Learning Closed Sets of Labeled Graphs for Chemical Applications

Sergei O. Kuznetsov and Mikhail V. Samokhin

All-Russia Institute for Scientific and Technical Information (VINITI),
Moscow, Russia

Abstract. Similarity of graphs with labeled vertices and edges is naturally defined in terms of maximal common subgraphs. To avoid computation overload, a parameterized technique for approximation of graphs and their similarity is used. A lattice-based method of binarizing labeled graphs that respects the similarity operation on graph sets is proposed. This method allows one to compute graph similarity by means of algorithms for computing closed sets. Results of several computer experiments in predicting biological activity of chemical compounds that employ the proposed technique testify in favour of graph approximations as compared to complete graph representations: gaining in efficiency one (almost) does not lose in accuracy.

1 Introduction

In last years the problem of learning from data given by labeled graphs attracted much attention in Machine Learning and Data Mining communities [1,2,3,4,5,6,7,8,9]. In our paper we address this issue using an approach based on generation of closed sets of labeled graphs and their approximations. On the one hand, this approach is related to computation of most specific (or least general) generalizations of positive (or negative) examples, which proved to be successful in real-life applications, including predictive toxicology [10]. On the other hand, generation of (frequent) closed itemsets turned out to be useful for computing the set of all well-supported association rules [11]. This explains recent attention to computing closed graphs in data mining [8]. As reported in [8], CloseGraph algorithm computes frequent graphs much faster than its forerunner gSpan [7], and WARMR [1], an ILP program.

An important application for learning with labeled graphs is the analysis of properties of chemical substances. Fragmentary Code of Substructure Superposition (FCSS) [12,10] has been designed and permanently refined for this purpose and proved to be a very efficient tool. For example, it was successfully applied (as estimated by ROC diagrams) in the open PTC competition [13,10,14]. As reported in [14], FCSS produced the largest number of useful attributes in comparison with other representations used in PTC. The drawbacks of FCSS are related to the loss of information about connection between molecular parts and the lack of flexibility w.r.t. different problems. To compensate for this, a similarity operation $\sqcap$ on sets of labeled graphs, representing molecules, was proposed

S. Kramer and B. Pfahringer (Eds.): ILP 2005, LNAI 3625, pp. 190–208, 2005.

in [15,16,3]. This operation, defining similarity of sets of labeled (hyper)graphs, has the property of a semilattice: it is idempotent ($X \sqcap X = X$), commutative ($X \sqcap Y = Y \sqcap X$), and associative ($X \sqcap (Y \sqcap Z) = (X \sqcap Y) \sqcap Z$). This allows one to compute similarity of graph sets by means of algorithms for computing closed sets (see review [17]) well-known in Formal Concept Analysis [18].

The main problem with practical implementation of this operation is that of computational complexity: to compute similarity of two graphs one needs to make several tests of subgraph isomorphism (which is in general NP-complete), and make tests for graph isomorphism.

A theoretical means for approximate computation in semilattices, called projections, was proposed in [19] and the first computer implementation was described in [20]. In this paper we study projections for semilattices on graph sets and their use in learning models. Here we consider a realization of similarity operation on graph sets and their projections realized by means of certain order-theoretic and lattice-theoretic techniques. We consider several applied problems in the analysis of biological activity of chemical compounds. To predict target attribute values (biological activities) we employ and compare several learning models: induction of decision trees, Naive Bayes classifier (see, e.g., [21]) and JSM-method or concept-based learning [22,3,19]. In this paper the issues of program realization and efficiency are not considered in details, since our programs are Java prototypes. We concentrate mostly on combinations of learning models with representation languages, and evaluations of their predictive accuracy. Results obtained for learning with graph projections for various values of projection parameter are compared with those obtained with FCCS representation.

The paper is organized as follows. In the second section we describe the general theoretical framework for computing similarity (meet) of graph sets together with a means for its approximate computations. In the third section we discuss the learning models used in this work. In the fourth section we describe computer experiments in the analysis of molecular graphs (of chemical compounds from the PTC dataset [13], halogen-substituted aliphatic hydrocarbons, alcohols, etc.) where the above representations and learning models are used. In the fifth section the results are discussed and some conclusions are made.

2 Closed Sets of Labeled Graphs and Their Projections

In [15,16,3] a semilattice on sets of graphs with labeled vertices and edges was proposed. This lattice is based on a natural domination relation between graphs with labeled vertices and edges. Consider an ordered set P of connected graphs[1] with vertex and edge labels from the set $\mathcal{L}$ with partial order $\preceq$. Each labeled graph Γ from P is a quadruple of the form $((V, l), (E, b))$, where V is a set of vertices, E is a set of edges, $l\colon V \to \mathcal{L}$ is a function assigning labels to vertices, and $b\colon E \to \mathcal{L}$ is a function assigning labels to edges.

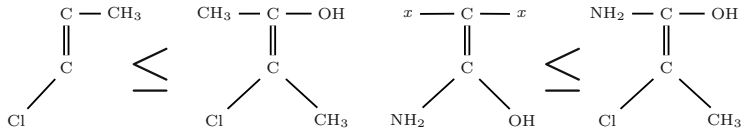
[1] Omitting the condition of connectedness, one obtains a (computationally harder) model that accounts for multiple occurrences of subgraphs.

For two graphs $\Gamma_1 := ((V_1, l_1), (E_1, b_1))$ and $\Gamma_2 := ((V_2, l_2), (E_2, b_2))$ from P we say that Γ_1 **dominates** Γ_2 or $\Gamma_2 \leq \Gamma_1$ (or Γ_2 is a **subgraph** of Γ_1) if there exists an injection $\varphi\colon V_2 \to V_1$ such that it

- respects edges: $(v, w) \in E_2 \Rightarrow (\varphi(v), \varphi(w)) \in E_1$,
- fits under labels: $l_2(v) \preceq l_1(\varphi(v))$, if $(v, w) \in E_2$ then $b_2(v, w) \preceq b_1(\varphi(v), \varphi(w))$.

Obviously, $(P, \leq)$ is a partially ordered set.

