Hyperspectral images classification with random walks

Michał Romaszewski

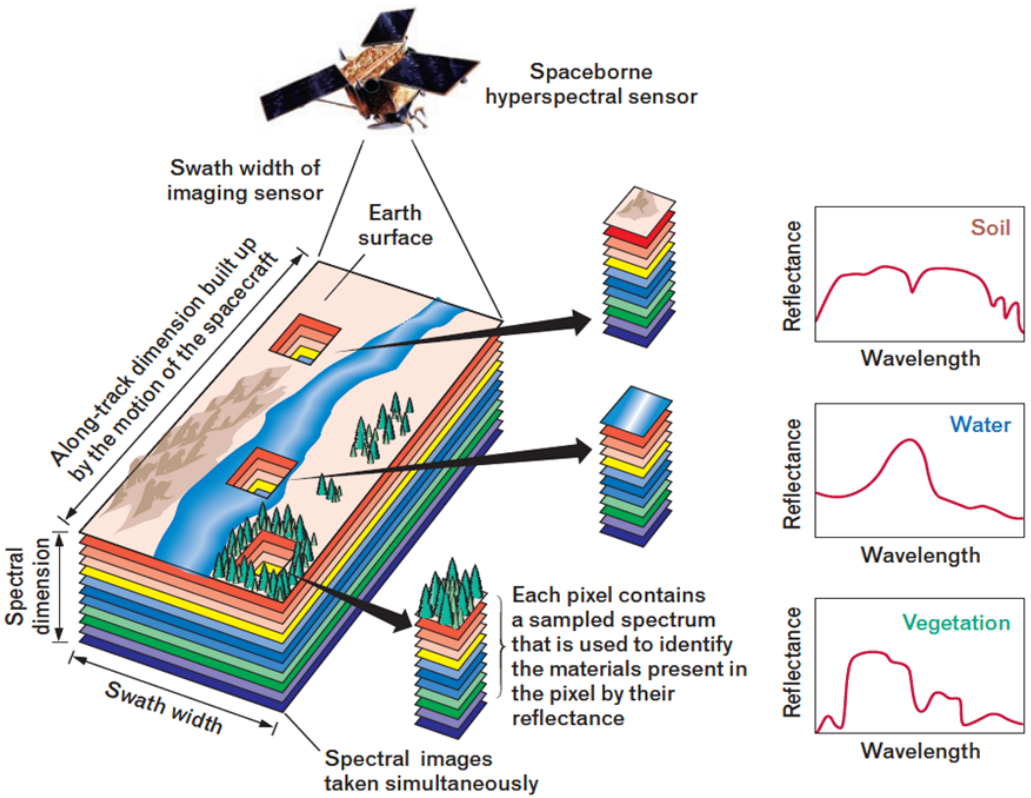
Institute of Theoretical and Applied Informatics, Polish Academy of Sciences

18 maja 2018

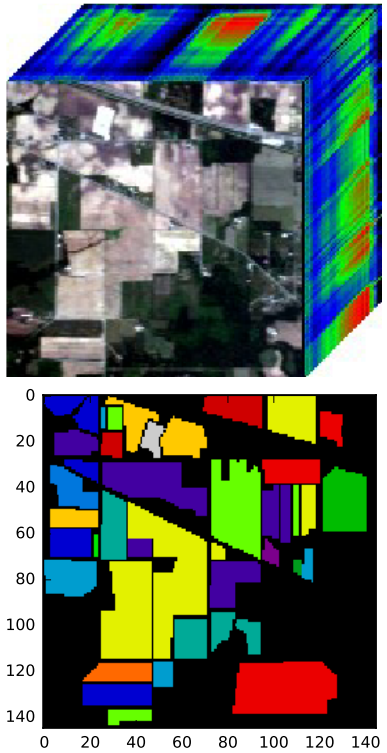
Presentation plan

- Hyperspectral data
- Assumptions of our problem
- Random walk-based algorithm
- Theoretical assumptions of learning
- Challenges and summary

Hyperspectral images (HSI)



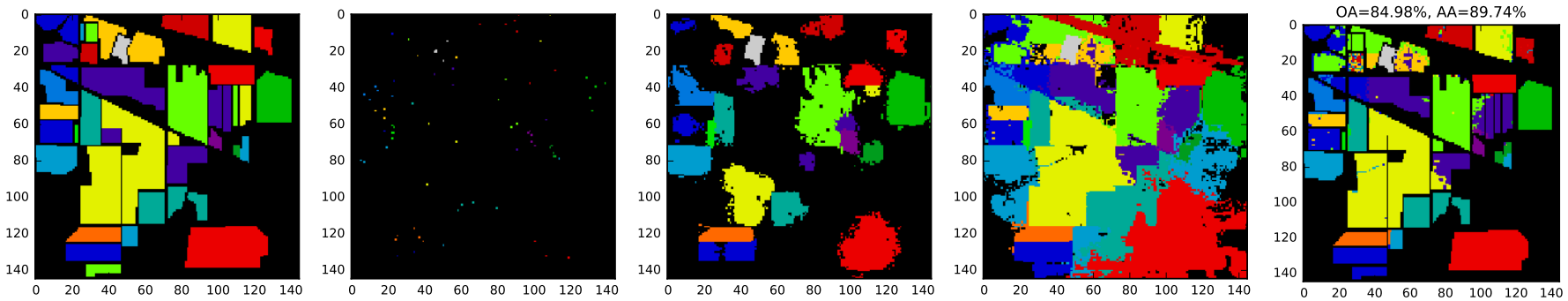
source: MIT Lincoln Laboratory



source: Teke, Mustafa, et al. „A short survey of hyperspectral remote sensing applications in agriculture"

Our goal

- We classify pixels in the HSI image
- We created a multi-view classification algorithm (using image spatial and spectral features)
  - Romaszewski, M., P. Głomb, and M. Cholewa, "Semi-supervised hyperspectral classification from a small number of training samples using a co-training approach", ISPRS Journal of Photogrammetry and Remote Sensing, vol. 121, pp. 60 - 76, 2016
- We are interested in its theoretical properties - we have modeled our idea with random walks

