

# SMIREP: Predicting Chemical Activity from SMILES

Andreas Karwath\* and Luc De Raedt

Institut für Informatik, Albert-Ludwigs Universität Freiburg, Georges-Köhler-Allee 079,  
D-79110 Freiburg, Germany

Received May 2, 2006

Most approaches to structure–activity-relationship (SAR) prediction proceed in two steps. In the first step, a typically large set of fingerprints, or fragments of interest, is constructed (either by hand or by some recent data mining techniques). In the second step, machine learning techniques are applied to obtain a predictive model. The result is often not only a highly accurate but also hard to interpret model. In this paper, we demonstrate the capabilities of a novel SAR algorithm, SMIREP, which tightly integrates the fragment and model generation steps and which yields simple models in the form of a small set of IF-THEN rules. These rules contain SMILES fragments, which are easy to understand to the computational chemist. SMIREP combines ideas from the well-known IREP rule learner with a novel fragmentation algorithm for SMILES strings. SMIREP has been evaluated on three problems: the prediction of binding activities for the estrogen receptor (Environmental Protection Agency's (EPA's) Distributed Structure-Searchable Toxicity (DSSTox) National Center for Toxicological Research estrogen receptor (NCTRER) Database), the prediction of mutagenicity using the carcinogenic potency database (CPDB), and the prediction of biodegradability on a subset of the *Environmental Fate Database* (EFDB). In these applications, SMIREP has the advantage of producing easily interpretable rules while having predictive accuracies that are comparable to those of alternative state-of-the-art techniques.

## 1. INTRODUCTION

In the past few decades, a number of computational methods to predict structure–activity relationships (SAR) or quantitative structure–activity relationships (QSAR) based on 2D or 3D models of molecules have been proposed for fast high-throughput screening. Most of these approaches assume that the relevant fragments, biophores, or fingerprints are provided by an expert or are calculated a priori and then induce a predictive model employing these. The commonly used MDL key sets<sup>1,2</sup> can be seen as an example of these predefined fragments or fingerprint approach. It is also possible to employ a number of other structural, topological, or physiochemical descriptors calculated by specialized software such as Molconn-Z<sup>3</sup> and use them for high-throughput screening of larger databases. However, as the generation of the relevant structural alerts or fragments is a nontrivial task, which greatly determines the quality of the learned model, several recent approaches from the field of data mining try to automate this generation process.

First, a number of graph-mining methods have been employed to SAR problems in order to discover the necessary relevant fragments. The vast majority of these approaches computes fragments (sometimes called local patterns) that frequently occur in or are significant with respect to a given data set, cf. Dehaspe,<sup>4</sup> Deshpande et al.,<sup>5</sup> Kramer et al.,<sup>6,7</sup> Zaki,<sup>8</sup> Yan and Han,<sup>9</sup> Borgelt and Berthold,<sup>10</sup> Inokuchi et al.,<sup>11,12</sup> and Kuramochi and Karypis.<sup>13</sup> The earliest approaches<sup>4</sup> to compute such fragments are based on techniques from inductive logic programming (ILP).<sup>14</sup> Whereas ILP

techniques are theoretically appealing because of the use of expressive representation languages, they exhibit significant efficiency problems, which in turn implies that their application has been restricted to finding relatively small fragments in relatively small databases. Recently proposed approaches to mining frequent fragments in graphs such as gSpan,<sup>9</sup> CloseGraph,<sup>15</sup> FSG,<sup>5</sup> MoFa,<sup>10</sup> Gaston,<sup>16</sup> and AGM<sup>12</sup> are able to mine complex subgraphs more efficiently. However, the key difficulty with the application of these techniques is—as for other frequent pattern mining approaches—the number of patterns that are generated. For instance, Inokuchi et al.<sup>11</sup> report on the order of  $10^6$  patterns being discovered. Furthermore, frequent fragments are not necessarily of interest to a molecular scientist. Therefore, Kramer et al.<sup>6</sup> and Inokuchi and Kashima<sup>11</sup> take into account the classes of the molecules. Kramer et al. compute all simple patterns that are frequent in the actives and infrequent in the inactives, whereas Inokuchi et al. compute correlated patterns.

Second, there exist a few approaches that integrate the discovery of the fragments with the learning of the predictive model, most notably the CASE/MULTICASE family<sup>17,18</sup> and the more recent LAZAR system by Helma.<sup>19</sup> For instance, the well-known MULTICASE system constructs fixed sized fragments from the compounds and then uses a divide-and-conquer strategy (based on statistical tests) to distinguish between major biophores for classification and modulators that can regulate activity of a primary biophore as well as for biophobes indicating inactivity. The generated biophores or fragments are in principle linear fragments (though MULTICASE also supports branches around the backbone) and, hence, do not necessarily capture more complex

\* Corresponding author phone: +49 761 203 8029; e-mail: karwath@informatik.uni-freiburg.de.

structures of chemical compounds. Furthermore, it is hard to find detailed information about the way the fragments are generated. The recent LAZAR approach by Helma<sup>19</sup> identifies linear fragments present in a compound database, identifies the relevant ones (using a statistical test), removes redundant ones, and predicts activity or inactivity for a given test compound based on majority vote. Similar to MULTICASE, the employed fragments are linear and include more complex structures, such as rings, only indirectly within their predictions. Furthermore, MULTICASE and LAZAR employ a weighting/scoring scheme on the fragments to make predictions, which are not always easy to understand or interpret.

The approach employed in SMIREP<sup>20</sup> is different. SMIREP combines the chemical modeling language SMILES (Simplified Molecular Input Line Entry System,<sup>21</sup> with IREP (Incremental Reduced Error Pruning), a state-of-the-art machine learning algorithm that produces a predictive model in the form of a small set of IF-THEN rules. It is essentially a specialized learning system for SAR and QSAR applications and for fast extraction of relevant structural fingerprints or features. In SMIREP, each IF-THEN rule lists one or more fragments that must be present in order for a compound to be active and, hence, describes directly a structural alert that is easy to interpret. The generation of the fragments is performed directly on the SMILES representations of the compounds and is guided by heuristics from the well-known rule-learner IREP.<sup>22,23</sup> We have applied SMIREP to three SAR problems: the prediction of binding activities for the estrogen receptor (EPA's DSSTox NCTRER Database), the prediction of mutagenicity using the carcinogenic potency database (CPDB), and the prediction of biodegradability on a subset of the Environmental Fate Database (EFDB). The experiments show that SMIREP produces *small* rule sets containing possibly *complex* fragments, that SMIREP is competitive in terms of predictive accuracy, and that SMIREP is quite efficient as compared to alternative methods.

## 2. METHODS

**2.1. Databases.** **2.1.1. DSSTox NCTRER.** The estrogen database was extracted from the EPA's DSSTox NCTRER Database ([http://www.epa.gov/nheerl/dsstox/sdf\\_nctrer.html](http://www.epa.gov/nheerl/dsstox/sdf_nctrer.html)). The original data set was published by Fang et al.<sup>24</sup> and is specially designed to evaluate QSAR approaches. The NCTRER database provides activity classifications for a total of 232 chemical compounds, which have been tested regarding their binding activities for the estrogen receptor. The database contains a diverse set of natural, synthetic, and environmental estrogens and is considered to cover most known estrogenic classes spanning a wide range of biological activity.<sup>24</sup>

The database distributed by the EPA's DSSTox is in SDF (Structure Data Format) and contains, in addition to the original database, a number of annotations: 6 indicator variables extracted from the original publication,<sup>24</sup> logP (octanol/water partition coefficient) values, and chemical class assignments (6 main classes, 20 subclasses) as well as the activity category ER-RBA (estrogen receptor relative binding affinity). This classification yields 131 active and 101 inactive compounds (with regard to their ER-RBA).

**2.1.2. CPDB.** The original carcinogenic potency database (CPDB: <http://potency.berkeley.edu/cpdb.html>) provides carcinogenic as well as mutagenic classifications as determined by the *Salmonella*/microassay for a number of chemical compounds mainly of industrial and pharmaceutical interest. The database employed here was published by Helma et al.,<sup>7</sup> filtered to eliminate mixtures and undefined structures, and annotated with SMILES strings for each compound. The filtered database was downloaded from [http://www.predictive-toxicology.org/data/cpdb\\_mutagens/](http://www.predictive-toxicology.org/data/cpdb_mutagens/).

Overall, the database contains 684 chemical structures (341 mutagens and 343 non-mutagens). Each entry is annotated with a variety of precalculated numerical attributes as well as other relevant information such as logP, homo, lumo, electronegativity, and other numerical properties.

**2.1.3. EFDB.** This database originates from a study about biodegradability of a number of commercially available chemical compounds. The data set was first published by Howard et al.<sup>25</sup> and has been used to evaluate the prediction capabilities of a number of relational classifications methods,<sup>26,27</sup> where a subset of 328 chemicals was used. We have selected this data set to be able to compare SMIREP's performance to some other state-of-the-art approaches from the machine learning and data-mining community. The data sets main source is the *Syracuse Research Cooperation's* (SRC) *Environmental Fate Database* (EFDB). The database contains degradation rates (in form of half-life times) for chemicals, considering *biotic*, *abiotic*, and *all* degradation within four environmental situations (soil, air, surface water, and groundwater). Furthermore, these degradation rates are measured within three environmental conditions *aerobic*, *anaerobic*, and *removal in wastewater treatment plants*. To be able to compare our approach to previously published work,<sup>26,27</sup> we restrict ourselves to the aqueous biodegradation in aerobic conditions. We use the same procedure of dividing the chemicals into degradable and nondegradable as Blockeel et al.<sup>27</sup> That is, compounds considered to degrade are compounds possessing half-life times of up to 4 weeks, or they are considered nondegradable otherwise. In addition to the 2D structure of the chemicals, global attributes are available like logP and the compound's molecular weight.

