Inductive database to support iterative data mining: Application to biomarker analysis on patient data in the Fight-HF project

Emmanuel Bresso, Joao-Pedro Ferreira, Nicolas Girerd, Masatake Kobayashi, Grégoire Preud'homme, Patrick Rossignol, Fayez Zannad, Marie-Dominique Devignes, Malika Smaïl-Tabbone

Inductive Database to Support Iterative Data Mining: Application to Biomarker Analysis on Patient Data in the Fight-HF project

Emmanuel Bresso[a,b], Joao-Pedro Ferreira[b], Nicolas Girerd[b], Masatake Kobayashi[b], Grégoire Preud'homme[b], Patrick Rossignol[b], Fayez Zannad[b], Marie-Dominique Devignes[a], Malika Smaïl-Tabbone[a,*]

[a] *Université de Lorraine, CNRS, Inria Nancy G.E., LORIA, UMR 7503, Vandoeuvre-lès-Nancy, France*
[b] *Université de Lorraine, Centre d'Investigations Cliniques Plurithématique 1433, INSERM 1116, CHRU de Nancy, France.*

Abstract

Machine learning is now an essential part of any biomedical study but its integration into real effective Learning Health Systems, including the whole process of Knowledge Discovery from Data (KDD), is not yet realised. We propose an original extension of the KDD process model that involves an inductive database. We designed for the first time a generic model of Inductive Clinical DataBase (ICDB) aimed at hosting both patient data and learned models. We report experiments conducted on patient data in the frame of a project dedicated to fight heart failure. The results show how the ICDB approach allows to identify biomarker combinations, specific and predictive of heart fibrosis phenotype, that put forward hypotheses relative to underlying mechanisms. Two main scenarios were considered, a local-to-global KDD scenario and a trans-cohort alignment scenario. This promising proof of concept enables us to draw the contours of a next-generation Knowledge Discovery Environment (KDE).

Keywords: Inductive database, Data mining, Heart Failure, Biomarkers, Knowledge Discovery from Data (KDD)

1. Introduction

Generating data to support clinical research involves the expensive process of clinical trials, including study design, legal and regulatory management, patient recruitment, data collection and patient follow-up. The General Data Protection Regulation (GDPR) requires, when collecting the consent of patients for the use of their data for research purposes, to indicate precisely the intended uses that will be made of such data and the duration of their retention. GDPR also allows data subjects to withdraw their consent to the processing of personal data concerning them. It is therefore reasonable to look for ways to efficiently make the most of this data to advance medical and clinical research and make of precision medicine a tangible reality. A common way of exploiting clinical data is to extract knowledge. Such knowledge is expected to be robust and of a nature to improve the diagnosis, care, and prognosis of patients, through a

*Corresponding author

growing awareness of the particularities of each patient or group of patients. These are precisely the goals of what is known as precision medicine.

Today, machine learning and data mining are widely recognised means for such purpose[1], albeit their integration into real effective Learning Health Systems is not achieved yet [1]. Our claim is that beyond specific ML techniques, we need to take a step back and consider the whole process of Knowledge Discovery from Data (KDD) which encompasses crucial downstream and upstream tasks. We propose to extend the KDD process model to make it able to support an iterative and traceable methodology thanks to an inductive database. We also design a generic model of Inductive Clinical DataBase (ICDB) aimed at hosting both patient data and learned models. We will show how this applies to successful iterative biomarker analyses on clinical patient data to extract relevant knowledge nuggets and address key underlying mechanisms of a disease of interest.

2. State of the Art

2.1. Learning Health Systems: dream or reality?

Patient data is the main asset to be developed in biomedical informatics for data mining purposes. Data mining is intended to bring out useful and reusable knowledge nuggets. In the hospital, patient data is generally collected from the various departments in a very heterogeneous manner. Laboratory data is produced by automatic assay devices and entered in a proprietary Laboratory Information Management System (LIMS). Clinical data is measured at the bedside and recorded in the patient record, which is not always electronic [2, 3]. Data reflecting the medical interview, carried out to detect previous pathologies or lifestyle characteristics that may influence the diagnosis or treatment, is also recorded in the patient record. When it comes to clinical trials, all these types of data are collected in a very careful and well-framed way, involving trained nurses and data managers, and tailored electronic health record (EHR) formats. For real-world data, EHRs are not yet the rule. The situation tends to evolve, albeit slowly in view of the immensity of the task, towards the constitution of patient data warehouses, exploiting the i2b2 (informatics for integrating biology and the bedside, https://www.i2b2.org) or PaDaWaN technologies [4, 5, 6].

Ideally and at a higher level, all the collected data should be used in Learning Health Systems (LHS) dedicated to learn from patient data the best practice for continuous improvement in health and healthcare [1]. In particular, LHS can provide the right framework to take advantage of the most effective and recent methods in machine learning and artificial intelligence, including the accountability and fairness of models and not forgetting the necessary regulatory environment to preserve patients' rights to privacy and equitable access to the benefits of research [7].

One of the first requirements in the design of such LHS is to keep track of analysis results to enable secondary analyses, building on previous results. The results of the secondary analyses should be stored in turn and re-used in further analyses and so on, in an endless iterative process. This requirement is especially important for the follow-up of the adopted predictive models for clinical decision support. Actually, designing efficient methods for model monitoring and concept drift detection is an active research topic[8, 9].

[1] In the rest of the paper, data mining and machine learning are considered as synonyms

2.2. *KDD process models and KDEs*

The KDD process has been widely studied in the last decades and several quite similar conceptualisations have resulted, such as Fayyad's early model in the 90's, the CRISP-DM (Cross-Industry Standard Process for Data Mining) process, or the SEMMA (Sample, Explore, Modify, Model, Assess) methodology (reviewed in [10]). Figure 1A shows the CRISP-DM modelling of the KDD process as an iterative process which is composed of several stages ranging from problem understanding to model deployment. What is not represented on the schema is the interactivity of the KDD process. Indeed, a human analyst or expert must often drive the process as many steps are hard to automate (e.g., tuning data preparation, selecting and tuning relevant data mining programs. . .). In practice, the iterative nature of the process often means that several alternatives are considered at key stages of the process such as various data preparation procedures (feature engineering, data engineering) in combination with candidate methods/algorithms for data modelling or data mining. When the results of the quantitative evaluation are judged to be satisfactory, the process usually ends with a qualitative assessment carried out by domain experts or based on relevant literature screening. It then gives rise, when possible, to the deployment of the learned model (for example in a decision support system). The favourable scenario is usually the one that concludes with a good quality publication. When the evaluation results are bad, a new iteration of the process can be started from its very beginning or at later stages, to try alternative solutions.

Today, many user-friendly platforms exist in the form of Knowledge Discovery Environments (KDEs) which certainly contribute to avoid fragmentation of the research results and accelerate the appropriation of the KDD concepts and techniques by scientists with very diverse backgrounds. A few popular KDEs are KNIME, SAS, WEKA [11, 12, 13]. These systems often include a rich set of data/feature engineering techniques, scalable implementations of mining algorithms and evaluation procedures [14]. Such KDEs are either standalone software with import/export functionalities (for importing datasets or exporting final or intermediate results and models in various proprietary formats), or strongly interfaced with DB management systems (with the ability to directly query a DB to build a dataset, store tables of transformed data, or save cross-validation results).