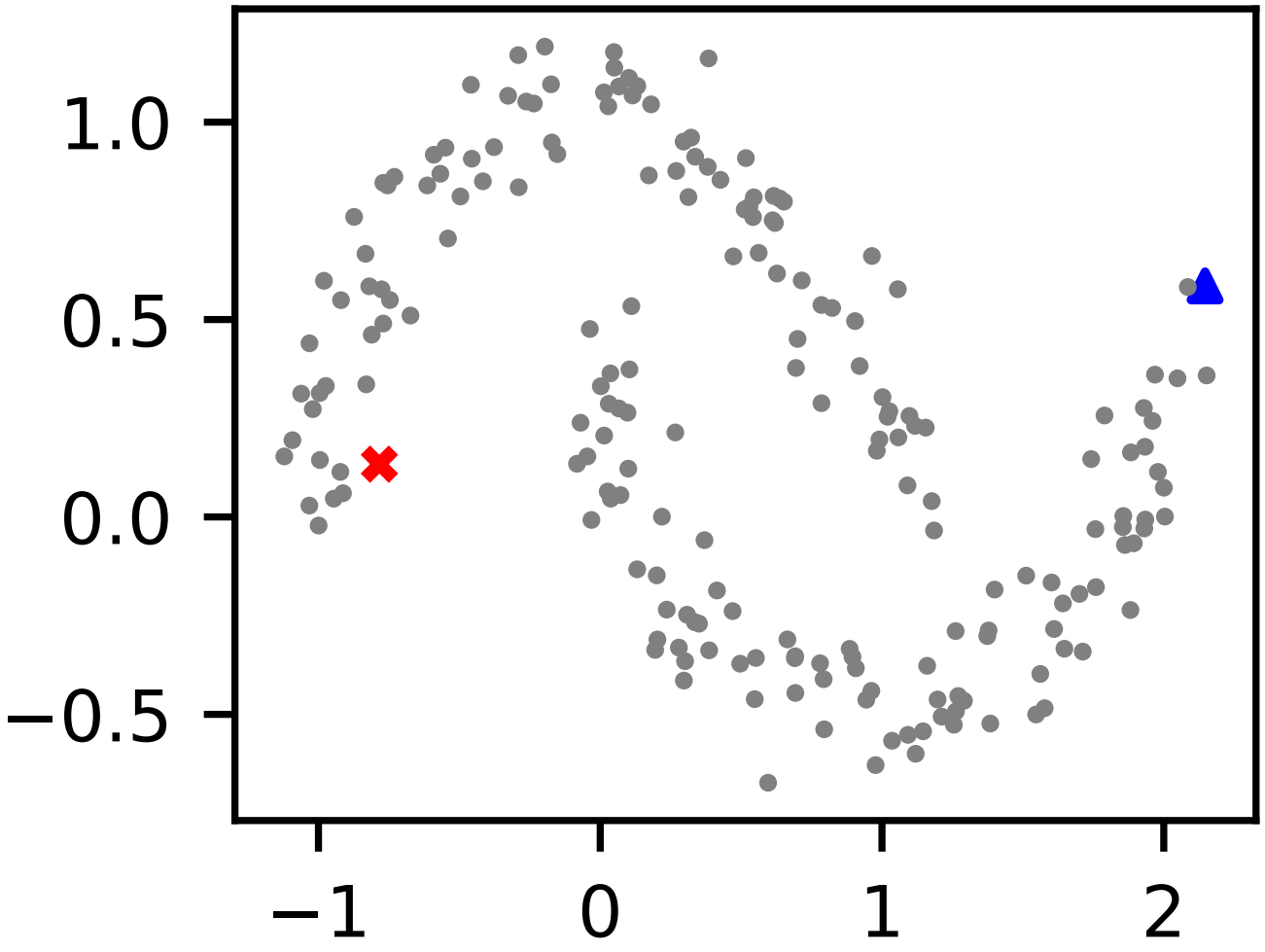


Assumptions of our classification algorithm

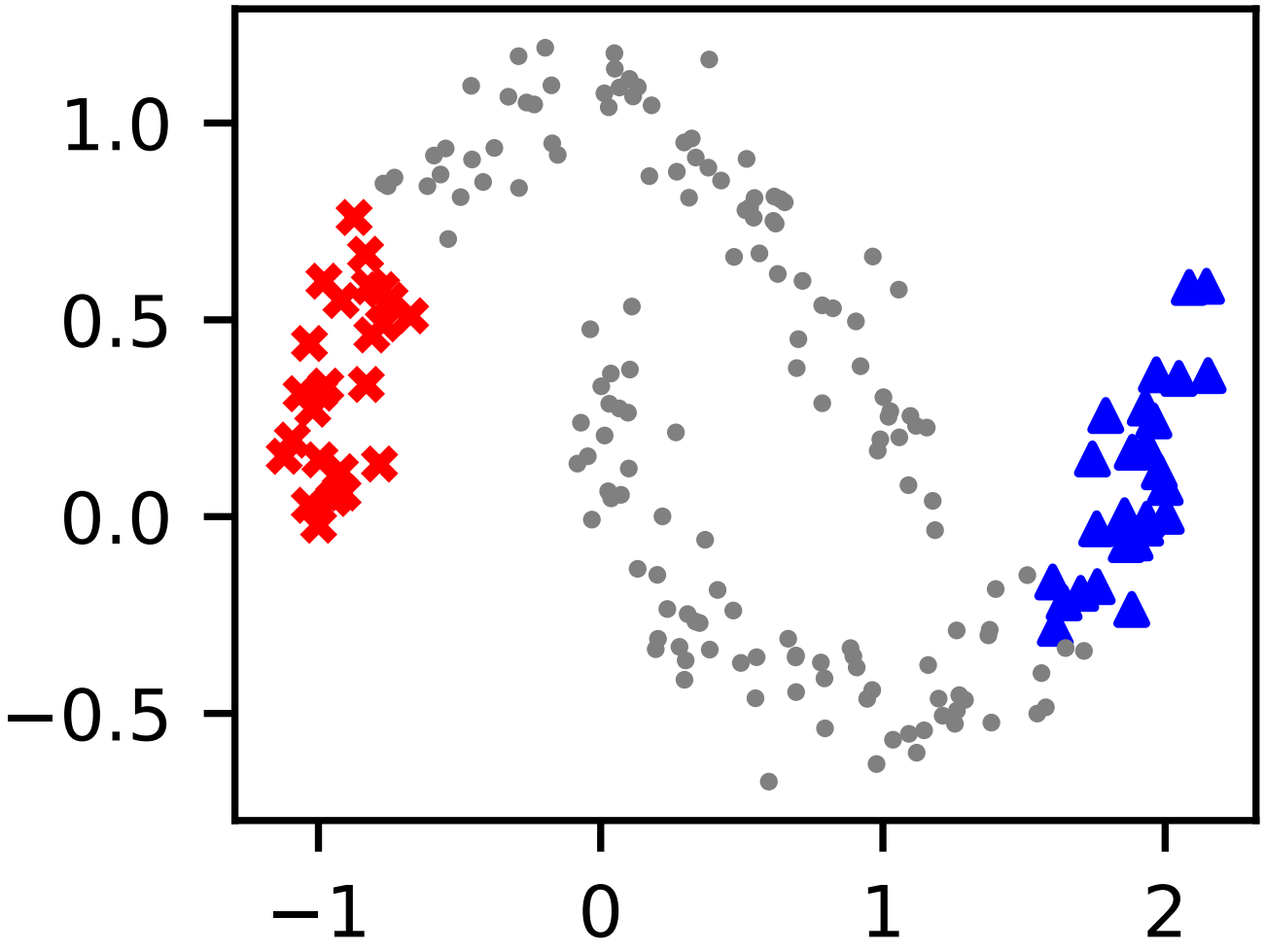
- $\mathcal{P}$: a set of HSI pixels with their positions,
- $\mathcal{C}$: a set of classes (e.g. bush,grain, dirt, building...).