**2.2. SMILES and SMARTS.** **2.2.1. SMILES.** SMILES<sup>21</sup> is a well-known linear string representation language for chemical molecules. The SMILES language is commonly used in computational chemistry and is supported by the major software tools in the field, such as the commercial Daylight toolkit and the Open-Source OpenBabel library.

The SMILES notation of chemical compounds is comprised of atoms, bonds, parentheses, and numbers. Atoms are represented by their atomic symbols. The four basic bond types are represented by the symbols '-', '=', '#', and ':'. Ionic bonds, or *disconnections*, are represented by a '.'. Branches are specified by enclosing brackets, "(" and ")". Cyclic structures are represented by breaking one bond in each ring. The atoms adjacent to the bond obtain the same number. Here, we refer to these numbers as *cyclic link numbers*. The cyclic link numbers are not necessarily unique within a SMILES representation of a molecule.

To search for subgraphs in compounds encoded in SMILES, one can use the SMARTS language.<sup>28</sup> While SMILES is a language representing molecules, SMARTS is a language representing SMILES fragments. Although

205 SMARTS allows the use of wildcards and more complicated  
206 constraints, SMIREP uses only the SMILES subset of the  
207 SMARTS pattern language, that is, we use the SMILES  
208 notation for fragments.

209 **2.2.2. Chirality.** As the SMIREP approach presented in  
210 this work is very much database driven, we have examined  
211 the databases used for occurrences of stereoisomers with  
212 different activities. We have done this by comparing the main  
213 layer of the InChI codes<sup>29</sup> of all molecules (generated  
214 with the InChI generation tool downloaded from <http://www.iupac.org>). This allows one to detect molecules having  
215 the same skeletons and atomic composition as well as  
216 stereoisomers.  
217

218 In the DSSTox NCTRER data set, only five compounds  
219 have a R/S complement, and only one compound has  
220 stereoisomers with different activities. Similarly, in the CPDB  
221 data set, we found only one pair of compounds possessing  
222 the same skeleton and atomic composition but having  
223 assigned different activities. In the EFDB we found no  
224 compounds possessing the same skeleton while being clas-  
225 sified in different categories. This information is insufficient  
226 for discovering chirality dependent rules. Therefore, we have  
227 chosen to explicitly disregard chirality information in all  
228 compounds in the three databases.

229 **2.3. SMIREP. 2.3.1. Setting.** SMIREP<sup>20</sup> aims at automati-  
230 cally discovering fragments, alerts, or biophores that dis-  
231 criminate the active compounds from the inactive ones. The  
232 discovered fragments are incorporated into IF-THEN rules,  
233 which essentially test whether a set of fragments is all  
234 present. When *all* fragments stated in the IF part of a rule  
235 are present in a compound, we also say that the rule *covers*  
236 the compound. For instance, consider the following two rules:

```
IF a compound contains the fragments:
    'cccc' AND 'ccN' AND 'ccO' AND 'OC'
THEN the compound is active

IF a compound contains the substructures:
    'c1ccccc1' AND 'Nccc'
THEN the compound is active
```

237 These rules are conjunctive and contain SMARTS patterns  
238 as their conditions. Furthermore, together they constitute to  
239 a rule-set, which is a predictive model that is used for  
240 classifying compounds as follows: if there is a rule that  
241 covers the compound, then predict "active"; otherwise predict  
242 "inactive". The rules are evaluated in SMIREP using the  
243 OpenBabel toolkit ([www.openbabel.org](http://www.openbabel.org)) and are also easy  
244 to interpret as one classifies on the basis of the presence (or  
245 absence) of certain fragments. An actual rule set computed  
246 by SMIREP can be found in Table 3.

247 The problem tackled by SMIREP can now be formulated  
248 as follows:

249 **Given:** a set of compounds in SMILES format, where  
250 each compound is classified as either *active* or *inactive*

251 **Find:** a rule-set that accurately discriminates *active* from  
252 *inactive* compounds.

253 As the discovered rule set should be used for classification,  
254 SMIREP searches for rules that satisfy many of the actives

**Table 1.** Confusion Matrix with Four Possible Outcomes: TP, TN, FP, and FN<sup>a</sup>

	predicted	
	active	inactive
active	TP	FP
inactive	FN	TN

<sup>a</sup> TP denotes the number of true positives, and TN denotes the number of true negatives. The number of errors made by predicting a compound of being active while it is not is denoted by FP (false positives), while predicting a compound to be inactive while it is active is denoted by FN (false negatives).

**Table 2.** Accuracy and Area Under ROC Curve for the NCTRER ER-Binding Data Set for the Different Settings and Beamsizes (*k*) from the 10 × 10-Fold Cross-Validations<sup>a</sup>

setting	<i>k</i>	acc training	AUC training	acc testing	AUC testing
SAR	5	80.34 (0.41)	0.832 (0.021)	78.96 (1.13)	0.816 (0.087)
	10	80.13 (0.51)	0.830 (0.021)	78.49 (1.92)	0.810 (0.077)
	20	80.05 (0.29)	0.826 (0.018)	77.62 (1.56)	0.800 (0.098)
QSAR	5	80.68 (0.63)	0.833 (0.024)	76.98 (1.41)	0.795 (0.090)
	10	80.68 (0.47)	0.831 (0.026)	77.69 (2.33)	0.802 (0.098)
	20	80.83 (0.45)	0.834 (0.022)	78.17 (1.90)	0.806 (0.074)

<sup>a</sup> The numbers in brackets denote the standard deviation. Surprisingly, the predictive performance drops slightly when numerical attributes are used in this experiment. A similar effect has also been reported by Helma et al.<sup>7</sup> on a different data set. Although, we have used a bin size of five to avoid overfitting, it still seems that in this experiment SMIREP does overfit slightly. An indicator for this is the difference in the training accuracy when compared to the testing accuracy. The training accuracy in the QSAR setting is always higher than the one for the SAR setting. However, the testing accuracy shows a higher drop as in the SAR setting.

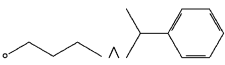
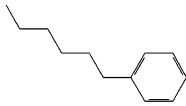
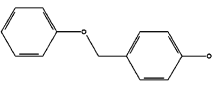
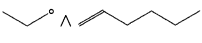
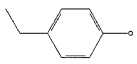
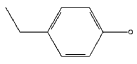
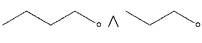
255 and few of the inactive ones. The task of finding rule sets  
256 for classification has been well-studied in the field of machine  
257 learning.<sup>22,30,31</sup> The key difference with traditional rule-set  
258 learning problems in machine learning lies in the use of the  
259 SMILES and SMARTS languages for representing com-  
260 pounds and patterns. SMIREP embraces several ideas from  
261 a well-known rule-learner from the field of machine learning,  
262 IREP,<sup>22,23</sup> but was adapted for the use of SMILES and  
263 SMARTS as representation languages.

264 In a previous preliminary publication,<sup>20</sup> we have introduced  
265 SMIREP in a computer science context demonstrating  
266 SMIREP's ability to tackle activity predictions within large  
267 databases of chemical compounds. The work presented here  
268 concentrates on the applicability and performance of SMIREP  
269 within a number of chemical applications.

270 **2.3.2. Overview.** SMIREP follows essentially a separate-  
271 and-conquer approach,<sup>32</sup> in which one iteratively searches  
272 for a single rule that covers many of the active compounds  
273 and none (or only very few) of the inactive ones. Once such  
274 a rule is found, it is added to the rule set, and the actives  
275 covered by the found rule are deleted. This process is then  
276 repeated until further rules do not yield any improvement  
277 with regard to a scoring function or all actives have been  
278 covered. The main SMIREP algorithm is depicted in  
279 Algorithm 1 (see Chart 1).

280 In order to search for one rule, SMIREP employs a so-  
281 called *seed compound*. The SMILES representation of the  
282 seed compound is decomposed in a *fragment tree*, which  
283 then determines the possible steps taken through the search  
284 space by the refinement operators (see section 2.3.4). To

**Table 3.** Example Rule Set Induced by SMIREP on the NCTRE ER-Binding Database<sup>a</sup>

No	Rules	Description
1	<chem>ccccO ^ C(c1ccccc1)(C)C</chem> 	This rule seems to be related to the Fang <i>et al.</i> rule set by enforcing the presence of an aromatic ring structure, as well as the presence of an oxygen attached to an aromatic ring structure (first part of the rule). Overall this rule also implies a certain hydrophobicity due to the large amount of aromatic bonds.
2	<chem>CCCCCc1ccccc1</chem> 	This rule seems to imply a certain size and hydrophobicity of the molecule but no precise information about the required H-bonding capabilities. In fact this rule covers only a fraction of features which active ER-binding substances should possess.
3	<chem>c1c(occ2ccc(cc2)O)cccc1</chem> 	The rule depicts a 2D chemical structure similar to the DES skeleton used in Hong <i>et al.</i> , <sup>39</sup> and describes an aromatic ring connected to a phenolic ring structure by two atoms, oxygen and carbon via aromatic bonds. This is not exactly the DES skeleton, as firstly the bonds are not variable and secondly not both atoms are carbons.
4	<chem>Occ ^ C=Ccccc</chem> 	This rule does not correspond to any rule in the Fang <i>et al.</i> rules. The two facts correspond to some particular parts of those rules: the <i>Occ</i> fragment implies the presence of at least one of two required H-bonding sites, while the <i>C=Ccccc</i> fragment implies some larger aromatic structures, especially in combination with the first fragment.
5	<chem>ccc1ccc(cc1)O</chem> 	Rule 5 as well as rules 6 capture the existence of an phenolic ring structure within a compound, that matches some part of the rule system by Fang <i>et al.</i>
6	<chem>ccc1ccc(cc1)O</chem> 	
7	<chem>Occcc ^ CCCO</chem> 	This rule is similar to rule 4. In addition to the two features noted above, this rule implies two OH groups instead of one, separated by a relatively long chain - or even one or more rings, partly aromatic and partly non-aromatic.