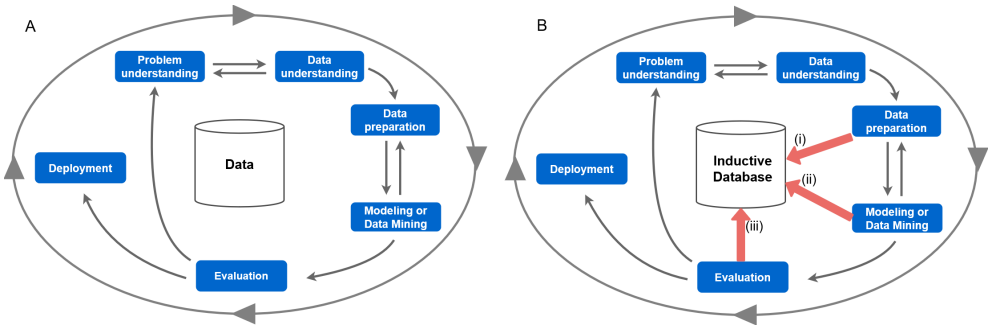

Figure 1: A: Cross-industry standard process for data mining (CRISP-DM). B: Endless data mining process supported by the inductive database approach.

2.3. *Do KDD process models really support iterative Data Mining?*

Albeit the numerous tasks composing the various steps of the KDD process were carefully described in a rich literature reviewed in [10], few solutions were proposed to really support the

KDD as a truly continuous and smooth process enabling easy chaining of successive iterations of the three main steps (data preparation, data mining, and evaluation).

A general solution was proposed by Imielinski and Mannila who introduced the concept of Inductive Database (IDB) as an extension of the classical database where data mining is considered a querying mechanism [15]. Subsequent studies in the 2000s focused on defining extended query languages based on the relational model and algebra and the SQL language (*e.g.,* MSQL, MINE SQL) [16, 17]. The main steps of the KDD process are thus expressible using the extended query language and the user can then query the IDB to find models satisfying a list of criteria. Unfortunately, these interesting works resulted in research prototypes limited to relational database management systems (DBMS) and a few mining algorithms, namely pattern mining and decision tree construction. The complexity and diversity of mining operations actually makes it difficult to integrate them into database query languages [18].

On the other hand, a local-to-global methodology –called LeGO– was proposed as a natural way to allow the identification of local patterns which are subsequently exploited to build a global model with a good predictive performance [19]. The authors present the LeGO approach as a possible model of the mining step of the KDD process and have shown that it unifies several published mining algorithms [20].

3. Our approach: Inductive Clinical Database Supporting Endless KDD process

3.1. Modeling Endless KDD Process

In order to support complex KDD methodologies, we propose an original extension of the KDD process model that involves an inductive database for pooling and cross-referencing the results of multiple data analyses. Figure 1B shows our extension of the CRISP-DM process model when switching from a database (as in Figure 1A) to an inductive database. It becomes now possible to store (i) transformed/prepared data, (ii) models resulting from data mining, and (iii) evaluation of the data mining results. Our process model differs from the inductive databases as defined by Imielinski and Mannila [15]. As a matter of fact, we use the DBMS as a persistence support for the mining results, letting the mining operations themselves be performed by any software. We will show that our extended KDD process model supports well the local-to-global methodology in order to chain several successive analyses corresponding to several iterations of the KDD process in view of constructing an efficient global model.

In this paper, our purpose is not to describe the implementation of a software that would support this extended KDD process. However, it is interesting to discuss the usability of three already implemented approaches to this aim. Firstly, the current KDEs already mentioned, even when they are interfaced with a DBMS, use proprietary formats for the storage of induced models, thus breaking the iteration possibilities. Secondly, the system prototypes proposed for supporting the Inductive DB approach (see Section 2.3) focus on extended querying languages. However, to include new ML models, we believe that it is easier to extend the DB data model than to extend a query language. Finally, emerging software engineering platforms for MLOps (such as neptune.ai, Flow ML, Comet) are proposed to support the ML cycle and facilitate the deployment of various experiments and their traceability[21]. Such platforms may be inspiring for the implementation of our extended KDD process, but conversely, the explicit persistence model that we propose in this paper could be interesting for MLOps tools.

3.2. Modeling Inductive Clinical Database

To instantiate the endless KDD process model in a clinical setting, it is necessary to make data and induced models compliant with a common data model. In Figure 2, we propose a generic model for inductive clinical database (ICDB) composed of (A) a data model for patient data collected during clinical trials, and (B) a data model allowing the description of the models learned from patients' data, such as patterns (e.g. frequent itemsets), classification rules, clusters... Following the open-science principles, each object in the ICDB is identified by a universal resource identifier (URI) which can either be defined locally or refer to a standard identifier.

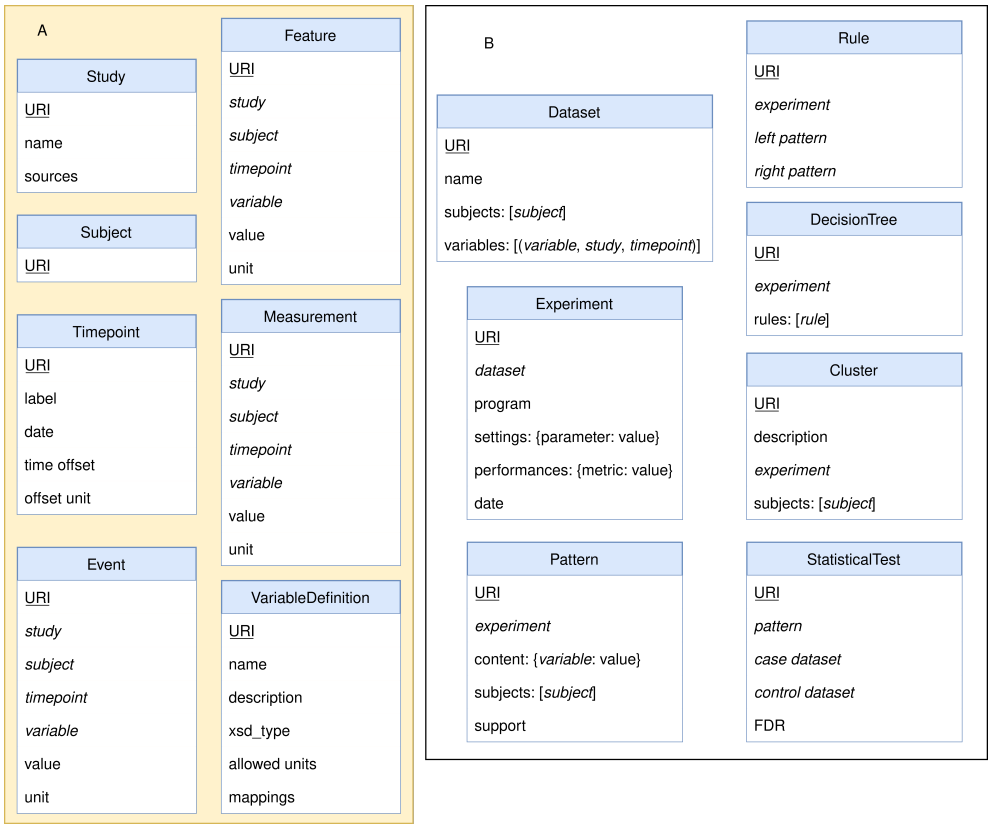

Figure 2: Inductive Clinical DataBase (ICDB) tables. Reference constraints are indicated in italics. (A) Tables on yellow background correspond to patient data and (B) tables on white background correspond to the inductive part of the model.

