Interpretable Machine Learning using Pattern Mining

Martin Atzmueller

CSLab

Tilburg University

JADS Jheronimus Academy of Data Science

Interpretable Machine Learning using **Pattern Mining**

Martin Atzmueller

CSLab

Tilburg University

JADS Jheronimus Academy of Data Science

Motivation

- Models for ..
  - Getting first insights into the data
  - Obtaining concept description
  - Finding descriptions/characteristics …
  - Identifying anomalies/outliers
  - Also: for complex data
  - …

- Interpretable models … in the form of human-interpretable patterns
  ➔ descriptive patterns (subgroups)

Pattern Mining

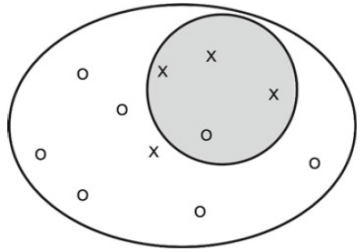
- Association rule mining

  {Bread, Butter, Milk, Sugar}: Bread, Butter, Milk => Sugar

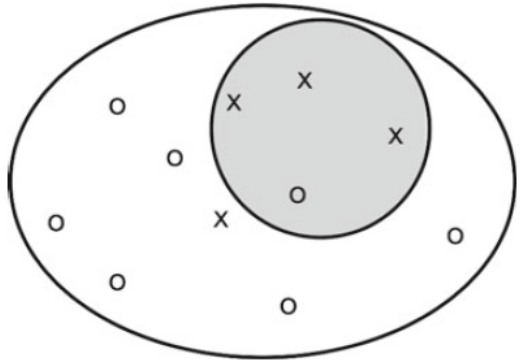
  {Butter, Flour, Milk, Sugar}: Butter, Flour, Sugar => Milk

  {Butter, Eggs, Milk, Salt}: Butter, Eggs, Milk => Salt

- ***Subgroup discovery***



Classic Subgroup Discovery



- Subgroup discovery (SD):
  „Find *descriptions* of subsets in the data,
  that *differ* significantly for the total
  population with respect to a *target concept.*“
  [Kloesgen 1996, Wrobel 1997]
- Example: "45% [p] of all men aged between 35 and 45 (100 individuals [n]) have a **medium income** in contrast to only 20% [p_0] in total."
- Descriptive pattern:
  Gender= Male ∧ Age = [35; 45] ➔ Income = medium

Classic Subgroup Discovery

- Given:
  - Data as set of cases (records) in tabular form
  - Target concept (e.g. "medium income")
  - Quality function (interesting measure)
- Result: Set of the best *k* **Subgroups:**
  - Description, e.g., sex=male ∧ age= 35-45

    ➔ Conjunction of *selectors*
  - Size *n,* e.g., in 180 of 1000 cases
  - Deviation (p = 45% in the subgroup vs. p_0=20% in all cases)

  ➔"*Quality*" of the subgroup: weight size and deviation

Measured by quality function, e.g., $n^a \cdot (p - p_0)$

Classic Subgroup Quality Functions

- Consider size and deviation in the target concept

a: weight size against deviation (parameter)

$$n^a \cdot (p - p_0)$$

*n:*Size of subgroup (number of cases)

p: share of cases with *target = true* in the subgroup
p_0: share of cases with *target = true* in the total population

  - Piatetsky Shapiro (a = 1) – similar to Weighted Relative Accuracy (WRAcc)
  - Simple Binomial (a = 0.5)
  - Added Value (a = 0)
- Continous: Mean value (m, m_0) of target variable

$$q_{CWRACC} = \frac{n}{N} \cdot (m - m_0), \quad q_{CPS} = n \cdot (m - m_0)$$

Subgroup Discovery for Complex Models (Exceptional Model Mining)

Subgroup Discovery

Exceptional Model Mining

- Variation of subgroup discovery:
  - Focus on **complex model class, e.g.,** *multiple target attributes*
  - Choose a ***model class***
    (e.g., correlation coefficient, linear regression model, spatial distribution, community structure, …)
- For a set of instances build *a model of the target attributes* and determine model parameters
- Search for descriptions, for which model parameters are different to those extracted from the total population

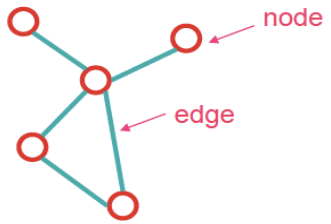
Example

"For women with university degree the correlation coefficient between age *and* income *is 0.6, while it is only 0.2 in the overall dataset. "*

Attributed Networks

- Represented as (attributed) graph
- Enables simultaneous analysis of relational + attributive data
- Structural information
  - Links/Ties
  - Specific relations
  - Make up **connections** in graph/network
- Attributes - compositional variables
  - Attributes: properties of nodes, e.g.
    - gender, age, for social networks
    - Message/text for information networks
  - Edge (link) attributes
  - Attribute vectors for actors and/or links
    ➔ here: node attributes
  - **Describe** nodes (e.g. actors)

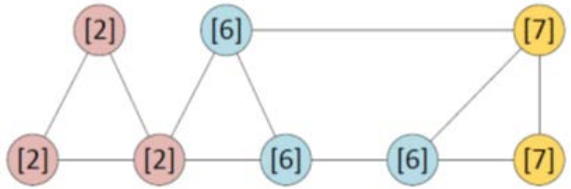
A graph $G =< V, E \subseteq V \times V >$



