Automated detection of toxicophores and prediction of mutagenicity using PMCSFG algorithm

Leander Schietgat, Bertrand Cuissart, Kurt De Grave, Kyriakos Efthymiadis, Ronan Bureau, Bruno Crémilleux, Jan Ramon, Alban Lepailleur

Automated detection of toxicophores and prediction of mutagenicity using PMCSFG algorithm

Leander Schietgat,[a,b] Bertrand Cuissart,[c] Kurt De Grave,[d] Kyriakos Efthymiadis,[a] Ronan Bureau,[e] Bruno Crémilleux,[c] Jan Ramon,[f] and Alban Lepailleur*[e]

Abstract: Maximum common substructures (MCS) have received a lot of attention in the chemoinformatics community. They are typically used as a similarity measure between molecules, showing high predictive performance when used in classification tasks, while being easily explainable substructures. In the present work, we applied the Pairwise Maximum Common Subgraph Feature Generation (PMCSFG) algorithm to automatically detect toxicophores (structural alerts) and to compute fingerprints based on MCS. We present a comparison between our MCS-based fingerprints and 12 well-known chemical fingerprints when used as features in machine learning models. We provide an experimental evaluation and discuss the usefulness of the different methods on mutagenicity data. The features generated by the MCS method have a state-of-the-art performance when predicting mutagenicity, while they are more interpretable than the traditional chemical fingerprints.

Keywords: maximum common substructure, MCS, toxicophore, structural alert, machine learning, mutagenicity

1 Introduction

Historically, the toxicity of new chemicals was determined through in vivo studies involving rodents and other mammals but due to ethical concerns and regulatory guidelines[1,2], there has been a shift toward the use of alternative methods[3–5]. In this context, the use of in silico toxicology, which is the application of computer technologies to detect relationships that connect chemical structures and toxicological activities, is extremely appealing. In silico methods for predicting the toxicity of compounds include approaches such as machine learning, quantitative structure–activity relationship (QSAR), read-across, and structural alerts[6,7]. Particularly, the definition of toxicophores (structural alerts) corresponds to one of the most interesting approaches of in silico toxicology since it defines the key features of a molecule that are responsible for the initiation of a toxicological pathway[8]. The Tennant and Ashby's set of toxicophores is a well-known example of such toxic fragments for DNA reactivity[9]. This set has been largely extended by other researchers and to date, one of the most advanced lists for evaluating the mutagenic and carcinogenic potential of chemicals is the one proposed by Benigni and Bossa[10]. This list has been implemented as rules in knowledge-based expert systems which try to formalize the knowledge of human experts and the scientific literature, like Toxtree[11], Derek Nexus[12,13], and the OECD QSAR Toolbox[14]. We can also mention ToxAlerts, a web-based platform that collects toxicophores from the literature[15]. However, the expansion of such knowledge bases requires a strong investment of domain experts and a detailed analysis of the scientific literature. The evolution of artificial intelligence and data mining tools should answer these limitations, and particularly the time and efforts needed to identify new toxicophores[16–19].

In recent times, machine learning has become increasingly used in predictive toxicology[20–23]. A crucial step corresponds to the transformation of the chemical structures into features that can be processed by machine learning methods. Chemical fingerprinting is a method of simplifying the chemical representation of molecules by encoding properties or structural features of the molecules[24]. There are two main methodologies based on 2D representations for transforming chemical structures. Firstly, dictionary-based methods use a predefined set of fragments which have been identified a priori by domain experts and create fingerprints based on pattern matching of the structures to the "key" set. However, it is recognized that using a fixed number of predefined fragments when generating a fingerprint leads to information loss. The MACCS keys[25] and the PubChem fingerprints[26] are well-known examples of such dictionary-based fingerprints. Secondly, the fingerprints can be "learned" from the structures themselves using a data-driven methodology. Advances in collecting, combining, storing, and mining huge amounts of data efficiently have led to many data mining methods which are able to retrieve relevant information from these data. In this case, the overall efficiency of structure characterization is increased due to the much greater number of encoded fragments. Several types of data-driven fingerprints exist depending on the atom

[a] *Artificial Intelligence Lab, Vrije Universiteit Brussel Brussel, Belgium*

[b] *Department of Computer Science, KU Leuven Leuven, Belgium*

[c] *Groupe de Recherche en Informatique, Image, Automatique et Instrumentation de Caen, UNICAEN, ENSICAEN, CNRS - UMR GREYC, Normandie Univ Caen, France*

[d] *Flanders Make Lommel, Belgium*

[e] *Centre d'Etudes et de Recherche sur le Médicament de Normandie, UNICAEN, CERMN, Normandie Univ Caen, France*
**e-mail: alban.lepailleur@unicaen.fr*
phone/fax: +33231566822/+33231566803

[f] *INRIA Lille Nord Europe Lille, France*

abstraction method and the encoding rules, including atom pair-based, path-based, and circular techniques, with studies showing that techniques encoding information beyond simple linear paths outperformed other fingerprint methods[27].

In the present study, we applied Pairwise Maximum Common Subgraph Feature Generation (PMCSFG), an algorithm developed by our group to generate molecular features by computing maximum common substructures (MCS) under the block-and-bridge-preserving subgraph isomorphism of pairs of graph-encoded molecules[28]. MCS has many applications in drug discovery[29,30] including similarity searching[31], chemical space analysis[32,33], and activity cliffs detection[34]. As a case study, we used a publicly available benchmark dataset containing 6512 chemicals with known mutagenicity[35]. Both qualitative and quantitative methods were used in this investigation. First, the most informative PMCSFG patterns were selected based on point-wise mutual information and were compared to well-known toxicophores for mutagenicity. Second, we compared fingerprints generated by the PMCSFG algorithm with 12 state-of-the-art 2D chemical fingerprints by using them as features in several learning algorithms widely used in chemoinformatics: decision trees, k-nearest neighbors, rule learners, naïve Bayes, and support vector machines[36,37]. In order to validate the effectiveness of the models, we used cross-validation on the benchmark dataset and furthermore we report results on an external test set. The results show that PMCSFG has a similar performance as the best non-dictionary-based fingerprints, while using a tractable number of patterns.

2 Material and methods

2.1 Data

Benchmark dataset: Hansen. We used a publicly available benchmark dataset reported by Hansen et al.[35]. The dataset consists of 6512 compounds annotated with Ames mutagenicity and can be downloaded from http://doc.ml.tu-berlin.de/toxbenchmark/. Although the dataset was already pretreated to remove duplicate structures and inorganic molecules, we cleaned the chemical data to normalize specific chemotypes (e.g., nitro group, organophosphate moiety), to convert the structures to their aromatic form, and to add hydrogens on the heteroatoms. This resulted in a well-balanced dataset containing 3503 mutagenic and 3009 non-mutagenic compounds.

