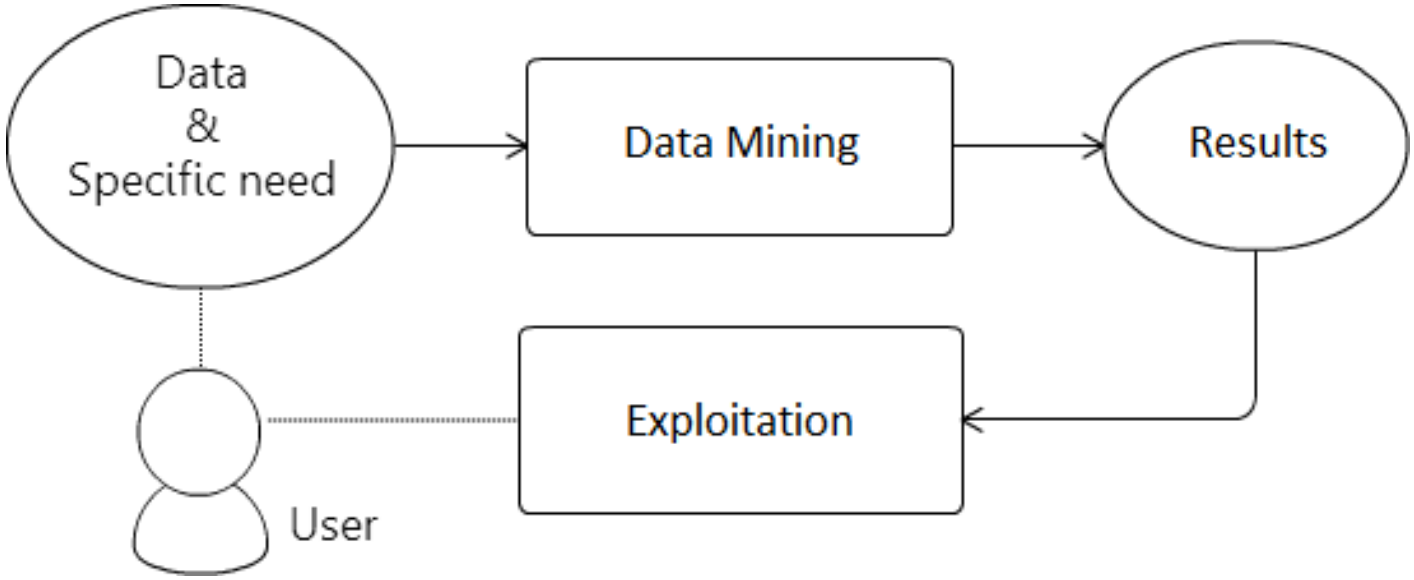
A dissimilarity approach to the characterization of datasets

William Raynaut

Chantal Soule-Dupuy, Nathalie Valles-Parlangeau
Cedric Dray, Philippe Valet

Introduction



Data analysis is a **complex** process.
The typical user needs **assistance**.

$\Rightarrow$ many **Intelligent Discovery Assistants (IDA)**

- Expert systems

  [Sleeman et al., 1995]
- Meta-learning methods

  [Kalousis, 2002, Giraud-Carrier et al., 2004, Sun and Pfahringer, 2013]
- Case bases

