Comparing Apples and Oranges

measuring differences between exploratory data mining results

Nikolaj Tatti & **Jilles Vreeken**

FWO VLAANDEREN

Universiteit Antwerpen

Question of the day

How can we decide whether
different results from **different algorithms**
provide significantly **different information**?

Why?

Suppose one dataset

- analyst 'Jaakko' applies clustering
- analyst 'Jilles' applies pattern set mining

How can Jaakko and Jilles compare their results?

- clearly, a clustering ≠ a set of patterns

More why

Goal of data mining is novel insight

- no way we can run all mining algorithms
- no way we can analyse all results

Data mining is iterative

- what method should we apply next? *or*
- what result should we analyse next?

Hence, we need to **measure** how different results are

However

No objective function for ‘insight’

Results are complex objects

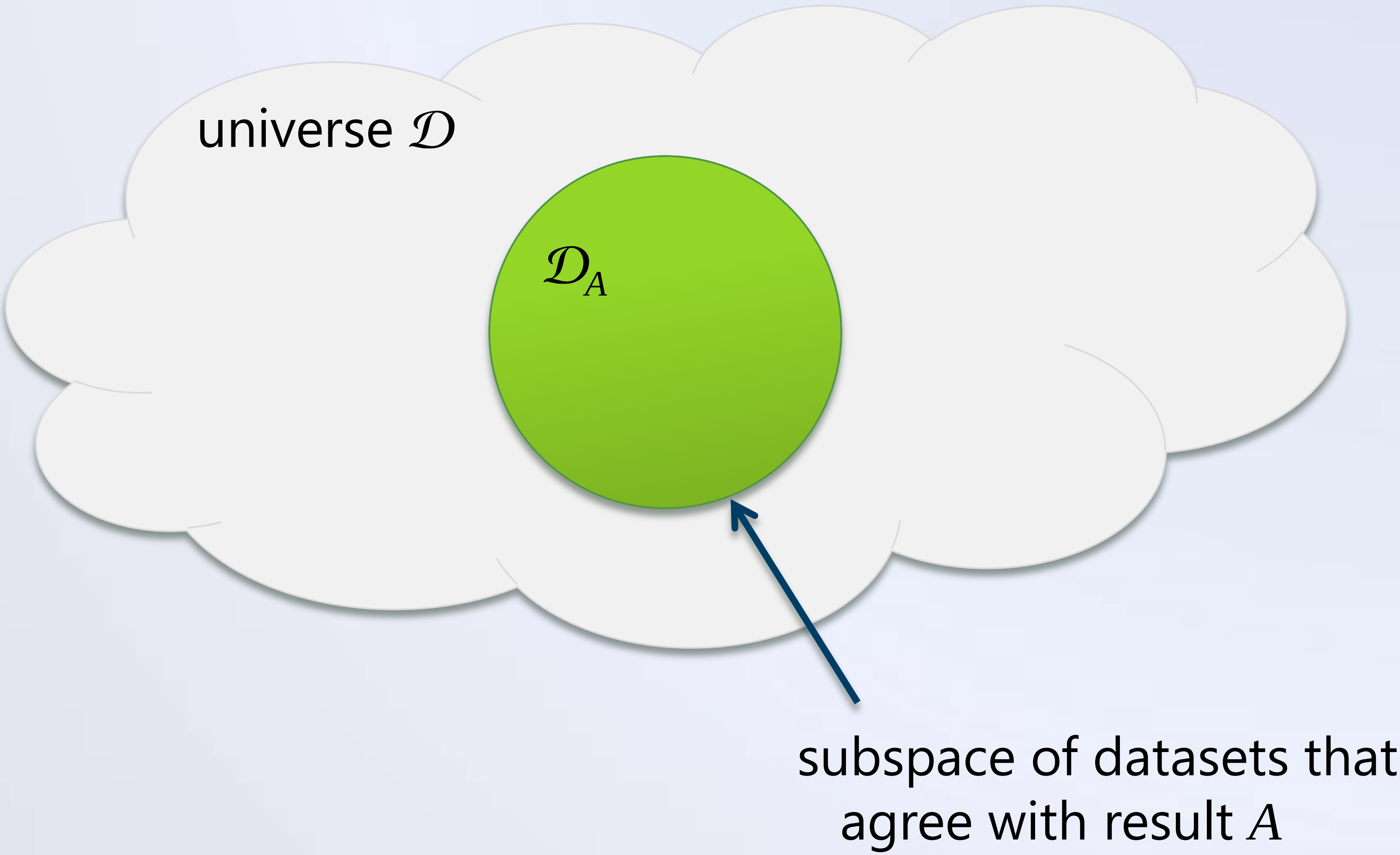
- hard to define a generic distance
- like comparing apples to oranges