External test set. To assess the predictivity of the classification models, we used an external test set. We collected 1125 non-redundant molecules annotated with Ames mutagenicity data (447 mutagenic and 678 non-mutagenic compounds), to measure the classification accuracy of our models on unseen data. We curated the chemical structures in the same way as for the Hansen data set, and we omitted molecules when inconsistent mutagenicity data were reported.

2.2 Graph-theoretical concepts

This section gives the relevant definitions to understand how PMCSFG operates[28]. For an overview of graph theory, we refer to an introductory textbook[38].

PMCSFG represents molecules by graphs, with atoms corresponding to vertices and bonds to edges. A labeled **graph** is a quadruple $G(V,E,\Sigma,\lambda)$, with V a finite set of vertices and $E \subseteq \{\{u,v\} \mid u,v \in V\}$ a set of edges. Σ is a finite set of labels and $\lambda: V \cup E \to \Sigma$ is a function assigning a label to each element of $V \cup E$. The size of a graph is defined as the sum of the number of vertices and edges of the graph.

A sequence $x_0, x_1, \ldots, x_n$ of vertices is a **path** from x_0 to x_n if and only if $\{x_i, x_{i+1}\} \in E(G)$, for all $i \in [0, n-1]$. A **cycle** $x_0, \ldots, x_n$ is a path such that $x_0 = x_n$. A graph G is **connected** if there is a path between any pair of its vertices; it is **biconnected** if for any two vertices u and v of G, there is a simple cycle (without repeated vertices apart from the start and end vertex) containing u and v.

A graph is **planar** if it has a planar embedding, that is, it can be drawn in the plane in such a way that no two edges intersect except at a common vertex. The regions formed by the edges in a planar embedding are called **faces**. There is one unbounded face, which is called the outer face. A biconnected component or **block** of a graph G is a subgraph of G of maximal size that is biconnected. A **bridge** is an edge that does not belong to a block. An **outerplanar** graph is a planar graph that can be embedded in the plane in such a way that all of its vertices lie on the boundary of the outer face. An outerplanar graph consists entirely of blocks and bridges.

Figure 1(a) shows an example of a non-outerplanar graph in which there is one vertex (v), highlighted in red, that is not on the outside of the graph. The graphs in Figure 1(b) and Figure 1(c), however, are outerplanar. Note that only the upper graphs in (b) and (c) have one bridge which connects two blocks. From a chemical viewpoint, blocks correspond to ring structures while bridges are linear fragments of the molecule.

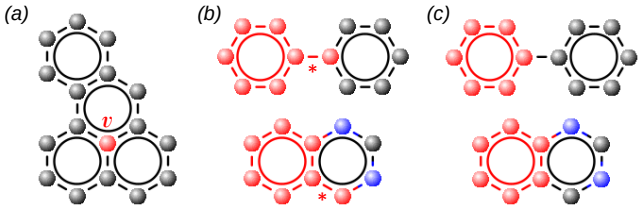


Figure 1. Examples of molecular graphs. (a) Example of a non-outerplanar graph, with the vertex that is not on the outside of the graph (v) highlighted in red. (b) A maximum common subgraph under the general subgraph isomorphism (MCS $_{\leq}$), highlighted in red. (c) A maximum common subgraph under the BBP subgraph isomorphism (MCS $_{\sqsubseteq}$), highlighted in red.

Let G and H be graphs. G is a **subgraph** of H, if (i) $V(G) \subseteq V(H)$, (ii) $E(G) \subseteq E(H)$, and (iii) $\lambda_G(x) = \lambda_H(x)$ holds for every $x \in V(G) \cup E(G)$. Two graphs G and H are **isomorphic** if there exists a bijection $\varphi: V(G) \to V(H)$ such that for every $u,v \in V(G)$ the following holds: (i) $\{u,v\} \in E(G)$ if and only if $\{\varphi(u),\varphi(v)\} \in E(H)$, (ii) $\lambda_G(u) = \lambda_H(\varphi(u))$, and (iii) if $\{u,v\} \in E(G)$ then $\lambda_G(\{u,v\}) = \lambda_H(\{\varphi(u),\varphi(v)\})$. A graph G is **subgraph isomorphic** to H,

denoted $G \preceq H$, if and only if G is isomorphic to a subgraph of H.

A **block-and-bridge-preserving (BBP) subgraph isomorphism** from G to H is a subgraph isomorphism from G to H, denoted $G \sqsubseteq H$, such that (i) $\{u,v\} \in E(G)$ is a bridge if and only if $\{\varphi(u),\varphi(v)\} \in E(H)$ is a bridge, and (ii) $\{u,v\} \in E(G)$ belongs to a block if and only if $\{\varphi(u),\varphi(v)\} \in E(H)$ belongs to a block. That is, BBP subgraph isomorphism is a special case of general subgraph isomorphism in which the constraint holds that bridges of G are only mapped to bridges of H and edges of blocks of G only to edges of blocks of H.

A **common connected subgraph** I of two graphs G and H is a connected graph such that $I \preceq G$ and $I \preceq H$; it is a **maximum common connected subgraph** when in addition there exists no other common subgraph J of G and H, such that $size(I) < size(J)$. From now on we call this an MCS $_{\preceq}$ (where $\preceq$ means that it is mined under the general subgraph isomorphism) and implicitly assume that it is always connected. In the same way, we define an MCS $_{\sqsubseteq}$. Interestingly, even though computing an MCS $_{\preceq}$ or an MCS $_{\sqsubseteq}$ between two general graphs is NP-hard[39], it is possible to compute an MCS $_{\sqsubseteq}$ between two outerplanar graphs in polynomial time[40].

Figure 1 shows a comparison between an MCS $_{\preceq}$ (b) and an MCS $_{\sqsubseteq}$ (c). In both examples, the MCS is highlighted in red. Note that one of the edges is a bridge in the upper graph of (b), while it belongs to a block in the lower graph (marked with a * in both graphs) and hence, it cannot be mapped under the BBP subgraph isomorphism. Chemically, it seems relevant not to map linear fragments to fragments that are part of a ring structure. This example shows that algorithms computing MCSs $_{\sqsubseteq}$ generate either smaller or equally large subgraphs than algorithms computing MCSs $_{\preceq}$.

For notational convenience, in the remainder of the text we will simply use MCS when we mean the MCS $_{\sqsubseteq}$.

2.3 Selection of the most informative MCS

We scored the features by their point-wise mutual information (PMI). The PMI expresses the correlation of a pattern m to a target variable t in the following way:

$$PMI(m,t) = log_{10}\frac{N \cdot p(m,t)}{p(m) \cdot p(t)}$$

where N denotes the number of molecules in the dataset, $p(m,t)$ the number of mutagenic molecules that have feature m, $p(m)$ denotes the number of molecules a pattern m occurs in and $p(t)$ are the number of mutagenic molecules. Then, we used the MMRFS algorithm to extract the top-50 features while maximizing PMI and minimizing redundancy between the features[41].