  [Goble et al., 2010]
- Heuristic search methods

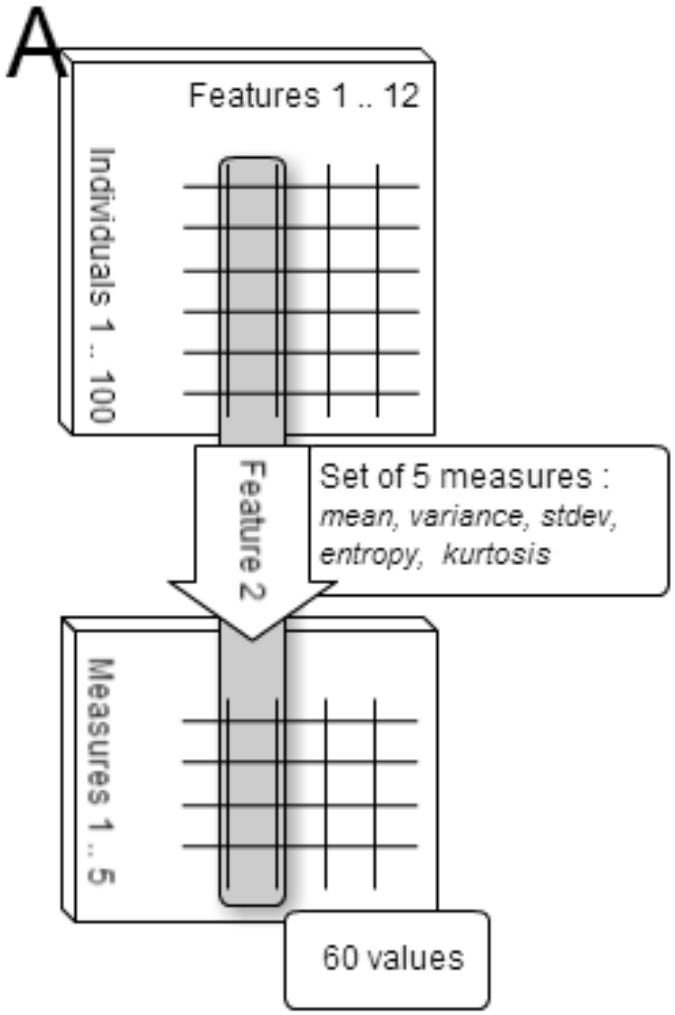
  [Escalante et al., 2009, Sun, 2014]
- DM workflows planners

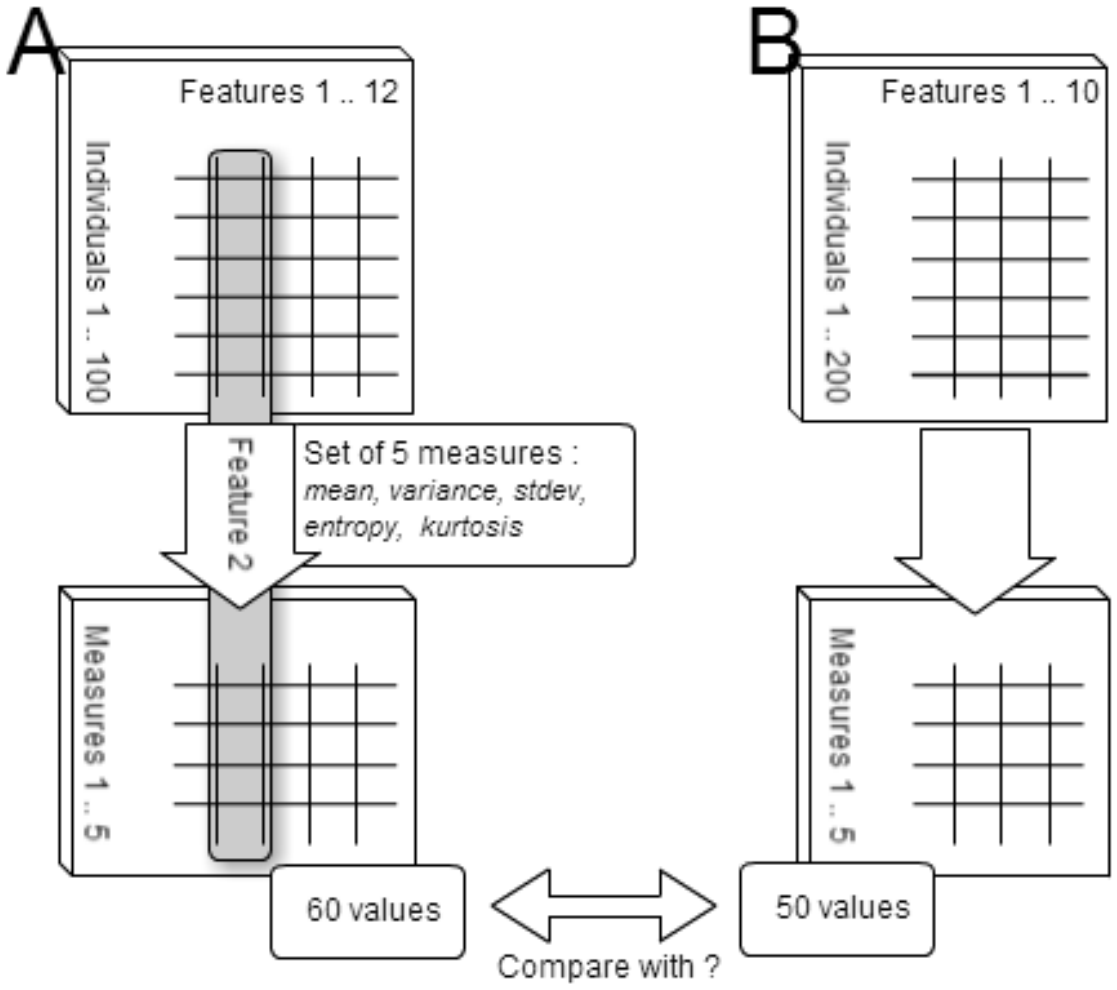
  [Zakova et al., 2011, Serban, 2013, Nguyen et al., 2014]