- Our algorithm:
  - Iterative (t=1,2,...)
  - We have $\mathcal{T}_{\text{train}} \subset \mathcal{P}$, $\mathcal{T}_{\text{test}} \subset \mathcal{P}$, and corresponding labels $\mathcal{L}_{\text{train}}, \mathcal{L}_{\text{test}}$
  - We use $\underbrace{\mathcal{T}_{\text{train}}, \mathcal{L}_{\text{train}} \text{ and } \mathcal{T}_{\text{test}}}_{\text{semi-supervised}}$ to obtain a mapping $\underbrace{\phi : \mathcal{P} \rightarrow \mathcal{C}}_{\text{transductive}}$

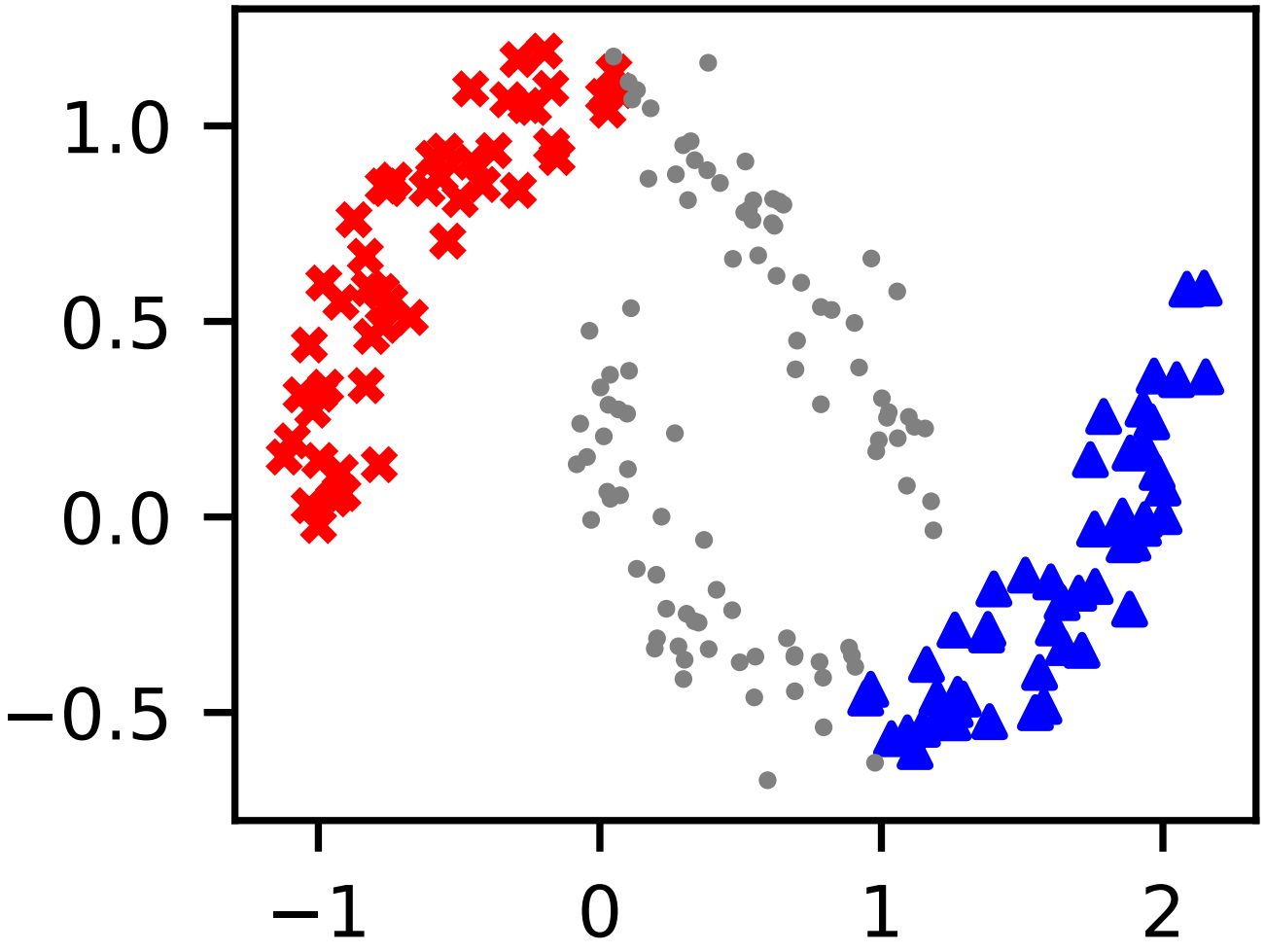
Transductive, semi-supervised classification (1/5)



