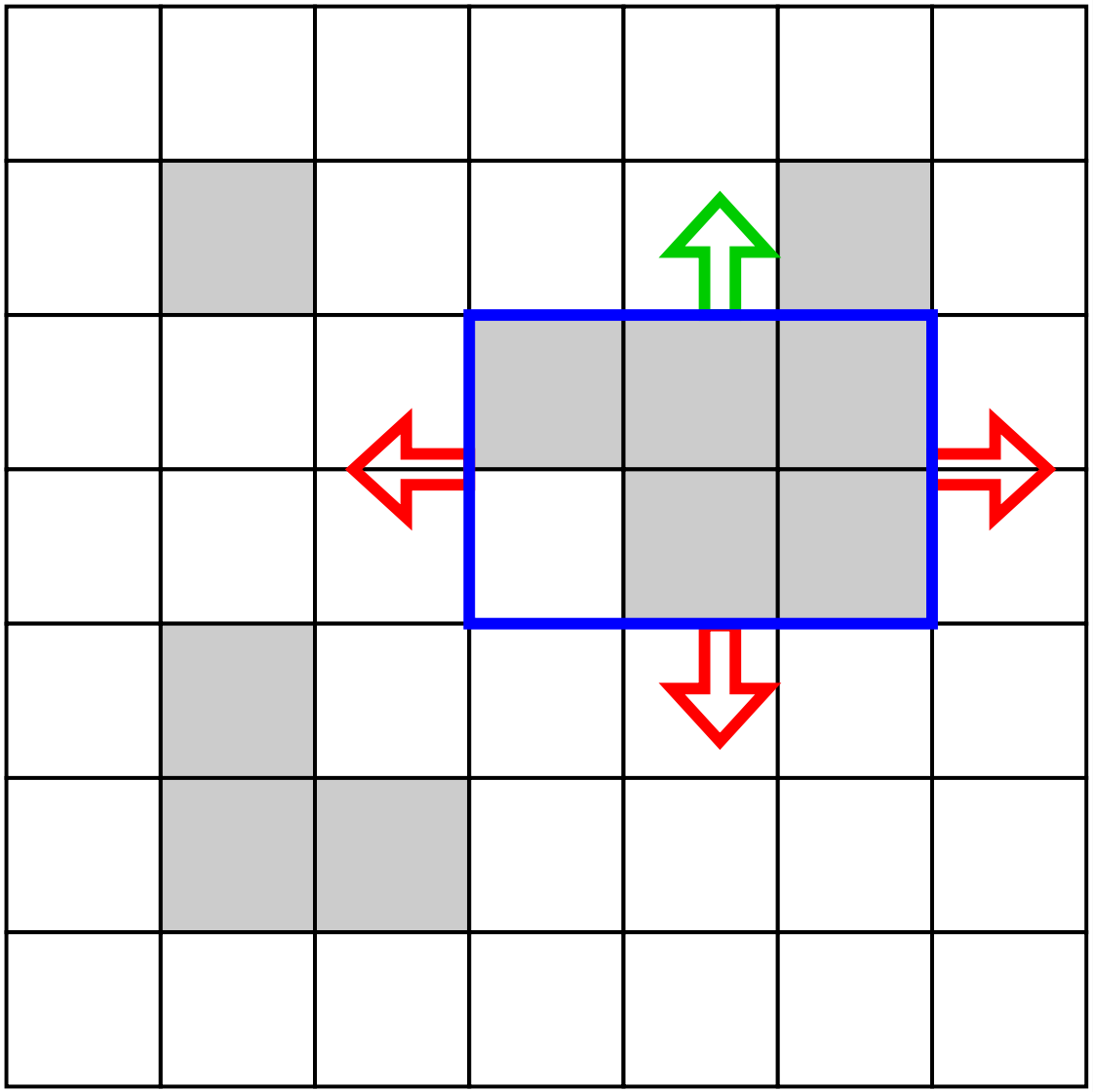
Fosca Giannotti et al. “Trajectory pattern mining”. In: *SIGKDD*. 2007

