Chance Discovery and Learning Minority Classes

TuBao HO and DucDung NGUYEN

Japan Advanced Institute of Science and Technology
Tatsunokuchi,Ishikawa, 923-1292 JAPAN

{bao, dungduc}@jaist.ac.jp

Received 12 March 2001

Abstract While chances are viewed in the chance discovery research context as events/situations with significant impact on human decision making, we are interested in a subset of those chances that are unexpected or contradictory with human common knowledge. In our view, the human role is essential in chance discovery, and there is a need for chance discovery support systems and methods. The objective of this paper is twofold. First to introduce the method LUPC that can learn minority classes from large unbalanced datasets with high performance. LUPC can be a powerful method for chance discovery with its visualization tools and incorporation of background knowledge in form of exclusive and inclusive constraints. Second to present case studies in which LUPC is used to support the user discovering significant unexpected patterns from stomach cancer and hepatitis databases.

Keywords Learning Minority Class, Chance Discovery, Visualization, Medical Data.

§1 Introduction

It happens that in many applications the user wants to find useful and new knowledge from only one particular object class, and this target class is represented by only a few examples while the others are represented by a large number. Facing with unbalanced datasets, common learning algorithms usually

produce unsatisfactory classifiers. Learning minority or rare classes in unbalanced datasets is among issues that were not much previously considered by the machine learning community, is now coming into light.[12, 6] The main approaches to this problem include rebalancing the training data (up-sampling of the minority classes[8] or down-sampling of the majority classes[7]) and cost sensitive classification with or without boosting.[15]

Chance discovery can be seen as a new attempt in knowledge discovery and data mining (KDD) with the aim of discovering chances.[10] In the author's formulation, a chance is defined "as an event or situation with significant impact on human decision making." Chance discovery is "to become aware of and to explain the significance of a chance, especially if the chance is rare and its significance has been unnoticed". Besides these aspects of chance discovery and data mining methods for finding rare but significant events, the author emphasizes human roles in chance discovery, the human interaction with data, and points out three keys to chance discovery (communication, imagination, data mining) as well criteria for evaluating chance (proposability, unnoticability, growability).

In our view point, there is a strong relation between knowledge discovery from minority classes and chance discovery, and this paper is an attempt to make the link between them. We are interested in a kind of chances: unexpected events/situations that are different from human common knowledge. In this paper we first introduce the method LUPC (stands for Learning the Unbalanced Positive Class) that can learn minority or rare classes in large unbalanced datasets with high performance. The main features of LUPC are a combination of separate-and-conquer rule induction[2] with association rule mining. In addition to the feature of learning minority classes, LUPC has recently two improvements: (1) it allows the user to incorporate his/her domain knowledge in terms of exclusive and inclusive constraints; and (2) it is associated with data and rule visualization tools. With these imrovements, LUPC can be used a chance discovery support system for rare or unexpected events in large databases.

Section 2 of the paper addresses aspects of learning minority classes, the algorithm LUPC, data and rule visualization with LUPC, and exclusive and inclusive constraints in the mining process of LUPC. Section 3 presents how chances are discovered by LUPC in stomach cancer and hepatitis databases.

§2 Learning Minority Classes

2.1 The Problem

Consider the rule induction problem where the target class is viewed as the *positive* class C^+ and all other classes as the *negative* class C^-. The training dataset D then can be seen as union of the positive instance subset Pos and the negative instance subset Neg, i.e., $D = Pos \cup Neg$. We are concerned with learning problem when the class C^+ is represented by only a few instances while the class C^- is represented by a large number, i.e., $|Pos| \ll |Neg|$.