2.4 Chemical fingerprints

The features generated by fingerprint methods are used to encode each molecule in a dataset as a k-dimensional binary vector (with k the number of features), where a 1 is marked in the i-th position if the i-th feature occurs in the molecule and a 0 otherwise. In this study we only consider methods that directly produce binary vectors, which excludes e.g., graph neural networks, graph attention networks, and the neighborhood subgraph pairwise distance kernel. We divide fingerprinting methods into five categories, based on the way features are generated.

Dictionary-based fingerprints rely on features which have been identified *a priori* by domain experts as important fragments. **MACCS** keys[25] consist of 166 predefined structural keys which are considered as significant fragments for bioactivity of chemicals. **PubChem** FP[26] have 883 features corresponding to PubChem substructures.

The remaining fingerprint types conceptually encode fragments based on the atom-bond structures in the dataset. **Path-based fingerprints** can enumerate all paths up to a certain length. **Dendritic** encodes the linear paths augmented with intersections of linear paths, with a maximum of five bonds per path to encode branched features[27]. **Torsion** encodes features of four consecutively bonded non-hydrogen atoms along with the number of non-hydrogen branches, corresponding to a torsion angle[42].

Radial-based fingerprints iteratively encode features that represent each heavy atom in larger and larger structural neighborhoods, up to a given diameter (2, 4, and 6 in this study). Extended-connectivity fingerprints[43] are generated directly from the dataset by first assigning an initial label to each atom and then applying a Morgan type algorithm[44], which was proposed as a method for solving the molecular isomorphism problem. In brief, an iterative process is used to generate features that represent each atom in larger and larger structural neighborhoods. After each iteration, the new feature codes for the atoms are added to the set of features from all previous steps. When the maximum diameter of the neighborhoods is reached, the process is complete, and the set of all features is returned as the fingerprint. A number of methods are available to define the atom abstraction used to generate the initial atom feature codes for the heavy (non-hydrogen) atoms in the molecule. The functional class connectivity fingerprints **(FCFP)** consist of a combination of a hydrogen-bond acceptor, hydrogen-bond donor, positively ionized or positively ionizable, negatively ionized or negatively ionizable, aromatic, and halogen. A variant called the “atom type” fingerprints **(ECFP)** uses a code derived from the number of connections to an atom, the element type, the charge, and the atomic mass. In another approach, **MOLPRINT2D**[45], each heavy atom in a structure is characterized by an environment that consists of all other heavy atoms within a distance of two bonds. Each member of the list is encoded into a string of the form Type-freq(Type)-d, where freq(Type) is the number of times a given atom type is found at a distance d from the central atom. The atom-typing scheme used is the Sybyl Mol2.

Atom pair-based fingerprints encode features representing two atoms and their corresponding distance. Specifically, we use here the **Pairwise** approach[46] which considers the Carhart atom types and the topological distance separating them.

The Pairwise Maximum Common Subgraph Feature Generation **(PMCSFG)** algorithm generates features by computing MCSs under the block-and-bridge-preserving

subgraph isomorphism between molecules from a graph-based dataset[28]. For efficiency reasons, the algorithm computes MCSs only from outerplanar graphs and it returns only one if there are multiple MCSs. It either does this exhaustively (an MCS is computed for every pair of examples) or randomly (pairs of examples are selected at random). When sampling random MCSs from data, we showed that the MCSs tend to retain a comfortable coverage of the entire data set because a frequent MCS has a higher chance to be selected by the random sampling[28]. The algorithm computing an MCS between two outerplanar graphs is based on a dynamic programming strategy that makes use of efficient matching algorithms[40]. The algorithm can be downloaded from https://dtai.cs.kuleuven.be/research/pmcsfg.

2.5 Experimental methodology

The features are generated as follows. Given a dataset G, we first generate features only from the training set. Then, we propositionalize each example in G to a one-bit vector encoding representation: given a feature set of size k, each graph $g \in G$ is encoded as a k-dimensional binary vector, where a 1 is marked in the i-th position if the i-th subgraph is subgraph isomorphic to g. Note that we use the general subgraph isomorphism here to embed the patterns, in order to ensure the same treatment of all fingerprints.

For PMCSFG, we selected 4000 unique MCSs by randomly sampling pairs of molecules. Sampling more than 4000 MCSs did not improve performance in an earlier study[28]. For the other fingerprints, there are no other parameters to be set.

In order to compare the different fingerprint methods, we used them as features in five different machine learning methods from the Weka[47] data mining tool:

Support vector machines (SVM) combined with the Tanimoto kernel. Tanimoto kernel computes a similarity between vector x and vector y by counting the number of common patterns (i.e., the set-intersection) between two molecules as a fraction of the total number of patterns that occurs in both molecules (i.e., the set-union)[48]:

$$K_T(x,y) = \frac{\sum_{i=1}^{N}(x_i = 1 \wedge y_i = 1)}{\sum_{i=1}^{N}(x_i = 1 \vee y_i = 1) - \sum_{i=1}^{N}(x_i = 1 \wedge y_i = 1)}$$

The Tanimoto-kernel is considered state-of-the-art for the classification of small molecules[24]. As implementation we used SVM ^{light}[49].

Decision trees. We use the J48 implementation from Weka with standard parameters. It builds a pruned decision tree according to the C4.5 algorithm[50].

k-nearest neighbors classifier. We use the standard Weka implementation with k=10[51].

Rule learner. We use the Repeated Incremental Pruning to Produce Error Reduction (RIPPER) algorithm[52] (as implemented in Weka) with standard parameters.

Naïve Bayes. We use the Naïve Bayes classifier as implemented in Weka[53].

The learning task is to discriminate the harmless molecules from the mutagenic ones in the Hansen dataset. To evaluate the classification models, we use the area under the ROC curve (AUROC) score[54]. For all experiments, a stratified 10-fold cross-validation is used. For all methods we used standard parameters, except for SVMs, where we tuned the regularization parameter out of 10 possible values through an internal 5-fold cross-validation on the training set.