- IDAs help the user design processes or workflows that are **appropriate** for his **dataset** and objectives.
- This implies a **characterization** of the user's dataset that fits the particular mechanics of the IDA.

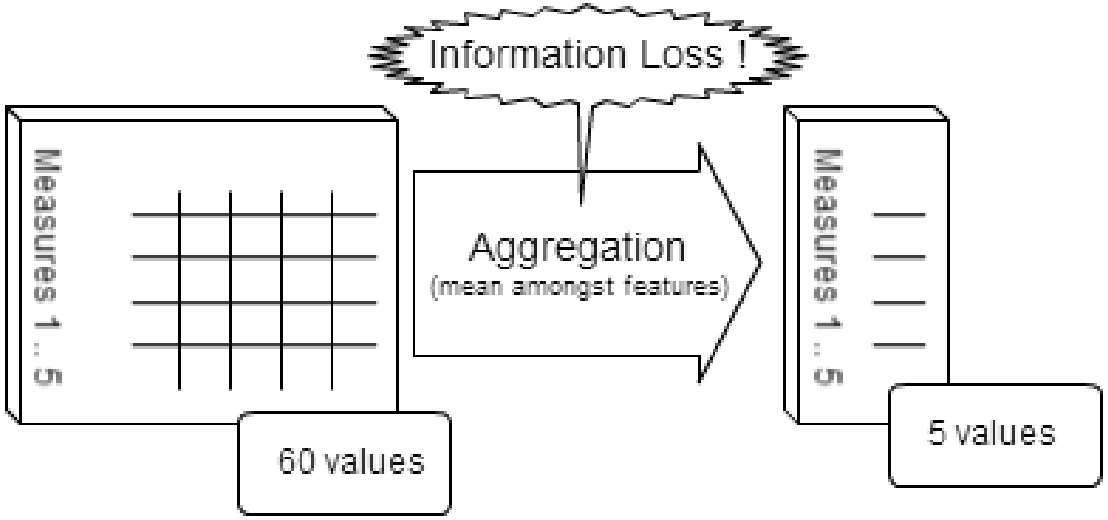
$\rightarrow$ We investigate the possibility that limitations in the classical characterization of datasets are among the main obstacles to well performing IDAs

