BIOINFORMATICS

Vol. 19 no. 10 2003, pages 1201–1207
DOI: 10.1093/bioinformatics/btg096

Toxicology analysis by means of the JSM-method

V. G. Blinova [1], D. A. Dobrynin [1], V. K. Finn [1], S. O. Kuznetsov [1, 2,] and E. S. Pankratova [1]*

[1]All-Russia Institute for Scientific and Technical Information (VINITI), Usievicha 20, 125219 Moscow, Russia and [2]Institut für Algebra, Technische Universität Dresden, Mommsenstr. 13, 01062 Dresden, Germany

Received on December 25, 2001; revised on November 4, 2002; accepted on November 6, 2002

ABSTRACT

Motivation: A model for learning potential causes of toxicity from positive and negative examples and predicting toxicity for the dataset used in the Predictive Toxicology Challenge (PTC) is presented. The learning model assumes that the causes of toxicity can be given as substructures common to positive examples that are not substructures of negative examples. This assumption results in the choice of a learning model, called the JSM-method, and a language for representing chemical compounds, called the Fragmentary Code of Substructure Superposition (FCSS). By means of the latter, chemical compounds are represented as sets of substructures which are 'biologically meaningful' from the expert point of view.

Results: The chosen learning model and representation language show comparatively good performance for the PTC dataset: for three sex/species groups the predictions were ROC optimal, for one group the prediction was nearly optimal. The predictions tend to be conservative (few predictions and almost no errors), which can be explained by the specific features of the learning model.

Availability: by request to finn@viniti.ru; serge@viniti.ru, http://ki-www2.intellektik.informatik.tu-darmstadt.de/~jsm/QDA

Contact: serge@viniti.ru

INTRODUCTION

Our approach for the analysis of the dataset used in the Predictive Toxicology Challenge is two-fold. The first component of the approach is a learning model called the JSM-method (in honor of John Stuart Mill, the English philosopher who proposed schemes of inductive reasoning in the 19th century). The JSM-method is based on Mill's idea that common effects are likely to be due to common causes. Originally the JSM-method was formulated in the early 1980s in terms of a predicate logic, which is an extension of the First-Order Predicate Logic with quantifiers over tuples of variable length (Finn, 1991). The fragment of the JSM-method that was used for the PTC (based on so-called *counterexample-forbidding hypotheses* (Finn, 1991), which will be called simply *hypotheses* in this paper) complies with the common paradigm of learning from positive and negative examples (Mitchell, 1997): given descriptions of positive and negative examples w.r.t. a goal attribute, positive hypotheses are 'generalized descriptions' of a subset of positive examples that do not 'cover' any negative example. Negative hypotheses are defined similarly.

The second component of the approach is the language for representing chemical structures. Toxicity of a chemical compound, as any other biological activity, depends on the character of weak bonds that arise between the compound and the biological receptor during their interaction. It is well-known that these bonds depend on π-electrons of the compound. This is why we used the so-called *fragmentary code of substructure superposition* (FCSS) as our descriptor language. FCSS was first proposed in (Avidon and Pomerantsev, 1982) and later developed within a coding software system (Leibov, 1991; Blinova and Dobrynin, 2000). By means of this language, a chemical compound is described by a set of substructures that are centers of localization of π-electrons. The description of a chemical compound by means of the FCSS language is often more relevant for the study of biological activities than the descriptions by structural formulae and/or by their simplifications.

LEARNING METHOD

Here, for the sake of simplicity, we present a fragment of the JSM-method (exactly the one used for the Predictive Toxicology Challenge) in terms of Formal Concept Analysis (FCA) (Ganter and Wille, 1999) in the way it was done in (Ganter and Kuznetsov, 2000). First, we recall some basic notions of FCA (Ganter and Wille, 1999).

A (*formal*) *context* is a triple of sets $K = (G, M, I)$, where G is called a set of objects, M is called a set of attributes, and $I \subseteq G \times M$ is a relation. For $g \in G$ and

*To whom correspondence should be addressed.