Execution of the algorithm



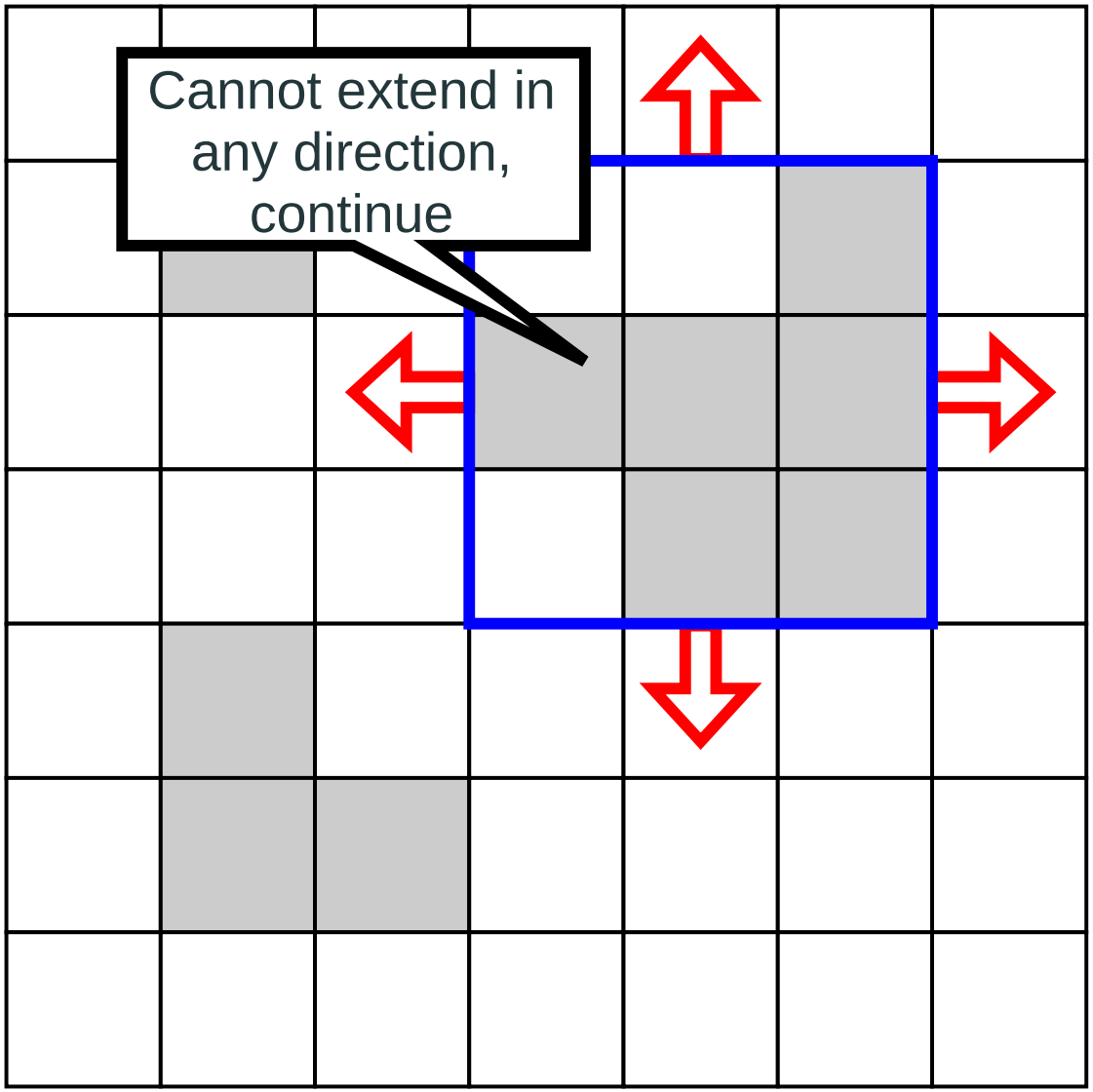
Fosca Giannotti et al. “Trajectory pattern mining”. In: *SIGKDD*. 2007