Let $A_1, A_2, ..., A_p$ denote *attributes* with domains $dom(A_1), dom(A_2), ...,$ $dom(A_p)$ that can be categorical or numeric. A prediction rule for C^+ is understood as the co-occurrence of m attribute-value pairs will imply the occurrence of the event C^+, and denoted by $\wedge_{j=1}^{m}(A_{i_j} = v_{i_j}) \rightarrow C^+$ where $i_j \in \{1, 2, \ldots p\}$ and $v_{i_j} \in dom(A_{i_j})$. Denote by $cov(R)$ the set of all training instances covered by a rule R. This set is divided into two subsets of covered instances in Pos and Neg, i.e., $cov(R) = cov^+(R) \cup cov^-(R)$. As usual, the accuracy of a rule C^+ is defined as $acc(R) = |cov^+(R)|/|cov(R)|$, and its support (cover ratio) is $sup(R) = |cov(R)|/|D|$. The general problem of *learning a minority class in a unbalanced dataset* to find a set of rules for the target positive class, $R^+ = \{R_1^+, R_2^+, \ldots, R_q^+\}$, so that $Pos \subseteq \cup_{i=1}^{q} cov(R_i^+)$ and the discovered rules are "best" in terms of quality measurements such as accuracy and support.

Common learning algorithms usually produce unsatisfactory results when learning minority classes. It is because the fundamental assumption of the same class distribution and measures of learning performance built in these algorithms are not met or suitable in unbalanced datasets. The approaches to the problem of learning minority classes include rebalancing the training data and cost sensitive classification with or without boosting. Rebalancing techniques include up-sampling of the minority classes[8] or down-sampling of the majority classes.[7] The cost sensitive classification is usually (though not strictly) applied to the divide-and-conquer approach.[15, 14] Cost sensitive methods typically bias decisions in different directions by raising or lowering the cost of a misclassification.

2.2 Algorithm LUPC

[1] Basic ideas and search bias

LUPC is a separate-and-conquer rule induction method that follows

the generic separate-and-conquer scheme[2)] with improvements to learn minority classes in unbalanced datasets. LUPC consequently learns a rule set from Pos and Neg given user-specified minimum accuracy threshold ($mina$) and minimum cover ratio ($minc$). If both $mina$ and $minc$ are set to high values, high accuracy and cover ratio rules can be found but they can often cover a small part of Pos, and many infrequent but significant rules may not be considered. If both $mina$ and $minc$ are set with low values then discovered rules from the huge set of acceptable rules can cover Pos completely but they are often of high redundancy. As measures of accuracy and support are independent, there is not any total order among rules that we can use to guide the search. However, we can partially order the goodness of rules in terms of accuracy or support. Given two thresholds α and β, $0 \leq \alpha, \beta \leq 1$, on accuracy and support of rules, respectively. A rule R is $\alpha\beta$-strong if $acc(R) \geq \alpha$ and $sup(R) \geq \beta$. An $\alpha\beta$-strong rule R_i is said better than an $\alpha\beta$-strong rule R_j with respect to α if R_i has accuracy higher than that of R_j. An $\alpha\beta$-strong rule R_i is better than an $\alpha\beta$-strong rule R_j with respect to β if R_i has support higher than that of R_j. LUPC distinguishes three alternatives that occur in practice and that lead to the three corresponding types of search heuristics:

1. *Bias on rule accuracy*: It is to sequentially find rules with cover ratio equal and greater than $minc$ but accuracy is as large as possible.
2. *Bias on rule cover ratio*. It is to sequentially find rules with accuracy equal and greater than $mina$ but the cover ratio is as large as possible.
3. *Alternative bias on rule cover ratio and accuracy*. LUPC starts with highest values of α and β, and alternatively learns rules with bias on either accuracy or cover ratio, then reduces one of the corresponding α or β while keeping the other. The search is done until reaching the stopping condition.

Note that $cov^+(R)$ can be quickly determined because $|Pos| \ll |Neg|$. When searching for $\alpha\beta$-strong rules, a candidate rule will be eliminated without continuing to scan though large set Neg if this property holds during scanning.