- V: set of nodes (a.k.a. vertices, actors, sites)
- E: set of edges (a.k.a. ties, links, bonds)

Attributed Network

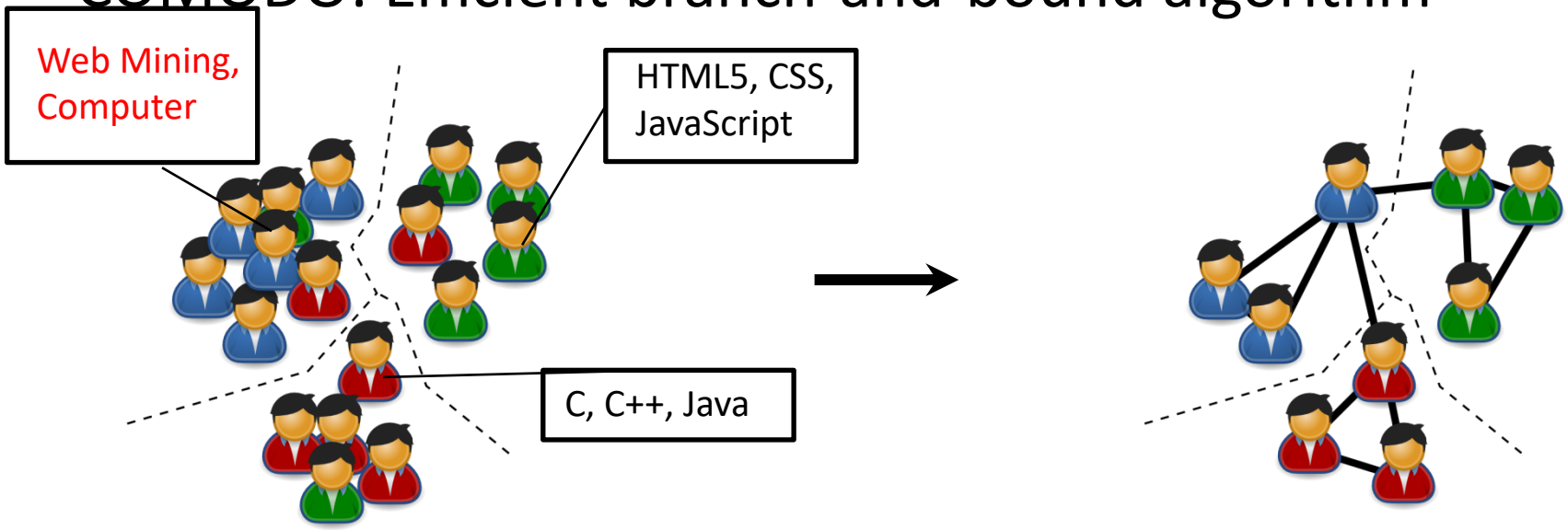
Definition 1 [Zhou2009] Network represented by a graph G = (V,E) where each node $v \in V$ is associated with a vector of attributes $v_j, j = \{1, .., p\}$.



Description-Oriented Community Detection

[Atzmueller et al. 2016, Information Sciences]

- Community detection (EMM) on attributed graphs
  - Mine patterns in description space (tags/topics)
    ➔ Subgroups of users **described** by tags/topics
  - Optimize quality measure in graph structure
    (Modularity, conductance, inverse average-out-degree, ….)
- COMODO: Efficient branch-and-bound algorithm



Community Evaluation Measures

- Modularity
  [Newman 2006]

$$MOD(S) = \frac{1}{2m}\sum_{i,j}(A_{i,j} - \frac{d(i)d(j)}{2m})\delta(C_i, C_j)$$

Compares the number of edges within a community with the expected such number in a corresponding null model

$$MODL(C) = \frac{1}{2m}\sum_{i\in C, j\in C}(A_{i,j} - \frac{d(i)d(j)}{2m})$$

$$MODL(C) = \frac{m_C}{m} - \sum_{i\in C, j\in C}\frac{d(i)d(j)}{4m^2}$$

- Conductance
  [Kannan et al. 2004]

$$CON(C) = \frac{\overline{m}_C}{2m_C + \overline{m}_C}$$

Compares the number of edges within a community and the number of edges leaving the community

$$COIN(C) = 1 - CON(C) = \frac{2m_C}{\sum_{u\in C} d(u)}$$

Pruning & Optimistic Estimates

- Local Molularity: Only consider local vertex subset (one community)