Example 1. Let $\mathcal{L} = \{\mathrm{C}, \mathrm{NH_2}, \mathrm{CH_3}, \mathrm{OH}, x\}$ then we have the following relations:



vertex labels are unordered $\qquad$ $x \preceq A$ for any vertex label $A \in \mathcal{L}$

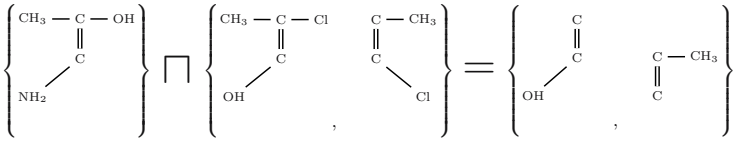
Now a *similarity operation* $\sqcap$ on graph sets can be defined as follows: For two graphs X and Y from P

$$\{X\} \sqcap \{Y\} := \{Z \mid Z \leq X, Y,\ \forall Z_* \leq X, Y\ Z_* \not\geq Z\},$$

i.e., $\{X\} \sqcap \{Y\}$ is the set of all maximal common subgraphs of graphs X and Y. Similarity of non-singleton sets of graphs $\{X_1, \ldots, X_k\}$ and $\{Y_1, \ldots, Y_m\}$ is defined as

$$\{X_1, \ldots, X_k\} \sqcap \{Y_1, \ldots, Y_m\} := \mathrm{MAX}_{\leq}(\cup_{i,j}(\{X_i\} \sqcap \{Y_j\})),$$

where $\mathrm{MAX}_{\leq}(X)$ returns maximal (w.r.t. $\leq$) elements of X. Here is an example of applying $\sqcap$:



The similarity operation $\sqcap$ on graph sets is commutative: $X \sqcap Y = Y \sqcap X$ and associative: $(X \sqcap Y) \sqcap Z = X \sqcap (Y \sqcap Z)$.

A set X of labeled graphs from P for which $\sqcap$ is idempotent, i.e., $X \sqcap X = X$ holds, is called a *pattern*. For patterns we have $\mathrm{MAX}_{\leq}(X) = X$. For example, for each graph $g \in P$ the set $\{g\}$ is a pattern. On the contrary, for $\Gamma_1, \Gamma_2 \in P$ such that $\Gamma_1 \leq \Gamma_2$ the set $\{\Gamma_1, \Gamma_2\}$ is not a pattern. Denote by D the set of all patterns, then $(D, \sqcap)$ is a semilattice with infimum (meet) operator $\sqcap$. The natural subsumption order on patterns is given by

$$c \sqsubseteq d :\iff c \sqcap d = c.$$

Let E be a set of example names, and let $\delta : E \to D$ be a mapping, taking each example name to $\{g\}$ for some labeled graph $g \in P$ (thus, g is "graph description" of example e). The triple $(E, (D, \sqcap), \delta)$ is a particular case of a *pattern structure* [19]. Another example of an operation $\sqcap$ may be the following semilattice on closed intervals from [16]: for $a, b, c, d \in R$, $[a, b] \sqcap [c, d] = [\max\{a, c\}, \min\{b, d\}]$ if $[a, b]$ and $[c, d]$ overlap, otherwise $[a, b] \sqcap [c, d] = \emptyset$. This semilattice, where numbers are values of activation energy (computed for molecules by a standard procedure, e.g. see [23]) was used in predicting toxicity of alcohols and halogen-substituted hydrocarbons (see Section 4). The resulting similarity semilattice in this application is that on pairs, where the first element is a graph set and the second element is a numerical interval.

Derivation operators are defined as

$$A^\diamond := \sqcap_{e \in A}\ \delta(e) \qquad \text{for } A \subseteq E$$

and

$$d^\diamond := \{e \in E \mid d \sqsubseteq \delta(e)\} \qquad \text{for } d \in D.$$

For $a, b \in D$ the *pattern implication* $a \to b$ holds if $a^\diamond \sqsubseteq b^\diamond$. Implications are exact association rules (with confidence = 1). Operator $(\cdot)^{\diamond\diamond}$ is an algebraical closure operator [24,18] on patterns, since it is

idempotent: $d^{\diamond\diamond\diamond} = d^{\diamond\diamond}$,
extensive: $d \sqsubseteq d^{\diamond\diamond}$,
monotone: $d^{\diamond\diamond} \sqsubseteq (d \sqcup c)^{\diamond\diamond}$.

For a set X the set $X^{\diamond\diamond}$ is called *closure* of X. A set of labeled graphs X is called *closed* if $X^{\diamond\diamond} = X$. This definition is related to the notion of a closed graph [8], which is important for computing association rules between graphs. Closed graphs are defined in [8] in terms of "counting inference" as follows.

Given a labeled graph dataset D, support of a graph g or *support*(g) is a set (or number) of graphs in D, in which g is a subgraph. A graph g is called *closed* if no supergraph f of g (i.e., a graph such that g is isomorphic to its subgraph) has the same support.

Note that the definitions distinguish between a closed graph g and the closed set $\{g\}$ consisting of one graph g. Closed sets of graphs form a *meet semilatice* w.r.t. infimum or meet operator. A finite meet semilattice is completed to a lattice by introducing a unit (maximal) element. Closed graphs do not have this property, since in general, there can be nonunique supremums and infimums of two closed graphs.

Proposition. Let a dataset described by a pattern structure $(E, (D, \sqcap), \delta)$ be given. Then the following two properties hold:

1. For a closed graph g there is a closed set of graphs G such that $g \in G$.
2. For a closed set of graphs G and an arbitrary $g \in G$, graph g is closed.

Proof. 1. Consider the closed set of graphs $G = \{g\}^{\diamond\diamond}$. Since G consists of all maximal common subgraphs of graphs that have g as a subgraph, G contains as an element either g or a supergraph f of g. In the first case, property 1 holds. In

the second case, we have that each graph in G that has g as a subgraph also has f as a subgraph, so f has the same support as g, which contradicts with the fact that g is closed. Thus, $G = \{g\}^{\diamond\diamond}$ is a closed set of graphs satisfying property 1. 2. Consider a closed set of graphs G and $g \in G$. If g is not a closed graph, then there is a supergraph f of it with the same support as g has and hence, with the same support as G has. Since G is the set of all maximal common subgraphs of the graphs describing examples from the set $G^{\diamond}$ (i.e, its support), $f \in G$ should hold. This contradicts the fact that $g \in G$, since a closed set of graphs cannot contain as elements a graph and a supergraph of it (otherwise, its closure does not coincide with itself). □

Therefore, one can use algorithms for computing closed sets of graphs, e.g., the algorithm in [3], to compute closed graphs. With this algorithm one can also compute all *frequent* closed sets of graphs, i.e., closed sets of graphs with support above a fixed *minsup* threshold (by introducing a minor variation of the condition that terminates computation branches).

Computing $\sqcap$ may require considerable computation resources: even testing $\sqsubseteq$ is NP-complete. To approximate graph sets we consider projection (kernel) operators [19], i.e. mappings of the form $\psi\colon D \to D$ that are

monotone: if $x \sqsubseteq y$, then $\psi(x) \sqsubseteq \psi(y)$,
contractive: $\psi(x) \sqsubseteq x$, and
idempotent: $\psi(\psi(x)) = \psi(x)$.