We need a common language

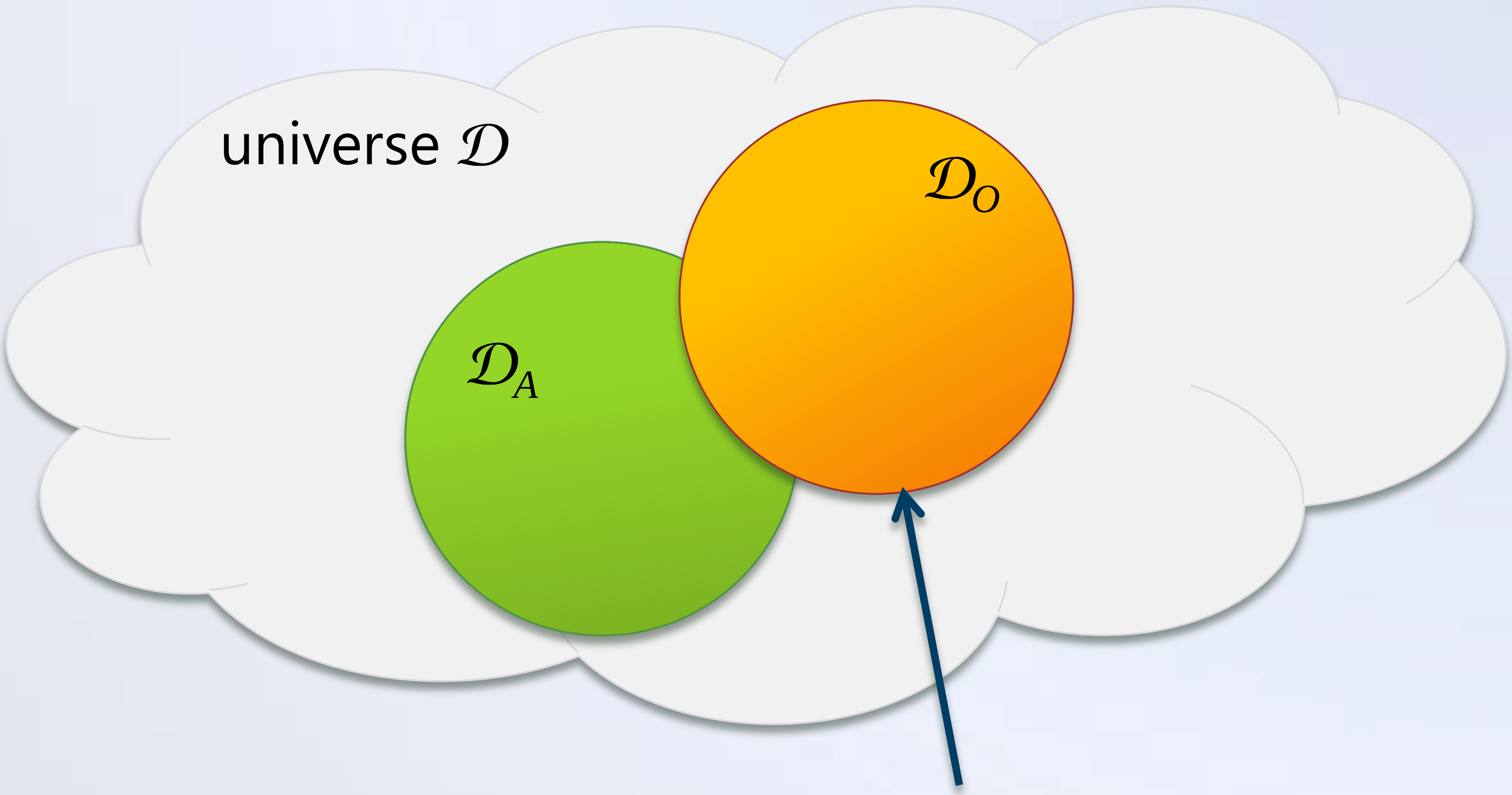
Towards measuring shared information

universe $\mathcal{D}$ of possible datasets

Towards measuring shared information

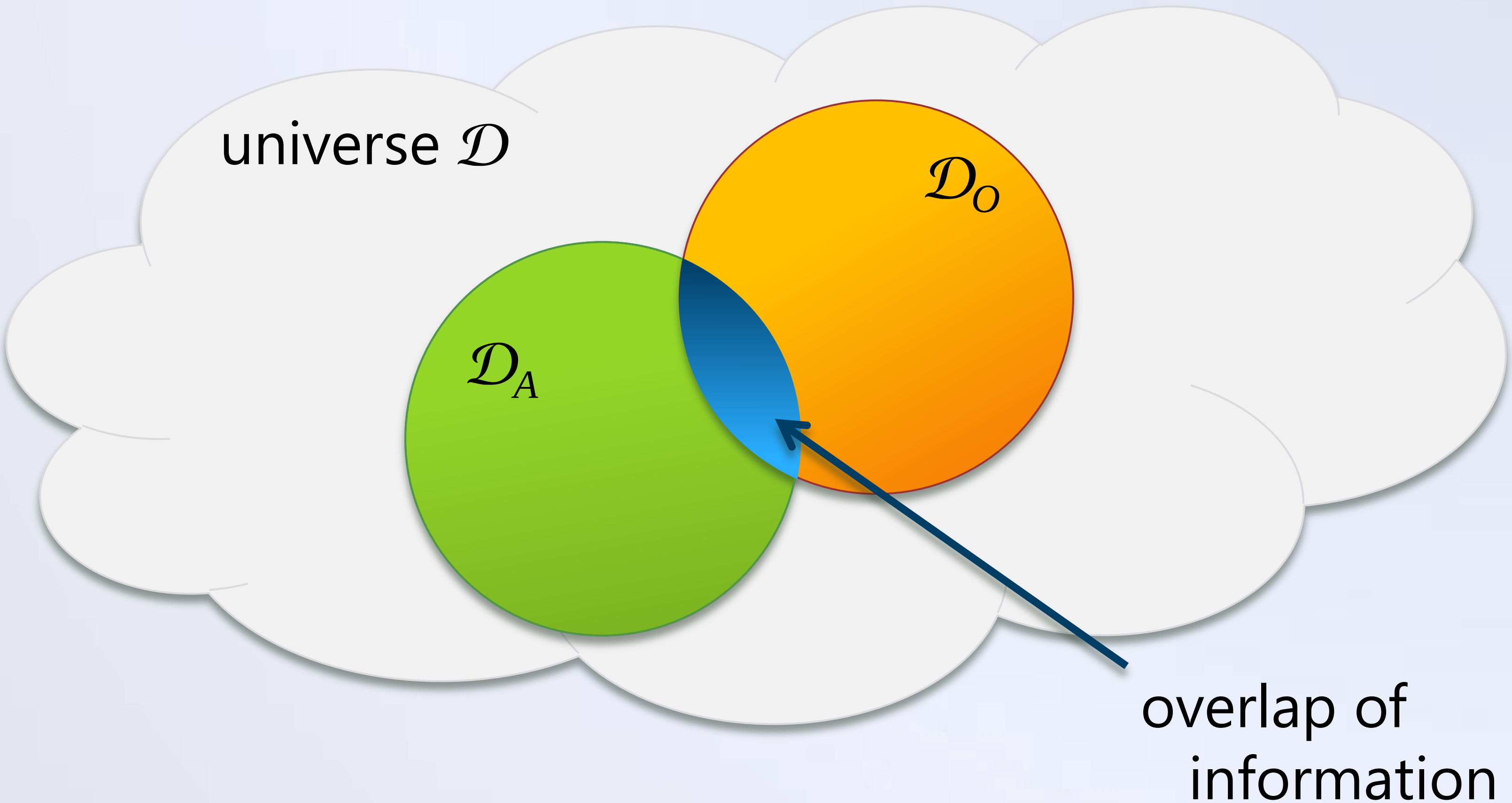


Towards measuring shared information



subspace of datasets that agree with result O

Towards measuring shared information



A bit more formal

Observation

a result R holds for a subset $\mathcal{D}_R$ of all possible datasets $\mathcal{D}$

R implies that some $D \in \mathcal{D}$ are more likely than others.

So, results implicitly define distributions over **datasets**

similar distributions ↔ same information

and hence…

comparing results → comparing distributions

the larger the overlap, the more shared information

The Big Question

How do we measure this overlap?