Dataset characterization

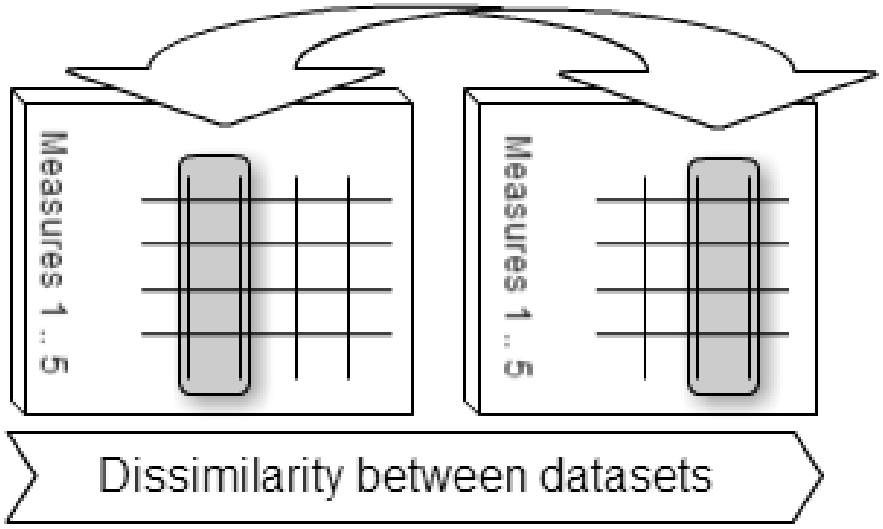



A
Features 1 .. 12
Individuals 1 .. 100
Feature 2
Set of 5 measures :
mean, variance, stdev,
entropy, kurtosis
Measures 1 .. 5
60 values
B
Features 1 .. 10
Individuals 1 .. 200
Measures 1 .. 5
50 values
Compare with ?

The usual solution is to average the measures along the features.



We propose to compare the features by most similar pairs.



Dissimilarity function

$$d_{\omega}^{ubr}(x,y) = \frac{d_{G(\omega)}^{ubr}(x,y)}{\max\limits_{(x',y')\in\omega^2}(d_{G(\omega)}^{ubr}(x',y'))} + \frac{d_{F(\omega)}(x,y)}{\max\limits_{(x',y')\in\omega^2}(d_{F(\omega)}(x',y'))}$$

The proposed dissimilarity take into account :

- Global properties of the datasets such as the number of instances and attributes, or the performance of landmarkers (simple learners) on them...
- Properties of the individual features of the datasets, such as entropy or kurtosis...

And compares the features of the datasets by most similar pairs.

Theoretical validation

- We use the *strong* (ϵ, γ)-*goodness* from [Wang et al., 2009]
- Characterizes the usefulness of a dissimilarity function for learning a concept from a dataset (here we try to learn the *appropriateness* of classifiers to different datasets)
- Gives upper bound of error rate for a simple classifier using the dissimilarity

	200 examples		1000 examples		5000 examples	
	δ	error bound	δ	error bound	δ	error bound
d_ω^{ubr}	0,973	1,023	0,871	0,921	0,501	0,551
Euclidean based	0,989	1,039	0,945	0,995	0,755	0,805
Manhattan based	0,990	1,040	0,952	1,002	0,783	0,833

	10000 examples		20000 examples		50000 examples	
	δ	error bound	δ	error bound	δ	error bound
d_ω^{ubr}	0,251	0,301	0,063	0,113	0,001	0,051
Euclidean based	0,570	0,620	0,325	0,375	0,060	0,110
Manhattan based	0,613	0,663	0,375	0,425	0,086	0,136

Table: Error bound achievable with probability $1 - \delta$ by dissimilarity based classifiers for different numbers of examples

Experimental validation

Extracting data from OpenML :

- Accuracy of a set of classifiers over a range of datasets

	$classifier_1$	$classifier_2$	...	$classifier_{93}$
$dataset_1$	0.8	0.9	...	...
$dataset_2$	0.9	0.7	...	...
...	...	...	...	...
$dataset_{434}$	...	...	...	...