$m \in M$ *gIm* is interpreted as 'object g has attribute m'. For $A \subseteq G$ and $B \subseteq M$ *derivation operators* $(\cdot)'$ are defined as follows:

$$A' = \{m \in M | \forall g \in A(gIm)\};$$
$$B' = \{g \in G | \forall m \in B(gIm)\}.$$

A (*formal) concept* of a context (G, M, I) is a pair (A, B), where $A \subseteq G$, $B \subseteq M$, $A' = B$, and $B' = A$. The set A is called the (*formal*) *extent* and B the (*formal*) *intent* of the concept (A, B).

Let w be a *goal* attribute, different from attributes from the set M (the latter can be called *structural* attributes). For example, in the toxicology analysis w corresponds to toxicity and the structural attributes from M correspond to particular molecular substructures.

Input data for learning are given by sets of *positive*, *negative* and *undetermined examples* w.r.t. the goal attribute w. The undetermined examples are to be classified by means of the learned hypotheses. In terms of FCA, this situation can be described by three contexts: a *positive context* $K_+ = (G_+, M, I_+)$, a *negative context* $K_- = (G_-, M, I_-)$, and an *undetermined context* $K_\tau = (G_\tau, M, I_\tau)$. Here G_+, G_-, and G_τ are sets of positive, negative and undetermined examples, respectively; M is the set of *structural* attributes; $I_\varepsilon \subseteq G_\varepsilon \times M$, $\varepsilon \in \{+, -, \tau\}$ are relations that specify the structural attributes of positive, negative, and undetermined examples, respectively. The derivation operators in these three contexts are denoted by superscripts $+, -$, and τ, respectively. For example, the intents of a positive example g_+, negative example g_-, and undetermined example g_τ are denoted by g_+^+, g_-^- and g_τ^τ, respectively. Intents of contexts K_+, K_- and K_τ are called positive, negative and undetermined intents, respectively.

EXAMPLE 1. Consider positive, negative, and undetermined contexts given by the set of attributes $M = \{a, b, c, d, e\}$, the set of positive examples $G_+ = \{g_1, g_2, g_3, g_4\}$, the set of negative examples $G_- = \{g_5, g_6, g_7\}$, and the set of undetermined examples $G_\tau = \{g_8, g_9, g_{10}\}$. The relations I_+, I_-, I_τ are given by the intents of examples as follows:

$$g_1^+ = \{a, b, c\}, \quad g_5^- = \{a, c, d\}, \quad g_8^\tau = \{a, b, c, e\},$$
$$g_2^+ = \{a, b, d\}, \quad g_6^- = \{b, c, d\}, \quad g_9^\tau = \{c, d, e\},$$
$$g_3^+ = \{a, b, e\}, \quad g_7^- = \{a, d, e\}, \quad g_{10}^\tau = \{a, b, c, d\}.$$
$$g_4^+ = \{a, c, e\},$$

By the definitions, besides intents of examples, we have the following positive intents:

$$\{g_1, g_2, g_3\}^+ = \{g_1, g_2\}^+ = \{g_1, g_3\}^+ = \{g_2, g_3\}^+ = \{a, b\},$$
$$\{g_3, g_4\}^+ = \{a, e\}, \{g_1, g_4\}^+ = \{a, c\}, \{g_1, g_2, g_3, g_4\}^+ = \{g_2, g_4\}^+ = \{a\},$$

and the following negative intents:

$$\{g_5, g_6\}^- = \{c, d\}, \{g_5, g_7\}^- = \{a, d\}, \{g_5, g_6, g_7\}^- = \{g_6, g_7\}^- = \{d\}.$$

Now, a *positive hypothesis* (called a *counterexample forbidding hypothesis* by Finn (1991)) is defined in the following way. If intent h_+ of a concept of the positive context K_+ is not contained in the intent of any negative example (i.e. $\forall g_- \in G_- h_+ \not\subset g_-^-$) and $|h_+^+| \geqslant 2$ (there are at least two positive examples with intents containing h_+), then it is called a *positive hypothesis* w.r.t. the goal attribute w. *Negative hypotheses* are defined similarly: If intent h_- of a concept of the negative context K_- is not contained in the intent of any positive example (i.e. $\forall g_+ \in G_+ h_- \not\subset g_+^+$) and $|h_-^-| \geqslant 2$ (there are at least two negative examples with intents containing h_-), then it is called a *negative hypothesis* w.r.t. the goal attribute w.

