J Comput Aided Mol Des (2013) 27:185–201
DOI 10.1007/s10822-013-9637-7

QSAR model based on weighted MCS trees approach for the representation of molecule data sets

Bernardo Palacios-Bejarano · Gonzalo Cerruela García · Irene Luque Ruiz · Miguel Ángel Gómez-Nieto

Received: 12 October 2012 / Accepted: 1 February 2013 / Published online: 6 February 2013

Abstract In this paper we propose a new method for the generation of 2D-QSAR models for the prediction of activity values of chemicals. Maximum common substructures which are extracted from the data set are used for molecule classification in a tree, where the node of the tree represents molecules or common structures to groups of molecules and the arcs of the tree represent non isomorphic substructures between two nodes of the tree. All paths between pairwise leaf nodes are used to represent the equation system used as representational space in the building of the QSAR model. The proposed model, which is based on the combining of non isomorphic structures, use of molecular descriptors for the calculation of path lengths and classification of the data set based on maximum common substructures, considerably improves the generation of QSAR models with regard to the classical model based only on the use of a set of molecular descriptors. Optimization algorithms based on genetic algorithm and differential evolution approximations have also been used, resulting in the improvement and refinement of the equations obtained.

B. Palacios-Bejarano · G. Cerruela García (✉) · I. Luque Ruiz · M. Á. Gómez-Nieto
Department of Computing and Numerical Analysis, University of Córdoba, Campus de Rabanales, Albert Einstein Building, E-14071 Córdoba, Spain
e-mail: gcerruela@uco.es

B. Palacios-Bejarano
e-mail: i82pabeb@uco.es

I. Luque Ruiz
e-mail: iluque@uco.es

M. Á. Gómez-Nieto
e-mail: mangel@uco.es

Keywords 2D-QSAR · Molecular descriptors · Maximum Common Substructure · Genetic algorithm · Differential evolution algorithm · Weighted MCS tree

Introduction

2D-QSAR (Quantitative Structure–Activity Relationship) build prediction models of the activity of chemicals and drugs using extracted topological and molecular information from the molecules forming part of a data set under study [1–4]. Most of the different 2D-QSAR approximations build a 2D matrix as representational space of the molecule data set, where the matrix elements represent either: (1) a topological or molecular property of the molecular structure [5–9], or (2) a similarity measurement of the resemblance between the molecular or structural characteristics of pairwise molecules [10–13].

The former approximation uses *MxD* matrixes, where *M* is the cardinality of the data set and *D* the number of molecular descriptors selected, for the building of the representational space by storing in each (*i,j*) matrix element the value of the *j* molecular descriptor for the molecule *i*. In these methods, the selection of an appropriate set of molecular descriptors is essential for the building of a prediction model. These descriptors should be related to the activity to be predicted, and their number should be as low as possible in order to generate simple and interpretable models [8, 9, 14].

The second approximation uses *MxM* matrixes, storing in each matrix element *(i,j)* the similarity value between the molecules *i* and *j*. Similarity measurements can be obtained in several ways: (1) using isomorphism algorithms and matching the graph structure representing the pair of molecules, (2) representing the molecules as fingerprints

[15–18] and using a matching algorithm for searching the commonality between pairwise bits set to 1 *s* (and 0 *s*) between pairs of fingerprints. In both cases, a similarity index is selected in order to obtain a scalable similarity measurement, the selection of this index sometimes being determinant of the goodness of the generated prediction model [19–21].

Lastly, combined 2D-QSAR approximations have been proposed [22]. These approximations use molecular descriptors together with similarity measurements for the building of the representational space used in the prediction process. Thus, structural similarity values are used for the calculation of an initial measurement of the resemblance between two molecules of the data set, and the molecular descriptor values of the non isomorphic fragments of these two molecules are used to correct and weigh that similarity measurement, obtaining a finer and more accurate similarity value called approximate similarity (AS) [23, 24]. Once the representational space is built, for any of the 2D-QSAR approximations, the corresponding matrixes are used as input in the correlation process. Partial Square Least Regression (PLSR) is the statistical method often used [25, 26]. However, the application of other statistical and artificial intelligence methods such as genetic algorithms, simulated annealing algorithms, differential evolution, etc. have demonstrated, in some cases, an improvement in the reliability and robustness of the generated models [27–33].

Approximate similarity approach is closely related to other proposals and based on the study of the structural transformation of pairs of matched molecules. Matched molecular pair (MMP) studies analyze the influence of small structural changes between pairs of molecules sharing a large structural core by considering that it is possible to predict the change in the activity value between two molecules by means of the observed structural change. Thus, matching between all molecule pairs of the data set should be performed and structural differences are extracted between pairs of molecules, analyzing how these small structural differences increase or decrease the molecule activity values.

Leach et al. [34] use MMP analysis to study the changes in the aqueous solubility property, assuming that the change in the property studied resulting from a specific transformation depends only on the substructural change that has taken place, independently of the context, that is, of the structural environment in which that transformation happens.

However, Papadatos et al. [35] demonstrate that the inclusion of contextual information can enhance the predictive power of MMP analyses. Analyzing the hERG inhibition, solubility, and lipophilicity of a large data set of molecules, they identify the most significant trends (both positive and negative) in the property values that are not apparent when using conventional, context-independent approaches. The authors show that the influence of a specific substituent in a core structure is dependent not only on the substituent structure but also of the anchor position and the local environment.

These MMP works in such a way that they enable the study of the behavior substituents in the property value of the molecule. Values of molecule activities are previously known, so in these interesting contributions the authors find the behavior of substituents in core structures, to allow determining how a small substituent can affect the activity value of the molecule.

We propose in this paper a QSAR model for the in vitro calculation of the activity value of molecules based on, as the above commented proposals, the consideration that the difference in activity value between two molecules depends on the structural differences between both molecules. Our proposal takes into account simple and large transformations between pairs of molecules by means of the consideration of all the non isomorphic fragments (NIFs) between both molecules.

Thus, as we have described in previous works [22], structural isomorphism among all pairs of the molecule data set is calculated in order to classify the data set in a tree structure based on the maximum common substructure (MCS), common to the whole data set or groups of molecules. This tree structure, called MCS tree, hierarchically organizes the molecule data set in structural complexity levels. Thus, the root node represents the maximum common substructure to the entire data set, leaf nodes represent the molecules and the nodes of the intermediate levels in the tree represent core structures common to all the subgroup of molecules in the node branch.

The advantage of this tree structure is the information storing in the arcs. The arcs represent the NIF between a parent and a children node. That is, those fragments to be added to the parent structure in order to obtain the children structure (a MMP essentially).

From this tree structure it is easy to obtain any path between two molecules. These paths can be seen as a set of NIFs, and therefore it is possible to represent the relationship between two molecules as a simple equation composed of the structure of the molecules and the NIFs implied in the path.

Furthermore, it is easy to generate a weighted MCS tree by simply selecting one or several appropriate molecular descriptors and calculating the value of the descriptors for the structures stored in the nodes and arcs of the tree. Therefore, using this weighted MCS tree we can obtain valuable measurements for the paths between two molecules and use these measurements for the building of the representational space used in the generation of 2D-QSAR models.

Thus, instead to representing a classic MMP representational work space based on pairs of molecules and their structural differences, we build a representational one based on paths. The weighted tree structure enables us to assign a cost measurement to each path between any pair of molecules and to relate that cost to the structural differences between each pair of molecules and, therefore, with the contribution of the NIFs structures existing between each pair of molecules to the difference in their activity values.