Experimental validation

Extracting data from OpenML :

- Accuracy of a set of classifiers over a range of datasets

	$classifier_1$	$classifier_2$	...	$classifier_{93}$
$dataset_1$	0.8	0.9	...	...
$dataset_2$	0.9	0.7	...	...
...	...	...	...	...
$dataset_{434}$	...	...	...	...

- Characterization of the datasets

	$NumberOfInstances$	$NumberOfFeatures$	...	$MetaAttribute_{105}$
$dataset_1$	100	62	...	...
$dataset_2$	5000	13	...	...
...	...	...	...	...
$dataset_{434}$	360	20	...	...

Building a Meta-dataset for a classification meta-problem

	$NumberOfInstances$	...	$MetaAttribute_{105}$	$Class$
$dataset_1$	100	...	4	$classifier_{18}$
$dataset_2$	5000	...	92	$classifier_7$
...	...	...	...	...
$dataset_{434}$	360	...	13	$classifier_{63}$

foreach *Dataset instance* $dataset_i$ **do**

- Exclude $dataset_i$ from the metadataset
- Apply an attribute selection algorithm on the metadataset
- Learn a classification model from the reduced metadataset using a classification algorithm
- Use this model to predict a class label $classifier_j$ for $dataset_i$

$x =$ actual criterion value achieved on $dataset_i$ by $classifier_j$

$best =$ best criterion value achieved on $dataset_i$ among the classifiers $classifier_{1..m}$

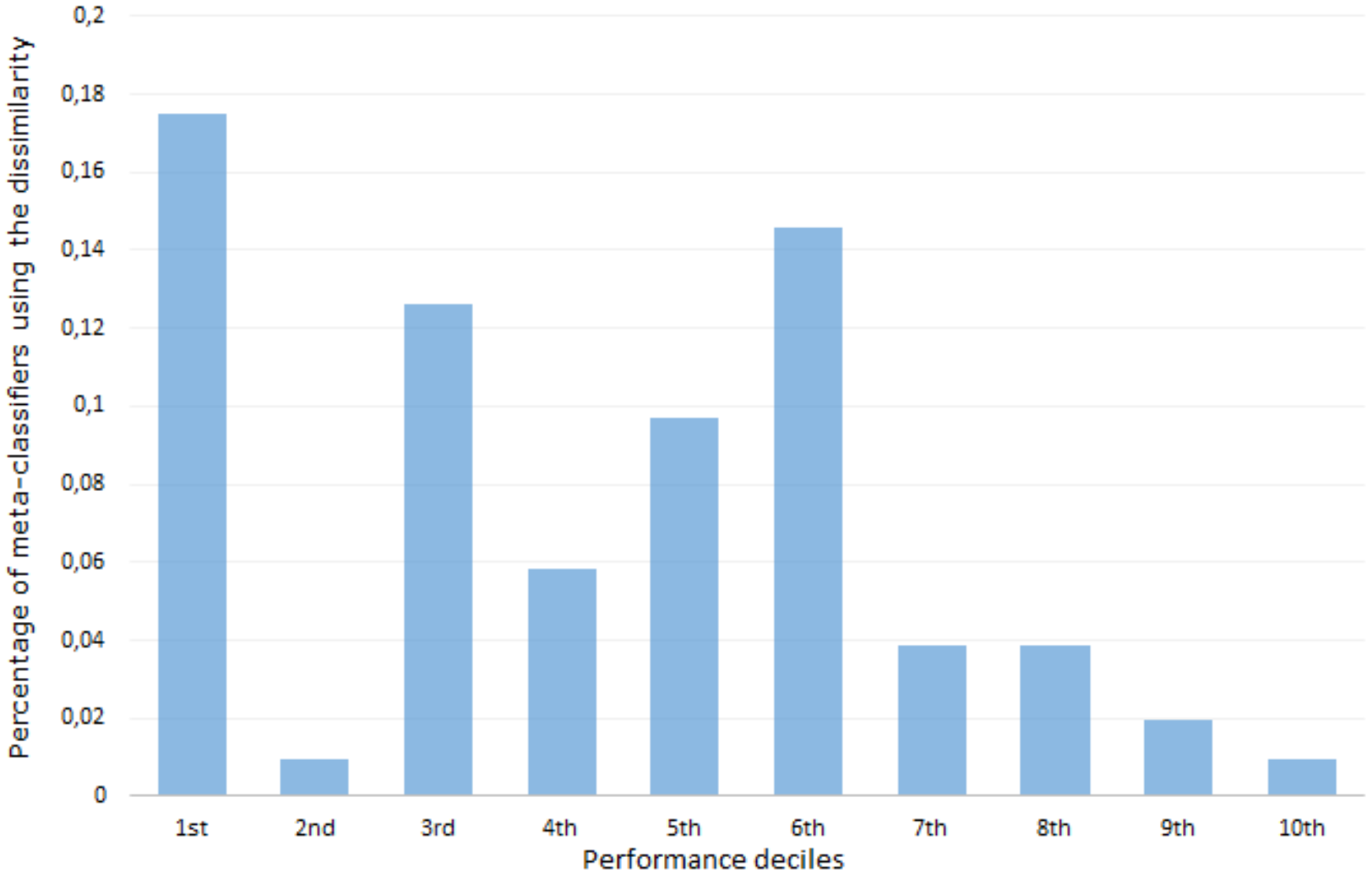
$def =$ actual criterion value achieved on $dataset_i$ by the default classifier (majority class classifier)