To design and implement the ICDB model, it appeared that the relational model respecting the first normal form constraint (limiting the value of an attribute in a tuple to one atomic value) is too restrictive. So we have opted to a more flexible data model allowing an attribute value to be a JSON[2] object [22, 23]. In the ICDB data model, "( )", "[ ]" and "{ }" are used to denote tuples, lists, and dictionaries respectively. For instance, in the *Dataset* table, a tuple representing a dataset composed of three patients and two variables from the same study but at two different

[2]JSON is a simple textual format for describing complex objects

timepoints could be:

```
(dataset_id: ``URI of the dataset'',
name: ``name of the dataset'',
subjects: [``Subject URI 1'', ``Subject URI 2'', ``Subject URI 3''],
variables: [(variable:``VDef URI 1'', study:``St URI 1'', timepoint:``TP URI 1''),
        (variable:``VDef URI 2'', study:``St URI 1'', timepoint:``TP URI 2'')
            ]
)
```

The patient data part of our ICDB model (Figure 2A) conceptualises data on patients (Table *Subject*) enrolled in studies or cohorts (Table *Study*) as a set of features (Table *Feature*) such as year of birth or gender, a set of measurements (Table *Measurement*) such as BMI or NPPB[3], and a set of events (Table *Event*) such as hospitalisation or death. The value of each feature/measurement/event corresponds to a study timepoint (Table *Timepoint*). The genericity of the model is enhanced by attaching each feature/measurement/event to a variable definition (Table *VariableDefinition*) which refers to standard URIs defined in shared vocabularies such as Schema.org or EFO (Experimental Factor Ontology)[4]. Interestingly, the attribute *mappings* in the table *VariableDefinition* allows one to map the variable to a concept/term/entry in any external source like an ontology, a database or a knowledge graph (see below). As for the inductive layer of the ICDB (Figure 2B), the model represents data analyses as experiments (Table *Experiment*). Each experiment is carried out on a dataset (Table *Dataset*) defined as a set of valued variables for a set of patients, using a mining program with specific settings and performance assessment. Such experiments produce various models which can be cluster sets (Table *Cluster*), decision trees (Table *DecisionTree*), rule sets (Table *Rule*), or patterns (Table *Pattern*). The patterns can in turn be tested statistically upon groups of interest (Table *StatisticalTest*). To keep the ICDB model simple, the list of included models is not exhaustive and the current selection is oriented towards self-explanatory (or descriptive) models that are more likely to be accepted by clinicians. Still, the list of models that can be induced can be extended as needed.

In the rest of the paper, the potential of ICDB to support complex KDD processes will be illustrated on two biomarker analysis scenarios performed with patient data available through the Fight-HF project. The first scenario illustrates the local-to-global KDD process, while the second is an alignment experiment on two studies conducted on two patient cohorts.

The following sections will describe firstly the context of the study, namely the motivation to study proteomic biomarkers in subgroups of patients with heart failure, and secondly, the two KDD scenarios.

3.3. Application to Proteomic Biomarker Analysis on Patient Data in the Fight-HF project

3.3.1. Context of the study

Biomarkers (BMs) can be defined as "characteristics that can be objectively measured and evaluated as indicators of normal biological processes, pathogenic processes or pharmacological response to a therapeutic intervention" [24]. Besides their role as surrogate endpoints in clinical trials [25], BMs are often considered as promising candidates to characterise subgroups

[3]Body Mass Index and Natriuretic ProPeptide B, respectively

[4]Standard URIs for the *gender* feature, the *body weight* measure, and the *death* event are https://schema.org/gender, http://www.ebi.ac.uk/efo/EFO_0004338, http://www.ebi.ac.uk/efo/EFO_0000480, respectively.

of patients displaying different responses to a given treatment. Such approach has been mostly successful in cancer research so far [26]. However, other complex diseases may also benefit from patient stratification thanks to BM analyses. Heart Failure (HF) and in particular HF with preserved ejection fraction (HF-pEF) are known to be heterogeneous syndromes in which it is still very difficult to predict the response to certain treatments. Several BMs have been reported to improve diagnostic and prognostic accuracy as well as provide relevant information regarding treatment response in acute HF. Moreover, it has been recently suggested that phenotyping HF-pEF patients according to fibrotic mechanisms using circulating biomarkers could help predict the response to anti-fibrotic drugs [27]. High-throughput quantification of proteomic BMs was therefore introduced in the Fight-HF project (2015-2021), the general objective of which is to fight against HF through precision medicine. The biological problem to solve here concerns the possibility of identifying BM patterns that could characterise different sub-groups of fibrotic patients, suggesting different underlying molecular mechanisms and consequently different responses to treatments. As pattern mining is a complex data mining task, we applied our approach for an iterative KDD process on a Fight-HF dataset composed of fibrotic and healthy individuals (described in section 4.1.1).

3.3.2. Scenarios of Biomarker Analyses

The first scenario illustrates a local-to-global KDD approach (Figure 3). A dataset composed of sick and control patients for which a list of proteomic BMs has been measured is retrieved from the ICDB and used for pattern mining. The first KDD iteration (on the left of the figure) consists in searching for BM patterns, maximal frequent itemsets or MFIs, in the sick subgroup. Evaluation is then performed by comparing the occurrence of MFIs in sick *versus* control groups using a statistical test. The patterns significantly more frequent in the sick group are qualified as contrasting patterns, here the contrasting MFIs or CMFIs. This set of CMFIs constitutes a set of local features. Actually, these features are the combinations of BM values that are specific to sick patients.

The second iteration of the scenario (on the right of the figure) involves learning a predictive classifier model, using as descriptors the sick specific patterns (CMFIs) identified before. The rules and performance of the learned classifier are stored in the ICDB. CMFIs occurring in the rules leading to the sick class are called positively predictive CMFIs (predCMFIs).

To evaluate the overall process, another classifier is built on the same patients, using as descriptors the complete set of BMs, through another instantiation of the second KDD iteration. It is then possible to compare the performance metrics of the two classifiers by querying the ICDB.

In addition, the BMs composing the predCMFIs extracted from the predictive model are then integrated into a biological knowledge graph and this allows hypotheses to be formulated on the possible underlying mechanisms of the disease.

In the second scenario (bottom of Figure 3), the results of independent KDD processes can be combined thanks to the ICDB in order to assess the relevance and consistency of extracted knowledge nuggets. For instance a KDD process can produce, on the basis of clinical variables distinct from BMs, patient clusters reflecting various stages of a disease. Once this clustering is included as a model in ICDB, one can test whether the predCMFIs identified previously are differentially distributed in such clusters, assuming that the predCMFI BMs were measured on the clustered patients as well. This type of trans-cohort alignment analysis is useful to ascertain whether BM patterns such as predCMFIs are associated with given stages of the disease. If the association is established then such BM patterns can be used for patient stratification.

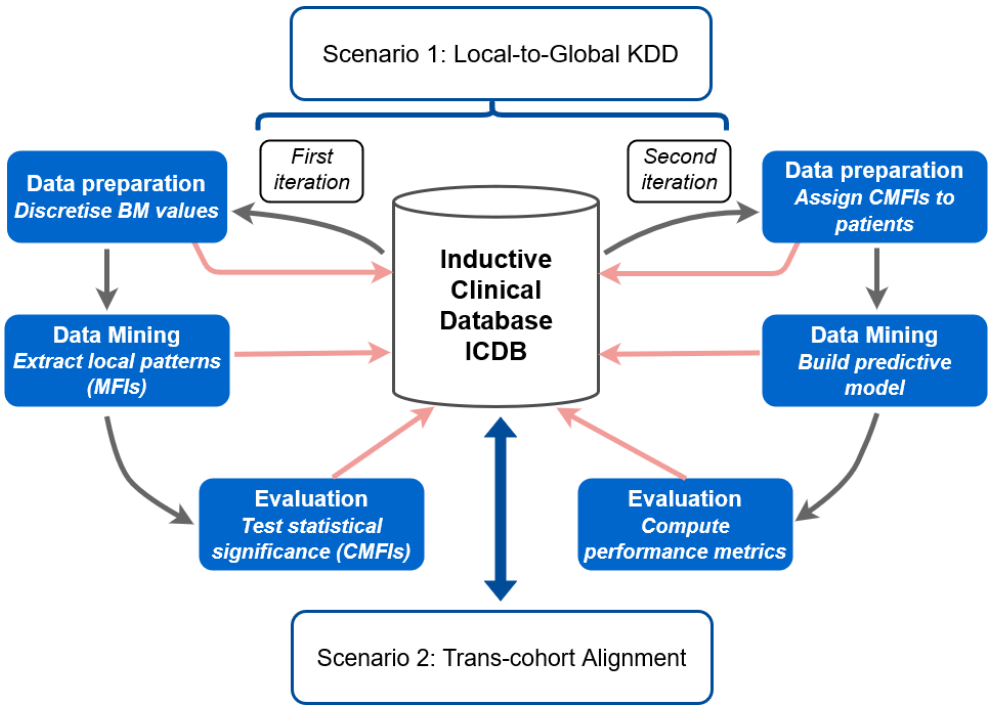

Figure 3: Outline of the two scenarios tested with the ICDB approach. Scenario 1 (top): Local-to-global KDD process. Left cycle corresponds to the first local KDD process aiming at extracting contrasted maximal frequent itemsets of BMs (CMFIs). Right cycle corresponds to the global KDD process in which local motifs are used to build a predictive model such as a decision tree aimed at predicting an outcome. Scenario 2 (bottom): Trans-cohort alignment. Mining results stored in the ICDB can be re-used in alignment studies, thus facilitating trans-cohort analyses.