Any projection of the semilattice $(D, \sqcap)$ is $\sqcap$-preserving, i.e., for any $X, Y \in D$

$$\psi(X \sqcap Y) = \psi(X) \sqcap \psi(Y),$$

which helps us to relate learning results in projections to those with initial representation (see Section 3).

As for practical complexity of computing $\sqsubseteq$ we can say the following. Using a Pentium PIII-1 GHz, 512 MB RAM, testing subgraph isomorphism for an average graph with 30-40 vertices and 30-40 edges took up to 5 seconds, but usually, less than a second.

In our computer experiments we used several types of projections of sets of labeled graphs that are natural in chemical applications:

- *k-chain* projection: a set of graphs X is taken to the set of all chains with k vertices that are subgraphs of at least one graph of the set X;
- *k-vertex* projection: a set of graphs X is taken to the set of all subgraphs with k vertices that are subgraphs of at least one graph of the set X;
- *k-cycles* projection: a set of graphs X is taken to the set of all subgraphs consisting of k adjacent cycles of a minimal cyclic basis of at least one graph of the set X.

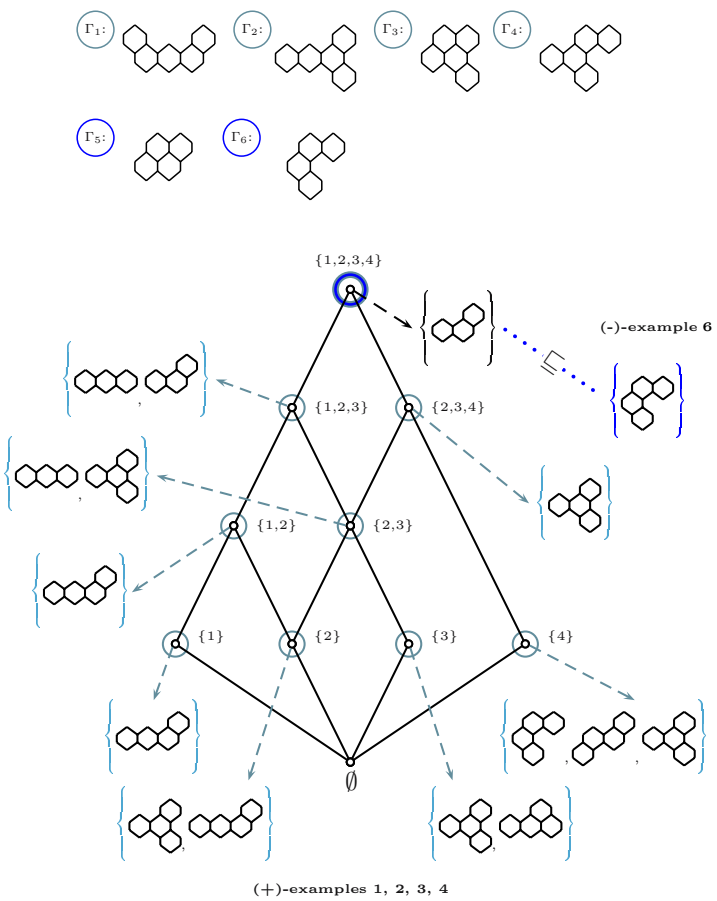


Fig. 1.

3 Learning Models

In this work we used several learning models realized within QuDA data miners' discovery environment [25][2]: concept-based learning (JSM-method) [22,3,26] and several machine learning algorithms from the Weka workbench [27]: C4.5 algorithm for induction of decision trees, Naive Bayes classifier, and JRip (induction of ripple-down rules).

JSM-hypotheses were defined in [22] for standard object-attribute representation in a special logical language. These hypotheses were redefined as *JSM-* or *concept-based hypotheses* in [16,3,26,19] in terms of Formal Concept Analysis (FCA). For graph sets hypotheses can be defined as follows. Suppose we have a set of positive examples E_+ and a set of negative examples E_- w.r.t. a target attribute.

[2] Free download: http://www.intellektik.informatik.tu-darmstadt.de/~peter/

A graph set $h \in D$ is a *positive hypothesis* iff

$$h^\diamond \cap E_- = \emptyset \text{ and } \exists A \subseteq E_+ : A^\diamond = h.$$

Informally, a positive hypothesis is a similarity of positive examples, which does not cover any negative example. A *negative hypothesis* is defined analogously, by interchanging + and −.

The meet-preserving property of projections implies that a hypothesis H_p in data under projection ψ corresponds to a hypothesis H in the initial representation for which the image under projection is equal to H_p, i.e., $\psi(H) = H_p$.

Hypotheses are used for classification of undetermined examples along the lines of [22] in the following way. If e is an undetermined example (example with the unknown target value), then a hypothesis h with $h \sqsubseteq \delta(e)$ is *for the positive classification* of g if h is a positive hypothesis and h is *for the negative classification* of e if h is a negative hypothesis.

An undetermined example e is *classified positively* if there is a hypothesis for its positive classification and no hypothesis for its negative classification. Example e is *classified negatively* in the opposite case. If there are hypotheses for both positive and negative classification, then some other methods (e.g., based on standard statistical techniques) may be applied. Obviously, for classification purposes it suffices to use only hypotheses minimal w.r.t. subsumption $\sqsubseteq$.

The definition of classification suggests that hypotheses can be considered as disjunctions of *lgg*s of positive and of negative examples. Notwithstanding its simplicity, the model of learning and classification with concept-based hypotheses proved to be efficient in numerous computer experiments, including PTC competition [13,10]. This learning/classification model, together with FCSS representation, produced Pareto-optimal classifications in each of the four sex/species groups (from {mice, rats} × {male, female}): in three groups the results were on the ROC curve and in the fourth group (male rats, MR) the result was slightly below the curve with no other strictly better classification result.

An algorithm for computing hypotheses on closed graph sets was described in [3]. Here we realize it by simulating $\sqcap$ operation with usual set-theoretic intersection $\cap$ in the following way. For each example e described by a labeled graph $\delta(e)$ first a set of all subgraphs of $\delta(e)$ is computed up to the projection level $k = N$. Each such subgraph is declared to be a binary attribute and example e is represented by the set $S(e)$ of binary attributes that correspond to subgraphs of $\delta(e)$. For two examples e_1 and e_2 intersection $S(e_1) \cap S(e_2)$ is equivalent to finding similarity $\psi(\delta(e_1)) \sqcap \psi(\delta(e_2))$.