This paper is organized as follows: in Sect. "Materials and methods" we describe the data sets used for the description for our proposal and its application. Data sets, extracted from the literature [36, 37], have been selected of different sizes and characteristics in order to be simple enough to allow us to give an explanation of the proposal and to demonstrate its application to different problems. Furthermore, in this same section we describe the foundation of our proposal, detailing the building of the MCS structure and the stages of the algorithm proposed for the building of the representational space. In Sect. "Experimental results", we show the experimental results for the two data sets selected. Results are compared with classical approximation using a descriptors' matrix as representational space. Then we compare the results of the application of our proposal with and without the use of optimization algorithms such as genetic and differential evolution algorithms. Finally we discuss the results obtained, commenting the advantages of the proposal described in this paper and our outgoing works.

Materials and methods

Data sets

In this study we have used two data sets collected from the bibliography for the development of QSAR models in the study of their activity against the *Mycobacterium tuberculosis*. The 2D-QSAR models proposed are based on topological and constitutional descriptors, as we have used in our proposal. Therefore, comparative results analysis can be carried out.

In order to explain our proposal in a clear and simple way, we have considered a small data set (19 molecules) composed of a series of 2-substituted isonicotinic acid hydrazide (INH) [37] (see Table 1). Isonicotinic acid hydrazide (INH, isoniazid) is one of the most effective agents in tuberculosis therapy and many INH derivatives have been synthesized involving ring substitution at position 2 and 6 of the pyridine moiety or modification at the hydrazine moiety showing an increase in their activity. Therefore, these kinds of compounds have been studied by other authors for the proposal of QSAR models in order to predict the activity of 2-substituted isonicotinic acid hydrazides. The low cardinality of this data set allows us to describe all data structures and algorithm steps, comparing the results of our study with previous results described in the literature.

The second data set (see Fig. 2 and Table 2) is composed of 34 molecules corresponding to 8-methoxyquinoline carboxylic acids derivatives [36], which also show potent in vitro and in vivo antitubercular activity. This data set is taken for the validation of our proposal and in this paper we compare our results with those previously described.

Tables 1 and 2 show the SMILES structures of the selected data set as well as the real and predicted molecule activity values. The activity values are given as MIC. The biological activity value MIC (μM) reported in the literature is converted to *-log* scale and subsequently used as the dependent variable for the QSAR analysis.

Genetic and differential evolution algorithms

Partial least square regression (PLSR) with leave-one-out (LOO) cross-validation processes was carried out over the representational space used by our proposal. Previously, initial data matrices, as we describe below, were built and representational spaces were obtained using genetic and differential evolution algorithms.

Genetic algorithm

Genetic algorithms (GA) are adaptive algorithms for finding the global optimum solution for an optimization problem. The algorithm begins by creating a random initial population and at each step creates a sequence of new populations, using the individuals in the current generation. To create the new population, the algorithm performs the following steps:

- Each member of the current population is scored by computing its fitness value.
- The raw fitness scores are scaled to convert them into a more usable range of values.
- Based on their fitness, some members, called parents, are selected.
- The individuals in the current population that have lower fitness are chosen as elite. These individuals automatically survive to the next generation.
- Creates children from the parents. Children are created using mutation introducing random changes, or mutations, to a single parent or by crossover combining the vector entries of a pair of parents.

Table 1 Data set of 2-substituted isonicotinic acid hydrazide

Molecule	SMILES formula	Experimental pMIC	pMIC (estimated)			
			Descriptors matrix	Proposed model		
				None	GA	DE
1	C1(CCNCC1)C(=O)NN	0.041	0.085	0.799	0.217	0.201
2	C1(CCNC(C1)C)C(=O)NN	0.716	1.323	1.232	0.892	0.876
3	C1(CCNC(C1)CC)C(=O)NN	1.324	1.809	1.330	1.364	1.372
4	C1(CCNC(C1)CCC)C(=O)NN	1.742	2.237	1.877	1.978	1.960
5	C1(CCNC(C1)CC(C)C)C(=O)NN	2.653	2.615	2.425	2.461	2.821
6	C1(CCNC(C1)OC)C(=O)NN	2.185	1.878	1.578	2.009	1.780
7	C1(CCNC(C1)OCC)C(=O)NN	2.655	2.038	2.148	2.228	2.416
8	C1(CCNC(C1)N)C(=O)NN	1.161	1.320	1.831	0.991	1.513
9	C1(CCNC(C1)NC(=O)C)C(=O)NN	3.332	2.662	3.731	3.268	3.663
10	C1(CCNC(C1)CNC(=O)C)C(=O)NN	2.386	3.581	2.164	2.436	2.268
11	C1(CCNC(C1)N(CC)CC)C(=O)NN	2.856	2.193	3.055	2.768	3.023
12	C1(CCNC(C1)F)C(=O)NN	2.415	1.425	1.800	2.239	2.255
13	C1(CCNC(C1)Cl)C(=O)NN	2.593	2.332	1.845	2.417	2.433
14	C1(CCNC(C1)Br)C(=O)NN	2.790	3.752	1.892	2.614	2.850
15	C1(CCNC(C1)I)C(=O)NN	2.404	1.423	1.967	2.056	2.220
16	C1(CCNC(C1)N(=O)=O)C(=O)NN	2.569	3.092	2.801	2.246	2.756
17	C1(CCNC(C1)C1CCCCC1)C(=O)NN	1.699	2.347	1.703	1.785	1.415
18	C1(CCNC(C1)CC1CCCCC1)C(=O)NN	1.585	0.819	1.782	1.885	1.690
19	C1(CCNC(C1)C=C)C(=O)NN	1.544	2.548	1.860	1.911	1.378

In-vivo activity values and estimated values using different approaches. *GA*: genetic algorithm, *DE*: Differential evolution algorithm

- Replaces the current population with the children to form the next generation.

The algorithm stops when one of the stopping criteria is met. Some of the conditions to determine when the GA stops are based on the number of generations, time limit, when the value of the fitness function for the best point in the current population is less than or equal to fitness limit, when the weighted average change in the fitness function value over all generations is less than function tolerance, when the weighted average change in the fitness function value over stall generations is less than function tolerance, etc.

The genetic algorithms are stochastic and then there is the possibility of slightly different results each time you run the genetic algorithm. Each time GA calls the pseudorandom number stream, its state changes. Therefore, the next time GA calls the stream; it returns a different random number. To reproduce our results exactly, we call GA with an output argument that contains the current state of the default stream, and then reset the state to this value before running again. To obtain better results we decided to execute the GA several times using the final population from a previous run as the initial population for a new run.

Differential evolution algorithm

Differential evolution (DE) is a class of genetic algorithms which use biology-inspired operations of crossover, mutation, and selection on a population in order to minimize an objective function [38]. As with the other evolutionary algorithms, DE solves optimization problems by evolving a population of candidate solutions using alteration and selection operators. DE uses floating-point instead of bit-string encoding of population members, and arithmetic operations instead of logical operations in mutation, in contrast to classic GA.

We consider NP to be the number of parameter vectors (members) $x \in R^d$ in the population, where d represents the dimension. In order to create the initial generation, NP guesses for the optimal value of the parameter vector are made, using random values between upper and lower bounds.