Execution of the algorithm



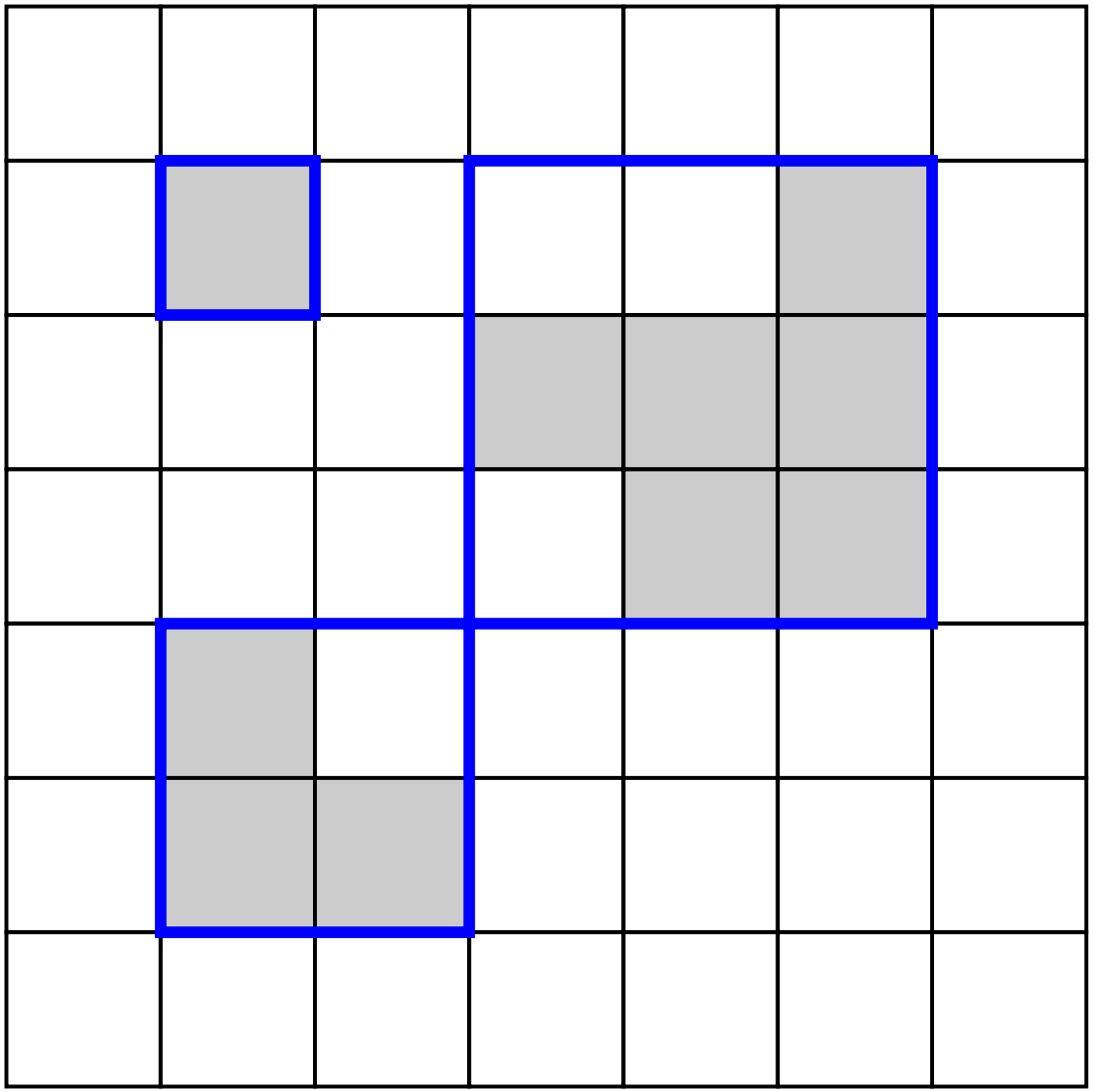
Fosca Giannotti et al. “Trajectory pattern mining”. In: *SIGKDD*. 2007