$$perf(run) = 1 - \frac{|best - x|}{|best - def|}$$

- 434 datasets
- 11 evaluation criteria
- $>$ 2600 (meta)-classifiers
- $>$ 60 (meta)-feature selection methods

$\Rightarrow$ over 200k experiments, 700M ML algorithms runs

How does the dissimilarity perform relatively to all other approaches ?


Percentage of meta-classifiers using the dissimilarity
0,2
0,18
0,16
0,14
0,12
0,1
0,08
0,06
0,04
0,02
0
1st
2nd
3rd
4th
5th
6th
7th
8th
9th
10th
Performance deciles

Contribution

- A dissimilarity accounting for the inner topology of the datasets attributes
- With theoritical guarantees when characterizing the appropriateness of classifiers
- Able to perform well in a meta-learning framework

Thank you for your attention

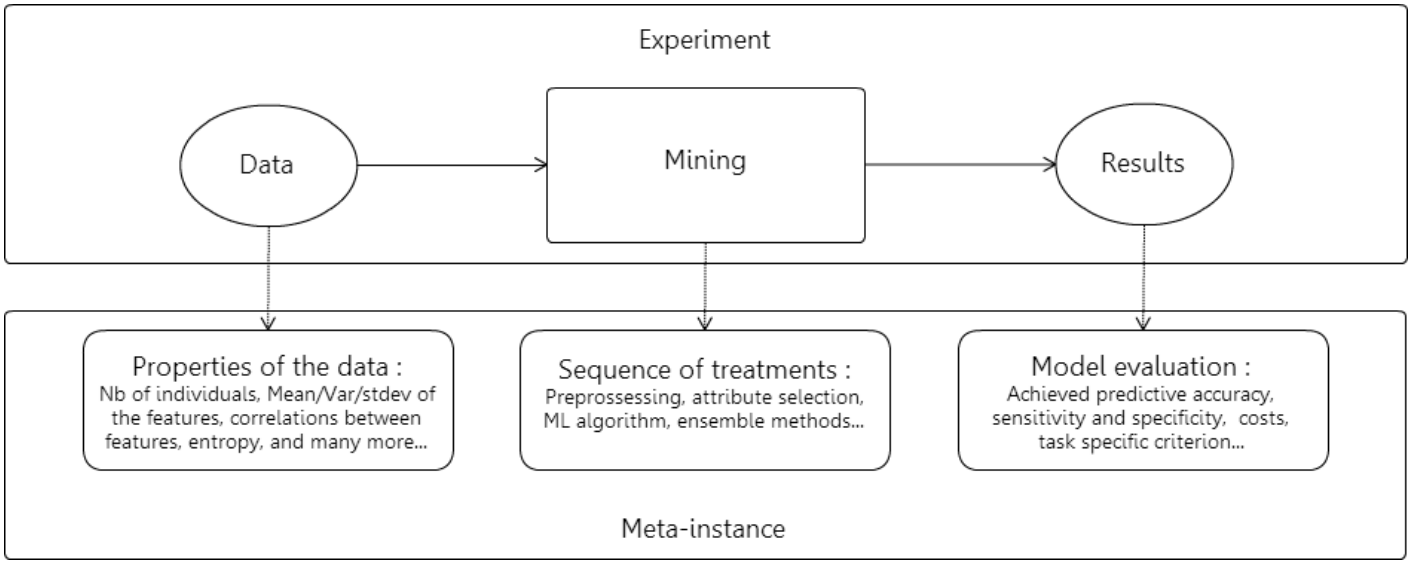
$$D = \{ \text{ classification datasets } \},\ A = \{ \text{ classifiers } \}$$