Proposition 1. *Given a threshold α, a rule R is not $\alpha\beta$-strong for any arbitrary β if $cov^-(R) \geq ((1-\alpha)/\alpha) \times cov^+(R)$.*

[2] The Algorithm

Figure 1 presents the scheme of algorithm LUPC that consists of two

Learn-Positive-Rule$(Pos, Neg, mina, minc)$

1. $RuleSet = \phi$
2. $\alpha, \beta \leftarrow$ **Initialize**$(Pos, mina, minc)$
3. while $(Pos \neq \phi$ & $(\alpha, \beta) \neq (mina, minc))$
4. $\quad NewRule \leftarrow$ **BestRule**$(Pos, Neg, \alpha, \beta)$
5. $\quad$ if $(NewRule \neq \phi)$
6. $\quad\quad Pos \leftarrow Pos \setminus Cover^{+}(NewRule)$
7. $\quad\quad RuleSet \leftarrow RuleSet \cup NewRule$
8. $\quad$ else **Reduce**(α, β)
9. $RuleSet \leftarrow$ **PostProcess**$(RuleSet)$
10. return$(RuleSet)$

BestRule$(Pos, Neg, \alpha, \beta)$

11. $CandRuleset = \phi$
12. **AttValPairs**$(Pos, Neg, \alpha, \beta)$
13. while **StopCond**$(Pos, Neg, \alpha, \beta)$
14. $\quad$ **CandRules**$(Pos, Neg, \alpha, \beta)$
15. $BestRule \leftarrow$ First $CandidateRule$ in $CandRuleset$
16. return$(BestRule)$

Fig. 1 The scheme of algorithm LUPC

main procedures **Learn-Positive-Rule** and **BestRule**.

The procedure **Learn-Positive-Rule** starts with an empty *RuleSet* (line 1) and two adaptive parameters α and β, initialized by **Initialize** (lines 2). If the bias is on one accuracy, then β will be set to *minc*, and vise-versa. If the bias in on both accuracy and cover rate or there is no bias specified by the user, then both α or β are set as the biggest value. Lines 3-8 describe a recursive procedure to learn one the best rule among $\alpha\beta$-strong rules, to add it to the *RuleSet*, to remove positive instances covered by this rule under some conditions, and to change adaptively thresholds α and β. If there are any instances remain in *Pos*, and α and β are still equal or greater than *mina* and *minc* (line 3), **Learn-positive-rule** calls the subroutine **BestRule** to learn a new rule that is "best" with respect to the user-specified search bias (line 4). If **BestRule** yileds a rule (line 5), some positive instances covered by that rule will be removed from *Pos* (line 6) and the learned rule is added to the *RuleSet* (line 7). An instance is removed from *Pos* when learning a new rule if it is covered by the new rule, and previously covered by $\delta - 1$ rules in *RuleSet*. If **BestRule** cannot find rule, α and/or β will be adaptively reduced by the subroutine **Reduce** (line 8). The loop between lines 4-8 is repeated until the stopping condition (line 3) holds. The obtained *RuleSet* can be optionally post-processed by **PostProcess**(*RuleSet*) (line 9) before the procedure returns the final *RuleSet* (line 10). The removing of only positive instances covered by new rule (line 6) is an *one-sided selection*.

The procedure **BestRule** conducts a search in the rule space to find the "best" in the subset of generated $\alpha\beta$-strong rules given α and β. The **BestRule**

is composed of the subroutine **AttValPairs** (line 12) for determining the ordered set $AttValPairset$ of candidate attribute-value pairs to be used for generating candidate rules, and the subroutine **CandRules** (line 14) for determining the ordered set $CandRuleset$ of candidate rules that is set empty at the beginning (line 11). The subroutine **CandRules** may require a lot of checks on Neg to see if a generated candidate rule is $\alpha\beta$-strong. However, thanks to the property in Proposition 1, many candidate rules are quickly rejected if they are found to match the condition $cov^-(R) \geq ((1-\alpha)/\alpha) \times cov^+(R)$ during the scan of Neg. It is easy to count $cov^+(R)$ for each candidate rule R as Pos is small, and we need only to accumulate the count of $cov^-(R)$ when scanning Neg until either we can reject the candidate rule as the constraint holds or we completely go throughout Neg and find the rule has a satisfied accuracy. Two parameters η and γ can influence on the findings of LUPC. Generally, the higher value η and γ the higher chance to discover better rules.