In case of data given in Example 1, the intent $\{a, b\}$ is a positive hypothesis, whereas $\{a\}$, $\{a, c\}$, and $\{a, e\}$ are not, since, e.g. $\{a\} \subseteq \{a, c\} \subseteq g_5^- = \{a, c, d\}$ and $\{a, e\} \subseteq g_7^- = \{a, d, e\}$. The negative intent $\{c, d\}$ is a negative hypothesis, whereas $\{d\}$ and $\{a, d\}$ are not, since $\{d\} \subseteq \{a, d\} \subseteq g_2^+ = \{a, b, d\}$.

Hypotheses are used for the classification of undetermined examples from G_τ. If intent g_τ^τ of an undetermined example $g_\tau \in G_\tau$ contains a positive hypothesis h_+ (i.e. $g_\tau^\tau \supseteq h_+$), we say that h_+ is *for the positive classification of* g_τ. A *hypothesis for the negative classification of* g_τ is defined similarly: If intent g_τ^τ contains a negative hypothesis h_- (i.e. $g_\tau^\tau \supseteq h_-$), we say that h_- is *for the negative classification of* g_τ.

If there is a hypothesis for the positive classification of g_τ and no hypothesis for the negative classification of g_τ, then g_τ is *classified positively*. *Negative classifications* of g_τ are defined similarly, i.e. if there is a hypothesis for its negative classification and there is no hypothesis for its positive classification. If g_τ^τ does not contain any negative or positive hypothesis, then no classification is made (g_τ remains undetermined). If g_τ^τ contains both positive and negative hypotheses, then the classification is said to be *contradictory*.

In the case of data from Example 1, the classification of g_8 is positive, since $\{a, b\}$ is a positive hypothesis, $\{a, b\} \subseteq g_8^\tau = \{a, b, c, e\}$, and no negative hypothesis is a subset of g_8^τ. The classification of g_9 is negative, since $\{c, d\}$ is a negative hypothesis, $\{c, d\} \subseteq g_9^\tau = \{c, d, e\}$, and no positive hypothesis is a subset of g_9^τ. The classification of g_{10} is contradictory, since $\{a, b\} \subseteq g_{10}^\tau = \{a, b, c, d\}$ and $\{c, d\} \subseteq g_{10}^\tau$.

In the application of the JSM-method, a kind of cross-validation technique called *criterion of sufficient grounds*