Example 2. In Figure 1 consider JSM-hypotheses for the dataset with positive examples described by graphs Γ_1,..,Γ_4 and negative examples described by graphs Γ_5 and Γ_6. Here $\Gamma_1 \sqcap \Gamma_2 \sqcap \Gamma_3$ and $\Gamma_2 \sqcap \Gamma_3 \sqcap \Gamma_4$ are minimal positive hypotheses, whereas $\Gamma_1 \sqcap \Gamma_2 \sqcap \Gamma_3 \sqcap \Gamma_4$ is not a positive hypothesis.

Then standard Weka procedures for C4.5, Naive Bayes and JRip are run in QuDA environment. Computing concept-based hypotheses in QuDA is realized by means of algorithms for computing lattices of closed sets (or concept lattices), see review [17].

After that we perform *reduction of attributes* [18]: each column of the example/attribute binary table that is equal to the (component-wise) product (conjunction) of some other columns, is removed.

Reduction is realized by an efficient algorithm based on results from FCA [18]. Lattice-theoretical properties guarantee [18] that thus reduced set of columns gives rise to the isomorphic lattice of closed sets of attributes and thus, to the same set of concept-based hypotheses as defined above.

Since in practice reduction often results in diminishing sets of attributes in several times (see experimental results in Section 4), in our experiments we wanted to find out how reduction affects performance of other learning methods, such as C4.5, Naive Bayes and JRip. Upon reduction, every learning method was executed for data tables again. Results for reduced and nonreduced tables were compared.

The general **PBRL** (project-binarize-reduce-learn) procedure looks as follows:

1. For each example e and for k compute i-projections of $\delta(e)$ for $1 \leq i \leq k$. The subgraphs from this projections are declared to be binary attributes;
2. Compose example/attribute binary table;
3. For each learning method LM run LM, classify examples from test sets, compute cross validation;
4. Reduce the binary (example/attribute) table;
5. For reduced table and for each learning method LM run LM, classify examples from test sets, compute cross-validation.

General procedure for computing with FCSS looks similar with first two lines replaced by the following ones:

1*. For each example e compute FCSS code (set of FCSS descriptors) of its molecular graph;
2*. Compose example/attribute binary table, where each attribute stays for an FCSS descriptor;

Another approach that uses learning with graph sets was realized by means of Subdue and SubdueCL [28,9] systems. Subdue finds subgraphs that appear repetitively in graph databases. SubdueCL can learn from positive and negative examples. It generates graphs common to *many* positive examples that are common to a *small* amount of negative examples (the corresponding values are captured exactly within the *error* estimate). As reported in [9], SubdueCL slightly outperformed ILP systems FOIL [29] and Progol [30] on the PTC dataset.

SubdueCL pursues the covering strategy: having found a subgraph with the best error estimate, SubdueCL excludes positive examples covered by this subgraph (i.e., example descriptions that contain it as a subgraph) and iterates on the remaining set of positive examples. Thus, skipping certain generalizations of positive examples, Subdue performs efficiently, however may lose in learning accuracy. The latter is much more important in such domains as Predictive Toxicology, where SubdueCL [31], as estimated by ROC diagrams, was outperformed

by the concept-based learning model [10] (classifications of SubdueCL were optimal only in one group (male rats) and were strictly worse than concept-based hypotheses [10] for male and female mice.

4 Experiments with Projections of Labeled Graphs

In this section we analyse results of applying the introduced data representation and learning models to the analysis of several chemical datasets [3]. For each dataset we computed graph projections (mostly, k-vertex projections, except for the 25PAH dataset (Section 4.5), where we computed k-cycles projections). Every subgraph of each graph in the projection (up to isomorphism) was declared to be a binary attribute, so each graph dataset was turned into a binary object-attribute table, which was then reduced. We also computed FCSS codes for each dataset. After that for each dataset we ran several learning methods realized within QuDA environment (JSM or concept-based hypotheses, induction of decision trees by C4.5, Naive Bayes, JRip). We computed 10-fold cross-validation and in several cases (PTC, halogen substituted hydrocarbons, alcohols, polycyclic aromatic hydrocarbons), where a known test set was available, we performed classifications for the test set. We compared cross-validation and results on the test set for each chemical dataset. Results of the analysis are presented in similar tables. For PTC datasets we plotted results of our experiments on the ROC curves of the PTC workshop [13].

4.1 Experiments with PTC Dataset

Participants of the workshop on Predictive Toxicology Challenge (PTC) [13] discussed results of competition of machine learning programs that generated hypothetical causes of toxicity from positive and negative examples.

The training dataset consisted of descriptions of 409 molecular graphs of chemical compounds with indication of whether a compound is toxic or not for a particular sex/species group out of four possible groups: {mice, rats} × {male, female}. For each group there were about 120 to 150 positive examples and 190 to 230 negative examples of toxicity. The test dataset consisted of 185 substances for which forecasts of toxicity should be made.

The average size of the initial graphs was 25 vertices and 26 edges in the training set, and 45 vertices and 46 edges in the test set. We generated graph k-vertex projections for k from 1 to 8, thus producing 8 binary object-attribute tables. For $k = 9$ we computed projections in 30 hours, but had to stop generation of the binary object-attribute matrix (which involves testing graph isomorphism) after 70 hours, having obtained 561921 attributes. With the growth of k, the number of attributes in the resulting tables becomes large, but reduction of attributes diminishes the size of tables in several times. Compared to computing projections of initial graphs (which comprises the major part of computation) and hypothesis generation, the reduction is relatively fast, see Table 1.

[3] These datasets can be downloaded from http://ilp05-viniti.narod.ru

Table 1. PTC dataset: number of attributes in representation tables before and after attribute reduction

projection size	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
# attributes in full table	22	95	329	1066	3275	9814	28025	76358
# attributes in reduced table	22	72	153	373	812	1548	2637	3981
reducing time (in sec.)	1	1	2	5	16	57	219	883

To estimate different classification strategies in combination with k-projections ($1 \leq k \leq 13$) the 10-fold cross-validation procedure was used for the given training dataset. Table 2 shows the best results w.r.t. predictive accuracy and total number of predictions. The best strategy for MR group w.r.t. predictive accuracy is the one based on JSM-hypotheses. This strategy attains predictive accuracy of 58% with k-projection representation. For FR group the best result (predictive accuracy of 66%) was obtained by JRip rules in combination with k-projections. The use of FCSS representation leads to the following results. For MR group the best strategies w.r.t. predictive accuracy are JSM-hypotheses and C4.5 algorithm, the both strategies attain predictive accuracy of 52%. For FR group JSM-hypotheses with FCSS codes also is the best strategy w.r.t. predictive accuracy. It attains predictive accuracy of 56%.