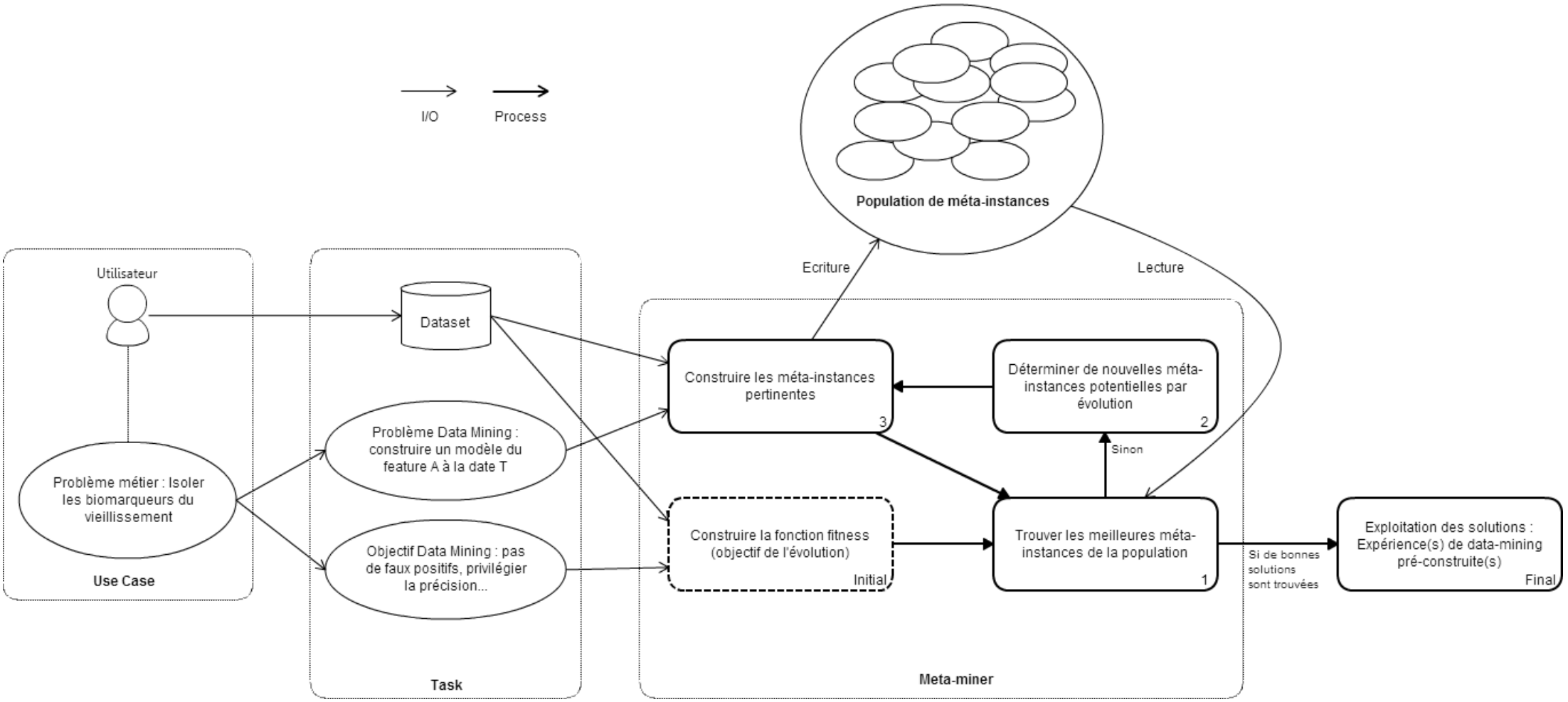
- We execute every classifier of A on every dataset of D and measure a performance criterion c of the resulting model.
- For each dataset x, we define the set A_x of the algorithms that are appropriate on this dataset along our performance criterion as those at most one standard deviation away from the best :