One can notice that such scenarios can be implemented without ICDB but this would reduce guarantees on overall data consistency, experiment traceability and result reproducibility.

4. Experiments and results

4.1. Material and Methods

4.1.1. Datasets

The Fight-HF project brings together various pre-existing studies and consortia aiming at better understanding the various mechanisms underlying HF. In particular, the FIBROTARGETS consortium merged data from 12 patient cohorts into a common database available to each consortium partner [28]. The database consists of over 12,000 patients with a large spectrum of cardiovascular clinical phenotypes. It integrates community-based population cohorts, cardiovascular risk cohorts, and heart failure cohorts.

For the experiments reported in this paper, 487 subjects were manually selected from the TIME-CHF, HVC and STANISLAS cohorts (described in [28]): 416 were considered healthy and 71 as fibrotic patients with either a history of heart failure or a history of myocardial infarction. This dataset is called hereafter the Fibro dataset. The demographic, clinical and biological characteristics at baseline of these two groups of patients are provided in Table 1.

In addition to these measures, patient plasma samples were analysed for 92 protein biomarkers using the Olink Proseek® Multiplex CVD-II panel as described in [29]. Six additional biomarkers (for a total of 98) were also measured by "standard methods" and independently of the Olink® technology. These were: galectin-3 (Gal-3), growth differentiation factor-15 (GDF-15), matrix metalloproteinase-1 (MMP-1), stromal cell derived factor-1 alfa (SDF-1α), N-terminal pro-peptide of type III collagen (PIIINP) and neutrophil gelatinase-associated lipocalin (NGAL) (see details in [29]).

Table 1: Demographic, clinical and biological characteristics at baseline in the Fibro dataset. Used abbreviations: BP: blood pressure; eGFR: estimated glomerular filtration rate; LVEF: left ventricular ejection fraction; ACE: angiotensin-converting enzyme; ARB: angiotensin receptor blocker.

	Global	Healthy (n = 416)	Fibrotic (n = 71)	p-value
Women	226 (46.4 %)	201 (48.3 %)	25 (35.2 %)	0.053
Age, years	60.9 (53.8 - 68.3)	59.0 (52.0 - 66.9)	68.5 (61.5 - 76.1)	$< 10^{-4}$
BMI	24.33 ± 2.95	24.30 ± 2.84	24.52 ± 3.52	0.56
Antihypertensive treatment	153 (31.4 %)	92 (22.1 %)	61 (85.9 %)	$< 10^{-4}$
ACE inhibitor	67 (13.8 %)	24 (5.8 %)	43 (60.6 %)	$< 10^{-4}$
Beta blocker	109 (22.4 %)	60 (14.4 %)	49 (69.0 %)	$< 10^{-4}$
ARB	28 (5.7 %)	16 (3.8 %)	12 (16.9 %)	2.10^{-4}
Systolic BP, mmHg	128.70 ± 19.02	129.72 ± 18.08	122.58 ± 23.13	0.005
Diastolic BP, mmHg	75.27 ± 10.34	75.81 ± 10.20	72.01 ± 10.68	0.006
Heart rate, bpm	68.61 ± 12.96	68.03 ± 12.57	72.02 ± 14.67	0.016
Total cholesterol, mmol/L	5.56 ± 1.11	5.68 ± 1.06	4.87 ± 1.19	$< 10^{-4}$
LDL cholesterol, mmol/L	3.50 ± 0.94	3.60 ± 0.90	2.91 ± 0.95	$< 10^{-4}$
HDL cholesterol, mmol/L	1.49 ± 0.43	1.53 ± 0.42	1.29 ± 0.45	$< 10^{-4}$
Triglycerides, mmol/L	1.33 ± 0.75	1.28 ± 0.70	1.63 ± 0.96	3.10^{-4}
eGFR, mL/min/1.73m²	88.86 ± 19.97	92.91 ± 16.89	63.57 ± 19.23	$< 10^{-4}$
LVEF, %	60.18 ± 12.00	63.49 ± 6.88	40.45 ± 16.35	$< 10^{-4}$
E/A ratio	1.07 ± 0.44	1.06 ± 0.39	1.14 ± 0.74	0.26

The other dataset used in this work was extracted from the STANISLAS cohort which is a longitudinal population-based cohort [30]. This dataset is composed of 827 patients subjected to echocardiographic measurements and also tested for the same OLINK proteomic BMs as the first dataset. These patients were clustered into three echocardiographic-based groups by Kobayashi *et al.* [31]. These clusters were successfully mapped to distinct phenotypes corresponding to various abnormalities of cardiac structure and function, occurring prior to development of heart failure. Their descriptions have been stored in the ICDB (see Figure 2, table *Cluster*). After removing patient overlaps with the Fibro dataset [32], this dataset consisting of 658 patients is referred to hereafter as the Stan dataset.

4.1.2. ICDB implementation

The ICDB was implemented in the frame of the Fight-HF project as part of a larger architecture that addresses the storage of patient data from different cohorts in a way that enables cross-cohort integration and analyses. We equipped the Fight-HF ICDB with a Python API to provide a unified interface for use in diverse analytics workflows including endless KDD processes. In particular, this allows the data manager to (i) map each dataset variable to the common vocabulary or create new variable definitions and (ii) write ETL scripts that call the API functions to add actual data from a study.

The storage of the actual ICDB data relies on a Postgres[5] implementation which heavily uses the JSONB data type to allow for a flexible data model. We made this choice because we favour high-quality and coherent data with a variety of applications revolving around. A pure NoSQL solution would be application-oriented and would require different data models for different analysis types. However, if needed, a NoSQL solution can be combined with this DBMS to facilitate analysis tasks requiring data distribution.

Dashboard programs are valuable for extending the querying interface of traditional DBMS through a GUI, in order to visualise various data statistics (such as averages and distributions). Using the Redash software, we have implemented a prototype of such an application in the frame of the Fight-HF project to explore trans-cohort data stored in the Postgres Fight-HF ICDB. To facilitate the reuse of the ICDB model we provide in a gitlab repository (https://gitlab.inria.fr/capsid.public_codes/icdb-api) a docker composition including (i) program sources to create a new instance of ICDB, (ii) a set of Python API functions to insert data, and (iii) a Jupyter notebook creating an ICDB instance, inserting dummy data, and querying them.

4.1.3. Contrasted maximal frequent itemsets (CMFIs)

In our study, frequent itemsets are combinations of discretised BM values that occur frequently (with a support or frequency greater than 70%). However, it is well known that any subset of a frequent itemset is itself frequent [33]. Thus, it is interesting here to limit ourselves to frequent itemsets of maximal length (MFI) to reduce redundancy within the set of extracted frequent itemsets. To extract MFIs from the Fibro dataset, BM values for all patients are discretised into three classes (low, medium, and high) using the tertiles as thresholds. This is the Data preparation step of the first iteration of the KDD process schematised in Figure 3.

From fibrotic patients described by discretised BM values, we use the MLxtend python library [34] to identify the maximal frequent itemsets (MFIs) with a minimal support of 70% (Data Mining step of the first iteration in Figure 3).