<sup>a</sup> The description compares these rules to the rule set published by Fang *et al.*<sup>24</sup>



**Chart 1.** Algorithm 1: SMIREP**Algorithm 1** SMIREP

```

1: /* INPUT: Databases Act and InAct in SMILES */
2: /* OUTPUT: A set of rules for Act */
3: Rule Set := {}
4: while Act ≠ {} do
5:   split (Act,InAct) into (GrowPos, GrowNeg, PrunePos, PruneNeg)
6:   select randomly k seeds ∈ GrowPos
7:   PrunedRules := {}
8:   for all seed in seeds do
9:     GrownRule := GROW(seed, GrowPos, GrowNeg)
10:    PrunedRule := PRUNE(GrownRules, PrunePos, PruneNeg)
11:    PrunedRules = PrunedRules ∪ PrunedRule
12:  end for
13:  select BestRule in PrunedRules by score
14:  if error rate of BestRule on (PrunePos, PruneNeg) > 50% then
15:    Rule Set := Rule Set ∪ {Rule}
16:    remove examples covered by Rule from (Pos, Neg)
17:  else
18:    return Rule Set
19:  end if
20: end while

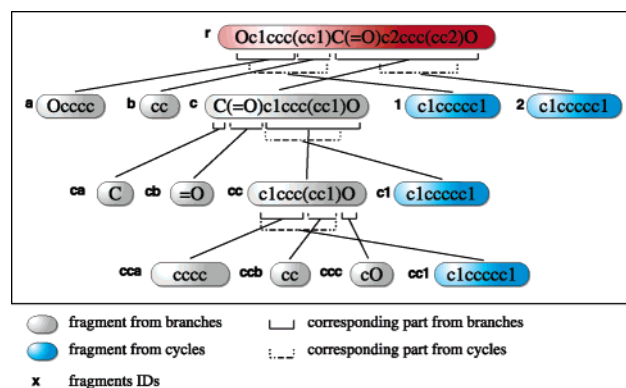
```

289 pruning, which employ different partitions of the training  
290 set. Indeed, the original data set is randomly divided into a  
291 growing and a pruning set. This division is done using a 2:1  
292 split, i.e., two-thirds for growing a rule and one-third for  
293 pruning the rule.

294 **2.3.3. Seeds and Fragments.** While searching for a single  
295 rule, SMIREP employs the *fragment tree* of a seed compound  
296 to guide the search through the space of potential rules. The  
297 fragment tree is obtained by syntactically decomposing the  
298 SMILES representation of the compound and all growing  
299 and pruning operations employ this fragment tree. Further-  
300 more, all rules evaluated (starting from a particular seed)  
301 will also cover that seed.

302 To obtain the fragment tree, SMIREP splits a SMILES  
303 string into cyclic fragments and branching fragments.  
304 Branching fragments are extracted from a SMILES string  
305 as follows: given a SMILES string of the form  $A(B)C$ , find  
306 the first branch, denoted by opening and corresponding  
307 closing brackets. The substring ranging from the start of the  
308 string to the opening branch is defined as fragment  $A$  with  
309 label  $a$ , the branch itself as fragment  $B$  with label  $b$ , and the  
310 rest after the branch as fragment  $C$  with label  $c$  (cf. Figure  
311 1). Each  $B$  and  $C$  fragment can contain further branches.  
312 This splitting is applied recursively, until no more branches  
313 can be found. Note that we neither use a unique SMILES  
314 representation nor a canonical form, when fragmenting the  
315 SMILES strings. However, as the fragments are later on  
316 evaluated using the OpenBabel toolkit’s SMARTS matching  
317 feature, any equivalent SMILES fragment would match a  
318 given compound.

319 Cyclic fragments are extracted in order to be able to  
320 represent ring structures and other types of cyclic structures.  
321 To ease the parsing of the string, each cycle number in the



**Figure 1.** An example fragment tree of 4,4'-dihydroxybenzophenone. The original SMILES string for this molecule is  $Oc1ccc(cc1)C(=O)c2ccc(cc2)O$ , which is shown at the top of the tree. The first three fragments in the second line (colored gray) result from the branching decomposition, the other ones (colored blue) from the cycle identification. After the first decomposition, only one fragment can further be decomposed, namely the fragment with ID  $c$  shown in the third level of the tree. Again only the fragment with ID  $cc$  can be decomposed further, resulting in the final level of the tree. After the last decomposition, no more fragments can be generated. The leaves of the tree, namely fragments  $a$ ,  $b$ ,  $1$ ,  $2$ ,  $ca$ ,  $cb$ ,  $c1$ ,  $cca$ ,  $ccb$ ,  $ccc$ , and  $cc1$  are the so-called ground fragments. Please note that the IDs last character denotes the label of the fragment. The IDs given here are purely used for clarity; they are neither constructed nor used in SMIREP.

SMILES string is first assigned a unique value, as the  
SMILES language allows the ‘reuse’ of cycle numbers. To  
split a string into cyclic fragments, we extract the substrings  
within the corresponding numbers. The fragments are  
‘cleaned’ before testing their coverage on the database, i.e.,  
other link numbers not denoting a full ring are removed as  
well as redundant opening or closing brackets. Examples of  
cyclic fragments are shown in Figure 1 colored in blue. Like  
the branching fragments, the extraction of cyclic fragments  
is done recursively.

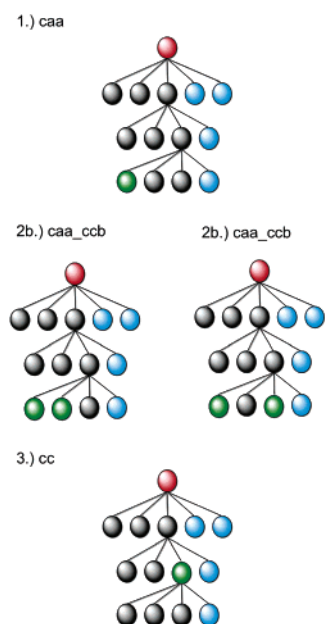
Please note that not all possible rings in a compound can  
be found this way. For instance, if a compound’s SMILES  
representation contains  $S = "c12ocnc2cccc1"$ , the fragment  
decomposition will only extract one fragment containing both  
rings ( $CF_1 = "c12ocnc2cccc1"$ ) and one containing the inner  
ring alone ( $CF_2 = "c1ocnc1"$ ). [Here, ‘inner ring’ is used  
in the sense that one ring is within the other in the SMILES  
representation.]

The cyclic and branching fragments found in the above  
way form a tree, a so-called *fragment tree*. The leaves in  
the tree are fragments, which cannot be decomposed any  
further. We call these fragments *ground fragments*. Figure  
1 shows such a tree for a small example compound.

**2.3.4. Growing.** For each of the growing iterations,  
SMIREP first selects a seed example, computes the corre-  
sponding fragment tree, and learns a rule as follows. First,  
the algorithm uses each ground fragment as an initial rule  
and evaluates it on the data set. The top  $k$  most promising  
rules (where  $k$  is a positive integer) are selected for the next  
refinement iteration. The parameter  $k$  is later on referred to  
as the *beamsize*.

The scoring heuristic used in SMIREP is weighted  
information gain (WIG), as originally suggested by Fürn-  
kranz.<sup>22</sup> It is defined as follows<sup>33</sup>

$$WIG(r) = -p(r)(IC(r) - IC(r'))$$



**Figure 2.** A hypothetical refinement example trace of an *Ascending* refinement, reflecting the example fragment tree in Figure 1. The green nodes correspond to the fragment or combined fragment evaluated during the *Ascending* refinement. Assume that the fragment **cca** is interesting according to the scoring function. Possible refinements would be combining fragment **cca** with **ccb**, resulting in a fragment **cca\_ccb**. If either of the two new fragments receive a good score, the next refinement is to combine all three fragments **cca\_ccb\_ccc**, which is actually the same as fragment **cc**, which is then the next one to be evaluated.

356 where  $p(x)$  denotes the number of active examples covered  
 357 by the rule  $x$ ,  $IC(x)$  denotes the information content of a rule  
 358  $x$ ,  $r$  denotes the current rule, and  $r'$  denotes the predecessor  
 359 of current rule, i.e., the current rule before the last refinement.  
 360 The information content (IC) is defined as

$$IC(x) = -\log \frac{p(x)}{p(x) + n(x)}$$

361 where  $p(x)$  denotes the number of active examples covered  
 362 by rule  $x$ , and  $n(x)$  denotes the number of inactive examples  
 363 covered by rule  $x$ . For the WIG measure, the difference in  
 364 the information content of a rule and the same rule after  
 365 refinement is weighted by the number of covered active  
 366 examples.

367 Like many machine learning algorithms, SMIREP uses a  
 368 *refinement operator*. A refinement operator essentially  
 369 generalizes or specializes an existing rule or pattern. The  
 370 refinement operator used in SMIREP is defined in Figure 3.  
 371 In principle, refinement proceeds in a bottom-up manner,  
 372 i.e., specializing a rule each time the operator is employed.  
 373 In SMIREP new rules are constructed by either combining  
 374 corresponding fragments from the tree (*Ascending*) or by  
 375 adding new fragments to an existing rule (*Lengthening*).

376 While the *Ascending* refinement operator allows only to  
 377 learn rules based on the fragments siblings and parents, the  
 378 *Lengthening* refinement operator allows more complex rules  
 379 to be learned. The *Lengthening* refinement operator allows  
 380 the addition of new fragments to an existing rule. Consider  
 381 the example where fragment **cc** (taken from the example  
 382 fragment tree in Figure 1) does not perform better than  
 383 **cca\_ccb**. Fragment **cc** is therefore not further refined using

*Input*: a conjunction of fragments  $X$  and corresponding labels of all fragments in the conjunction, and the fragment tree  $T$ .