2.3 Improved LUPC and Chance Discovery

We adopt the view that chances are events/situations, often rare, with significant impact on human decision making or influence.[10] As reported, LUPC is able to learn minority classes from large unbalanced datasets due to its search strategy, and experimental comparative evaluation has shown its high performance.[4] However, the original version of LUPC does not permit it to discover rare events with significance. The reason of this limitation, also in most learning methods, probably is that "a rule is ignored if its support or confidence was extremely small".[10] Rare events mean infrequent events, and in fact, rare events can be detected by a learning algorithm if using a very low threshold on support. However, the set of discovered patterns using a low support threshold is often too large to decide what to be selected. On the other hand, the significance is subjective and can only be judged by a human being, i.e., without the human evaluation, an event/situation cannot be known significant or not. In this sense, we have improved LUPC so that the user can play an active role in the learning process, and LUPC can support the chance discovery. Also, we focus on a particular kind of chances that are *unexpected* events/situations. Such chances can be detected in databases regarding their difference from common knowledge. This section concerns with two improvments of LUCP that relates to discovery of such chances: the first is visual LUPC, and the second is about using incoporated background knowledge.

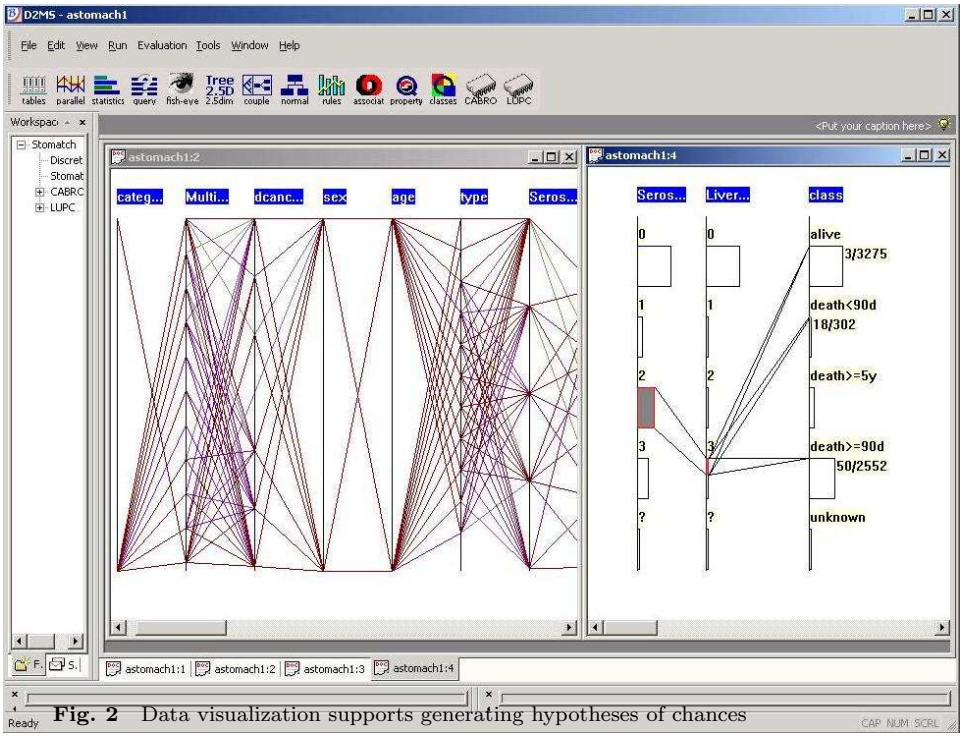

Fig. 2 Data visualization supports generating hypotheses of chances

[1] Visual LUPC

Visualization of data plays an extreme important role in understanding the application domain and in suggesting or verifying infrequent but significant relations in the data. We have chosen the parallel coordinate technique[1)] for visualizing 2D tabular datasets defined by n rows and p columns. LUPC improves parallel coordinates in several ways to adapt them to the knowledge discovery context. It is because when viewing a large dataset with many attributes, particularly categorical attributes, the advantage of parallel coordinates may lose as many polylines or their parts are partially overlapped, and certain kinds of summarization might be needed.