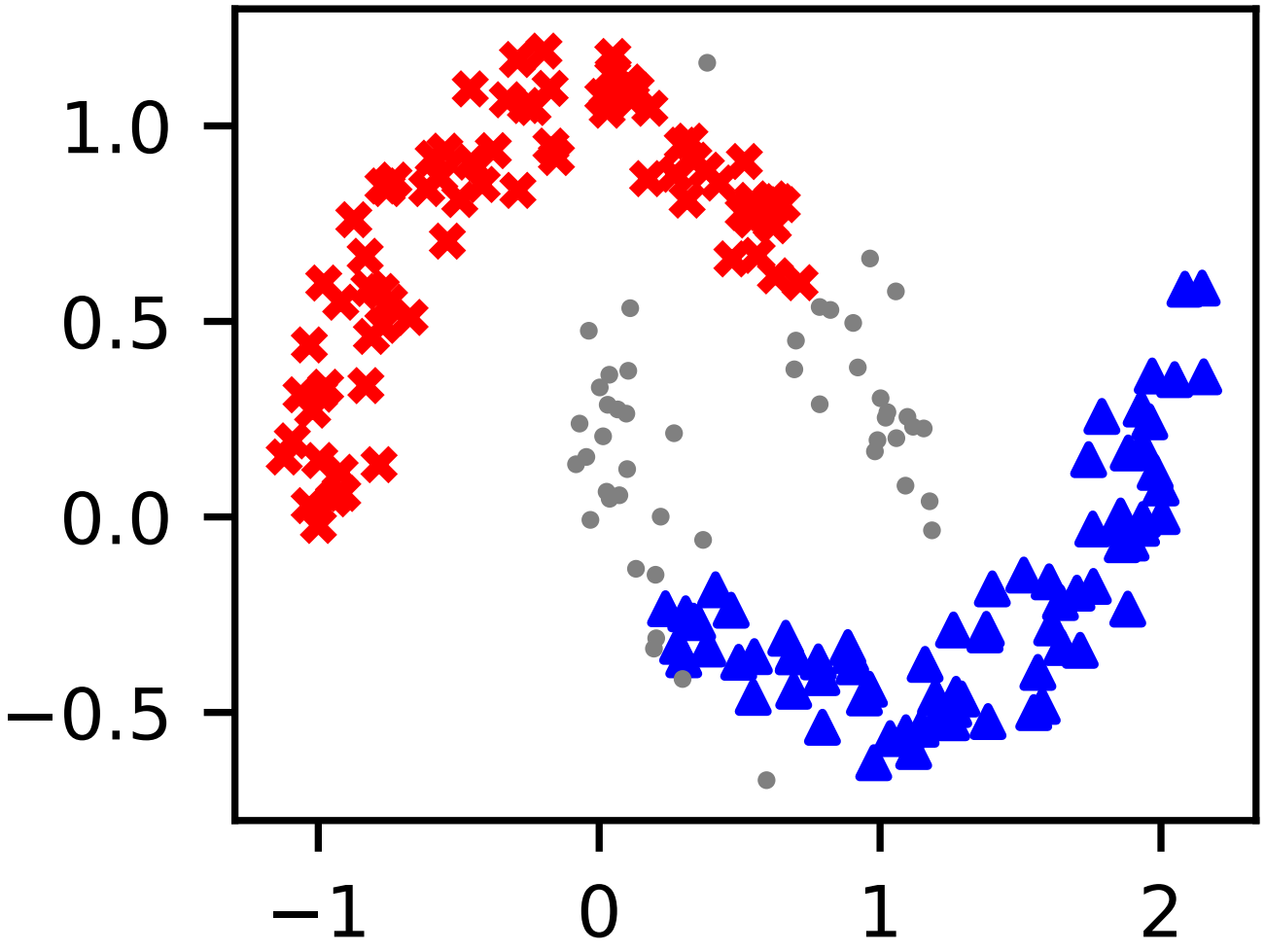
Transductive, semi-supervised classification (2/5)



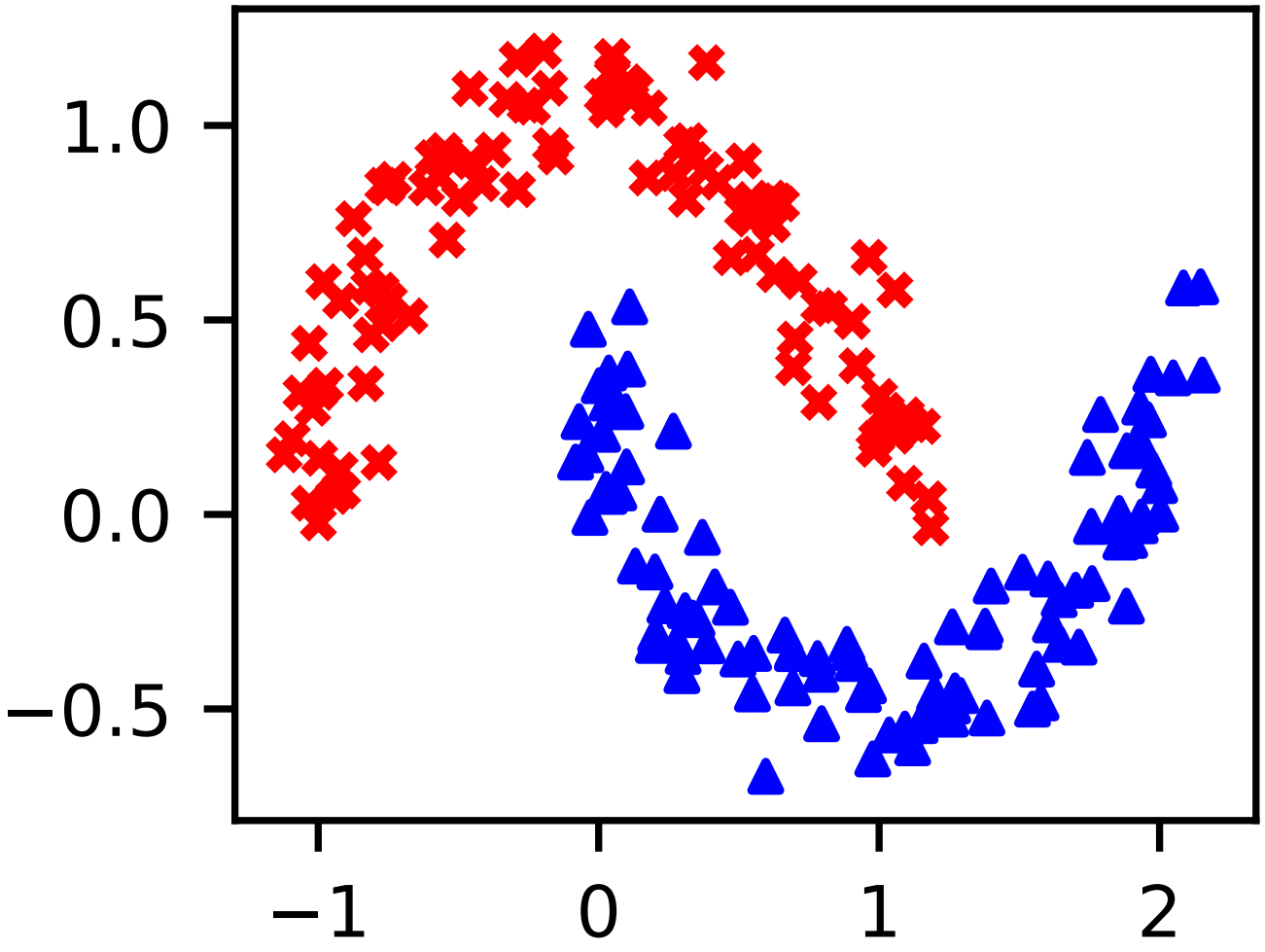
Transductive, semi-supervised classification (3/5)