*Output*: the refinements  $X'_l$  from the *Lengthening* refinement and the refinement  $X'_a$  from the *Ascending* refinement.

- *Lengthening*:

A ground fragment or a numerical fragment  $f$  is added to the existing conjunctions of fragments, that is  $X'_l := X \wedge f$  for all  $f$ 's.

- *Ascending*:

The last fragment of the conjunction is refined with respect to its label and parent (taken from  $T$ ). Depending on the label of the last fragment of  $X$ , the following patterns are constructed (where  $A(B)$  for example denotes the SMILES code generated by combining the SMILES codes of the fragments labeled  $a$  and its sibling labeled  $b$ ):

If the fragment label was:

- $a$ : then construct new fragments  $A(B)$  and  $AC$ , with labels  $ab$  and  $ac$  respectively.
- $b$ : then construct a new fragment  $A(B)$  with label  $ba$
- $c$ : then construct a new fragment  $AC$  with label  $ca$
- $i$ , where  $i$  is an integer (the unique cyclic number from the extracted cyclic fragment): then construct a new fragment, where the fragment is the parent's fragment. (this indicates ring structures)
- $ab$ ,  $ac$ ,  $ba$ , or  $ca$ : Construct a fragment  $A(B)C$ , where  $A(B)C$  is the parent of  $A$ ,  $B$ , or  $C$ .
- $r$ : then do not construct a new fragment as the last fragment was a root node.

**Figure 3.** The refinement operator used in SMIREP. In each iteration of the algorithm, both refinements of the operator can be applied. During the search, the rule with the best *score* (see text) is selected. Please note, that in the ascending part of the operator, no construction of a new fragment labeled  $b$  and  $c$  alone is performed, as both fragments rely on an atom and bond from fragment labeled  $a$ . Consider the fragment  $C(Cl)Cl$ , with subfragments  $\mathbf{a} = C$ ,  $\mathbf{b} = Cl$ , and  $\mathbf{c} = Cl$ . Combining  $\mathbf{b}$  and  $\mathbf{c}$  would require the carbon atom  $C$  from fragment  $\mathbf{a}$ .

the *Ascending* operator. However, combining it with another ground fragment might potentially perform better than the fragment itself. Therefore the *Lengthening* refinement can add new fragments (in the form of ground fragments) to an existing rule. The meaning of such a composite rule is that both fragments have to occur simultaneously within a compound to be classified as active.

Furthermore, SMIREP allows the use of numerical attributes. This has been incorporated in the algorithm by generating new types of fragments, *numerical fragments*, denoting that a particular numerical attribute is *less than* or *greater or equal to* some numerical value. Only those values which are true on the current seed are considered. These fragments can only be added during the *Lengthening* refinement of the growing stage. By adding more than one numerical constraint using the same attribute, it is possible for SMIREP to use intervals, i.e., it is possible to have rules containing the following constraint:  $\text{'logP} > -1.11' \wedge \text{'logP} \leq 3.21'$ . These numerical constraints allow SMIREP to be

used in the quantitative structure–activity relationships (QSAR) setting. To avoid overfitting, SMIREP first discretizes the numerical attributes into equal frequency bins. During each iteration of the growing stage, all borders of these bins are dynamically evaluated on the current growing set, and the borders are added as new attributes. If a particular seed possesses a  $\log P$  value of 3.24, and the binning resulted in the four borders  $B_s = [0.33, 2.66, 4.99, 7.32]$ , then the following attributes are evaluated: ‘ $\log P > 0.33$ ’, ‘ $\log P > 2.66$ ’, ‘ $\log P \leq 4.99$ ’, and ‘ $\log P \leq 7.32$ ’. The number of bins used in this work has arbitrarily been set to five.

**2.3.5. Pruning.** To avoid overfitting of the rules learned in the growing stage of the algorithm, the rules are pruned using the pruning set. The pruning is performed in reverse order of the growing of rules, i.e., the refinements are “undone”. To this aim, the actual refinement history is stored for each rule. All rules resulting from this reverse refinement are evaluated using the scoring function on the examples in the pruning set, and the *best* one is selected as the rule learned for the particular seed.

The pruning metric (or scoring function) used is the improved pruning method  $\nu^*$  as suggested by Cohen<sup>23</sup> and is defined as follows

$$\nu^*(r) = \frac{p(r) - n(r)}{p(r) + n(r)}$$

where  $p(r)$  denotes the number of active examples covered by rule  $r$ , and  $n(r)$  denotes the number of inactive examples covered by  $r$ . The  $\nu^*$  measure is equivalent to precision.<sup>32</sup>

**2.4. Implementation.** The SMIREP system has been developed in the programming language Python (version 2.3). Python allows rapid prototype development, due to a wide range of available libraries. For SMARTS matching, the open-source chemical library OpenBabel (version 1.100, <http://openbabel.sourceforge.net>) is employed. All experiments were run on a PC running Suse Linux 9.2 with an Intel Pentium IV-3.2 GHz CPU and 2 GB of main memory. The SMIREP source code is freely available under the GNU General Public License (see section 7 for details).

### 3. RESULTS

**3.1. Validation. 3.1.1. Cross-Validation.** Tenfold cross-validation was used to evaluate the performance of SMIREP on the three different databases. This means that each complete database was randomly divided into 10 equally sized parts. Each part was once removed from the complete database as a hold out test set, while the remaining other 9 parts were used as a training set for the model. Predictions for the test sets were compared to the actual classifications, to estimate the predictive accuracy. This process was repeated for all 10 parts, so that each part served once as a test set, and predictions for all compounds in the data set are available. As SMIREP’s algorithm is heuristic, we repeated the 10-fold cross-validation 10 times to obtain a good estimate of the algorithm’s mean accuracy. We call this a  $10 \times 10$ -fold cross-validation. In the following sections, we report on the mean predictive accuracies as well as the mean area under ROC curve (see below).

**3.1.2. ROC Analysis.** A common way to evaluate the performance of a classifier is to employ a confusion matrix.

In a confusion matrix the four different possible outcomes (see Table 1) of a single prediction for a two-class problem are displayed in a two-by-two matrix, where the rows represent the number of entries belonging to the actual class, while the columns represent the entries belonging to the predicted class.

Often however, a simple confusion matrix does not properly reflect the classifier’s performance. For a more detailed and proper analysis of a classifier, receiver operating characteristics (ROC) curves are employed. ROC curves were first developed for signal detection.<sup>34–37</sup> They are substantially employed in medical tests and have become a standard in the data-mining and machine learning communities to compare different classifiers.

To construct an ROC curve for a classifier, one orders the classifier’s predictions by some criterion (typically confidence of a prediction) and then plots the *true positive rate* (defined as  $\text{TP} / (\text{TP} + \text{FN})$ ) along the  $y$ -axis against the *false positive rate* (defined as  $\text{FP} / (\text{TN} + \text{FP})$ ) along the  $x$ -axis for all possible cutoff values of the criterion values. The resulting curve lies within the unit-square (the ROC space). An ideal ROC curve would be a line along the top left-hand corner (0,1) in ROC space, as it would not produce any false positives (or false actives). In real-world applications this occurs only rarely. The ROC curve for a good prediction should however always be to the left of the diagonal between the two axes. The closer the curve tends toward (0,1), the more accurate are the predictions made.

To compare two different prediction methods, both ROC curves are plotted in the same ROC space. The curve running closer to the left and top border is considered to provide a *better* predictor. Another good measurement to compare ROC curves analysis is that of the *area under the ROC curve* (AUC).<sup>37,38</sup> The AUC gives an overall measure of accuracy of a predictor.

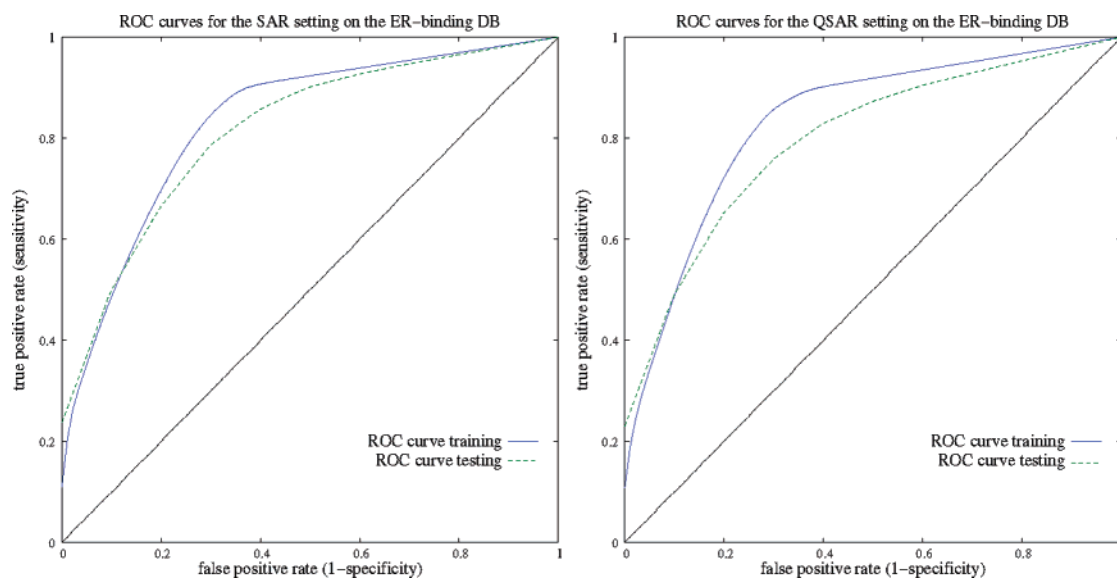
For a more detailed introduction to ROC curves and the construction of ROC curves for rule learner, we refer the reader to Appendix A (Supporting Information).