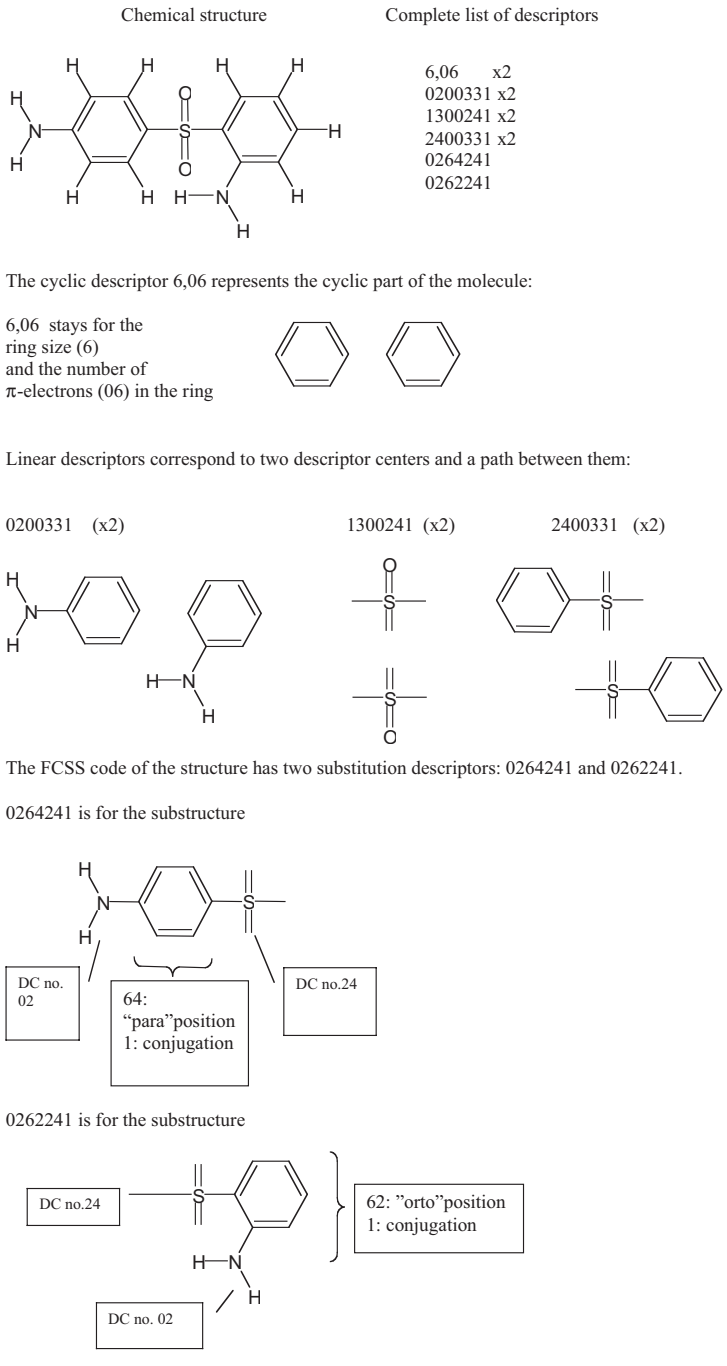


Fig. 1. An example of FCSS encoding of a chemical structure.

(CSG) is used. Suppose that we generated hypotheses according to the definitions above. Then, instead of applying them to undetermined examples, we try to reclassify original positive and negative examples. CSG is then the ratio of correctly classified original examples.

Note that the definitions above can be applied to more general data structures, e.g. for multisets (where the number of occurrences in a set are given for each element of the set) and sets of graphs (Kuznetsov, 1991). In Figure 1 the case of multisets is considered. Predictions for the PTC dataset were also carried out for the multiset representation.

As for computing hypotheses, the following can be said. By definition, a hypothesis is a special kind of formal

intent. Hence, positive hypotheses are naturally generated as intents satisfying the condition of not being contained in the intents of negative examples (similarly for negative hypotheses). In the worst case, testing this condition for one intent takes $O(|M| \cdot |G_-|)$ time. Several algorithms for computing formal intents are known, a review of them can be found in (Kuznetsov and Obiedkov, 2001). The algorithms mostly used in practice can compute the set of all positive intents Int_+ in time $O(|M|^2|G_+| \cdot |Int_+|)$. In the worst case the number of all intents (and hypotheses) can be exponential in the size of the underlying context, however for many real-life datasets the number is not very large. To reduce storage space, one can retain only hypotheses minimal w.r.t. inclusion (thus, any proper subset of a minimal hypothesis is not a hypothesis), since the set of all minimal hypotheses brings about the set of classifications equal to that brought about by the set of all hypotheses.

DESCRIPTOR LANGUAGE FOR REPRESENTING CHEMICAL COMPOUNDS

Toxicity of a chemical compound, as any other biological activity, depends on the character of weak bonds that arise between the compound and the biological receptor during their interaction. It is well-known that these bonds depend on π-electrons of the compound. That is why we used a descriptor language called the *fragmentary code of substructure superposition* (FCSS). The first version of this language was proposed in (Avidon and Pomerantsev, 1982), later on it was elaborated as a machine coding system (Leibov, 1991; Blinova and Dobrynin, 2000). By means of this language a chemical compound is described as a set of substructures that are centers of localization of π-electrons. The description of a chemical compound by means of FCSS language is often more relevant for the study of biological activities than the descriptions by structural formulae or their simplifications.

Now we give a short description of the FCSS language. First, active or *descriptor centers* (DC) are distinguished in a chemical compound. DCs are atoms or groups of atoms that can be centers of 'weak' interaction. They are atoms and groups of atoms that contain movable π- and d-electrons or a whole electrostatic charge, i.e. all heteroatoms (N, O, S, P, halloids, metals, etc.), carbon pairs connected with multiple (double, triple) bonds and aromatic cyclic systems as a whole. The list of descriptor centers of FCSS is given in Tables 1 and 2.