Transductive, semi-supervised classification (4/5)



Transductive, semi-supervised classification (5/5)

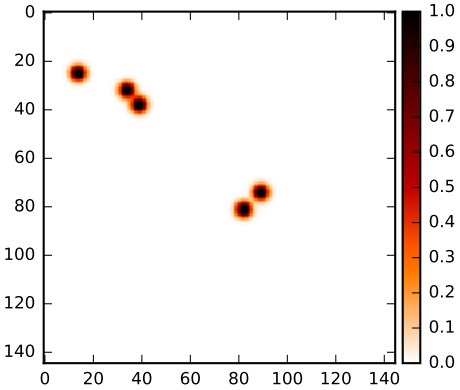


The general idea

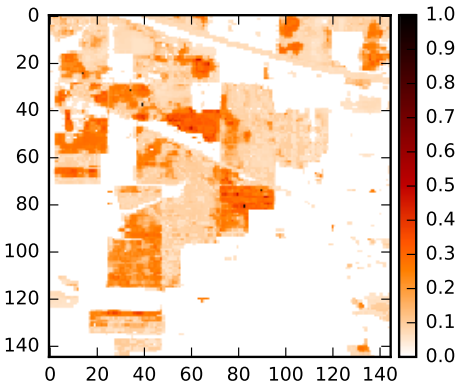
- We have a labelled set $\mathcal{T}^t_{\mathrm{la}} \subseteq \mathcal{P}$ and an unlabelled set: $\mathcal{T}^t_{\mathrm{unl}} \subseteq \mathcal{P}$
- In each iteration $t > 0$:
  - We model spatial and spectral features separately. For each view $v \in \mathcal{V}$:
    - We model classes $c \in \mathcal{C}$
    - From the model we obtain class probabilities for pixels $\mathbf{F}^t_v \in \mathbb{R}^{\mathcal{P} \times \mathcal{C}}$
  - We combine class probabilities from all views
  - W label a subset of most probable pixels in $\mathcal{T}^t_{\mathrm{unl}}$
  - We finish if the set $\mathcal{T}^t_{\mathrm{unl}} = \emptyset$

Example: Two iterations of the algorithm

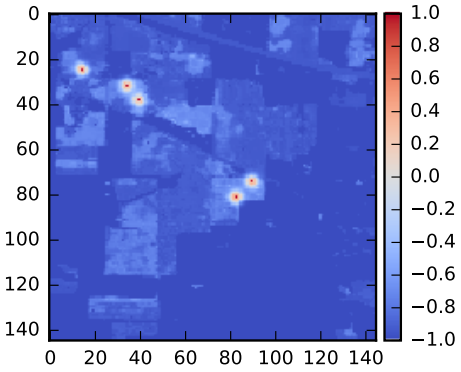
t=1



Spatial probabilities

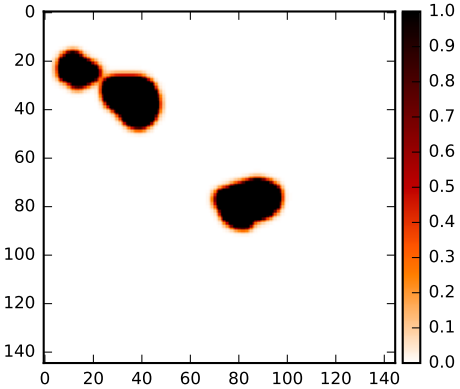


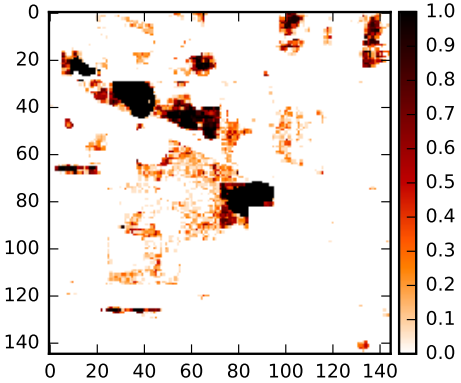
Spectral probabilities

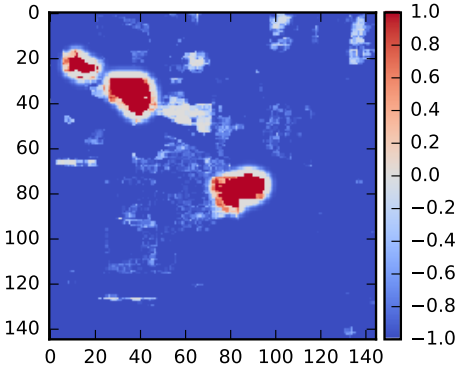


Combined score

t=4







Modelling HSI spatial and spectral features

- We have two views $v \in \mathcal{V}$ and pixel features given by functions $f_v : \mathcal{P} \to \mathbb{R}^{d_v}$ (pixel position and reflectance spectrum)
- We model pixel similarities with graphs $\mathcal{G}_v = \langle \mathcal{P}, \mathcal{E}_v, \mathbf{W}_v \rangle$:
  - weight matrices $\mathbf{W}_v \in \mathbb{R}_+^{\mathcal{P} \times \mathcal{P}}$ where $\mathbf{W}_v(p, r) = \exp\left(-\frac{\|f_v(p) - f_v(r)\|^2}{2\delta^2}\right)$ and $p, r \in \mathcal{P}$, $\delta > 0$,
  - a set of edges $\mathcal{E}_v = \{e : e \in \mathcal{P} \times \mathcal{P} \wedge \mathbf{W}_v(e) > 0\}$