**3.2. Experiments.** We evaluated SMIREP on the three databases described in section 2.1. The aim of these experiments was 2-fold: first to demonstrate that activity classification using SMIREP yields accurate rules, and second, to show that meaningful rules can be found, which are sometimes in consensus with the published literature.

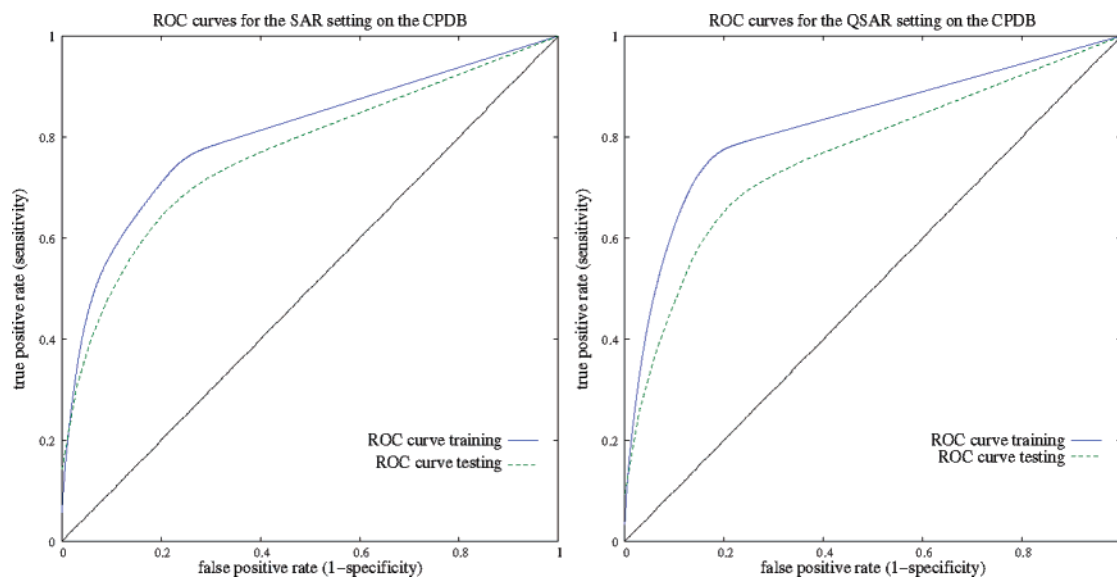
**3.2.1. Settings.** For each of the following experiments, we have chosen arbitrarily the number of seeds such that 10% of compounds classified as active in the database are employed during the growing stage of SMIREP. For example, if the database contain 120 active compounds and 150 inactive ones, we chose the number of seeds to be 12. That means, that SMIREP induces 12 rules for each iteration. To test the effect of different beamsizes, we evaluated SMIREP for each experiment using beamsizes for  $k = 5, 10, \text{ and } 20$ . To examine the influence of numerical attributes, we performed two separate experiments, one using only structural information (SAR-setting) and one using the structural information together with  $\log P$  values and the overall molecular weight of the compounds (QSAR-setting).

**3.2.2. DSSTox NCTREER.** The database of the 232 chemical compounds from the EPA’s DSSTox NCTREER Database was downloaded from [http://www.epa.gov/nheerl/dsstox/sdf\\_nctrer.html](http://www.epa.gov/nheerl/dsstox/sdf_nctrer.html). We translated this database to SMILES





**Figure 4.** The ROC curves from the SAR and QSAR experiments for training (blue) and testing (green) on predicting the ER-binding database with a beamsize  $k = 5$ . The ROC curves are averaged over the  $10 \times 10$  ROC curves resulting from performing ten times a 10-fold cross-validation. The black line indicates the diagonal.



**Figure 5.** The ROC curves from the SAR and QSAR experiments for training (blue) and testing (green) on predicting the CPDB mutagenicity database with beamsize  $k = 5$ . The black line indicates the diagonal.

521 codes using the OpenBabel toolkit. This procedure was  
 522 necessary, as some SMILES codes provided in the database  
 523 were corrupt. Furthermore, we removed chiralities (see  
 524 section 2.2.2) and bond directions from the SMILES strings,  
 525 as the current version of SMIREP cannot deal with this  
 526 information. We believe (and the experiments will show)  
 527 that omitting this information provides SMIREP with enough  
 528 structure information to induce meaningful and accurate  
 529 patterns.

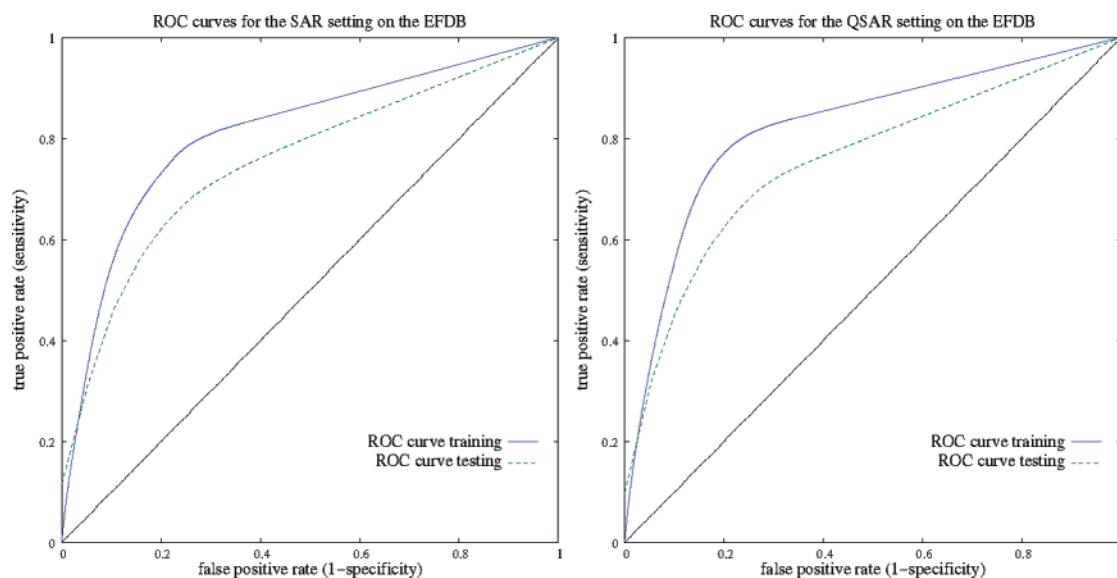
530 The results are depicted in Table 2. Overall, SMIREP  
 531 seems to perform comparably to the decision tree approach  
 532 of Hong et al.<sup>39</sup> Although, no specific accuracy is given, Tong  
 533 et al.<sup>40</sup> report on accuracies of a 3-fold cross-validation  
 534 experiment resulting roughly in the same prediction accu-  
 535 racies on the training set (approximately 76%, taken from  
 536 Figure 13.8, p 302). In another recent publication, Hong et  
 537 al.<sup>41</sup> report on accuracies of 96.6% employing a method  
 538 called decision forest (DF). DF essentially induces a large

539 number of decision trees and builds a consensus model. In  
 540 comparison of the original decision tree approach by Hong  
 541 et al.<sup>39</sup> to SMIREP, no preselection of structural alerts has  
 542 been performed, as SMIREP is able to extract the relevant  
 543 information during the learning stage. An example set of  
 544 rules found during one round of a 10-fold cross-validation  
 545 is shown in Figure 3.

546 The computation time varies between the SAR and the  
 547 quantitative SAR (QSAR) setting. While one complete 10-  
 548 fold cross-validation using beamsize  $k = 5$  averages at around  
 549 75 s in the SAR-setting, while SMIREP requires 513 s for  
 550 a 10-fold cross-validation in the QSAR setting.

551 Figure 4 depicts the two averaged ROC curves for  
 552 SMIREP for the SAR setting and the QSAR setting when  
 553 predicting the ER-binding database for beamsize  $k = 5$ . The  
 554 curves represent the averaged  $10 \times 10$ -fold cross-validation  
 555 results. The averaged area under the ROC curve (AUC) was  
 556 0.832 (SAR) and 0.833 (QSAR) for the training sets and





**Figure 6.** The ROC curves from the SAR and QSAR experiments for training (blue) and testing (green) on predicting the ER-binding database with a beamsize  $k = 20$ . The ROC curves are averaged over all fold experiments.

557 0.816 (SAR) and 0.795 (QSAR) for the corresponding test  
 558 sets. The ROC curves for testing and training are similar,  
 559 which indicates that SMIREP does not overfit.

560 As mentioned above, Tong et al.<sup>39</sup> have identified a  
 561 number of structural features contributing to the ER-binding  
 562 activity of chemical compounds. Three structural alerts, the  
 563 steroid skeleton, the steroid diethylstilbestrol (DES) skeleton,  
 564 and the phenolic ring skeleton, were manually selected as  
 565 structural alerts, in order to predict the activity class of a  
 566 potential ER-binding compound. In an earlier publication,  
 567 Fang et al.<sup>24</sup> have used information about the presence of a  
 568 ring structure, an aromatic and possibly a phenolic ring  
 569 structure, and the DES skeleton to build a rule system  
 570 evaluating the likeliness of a compound being a possible ER  
 571 ligand. The handcrafted rule system by Fang et al. is as  
 572 follows:

573 1. If a chemical contains no ring structure, then it is  
 574 unlikely to be an ER ligand.

575 2. If a chemical has a nonaromatic ring structure, then it  
 576 is unlikely to be an ER ligand if it does not contain an O, S,  
 577 N, or other heteroatom for bonding. Otherwise its binding  
 578 potential is dependent on the existence of the key structural  
 579 features.