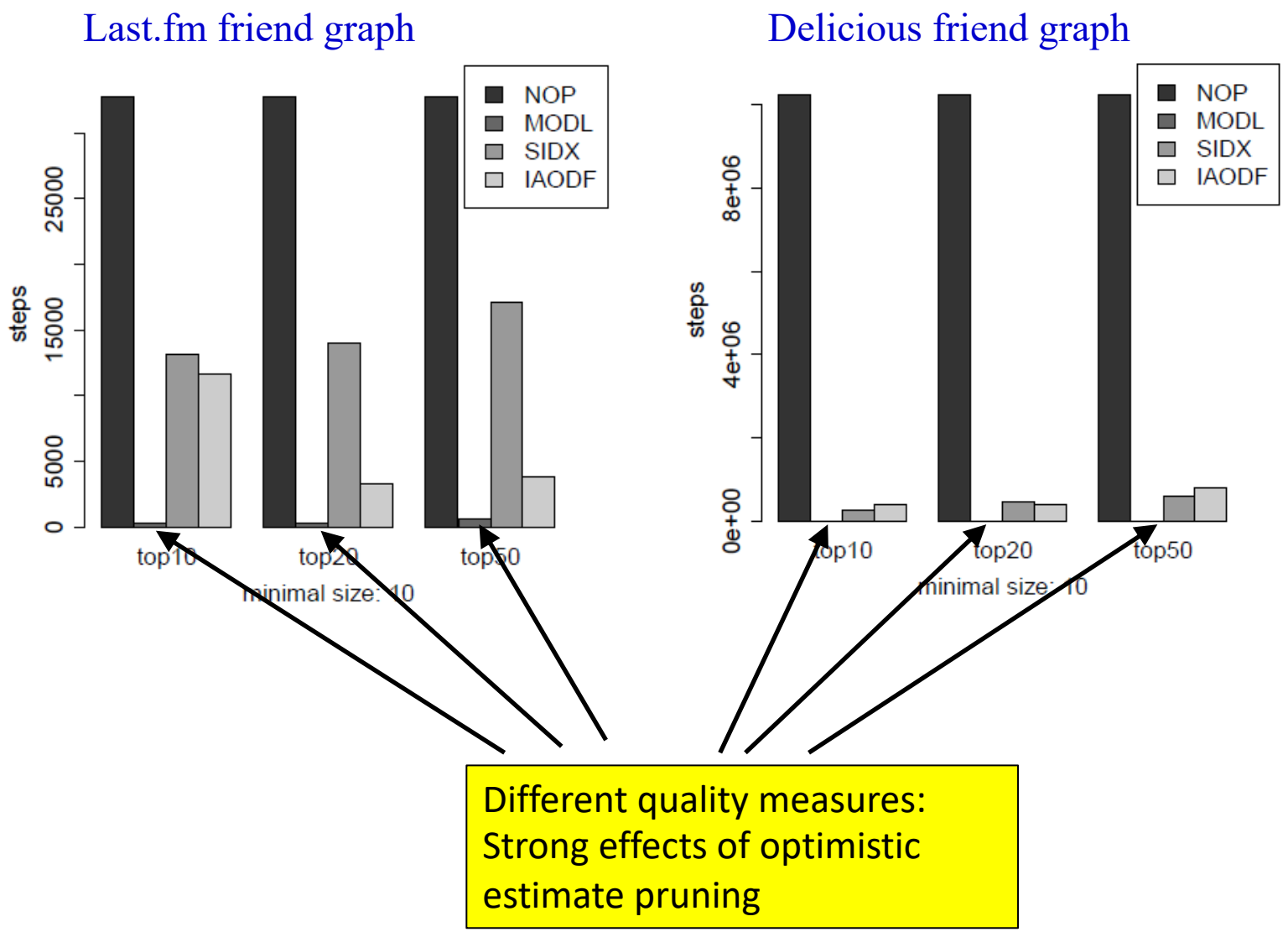
$$\mathrm{MODL}(W) = \frac{m_W}{m} - \sum_{u,v \in W} \frac{\mathrm{d}(u)\,\mathrm{d}(v)}{4m^2}$$

- Optimistic Estimate Pruning:
  - Top-K Pruning
  - Optimistic estimate:
    - Upper bound for the quality of a pattern and all its specializations
    - Enables pruning of (large) paths of the pattern search tree, if quality of the current node estimate is below the quality of the k-best patterns so far
- For local Modularity:

$$\mathrm{oe}(\mathrm{MODL})(W) = \begin{cases} 0.25, & \text{if } m_W \geq \frac{m}{2} \\ \frac{m_W}{m} - \frac{m_W^2}{m^2}, & \text{otherwise.} \end{cases}$$

Performance: Different Community Quality Functions

Social media data – friend graphs labeled with interests



Compositional Subgroup Discovery

[Atzmueller, DS 2019]

- INPUT: Attributed graph dataset
  - Data set in tabular form, describing node properties
  - Network, represented as (multi-)graph with weighted edges
  - Target concept: Duration or frequency – looking for deviations
- OUTPUT: Set of the best *k* **Subgroups:**

– Specifically, consider dyadic structure for subgroup discovery

  - Focus on social interactions structure (dyads), e.g., frequency of interactions, duration of interactions, etc.
  - Key question: What is exceptional? Compare with expectation/null model!

$$q_S(P) = Z(\frac{1}{n_E} \cdot \sum_{e \in E_P} w(e)) \qquad n_E = \frac{n_{E_P}(n_{E_P}-1)}{2}$$

Example: Conference Dataset

[Atzmueller, DS 2019]

- Edge weights: Duration of contacts
- Edge multiplicity: Number of interactions between two participants
- Socio-demographic information:
  - Gender
  - Country of Origin
  - Affiliation (University)
  - Academic status (Professor, PostDoc, PhD student, ...)
  - Main conference track of interest
- Constructed labels of the edges (dyads) on whether properties of concerned nodes conincide (“EQ“) or not (“NEQ“)