Some descriptor centers, e.g. 33, 34, 35, 36, 37, 40 and 41 are not mentioned in Tables 1 and 2 and are defined as follows. DC #33 corresponds to an arbitrary atom in an aromatic cycle that does not contain any heteroatoms; DC #34 corresponds to an arbitrary heteroatom in an aromatic cycle; DC #35 (#36, #37) correspond to a carbon atom in an aromatic cycle that is separated with one (two, three) bond(s) from a heteroatom; DC #40 corresponds to an arbitrary heteroatom in an aromatic cycle. DC #41 corresponds to carbon atoms that belong to cycles consisting entirely from carbon atoms in the state of SP3-hybridization. Definitions of some other descriptor centers can be found in (Blinova and Dobrynin, 2000).

Table 1. List of FCSS Descriptor Centers of the first type

Atom	Valences	DC Number	Atom	Valences	DC Number
Li	1	43	Ga	3	43
Be	1	43	Ge	4	43
B	3	53	As	3,5	51
N–	2	00	As+	4	51
O–	1	15	Se	2,4,6	54
O+	3	16	Br	1	31
F	1	32	Br	1	48
Na	1	43	Rb	1	43
Mg	2	43	Sr	2	43
Al	3	43	Y	3	43
Si	2,4	52	Zr	4	43
P	3,4,5	47	Nb	2,5	43
P+	4	47	Mo	2,4,6	43
S	6	23	Ag	1,2	43
S+	3,4	23	Cd	2	43
Cl	1	31	Sn	2,4	43
K	1	43	Sb	3,5	51
Ca	2	43	Te	2,4,6	54
Sc	3	43	I	1	31
Ti	4	43	I	1	49
V	2,3,4,5	43	Ba	2	43
Cr	2,3,4,6	43	Pt	2	43
Mn	2,4,7	43	Au	1,2	43
Fe	2,3	43	Hg	1,2	43
Co	2	43	Ti	3	43
Ni	2	43	Pb	2,4	43
Cu	1,2	43	Bi	2,3,5	43
Zn	2	43			

Table 2. List of FCSS Descriptor Centers of the second type (Z denotes any atom, R denotes any atom except for H)

DC	Valence	Code	DC	Valence	Code	DC	Valence	Code
	4	01	O=R	2	13	R–OH	2	11
	4	07	Z–SH	2	21	R–O–R	2	12
Z–NH–Z	3	02	R–S–R	2	22		6	24
	3	03		2	14	$Z–CH_3$	4	41
R=NH	3	04	S=R	2	25	R ≡ CH	4	41,80
R=N-R	3	05	$R=CH_2$	4	41, 80	R ≡ N	3	06

Table 3. The form of a linear descriptor

Descriptor Center 1	Chain length (in carbon atoms)	Descriptor Center 2	Conjugation attribute (binary)

Table 4. The form of a cyclic descriptor

Head	Body	Tail
Geometric form of the cyclic system	The number of π-electrons in the conjugated system	Location of heteroatoms

The elements of FCSS descriptor language fall into three classes: linear, cyclic, and substitution descriptors.

A linear descriptor is given by a pair of DCs connected by a chain of -CH_2- groups. For these chains it is also essential whether there is a conjugation (common d- and π-electrons) between C-atoms of these chains. The form of a linear FCSS descriptor is described in Table 3. A linear descriptor is given by seven digits: two digits for each descriptor center (first goes a DC with the smaller number) and for the chain length. Conjugation attribute is given by a single bit (1 if there is conjugation in the chain and 0 otherwise).

The form of cyclic descriptors is described in Table 4. The *Head* component gives the size of the cycle (the number of atoms) in the case of a monocycle and the size and location of separate simple cycles in the case of polycyclic systems.

Substitution descriptors encode the mutual location of substituents in aromatic systems, e.g. they allow to distinguish between 'ortho'- and 'para'-positions of substitutors. The first two digits, as well as the fifth and the sixth digits, of a descriptor of this type correspond to the beginning and the end of the chain, the third and the fourth digits denote the mutual location of descriptor centers and the seventh (binary) digit corresponds to the presence/absence of resonance interaction (conjugation) between ultimate elements of the chain, i.e. by the corresponding descriptor centers. The code of a chain between descriptor centers A and B is given in accordance with Table 5.

Since the number of all possible descriptors can be very large, we generate for each dataset only those descriptors that arise from molecular structures in the dataset. This preprocessing is done by a special algorithm that traverses the molecular graph and constructs the descriptors.