Viewing original data. The basic idea of viewing a p-dimensional dataset by parallel coordinates is to use p equally spaced axes to represent each data instance as a polyline that crosses each axis at a position proportional to its value for that dimension. This view gives the user a rough idea about the distribution of data on values of each attribute. The original stomach cancer dataset (Section 3) is visualized by parallel coordinates in LUPC (the left window

in Fig. 2).

Summarizing data. This view is significant as the dataset may be very large. The key idea is not to view original data points but to view their summaries on parallel attributes. LUPC uses bar charts in the place of attribute values on each axis. The bar charts in each axis have the same height (depending on the number of possible attribute values) and different widths that signify the frequencies of attribute values. LUPC also provides interactively common statistics on each attribute as mean or mode, median, variance, boxplots, etc.

Querying data. This view serves the hypothesis generation and testing by the user, in particular it helps to observe rare relations/hypotheses those may be significant in the user's view. The querying data allows the user to view such relations by queries. There are three types of queries: (1) those based on a value of the class attribute where the query determines the subset of all instances belonging to the indicated class; (2) those based on a value of a descriptive attribute where the query determines the subset of all instances having this value, and (3) those based on a conjunction of attribute-values pairs where the query determines the subset of all instances satisfied this conjunction. The grey regions on each axis show the proportions of specified instances on values of this attribute as shown in right window in Fig. 2.

Viewing rules. A rule is a pattern related to several attribute-values and a subset of instances. The importance in visualizing a rule is how this local structure is viewed in its relation to the whole dataset, and how the view support the user's evaluation on the rule interestingness. Thus, as a rule stands for an event/situation, this mode supports evaluation the significance of the rule. A rule is displayed by a subset of parallel coordinates included in antecedent and consequent. The LUPC's rule visualizer has the following functions. Each rule is displayed by polyline that goes through the axes containing attribute-values occurred on the antecedent part of the rule leading to the consequent part of the rule which are displayed with different color. The ratio associated with each class in the class attribute corresponds to the number of instances of the class covered by the rule over the total number of instances in the class (the upper-windows in Fig. 3).

Viewing rules and data. The significance of a rule can be seen better in its relation with the environment (the dataset). The subset of instances covered by a rule is visualized together with the rule by parallel coordinates or by summaries on parallel coordinates (the bottom window in Fig. 3). From this

Fig. 3 Rule visualization supports evaluating the significance of discovered chances

subset of instances, the user can see the set of rules each of them cover some of these instances, or the user can smoothly change the values of an attribute in the rule to see other related possible rules. These possible operations facilitate the user in evaluating if a rule is significant: a rule might be significant if instances covered by it are not recognized by other rules, and vice-versa. The rules for a class can be displayed together, and instances of the class as well of other classes covered by these rules are displayed.

[2] LUPC with Incoprporated Domain Knowledge

The domain knowledge is incoporated in LUPC as constraints (features) in the form of attribute-value pairs $A_{i_j} = v_{i_j}$. There are two kinds of constraints

that can be incorporated into the mining process of LUPC:

1. *Exclusive constraints*: If imposed, LUPC will find only rules that do not contain any of such attribute-value pairs in their condition part.
2. *Inclusive constraints*: If imposed, LUPC will find only rules each of them must contain at least one of such attribute-value pairs in their condition part.

The exclusive constraints serve to avoid mining rules with high frequent but less significant features, or to avoid mining rules having features that are unexpected/unuseful in some domain or class of objects (e.g., avoiding rules that contain typical features of a class). The inclusive constraints serve to mine rules containing some infrequent but significant features, or to mine rule containing features related to common knowledge in order to detect events that are contradictory with common knowledge of the domain.

The incorporated exclusise and inclusive constraints are used in the subroutines **AttValPairs** and **CandRules**. It is possible to put both kinds of exclusise and inclusive constraints in the mining process. Section 3 will illustrate how such domain knowledge is used to discover chances.