Modelling classes $c \in \mathcal{C}$ with random walks

- Let $\mathcal{Y}_c^t \subset \mathcal{T}_{\mathrm{la}}^t$ be a set of nodes with label $c \in \mathcal{C}$ in iteration t
- For each view $v \in \mathcal{V}$ we use weight matrix $\mathbf{W}_v$ to obtain RW probability transition matrix $\mathbf{N}_v \in \mathbb{R}_+^{\mathcal{P}\times\mathcal{P}}$
- For an iteration $t > 0$:
  - For a set $\mathcal{Y}_c^t$ from class $c \in \mathcal{C}$:
    - We use $\mathcal{Y}_c^t$ to obtain RW starting probabilities $\mathbf{p}_c \in \mathbb{R}_+^{\mathcal{P}}$,
    - After $n \geqslant 1$ steps of the RW we have $\mathbf{f}_c = \underbrace{(\mathbf{N}_v^T)^n}_{\mathbf{S}_v \in \mathbb{R}_+^{\mathcal{P}\times\mathcal{P}}} \mathbf{p}_c$
  - Column-wise stacking vectors $\mathbf{f}_c$ and $\mathbf{p}_c$ for $c \in \mathcal{C}$ we obtain

$$\mathbf{F}_v^t = \mathbf{S}_v \mathbf{P}_v^t,$$

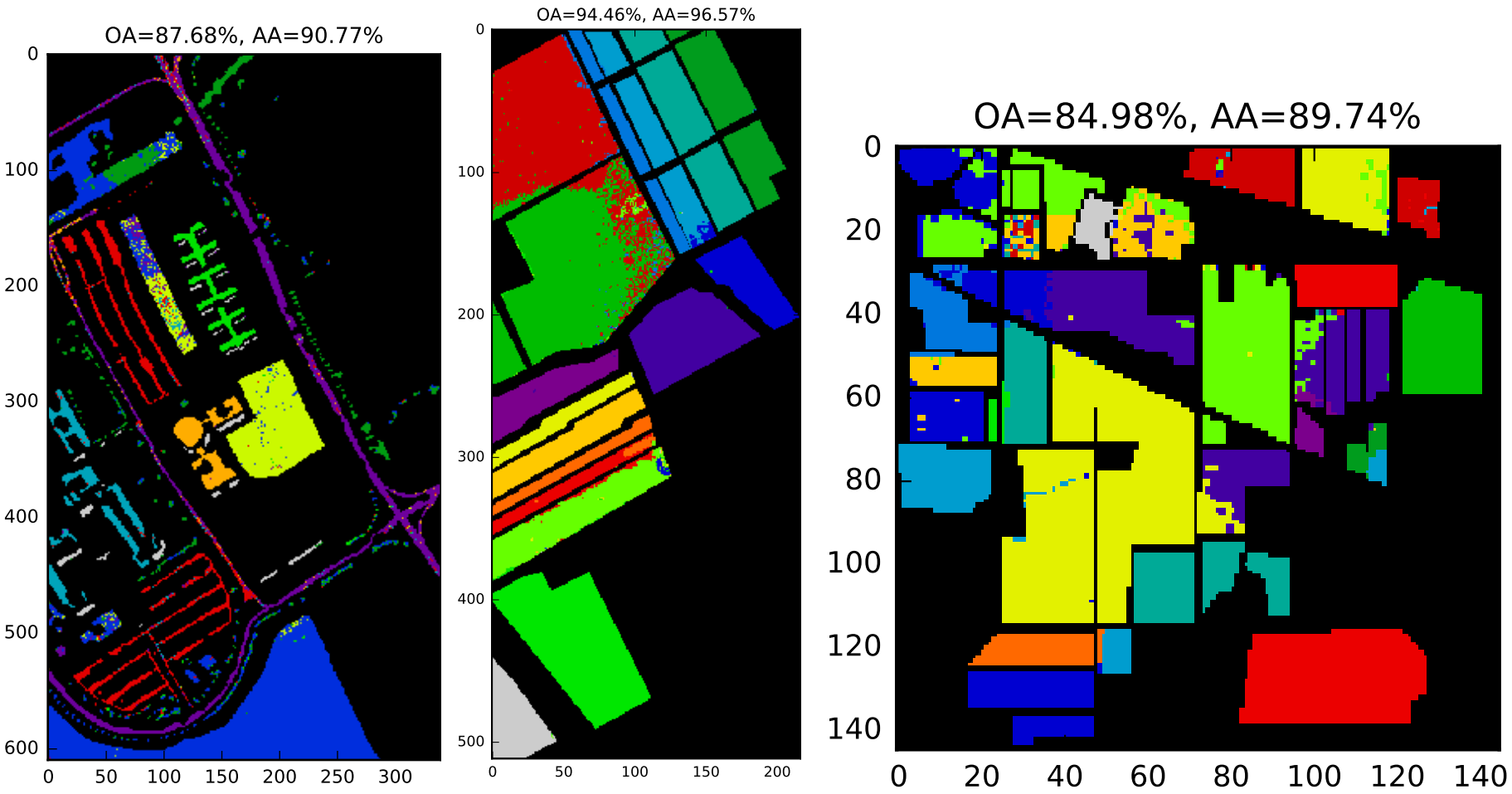
where the matrix $\mathbf{P} \in \mathbb{R}_+^{\mathcal{P}\times\mathcal{C}}$.

Data: Matrices $\mathbf{S}_v$, $\mathcal{T}_{\mathrm{la}}^0$, coefficients α_v, RW parameters.
Result: Labels of nodes in $\mathcal{T}_{\mathrm{la}}^m$ for the last iteration $m > 0$.

$t \longleftarrow 0$;
repeat
  for $v \in \mathcal{V}$ **do**
    $\mathbf{P}_v^t \longleftarrow$ obtain RW starting probabilities from $\mathcal{T}_{\mathrm{la}}$;
    $\mathbf{F}_v^t \longleftarrow \mathbf{S}_v \mathbf{P}_v^t$;
  end
  $\mathcal{T}_{\mathrm{selected}} \longleftarrow \emptyset$;
  repeat
    $(p, c) \longleftarrow \arg\max_{k \in \mathcal{T}_{\mathrm{unl}}^t \setminus \mathcal{T}_{\mathrm{selected}}^*, l \in \mathcal{C}} \sum_{v \in \mathcal{V}} \alpha_v \mathbf{F}_v^t(k, l)$;
    $\mathcal{T}_{\mathrm{selected}} \longleftarrow \mathcal{T}_{\mathrm{selected}} \cup (p, c)$;
  until $\mathcal{T}_{\mathrm{selected}}$ *is sufficient*;
  update $\mathcal{T}_{\mathrm{la}}^{t+1}$ with $\mathcal{T}_{\mathrm{selected}}$;
  $t \longleftarrow t + 1$;
until $\mathcal{T}_{\mathrm{la}}^{t+1} = \mathcal{T}_{\mathrm{la}}^t$;