LWDA 2010 (simple interaction graph) [Atzmueller, DS 2019]

Description	Size	∅CLength	Quality (Z)
track=EQ	456	182.05	19.01
affiliation=NEQ	959	245.39	18.91
position=NEQ	885	227.44	17.93
affiliation=NEQ, position=NEQ	868	220.01	17.36
affiliation=NEQ, track=EQ	428	158.18	16.22
position=NEQ, track=EQ	392	145.7	15.71
gender=NEQ	705	182.5	15.43
affiliation=NEQ, position=NEQ, track=EQ	381	139.92	15.2
gender=NEQ, track=EQ	312	123.84	14.01
affiliation=NEQ, gender=NEQ	669	160.01	13.2
gender=NEQ, position=NEQ	627	152.02	12.89
affiliation=NEQ, gender=NEQ, position=NEQ	614	145	12.1
gender=EQ	299	257.69	11.91
gender=EQ, track=EQ	144	189.02	11.75
affiliation=NEQ, gender=NEQ, track=EQ	289	102.15	11.35
affiliation=NEQ, gender=EQ, track=EQ	139	179.23	11.25
affiliation=NEQ, gender=EQ, position=NEQ, track=EQ	120	179.59	11.13
gender=EQ, position=NEQ, track=EQ	123	180.46	11.06
affiliation=NEQ, gender=EQ	290	252.35	11.01
affiliation=EQ, track=EQ	28	298.74	11

Complex Data allows Complex Descriptions

[Centeio-Jorge et al., EPIA 2019]

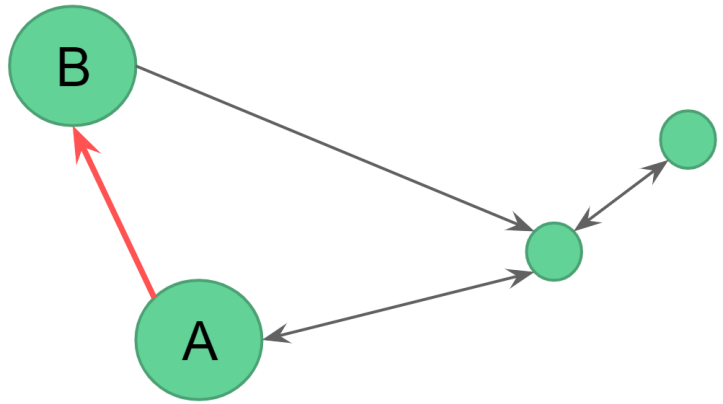


FIGURE 1 Example of Compositional Digraph. Let's assume that node A has the attributes {Gender = M, Age = 1} and node B {Gender = F, Age = 2}. Then, the directed edge from A to B, in red, has the following properties: {Gender = (M,F), Age = > } for simple version, {Gender = M, Age = 1} for from-node version and {Gender = F, Age = 2} for to-node version. For the sake of interpretability, we describe the subgroups (of edges) with these attributes as { Gender=A $\rightarrow$ Gender=M $\wedge$ Age=lower $\rightarrow$ Age=higher} in the simple version.

[Centeio-Jorge et al., EPIA 2019]

Socio-Spatial Playground Data

R	V	Pattern	Size	E	$\|L\|$	Z
1	comp	Gender=(M → M)	9	51	21.1	28.6
2	comp	Gender=(F → F)	9	50	15.4	19.5
3	comp	Emotion=(higher → lower)	18	73	7.4	3.9
4	comp	Hyper=EQ	18	72	7.3	3.7
5	comp	Conduct=(lower → higher)	18	74	7.2	3.4
6	comp	Age=(higher → lower)	18	85	7.9	3.2
7	comp	Emotion=(lower → higher)	18	74	7.0	3.1
8	comp	ProSoc=EQ	18	69	6.5	2.8
9	comp	Conduct=(higher → lower)	18	75	6.8	2.8
10	comp	Peer=EQ	18	105	9.0	2.8
1	to	Conduct= low ∧ Peer=low ∧ Hyper=low	18	174	1.5	2.0
2	to	Age=medium ∧ ProSoc=medium ∧ Emotion=low	17	184	1.6	2.0
3	to	Age=medium ∧ Emotion=low	17	184	1.6	1.9
4	to	Age=Medium ∧ Conduct=low	16	157	1.4	1.7
1	from	Peer=low $_{\wedge}$ Age=high ∧ Hyper=low	18	135	1.3	2.8
2	from	Peer=low $_{\wedge}$ Emotion=low ∧ ProSoc=low	18	158	1.4	2.4
3	from	Age=high $_{\wedge}$ Hyper=low	18	135	1.3	2.2
4	from	Gender = M ∧ Emotion=low ∧ Hyper=low	18	147	1.3	2.2

MinerLSD algorithm
[Atzmueller et al. 2019]

Exceptional Pattern Mining on Attributed Graphs

- Large & complex datasets ➔ efficient & effective methods
- "A Plan for Efficient Pattern Mining" ➔ **MinerLSD algorithm**
  - Closed patterns: Restricting the pattern enumeration
  - Graph abstractions, e.g. **k-cores** ➔ **interesting structures**
  - Pruning the search space: Optimistic estimates/interestingness measure