§3 Chance Discovery in Medical Data with LUPC

3.1 The Stomach Cancer and Hepatitis Datasets

[1] The Stomach Cancer Dataset

The stomach cancer dataset collected at the National Cancer Center in Tokyo during 1962-1991 is a very precious source for this research. It contains data of 7,520 patients described originally by 83 numeric and categorical attributes. These include information on patients, symptoms, type of cancer, longitudinal and circular location, serosal invasion, metastasis, pre-operative complication, post-operative complication, etc. One goal is to use attributes containing patient information before the operation to predict the patient status after the operation. The domain experts are particularly interested in finding predictive and descriptive rules for the class of patients who died within 90 days after operation amidst a total of 5 classes “death within 90 days”, “death after 90 days”, “death after 5 years”, “alive”, “unknown”).

The extraction of prediction and description rules for the class “death

within 90 days" from stomach cancer data is a difficult task because the classes in this dataset have a very imbalanced distribution and they are not well separated. Several well-known systems were applied to this dataset such as See5, a successor version of C4.5[13] and CBA[9], but the obtained results are far from expectation. However, the obtained results were far from expectations: they have low support and confidence, and usually relate to only a small percentage of patients of the target class. We have used visual interactive LUPC to investigate the stomach cancer data, and found significant results some of them are presented here, including preliminary analysis of data by visualization tool, unusual findings in two extreme classes "dead within 90 days" and "alive".

[2] The Hepatitis Dataset

The database is composed of 6 relational tables of time-series data on 983 laboratory examinations of 771 patients of hepatitis B and C. The data are broadly split into two categories. The first includes administrative information such as patient's information (age and date of birth), pathological classification of the disease, date of biopsy, result of biopsy, and duration of interferon therapy. The second includes temporal records of blood examination and urinalysis that can be fuPrther split into two subcategories, in-hospital and out-hospital examination data. In-hospital examination data contain the results of 230 examinations that were performed using the hospital's equipment. Out-hospital examination data contain the results of 753 examinations, including comments of staffs, performed using special equipment on the other facilities. Consequently, the temporal data contain the results of 983 types of examinations. The database is given recently to challenge the data mining reseach community (http://www.cs.helsinki.fi/events/ecmlpkdd/challenge.html). The medical doctors posed several challenging problems: (P1) Discover the course differences in temporal patterns between hepatitis B and C; (P2) Discover the relationships between the stage of liver fibrosis and the laboratory data; (P3) Discover the relation between effectiveness of interferon treatment and laboratory data; (P4) Determine the period until HCC; (P5) Determine the factors related to HCC.

3.2 Chance Discovery in Stomach Cancer Data

[1] Preliminary analysis of data with visualization tools

The visualization tools associated in LUPC allow us to examine the data

and to gain better insight into complex data before learning. While the viewing mode of original data offers an intuition about the distribution of individual attributes and instances, the summarizing and querying modes can suggest irregular or rare events to be investigated, or to guide which biases could be used to narrow the huge search space.

It is commonly known that patients who have symptoms "liver_metastasis" of all levels 1, 2, or 3 will certainly not survive. Also, "serosal_invasion = 3" is a typical symptom of the class "death within 90 days." With the visualization tools, we found several unusual events. For example, among 2329 patients in the class "alive", 5 of them have heavy metastasis of level 3, and 1 and 8 of them have metastasis level 2 and 1, respectively. Moreover, the querying data allow us to verify some significant combination of symptoms such as "serosal_invasion = 2" and "liver_metastasis = 3" as shown in the right window in Figure 2.

[2] Finding irregular rules in class "dead within 90 days"

It is commonly known that patients will die when liver metastasis occurs aggressively. Other learning methods when applied to this datasets often yield rules for the class "death within 90 days" containing "liver_metastasis" that are considered acceptable but not useful by domain experts. Also, these discovered rules usually cover only a subset of patients of this class. This low coverage means that there are patients of the class who are not included in "liver_metastasis" and, therefore, it is difficult to detect them.