580 3. If a chemical has a non-OH aromatic structure, then its  
 581 binding potential is dependent on the existence of key  
 582 structural features (e.g., logP, precise O–O distance, etc.).

583 4. If a chemical contains a phenolic ring, then it tends to  
 584 be an ER ligand if it contains any additional key structural  
 585 features. For the chemical containing a phenolic ring  
 586 separated from another benzene ring with the number of  
 587 bridge atoms ranging from none to three, it will be most  
 588 likely an ER ligand.

589 The main structural rule in this system is rule number 4,  
 590 which translates into the following: “if a compound pos-  
 591 sesses an aromatic ring connected by one to three atoms to  
 592 a phenolic ring, then the compound is likely to be an ER  
 593 ligand”. We have assessed this rule using the OpenBabel  
 594 tool *obgrep*. *obgrep* works similar to the UNIX grep  
 595 command, but instead of using regular expressions it  
 596 performs a SMARTS search through a database of chemical

**Table 4.** Accuracy and Area under ROC Curve for the CPDB Data Set for Different Beamsizes ( $k$ ) from the  $10 \times 10$ -Fold Cross-Validations<sup>a</sup>

setting	$k$	acc training	AUC training	acc testing	AUC testing
SAR	5	76.11 (0.28)	0.804 (0.012)	72.87 (1.21)	0.768 (0.056)
	10	75.95 (0.16)	0.801 (0.011)	72.16 (1.05)	0.761 (0.055)
	20	76.07 (0.23)	0.803 (0.010)	72.60 (0.50)	0.765 (0.057)
QSAR	5	79.73 (0.26)	0.825 (0.015)	73.90 (0.99)	0.764 (0.055)
	10	79.61 (0.52)	0.821 (0.016)	74.26 (1.53)	0.766 (0.054)
	20	78.58 (0.44)	0.811 (0.019)	74.20 (1.30)	0.766 (0.040)

<sup>a</sup> The numbers in brackets denote the standard deviation. Increasing the beamsizes in the CPDB experiment does not yield any significant change in the performance of SMIREP.

597 structures. Overall, this single rule matches 33 of the  
 598 classified as active compounds, while matching 15 of the  
 599 non-ER ligands. This however, does not seem to be a very  
 600 good structural rule when predicting the activity class of  
 601 unseen compounds. We have compared an example SMIREP  
 602 rule set with the rule set by Fang et al. This discussion is  
 603 included in Table 3. Overall, a number of rules correspond  
 604 to these expert rules.

605 We believe that this example rule set could aid initially  
 606 in the construction of an expert rule system for classifying  
 607 potential ER-binding molecules, like the one presented by  
 608 Fang et al.<sup>24</sup> Although the discovered rules do not present  
 609 previously unknown knowledge, they can be used as a first  
 610 step and guideline for experts.

611 **3.2.3. CPDB.** The second data set we evaluated SMIREP  
 612 on was derived from the carcinogenic potency database  
 613 (CPDB). The settings used were the default settings described  
 614 in section 3.2.1. The number of seeds was set to 30, which  
 615 corresponds to 10% of compounds classified as active in the  
 616 training set.

617 The results are depicted in Table 4. Overall, the predictive  
 618 performance does not vary as much as in the ER-binding  
 619 experiment and results also in a lower predictive perfor-  
 620 mance. This might be due to the nature of the application:  
 621 the CPDB database contains more heterogeneous molecules  
 622 when compared to the NCTRER database. When comparing

**Table 5.** Performance of SAR Models for *Salmonella* Mutagenicity Reported in the Literature<sup>7 a</sup>

author	citation	method	accuracy
Perrotta et al.	42		73.9
Klopman and Rosenkranz	18	CASE	72
Klopman and Rosenkranz	18	MULTICASE	80
Klopman and Rosenkranz	18	CASE/GI	47
Helma et al.	7	MOLFEA/J48	75.5
Helma et al.	7	MOLFEA/PART	75.0
Helma et al.	7	MOLFEA/SMO,E1	76.1
Helma et al.	7	MOLFEA/SMO,E2	73.7

<sup>a</sup> The accuracies for the different MOLFEA approaches were the results on unoptimized structures and averaged overall four different settings.

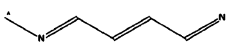

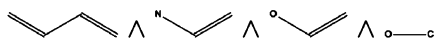
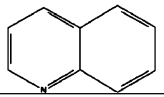


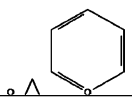

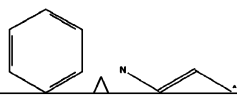
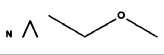

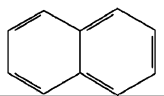
623 these results to the literature, i.e., to the MOLFEA data-  
 624 mining system<sup>7</sup> and to CASE<sup>17</sup> and MULTICASE,<sup>18</sup> this drop  
 625 in predicted accuracy has to be seen in a different perspective.  
 626 The results published by Helma<sup>7</sup> and others<sup>18,42</sup> are sum-  
 627 marized in Table 5. Although SMIREP achieves similar,  
 628 though slightly lower, accuracies than the other methods, it  
 629 is not quite clear whether these differences are statistically  
 630 significant, as it is not possible to test for statistical  
 631 significance purely based on accuracies from 10-fold cross  
 632 experiments without the standard deviations. Therefore, we  
 633 have calculated a 99% confidence interval for the best result  
 634 from SMIREP. The accuracy of the best MOLFEA approach  
 635 (SMO/E1) lies within the interval, indicating that the

636 differences are not statistically significant. In contrast to  
 637 MOLFEA, however, the produced rules by SMIREP are easy  
 638 to understand, and SMIREP's ability to employ more  
 639 complex structures than just linear fragments seems to aid  
 640 in the rule induction. An example set of rules is presented  
 641 in Table 6. Some parts of the fragments used in this example  
 642 rule set were also identified in the MOLFEA approach.  
 643 However, many of the SMIREP rule sets contain the  
 644 fragments 'C1OC1' (coding for epoxide, a structure that is  
 645 often associated with mutagenicity) or more complex frag-  
 646 ments like 'c1c2ncccc2ccc1', like in the example rules set  
 647 in rules 8 and 4, respectively, which can neither be found  
 648 nor represented using linear fragments.

**3.2.4. EFDB.** For prediction of the biodegradability in  
 649 terms of biodegradable or nonbiodegradable, we have used  
 650 328 compounds, 185 considered active (biodegradable) and  
 651 143 inactive (nondegradable) compounds. As with the other  
 652 experiments, we evaluated the results in terms of accuracy  
 653 as well as using ROC analysis. The accuracies are compared  
 654 to previously published results.  
 655

To compare to other published approaches we have  
 656 modified the evaluation and performed a  $5 \times 10$ -fold cross-  
 657 validation (using the original folds published by Blockeel et  
 658 al.<sup>27</sup>). We repeated each of these fold-wise experiments  
 659 arbitrarily five times to allow for a more accurate estimate  
 660 on the accuracy due to the selection of seeds in SMIREP.  
 661 We have selected the same folds as Blockeel et al.<sup>27</sup> The  
 662

**Table 6.** Example Rule Set Induced by SMIREP on the CPDB

No	Rules (SMILES)	Rules (2D)
1	nccccnc	
2	N=O	
3	cccc $\wedge$ ccN $\wedge$ ccO $\wedge$ OC	
4	c1c2ncccc2ccc1	
5	CBr	
6	CCl $\wedge$ CN	
7	=O $\wedge$ c1occcc1	
8	C1OC1	
9	c1ccccc1 $\wedge$ Nccc	
10	N $\wedge$ CCOC	
11	NN	
12	c1cc2ccccc2cc1	

**Table 7.** Average Accuracies and Areas Under ROC Curve for the EFDB Data Set for a Number of Different Beamsizes ( $k$ )<sup>a</sup>

setting	$k$	acc training	AUC training	acc testing	AUC testing
SAR	5	77.68 (1.41)	0.811 (0.016)	71.81 (5.61)	0.747 (0.078)
	10	77.05 (1.02)	0.811 (0.015)	71.53 (6.27)	0.756 (0.086)
	20	77.07 (1.18)	0.810 (0.015)	71.65 (6.29)	0.754 (0.084)
QSAR	5	79.90 (1.06)	0.826 (0.014)	73.51 (5.08)	0.744 (0.081)
	10	79.50 (1.25)	0.826 (0.015)	73.14 (5.99)	0.743 (0.079)
	20	79.02 (1.55)	0.822 (0.016)	74.32 (5.83)	0.756 (0.080)

<sup>a</sup> The accuracies of SMIREP in the QSAR setting are comparable to that published by Blockeel et al.<sup>27</sup> when using the *Global* and *R* information.

**Table 8.** Mean Accuracies on the  $5 \times 10$ -Fold Cross-Validation of a Number of Relational Learning Approaches Published by Blockeel et al.<sup>27</sup> Using the *Global* as Well as the Atom/Bond Information (*R*)<sup>a</sup>

method	accuracy
ICL	73.2
Tilde	74.1
S-CART	71.9

<sup>a</sup> The SMIREP QSAR approach performs comparably to these approaches with a mean accuracy on the test data ranging from 73.81% to 74.32% depending on the beamsizes  $k$ .