Each generation involves the creation of a new population from the current population members: $x_i | i = 1, \ldots, NP$, where i indexes the vectors that make up the population. This is accomplished using differential mutation of the population members. A mutant parameter vector v_i is created by choosing three members of the

Table 2 Data set of 8-methoxyquinoline carboxylic acids derivatives

Molecule	SMILES formula	Experimental pMIC	pMIC (estimated)			
			Descriptors matrix	Proposed model		
				None	GA	DE
1	COC1=C(CN2CCN(CC2)C(C2=CC=CC=C2)C2=CC=C(Cl)C=C2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−1.0460	−0.4543	−0.6781	−0.9850	−0.9857
2	COC1=C(CN2CCN(CC2)C(C2=CC=CC=C2)C2=CC=C(Cl)C=C2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.1072	−0.6933	−0.4368	−0.4229	−0.3900
3	COC1=C(CN2CCN(CC2)C(=O)C2=CC=CO2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−1.1373	−0.5977	−0.4459	−1.0763	−1.0770
4	COC1=C(CN2CCN(CC2)C(=O)C2=CC=CO2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.7958	−0.6373	−0.2187	−0.7348	−0.7355
5	COC1=C(CN2CCN(CC3=CC4=C(OCO4)C=C3)CC2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.1958	−0.2801	−0.2885	−0.3305	−0.3949
6	COC1=C(CN2CCN(CC3=CC4=C(OCO4)C=C3)CC2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	0.4559	−0.2138	−0.0459	0.3949	0.3956
7	COC1=C(CN2CCN(CC3=CC4=C(OCO4)C=C3)CC2)C(F)=C(N)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.4857	−0.1881	−0.6115	−0.3309	−0.5460
8	CCC1CN(CC2=C(OC)C3=C(C=C2F)C(=O)C(=CN3C2CC2)C(O)=O)CCN1C	−1.1411	−0.4526	−0.6548	−1.0801	−1.0808
9	CCC1CN(CC2=C(OC)C3=C(C(=O)C(=CN3C3CC3)C(O)=O)C(N(O)O)=C2F)CCN1C	−0.4969	−0.6470	−0.4139	−0.3328	−0.3493
10	COC1=C(CN2CCN(CC2)C(=O)C2COC3=CC=CC=C3O2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.1731	−0.3406	−0.2877	−0.3511	−0.4012
11	COC1=C(CN2CCN(CC2)C(=O)C2COC3=CC=CC=C3O2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.1367	−0.1893	0.0115	−0.0332	−0.1319
12	COC1=C(CN2CCSCC2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.3138	−0.0943	−0.6326	−0.6255	−0.5753
13	COC1=C(CN2CCSCC2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.2648	−0.2663	−0.3874	−0.2205	−0.3659
14	COC1=C(CN2CC(C)OC(C)C2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.6009	−0.5162	−0.5416	−0.4760	−0.5011
15	COC1=C(CN2CC(C)OC(C)C2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	0.0506	−0.6787	−0.2944	−0.2442	−0.2122
16	COC1=C(CN2CCC(CC2)N2CCCCC2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.8488	−0.5721	−0.4226	−0.7878	−0.7885
17	COC1=C(CN2CCC(CC2)N2CCCCC2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.8068	−0.5949	−0.1837	−0.7458	−0.7465
18	COC1=C(CN2CCC(O)(CC2)C2=CC=C(Cl)C=C2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.8082	−0.6154	−0.9513	−0.8422	−0.8001
19	COC1=C(CN2CCC(O)(CC2)C2=CC=C(Cl)C=C2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.7693	−0.6169	−0.7184	−0.6327	−0.5584
20	COC1=C(CN2CCC(CC2)N2C(=O)NC3=C2C=CC(Cl)=C3)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.7737	−0.3243	−0.8540	−0.8227	−0.8445
21	COC1=C(CN2CCC(CC2)N2C(=O)NC3=C2C=CC(Cl)=C3)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.7379	−0.2122	−0.5897	−0.5518	−0.6499
22	CCCCNC(=O)C1CCCN(CC2=C(OC)C3=C(C=C2F)C(=O)C(=CN3C2CC2)C(O)=O)C1	−0.2278	−0.3054	−0.6104	−0.2888	−0.5432
23	CCCCNC(=O)C1CCCN(CC2=C(OC)C3=C(C(=O)C(=CN3C3CC3)C(O)=O)C(N(O)O)=C2F)C1	−0.1903	−0.3003	−0.3685	−0.2898	−0.3095
24	COC1=C(CN2CCC3(CC2)OCCO3)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	0.0315	−0.8044	−0.3579	−0.0295	−0.0288
25	COC1=C(CN2CCC3(CC2)OCCO3)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	0.0757	−0.8712	−0.1222	−0.1585	−0.1347
26	COC1=C(CN2CCC3=CC=CC=C3C2C(=O)NC(C)(C)C)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−1.0902	−0.0614	−0.4079	−1.0292	−1.0299
27	COC1=C(CN2CCC3=CC=CC=C3C2C(=O)NC(C)(C)C)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.1492	−0.1206	−0.1057	−0.1582	−0.1650
28	COC1=C(COC2=CC=C(CC3N(C)CCC4=C3CCCC4)C=C2)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.1643	−0.1854	−0.2132	−0.2344	−0.2759
29	COC1=C(COC2=CC=C(CC3N(C)CCC4=C3CCCC4)C=C2)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	0.7958	−0.2545	0.6576	0.7830	0.7714
30	COC1=C(COC2=CC=C(CC3N(C)CCC4=C3CCCC4)C=C2)C(F)=C(N)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.4548	−0.0765	−0.3975	−0.4078	−0.4262
31	COC1=C(CN2CCN3C=C(N=C3C2)C(O)=O)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.5477	−0.4797	−0.5339	−0.5805	−0.6542
32	COC1=C(CN2CCN3C=C(N=C3C2)C(O)=O)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.4941	−0.4476	−0.2826	−0.3380	−0.4102
33	COC1=C(CN2COCC2(C)C)C(F)=CC2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.6170	−0.5688	−0.5546	−0.4990	−0.5141
34	COC1=C(CN2COCC2(C)C)C(F)=C(N(O)O)C2=C1N(C=C(C(O)=O)C2=O)C1CC1	−0.2671	−0.7040	−0.3304	−0.2680	−0.2424

In-vivo activity values and estimated values using different approaches. *GA* genetic algorithm, *DE* differential evolution algorithm

population, x_{i1}, x_{i2} and x_{i3}, at random. Then v_i is generated as: $v_i = x_{i1} + F(x_{i2} - x_{i3})$, where F is a positive scale factor, effective values which are typically less than one.

To complement the differential mutation search strategy, DE also employs uniform crossover. Sometimes referred to as discrete recombination, (dual) crossover builds trial vectors out of parameter values that have been copied from two different vectors [38]. After obtaining the NP noisy random vectors, crossover takes place at random, comparing these with the original vectors. Specifically, DE crosses each vector with a mutant vector assigning to the new trial vectors a v_i value if $rand(0,1) \leq CR$ or an x_i value otherwise.

After the first mutation operation, mutation is continued until d mutations have been made, with a crossover probability: $CR \in [0,1]$. The crossover probability CR controls the fraction of the parameter values that are copied from the mutant. Mutation is applied in this way to each member of the population. If an element of the trial parameter vector is found to violate the bounds after mutation and crossover, it is reset in such a way that the bounds are respected. Then, the objective function values associated with the children are determined. If a trial vector has equal or lower objective function value than the previous vector it replaces the previous vector in the population; otherwise the previous vector remains.