Execution of the algorithm



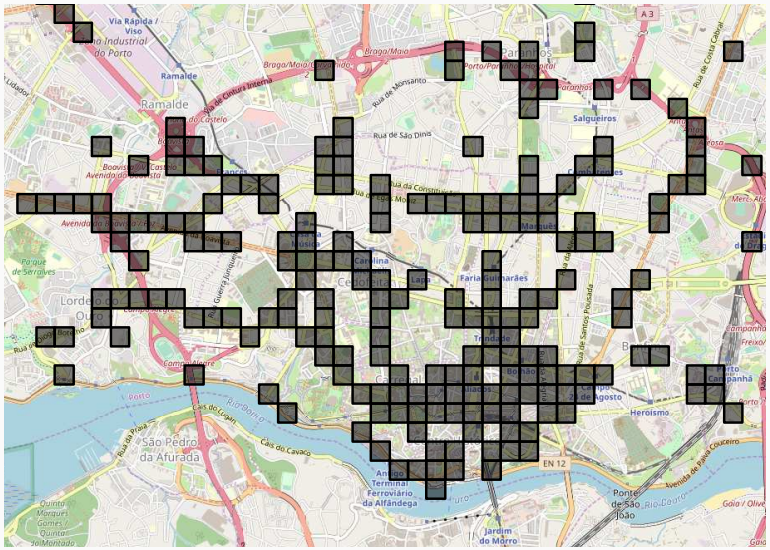
Fosca Giannotti et al. "Trajectory pattern mining". In: *SIGKDD*. 2007

Execution of the algorithm

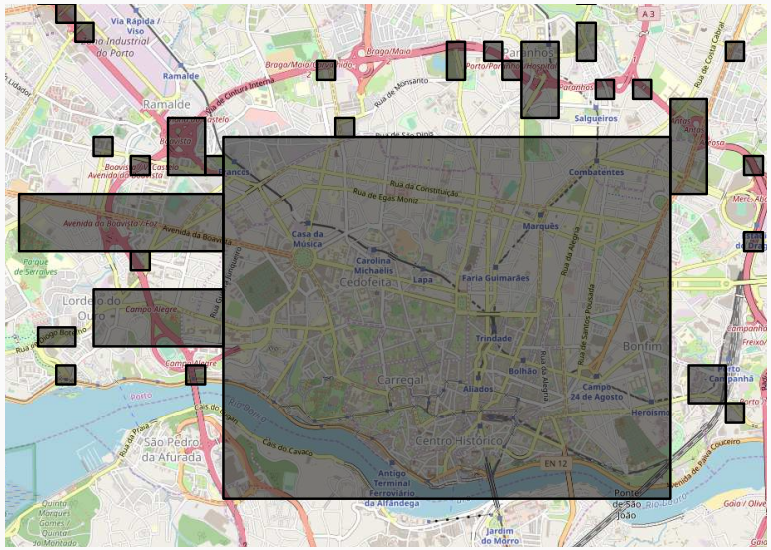


Fosca Giannotti et al. “Trajectory pattern mining”. In: *SIGKDD*. 2007

Result of the algorithm



(a) Initial set of dense cells



(b) Solution with 5% min average density

Advantages and disadvantages

- Scalable
- Intuitive and good results for most configurations

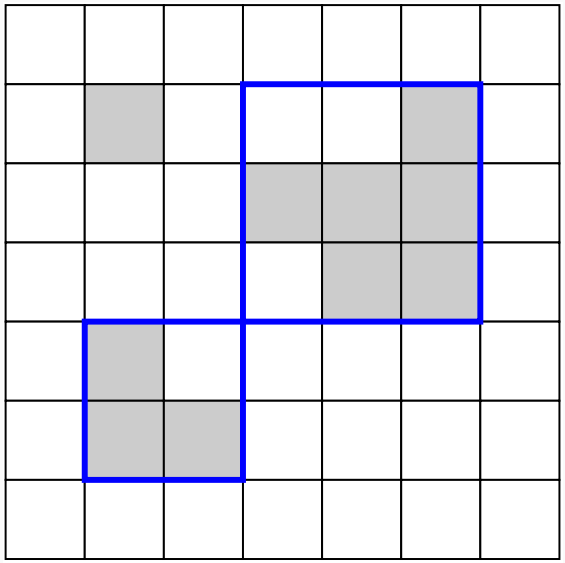
But...