3 Results and discussion

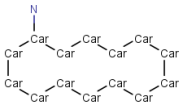
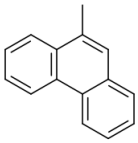
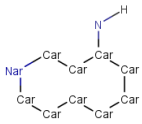
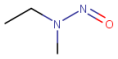
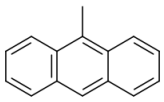
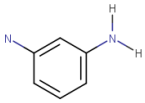
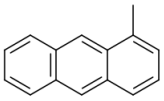
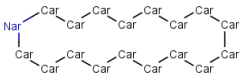
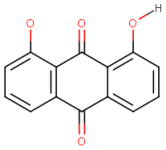
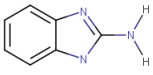
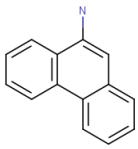
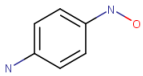
3.1 Qualitative analysis

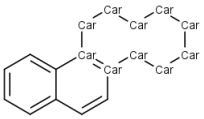
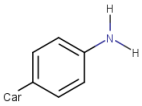
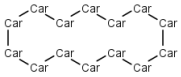
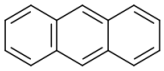
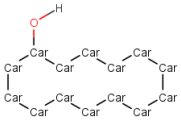
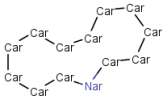
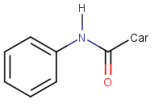
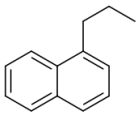
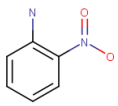
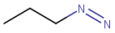
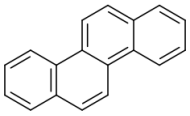
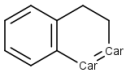
Testing of chemicals for mutagenicity in *S. typhimurium* is based on the knowledge that a substance that is mutagenic in the bacterium is likely to be a carcinogen in laboratory animals, and thus, by extension, present a risk of cancer to humans[55]. Toxicophores are very helpful not only for the classification of potential carcinogens, but also to understand the mechanisms of mutagenicity.

To be effective, a carcinogen must interact with cellular macromolecules such as DNA, RNA, or proteins. Compounds that have structures permitting these types of reactions are direct-acting carcinogens; those that require metabolic activation are indirect-acting carcinogens. In their ultimate form, direct- or indirect-acting carcinogens work as reactive electrophiles[56]. Relatively few carcinogens are direct-acting since the high reactivity of such compounds tends to make them unstable. Well-known examples of such carcinogens are epoxides, imines, nitrogen mustards, and sulphate esters. Indirect-acting carcinogens, often called procarcinogens, constitute those that are stable in the environment and thus are more likely to be in contact of the population. The initial metabolic reaction for most carcinogens involves oxidation to a form that is closer to the activated carcinogen. Typical indirect carcinogens are polycyclic aromatic hydrocarbons, nitrosamines, nitrosoureas, and aromatic amines. Finally, when we are talking about chemical carcinogenesis we cannot skip oxidative stress caused by reactive oxygen/nitrogen species[57]. Most of them are generated by redox cycling induced by chemical carcinogens that contain structural alerts such as halogenated compounds, aromatic hydrocarbons, aromatic N-oxides, quinones, aromatic nitro compounds, conjugated imines, heterocyclic amines, and pyridyl compounds[58].

Some of the top-ranked MCSs perfectly match known structural alerts for mutagenicity. As shown in Table 1, we retrieve nitro-aromatic (410) and dinitro-aromatic groups (344), aromatic amine (2498), alkyl N-nitroso (358) and aryl N-nitroso groups (3552), aromatic hydroxylamine (762) and their derived esters (638), aromatic azo group (563), hydrazine (1539), alkyl ester of sulfonic acid (3945), acyl halide (2121) haloalkyl ether (2506), haloethyl amine (2131), polycyclic aromatic hydrocarbons (1010), quinone (1847), and nitrogen mustard and its derivatives including cyclophosphamide (1675). More specific structural alerts known or considered to be carcinogen are also overrepresented in the Hansen dataset. They correspond to the 2-aminobenzimidazole (2102), the dantron moiety (1028), and the substructure of the

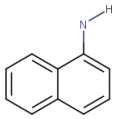
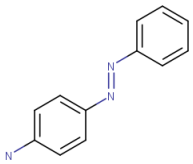
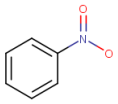
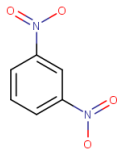
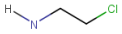
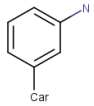
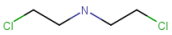
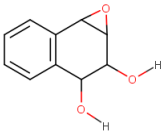
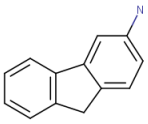
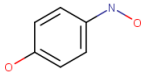
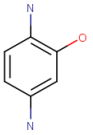
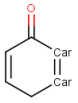
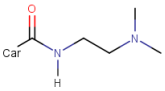
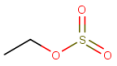
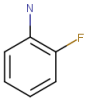
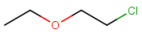
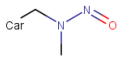
Table 1. The 50 top-rank MCSs scored by their point-wise mutual information (PMI) and selected using the MMRFS algorithm.

ID	MCS	PMI[a]	ACC[b]	Freq[c]
1713		0.247	0.950	101
2799		0.244	0.943	88
966		0.242	0.939	82
1513		0.233	0.920	75
358		0.232	0.918	147
406		0.232	0.917	85
86		0.231	0.916	83
3185		0.229	0.911	56
1669		0.222	0.896	48
1028		0.220	0.894	47
2102		0.218	0.889	45
594		0.218	0.889	45
638		0.217	0.887	53
751		0.214	0.880	108

ID	MCS	PMI[a]	ACC[b]	Freq[c]
1141		0.213	0.878	41
762		0.211	0.875	48
2498		0.211	0.875	48
383		0.211	0.874	530
1010		0.209	0.872	226
1855		0.209	0.871	62
3940		0.208	0.868	38
2634		0.206	0.865	52
1189		0.202	0.857	35
2741		0.202	0.857	91
585		0.201	0.855	62
1140		0.201	0.854	48
269		0.198	0.848	106
2894		0.197	0.846	78

[a] PMI: point-wise mutual information.
[b] ACC: Accuracy rate.
[c] Freq: Frequency of occurrence of the MCS in the Hansen data set.

Table 1. Continued.

ID	MCS	PMI[a]	ACC[b]	Freq[c]	ID	MCS	PMI[a]	ACC[b]	Freq[c]
1435		0.196	0.844	45	**563**		0.183	0.821	78
410		0.196	0.844	122	**344**		0.182	0.818	88
2131		0.195	0.844	32	**1252**		0.181	0.816	38
1675		0.195	0.842	38	**1272**		0.180	0.815	27
7		0.193	0.839	1237	**1539**		0.177	0.808	26
562		0.188	0.830	35	**921**		0.175	0.805	41
333		0.187	0.828	64	**2107**		0.170	0.795	39
2121		0.187	0.828	29	**1796**		0.169	0.793	29
1847		0.187	0.828	29	**1421**		0.168	0.792	24
3945		0.187	0.828	29	**402**		0.161	0.760	24
2506		0.184	0.821	28	**3552**		0.161	0.760	24

[a] PMI: point-wise mutual information.
[b] ACC: Accuracy rate.
[c] Freq: Frequency of occurrence of the MCS in the Hansen data set.

benzo[a]pyrene-7,8-dihydrodiol-9,10-epoxide (562); the latter being the metabolite of benzo[a]pyrene responsible for the disruption of the normal process of copying DNA by intercalating its double-helical structure.