For our experimental tests the differential evolution source code proposed in [38] was used. The descriptors were normalized in the rank [0,1] and we fixed the following parameters in the algorithm: the initial lower and upper bounds = [−1,1], the maximum number of iterations (generations) = 200, the differential evolution step size (F) = 0.8 and the crossover probability constant = 0.8.

Building of hierarchical representation of the data set

Algorithms used in MMP use two different approximations: (1) based on MCS, and (2) based on molecule fragmentation [39–42]. Algorithms based on MCS match each pair of molecule data set obtaining the common and non common fragments. These proposals allow obtaining single and multiple transformations, although they usually entail high cost processing. By other hand, molecule fragmentation algorithms have a high performance but only single transformation can be identified easily.

Our proposal is based on the MCS calculation, for which we have used an efficient algorithm proposed by Vargyas et al. [43] included in the JChem software. This algorithm does not calculate the MCS isomorphism between all pairs of molecule data sets, but only between clusters of molecules in order to generate the MCS tree. Thus, this algorithm generatees a MCS tree structure in only 55 h for a data set of one hundred thousand of molecules, being more efficient than classic algorithms for MCS calculation [35, 44].

Initially, the molecule data set, represented as SDF or MOL structures, is processed in order to build a hierarchical structure based on the common substructures to subset of molecules.

Thus, the molecule data set is set as leaf nodes (last level) of branches of a tree structure. Calculation of the MCS isomorphism between the molecule data set is performed obtaining a set of structures common to the subset of molecules. These structures are assigned to the upper level as parent nodes of the common molecules. Next, MCS isomorphism is calculated for all the structures of the parent nodes, again obtaining structures common to the subset of parent nodes. These structures are assigned as parent nodes in the next upper level. The process is repeated until a unique MCS substructure is obtained that is assigned as root of the tree.

As observed, the size of the structures assigned to the nodes of the tree increases from the root to the leaf nodes. In addition, the arcs of the tree represent the structural differences between two nodes linked by the arc. Hence, a molecule (leaf node) can be seen as the set of substructures composed of the root node and the arcs of the path between the root node and the leaf node.

The algorithm used builds this tree structure in a bottom-up process, using different heuristics whose aim is to minimize the computational cost of the process and, as we describe below, to extract added information about the isomorphism between molecules in the process.

Therefore, the algorithm starts [22] by classifying all the structures on the dataset in branches (bottom) of the hierarchy. The next level contains the MCSs as clusters of the initial molecules, all the molecules that share a common structure are assigned to the same cluster. At the root of the hierarchy there is one (or many) MCS common to the whole dataset, creating disjoint clusters (one molecule belongs to just one unique cluster).

Heuristics such as: (1) use of the similarity matrix obtained from the data set fingerprints for predefining a similarity threshold as main condition for the calculation of the MCS, (2) no consideration of MCSs that could contain a number of atoms under a preset value, and (3) two classification models, one approximate and one exact, are used in order to obtain hierarchical structures with just one root node, and balanced tree with a not excessive number of levels.

Figure 1 shows the tree structure obtained for 2-substituted isonicotinic acid hydrazide. Molecules, represented in the branches of the tree, are grouped by their common substructures in nodes of a higher level (**N11**, **N12**, **N21**, **N22**, **N31**, **N32** and **N33**). The grouping process is repeated building new nodes of the tree until a common substructure

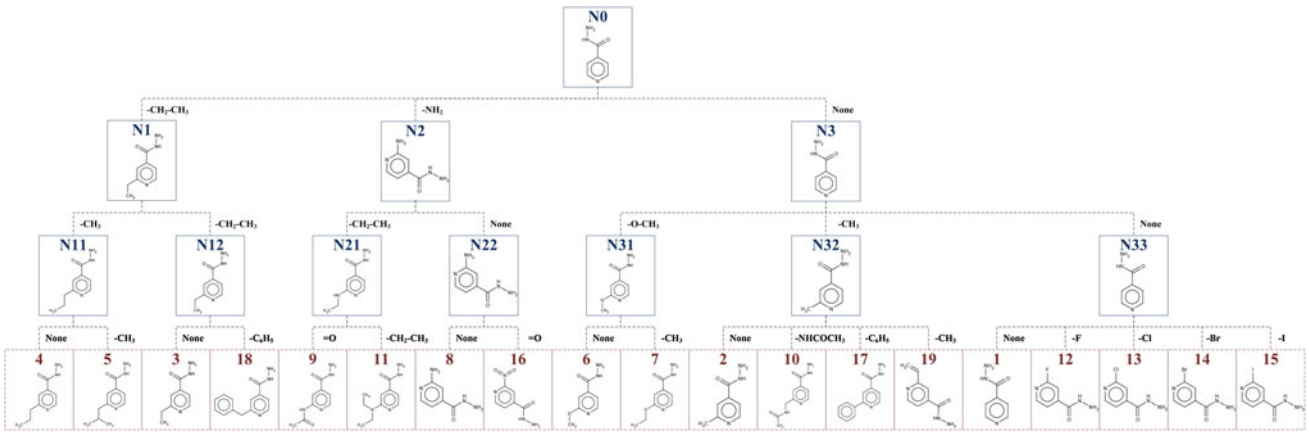

Fig. 1 MCS hierarchical structure for 2-substituted isonicotinic acid hydrazide data set

for the whole data set is obtained, this node (**N0**) being the root of the tree.

Extracting information of non isomorphic fragments

The matching process between any two molecular structures generates information with: (1) one or several substructures common to the structures of the compared molecules and, no less important, (2) one or several substructures not common and belonging to one or both of the compared molecules.

The former structures represent either: the maximum common structures (MCS) or the all maximum common structures to the matched molecules (AMCS). The algorithm developed by authors [45] allows the extraction of any of the two kinds of isomorphism. MCS and AMCS can be used for the obtaining of different structural similarity measurements in order to carry out QSAR/QSPR models, cluster analysis, etc.

Besides this, in the matching process our algorithm is able to extract those fragments or substructures not common to the compared molecules. In the building of the hierarchical structure represented in Fig. 1, these non isomorphic fragments (NIF) are obtained, and their information is properly retained.

Non isomorphic fragments are stored in the arcs of the tree structure as Fig. 1 shows. Thus, we can picture the hierarchical structure as a balanced and weighted tree. Upper nodes represent isomorphic fragments to the largest structure subsets (the nodes of the last level represent the molecule data set), and arcs represent the non isomorphic fragments (NIF) between a parent and a child node.

The weight of the nodes and arcs can be obtained in any way, for instance, the value of selected molecular descriptors. Moreover, the arch weight can be considered as a distance measurement, meaning the cost to reach a child node from a parent node. Thus, many weighted tree structures can be built using different approaches being used for the building of snowflake diagrams, allowing the study and classification of data sets [22].

The basis of the proposal

The development of QSAR is based on the building of an equation able to predict the activity of a set of molecules using other sets of similar molecules (training set) for the building of a correlation model representing the relationship between the molecule activities and some measurement extracted from the molecules.

In 2D-QSAR approaching the measurements extracted from the molecules is based on molecule structural information, such as: (1) structural similarity measurements obtained from molecular graph or fingerprint matching, and (2) molecular descriptors values calculated using square matrices representing some chemical, physical and/or mathematic information of the molecular graph.