Then, the support of each MFI is calculated in the healthy group. Itemset supports in the two groups are compared by performing a Fisher's exact test and resulting p-values are corrected for multiple testing using Benjamini-Hochberg method, leading to FDR values (False Discovery Rate). Finally, MFIs with an FDR lower than 0.05 are considered as Contrasting Maximal Frequent Itemsets (CMFIs). This corresponds to the Evaluation step of the first iteration in Figure 3.

4.1.4. Decision trees

Three different sets of features were built to induce binary decision trees: raw BM values, discretised BM values using the same thresholds as the MFIs, and CMFIs. Actually, this task corresponds to three distinct instantiations of the Data preparation step of the second iteration in Figure 3. Using the WEKA software, decision trees were built with the J48 program, a java implementation of the popular Quinlan's C4.5 algorithm (Data Mining step in Figure 3). The classes were balanced using the ClassBalancer filter which reweights the instances in the dataset so that each class has the same total weight. Due to the small size of the dataset, the performance metrics could be accurately estimated using the leave one out (LOO) procedure (Evaluation step in Figure 3).

[5] www.postgresql.org

4.1.5. Knowledge graph analysis

The Fight-HF graph knowledge box (FHF-GKBox) is a graph database containing data extracted from public data sources, thus providing most of the public knowledge available on human proteins, diseases, drugs, pathways and their relationships. The current version of the FHF-GKBox resource contains a total of 20,431 protein nodes imported from UniProt [35] (including all proteins involved in this study) and 2,272 pathway nodes coming from Reactome (v69) [36]. Protein–pathway relationships were retrieved from Reactome as well. Thanks to the *mappings* attribute stored in the ICDB table *VariableDefinition*, it was easy to transform the lists of BMs composing the predCMFIs into queries on the knowledge graph, using the appropriate object identifiers. The FHF-GKBox was searched using query patterns which are templates, predefined in the Cypher Query language, allowing to extract sub-graphs from a Neo4J graph database in a systematic way. In particular, a *biomarker-pathway-biomarker* query pattern was defined and used to retrieve all pathways involving at least two distinct biomarkers in a list of interest. The resulting sub-graph encompassing all identified pathways and related biomarkers is then automatically visualised using the Cytoscape software [37] with proper settings.

4.2. Results and Discussion

4.2.1. Local-to-global KDD Scenario in the ICDB

Identification of contrasting patterns. The list of MFIs obtained at the first KDD iteration is ranked in ascending order of FDR value and presented in Table 2. All the 39 MFIs have an FDR value lower than 0.05 and thus are considered as CMFIs. We counted 29 distinct BMs involved in these 39 itemsets. A large number of CMFIs, 22, are combinations of two BMs. The most represented BM is TRAIL_R2 (TNF receptor superfamily member 10b) which appears in half of the patterns. Hence, through the first iteration of the local-to-global scenario, we could identify 39 frequent (present in at least 70% of the fibrotic patients) biomarker combinations which are specific to the fibrotic group. The next question that arises concerns the predictive power of such patterns.

Decision Tree-based classification model using CMFIs. The second KDD iteration results in a decision tree classifier using CMFIs as descriptors on the two groups of patients (fibrotic and healthy). The decision tree is presented in Figure 4 and we can notice that it is simple yet powerful with a F-measure[6] of 0.81. Its specificity and sensitivity are 0.78 and 0.84, respectively. The decision tree involves only 11 CMFIs over 39 in the dataset. These 11 CMFIs concern 12 distinct BMs.

To go further in the analysis, one can extract from the decision tree a classification rule for each path starting at the root node and ending at a leaf. For instance, here are two rules predicting the fibrotic outcome (first paths starting from the right):

```
IF (REN high AND TNFRFS11A high) AND (TRAIL_R2 high AND CD4 high)
THEN predicted_phenotype = Fibrotic

IF (REN high AND TNFRFS11A high) AND NOT (TRAIL_R2 high AND CD4 high)
AND NOT (BOC high)
THEN  predicted_phenotype = Fibrotic
```

[6]The F-measure is defined as the harmonic mean of precison and recall metrics.

Predictive CMFIs occurring without negation (NOT) in at least one rule leading to fibrotic prediction are named positively predictive CMFIs (predCMFIs) in the frame of this analysis. Hence, the above rules have 2 predCMFIs: (REN high AND TNFRFS11A high) and (TRAIL_R2 high AND CD4 high). In total we have 8 predCMFIs (boxed in bold on Figure 4) which contain 9 individual BMs, namely REN, TNFRSF11A, PTX3, TRAIL_R2, CD4, FGF 23, SDF1A, TF, and XCL1.

Figure 4: Fibrotic *versus* Healthy Decision tree built with CMFIs. The leaves are labeled by the class corresponding to the majority of the patients they contain. The first number is the number of patients in the leaf whereas the second one is the number of misclassified patients. Numbers are non integer because of the class balancing procedure.

The performances of the decision tree classifier obtained with CMFIs as descriptors was compared with those of the same type of classifier obtained on the same groups of patients but using all the BMs as individual descriptors, with raw or discretised values. The results of the comparison are presented in Table 3. We notice a better value of the F-measure when using CMFIs as features. In other words, this model has the better tradeoff between recall and precision and this is interesting for several reasons. On the one hand, this means that while using less BM measurements (29 *versus* 98 in the Fibro dataset), it is possible to build a better predictive model. In addition to being more predictive, the model is also simpler. Actually, the CMFI-based decision tree only requires the measurement of 12 BMs if it is to be used as a decision support instrument.

On the other hand, the fact that the CMFIs outperform the discretised BMs also underlines the benefit of considering BMs in combination rather than in isolation. Indeed, a BM can be involved in various biological processes (or pathways) with diverse BM partners.

Interpretation of classification rules. The objective here is to further investigate the posi-

tively predictive CMFIs (predCMFIs) in order to gain an understanding of the possibly different mechanisms involved in the rules extracted from the decision tree. To do so, we use our FHF-GKBox in-house knowledge graph constructed from public sources. We extract the protein-pathway-protein relationships connecting any pair of biomarkers among the 9 BM, appearing in the predCMFIs. We then highlight with distinct colors the shortest paths connecting the BMs involved in the rules predicting fibrotic outcome in the decision tree. The resulting picture is shown in Figure 5. It appears that the 9 BMs present in predCMFIs can all be connected through shared pathways. The main pathways involved in the connections can be grouped as "Immunity, cytokines" on the one hand, and "Signal transduction, chemokines, GPCR" on the other hand. Other isolated pathways are Metabolism of proteins, Hemostasis and Innate immune system. Strikingly, the two BMs present in each predCMFIs of size 2 (REN high, TNFRSF11A high), (TRAIL_R2 high, CD4 high) or (TRAIL_R2 high, REN high) are not directly connected through a unique shared pathway but rather through a shortest path of length 4, containing two pathways inter-connected by FGF23. This suggests that the co-occurrence of 2 BMs in a predCMFI is not trivial and does not result simply from the co-occurence of these BMs in the same pathway. Moreover, BM FGF23 seems to play a central role in this network of BMs as all the paths share the edge connecting FGF23 to the Signal transduction pathway, and all but one path share the edge connecting FGF23 to the Immune system pathway.

Paths colored in red represent the two rules found in the right part of the tree when predCMFI (REN high, TNFRSF11A high) is present. These rules concern the majority of patients (173 *versus* 72 for the other rules) and the corresponding paths are the only ones that involve the "Cytokine signaling in immune system" pathway. However these paths do not involve the other signaling pathways present in the subgraph, nor the Hemostasis pathway. Other colored paths represent rules that can be verified in the absence of predCMFI (REN high, TNFRSF11A high). They do not involve the "Cytokine signaling in immune system" pathway. The blue path connects PTX3, SDF1a, and predCMFI (TRAIL_R2 high, REN high) together through FGF23. The orange paths connect PTX3, SDF1a and XCL1 and also require FGF23. They involve a bunch of signaling pathways that are not used by the other paths. The purple path is a subset of the orange one without PTX3 and XCL1. Finally the green one that connects PTX3 and TF is the longest one (6 edges) and the only one that involves the Hemostasis pathway.