- No formalization of the output
- Only rectangular regions
- Does not easily accept background knowledge
- Easy to create pathological input

Our method

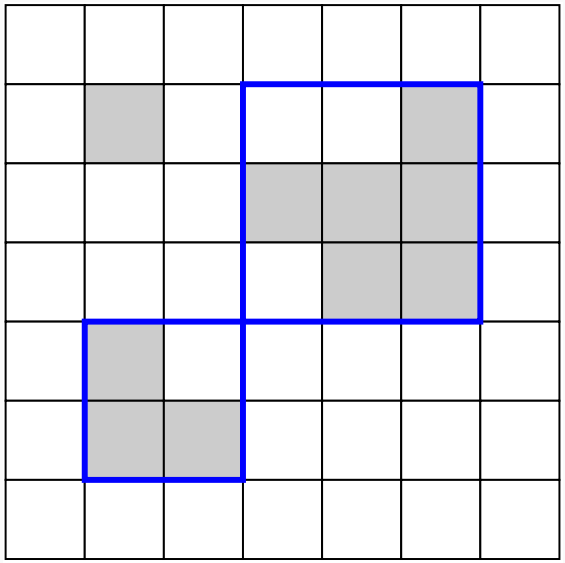
ROIs as an encoder

- The ROIs encode the dense status of the cells
- Example of encoding with two rectangles (we kept the non-overlap)



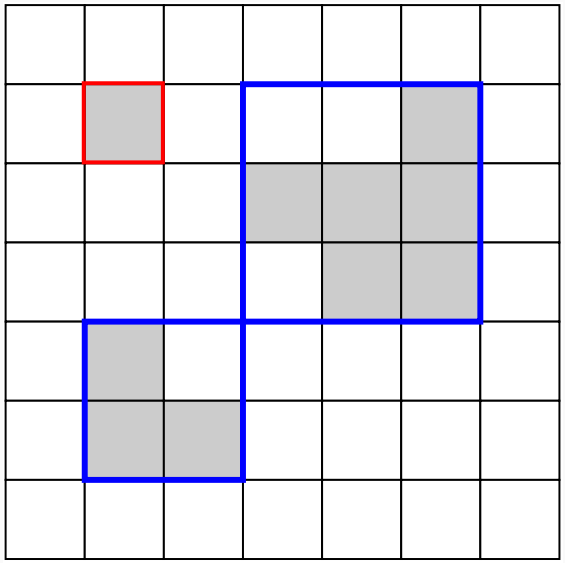
ROIs as an encoder

- The ROIs encode the dense status of the cells
- Example of encoding with two rectangles (we kept the non-overlap)
- The encoding makes 5 errors



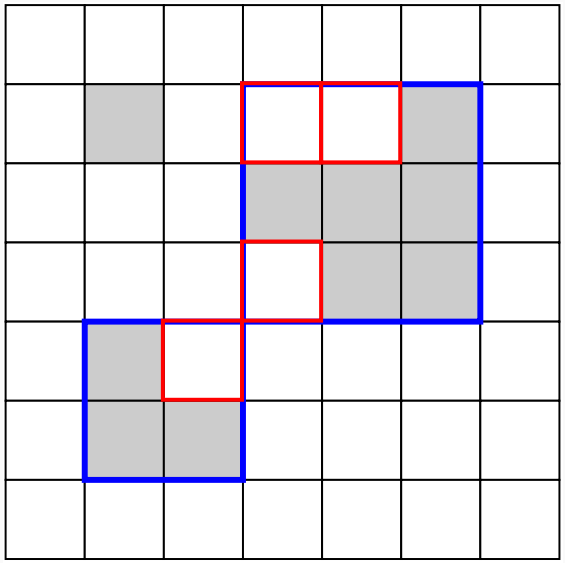
ROIs as an encoder

- The ROIs encode the dense status of the cells
- Example of encoding with two rectangles (we kept the non-overlap)
- The encoding makes 5 errors
  - 1 dense cells is not covered