Basic methodology

- Support set (aka extension) based pattern mining : Enumerating closed patterns
- Reducing a pattern subgraph to its core subgraph : abstract support set

Support-Closed Pattern Mining

- Attributed graph G : each vertex has a label in a pattern language
- Pattern t : a constraint on the label $\Rightarrow$ the pattern t subgraph.
- Core subgraph : unique maximal subgraph satisfying some topological property P

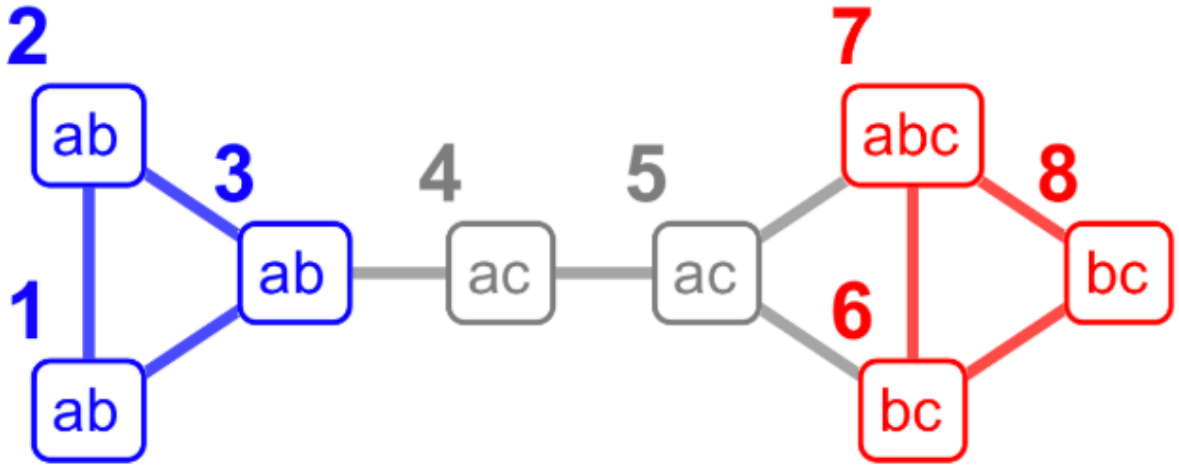
Let L be a lattice, V a set of objects v described as $d(v) \in L$

Definition (Support-closed patterns)

- $\mathrm{ext}(t) = \{v \in V | t \text{ occurs in } v\}$ is the support set of t in V
- $t \equiv t'$ iff $\mathrm{ext}(t) = \mathrm{ext}(t')$ and for each equivalence class
  The largest pattern with support set $\mathrm{ext}(t)$ is $f(t) = \mathrm{int} \circ \mathrm{ext}(t)$
  where
  $\mathrm{int}(X) = \bigwedge_{v \in X} d(v)$ is the greatest element of L which occurs in X

- **f is a closure operator and $c = f(t)$ is a closed pattern**

Example: Abstract Closed Pattern Mining



$P(v, X)$: **In G_X any vertex v belongs to a triangle** xwk

- 123457 = support set $\mathrm{ext}(a)$ of a
- G_{123457} = pattern a subgraph, G_{123}= pattern a core subgraph
- 123 $= p \circ \mathrm{ext}(a)$ = abstract support set of a
- $ab = \mathrm{int}(123) = ab \cap ab \cap ab$ is the largest pattern in the core

ab is an abstract closed pattern

Impact of Closed Patterns

Data / #c	0.005	0.01	0.02	0.03	0.05	0.15
S50	83					
#lmeSD	493	493	357	326	259	83
#lme	83	83	77	72	67	36
CoExp	196					
#lmeSD	1232895	991231	806911	468991	285183	77823
#lme	178	166	150	133	114	64
DBLP.P	2396					
#lmeSD	148	32	18	8	4	2
#lme	34	22	15	9	5	3
Lawyers	3221					
#lmeSD	3021675	1535949	677089	420699	168689	10339
#lme	2929	2512	1970	1640	1146	295
DBLP.C	14820					
#lmeSD	179	65	23	15	6	0
#lme	179	66	24	16	7	1