Other MCSs summarize several structural alerts in only one “general” toxicophore. The generation of such generalized toxicophore is an inner property of the MCS approach. In some cases, the MCS summarizes only two structural alerts like 3940 which corresponds to the nitrosourea and the semicarbazone groups, but for others the matching is fuzzier. For example, nitroso, nitrosamine, nitrite, aromatic nitro, and N-nitro groups are generalized by 7. Another example is 1140 which generalizes the azo, azoxy, azide, and triazene groups. Sometimes, a molecular fragment is not easily interpretable because of the presence of atoms which are not really involved in the structural alert despite their presence in all the supporting molecules. As an example, the analysis of the extent of 1272 showed that the hazardous moiety of this molecular fragment is not related to the sulfur atom but to the chlorine which is part of trichloromethyl or di/trichlorovinyl substituents. Concerning 1435, at a first glance it can be viewed as an aromatic amine but it can also be associated to the same structural alerts as 333 and 751, i.e. an aromatic nitro, an aromatic hydroxylamine and its derived esters, and an aromatic N-acyl amine. Numerous MCSs display a nitrogen related to an (hetero)aromatic ring (585, 86, 1796, 402, 2799, 2107, 1252, 594, 921). This nitrogen can be part of a nitro, a nitroso, an amine, an N-acyl amine,

a hydroxylamine (ester), a hydrazine or an azo(xy) group. To complete the list of the generalized toxicophores, a polycyclic (aromatic) hydrocarbon system is often the central core of mutagenic compounds (406, 3185, 966, 2741, 2894, 269). This system can be substituted with various substituents such as nitro, halogen, epoxyde, aziridine, hydroxyle, ester, and carboxylic or sulfuric acid.

Some MCSs correspond to outlines of polycyclic aromatic systems which can be substituted or not. These outlines are obtained because the molecules are mined under the block-and-bridge-preserving (BBP) subgraph isomorphism. Although these MCSs are ambiguous chemically speaking due to the absence of some ring closures, they are of interest since they allow to cover analogous polycyclic aromatic systems as exemplified by MCS 1669 on Figure 2.

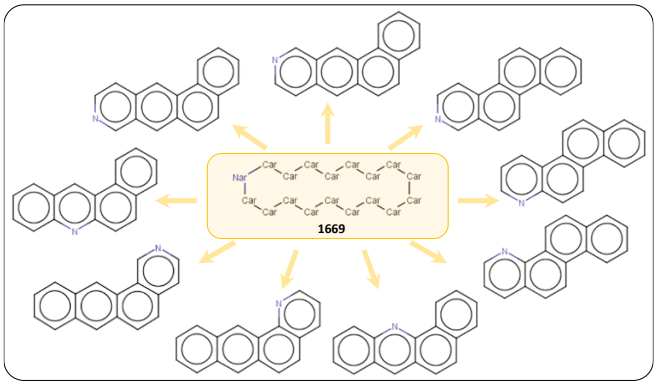


Figure 2. Example of an MCS able to cover multiple analogous polycyclic aromatic systems.

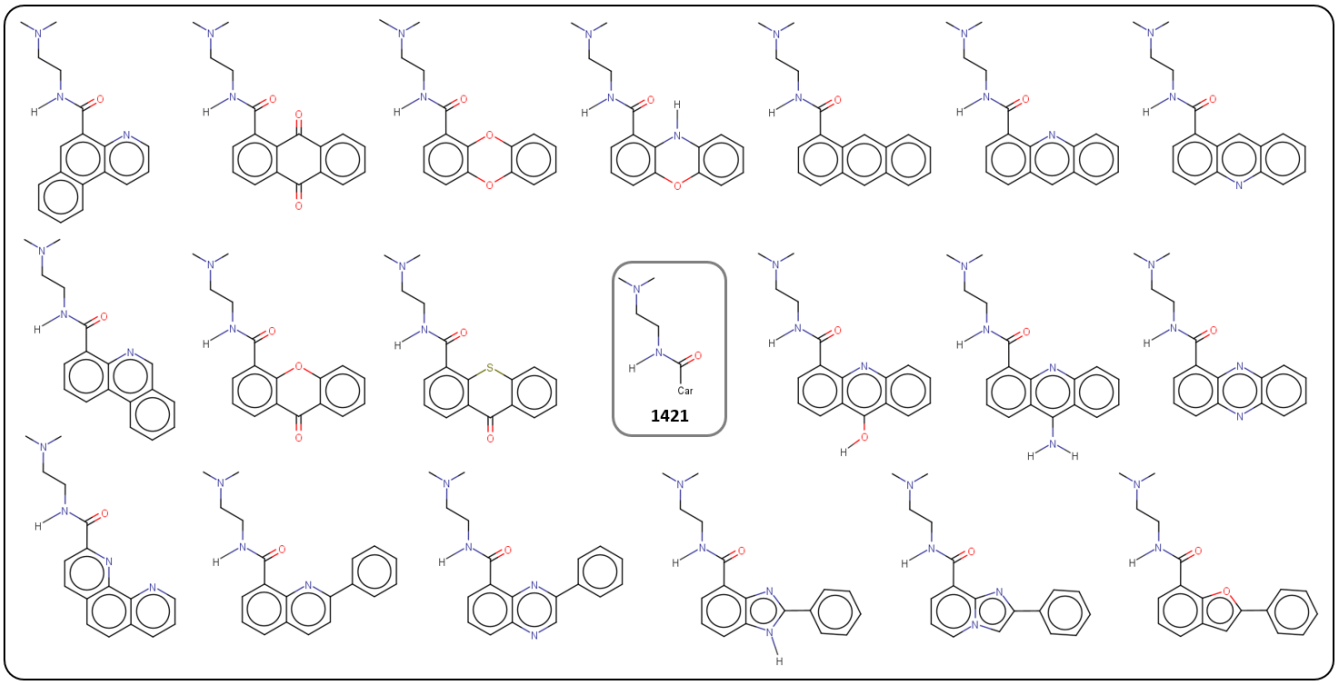


Figure 3. Mutagenic compounds covered by MCS 1421.