If we consider both precision and number of predictions then the best result for k-projections representation for MR group is obtained by Naive Bayes (it attains predictive accuracy of 56% and 64% of total number of predictions). For FR group JRip results in 66% of predictive accuracy with 40% of total number of predictions. Naive Bayes also turns out to be the best strategy in combination with FCSS representation for both groups (predictive accuracy of 51% and 49% of total number of predictions for MR group; corresponding values for FR group are 50% and 25%). The results of 10-fold cross-validation suggest that the performance of learning methods stabilizes with the growth of k.

Table 2. The results of 10-fold cross-validation procedure for PTC dataset obtained with JSM-hypotheses (**J**), C4.5 (**C**), Naive Bayes classifier (**N**), and JRip rules (**R**) with FCSS-encoding (**F**) and $3, \ldots, 14-$ projections (**PR**); **A** – predictive accuracy, **TP** – total number of predictions

	MR (male rats)								**FR (female rats)**							
	J-F	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**	**J-F**	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**
A	0.523	0.527	0.511	0.475	0.586	0.556	0.552	0.556	0.560	0.462	0.500	0.385	0.464	0.571	0.468	0.662
TP	0.164	0.397	0.493	0.199	0.266	0.643	0.552	0.448	0.123	0.263	0.246	0.044	0.109	0.403	0.429	0.395

4.2 Classification in Projections Estimated by ROC Curves

The results are shown in Figure 2, where the following abbreviations are used:

– J-PR1, J-PR2, ..., J-PR8 – the results obtained using 1- to 8-projection representations, respectively, in combination with JSM-hypotheses; similarly, for

other methods (C4.5, Naive Bayes, JRip), the results marked as C45-PRi, NB-PRi and R-PRi (where $1 \leq i \leq 8$);

– WAI1, GONZ, KWAI, LEU3 are other Pareto-optimal models submitted to the Predictive Toxicology Challenge for this animal group.

Note that the Figure 2 shows both the "old" ROC-curve (composed by LEU3, KWAI, GONZ, and WAI1 models) and the "new" one (composed by LEU3, J-PR5, C45-PR3, NB-PR3, and R-PR7 models).

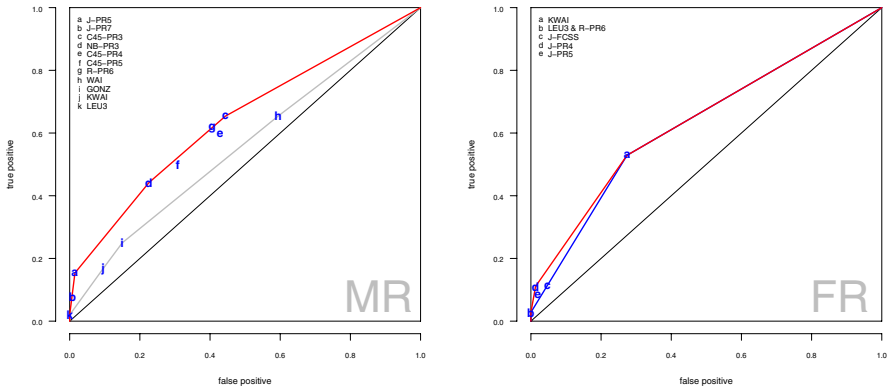


Fig. 2. Projected pattern structures "On the ROC" for groups MR (male rats) and FR (female rats)

For the MR group (male rats; see Figure 2) the following results were obtained. The use of k-projections (with $k \leq 3$) together with JSM-hypotheses does not lead to any good classifications. However, the C4.5 algorithm and Naive Bayes classifier appears on the "new" ROC-curve. The use of 4-projections in combination with JSM-hypotheses and C4.5 results in better classifications: the corresponding points are above the "old" ROC-curve. The 5-projections representation in combination with JSM-hypotheses happens to be one of the five "new" best strategies: it results in making 8 true positive predictions with only 2 false positive ones. As in the case with JSM-hypotheses the use of decision tree induction strategy leads to the classification that is also better than those on the "old" ROC-curve. The use of 6-projections with JSM-hypotheses, however, does not result in better classification: the number of true positives decreases to 6; the number of false positives remains the same. At the same time new classifications of the C4.5 are among the best ones. The corresponding point lies on the "new" ROC-curve. The use of 7-projections and JSM-hypotheses, with 4 true positives and 1 false positives again appears on the "new" ROC-curve. The classification based on the 8-projections representation and JSM-hypotheses increases the number of true positives to 6 but also increases the number of false positives to 2; this strategy is thus strictly worse than using 5-projections (assuming positive cost of making a true positive classification).

For the FR group (female rats; see Figure 2) the points corresponding to the results for 4-, 5-, 6-, and 8-projections in combination with JSM-hypotheses, also lie above the "old" PTC ROC-curve, where concept-based hypotheses were computed for FCSS representation. However, other methods do not lead to any good classifications. None of them in combination with k-projections appears above the "old" ROC-curve. There was only one exception: ripple-down rules (JRip) using 6-projections representation show the same result as LEU3.

Computer experiments with PTC data in comparison of FCSS, k-projections and reduced/nonreduced tables showed that the use of reduced tables, as compared with nonreduced ones, does not make any difference for concept-based hypotheses, makes a very slight difference (no more than 5%) for induction of decision trees and small difference (about 10-15%) for other methods such as Naive Bayes and JRip. As for comparison of projections of different type, first there is an obvious improvement with the growth of the projection parameter k. Starting from a certain value of k there is no further improvement.

4.3 Toxicity of Alcohols

In [32] the results of studies on the relationships between structures of miscellaneous alcohols (from [33]) and their acute toxicity for rats and mice using JSM-(concept-based) hypotheses with FCSS representation are described.

The training set contains descriptions of 89 molecular graphs of chemical compounds with indication of acute toxicity degree (high, moderate, and low). Separate computations were made for two target values: high and moderate. In the first case moderate and low toxic substances were considered as negative examples. In the second case only low toxic substances were considered as negatives. The test set consisted of 22 substances. The average size of a molecular graph was 24 vertices and 23 edges.

Tables 3 and 4 report on the results obtained with FCSS and k-projections ($4 \leq k \leq 13$) in combination with various learning models.