In summary, this network analysis reveals the central position of FGF23, and the possible role of the Cytokine signaling in immune system pathway to discriminate between two groups of rules, and therefore two possible groups of mechanisms leading to fibrosis.

4.2.2. Alignment scenario in the ICDB

In this scenario, the models learned through two distinct KDD processes and stored in the ICDB are exploited in order to assess the relevance of predCMFIs. Using the Stan dataset that results from the clustering of patients based on echocardiographic measurements, and the dashboard application of ICDB, the frequencies (or support) of each predCMFI were computed in the three clusters of patients. A χ^2 test was performed for pairwise comparison of frequencies across clusters. The results are presented in Figure 6.

No predCMFI was found significantly different between clusters 1 and 2. By contrast, all predCMFI frequencies are significantly greater in cluster 3 than in cluster 2. Finally, the predCMFI are also significantly greater in cluster 3 than in cluster 1. This reinforces the fact that predCMFIs indicate promising BM candidates for detecting fibrosis even before HF symptoms, since cluster 3 contains patients with the most severe phenotype [31].

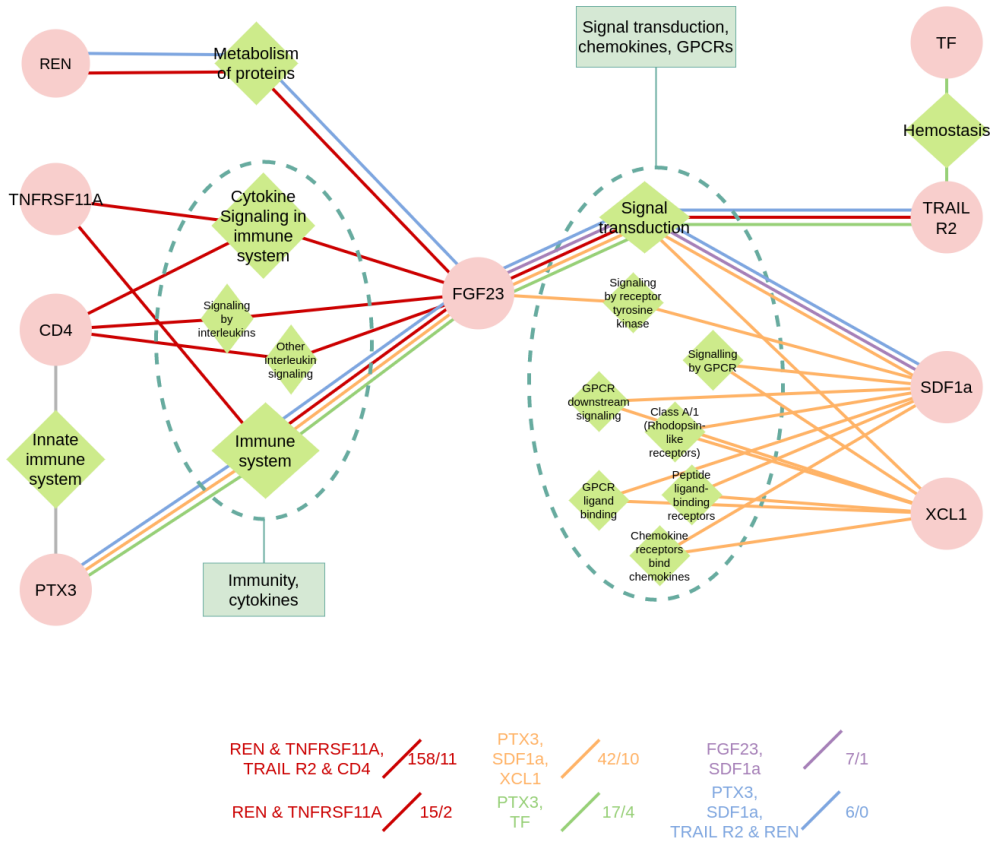

Figure 5: Protein-pathway-protein network to interpret the positively predictive CMFIs. The colored paths represent the 6 branches of the decision tree leading to fibrotic patients. N/n : N, number of patients in the leaf ; n, number of mis-classified patients. CD4: CD4 molecule ; FGF23: fibroblast growth factor 23 ; PTX3: pentraxin 3 ; REN: renin ; SDF1A: Fas ligand ; TF: coagulation factor III, tissue factor ; TNFRSF11A: TNF receptor superfamily member 11a ; TRAIL R2: TNF receptor superfamily member 10b ; XCL1: X-C motif chemokine ligand 1 (or lymphotactin).

5. Conclusions

Our objective was to show that the extended KDD process model relying on an ICDB can efficiently support sophisticated data analysis methodologies. Our approach allows successive or competing analyses which correspond to several iterations of the KDD process to be chained in view of extracting the best added value from patient data. Moreover, having an integrated view on data and learning results comes with guarantees on their global consistency as well as experiment traceability and result reproducibility. In the reported experiments this approach allowed us to extract relevant and robust knowledge nuggets from a rather small dataset for which statistical methods were unsuccessful (unpublished negative results). Moreover the subsequent qualitative analysis in the knowledge graph suggests a way to discriminate between two groups of mechanisms leading to fibrosis.

Actually, an infinite number of scenarios can be supported with our ICDB-based KDD process model and the two presented scenarios are just proofs of concept. The first scenario illus-

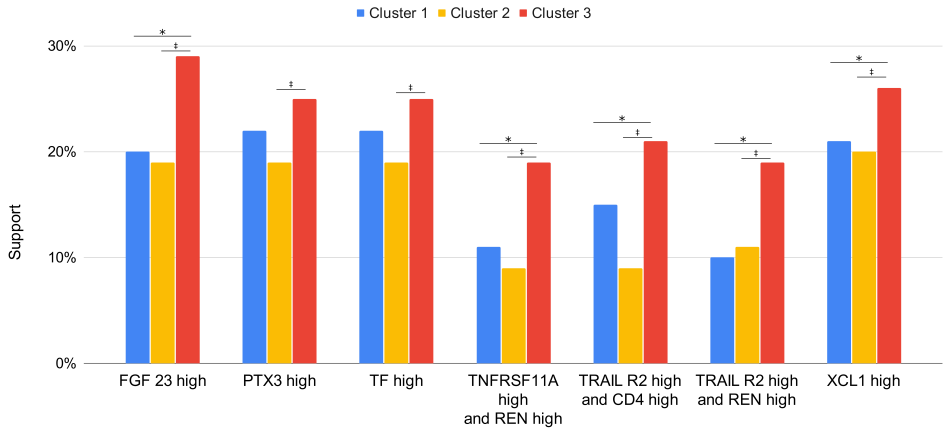

Figure 6: Support of predCMFIs extracted from the Fibro dataset in the three clusters from the Stan dataset. * and ‡ indicate significant difference between cluster 1 and 3 and cluster 2 and 3 respectively. Only 7 from the 8 predCMFIs are shown because one BM (Sdf1a) was not measured on the Stan dataset.

trates how the ICDB framework can facilitate a local-to-global KDD approach in which local patterns extracted from the data can be re-used in KDD iterations in order to improve the performance of a model. As an alternative, local patterns extracted from two independent datasets can also be combined in the ICDB framework for identifying both robust and highly predictive patterns.

As for the second alignment scenario, it illustrates that dashboard-like functionalities applied to the inductive layer of the ICDB make it possible to perform statistical queries on the models discovered from the data. Such queries can help to detect inconsistencies or evaluate agreement between the learned models.