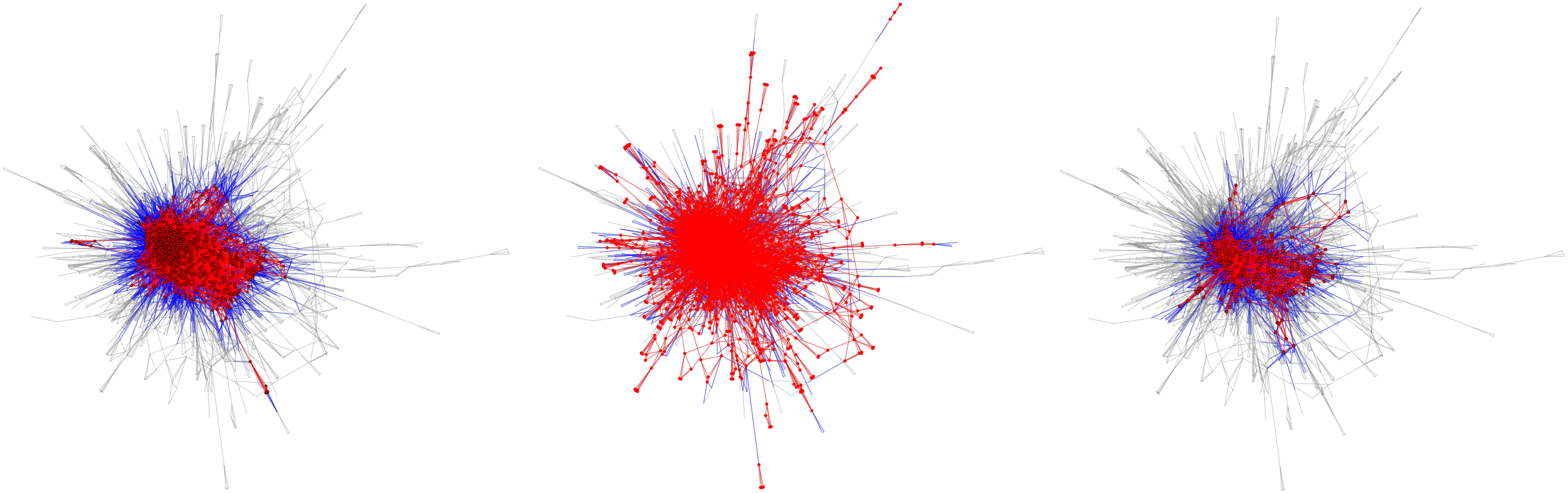


Fig. 8 Illustrative patterns (DBLP.C). Left: 5-core empty pattern with a local modularity of MODL = 0.1223; middle: 3-core empty pattern with a local modularity of MODL = 0.0430; right: 3-core "mine" pattern with a local modularity MODL = 0.0503. In the plots, red color indicates the core graph (i. e., the in-edges of the pattern), blue color shows the edges incident to the nodes of the core graph, gray depicts the edges of the rest of the graph.

Results

MinerLSD algorithm [Atzmueller et al. 2019]

DBLP.S	1-core	#c ≥ 3457143		time = STOPPED AFTER 36h			
l	0.005	0.01	0.02	0.03	0.04	0.05	0.15
#lme	1150	351	103	50	26	18	1
#lm	778	230	68	25	12	6	0
time (s)	59989	37645	24906	20634	17299	16167	8332
	2-core	#c ≥ 3584834		time = STOPPED AFTER 36h			
l	0.005	0.01	0.02	0.03	0.04	0.05	0.15
#lme	958	303	94	44	24	16	1
#lm	722	218	64	24	12	6	0
time (s)	36302	25949	19065	16068	13869	12907	7073
	3-core	#c = 1576164		time = 45720			
l	0.005	0.01	0.02	0.03	0.04	0.05	0.15
#lme	621	208	72	28	17	9	1
#lm	533	165	49	20	9	6	0
time (s)	19799	15531	12329	10221	9149	8276	5143
	5-core	#c = 44345		time = 3791			
l	0.005	0.01	0.02	0.03	0.04	0.05	0.15
#lme	200	71	26	10	6	4	1
#lm	180	59	21	7	3	2	0
time (s)	4410	3760	3173	2877	2709	2533	2044
	7-core	#c = 5659		time = 881			
l	0.005	0.01	0.02	0.03	0.04	0.05	0.15
#lme	62	24	10	4	2	1	1
#lm	62	23	10c	3	1	1	0
time (s)	1005	908	812	784	756	687	689