663 results of the SMIREP experiments are depicted in Table 7.  
 664 The performance of SMIREP to other approaches on this  
 665 data set can be seen in Table 8. Here, we can only compare  
 666 the accuracies and not the average AUC. In their publication,  
 667 Blockeel et al.<sup>27</sup> have tested a number of relational learning  
 668 methods on this data set, with varying background knowl-  
 669 edge. The information about the compounds contained in  
 670 the database is organized into four types of background  
 671 knowledge: *Global*, *P1*, *P2*, and *R*. The *Global* information

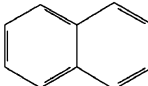
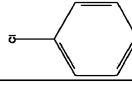




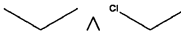

672 reflects global information about a molecule such as the  
 673 molecular weight and logP. The *R* type encapsulate the atom  
 674 and bond information, while the types *P1* and *P2* are used  
 675 as aggregates reflecting information about the frequencies  
 676 of certain substructures occurring in a given compound (*P2*  
 677 employs a set of 59 predefined substructures). We have  
 678 compared the performance of SMIREP against the algorithms  
 679 used by Blockeel et al.<sup>27</sup> in the *Global* and *R* type setting,  
 680 as this is the closest to SMIREP in the QSAR setting. The  
 681 relational learning approaches tested by Blockeel et al. were  
 682 not evaluated on the pure atom/bond information *R*. Compar-  
 683 ing the mean accuracies of SMIREP against the ones from  
 684 Blockeel et al.<sup>27</sup> (in Table 8) shows that SMIREP achieves  
 685 comparable performance to that of relational learning  
 686 methods on the classification task.

687 Although the prediction accuracies of SMIREP when  
 688 compared to other approaches are quite similar, the perfor-  
 689 mance of SMIREP comes to some extent as a surprise. As  
 690 SMIREP is neither able to induce rules for both activity  
 691 classes at the same time nor is—currently—able to incorporate  
 692 the absence of a fragment within a compound, the good  
 693 results are hard to explain. However, in the light of this  
 694 application, these features would very much be required to  
 695 predict readily degradable compounds and specific half-life  
 696 times.

#### 4. DISCUSSION

697 SMIREP is essentially a QSAR based approach able to  
 698 extract relevant structural fingerprints quickly. Obviously,  
 699 modeling chemical problems on this level has certain  
 700 drawbacks. Other approaches in activity prediction employ,  
 701 for example, the docking site of the protein under consid-  
 702 eration. However, in most cases, this information is not or  
 703

**Table 9.** Example Rule Set Discovered on the EFDB Data Using the QSAR Setting <sup>a</sup>

No	Rules (SMILES)	Rules (2D)
1	<chem>c1ccc2ccccc2c1</chem>	
2	$\log P > 2.97 \wedge \text{c1ccccc1[Cl]}$	$\log P > 2.97 \wedge$ 
3	<chem>C1CCCCC1</chem>	
4	<chem>C([Cl])[Cl]</chem>	
5	<chem>O=NN</chem>	
6	<chem>N=O</chem>	
7	$\text{CCC} \wedge \text{CC[Cl]}$	
8	$\text{mweight} > 118.175 \wedge \log P \leq 2.97 \wedge \text{Ncc}$	$\text{mweight} > 118.175 \wedge \log P \leq 2.97 \wedge$ 

<sup>a</sup> The rules predict nonbiodegradation. The accuracy of this specific rule set is 81.8% (correctly predicting 11 out of 14 examples on the corresponding test set).



704 only partially available. And, equally important, these more  
705 sophisticated approaches typically require substantially more  
706 computation time.

707 SMIREP is, to some extent, similar to MULTICASE,<sup>17</sup>  
708 in that it constructs fragments based on the structure of  
709 chemical compounds and uses these fragments as an integral  
710 part of the machine learning approach. However, MULTI-  
711 CASE constructs all possible linear (and to some extent  
712 branched) fragments up to a fixed length of 10 non-hydrogen  
713 atoms. Using statistical testing, the found fragments are  
714 divided into significantly activating fragments (*biophores*)  
715 and significantly deactivating fragments (*biophobes*). MUL-  
716 TICASE then automatically defines major biophore classes,  
717 and within each class it identifies fragments modifying the  
718 overall class activity. These modifying fragments together  
719 with calculated numerical attributes, such as logP, octanol-  
720 water, charges, densities, etc., are used as so-called *modula-*  
721 *tors* to refine an activity model of a biophore using a divide-  
722 and-conquer search strategy.

723 The differences between MULTICASE and SMIREP are  
724 3-fold: First, SMIREP does not use biophobes to indicate  
725 inhibition (or inactivity), as the use of deactivating fragments  
726 is typically not desired within the SAR setting. Second, to  
727 avoid overfitting of rules, SMIREP does not employ modula-  
728 tors to fine-tune activities of found biophores. Third, in  
729 contrast to MULTICASE, SMIREP is able to employ more  
730 complex structures, including rings, and it does not impose  
731 a restriction on the overall length of fragments.

732 The difference of SMIREP with regard to other QSAR  
733 approaches can be seen in SMIREP's ability to extract  
734 relevant knowledge in the form of structural alerts on during  
735 the learning stage. Other approaches rely on precalculated  
736 fingerprints, like MDL keys,<sup>1,2</sup> or use physiochemical and  
737 structural indices calculated by specialized software like  
738 Molconn-Z.<sup>3</sup> Here, we have presented a system, able to focus  
739 on only the important structural features hidden in the  
740 database.

741 In a previous publication,<sup>20</sup> we have compared SMIREP  
742 to graph mining approaches which have been employed in  
743 SAR. The advantage of SMIREP over most of these systems  
744 lies in the small set of rules produced. While approaches  
745 such as gSpan, closeGraph,<sup>15</sup> FSG,<sup>5</sup> and AGM<sup>12</sup> typically  
746 find a *large* set of patterns satisfying a minimum frequency  
747 threshold, which are not necessarily predictive, SMIREP  
748 directly builds a *small* set of predictive rules. Furthermore,  
749 as these graph based approaches traverse the complete search  
750 space of possible patterns, they tend to be inefficient.  
751 SMIREP, on the other hand, is a heuristic approach and is  
752 able to induce rules faster.

753 Several improvements are possible for the SMIREP  
754 system. As SMIREP employs principle ideas from the  
755 machine learning algorithm IREP,<sup>22,23</sup> a next step could be  
756 the upgrade of the underlying learning algorithm to some of  
757 its successors, for instance RIPPER<sup>23</sup> (Repeated Incremental  
758 Pruning to Produce Error Reduction). RIPPER employs the  
759 rules found by IREP and repeatedly grows and prunes the  
760 found rules to improve the prediction accuracy on different  
761 splits for the training and validation sets. Compared to a  
762 number of other approaches, SMIREP only induces rules for  
763 compounds considered to be active, and no rules are found  
764 specifically for inactive compounds. That means that SMIREP  
765 has no mechanism to filter out obvious inactive compounds,

as done in the decision tree approach by Hong et al.<sup>39</sup> This  
is clearly a disadvantage when comparing predictive per-  
formance. One possible way to overcome this limitation is  
to employ the fragmentation approach together with a  
decision tree learner as the underlying learning algorithm or  
to learn rules for more than one class. For the actual  
application of SMIREP to SAR, learning rules from active  
compounds only is less problematic, as one is typically more  
interested in the understanding of activity rather than  
inactivity.

In the current implementation, SMIREP does not cater for  
stereoisomers possessing different activity levels. Indeed, the  
chirality information is disregarded during learning. How-  
ever, this information is a major factor when investigating  
the activity levels of natural compounds. In future versions  
of SMIREP, we intend to incorporate the chirality informa-  
tion encoded in the compound's SMILES codes into the  
system.

To accommodate the possibility to incorporate some sort  
of background or domain knowledge, it is further possible  
to add predefined structural alerts as new fragments during  
the growing stage of SMIREP. Although this would be in  
contrast to the idea of inducing alerts without any prior  
knowledge, it could overcome some limitations of SMIREP  
regarding larger ring structures.

## 5. CONCLUSION

In this paper we have presented a number of applications  
of a novel system, SMIREP, to predict activity classification  
in the SAR and QSAR problem setting. SMIREP combines  
principles of the chemical representation language SMILES  
with the inductive rule learner IREP. The novelty behind  
the SMIREP approach is the use of linear strings to induce  
rules containing complex structures such as trees and cycles.  
The applicability of SMIREP to classify chemical compounds  
was demonstrated on three diverse data sets. Overall, the  
predictive performance of SMIREP is comparable to existing  
methods. In contrast to other methods,<sup>24,39</sup> there is no need  
to employ preselected structural alerts or fingerprints for the  
classification task. Furthermore, as SMIREP is able to induce  
these alerts ab initio, the found rules can be employed to  
construct more fine-grained rules sets. We believe, therefore,  
that SMIREP is a valuable tool to analyze chemical  
databases.

The SMIREP system is available from <http://www.karwath.org/systems/smirep/> under the GNU General  
Public License. The Web page also contains the data files  
used in the Experimental Section. The system is provided  
in Python and C source code, including the required Python  
OpenBabel module OBGrep.

## ACKNOWLEDGMENT

The authors would like to thank Christoph Helma, David  
P. Enot, Ross D. King, and Siegfried Nijssen for helpful  
discussions and suggestions. This research was supported  
by the EU grant FP6-516169 Inductive Querying.

**Supporting Information Available:** Appendix A describ-  
ing the generation of ROC curves and a more detailed  
description of ROC analysis. This material is available free of  
charge via the Internet at <http://pubs.acs.org>.