Hence, in 2D-QSAR approaches we can find in the literature the use the representational models: (1) symmetrical square matrices storing structural similarity measurements obtained from graph isomorphism (MCS, AMCS) or fingerprint resemblance [46–50], (2) non symmetrical matrices, with rows representing the molecule data set and columns representing a selected set of molecular descriptors, storing in each matrix element the value of a descriptor [2, 7, 31, 51, 52], and (3) a combined representation of both, that is, square non symmetrical matrixes storing approximate similarity measurements obtained by means of the consideration of structural similarity and molecular descriptor values [11, 24].

Basically, approximate similarity (AS) approach tries to build equations considering the influence of the non isomorphic fragments extracted in the matching of two molecules to the difference of the activity or property in the

study of these molecules. Thus, on the one hand, the AS approach takes into account that the activity of a molecule should have a close value to the activity of a similar molecule (Maggiora principle [53]), and on the other hand, the AS approach considers the difference between the activity values that can be measured, taking into account the specific and weighted contribution of the non isomorphic fragments.

If we observe the tree structure of Fig. 1, we can appreciate the non isomorphic fragments between any pair of molecules. For instance, when we observe the left branch of the tree, if we want to reach molecule **18** from molecule **5**, it is necessary to traverse the arc between molecule **5** to node **N11**, to reach node **N1**, the arc between nodes **N11** and **N1**, the arc between nodes **N1** and **N12** and finally to traverse the arch between node **N12** and molecule **18**. As we can observe in Fig. 1, theses arcs are weighted with NIF structures.

Thus, we may consider that the cost to reach molecule **18** from molecule **5** be calculated considering the costs of the navigation in the tree from molecule **5** to **18**. If, in addition, we consider that the nodes and arcs of the tree structure of Fig. 1 are weighted, then we can find a simple equation to calculate the weight of a node knowing the weight of any other node and the weights of the arcs necessary to reach one node from another. The foundation of our proposal is based on this characteristic of a MCS weighted tree.

Therefore, we consider the existence of some function *f()* used for the calculation of the weights of the nodes and arcs of a MCS tree structure, as the shown in Fig. 1, able to satisfy the following equation, among others:

$$W(N_i) - \sum_{l=i}^{l=k} W(NIF_l) = W(N_j) - \sum_{l=j}^{l=k} W(NIF_l) \quad (1)$$

where $W(N_i)$ and $W(N_j)$ are the weight of any two nodes i and j of the tree structure, and $W(NIF_l)$ are the weights of the arcs present in the path between the nodes i and j crossing any other common parent node N_k present in the tree.

Applying the Eq. (1) to the tree of Fig. 1, the relationship between molecules **5** and **18** could be expressed for the path between molecules **M5** and **M18** crossing by the common parent node **N1**:

$$W(M5) - W(NIF_{[M5-N11]}) - W(NIF_{[N11-N1]}) = W(M18) - W(NIF_{[M18-N12]}) - W(NIF_{[N12-N1]}) \quad (2)$$

In the building process of the MCS tree of Fig. 1, the Eq. (2) is not satisfied. The weights of nodes and arcs are obtained by means of one or more sets of molecular descriptors, of course, but most molecular descriptors are not additives. For instance, if we use the Wiener index to weight the nodes and arcs, we can find for any branch that:

$$W(N_i) - \sum_{l=i}^{l=k} W(NIF_l) \neq W(N_j) - \sum_{l=j}^{l=k} W(NIF_l) \quad (3)$$

Furthermore, 2D-QSAR models using molecular descriptors for the building of prediction equations are based on the following:

$$A_i = g[\vec{D}_i] \quad (4)$$

where A_i represents the activity of a molecule i, and $\vec{D}_i$ represents an array of values of molecular descriptors extracted from the molecule i, and *g()* is the correlation function.

The approach proposed in this paper is based on the combining of Eqs. (1) and (4), and can be formulated as follows: if there exists a function g() able to relate the activities of a set of molecules to the values of a set of molecular descriptors $\vec{D}_i$, then it is possible to find a more robust function $g^*(g(f))$ relating the changes of the activity values between any pair molecules of the data set with the changes in the weight of the path crossing by a common parent node between that pair of molecules.

Thus, we picture the MCS tree as a reaction space representation which describes how we can get a molecular structure (a leaf node) from any other node of the tree adding/subtracting the arcs of the path between them. Then we are able to build a relationship between a selected set of molecular descriptors calculated for the nodes and arcs (NIF structures) of the tree and the activities of the molecules (leaf nodes), considering that the changes in the activity values of the molecules are related to the cost of the path between any pair of molecules, this cost being calculated from the values of the selected descriptors for the structures involved in the path.

However, if we consider two different NIF joined to a same core, the influence of theses NIF over the molecule activity in most cases is not directly related only to the NIF structure, and therefore to the weight of the arcs representing those NIFs in the tree structure. In addition, if we observe the tree of Fig. 1 different arcs have assigned a same NIF structure (i.e. the arcs **N31**-**M7** and **N32**-**M19**, where the NIF structure is $-CH_3$), but this NIF could affect the activity of the molecules in different ways.

Therefore, the descriptor values for the NIF structures involved in a path should be weighted depending on the structures of the nodes joined by the arc (NIF) in the tree. Therefore, as Papadatos et al. [35] does, our proposal considers the influence of the context, that is, a substructure affect to the activity value depending on the molecular in which the substructure is anchored.

The proposed algorithm

The algorithm developed is based on a genetic algorithm [54–57] working in different stages as follows:

Preprocessing stage

In this stage the molecule data set is analyzed and the MCS tree is built. In this process isomorphism between any pair of molecules is obtained, and the nodes and arcs of the MCS tree are labeled assigning the corresponding structure (NIFs for the arcs, molecules for the leaf nodes and cores with maximum substructures for the remaining nodes). We have used the JChem package [58] for the building of the tree, implementing the algorithm proposed by Vargyas et al. [43].

In the first step, the algorithm considers all the molecules in the data set as singletons, and in the following steps a merged process is carried out where all the elements with the same MCS are assigned to a new node in the next high level of the hierarchy. Then, the arcs between the new node and the children nodes are labeled with the NIFs extracted between the MCS assigned to the parent node and the corresponding structures of the children nodes. This process is repeated using as input the MCS assigned to the nodes of the new level created.

At the end of the process, a tree (MCS tree) is created where the root node contains one (or several) common MCS to the whole data set. This method creates disjoint clusters where a molecule belongs to one and only one cluster.

The algorithm combines similarity search with MCS and substructure search in order to find the MCS of multiple structures efficiently [43]. A fingerprint similarity matrix is created for all elements in the data set to get the most similar elements to calculate the MCS, a threshold and a minimum number of atoms were established as a necessary condition to calculate the MCS or make singletons.

Molecular descriptors study stage

In this stage we study the behavior of the molecule activity against a set of molecular descriptors. A matrix of m rows (number of molecules) and d columns (number of selected molecular descriptors) is built. Correlation analysis is carried out. The correlates descriptors are eliminated and those descriptors showing better behavior are selected for the following stages.