Table 3. Toxicity of alcohols: results obtained with JSM-hypotheses, FCSS-encoding (**F**) and $4-, \ldots, 13-$projections

	F	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
# correct predictions	8	9	10	10	14	14	13	12	11	11	11
# incorrect predictions	2	0	3	1	1	2	3	3	3	3	3
# unclassified substances	12	13	9	11	7	6	6	7	8	8	8

Table 4. Predicting toxic potential of alcohols: the best results obtained with JSM-hypotheses (**J**), C4.5 (**C**), Naive Bayes classifier (**N**), and JRip rules (**R**) in combination with FCSS-encoding (**F**) and $3, \ldots, 14-$ projections (**PR**)

	J-F	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**
# correct predictions	8	19	16	19	14	19	12	17
# incorrect predictions	2	3	6	3	1	3	10	5

For k-projections with $1 \leq k \leq 3$ there was no classification with JSM-hypotheses whatsoever. For $k = 4$ with JSM-hypotheses a result is better than that for FCSS was obtained. For $k = 5$ and $k = 6$ results are not comparable with those for $k = 4$: the number of correct classifications is 10, but the numbers of incorrect predictions are equal to 3 and 1, respectively. The result obtained with JSM-hypotheses for 7-projections is among the best results for $1 \leq k \leq 13$: 14 correct predictions with only 1 mistake. Starting with $k = 8$ the growth of k results in the decrease of predictive accuracy for JSM-hypotheses. The predictive accuracy of other methods also decreases with the growth of k. For example, the use of C4.5 algorithm and 8-projections leads to the classification with 19 correct predictions and 3 incorrect predictions, but the use of 11-projections representation in combination with the same learning model results in 18 correct predictions with 4 mistakes.

In general, we observe that the use of reduced vs. nonreduced tables does not affect results obtained with the JSM-hypotheses and slightly affects results of other methods. The best classifications were obtained for average projection values ($4 \leq k \leq 8$). Experimental complexity of computing projections for this dataset, is given in Table 5

Table 5. Alcohol dataset: time of computing projections

size of projection	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
time elapsed, sec.	12	21	44	109	317	937	3163	12402	45822	156297

4.4 Predicting Carcinogenic Potential in Halogen-Substituted Aliphatic Hydrocarbons

The training set [34] contained descriptions of 57 molecular graphs with values of carcinogenic potential. The unique target property here was "to be carcinogenic". The test set consisted of 13 molecular graphs. The results for different k-projections and FCSS in combination with different learning models are shown in Table 6.

Table 6. Predicting carcinogenic potential in hydrocarbons: the results obtained with JSM-hypotheses and FCSS-encoding (**F**) and $3, \ldots, 14-$ projections

	F	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
# correct predictions	2	6	6	6	7	7	7	7	7	7	7	7	7
# incorrect predictions	0	0	0	0	0	0	0	1	1	1	1	1	1
# unclassified substances	11	7	7	7	6	6	6	5	5	5	5	5	5

The average size of the initial graphs was 8 vertices and 7 edges in the training set and 13 vertices and 12 edges in the test set. As the projection size increases the classification accuracy first grows and then (for $k \geq 9$) starts to decrease (Table 6).

For another dataset with 25 molecular graphs in the training set, 17 graphs in the test set, and the same sizes of molecules as above a numerical value (characteristic of a specific activation energy of a molecule) [35] was supplied for each substance. This value was treated by means of the semilattice on intervals as described in Section 2. The resulting similarity semilattice is that on pairs of the form (graph set, numerical interval). The computation results are shown in Table 7.

Table 7. Predicting indirect carcinogenic potential in hydrocarbons with JSM-hypotheses, FCSS-encoding (**F**), and $3, \ldots, 14-$ projections

	F	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
# correct predictions	6	8	9	11	11	11	11	11	11	11	11	11	11
# incorrect predictions	5	2	2	1	1	1	1	1	1	1	1	1	1
# unclassified substances	6	7	6	5	5	5	5	5	5	5	5	5	5

Again, we computed k-projections of the initial molecular graphs for $1 \leq k \leq 13$. The stratified 10-fold cross-validation procedure was used to estimate different classification strategies in combination with k-projections ($1 \leq k \leq 13$). Table 8 shows the best results w.r.t. predictive accuracy and total number of predictions. On the one hand the best strategies w.r.t. predictive accuracy are C4.5 algorithm, the one based on JSM-hypotheses, and JRip rules. C4.5 attains predictive accuracy of 83% with FCSS representation, JSM and JRip attained predictive accuracy of 78% with k-projections. On the other hand, if we consider both precision and number of predictions, then the best result is obtained with JRip rules (78% of predictive accuracy and 93% of total number of predictions with k-projections). 66% of total number of predictions was attained with the use of C4.5 and FCSS representation. Table 8 also shows the results of 10-fold cross-validation for reduced tables. The use of reduced tables, as compared with nonreduced ones, does not make any difference for concept-based hypotheses, makes a very slight difference (no more than 5%) for C4.5 and small difference (about 10-15%) for other methods such as Naive Bayes and JRip.

Table 8. The results of 10-fold cross-validation for hydrocarbons dataset obtained with JSM-hypotheses (**J**), C4.5 (**C**), Naive Bayes classifier (**N**), and JRip rules (**R**) in combination with FCSS-encoding (**F**) and $3, \ldots, 14-$ projections (**PR**); **A** – predictive accuracy, **TP** – total number of predictions

	nonreduced table								**reduced table**							
	J-F	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**	**J-F**	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**
A	0.800	0.833	0.722	0.765	0.778	0.750	0.765	0.778	0.800	0.833	0.722	0.765	0.778	0.812	0.750	0.765
TP	0.533	0.667	0.867	0.867	0.467	0.800	0.867	0.933	0.533	0.667	0.867	0.867	0.467	0.867	0.800	0.867

The predictions made by different strategies were compared with known experimental results from [34] the following evaluations were obtained. As in the previous experiment with the k-projections and JSM-hypotheses where $1 \leq k \leq$

2 no classification was made at all. The strategies based on 3-, 4-projections and JSM-hypotheses result in better classifications than those with FCSS-encoding. The same result was attained by the strategy based on induction of decision trees (C4.5 algorithm). It results in 12 correct predictions and 5 incorrect predictions. For $k \geq 5$ in combination with JSM-hypotheses the best result was obtained w.r.t. predictive accuracy. Other methods with k-projections ($k \geq 5$) made approximately the same number of correct predictions as the strategy based on JSM-hypotheses. However, the number of incorrect predictions is also a little bit larger. Starting from a certain value of k the results stabilize and no further improvement is made. For example, with JSM-hypotheses and Naive Bayes, k is equal to 5, while for C4.5, $k = 11$. Table 9 shows the best results obtained with different methods.