Using visual interactive LUPC, we ran different trials and specified parameters and constraints to find only rules that do not contain the characterized attribute "liver_metastasis" and/or its combination with two other typical attributes, "Peritoneal_metastasis" and "Serosal_invasion." Below is a rule with accuracy 100% discovered by LUPC that can be seen as a rare and irregular event in the class.

```
Rule 8   accuracy = 1.0 (4/4), cover = 0.001 (4/6712)
IF       category = R AND sex = F AND proximal_third = 3
         AND middle_third = 1
THEN     class = death within 90 days
```

[3] Finding rare events in class "alive"

The prediction of rare events is becoming particularly interesting. When supposing that some attribute-value pairs may characterize some rare and/or significant events, LUPC, thanks to its associated visualization tools, allow us

examine effectively the hypothesis space and identify rare rules with any given small support or confidence. An example is to find rules in the class "alive" that contain the symptom "liver_metastasis." Such events are certainly rare and influence human decision making. We found rare events in the class "alive", such as male patients getting "liver_metastasis" at serious level 3 can survive with the accuracy of 50%.

Rule 1	accuracy = 0.500 (2/4); cover = 0.001(4/6712)
IF	sex = M AND type = B1 AND liver_metastasis = 3
	AND middle_third = 1
THEN	class = alive

3.3 Chance Discovery in Hepatitis Database

The tasks of knowledge discovery in P1-P5 require a great work of preprocessing as the hepatitis relational database is temporal with irregular periods and time-stamped data points. A temporal abstraction twchnique has been developed to transform temporal data of each patient into a record, i.e., to transform multi time-stamped points of each patient in one examination into a fix number of values in the record[5]. We skip the temporal abstraction step here, and describe chance discovery from transformed data where each examination is abstracted in the structure $< episode \ \& \ state - trend >$. We focus here on the problem P1 where two episodes are taken as 3 months and 6 months, the state has values "very low", "low", "normal", "high", "very high"; and the strend has composed values from "decreasing" and "increasing" found by a two-piecewise linear regression technique on each episode[5].

Based on common medical knowledge, the doctors group the changes of some significant examinations into two groups:

1. Short term change: GOT (up), GPT (up), TTT (up), ZTT (up).
2. Long term change: T-CHO (down), CHE (down), ALB (down), TP (down), PLT (down), WBC (down), HGB (down), T-BIL (up), D-BIL (up), I-BIL (up), ICG-15 (up).

We present here our results in attempts of finding unexpected patterns that may differ from the common knowledge. The findings suggest that many patterns that could be further considered.

[1] Unexpected patterns found in the short term change group

Our experiments consist of running LUPC to find all possible rules con-

taining examinations GOT, GPT, TTT, and ZTT with decreasing trends. With thresholds $mina = 95\%, minc = 2$ LUPC detected 254 rules (113 rules on hepatitis B and 142 rules on hepatitis C) that contain GOT, GPT, TTT, and ZTT with decreasing trends, i.e., these rules may be not consistent with the common medical knowledge. There is no dominant occurrence of decreasing trends of GOT, GPT, TTT, and ZTT in either hepatitis B or hepatitis C as in the previous case. For example, two rules found on hepatitis B and on hepatitis C are the followings:

Rule 66	accuracy = 1.0 (9/9); coverage = 0.013 (9/702)
IF	CHE6 = very low & decreasing-increasing, AND
	G.GL6 = high & decreasing-decreasing, AND
	TTT6 = high & decreasing-decreasing, AND
THEN	class = hepatitis B
Rule 151	accuracy = 1.0 (26/26); coverage = 0.037 (26/702)
IF	ALB6 = normal & decreasing-decreasing, AND
	LDH6 = normal & decreasing-decreasing, AND
	NA6 = normal & decreasing-decreasing, AND
	TTT6 = normal & decreasing-decreasing, AND
THEN	class = hepatitis C

[2] Unexpected patterns found in the long term change group

The common medical knowledge given by physicians is "damaged liver cannot produce ALB any more" and "low T-CHO relates to damaged liver" (T-CHO (down), and ALB (down)). Our experiments consist of running LUPC to find all possible rules containing examinations ALB and T-CHO with either increasing or decreasing trends. The outcomes are as follows. We found 274 rules (118 rules on hepatitis B and 156 rules on hepatitis C) with average accuracy 92.5% (on training data) that contain ALB and T-CHO with increasing trends, i.e., these rules may be not consistent with the common medical knowledge. The following two rules are examples of complex temporal abstractions that may suggest further investigations.