824

## REFERENCES AND NOTES

- 825 (1) M.D.L. Information Systems, I. 14600 Catalina Street, San Leandro,  
826 CA 94577, U.S.A.
- 827 (2) Durant, J. L.; Leland, B. A.; Henry, D. R.; Nourse, J. G. Reoptimization  
828 of MDL Keys for Use in Drug Discovery. *J. Chem. Inf. Comput. Sci.*  
829 **2002**, *42*, 1273–1280.
- 830 (3) *Molconn-Z, version 3.50*; available from Hall Associates Consulting,  
831 2 Davis Street, Quincy, MA, 02170. Also available from EduSoft,  
832 LC, P.O. Box 1811, Ashland, VA 23005 and SciVision, Inc., 200  
833 Wheeler Road, Burlington, MA 01803.
- 834 (4) Dehaspe, L. Frequent Pattern Discovery in First-Order Logic, Thesis,  
835 K. U. Leuven: Belgium, 1998.
- 836 (5) Deshpande, M.; Kuramochi, M.; Karypis, G. Frequent Sub-Structure-  
837 Based Approaches for Classifying Chemical Compounds. In *Proc.*  
838 *ICDM-03*; IEEE Computer Society: Piscataway, NJ, U.S.A., 2003;  
839 pp 35–42.
- 840 (6) Kramer, S.; De Raedt, L.; Helma, C. Molecular Feature Mining in  
841 HIV data. In *Proc. KDD-01*; Provost, F., Srikant, R., Eds.; ACM  
842 Press: New York, U.S.A., 2001; pp 136–143.
- 843 (7) Helma, C.; Kramer, T.; Kramer, S.; De Raedt, L. Data Mining and  
844 Machine Learning Techniques for the Identification of Mutagenicity  
845 Inducing Substructures and Structure-Activity Relationships of Non-  
846 congeneric Compounds. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1402–  
847 1411.
- 848 (8) Zaki, M. Efficiently Mining Frequent Trees in a Forest. In *Proc. KDD-*  
849 *02*; Hand, D., Keim, D., Ng, R., Eds.; ACM Press: New York, U.S.A.,  
850 2002; pp 71–80.
- 851 (9) Yan, X.; Han, J. gSpan: Graph-based substructure pattern mining. In  
852 *Proc. ICDM-02*; IEEE Press: Piscataway, NJ, U.S.A., 2002; pp 721–  
853 724.
- 854 (10) Borgelt, C.; Berthold, M. R. Mining Molecular Fragments: Finding  
855 Relevant Substructures of Molecules. In *Proc. ICDM-02*; IEEE  
856 Press: Piscataway, NJ, U.S.A., 2002; pp 51–58.
- 857 (11) Inokuchi, A.; Kashima, H. Mining Significant Pairs of Patterns from  
858 Graph Structures with Class Labels. In *Proc. ICDM-03*; IEEE Press:  
859 Piscataway, NJ, U.S.A., 2003; pp 83–90.
- 860 (12) Inokuchi, A.; Washio, T.; Motoda, H. Complete Mining of Frequent  
861 Patterns from Graphs: Mining Graph Data. *Machine Learning* **2003**,  
862 *50*, 321–354.
- 863 (13) Kuramochi, M.; Karypis, G. Frequent subgraph discovery. In *Proc.*  
864 *ICDM-01*; IEEE Press: Piscataway, NJ, U.S.A., 2001; pp 179–186.
- 865 (14) Muggleton, S. Inductive logic programming. *New Generation Comput-*  
866 *ing* **1991**, *8*, 295–318.
- 867 (15) Yan, X.; Han, J. CloseGraph: Mining Closed Frequent Graph Patterns.  
868 In *Proc. KDD-03*; ACM Press: New York, U.S.A., 2003; pp 286–  
869 295.
- 870 (16) Kazius, J.; Nijssen, S.; Kok, J.; Bäck, T.; IJzerman, A. P. Substructure  
871 Mining Using Elaborate Chemical Representation. *J. Chem. Inf. Model.*  
872 **2006**, *46*, 597–605.
- 873 (17) Klopman, G. Artificial intelligence approach to structure-activity  
874 studies: Computer automated structure evaluation of biological activity  
875 of organic molecules. *J. Am. Chem. Soc.* **1984**, *106*, 7315–7320.
- 876 (18) Klopman, G. MultiCASE: A hierarchical computer automated structure  
877 evaluation program. Quantitative Structure-Activity Relationships.  
878 *Quant. Struct.-Act. Relat.* **1992**, *11*, 176–184.
- 879 (19) Helma, C. Iazar: Lazy Structure-Activity Relationships for Toxicity  
880 Prediction. In *Predictive Toxicology*; Helma, C., Ed.; Taylor &  
881 Francis: Boca Raton, London, New York, 2005; pp 479–499.
- 882 (20) Karwath, A.; De Raedt, L. Predictive Graph Mining. In *Proc. 7th*  
883 *International Conference of Discovery Science, DS 2004*; Suzuki, E.,  
884 Arikawa, S., Eds.; Springer-Verlag: 2004; Vol. 3245, pp 1–15.
- 885 (21) Weininger, D. SMILES, a Chemical Language and Information System  
886 I. Introduction and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988**,  
887 *28*, 31–36.
- (22) Fürnkranz, J.; Widmer, G. Incremental Reduced Error Pruning. In *Proc.*  
888 *ICML 1994*; Cohen, W. W., Hirsh, H., Eds.; Morgan Kaufmann: 1994;  
889 pp 70–77.
- (23) Cohen, W. W. Fast effective rule induction. In *Proc. ICML 1995*;  
890 Morgan Kaufmann: 1995; pp 115–123.
- (24) Fang, H.; Tong, W.; Shi, L.; Blair, R.; Perkins, R.; Branham, W.;  
891 Hass, B.; Xie, Q.; Dial, S.; Moland, C.; Sheehan, D. Structure–activity  
892 relationships for a large diverse set of natural, synthetic, and  
893 environmental estrogens. *Chem. Res. Toxicol.* **2001**, *14*, 280–294.  
894 895 896
- (25) Howard, P.; Boethling, R.; Jarvis, W.; Meylan, W. M.; Michalenko,  
897 E. M. *Handbook of Environmental Degredation Rates*; Lewis Publish-  
898 ers Inc.: Chelsea, MD.  
899
- (26) Džeroski, S.; Blockeel, H.; Kompars, B.; Kramer, S.; Pfahringer, B.;  
900 Laer, W. V. Experiments in Predicting Biodegradability. *Lect. Notes*  
901 *Comput. Sci.* **1999**, *1634*, 80–91.
- (27) Blockeel, H.; Džeroski, S.; Kompars, B.; Kramer, S.; Pfahringer, B.;  
902 Laer, W. V. Experiments In Predicting Biodegradability. *Appl. Artif.*  
903 *Intelligence* **2004**, *18*, 157–181.
- (28) Sayle, R. 1st-class SMARTS patterns, presented at EuroMUG 97,  
904 Verona, Italy, 1997. <http://www.daylight.com/meetings/emug97/Sayle/>.
- (29) McNaught, A.; Heller, S.; Stein, S. IUPAC- International Chemical  
905 Identifier. <http://www.iupac.org/projects/2000/2000-025-1-800.html>.
- (30) Mitchell, T. *Machine Learning*; McGraw-Hill: Boston, MA, U.S.A.,  
906 1997.  
907 908 909
- (31) Clark, P.; Niblett, T. The CN2 Induction Algorithm. *Machine Learning*  
910 **1989**, *3*, 261–283.
- (32) Fürnkranz, J.; Flach, P. A. ROC ‘n’ Rule Learning - Towards a Better  
911 Understanding of Covering Algorithms. *Machine Learning* **2005**, *58*,  
912 913 914 915 916
- (33) Fürnkranz, J. Separate-and-Conquer Rule Learning. *Artif. Intelligence*  
917 *Rev.* **1999**, *13*, 3–54.
- (34) Van Trees, H. L. *Detection, Estimation, and Modulation Theory (Part*  
918 *I)*; Wiley: New York, U.S.A., 1968.
- (35) Egan, J. P. *Signal Detection Theory and ROC Analysis*; Academic  
919 Press: New York, U.S.A., 1975.  
920 921 922
- (36) Swets, J. A. Measuring the accuracy of diagnostic systems. *Science*  
923 **1988**, *240*, 285–293.
- (37) Bradley, A. P. The use of the area under the ROC curve in the  
924 evaluation of learning algorithms. *Pattern Recognit.* **1997**, *30*, 1145–  
925 1159.  
926 927
- (38) Provost, F.; Fawcett, T. Analysis and Visualization of Classifier  
928 Performance: Comparison under Imprecise Class and Cost Distributions.  
929 In *Proceedings of KDD-97*; Heckerman, D., Mannila, H.,  
930 Pregibon, D., Eds.; AAAI Press: Menlo Park, CA, U.S.A., 1997; pp  
931 43–48.
- (39) Hong, H.; Tong, W.; Fang, H.; Shi, L.; Xie, Q.; Wu, J.; Perkins, R.;  
932 Walker, J. D.; Branham, W.; Sheehan, D. M. Prediction of Estrogen  
933 Receptor Binding for 58,000 Chemicals Using an Integrated System  
934 of a Tree-Based Model with Structural Alerts. *Environ. Health*  
935 *Perspect.* **2002**, *110*, 29–36.  
936 937 938
- (40) Tong, W.; Fang, H.; Hong, H.; Xie, Q.; Perkins, R.; Sheehan, D.;  
939 Receptor-Mediated Toxicity: QSARs for Estrogen Receptor Binding  
940 and Priority Setting of Potential Estrogenic Endocrine Disruptors. In  
941 *Predicting Chemical Toxicity and Fate*; Cronin, M. T. D., Livingstone,  
942 D. J., Eds.; CRC Press: Boca Raton, London, New York, 2004;  
943 Chapter 13, pp 285–314.
- (41) Tong, W.; Xie, Q.; Hong, H.; Fang, H.; Shi, L.; Perkins, R. Assessment  
944 of prediction confidence and domain extrapolation of two structure-  
945 activity relationship models for predicting estrogen receptor binding  
946 activity. *Toxicogenomics* **2004**, *112*, 1249–125.  
947 948 949
- (42) Perrotta, A.; Malacarne, D.; Taningher, M.; Pesenti, R.; Paolucci, M.;  
950 Parodi, S. A computerized connectivity approach for analyzing the  
951 structural basis of mutagenicity in Salmonella and its relationship with  
952 rodent carcinogenicity. *Environ. Mol. Mutagen.* **1996**, *28*, 31–50.

CI060159G

952