Table 9. Predicting of indirect carcinogenic potential in hydrocarbons: the best results obtained with JSM-hypotheses (**J**), C4.5 (**C**), Naive Bayes classifier (**N**), and JRip rules (**R**) in combination with FCSS-encoding (**F**) and $3, \ldots, 14-$ projections (**PR**)

	J-F	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**
# correct predictions	6	12	8	8	11	15	13	12
# incorrect predictions	5	5	9	9	1	2	4	5

Thus, as for other datasets the best results were obtained for average projection values and were almost similar for reduced/nonreduced tables.

4.5 Analysis of Carcinogenicity of Polycyclic Aromatic Hydrocarbons

In the following experiment we considered data from [36]. The training dataset contains the descriptions of 25 molecular graphs of polycyclic aromatic hydrocarbons with indication of carcinogenic degree. As in Section 4.3 two separate computations were made for 2 target properties. To compare different classification methods in combination with k-cycles projections representation, we computed leave-one-out cross-validation. The best results w.r.t. predictive accuracy for the first target property are shown in Table 10. Learning with JSM-hypotheses attains the best results in most of the cases. However, other methods (e.g., C4.5) make more total predictions, see Table 10. Computer experiments with 25PAH data in comparison of FCSS, k-projections and reduced/nonreduced tables showed that the use of reduced tables, as compared with nonreduced ones, does not make any difference for any method.

To test the strength of methods we considered the test dataset from [37,38] and applied the hypotheses computed for k-cycles projections representation (with $1 \leq k \leq 7$) to classification of substances from the test set. There were 19 substances in the test dataset and Table 11 shows the best results obtained by different methods. From Table 11 we can conclude that among all methods w.r.t. predictive accuracy and completeness, the strategy based on JSM-hypotheses is

Table 10. The values of Leave-One-Out on 25PAH for the first target property, different methods, FCSS-encoding (**F**), and $3-,\ldots,7-$cycles projections

	J-F	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**
predictive accuracy	0.818	0.688	1.000	0.846	0.909	0.818	1.000	0.846
total number of predictions	0.643	0.786	0.500	0.786	0.714	0.643	0.500	0.786

Table 11. Predicting indirect carcinogenic potential in PAH: different learning methods, FCSS-encoding (**F**) and $3-,\ldots,7-$ cycles projections

	J-F	**C-F**	**N-F**	**R-F**	**J-PR**	**C-PR**	**N-PR**	**R-PR**
# correct predictions	6	5	13	1	7	7	11	7
# incorrect predictions	5	14	6	18	6	12	8	12

the best one for the second target property in combination with k-cycles projections representation (for all values $1 \leq k \leq 7$). For the first target property the best result was obtained by Naive Bayes, next comes the JSM-method. At the same time we consider the combination of two target properties to predict the carcinogenic degree of a substance from the test dataset. Thus the comparison between different methods was drawn w.r.t. both target properties. From Table 11 we can conclude that best results w.r.t. predictive accuracy were obtained with JSM-hypotheses for both FCSS codes and k-cycles projections.

As for practical complexity, cyclic projections were generated in less than 0.5 second for all values of k, parameter of projection, since each graph in this dataset contains no more than 7 cycles in the minimal cyclic base.

5 Conclusions

Definitions of graph similarity operations and its approximations (projections), based on order- and lattice-theoretic ideas, were considered and studied experimentally on several chemical datasets with several learning models. In many cases the proposed graph representation results in better predictive accuracy as compared to that with standard FCSS language for the analysis of biological activity of chemicals. We experimentally studied a technique for lowering dimensionality of datasets, called reduction of attributes. For JSM or concept-based learning the reduction of attributes is strictly information lossless. The reduction proved to be useful for decision tree induction, Naive Bayes classifiers, and JRip: while lowering the number of attributes in several times, it results in almost no loss of accuracy in case of decision tree induction and results in minor loss of accuracy in case of Naive Bayes and JRip classifiers. On the other hand, we studied the performance of learning methods with respect to precision of graph approximation controlled by projection level. With the increase of representation accuracy (k, parameter of projection), the performance of learning methods first improves, then stabilizes and in some cases becomes worse after a certain threshold, seemingly due to overfitting effects. This picture, standard for the role of dimensionality in machine learning, suggests the use of molecular graph

approximations instead of complete graphs: keeping dimensionality in a certain range, we can even gain in predictive accuracy. Further work on improving the representation model with labeled graphs will be related to accounting for 3D information, e.g. various types of isomerisms.

Acknowledgments. This work was supported by the Russian Foundation for Basic Research, project no. 05-01-00914a and by the Presidium of Russian Academy of Sciences, 2005 project "Problem solver for causal dependencies." The first author was partially supported by the Alexander-von-Humboldt Foundation.

References

1. King, R., Srinivasan, A., Dehaspe, L.: WARMR: A Data Mining tool for chemical data. J. of Computer-Aided Molecular Design **15** (2001) 173–181
2. Kramer, S.: Structural Regression Trees. In: Proc. 13th National Conference on Artificial Intelligence, AAAI-96, Cambridge/Menlo Park, AAAI Press/MIT Press (1996) 812–819
3. Kuznetsov, S.: Learning of Simple Conceptual Graphs from Positive and Negative Examples. In Zytkow, J., Rauch, J., eds.: Proc. Principles of Data Mining and Knowledge Discovery, Third European Conference, PKDD'99. Volume 1704 of Lecture Notes in Artificial Intelligence., Springer (1999) 384–392
4. Borgelt, C., Berthold, M.: Mining Molecular Fragments: Finding Relevant Substructures of Molecules. In Zhong, N., Yu, P., eds.: Proc. 2nd IEEE International Conference on Data Mining, ICDM'02, Piscataway, NJ, USA, IEEE Press (2002) 51–58
5. Inokuchi, A., Washio, T., Motoda, H.: Complete Mining of Frequent Patterns from Graphs: Mining Graph Data. Machine Learning **50** (2003) 321–354
6. Washio, T., Motoda, H.: State of the art of graph-based data mining. SIGKDD Explorations Newsletter **5** (2003) 59–68
7. Yan, X., Han, J.: gSpan: Graph-Based Substructure Pattern Mining. In: Proc. IEEE Int. Conf. on Data Mining, ICDM'02, IEEE Computer Society (2002) 721–724
8. Yan, X., Han, J.: CloseGraph: mining closed frequent graph patterns. In Getoor, L., Senator, T., Domingos, P., Faloutsos, C., eds.: Proc. of the 9th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, KDD'03, ACM Press (2003) 286–295
9. Gonzalez, J., Holder, L., Cook, D.: Experimental Comparison of Graph-Based Relational Concept Learning with Inductive Logic Programming System. In Matwin, S., Sammut, C., eds.: Proc. Inductive Logic Programming, ILP'2002. Volume 2583 of Lecture Notes in Artificial Intelligence., Springer (2003) 84–100
10. Blinova, V., Dobrynin, D., Finn, V., Kuznetsov, S., Pankratova, E.: Toxicology analysis by means of the JSM-method. Bioinformatics **19** (2003) 1201–1207
11. Pasquier, N., Bastide, Y., Taouil, R., Lakhal, L.: Efficient Mining of Association Rules Using Closed Itemset Lattices. J. Inf. Systems **24** (1999) 25–46
12. Avidon, V., Pomerantsev, A.: Structure-Activity Relationship Oriented Languages for Chemical Structure Representation. J. Chem. Inf. Comput. Sci. **22** (1982) 207–214