In the study only a subset of topological and constitutional descriptors were selected, that which the CoChiSE software [59] used for the calculation is able to calculate the value for non-isomorphic fragments. Finally, the descriptors used in the experimental were as follows: *Randic connectivity, cyclomatic number, molecular weight, relative C-atoms, relative Cl-atoms, relative F-atoms, relative H-atoms, relative I-atoms, relative N-atoms, relative O-atoms, Detour-Wiener, First Zagreb, Harary, Harary-Szeged, Kier shape 1, Schultz and Wiener.*

Building the representational space stage

In the algorithm all the non redundant paths present in the MCS tree between two any pairs of nodes is used as the representational space. Thus, an R matrix with p rows (number of all non redundant paths), s fragments (the number of different structures participating in the paths) and d descriptors (the selected descriptor in the previous stage) is built. In this representational space, each element $R(i, j, k)$ represents the value of a molecular descriptor k, for a fragment j (NIF, core or molecule) involved in a path i.

The magnitude order of the $R(i, j, k)$ values is very different depending on the considered number of descriptors k, therefore matrix R is normalized in order that all elements show values in a magnitude order close to the activity values studied.

Algorithm procedure

The algorithm developed works in several steps:

Step 1 In the first step each of the molecule data set is taken as origin of the all paths to the remaining molecules. Thus, for each molecule as origin, $M-1$ equations (M is the number of molecules in the data set) are built as follows:

$$\Delta A = A_x - A_y = R_{(x,y)} \tag{5}$$

where A_x and A_y are the activity values of the molecules x and y, respectively, and $R_{(x,y)}$ is a submatrix of R composed by the paths (rows) considering the molecule x as origin and the molecule y as end of the path. Thus, R takes into account the path between each x molecule in the tree as origin of the path and the remaining molecules in the tree as end in the path.

Step 2 Genetic algorithm is used, weighting the matrixes$R_{(x,y)}$. In this step the fulfillment of Eq. (1) was used as fitness function, and PLS regression using LOO (leave-one-out) cross-validation is carried out to predict the values of the activity for each molecule (A_i^p) taking into account the weights in R; these values will be used in the following step of the algorithm. The minimization of standard error in cross validation (SECV) is the criterion used to decide the best activity prediction for the molecules when they are taken as origin of the path.

The procedure used to weight searches the weights values according to the element types in all equations. Hence, an NIF (arc), a molecule (node) will have the same

weight in all equations where these elements intervene. Therefore, in this step weights for the different path elements and weights for the different molecular descriptors are fitted.

After this step, we retrieve some molecules can be found as outliers. In the PLSR analysis we have established a confidence limit of 95 % ($t = 2.5$). Outliers are disregarded from the model in the current step, and for the next step the knowledge value of activity is used.

Step 3 The best result for (A_i^p) (that presenting the lower error in the previous step) is fixed, and equations where molecule i is the origin of the path are erased from the process.

In addition, the weights corresponding to the NIFs structures involved in the paths considered in the matrix R are used as input for the next run.

Then, the process is repeated (steps 2 and 3) until all activity values are fixed. The new fixed activity values are used in the next round of the algorithm. That is, if in a round the molecule x is taken as origin of the path, a new predicted activity value is obtained and fixed for this molecule x for the next rounds where new molecules y are taken as origin and the molecule x intervenes in the path. This iterative process taking estimated activities allows finer adjustment for weights.

Step 4 Finally, considering initial activities values, a PLS regression with LOO cross-validation analysis is performed, and correlation equations are obtained.

Experimental results

Experimental tests of the proposed method have been carried out with the two data sets selected and described above. As we can observe in Figs. 1 and 2 the selection of this data set is not trivial. The data set of 2-substituted isonicotinic acid hydrazide is organized in the tree structure in a very different way to the data set of 8-methoxyquinoline carboxylic acids derivatives. The first generates arcs of the tree storing small non isomorphic substructures and most of these arcs store nothing since the children node of a branch stores the same structure as the parent node. However, in the second data set (see Fig. 2) most of the arcs store large non isomorphic substructures, the parent being very different to the children node.

In order to demonstrate the goodness of the proposed model, we have carried out a PLS regression using as representational space the descriptors' matrix of the data sets. This classical representational space is composed to an MxD matrix, where M is the cardinality of the data set and D is the number of descriptors considered. The matrix elements store the value of the corresponding descriptor for each molecule.

Using the different representational spaces (classical and the one generated by our model) we have performed a PLSR with LOO analysis as follows: a) without using optimization algorithms, b) using genetic algorithms for optimization, and c) using differential evolution algorithms for optimization. For each of the data set considered we have obtained the equations and statistical parameters have been compared.

The application of the proposed model implies the prediction of the values of the molecule activities. That is, the process not only implies the searching of a correlation function between the variables considered in the representational space and the molecule activities, but also the prediction of these activities, because in each step of the algorithm we consider an initial activity value in order to obtain and estimate the value of the activity, and these initial values can be obtained also in a previous estimation step. Thus, our model can be used either for correlation and prediction at time, depending on the reliability to provide, as initial values of the activity, values that are very close to the real ones or those obtained experimentally.

Results for data set of 2-substituted isonicotinic acid hydrazides

Table 3 shows the equations obtained for the data set of 2-substituted isonicotinic acid hydrazide. In the classic procedure using the descriptors matrix composed of 17 columns (number of molecular descriptors considered) storing the descriptors values for the molecule data set, the equation obtained shows a very poor correlation. Values of $slope = 0.61$, $bias = 0.75$ and $Q^2 = 0.45$ determine there is no correlation between the activity values and the molecular descriptors. The very low value of Q^2 and the high value of SECV = 0.70, show that this method can only be used for classification proposes.

When genetic and differential evolution algorithms are used for this representational space the statistical values of the equations improve considerably as shown in Table 3. Using optimization algorithms the values of Q^2 increase to 0.7, although slope and bias values are still poor.

However, the equations obtained using the proposed model show excellent values of the statistical parameters. Values of $slope = 0.98$, $bias = 0.05$, $Q^2 = 0.94$ and SECV = 0.49 are obtained without the use of any optimization algorithms. Moreover, when optimization is used similar results are obtained with both methods (genetic and differential evolution algorithms). With both methods the value of Q^2 is also higher than 0.9 and the SECV values are much lower than the variance of activity values of the data