The widespread deployment of our approach requires more software engineering to achieve a next-generation KDE. Hence, the integration of patient data in the ICDB requires the development of wrappers which rely on data dictionaries and mappings. Moreover, making persistent the results of each machine learning program necessitates structuring them in a format compliant with the ICDB model. An interesting option for the development of our envisioned next generation KDE as a whole would be to study the integration of ICDB (or any domain-specific inductive DB) into MLOps platforms, thereby extending their metadata repositories, connecting learned models to data and making models machine-actionable. In particular this would facilitate the model monitoring in case of continuous import of patient data in the ICDB, in order to detect when the model is no more able to predict the outcome, because of unforeseen changes (in data distribution, patient profiles...). This "concept drift" phenomenon is generating a lot of interest nowadays [9, 8].

Finally, the current recommendations suggest to share models instead of data. Indeed, since the patient data cannot be kept indefinitely due to regulation constraints, the persistent inductive part of the ICDB warrants not to loose the benefit of the analyses carried out and not to deprive oneself of the follow-up of the learned models.

Acknowledgments

This work was supported by the RHU Fight-HF, a public grant overseen by the French National Research Agency (ANR) as part of the second "Investissements d'Avenir" program (reference: ANR-15-RHUS-0004) and by the CPER IT2MP and FEDER Lorraine. EB's salary was funded by the RHU Fight-HF grant.

References

[1] C. Friedman, J. Rubin, J. Brown, M. Buntin, M. Corn, L. Etheredge, C. Gunter, M. Musen, R. Platt, W. Stead, K. Sullivan, D. Van Houweling, J Am Med Inform AssocToward a science of learning systems: a research agenda for the high-functioning Learning Health System, J Am Med Inform Assoc 22 (1) (2015) 43–50.

[2] R. S. Evans, Yearb Med InformElectronic Health Records: Then, Now, and in the Future, Yearb Med Inform Suppl 1 (2016) 48–61.

[3] E. Joukes, N. F. de Keizer, M. C. de Bruijne, A. Abu-Hanna, R. Cornet, Appl Clin InformImpact of Electronic versus Paper-Based Recording before EHR Implementation on Health Care Professionals' Perceptions of EHR Use, Data Quality, and Data Reuse, Appl Clin Inform 10 (2) (2019) 199–209.

[4] R. W. Majeed, R. Röhrig, Stud Health Technol InformAutomated realtime data import for the i2b2 clinical data warehouse: introducing the HL7 ETL cell, Stud Health Technol Inform 180 (2012) 270–274.

[5] G. Fette, M. Kaspar, G. Dietrich, M. Ertl, J. Krebs, S. Stoerk, F. Puppe, Stud Health Technol InformA Customizable Importer for the Clinical Data Warehouses PaDaWaN and I2B2, Stud Health Technol Inform 243 (2017) 90–94.

[6] G. Fette, M. Kaspar, L. Liman, G. Dietrich, M. Ertl, J. Krebs, S. Störk, F. Puppe, Stud Health Technol InformQuery Translation Between openEHR and i2b2, Stud Health Technol Inform 258 (2019) 16–20.

[7] T. Panch, P. Szolovits, R. Atun, J Glob HealthArtificial intelligence, machine learning and health systems, J Glob Health 8 (2) (2018) 020303.

[8] R. G. de Sousa, S. M. Peres, M. Fantinato, H. A. Reijers, Concept drift detection and localization in process mining: an integrated and efficient approach enabled by trace clustering, in: C. Hung, J. Hong, A. Bechini, E. Song (Eds.), SAC '21: The 36th ACM/SIGAPP Symposium on Applied Computing, Virtual Event, Republic of Korea, March 22-26, 2021, ACM, 2021, pp. 364–373. doi:10.1145/3412841.3441918.
URL https://doi.org/10.1145/3412841.3441918

[9] M. Maisenbacher, M. Weidlich, Handling concept drift in predictive process monitoring, in: X. F. Liu, U. Bellur (Eds.), 2017 IEEE International Conference on Services Computing, SCC 2017, Honolulu, HI, USA, June 25-30, 2017, IEEE Computer Society, 2017, pp. 1–8. doi:10.1109/SCC.2017.10.
URL https://doi.org/10.1109/SCC.2017.10

[10] G. Mariscal, O. Marbán, C. Fernández, A survey of data mining and knowledge discovery process models and methodologies, Knowl. Eng. Rev. 25 (2) (2010) 137–166. doi:10.1017/S0269888910000032.
URL https://doi.org/10.1017/S0269888910000032

[11] M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, K. Thiel, B. Wiswedel, KNIME - the konstanz information miner: version 2.0 and beyond, SIGKDD Explor. 11 (1) (2009) 26–31. doi:10.1145/1656274.1656280.
URL https://doi.org/10.1145/1656274.1656280

[12] M. M. Holt, Learning SAS by example: A programmer's guide, Technometrics 50 (1) (2008) 91–92. doi:10.1198/tech.2008.s532.
URL https://doi.org/10.1198/tech.2008.s532

[13] I. H. Witten, E. Frank, M. A. Hall, Data mining: practical machine learning tools and techniques, 3rd Edition, Morgan Kaufmann, Elsevier, 2011.
URL https://www.worldcat.org/oclc/262433473

[14] J. C. Freytag, R. Ramakrishnan, R. Agrawal, Data mining: The next generation, it Inf. Technol. 47 (5) (2005) 308–312. doi:10.1524/itit.2005.47.5_2005.308.
URL https://doi.org/10.1524/itit.2005.47.5_2005.308

[15] T. Imielinski, H. Mannila, A database perspective on knowledge discovery, Commun. ACM 39 (11) (1996) 58–64. doi:10.1145/240455.240472.
URL https://doi.org/10.1145/240455.240472

[16] L. Richter, J. Wicker, K. Kessler, S. Kramer, An inductive database and query language in the relational model, in: A. Kemper, P. Valduriez, N. Mouaddib, J. Teubner, M. Bouzeghoub, V. Markl, L. Amsaleg, I. Manolescu (Eds.), EDBT 2008, 11th International Conference on Extending Database Technology, Nantes, France, March 25-29, 2008, Proceedings, Vol. 261 of ACM International Conference Proceeding Series, ACM, 2008, pp. 740–744.

doi:10.1145/1353343.1353440.
URL https://doi.org/10.1145/1353343.1353440

[17] J. Wicker, L. Richter, S. Kramer, SINDBAD and SiQL: Overview, applications and future developments, in: S. Dzeroski, B. Goethals, P. Panov (Eds.), Inductive Databases and Constraint-Based Data Mining, Springer, 2010, pp. 289–309. doi:10.1007/978-1-4419-7738-0_12.
URL https://doi.org/10.1007/978-1-4419-7738-0_12

[18] J.-F. Boulicaut, M. Klemettinen, H. Mannila, Modeling kdd processes within the inductive database framework, in: International Conference on Data Warehousing and Knowledge Discovery, Springer, 1999, pp. 293–302.

[19] A. Knobbe, B. Crémilleux, J. Fürnkranz, M. Scholz, From local patterns to global models: The lego approach to data mining, in: International Workshop From Local Patterns to Global Models co-located with ECML/PKDD08, Antwerp, Belgium, 2008, pp. 740–744.

[20] J. Fürnkranz, A. J. Knobbe, Guest editorial: Global modeling using local patterns, Data Min. Knowl. Discov. 21 (1) (2010) 1–8. doi:10.1007/s10618-010-0169-7.
URL https://doi.org/10.1007/s10618-010-0169-7

[21] M. M. John, H. H. Olsson, J. Bosch, Towards mlops: A framework and maturity model, in: M. T. Baldassarre, G. Scanniello, A. Skavhaug (Eds.), 47th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2021, Palermo, Italy, September 1-3, 2021, IEEE, 2021, pp. 1–8. doi:10.1109/SEAA53835.2021.00050.
URL https://doi.org/10.1109/SEAA53835.2021.00050

[22] C. Chasseur, Y. Li, J. M. Patel, Enabling json document stores in relational systems., in: WebDB, Vol. 13, 2013, pp. 14–15.