13. Helma, C., King, R., Kramer, S., Srinvasan, A., eds.: Proc. of the Workshop on Predictive Toxicology Challegnge at the 5th Conference on Data Mining and Knowledge Discovery, PKDD'01, http://www.predictive-toxicology.org/ptc/ (2001, September 7)
14. Pfahringer, B.: (The Futility of) Trying to Predict Carcinogenicity of Chemical Compounds. In Helma, C., King, R., Kramer, S., Srinvasan, A., eds.: Proc. of the Workshop on Predictive Toxicology Challegnge at the 5th Conference on Data Mining and Knowledge Discovery, PKDD'01, http://www.predictive-toxicology.org/ptc/ (2001, September 7)
15. Kuznetsov, S.: Similarity operation on hypergraphs as a basis of plausible inference. In: Proc. 1st Soviet Conference on Artificial Intelligence. (1988) 442–448
16. Kuznetsov, S.: JSM-method as a machine learning method. Itogi Nauki i Tekhniki, ser. Informatika **15** (1991) 17–50 in Russian.
17. Kuznetsov, S., Obiedkov, S.: Comparing performance of algorithms for generating concept lattices. J. Exp. Theor. Artif. Intell. **14** (2002) 189–216
18. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, Berlin, Heidelberg (1999)
19. Ganter, B., Kuznetsov, S.: Pattern Structures and Their Projections. In Stumme, G., Delugach, H., eds.: Proc. 9th Int. Conf. on Conceptual Structures, ICCS'01. Volume 2120 of Lecture Notes in Artificial Intelligence., Springer (2001) 129–142
20. Ganter, B., Grigoriev, P., Kuznetsov, S., Samokhin, M.: Concept-Based Data Mining with Scaled Labeled Graphs. In Delugach, H., Pfeiffer, H., Wolff, K., eds.: Proc. 12th Int. Conf. on Conceptual Structures, ICCS'04. Volume 3127 of Lecture Notes in Artificial Intelligence., Springer (2004) 94–108
21. Mitchell, T.: Machine Learning. The McGraw-Hill Companies (1997)
22. Finn, V.: Plausible Reasoning in Systems of JSM Type. Itogi Nauki i Tekhniki, Seriya Informatika **15** (1991) 54–101 in Russian.
23. Yan, L.S.: Study of carcinogenic mechanism of polycyclic aromatic hydrocarbons-extended bay region theory and its quantitative model. Carcinogenesis **6** (1985) 1–6
24. Birkhoff, G.: Lattice Theory. Amer. Math. Soc., Providence (1979)
25. Grigoriev, P.A., Yevtushenko, S.A.: Elements of an Agile Discovery Environment. In Grieser, G., Tanaka, Y., Yamamoto, A., eds.: Proc. 6th International Conference on Discovery Science, ICDS'03. Volume 2843 of Lecture Notes in Artificial Intelligence., Springer (2003) 309–316
26. Ganter, B., Kuznetsov, S.: Formalizing Hypotheses with Concepts. In Ganter, B., Mineau, G., eds.: Proc. 8th Int. Conf. on Conceptual Structures, ICCS'00. Volume 1867 of Lecture Notes in Artificial Intelligence., Springer (2000) 342–356
27. Witten, I., E.Frank: Data Mining: Practical Machine Learning Tools with Java Implementations. Morgan kaufmann, San Francisco (2000)
28. Cook, D., Holder, L.: Graph-Based Data Mining. IEEE Intelligent Systems **15** (2000) 32–41
29. Cameron-Jones, R., Quinlan, J.: Efficient Top-down Induction of Logic Programs. SIGART Bulletin **5** (1994) 33–42
30. Muggleton, S.: Inverse Entailment and Progol. New Generation Computing **13** (1995) 245–286
31. Gonzalez, J., Holder, L., Cook, D.: Application of Graph-Based Concept Learning to the Predictive Toxicology Domain. In Helma, C., King, R., Kramer, S., Srinvasan, A., eds.: Proc. Workshop on Predictive Toxicology Challegnge at the 5th Conference on Data Mining and Knowledge Discovery, PKDD'01, http://www.predictive-toxicology.org/ptc/ (2001, September 6)

32. Blinova, V.G., Dobrynin, D.A., Zholdakova, Z.I., Kharchevnikova, N.V.: Studies on the structure-activity relationships of alcohols by means of the JSM-method. Nauch. Tekh. Inf., ser. 2 (2001) 13–18 in Russian.
33. Guilian, W., Naibin, B.: Structure-activity relationships for rat and mouse LD50 of miscellaneous alcohols. Chemosphere **35** (1998) 1475–1483
34. Woo, Y.T., Lai, D., McLain, J., et al.: Use of mechanism-based structure-activity relationships analysis in carcinogenic ranking for drinking water desinfection by-products. Environ. Health Perspect. (2002) 75–87
35. Kharchevnikova, N.V., Blinova, V.G., Dobrynin, D.A., Maksin, M.V., Zholdakova, Z.I.: Application of JSM-method and quantum-chemical computations for predicting of carcinogenic potential and chronic toxicity in halogen-substituted aliphatic hydrocarbons. Nauch. Tekh. Inf., ser. 2 (2004) 21–28 in Russian.
36. Jerina, D., Lehr, R.: The bay-region theory: quantum mechanical approach to aromatic hydrocarbon-induced carcinogenecity. In: Microsomes and Drug Oxidation. Pergamon Press, Oxford (1977) 709–720
37. Dipple, A.: Polynuclear Aromatic Carcinogens. Number 172 in ACS Monograph. In: Chemical Carcinogens. Amer. Chem. Soc., Washington, DC (1976) 245–314
38. Lowe, J., Silverman, B.: Mo theory of ease of formation of carbocations derived from nonalternant polycyclic aromatic hydrocarbons. J. Amer. Chem. Soc. **106** (1984) 5955–5958