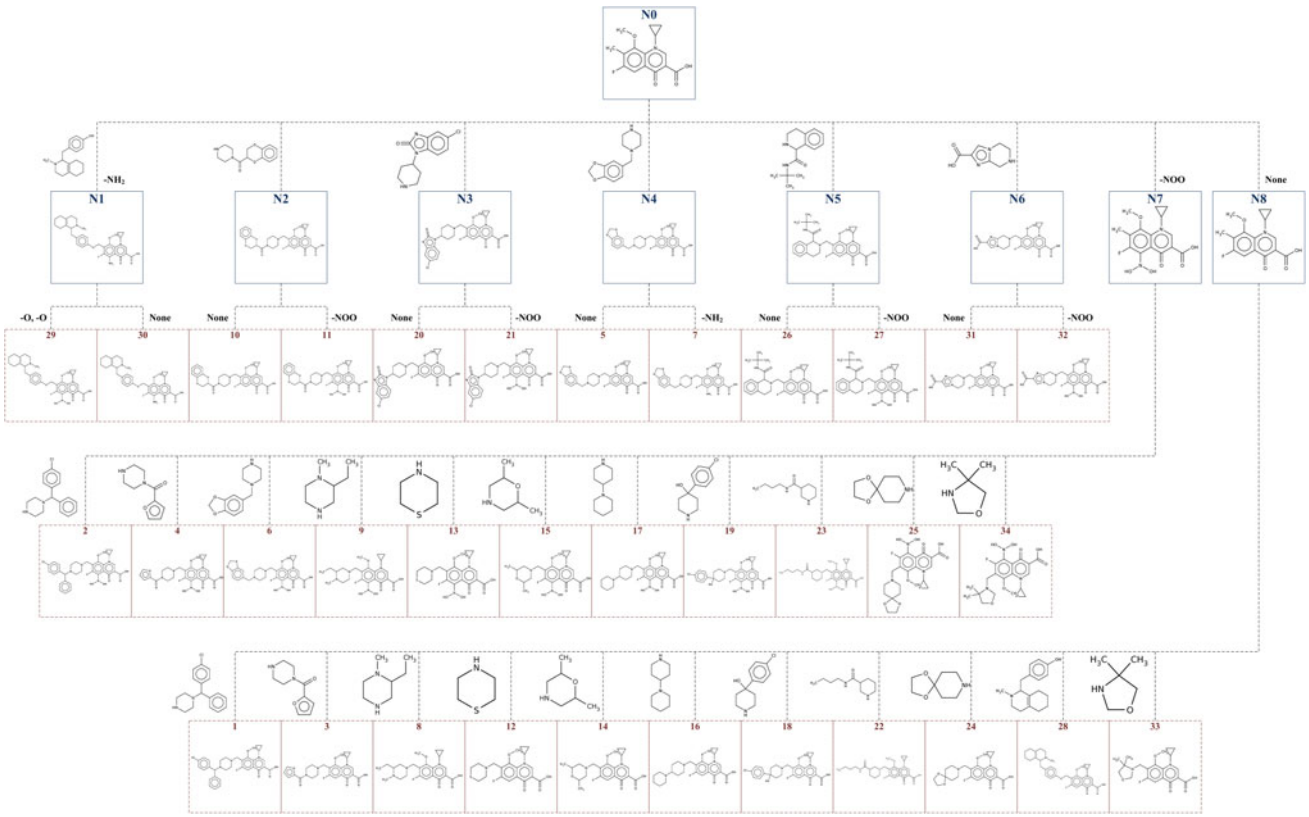

Fig. 2 MCS hierarchical structure for 8-methoxyquinoline carboxylic acids derivatives data set

Table 3 Results of the QSAR models for 2-substituted isonicotinic acid hydrazide data set using different algorithms and representational spaces

Representational space	Optimization algorithm	Slope	Bias	Q^2	SECV	Factors
Descriptors' matrix	None	0.6178	0.7505	0.4574	0.7029	8
	GA	0.8333	0.3255	0.7008	0.4740	8
	Stdev	0.0113	0.0484	0.0145	0.0115	
	DE	0.9170	0.1485	0.7310	0.4349	5
	Stdev	0.0075	0.0034	0.0038	0.0034	
Proposed model	None	0.9874	0.0554	0.9467	0.4922	8
	GA	1.1192	−0.1904	0.9359	0.2314	7
	Stdev	0.0035	0.0092	0.0024	0.0036	
	DE	0.9367	0.0555	0.9310	0.2197	7
	Stdev	0.0014	0.0062	0.0023	0.0029	

GA genetic algorithm, *DE* differential evolution algorithm, *Stdev*: standard deviation

set (*variance* = 0.68). Optimization methods generate a diminishing of the SECV value to 50 % of that when optimization is not used; they also reduce the number of factors used in the equations.

Table 3 also shows the standard deviation calculated for the statistical parameters after the fifteen rounds performed with the optimization algorithms. We can observe the low values of the standard deviation for all the equation parameters, showing the high repeatability of the proposed method. It can seen noticed that both genetic and differential evolution algorithms are stochastic methods devoted to find weight values for the variables that minimize the error and maximize the correlation. Different rounds of the algorithms have generated very close values of the statistical values and the predicted values of the molecule activities.

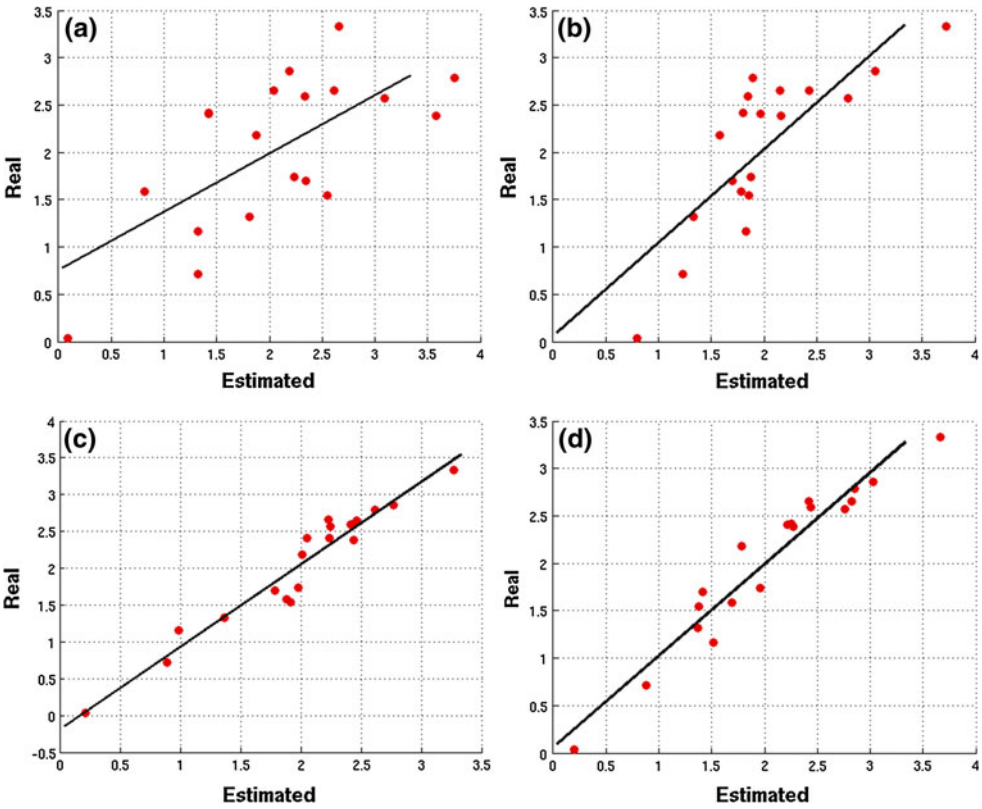

Fig. 3 Representation of real versus estimated *pMIC* for 2-substituted isonicotinic acid hydrazide: **a** using descriptor matrix, **b** using the proposed model without algorithms, **c** using proposed model with genetic algorithm, **d** using proposed model with differential evolution algorithm

Predicted values for the molecule activity are shown in Table 1 and the representations of the experimental versus predicted values are shown in Fig. 3. Observing Fig. 3a (and Table 1) we can see the extremely poor results obtained when the classic method is used. The use of 2D-QSAR method based on descriptor matrix does not provide an appropriate prediction method because a high deviation between experimental and estimated values of the activity is obtained.

When our proposed algorithm is used without applying optimization algorithms we observe a clear improvement, in the results, as observed in Fig. 3b. Furthermore, when we applied our model with optimization algorithms we can observe in Fig. 3c, d a very good linearity between experimental and estimated values of the activity. Better results are obtained when a genetic algorithm is used instead of a differential evolution algorithm. In both cases, the higher deviation of estimated versus experimental values of the activity are obtained for those molecules containing functional groups not present in other molecules (for instance, molecule **16** containing the $-NO_2$ moiety) or molecules showing an activity value very different to very similar molecules. For instance, molecule **7** is very similar to molecule **6**, however the substitution of a $-OCH_3$ by a $-OCH_2CH_3$ generates an increase of *0.5* in the activity value, on the other hand when a similar situation happens see molecules **17** and **18** the activity values are very close.