In Figure 1 we give an example of FCSS encoding of a chemical structure. Here the multiplicity of attribute occurrences is taken into account (x2 stays for double occurrence).

Table 5. Encoding of substitutor locations

Location	Chain code	Conjugation
('ortho')	62	1
('meta')	63	0
('para')	64	1
('peri')	65	0
('amphi')	66	1
	67	1
	61	1

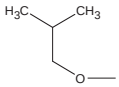
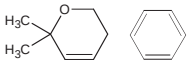
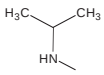
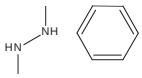
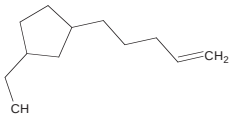
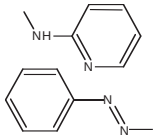
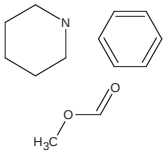
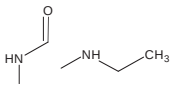
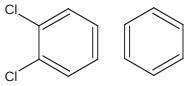
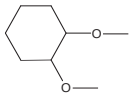
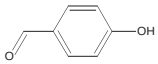
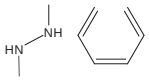
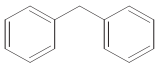
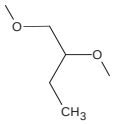
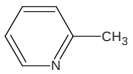
PREDICTIONS FOR TOXICOLOGY DATA SET

Predictions for the PTC dataset were made according to the classification model from Section 2 with the use of FCCS descriptors as structural attributes (set *M*). 9036 FCSS descriptors were generated to encode examples from the training set and undetermined examples provided by the PTC organizers (it took about 8s for Pentium III 866 MHz computer). For each of the four sex/species groups the goal attribute was toxicity. Positive and negative classifications were considered as the corresponding predictions for toxicity. Contradictory classifications were considered as ambiguous and thus ignored.

In Tables 6 and 7 we present some hypotheses generated by the JSM-system and used for correct classifications. As for positive hypotheses, only thirteen of them were used for positive correct classifications of compounds from the test set. Some of these hypotheses are common for several sex/species groups. We show them in Table 6, where the first column gives a molecular graph and the second column gives its FCSS descriptors, the third column gives the number of correct predictions of toxicity for particular sex/species groups (F and M stand for female and male, respectively, R and M stand for rats and mice, respectively). For example, 1FR 3MR means that a positive hypothesis was used for correct predictions of toxicity of one substance for female rats and three substances for male rats.

Far more (around 100) negative hypotheses used for classifications were generated for each sex/species group. Due to space limitations, in Table 7 we only present those

Table 6. Positive hypotheses

Molecular graph	FCSS Descriptors	No. of correct predictions in sex/species group(s)	Molecular graph	FCSS Descriptors	No. of correct predictions in sex/species group(s)
	1203410 1203410	1 FR		6,06 3501411 3501411	1 MR
	0202410 0202410	1 FR 3 MR		3102410	1 FM
	6,06 0200021	2 FR		5,00 4108460	1 FM 1 MM
	6,06 0500051 0500331 6,06M1 0200351	1 FR 1 MR 1 FM		1201411 1201131 6,06 6,00M1	1 FM 1 MM
	0201131 0202410	1 FR 1 MM		6,06 6,06 3162311 3100331 3100331	1 MM
	1204411 1204411 6,00	1 MR		1301331 1200331 6,06	2 MR
	6,06 0200021	2 FM 2 MR		6,06 6,06 3301331	1 MR
	1203410 1204410	1 MR		3501411 6,06M1	1 MR

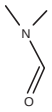
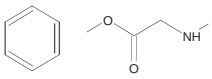
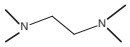
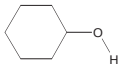
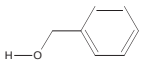
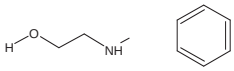
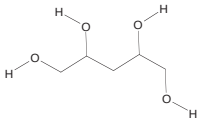
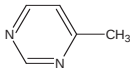
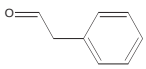
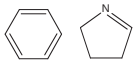
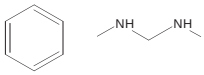
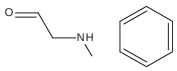
negative hypotheses that were used for the greatest number of correct negative predictions (i.e. predictions of non-toxicity).