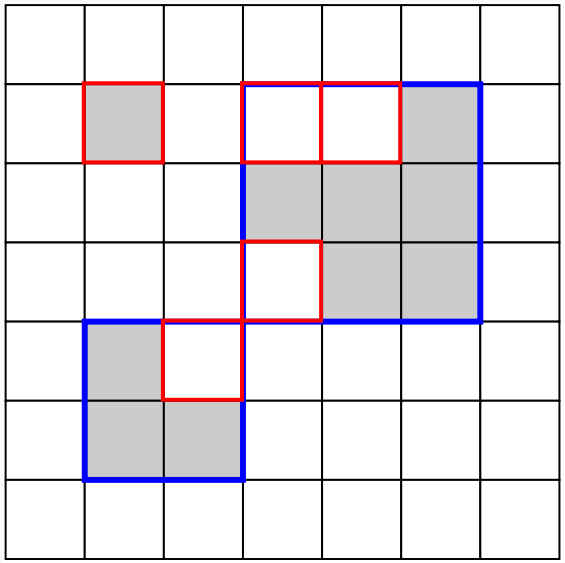
ROIs as an encoder

- The ROIs encode the dense status of the cells
- Example of encoding with two rectangles (we kept the non-overlap)
- The encoding makes 5 errors
  - 1 dense cells is not covered
  - 4 non-dense cells are covered



ROIs as an encoder

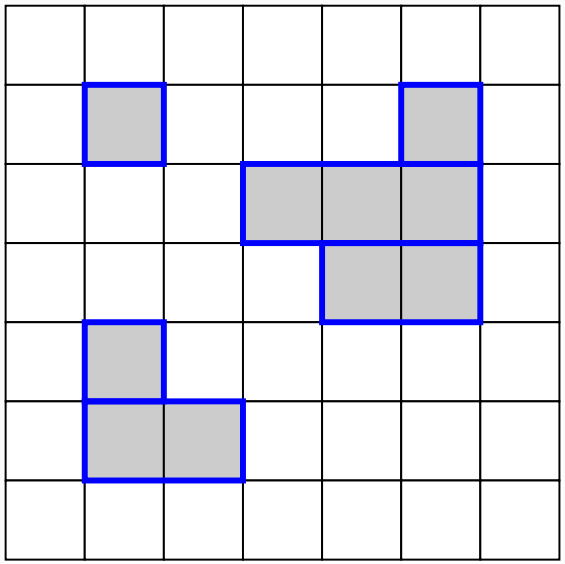
- The ROIs encode the dense status of the cells
- Example of encoding with two rectangles (we kept the non-overlap)
- The encoding makes 5 errors
  - 1 dense cells is not covered
  - 4 non-dense cells are covered
- We prefer encoding with fewer errors



Complexity of the models

We want to minimize the number of errors, but what about the complexity of the model?

- This model make no error but it requires 6 rectangles
- It does not represent well the dense cells
- We should limit the number of ROI to avoid these cases, but how to set the limit?



MDL Principle

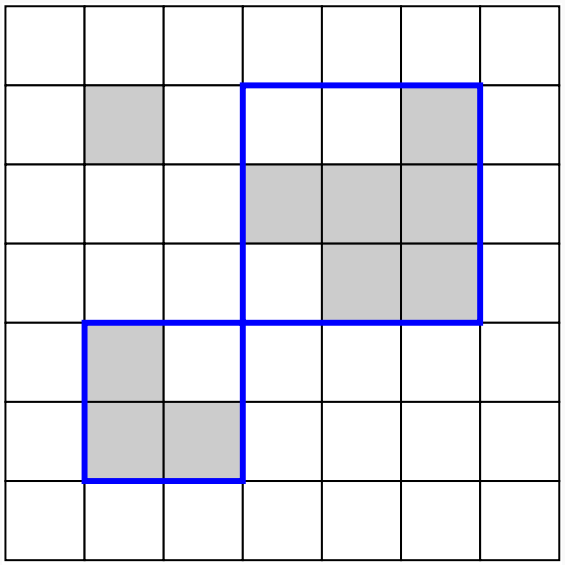
- The Minimum Description Length (MDL) principle is a formalization of Ocam's razor
- The best hypothesis is the ones that compresses the most the data
- It is a two stages encoding:
  - Encode a model with length $L(M)$
  - Encode the data D given the model M with length $L(D \mid M)$
  - Best model is $\arg\min_M L(D \mid M) + L(M)$
- Trade-off between complexity of the model and generalization of the data