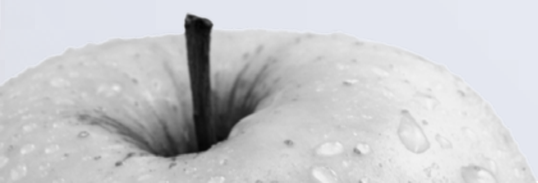
1. translate results into distributions
2. use Information Theory to measure amount of shared information

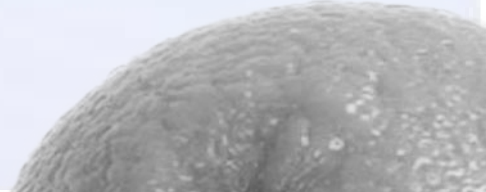
How it works for binary data

We show how to do it for binary data

1. translate results into sets of (noisy) tiles
2. infer Maximum Entropy model from tile set
3. use Kullback-Leibler to build our measure

$$KL(p \,\|\, q) = \sum_{D \in \mathcal{D}} p(D) \log \frac{p(D)}{q(D)}$$





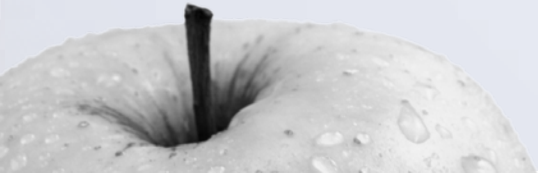
How it works for binary data

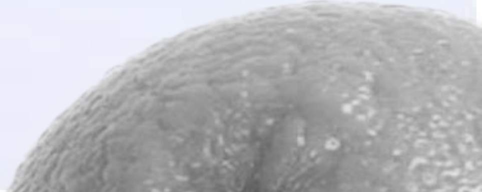
1. translate results into sets of tiles

Indicate **what parts** of the data show **what structure**

Many results on 0/1 data can be reduced to noisy tiles

- **noisy tile** attributes and tids, density of 1s
- **exact tile** attribute and tids with density 0% or 100%





How it works for binary data

1. **translate results into sets of tiles**

itemsets and alike naturally translate to tiles,
as do **boolean matrix factorizations**

so can **clusterings**
k-means with l_1 distance, centroids on 0/1 data:
for rows in the cluster, avg. density per attribute

and so does **subspace clustering**

How it works for binary data

2. infer Maximum Entropy model for tile set

MaxEnt: the most unbiased probabilistic model

model:

$$\begin{matrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{matrix} \qquad \begin{matrix} 1 & 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 1 & 1 & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{6} & 1 & 1 & 1 \\ \frac{1}{6} & \frac{1}{6} & 1 & 1 & 1 \end{matrix} \qquad \begin{matrix} 1 & 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 1 & 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{matrix}$$

tile set: empty — 2 exact & 1 tile of *fr* 1/2 — 4 exact tiles

(MaxEnt formalized for 01 datasets by De Bie, 2011)

Background knowledge

What **you** already know determines what is informative to **you**

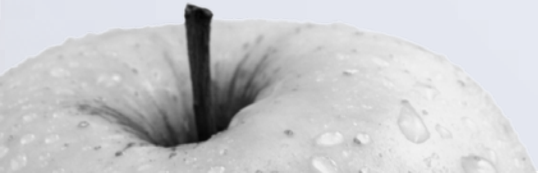
We allow to easily incorporate background knowledge such as tiles, row and/or column margins in our measure

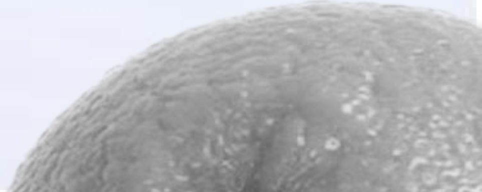
Our measure

Given tile sets T_1 and T_2, and background knowledge tile set B, with $M = T_1 \cup T_2 \cup B$

$$d(T_1, T_2; B) = \frac{KL(M \,\|\, T_1 \cup B) + KL(M \,\|\, T_2 \cup B)}{KL(M \,\|\, B)}$$

(for **exact** tiles, d coincides with Jaccard dissimilarity)





Our measure

We can use our measure to

- visualise the big picture between methods
- redescribe between (partial) results
- mine data iteratively

Experiments

We applied 10 different algorithms on 4 real datasets, for 4 different backgrounds

6 Pattern Set Miners

Asso (Miettinen et al.)

Hyper (Fuhry et al.)

Inf-Th. Tiling (Kontanasios et al.)