As for cross-validation, we computed the value of the criterion of sufficient grounds defined in the first section. For the training sample the value of CSG is 42%, for the union of the training and test samples, the value is 91%. These results show that the training sample was fairly heterogeneous.

CONCLUSIONS

The evaluation of results of the ROC analysis shows that the predictions made were among optimal predictions for three sex/species groups and almost optimal for the fourth group (note also that we did not make use of the possibility to submit multiple sets of predictions). The ROC analysis demonstrates the conservatism of predictions made by the JSM-method: it produces few predictions, but commits almost no errors. Besides the relevance of the FCSS language to the problem solved, the latter fact can be explained by two features of the learning model: first, we strictly forbid counterexamples of a hypothesis (i.e. a positive hypothesis does not cover any negative example). Second, a hypothesis itself, as a representation of the common properties of positive examples, is their *least general* (and thus, 'most cautious') generalization. To allow for less cautious predictions we can change the definition of a hypothesis by permitting a certain number of counterexamples (i.e. negative examples for positive hypotheses and positive examples for negative hypotheses) to be contained in a given hypothesis. One might also change the definition of a prediction by making it

Table 7. Negative hypotheses

Molecular graph	FCSS Descriptors	No. of correct predictions in sex/species group(s)	Molecular graph	FCSS Descriptors	No. of correct predictions in sex/species group(s)
	0301131	13 FM 15 MM		1101131 6,06 0202130	7 FM
	0302030	7 FR 4 MR 5 FM 6 MM		1104411	3 FR 4 MR 6 FM 3 MM
	1101330 6,06	7 FR		6,06 0202110	19 FM 19 MM
	1105110 1104110	3 FM 3 MR 3 MM		3501411	6 FM
	1302330 6,06	9 MR		6,06 5,00M1	24 MM
	6,06 0201021	4 FR 3 MR 5 FM 4 MM		6,06 0202130	1 MR 9 FM 10 MM

asymmetric with respect to positive and negative hypotheses, e.g. by allowing for a certain rate of negative hypotheses that would not violate the positive classification. As for the method of representing molecular structures, the work on introducing 3D-attributes into the FCSS language is now in progress.

ACKNOWLEDGEMENTS

This work was supported by the 'Computer Intelligent System' Program of the Russian Academy of Sciences, project no. 2.2. The fourth author was also supported by the Alexander von Humboldt Foundation.

REFERENCES

Avidon,V.V. and Pomerantsev,A.B. (1982) Structure-activity relationship oriented languages for chemical structure representation. *J. Chem. Inf. Comput. Sci.*, **22**, 207–214.

Blinova,V. and Dobrynin,D. (2000) Languages for Representing Chemical Conpounds for Intelligent Systems of Chemical Design. *Automatic Documentation and Mathematical Linguistics*, **3**.

Finn,V.K. (1991) Plausible Reasoning in Systems of JSM Type. *Itogi Nauki i Tekhniki, Ser. Informatika*, **15**, 54–101. (in Russian).

Ganter,B. and Wille,R. (1999) *Formal Concept Analysis. Mathematical Foundations*. Springer.

Ganter,B. and Kuznetsov,S.O. (2000) Formalizing Hypotheses with Concepts. In Ganter,B. and Mineau,G. (eds), *Proc. of the 8th International Conference on Conceptual Structures (ICCS 2000)*, Lecture Notes in Artificial Intelligence, 1867, pp. 342–356.

Leibov,A.E. (1991) Automatic Coding of Chemical Structures by means of FCSS codes. *Itogi Nauki i Tekhniki, ser. Informatika*, **15**, 141–158. (in Russian).

Kuznetsov,S.O. (1991) JSM-Method as a Machine Learning Method. *Itogi Nauki i Tekhniki, ser. Informatika*, **15**, 17–53. (in Russian).

Kuznetsov,S.O. and Obiedkov,S.A. (2001) Algorithms for the Construction of Concept Lattices and Their Diagram Graphs. In de Raedt,L. (ed.), *Proceedings of the 5th Conference on Principles of Data Mining and Knowledge Discovery (PKDD'2001)*, Lecture Notes in Artificial Intelligence, 2168, pp. 289–300.

Mitchell,T. (1997) *Machine Learning*, McGraw Hill.