MDL for the ROIs

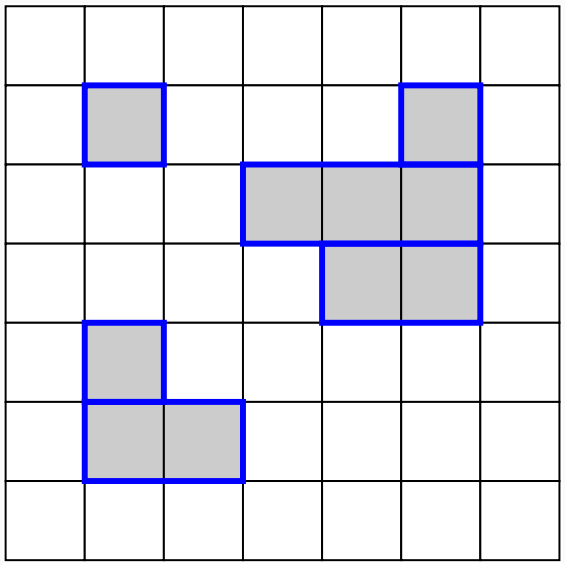
- Each cell is encoded with 2 integers (its row and its column)
- The length of a model, $L(M)$, is the sum of the length of the ROIs
  - A rectangle is encoded with two cells (4 integers)
  - A circle is encoded with one cell and a radius (3 integers)
  - Other forms have other encoding
- The length of data given a model, $L(D \mid M)$, is two times the number of errors of the model

MDL example

- $L(M) = 4 \cdot 2 = 8$
- $L(D \mid M) = 2 \cdot (4 + 1) = 10$
- $L(M) + L(D \mid M) = 18$



- $L(M) = 4 \cdot 6 = 24$
- $L(D \mid M) = 2 \cdot 0 = 0$
- $L(M) + L(D \mid M) = 24$



We prefer the model with 2 rectangles!

Overall method

1. Generate the set of candidates $\mathcal{S}$ (e.g. enumerate all distinct rectangle on the grid)
   - Candidate can have any shape
   - Apply *intra-ROI* constraints to filter the candidate set
   - Compute their contribution to the description length
2. Solve an Integer Linear Problem (ILP) to select the ROIs in $\mathcal{S}$
   - Model *inter-ROI* constraints with linear constraints in the ILP
   - Solve the ILP, the binary decision variables give the set of ROIs

The ILP to solve

If we denote d_i (resp. u_i) the dense (resp. non-dense) cells covered by the candidate $R_i \in \mathcal{S}$ on the grid $\mathcal{G}$, we need to solve the following ILP to select the ROIs.

$$\text{minimize} \sum_{R_i \in \mathcal{S}} x_i \cdot \overbrace{\left(\underbrace{2(u_i - d_i)}_{\text{added to } L(D|M)} + \underbrace{size(R_i)}_{\text{added to } L(M)} \right)}^{\text{Contribution to the description length}}$$

$$\text{subject to}$$

$$\sum_{R_i \in \mathcal{S} | c \in R_i} x_i \leq 1 \qquad \forall c \in \mathcal{G}$$

$$x_i \in \{0, 1\} \quad \forall R_i \in \mathcal{S}$$

Experiments

Setup

- Two versions of our method
  - With only rectangular regions
  - With rectangular and circular regions
- Showing results on Kaggle taxis dataset (≈1.6 million trajectories)
- Comparing with PopularRegion[1] and OPTICS[2] (when clustering the dense cells)

[1] Fosca Giannotti et al. "Trajectory pattern mining". In: *SIGKDD*. 2007.

[2] Mihael Ankerst et al. "OPTICS: ordering points to identify the clustering structure". In: *ACM Sigmod record* (1999).