Finally, two MCSs can be considered as undefined (1189, 1421). The analysis of the extension of 1189 showed this MCS mainly encompasses nitro derivatives of arylamide. Since the nature of the aryl moiety can vary (thiophene, benzene, pyridine), the resulting MCS only indicates the presence of an aromatic ring without information about the nitro substituent which is, however, definitely of importance. When applied outside the learning dataset, MCS 1189 is able to match compounds without nitro substituent on the arylamide substructure. This highlights a weakness of our method since many molecules not reported as mutagens exhibit such substructures. As examples, we can cite the biocides Tibromsalan (bactericide), Boscalid (fungicide), and Picolinafen (herbicide). The other undefined MCS (1421) is more of interest because this unambigous substructure is not reported on ToxAlerts[15], an open expert-knowledge-based platform that contains more than 3000 toxicophores from the literature for several endpoints including mutagenicity (117). Nineteen out of the 24 supporting molecules exhibiting this MCS belong to the mutagenic class (Figure 3). The analysis of the supporting molecules showed that the vertex labeled C_{ar} is always part of a tricyclic (aromatic) system. It should be noted that some of these systems exhibit substructures discussed above and already considered as structural alerts. By the way, experimental validation is required before classifying MCS 1421 by itself as a new structural alert for the mutagenicity endpoint.

3.2 Quantitative analysis

From the 4000 MCSs generated using the PMCSFG algorithm, we constructed molecular fingerprints and used them as features in machine learning models. Table 2 shows the results of the comparison with 12 state-of-the-art 2D chemical fingerprints on the Hansen dataset.

Table 2. Area under the ROC curve (AUC) under 10-fold cross-validation for the different fingerprint methods and learning algorithms on the Hansen dataset.

Fingerprint method	#Patterns[a]	DT[b]	k-NN[b]	NB[b]	RBL[b]	SVM[b]	Average
Dictionary-based							
MACCS keys	168	0.80	0.85	0.73	0.77	0.88	0.81±0.06
PubChem FP	883	0.80	0.84	0.71	0.76	0.88	0.80±0.07
Path-based							
Dendritic	112853	0.80	0.80	0.74	0.70	0.88	0.79±0.07
Torsion	5611	0.79	0.81	0.74	0.69	0.87	0.78±0.07
Radial-based							
ECFP_2	3701	0.80	0.85	0.74	0.76	0.88	0.81±0.06
ECFP_4	21608	0.81	0.85	0.73	0.75	0.89	0.81±0.07
ECFP_6	51308	0.80	0.85	0.73	0.76	0.89	0.81±0.06
FCFP_2	3445	0.81	0.85	0.74	0.76	0.88	0.81±0.06
FCFP_4	20832	0.81	0.85	0.73	0.76	0.89	0.81±0.06
FCFP_6	50223	0.80	0.85	0.73	0.76	0.89	0.81±0.06
MOLPRINT2D	8454	0.80	0.79	0.76	0.66	0.88	0.78±0.08
Atom pair-based							
Pairwise	11140	0.77	0.82	0.73	0.71	0.88	0.78±0.07
Data mining							
PMCFSG	4000	0.81	0.85	0.76	0.73	0.88	0.81±0.06
Average		0.80±0.01	0.83±0.02	0.74±0.01	0.74±0.04	0.88±0.01	

[a] Number of patterns generated from the Hansen dataset.
[b] DT: decision trees; k-NN: *k-nearest neighbors*; NB: Naïve Bayes; RBL: rule-based learning; SVM: support vector machines.

Table 3. Area under the ROC curve (AUC) for the different fingerprint methods on the external test set. The learning model is SVM.

Fingerprint method	#Patterns[a]	AUC	Avg. freq.[b]
Dictionary-based			
MACCS keys	168	0.85	237.77
PubChem FP	883	0.84	126.34
Path-based			
Dendritic	112853	0.83	1.70
Torsion	5611	0.78	4.12
Radial-based			
ECFP_2	3701	0.84	26.15
ECFP_4	21608	0.84	9.79
ECFP_6	51308	0.84	5.52
FCFP_2	3445	0.84	27.64
FCFP_4	20832	0.840	10.0
FCFP_6	50223	0.84	5.56
MOLPRINT2D	8454	0.82	1.70
Atom pair-based method			
Pairwise	11140	0.81	12.15
Data mining			
PMCFSG	4000	0.85	17.84

[a] Number of patterns generated from the Hansen dataset.
[b] Average of the number of external test set molecules a pattern occurs in.

The different fingerprints perform similarly, with slightly higher results obtained by the dictionary-based fingerprints (MACCS keys and PubChem FP), the ECFP/FCFP fingerprints and the PMCSFG fingerprints, although the differences are not statistically significant (assuming a normal distribution and requiring at least 3 standard deviations of difference to conclude that one method significantly outperforms another). Interestingly, the fingerprints with the most features do not necessarily have the best performance. It should be noted that our MCSs were extracted by randomly sampling 4000 pairs of molecules since we showed in an earlier study that sampling more than 4000 MCSs did not improve performance.[28] Therefore, the computation was really fast, only lasting a couple of minutes. The average performance of the different learning methods varies from 0.74 to 0.88. Quantitatively, *whatever the molecular description is*, SVM appears to be the best performing learning method (which is statistically significant) while rule-based learning and naïve Bayes lead to the poorest results. Moreover, there is a substantial difference between the average score of the SVMs and the second best learner, k-NN.

Table 3 shows the results of the SVM method on the external test set. Using SVMs as learning methods (which performed the best according to Table 2), PMCSFG obtains the best AUROC on the test set, closely followed by MACCS keys. Furthermore, the average frequency of the patterns, i.e. the number of external test set molecules it occurs in, is shown. Having high-frequency

patterns seems to moderately correlate with the performance of the fingerprints.

3.3 Comments with respect to the state-of-the-art

Many (Q)SAR models and structural alerts are available to assess the possible mutagenicity of substances[59–61]. In this paper, we used the data set reported by Hansen which is the most popular benchmark data set for chemical Ames mutagenicity[35]. It should be noted that the comparisons of the available models reported in the literature have sometimes been performed on subsets or supersets of the Hansen data set. A trend can nevertheless be observed: the predictive power of (Q)SAR models is high, often achieving AUC around 0.8-0.9, and our methodology is not an exception to the rule. For recent comparative studies on this topic the reader is referred to Benigni *et al.*[62] and Yang *et al.*[63]

Regarding the identification of structural alerts, a dominant paradigm is based on the idea that a substructure is of lesser interest when it relies on too few occurrences in the data set. Therefore, most of the detecting methods regarded the accuracy rate as the most important index to assess the hazard of a substructure. The accuracy rate compares the number of mutagenic compounds that contain a certain substructure and the number of all compounds that contain the substructure. Other metrics have been used including positive rate, likelihood ratio[61], or information gain[63]. As for us, we ranked the potential structural alerts according to their point-wise mutual information (PMI), a measure which compares the probability of two events occurring together (here, the occurrence of an MCS and the detection of mutagenicity hazard) to what this probability would be if they were independent.