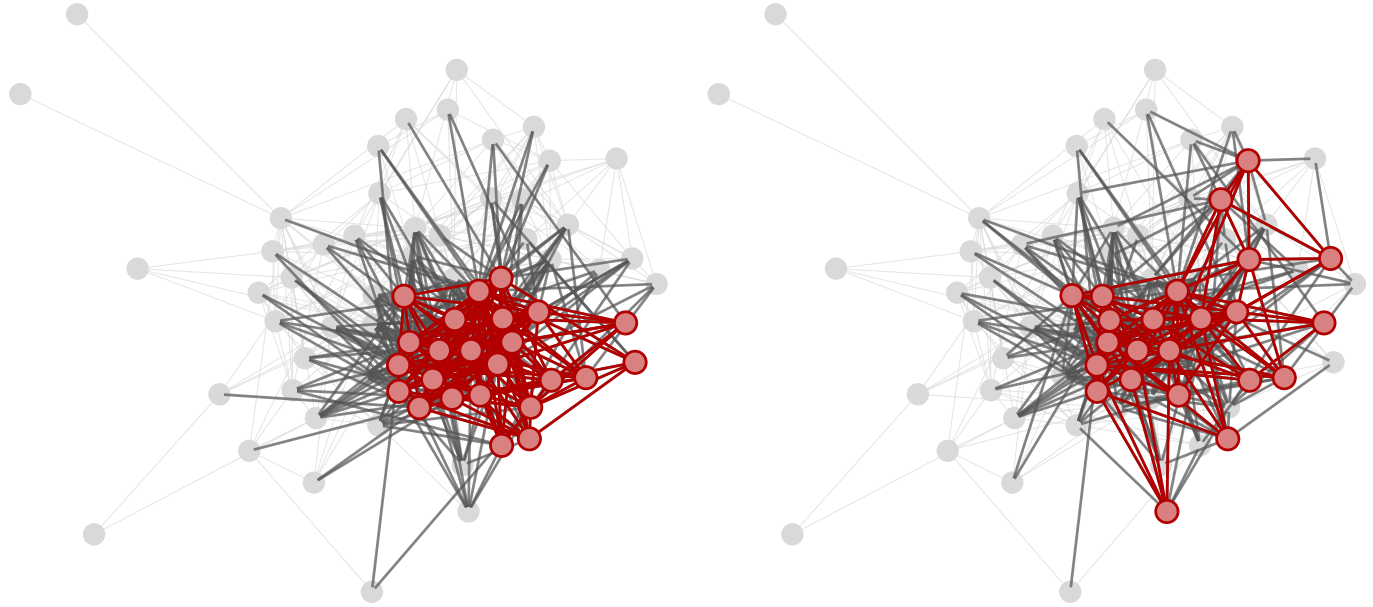
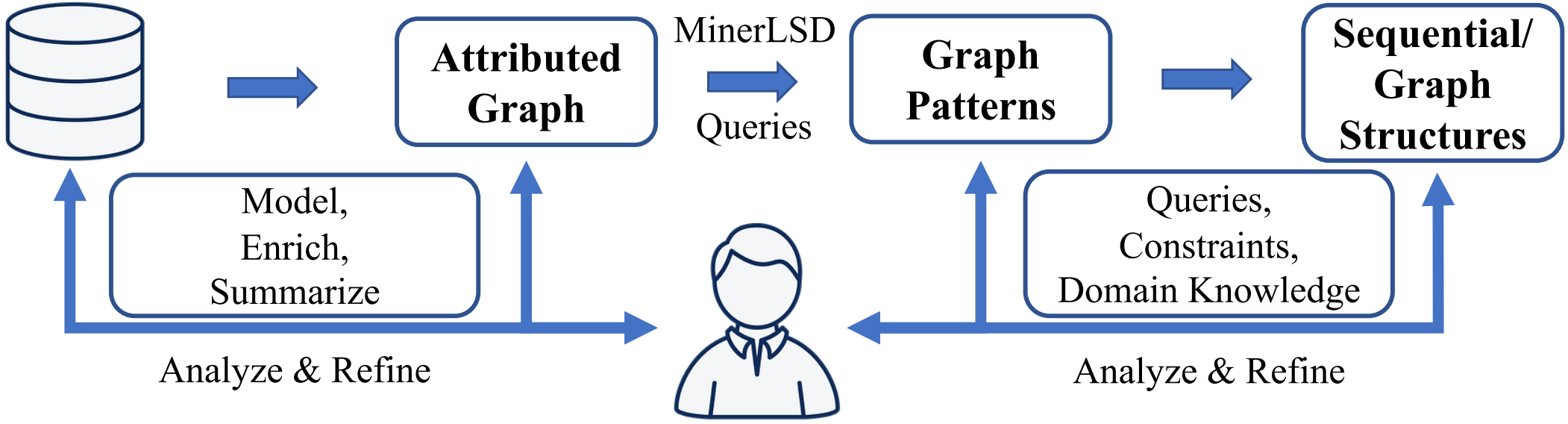


Fig. 7 Example patterns from the Lawyers dataset: Both patterns are similar 5-cores, with a Jaccard similarity considering the nodes of the respective pattern-induced subgraphs of 0.52. The pattern on the left (described by $35 < Age \leq 65$ AND $Seniority < 5$ AND $Status = Partner$, with $size = 24$) is considerably denser with a local modularity of MODL = 0.058, compared to the pattern on the right (described by $Age < 40$ AND $Seniority \leq 30$, with $size = 23$) which only has a local modularity of MODL = 0.013. In the figures, we depict in red the edges and the vertices in the pattern extension, in gray the out-edges of the pattern (i. e., one vertex of a gray edge is contained in the pattern subgraph and the other vertex is not) and in light gray the rest of the graph.

Interactive Graph Summarization and Exploration

[Atzmueller, Bloemheuvel, Kloepper. 2019]



event_timestamp	message title	description	line	cell	robot
2014-6-3 12:29:37 10011	Motors ON State	Motor on . . .	13	2	48
2014-6-3	Safety Guard Stop State	Entering guard state . . .	13	2	48
2014-6-3 12:29:36 10010	Motors OFF State	Motor off . . .	13	2	7
2014-6-3 12:29:39 20205	Auto Stop Open	Entering auto stop . . .	13	2	48

Data/Event Log

Interactive Graph Summarization & Exploration: Examples/Patterns/Subgraphs

Pattern: *Warning AND cell6 and line20*

➔ induces a specific subgraph.