[23] D. Petković, Json integration in relational database systems, Int J Comput Appl 168 (5) (2017) 14–19.

[24] J. A. Wagner, A. J. Atkinson, Measuring Biomarker Progress, Clin Pharmacol Ther 98 (1) (2015) 2–5.

[25] K. Strimbu, J. A. Tavel, What are biomarkers?, Curr Opin HIV AIDS 5 (6) (2010) 463–466.

[26] M. Chand, D. S. Keller, R. Mirnezami, M. Bullock, A. Bhangu, B. Moran, P. P. Tekkis, G. Brown, A. Mirnezami, M. Berho, Novel biomarkers for patient stratification in colorectal cancer: A review of definitions, emerging concepts, and data, World J Gastrointest Oncol 10 (7) (2018) 145–158.

[27] P. Rossignol, J. P. Ferreira, F. Zannad, Eur J Heart FailFibrosis mechanistic phenotyping and antifibrotic response determination with biomarkers in heart failure: one single biomarker may not fit all settings, Eur J Heart Fail 20 (9) (2018) 1300–1302.

[28] J. P. Ferreira, J.-L. Machu, N. Girerd, F. Jaisser, T. Thum, J. Butler, A. González, J. Diez, S. Heymans, K. McDonald, M. Gyöngyösi, H. Firat, P. Rossignol, A. Pizard, F. Zannad, Rationale of the FIBROTARGETS study designed to identify novel biomarkers of myocardial fibrosis, ESC heart failure 5 (1) (2018) 139–148. doi:10.1002/ehf2.12218.

[29] J. P. Ferreira, A. Pizard, J.-L. Machu, E. Bresso, H.-P. Brunner-La Rocca, N. Girerd, C. Leroy, A. González, J. Diez, S. Heymans, et al., Plasma protein biomarkers and their association with mutually exclusive cardiovascular phenotypes: The FIBRO-TARGETS case–control analyses, Clinical Research in Cardiology 109 (1) (2020) 22–33.

[30] J. P. Ferreira, N. Girerd, E. Bozec, L. Mercklé, A. Pizard, S. Bouali, E. Eby, C. Leroy, J.-L. Machu, J.-M. Boivin, Z. Lamiral, P. Rossignol, F. Zannad, Cohort Profile: Rationale and design of the fourth visit of the STANISLAS cohort: a familial longitudinal population-based cohort from the Nancy region of France, International Journal of Epidemiology 47 (2) (2017) 395–395j. doi:10.1093/ije/dyx240.

[31] M. Kobayashi, O. Huttin, M. Magnusson, J. P. Ferreira, E. Bozec, A.-C. Huby, G. Preud'homme, K. Duarte, Z. Lamiral, K. Dalleau, E. Bresso, M. Smaïl-Tabbone, M.-D. Devignes, P. M. Nilsson, M. Leosdottir, J.-M. Boivin, F. Zannad, P. Rossignol, N. Girerd, Machine learning-derived echocardiographic phenotypes predict heart failure incidence in asymptomatic individuals, JACC: Cardiovascular Imaging 0 (0). doi:10.1016/j.jcmg.2021.07.004.

[32] J. P. Ferreira, Z. Lamiral, C. Xhaard, K. Duarte, E. Bresso, M.-D. Devignes, E. L. Floch, C. D. Roulland, J.-F. Deleuze, S. Wagner, B. Guerci, N. Girerd, F. Zannad, J.-M. Boivin, P. Rossignol, Circulating plasma proteins and new-onset diabetes in a population-based study: proteomic and genomic insights from the stanislas cohort, European Journal of Endocrinology 183 (3) (2020) 285 – 295. doi:10.1530/EJE-20-0246.

[33] G. Grahne, J. Zhu, High performance mining of maximal frequent itemsets, in: 6th International workshop on high performance data mining, Vol. 16, 2003, p. 34.

[34] S. Raschka, Mlxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack, The Journal of Open Source Software 3 (24). doi:10.21105/joss.00638.

[35] The UniProt Consortium, UniProt: The universal protein knowledgebase in 2021, Nucleic Acids Research 49 (D1) (2021) D480–D489.

[36] B. Jassal, L. Matthews, G. Viteri, C. Gong, P. Lorente, A. Fabregat, K. Sidiropoulos, J. Cook, M. Gillespie, R. Haw, F. Loney, B. May, M. Milacic, K. Rothfels, C. Sevilla, V. Shamovsky, S. Shorser, T. Varusai, J. Weiser, G. Wu, L. Stein, H. Hermjakob, P. D'Eustachio, The reactome pathway knowledgebase, Nucleic Acids Research 48 (D1)

(2020) D498–D503.
[37] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, T. Ideker, Cytoscape: A software environment for integrated models of biomolecular interaction networks, Genome Research 13 (11) (2003) 2498–2504.

Table 2: Contrasted Maximal Frequent Itemsets in fibrotic *versus* healthy groups

CMFI	Fibrotic Support	Healthy support	FDR
TRAIL_R2 high & AGRP high	0.77	0.10	1.73e-30
TRAIL_R2 high & TNFRSF11A high	0.79	0.13	4.12e-27
TRAIL_R2 high & CD4 high	0.77	0.13	9.23e-27
REN high & TNFRSF11A high	0.70	0.09	9.23e-27
TRAIL_R2 high & REN high	0.72	0.11	3.05e-25
TRAIL_R2 high & Gal_9 high	0.76	0.14	6.97e-25
TRAIL_R2 high & IL_4RA high	0.73	0.13	1.69e-23
IL16 high & CTSL1 high	0.70	0.12	5.33e-23
CTSL1 high & TNFRSF11A high	0.72	0.13	6.52e-23
TRAIL_R2 high & CCL3 high	0.70	0.12	7.34e-23
CCL3 high & CD4 high	0.70	0.12	7.34e-23
TRAIL_R2 high & IL16 high	0.72	0.13	9.01e-23
DCN high & TNFRSF11A high	0.70	0.13	1.08e-22
CD4 high & PGF high	0.70	0.13	1.08e-22
IL16 high & TNFRSF11A high	0.70	0.13	3.43e-22
TRAIL_R2 high & PGF high	0.73	0.16	1.37e-21
TRAIL_R2 high & CTSL1 high	0.72	0.15	2.08e-21
IL16 high & PGF high	0.70	0.14	2.98e-21
IL16 high & AGRP high	0.70	0.14	4.97e-21
TRAIL_R2 high & DCN high	0.72	0.15	5.30e-21
TNFRSF11A high & PGF high	0.70	0.15	2.33e-20
IL16 high & CD4 high	0.70	0.15	3.80e-20
Gdf15 high	0.76	0.22	8.83e-18
Sdf1a high	0.77	0.25	9.35e-17
TNFRSF13B high	0.73	0.22	3.68e-16
TGM2 high	0.75	0.24	3.87e-16
PRSS8 high	0.70	0.21	2.63e-15
IL_1ra high	0.70	0.22	3.74e-15
VSIG2 high	0.72	0.23	4.16e-15
ACE2 high	0.72	0.24	2.48e-14
TF high	0.75	0.27	2.48e-14
FGF_23 high	0.72	0.25	6.72e-14
XCL1 high	0.72	0.25	1.29e-13
PTX3 high	0.75	0.29	3.13e-13
PD_L2 high	0.70	0.25	3.23e-13
CEACAM8 high	0.72	0.26	4.47e-13
BOC high	0.72	0.27	1.56e-12
AMBP high	0.70	0.26	2.05e-12
LOX_1 high	0.70	0.26	2.05e-12

Table 3: Comparison of decision tree classifier metrics

	All BMs		CMFIs
	Raw	Discretised	
F-Measure	0.78	0.7	0.81
Precision	0.93	0.79	0.79
Recall (Sensitivity)	0.67	0.63	0.84
Specificity	0.95	0.83	0.78