Execution time

Minimum density threshold	2%			5%		
Grid side size	100	150	200	100	150	200
Number of ILP candidates	23 814	7 779	3 399	2 880	1 232	434
ILP optimization time (s)	4.328	0.464	0.109	0.113	0.044	0.029
PopularRegion run time (s)	0.003	0.005	0.006	0.002	0.003	0.004
OPTICS run time (s)	0.209	0.222	0.200	0.084	0.065	0.051

Description Length

- For high density threshold, number of errors becomes similar
- ILP-based methods produce smaller models
- Overall the Description Length is inferior for ILP-based methods

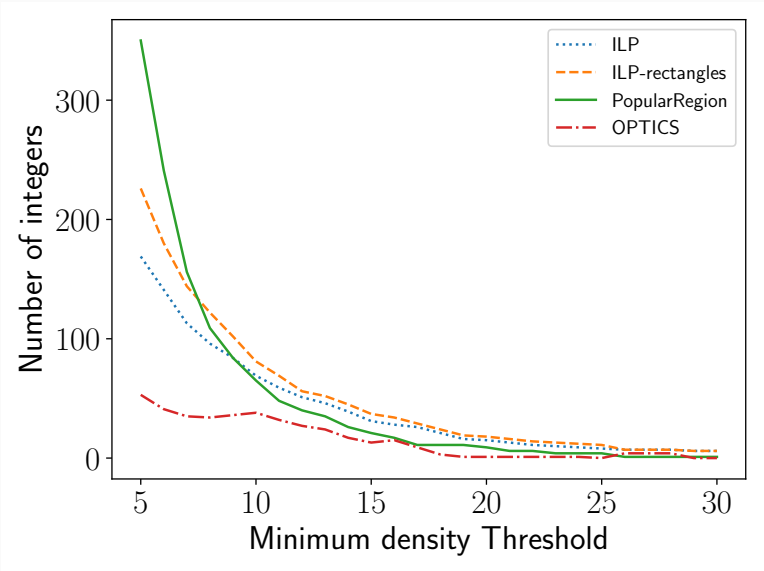


Figure 3: Encoding of the errors

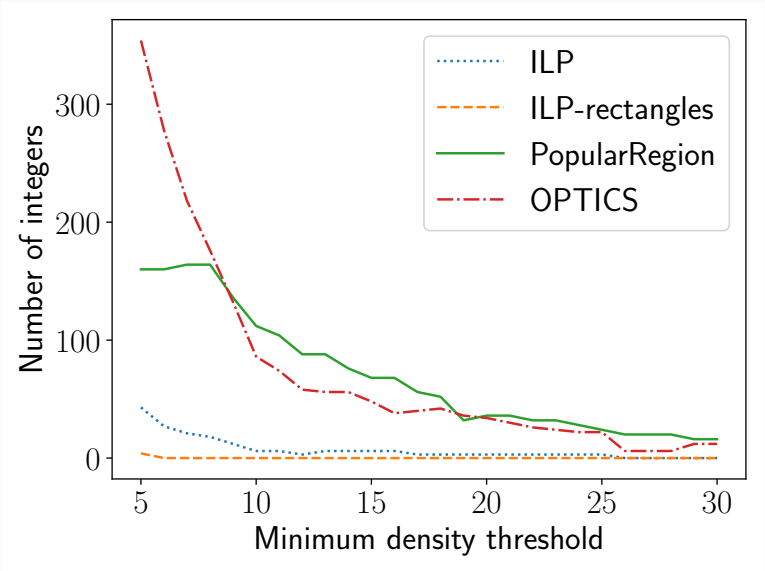


Figure 4: Encoding fo the models

Conclusion and Future work

What we did:

- We propose an optimization model to extract ROIs from trajectory data
- Our method is more flexible than specific method since it accepts a wide range of constraints
- The runtime of the ILP becomes reasonable as long as there is not too much candidates
- Everything is Open Source, see `https://github.com/AlexandreDubray/mining-roi`

The next steps:

- Get rid of the grid
- Use the density information (instead of just dense/not dense)
- Provide support for more complex constraints