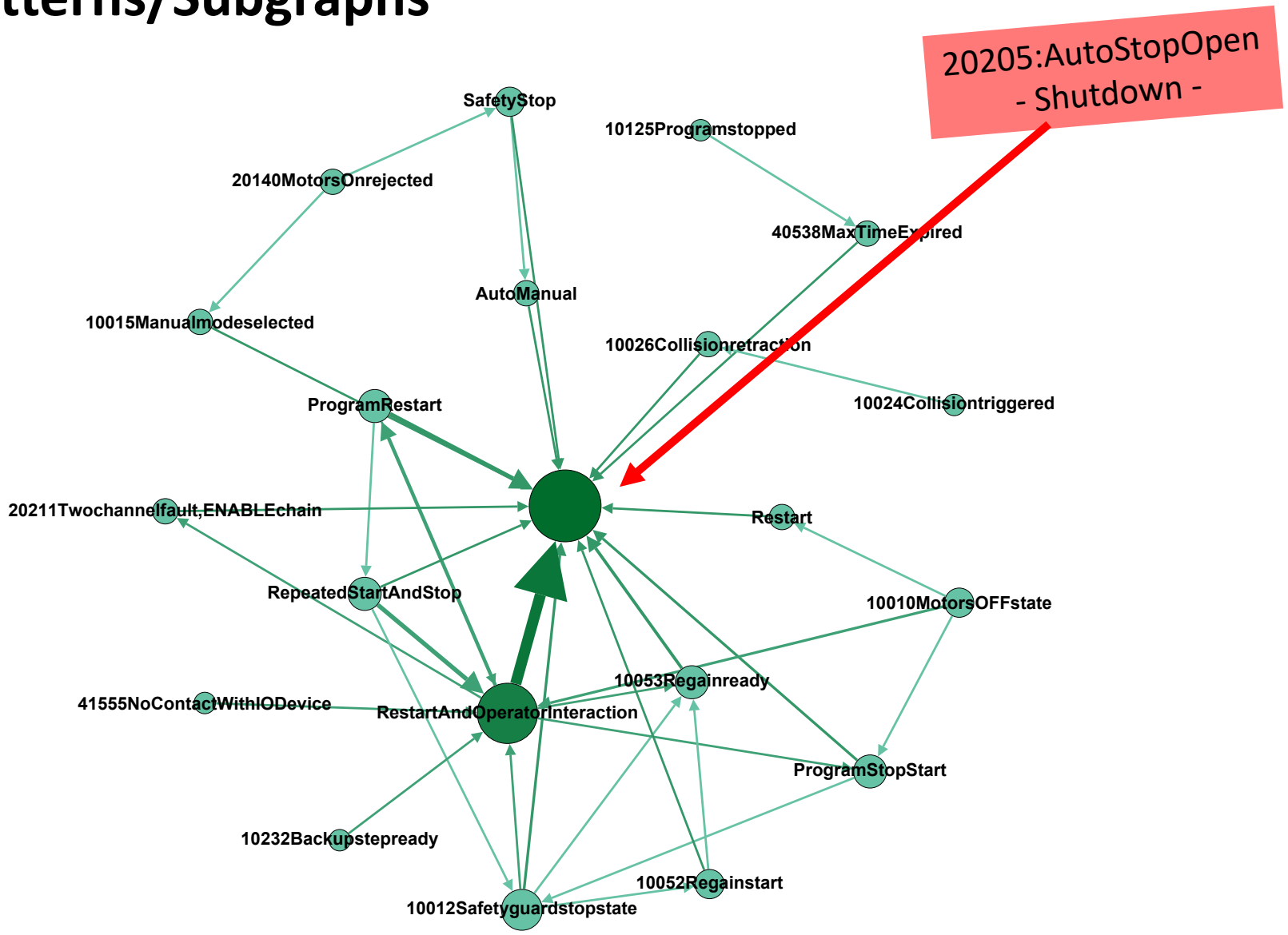
Interesting sequences extracted from the subgraph

Sequence	Prob.
71414:Concurrentchangesofsignalvalue → 80002:UserDefinedEvent3	6e-7
71414:Concurrentchangesofsignalvalue → 80002:UserDefinedEvent4	6e-7
20314:Enable2supervisionfault → 40538:MaxTimeExpired → 80002:UserDefinedEvent3	2e-9
20481:SCOVRactive → 40538:MaxTimeExpired → 80002:UserDefinedEvent3	2e-8
20481:SCOVRactive → 40538:MaxTimeExpired → 80002:UserDefinedEvent4	7e-9

Problems/(Root) Causes

Maintenance/Troubleshooting

Interactive Graph Summarization & Exploration: Examples/Patterns/Subgraphs



Software

- VIKAMINE:
  http://www.vikamine.org/

  Can also be applied using the “Java kernel"

- rsubgroup – R package:
  A wrapper around the VIKAMINE kernel
  http://www.rsubgroup.org/

- MinerLC/MinerLSD:
  Closed pattern miners in attributed graphs
  https://lipn.univ-paris13.fr/MinerLC/

Summary

- Subgroup discovery
  - A powerful approach for interpretable models
  - For interpretable machine learning
- Can be applied for simple to complex models (and complex data)
- Examples:
  - Feature generation in spatial contexts, outlier detection
  - Attributed graphs – modeling complex multi-relational structures
  - Targeted quality measures
- Outlook:
  - Integration of domain knowledge
    ➔ Combination with declarative approaches
    ➔ Integration into "hybrid" explanation approaches

Interpretable Machine Learning using Pattern Mining

Martin Atzmueller

CSLab

Tilburg University

JADS Jheronimus Academy of Data Science