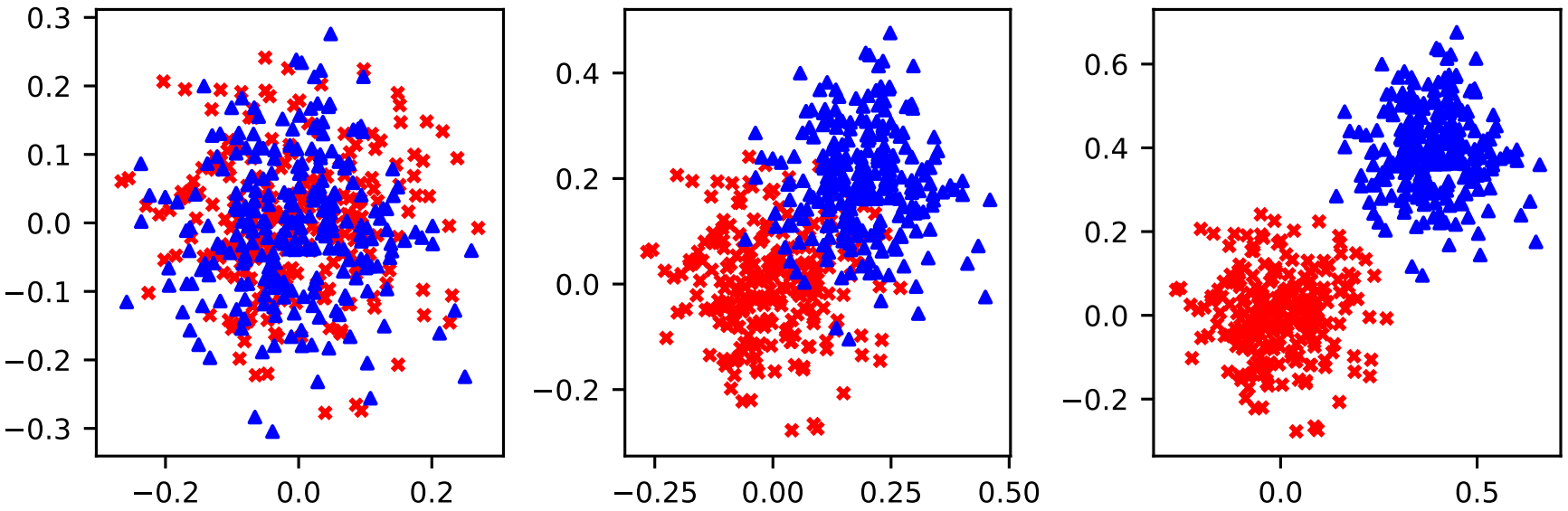
Results of our original algorithm

Small training set of 5 samples/class



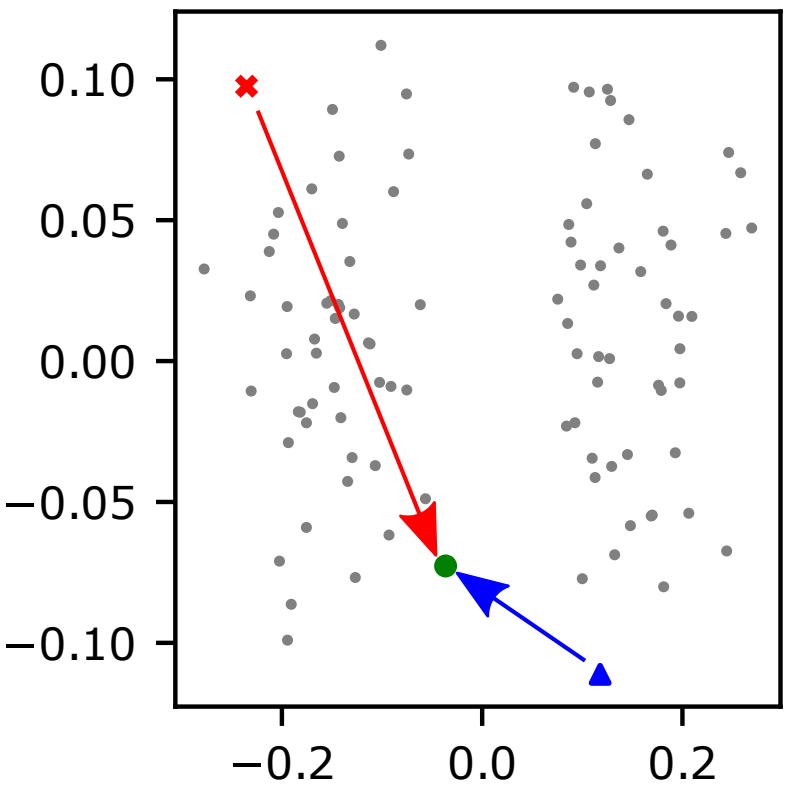
When classification works?

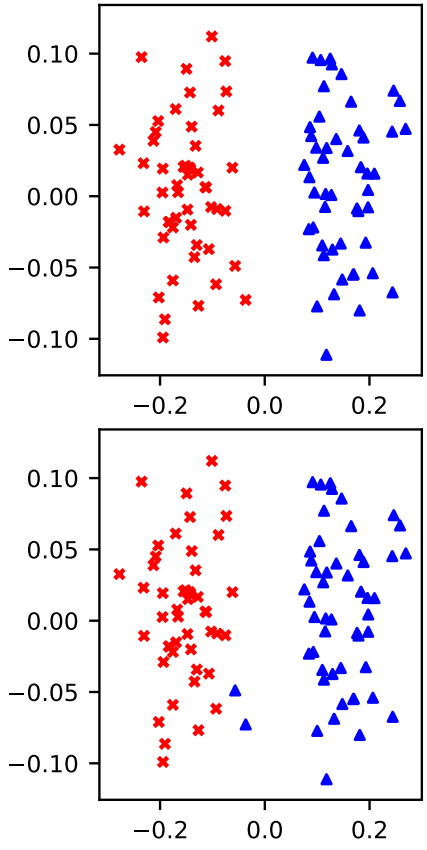
- We are looking for structures in data
- It's sometimes formulated as the **clustering assumption**
- Random walks are our tool of choice for exploring this assumption



We can define clusters through random walks

- Node denoted as • should be labelled as ×
- Its combined probability from $\mathcal{Y}^t_{\times}$ must be the highest