KRIMP (Siebes et al.)

MTV (Mampaey et al.)

Tiling (Geerts et al.)

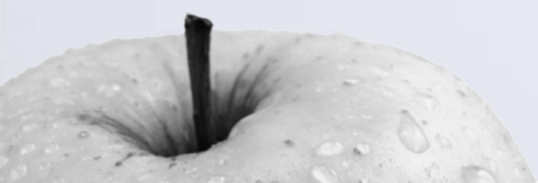
4 Clusterers

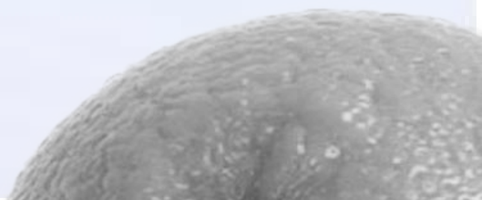
***k*-means** (MacQueen)

bi-clustering (Puolomäki et al)

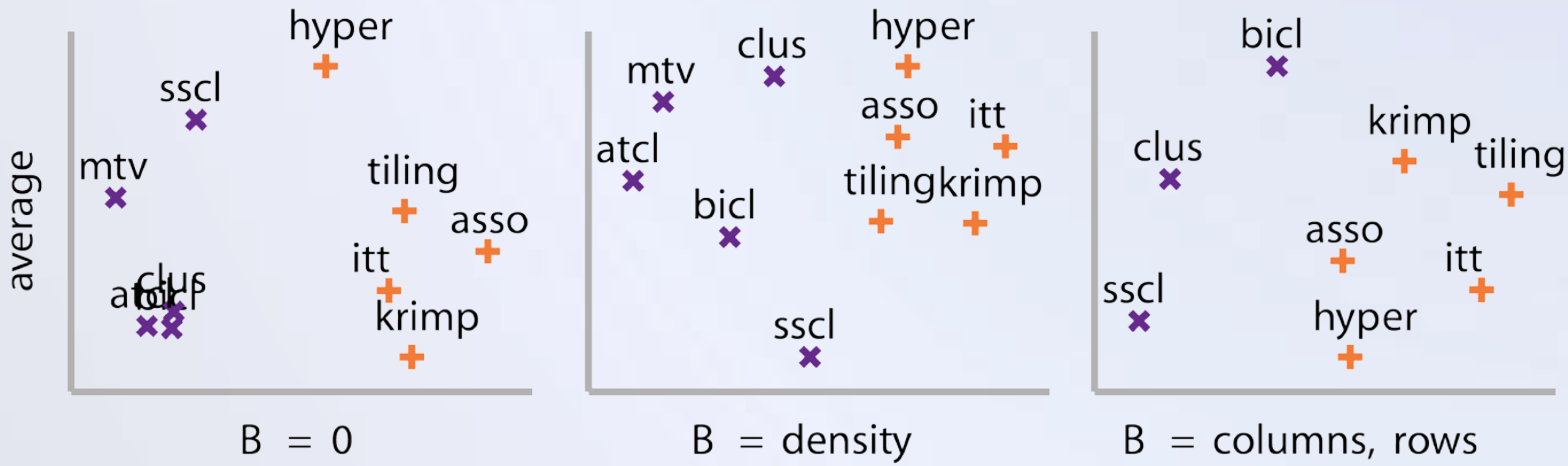
attr. clus. (Mampaey et al.)

proClus (Aggarwal et al.)





The big picture



Redescribing results

KRIMP

association rule
significantly outperform
high dimension
experiment evaluation show
vector support machine

ASSO (*d*=0.83)

association rule mine algo.
vector method support
algo. method high dimension
algo. show

INF-TH. TILES (0.77)

vector support machine
association rule
dimension
outperform

Conclusions

Comparing results is an important,
yet understudied aspect of data mining

We propose to regard **information content** to
meaningfully compare apples and oranges

We give an example for 01 data

- translate results into sets of tiles
- build a global model
- use information theory to measure differences

Conclusions

Our measure allows for

- visualisation of the big picture between methods
- redescription between (partial) results
- and enables iterative data mining

Future work includes

- richer and structured data/pattern types
- consider other translations into distributions
- applying the distance in real-world data mining

Thank you!

Our measure allows for

- visualisation of the big picture between methods
- redescription between (partial) results
- and enables iterative data mining

Future work includes

- richer and structured data/pattern types
- consider other translations into distributions
- applying the distance in real-world data mining