In their publication, Yang *et al.*[63] emphasized that current methods for structural alerts might identify redundant and overspecific substructures. Since redundant and non-discriminative substructures often overfit the model and deteriorate the classification accuracy, we tackled this issue by using the MMRFS algorithm[41]. The MMRFS algorithm searches over the feature space in a heuristic way. A feature is selected if it is relevant to the class label and contains very low redundancy to the features already selected. In the present case, this feature selection algorithm tended to output a subset of MCSs whose elements were discriminative, different, and representative of the mutagenic compounds. The method demonstrated its efficiency as the 50 top-ranked MCSs matched around 40 known structural alerts for mutagenicity. In addition, due to their intrinsic properties it turned out that the MCS features were typically much larger than the ones generated with the other methodologies (9 atoms on average and up to 28 atoms for the Hansen dataset) and corresponded to recognizable molecular fragments.

In order to have a complete picture of the results obtained in this study we also report here the results obtained in the same dataset by using learning representation techniques; namely graph neural networks. Results reported in Li *et al.*[64] show that employing a graph neural network achieves an AUC of 0.878. These results are to be expected since learning features in an end-to-end fashion conditioned on the downstream task is always more beneficial than handcrafted features. However, this does no invalidate the results we obtained in this study. One limitation of using learning representation techniques is explainability and interpretability of the representations. A major part of machine learning models is to be able to inform domain experts regarding the reasons behind model decisions. Deep learning techniques are still at their infancy with respect to explainability even if some methods exist as a means of understanding the underlying phenomena including Local Interpretable Model-agnostic Explanations (LIME)[65], Deep Learning Important FeaTures (DeepLIFT)[66], and Shapley Additive exPlanations (SHAP)[67].

3 Conclusion

In this paper, we compared 12 traditional chemical fingerprints to the PMCSFG fingerprint method, which learns fingerprints based on sampling maximum common subgraphs from a molecular dataset. Although we found no significant differences in predictive performance between the different fingerprints, PMCSFG has multiple advantages. First, they can be automatically and efficiently learned from data, unlike the dictionary-based fingerprints which are selected by hand. Second, compared to the non-dictionary-based fingerprints, the number of features can be easily controlled. Third, PMCSFG reaches the same predictive performance as the other fingerprints with a smaller number of features (except for the manually designed MACCS fingerprints). It turns out that the MCS features are typically much larger than the ones in the former fingerprints and correspond to recognizable molecular fragments. This makes it easier to give an interpretation to the features. For the Hansen dataset, our process has recovered over 40 known structural alerts; moreover, it has been able to propose one additional structural alert.

In further work, we plan to investigate the exact properties of MCS fingerprints which are responsible for the state-of-the-art performance and we also plan to apply the PMCSFG fingerprints to other prediction tasks.

Acknowledgements

This work has been achieved thanks to the financial support of ERC Starting Grant 240186 “MiGraNT”, Research Fund KU Leuven, IWT (SBO Nemoa, SBO InSPECtor) and Région Normandie.

References

[1] S. Festing, R. Wilkinson, *EMBO Rep.* **2007**, *8*, 526–530.
[2] O. E. Varga, A. K. Hansen, P. Sandøe, I. A. S. Olsson, *Altern. Lab. Anim.* **2010**, *38*, 245–248.
[3] M. N. H. Khabib, Y. Sivasanku, H. B. Lee, S. Kumar, C. S. Kue, *Toxicology* **2022**, *465*, 153053.
[4] N. Burden, C. Mahony, B. P. Müller, C. Terry, C. Westmoreland, I. Kimber, *Toxicology* **2015**, *330*, 62–66.