$$A_x = \{a \in A \text{ such that } |\max_{a' \in A}(c(a', x)) - c(a, x)| \leq \sigma_x\}$$

We can then consider, for each algorithm $a \in A$, the binary classification problem where instances are the datasets $x \in D$, and their class label stating whether a is appropriate on them.

1 Experiment = 1 Meta-instance




I/O
Process
Population de méta-instances
Ecriture
Lecture
Utilisateur
Dataset
Problème métier : Isoler les biomarqueurs du vieillissement
Problème Data Mining : construire un modèle du feature A à la date T
Objectif Data Mining : pas de faux positifs, privilégier la précision...
Use Case
Task
Construire les méta-instances pertinentes
3
Déterminer de nouvelles méta-instances potentielles par évolution
2
Sinon
Construire la fonction fitness (objectif de l'évolution)
Initial
Trouver les meilleures méta-instances de la population
1
Si de bonnes solutions sont trouvées
Exploitation des solutions : Expérience(s) de data-mining pré-construite(s)
Final
Meta-miner

References I

Escalante, H. J., Montes, M., and Sucar, L. E. (2009).
Particle swarm model selection.
The Journal of Machine Learning Research, 10:405–440.

Giraud-Carrier, C., Vilalta, R., and Brazdil, P. (2004).
Introduction to the special issue on meta-learning.
Machine learning, 54(3):187–193.

Goble, C. A., Bhagat, J., Aleksejevs, S., Cruickshank, D., Michaelides, D., Newman, D., Borkum, M., Bechhofer, S., Roos, M., Li, P., et al. (2010).
myexperiment: a repository and social network for the sharing of bioinformatics workflows.
Nucleic acids research, 38(suppl 2):W677–W682.

Kalousis, A. (2002).
Algorithm selection via meta-learning.
PhD thesis, Universite de Geneve.

Nguyen, P., Hilario, M., and Kalousis, A. (2014).
Using meta-mining to support data mining workflow planning and optimization.
Journal of Artificial Intelligence Research, pages 605–644.

Serban, F. (2013).
Toward effective support for data mining using intelligent discovery assistance.
PhD thesis.

References II


Sleeman, D., Rissakis, M., Craw, S., Graner, N., and Sharma, S. (1995).
Consultant-2: Pre-and post-processing of machine learning applications.


Sun, Q. (2014).
Meta-learning and the full model selection problem.
Unpublished PhD thesis, University of Waikato.


Sun, Q. and Pfahringer, B. (2013).
Pairwise meta-rules for better meta-learning-based algorithm ranking.
Machine learning, 93(1):141–161.


Wang, L., Sugiyama, M., Yang, C., Hatano, K., and Feng, J. (2009).
Theory and algorithm for learning with dissimilarity functions.
Neural computation, 21(5):1459–1484.


Zakova, M., Kremen, P., Zelezny, F., and Lavrac, N. (2011).
Automating knowledge discovery workflow composition through ontology-based planning.
Automation Science and Engineering, IEEE Transactions on, 8(2):253–264.