Defining formally

Definition

A cluster $\phi_c \subset \mathcal{P}$ of the class $c \in \mathcal{C}$ is a subset of nodes such that for every subset $(\mathcal{T}^t_{\mathrm{la}} \subset \mathcal{P}) \wedge (\mathcal{Y}^t_c \cap \phi_c \neq \emptyset) \wedge (\forall_{\bar{c} \in \mathcal{C} \setminus c} \mathcal{Y}^t_{\bar{c}} \cap \phi_c = \emptyset)$, every node $p \in \phi_c \cap \mathcal{T}^t_{\mathrm{unl}}$ and every iteration $t > 0$, the sum

$$\sum_{v \in \mathcal{V}} \alpha_v \sum_{k \in \mathcal{Y}^t_c \cap \phi_c} \mathbf{S}'_v(p,k) \mathbf{P}^t_v(k,c) >$$

$$\max_{c' \in \mathcal{C}} \sum_{v \in \mathcal{V}} \alpha_v \sum_{k \in \mathcal{Y}^t_{c'} \setminus \phi_c} \mathbf{S}'_v(p,k) \mathbf{P}^t_v(k,c'),$$

We can give sufficient condition for classification

Theorem

Given graphs[a] $\mathcal{G}_v$ in views $v \in \mathcal{V}$, every class $c \in \mathcal{C}$, every iteration $t > 0$ and every node $p \in \mathcal{T}^t_{\text{unl}} \cap \mathcal{N}_c \cap \mathcal{T}_{\text{selected}}$, where $\mathcal{N}_c \subset \mathcal{P}$ is a set of nodes from class c, if

$$\sum_{v \in \mathcal{V}} \alpha_v \sum_{k \in \mathcal{N}_c \cap \mathcal{T}^t_{\text{la}}} \mathbf{S}_v(p,k)\mathbf{P}^t_v(k,c) >$$

$$\max_{\bar{c} \in \mathcal{C} \setminus \{c\}} \sum_{v \in \mathcal{V}} \alpha_v \sum_{k \in \mathcal{N}_{\bar{c}} \cap \mathcal{T}^t_{\text{la}}} \mathbf{S}_v(p,k)\mathbf{P}^t_v(k,\bar{c}),$$

then the node p will be classified correctly.

[a] mild assumptions regarding graph structure are also required.

We can bound the sufficient condition

Corollary

1. If for every node $p \in \mathcal{N}_c \cap \mathcal{T}_{\text{test}}$ from every class $c \in \mathcal{C}$ there exists a cluster $\phi_c \subseteq \mathcal{N}_c$ such that $(p \in \phi_c) \wedge (\phi_c \cap \mathcal{T}_{\text{train}} \neq \emptyset)$,
2. in any iteration $t > 0$ the cardinality of the set $|\mathcal{T}^t_{\text{selected}}| \geqslant 1$,
3. for every labelled node $l \in \mathcal{Y}^t_c$ from every class $c \in \mathcal{C}$ in every iteration $t > 0$ for every view $v \in \mathcal{V}$ the probability $\mathbf{P}^t_v(l, c) = \frac{1}{|\mathcal{Y}^t_c|}$,

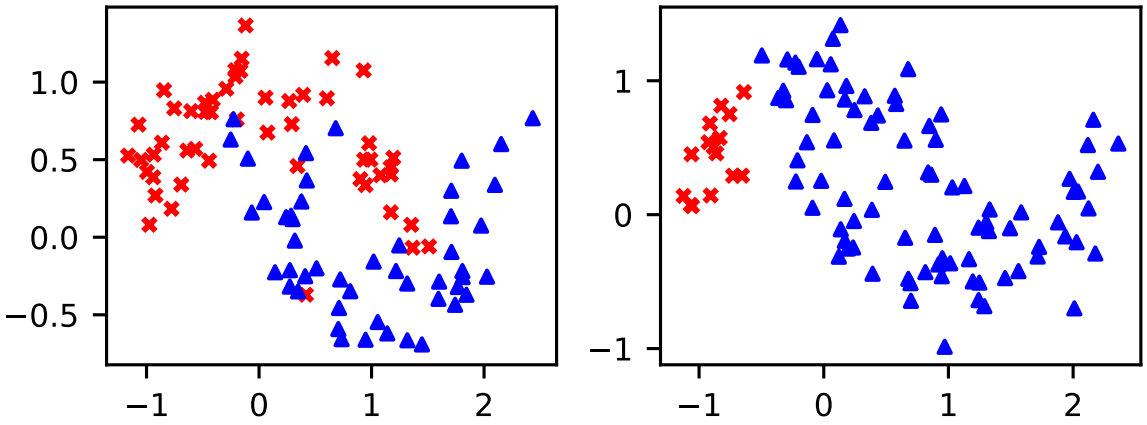
then

$$\min_{k \in \phi_c} \sum_{v \in \mathcal{V}} \alpha_v \mathbf{S}'_v(p, k) >$$

$$\max \left((|\mathcal{N}_c| - |\phi_c| + 1) \max_{\substack{\bar{c} \in \mathcal{C} \setminus \{c\} \\ k \in \mathcal{N}_{\bar{c}}}} \sum_{v \in \mathcal{V}} \alpha_v \mathbf{S}'_v(p, k) \, , \max_{k \in \mathcal{N}_c \setminus \phi_c} \sum_{v \in \mathcal{V}} \alpha_v \mathbf{S}'_v(p, k) \right) .$$

Challenges (RW+HSI)

- Greedy solution (e.g. early error may propagate)
- Short random walks (no clusters in the stationary state!)
  - Not necessary a problem of all RW/properties
- How to assign starting probabilities?
  - For HSI data, spatial-based heuristics seems promising
- View/graph creation is very important and not trivial



To summarise

- Useful model, very nice for analysis of theoretical properties
- State of the art classification algorithms can be explained with RW
- Easy to understand, sometimes hard to master with real data
- Extendable
  - The algorithm + absorbing random walks $\approx$ label propagation

Thank you. Any questions? Suggestions?
contact me: michal@iitis.pl

Extension: Absorbing random walks[a]

[a] This is similar to label propagation!

- An absorbing state - impossible to leave once entered.
- Other states are transient.
- Absorbing states are reachable from any state.
- Transition matrix in the canonical form:

$$P = \begin{array}{c} \\ Transient \\ Absorbing \end{array} \begin{array}{c} \begin{array}{cc} Transient & Absorbing \end{array} \\ \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I}_r \end{pmatrix} \end{array}$$

  - A set $\mathcal{T}$ of transient states
  - $\mathbf{Q} \in \mathbb{R}^{\mathcal{T} \times \mathcal{T}}$ - transient to transient transition probabilities
  - $\mathbf{R} \in \mathbb{R}^{\mathcal{T} \times \mathcal{R}}$ - transient to absorbing transition probabilities
  - Zero matrix $\mathbf{0} \in \{0\}^{\mathcal{R} \times \mathcal{T}}$, identity matrix $\mathbf{I}_r \in \{1\}^{\mathcal{R} \times \mathcal{R}}$