[5] K. L. Chapman, H. Holzgrefe, L. E. Black, M. Brown, G. Chellman, C. Copeman, J. Couch, S. Creton, S. Gehen, A. Hoberman, L. B. Kinter, S. Madden, C. Mattis, H. A. Stemple, S. Wilson, *Regul. Toxicol. Pharmacol.* **2013**, *66*, 88–103.
[6] H. Raunio, *Front. Pharmacol.* **2011**, *2*.
[7] A. B. Raies, V. B. Bajic, *WIREs Comput. Mol. Sci.* **2016**, *6*, 147–172.
[8] P. K. Singh, A. Negi, P. K. Gupta, M. Chauhan, R. Kumar, *Arch. Toxicol.* **2016**, *90*, 1785–1802.
[9] J. Ashby, R. W. Tennant, *Mutat. Res.* **1991**, *257*, 229–306.
[10] R. Benigni, C. Bossa, *Chem. Rev.* **2011**, *111*, 2507–2536.
[11] R. Benigni, *JRC Sci. Tech. Rep.* **2008**, *EUR 23241*, 1–70.
[12] J. E. Ridings, M. D. Barratt, R. Cary, C. G. Earnshaw, C. E. Eggington, M. K. Ellis, P. N. Judson, J. J. Langowski, C. A. Marchant, M. P. Payne, W. P. Watson, T. D. Yih, *Toxicology* **1996**, *106*, 267–279.
[13] D. M. Sanderson, C. G. Earnshaw, *Hum. Exp. Toxicol.* **1991**, *10*, 261–273.
[14] K. van Leeuwen, T. W. Schultz, T. Henry, B. Diderich, G. D. Veith, *SAR QSAR Environ. Res.* **2009**, *20*, 207–20.
[15] I. Sushko, E. Salmina, V. A. Potemkin, G. Poda, I. V. Tetko, *J. Chem. Inf. Model.* **2012**, *52*, 2310–2316.
[16] A. Lepailleur, G. Poezevara, R. Bureau, *Comput. Struct. Biotechnol. J.* **2013**, *5*, e201302013.
[17] M. Floris, G. Raitano, R. Medda, E. Benfenati, *Mol. Inform.* **2017**, *36*, 1600133.
[18] J. Rabatel, T. Fannes, A. Lepailleur, J. Le Goff, B. Crémilleux, J. Ramon, R. Bureau, B. Cuissart, *Mol. Inform.* **2017**, *36*, 1700022.
[19] J.-P. Métivier, A. Lepailleur, A. Buzmakov, G. Poezevara, B. Crémilleux, S. O. Kuznetsov, J. L. Goff, A. Napoli, R. Bureau, B. Cuissart, *J. Chem. Inf. Model.* **2015**, *55*, 925–940.
[20] I. I. Baskin, in *Comput. Toxicol. Methods Protoc.* (Ed.: O. Nicolotti), Springer, New York, NY, **2018**, pp. 119–139.
[21] M. W. H. Wang, J. M. Goodman, T. E. H. Allen, *Chem. Res. Toxicol.* **2021**, *34*, 217–239.
[22] H. Yang, L. Sun, W. Li, G. Liu, Y. Tang, *Front. Chem.* **2018**, *6*.
[23] L. Zhang, H. Zhang, H. Ai, H. Hu, S. Li, J. Zhao, H. Liu, *Curr. Top. Med. Chem.* **2018**, *18*, 987–997.
[24] P. Willett, *Drug Discov. Today* **2006**, *11*, 1046–1053.
[25] J. L. Durant, B. A. Leland, D. R. Henry, J. G. Nourse, *J. Chem. Inf. Comput. Sci.* **2002**, *42*, 1273–1280.
[26] *PubChem Subgraph Fingerprint*, National Center For Biotechnology Information, Bethesda, MD, **2009**.
[27] M. Sastry, J. F. Lowrie, S. L. Dixon, W. Sherman, *J. Chem. Inf. Model.* **2010**, *50*, 771–784.
[28] L. Schietgat, F. Costa, J. Ramon, L. De Raedt, *Mach. Learn.* **2011**, *83*, 137–161.
[29] H.-C. Ehrlich, M. Rarey, *WIREs Comput. Mol. Sci.* **2011**, *1*, 68–79.
[30] E. Duesbury, J. Holliday, P. Willett, *ChemMedChem* **2018**, *13*, 588–598.
[31] J. W. Raymond, P. Willett, *J. Comput. Aided Mol. Des.* **2002**, *16*, 521–533.
[32] R. Schmidt, R. Klein, M. Rarey, *J. Chem. Inf. Model.* **2022**, *62*, 2133–2150.
[33] B. Zhang, M. Vogt, G. M. Maggiora, J. Bajorath, *J. Comput. Aided Mol. Des.* **2015**, *29*, 937–950.
[34] J. Jiménez-Luna, M. Skalic, N. Weskamp, *J. Chem. Inf. Model.* **2022**, *62*, 274–283.
[35] K. Hansen, S. Mika, T. Schroeter, A. Sutter, A. ter Laak, T. Steger-Hartmann, N. Heinrich, K.-R. Müller, *J. Chem. Inf. Model.* **2009**, *49*, 2077–2081.
[36] A. Varnek, I. Baskin, *J. Chem. Inf. Model.* **2012**, *52*, 1413–1437.
[37] Y.-C. Lo, S. E. Rensi, W. Torng, R. B. Altman, *Drug Discov. Today* **2018**, *23*, 1538–1546.
[38] Diestel, *Graph Theory*, Springer, New York, NY, **2017**.
[39] M. R. Garey, D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, New York, **1979**.
[40] L. Schietgat, J. Ramon, M. Bruynooghe, *Ann. Math. Artif. Intell.* **2013**, *69*, 343–376.
[41] H. Cheng, X. Yan, J. Han, C.-W. Hsu, in *2007 IEEE 23rd Int. Conf. Data Eng.*, **2007**, pp. 716–725.
[42] R. Nilakantan, N. Bauman, J. S. Dixon, R. Venkataraghavan, *J. Chem. Inf. Comput. Sci.* **1987**, *27*, 82–85.
[43] D. Rogers, M. Hahn, *J. Chem. Inf. Model.* **2010**, *50*, 742–754.
[44] H. L. Morgan, *J. Chem. Doc.* **1965**, *5*, 107–113.
[45] A. Bender, H. Y. Mussa, R. C. Glen, S. Reiling, *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1708–1718.
[46] R. E. Carhart, D. H. Smith, R. Venkataraghavan, *J. Chem. Inf. Comput. Sci.* **1985**, *25*, 64–73.
[47] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, *ACM SIGKDD Explor. Newsl.* **2009**, *11*, 10–18.
[48] S. J. Swamidass, J. Chen, J. Bruand, P. Phung, L. Ralaivola, P. Baldi, *Bioinformatics* **2005**, *21*, i359–i368.
[49] T. Joachims, *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*, Kluwer Academic Publishers, USA, **2002**.
[50] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, Calif, **1992**.
[51] D. W. Aha, D. Kibler, M. K. Albert, *Mach. Learn.* **1991**, *6*, 37–66.
[52] W. W. Cohen, in *Mach. Learn. Proc. 1995* (Eds.: A. Prieditis, S. Russell), Morgan Kaufmann, San Francisco (CA), **1995**, pp. 115–123.
[53] G. H. John, P. Langley, in *Proc. Elev. Conf. Uncertain. Artif. Intell.*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, **1995**, pp. 338–345.
[54] T. Fawcett, *Pattern Recognit. Lett.* **2006**, *27*, 861–874.
[55] B. N. Ames, F. D. Lee, W. E. Durston, *Proc. Natl. Acad. Sci. U. S. A.* **1973**, *70*, 782–786.
[56] E. C. Miller, J. A. Miller, *Cancer* **1981**, *47*, 2327–2345.
[57] M. Kulis, M. Esteller, *Adv. Genet.* **2010**, *70*, 27–56.
[58] J. E. Klaunig, Z. Wang, X. Pu, S. Zhou, *Toxicol. Appl. Pharmacol.* **2011**, *254*, 86–99.
[59] C. Bossa, R. Benigni, O. Tcheremenskaia, C. L. Battistelli, in *Comput. Toxicol. Methods Protoc.* (Ed.: O. Nicolotti), Springer, New York, NY, **2018**, pp. 447–473.
[60] R. Benigni, C. Bossa, *Mutat. Res.* **2008**, *659*, 248–261.
[61] T. Ferrari, D. Cattaneo, G. Gini, N. Golbamaki Bakhtyari, A. Manganaro, E. Benfenati, *SAR QSAR Environ. Res.* **2013**, *24*, 365–383.
[62] E. Benfenati, A. Golbamaki, G. Raitano, A. Roncaglioni, S. Manganelli, F. Lemke, U. Norinder, E. Lo Piparo, M. Honma, A. Manganaro, G. Gini, *SAR QSAR Environ. Res.* **2018**, *29*, 591–611.
[63] H. Yang, J. Li, Z. Wu, W. Li, G. Liu, Y. Tang, *Chem. Res. Toxicol.* **2017**, *30*, 1355–1364.
[64] S. Li, L. Zhang, H. Feng, J. Meng, D. Xie, L. Yi, I. T. Arkin, H. Liu, *Interdiscip. Sci. Comput. Life Sci.* **2021**, *13*, 25–33.
[65] M. T. Ribeiro, S. Singh, C. Guestrin, in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Association For Computing Machinery, New York, NY, USA, **2016**, pp. 1135–1144.
[66] A. Shrikumar, P. Greenside, A. Kundaje, in *Proc. 34th Int. Conf. Mach. Learn. - Vol. 70*, JMLR.Org, Sydney, NSW, Australia, **2017**, pp. 3145–3153.
[67] S. M. Lundberg, S.-I. Lee, in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Curran Associates Inc., Red Hook, NY, USA, **2017**, pp. 4768–4777.