Rule 115	accuracy = 1.000 (5/5); coverage = 0.007 (5/702)
IF	ALP6 = low & increasing-decreasing, AND
	T-CHO12 = normal & increasing-increasing
THEN	class = hepatitis B
Rule 126	accuracy = 1.000 (13/13); coverage = 0.019 (13/702)
IF	T-CHO12 = normal & increasing-increasing, AND
	ZTT12 = high & increasing-increasing
THEN	class = hepatitis C

§4 Conclusions

We have shown a strong relation between chance discovery and learning from minority classes in large unbalanced datasets. In particular they share the objective of finding rare and significant unnoticed patterns (events) in data. Our visual interactive system LUPC illustrated the findings of such knowledge from the stomach cancer and hepatitis databases.

References

1) Fayyad, U.M., Grinstein. G.G., and Wierse, A., *Information Visualization in Data Mining and Knowledge Discovery*, Morgan Kaufmann, 2002.
2) Furnkranz, J., "Separate-and-Conquer Rule Learning, *Journal Artificial Intelligence Review*, *13*, 3–54, 1999.
3) Ho, T.B., Nguyen T.D., Nguyen D.D., Kawasaki S., "Visualization Support for User-Centered Model Selection in Knowledge Discovery and Data Mining", *International Journal of Artificial Intelligence Tools*, Vol. 10, No. 4, 691-713, 2001.
4) Ho, T.B., Nguyen, D.D., Kawasaki, S., "Mining Prediction Rules from Minority Classes", *14th International Conference on Applications of Prolog (INAP2001)*, Tokyo, October 2001, 254-264.
5) Kawasaki, S., Nguyen, D.D., Nguyen, T.D., Ho, T.B., "Study of Hepatitis Data by Visual Data Mining System D2MS, *JSAI SIG-KBS-A201 Workshop Active Data Mining*, Pusan, 43–48, May 2002.
6) Japkowicz, N.,"The Class Imbalance Problems: Significance and Strategies. *AAAI Workshop on Learning in Imbalanced Datasets*, 2000.
7) Kubat, M., and Marvin, S., "Addressing the Curse of Imbalanced Training Sets: One-Sided Selection", *Proceedings of the Fourteenth International Conference on Machine Learning*, 179–186, 1997.
8) Ling, C. X. & Li, C, "Data Mining for Direct Marketing: Problems and Solutions", *International Conference on Knowledge Discovery and Data Mining KDD-97*, 258–267, 1997.
9) Liu, B., Hsu, W., & Ma, Y., "Integrating Classification and Association Rule Mining", *Fourth Conference on Knowledge Discovery and Data Mining*, 80–86, 1998.
10) Ohsawa, Y., "Chance Discoveries for Making Decisions in Complex Real World", *New Generation Computing*, Vol. 20, No. 2, 2002.
11) Padmanabhan, B. & Tuzhilin, A., "Knowledge Refinement based on the Discovery of Unexpected Patterns in Data Mining", *Decision Support Systems*, Vol. 33, 309–321, 2002.
12) Provost, F., "Learning with Imbalanced Data Sets" *AAAI'2000 Workshop on Imbalanced Data Sets*, 2000.
13) Quinlan, J. R., *C4.5: Programs for Machine Learning*. San Francisco: Morgan Kaufmann, 1993.

14) Ting, K. M.,"A Comparative Study of Cost-Sensitive Boosting Algorithms", *Seventeenth International Conference on Machine Learning*, 983–990, 2000.

15) Turney, P.D., "Cost-Sensitive Classification: Empirical Evaluation of a Hybrid Genetic Decision Tree Induction Algorithm", *Journal of Artificial Intelligence Research 2*, 369–409, 1995.