Results for data set of 8-methoxyquinoline carboxylic acids derivatives

Table 4 shows the results obtained for the 8-methoxyquinoline carboxylic acid derivatives data set. Again, we can observe that the use of the classic 2D-QSAR method based on the descriptors' matrix produces extremely poor results. Only when optimization algorithms are used do the statistical parameters improve, although values of $slope = 0.7$ and $Q^2 = 0.38$ using genetic algorithms and $slope = 0.8$ and $Q^2 = 0.5$ using differential evolution algorithms determine the infeasibility of the descriptors' matrix as representational space.

However, as shown in Table 4, the equations generated using the proposed model present good values for the statistical parameters. Without the use of optimization algorithms values of $slope = 1.0$, $bias = 0.0$, $Q^2 = 0.9$ are obtained, although the value of standard error in cross-validation of 0.32 is higher than the value of the variance of the sample (variance = 0.19).

The use of optimization algorithms even improves the results obtained. For both optimization methods the values of *slope*, *bias* and Q^2 are very similar to those without the use of optimization, but the values of SECV are lower than the variance of the sample.

Estimated values with or without optimization algorithms are shown in Table 2. Figure 4 represents the relationship

Table 4 Results of the QSAR models for 8-methoxyquinoline carboxylic acids derivatives data set using different algorithms and representational spaces

Representational space	Optimization algorithm	Slope	Bias	Q^2	SECV	Factors
Descriptors' matrix	None	0.2010	−0.3396	0.0105	0.4750	5
	GA	0.7135	−0.1068	0.3820	0.3650	10
	Stdev	0.0005	0.0002	0.0004	0.0006	
	DE	0.8359	−0.0704	0.5681	0.2977	10
	Stdev	0.0045	0.0013	0.0005	0.0008	
Proposed model	None	1.0047	0.0166	0.9078	0.3202	8
	GA	1.0694	0.0388	0.9083	0.1369	7
	Stdev	0.0040	0.0012	0.0023	0.0006	
	DE	1.0907	0.0792	0.9134	0.1400	7
	Stdev	0.0036	0.0016	0.0009	0.0003	

GA genetic algorithm, *DE* differential evolution algorithm, *Stdev* standard deviation

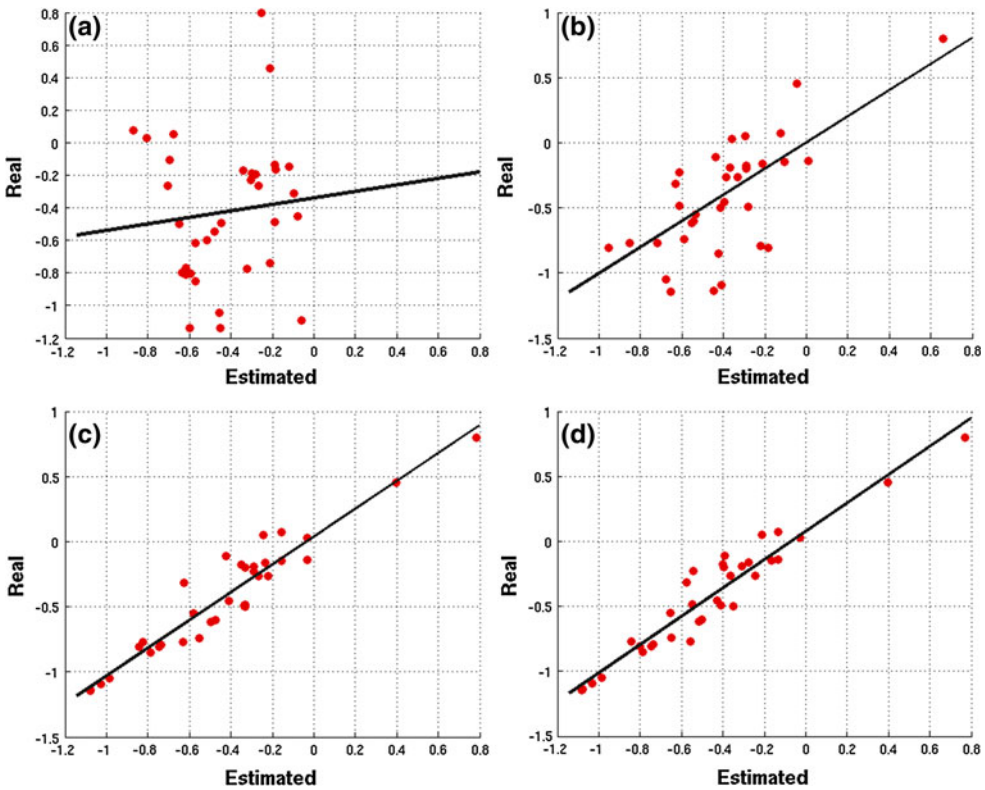

Fig. 4 Representation of real versus estimated *pMIC* for 8-methoxyquinoline carboxylic acids derivatives: **a** using descriptor matrix, **b** using the proposed model without algorithms, **c** using proposed model with genetic algorithm, **d** using proposed model with differential evolution algorithm

between real and estimated values of the activity of the different approximations studied. This figure clearly shows the improvement of the proposed model against the descriptors' matrix and the improvement of the use of the optimization algorithms.

Deviation between real and estimated values of the activity is difficult to interpret. The organization of the hierarchical structure (see Fig. 2) for this data set is complex. Tree levels are organized very irregularly. The arcs between the root node (**N0**) and nodes **N1** to **N6** making up the first level of the tree are composed of large structures, thus the non isomorphic structures describing the arcs between those nodes and the children molecules are none or very small structures. However, the arcs between the root node and nodes **N7** and **N8** of the first level of the tree are composed of small structures, while the arcs between those nodes and their children molecules describe very large non isomorphic structures. The use of the non

Table 5 Results obtained in the prediction for 8-methoxyquinoline carboxylic acids derivatives data set

Molecules	pMIC values						
	Real	Initial estimated values			Refined predicted values		
		Optimization method			Optimization method		
		None	GA	DE	None	GA	DE
9	−0.4969	−0.3793	−0.2018	−0.3678	−0.4178	−0.3418	−0.4060
14	−0.6009	−0.5068	−0.3494	−0.4285	−0.5511	−0.4938	−0.5044
21	−0.7379	−0.5277	−0.6011	−0.4806	−0.5719	−0.5750	−0.7005
29	0.7958	0.5176	0.7312	0.7521	0.6825	0.7462	0.8617
31	−0.5477	−0.4742	−0.5271	−0.5579	−0.5553	−0.6340	−0.5804
33	−0.6170	−0.5163	−0.3560	−0.4376	−0.5649	−0.4480	−0.5173

Initial and refined values of activity are shown without and using optimization algorithms. *GA* genetic algorithm, *DE* differential evolution algorithm

isomorphic structures to construct the representational spaces and the high differences between the size of these moieties and their structural characteristics, determine the increment higher or lower between the real and estimated values of the activity.

Application to the prediction of activity with refinement

The proposed method we have described in the paper above not only implies the searching of a correlation function between the variables considered in the representational space and the molecule activities, but also the prediction of these activities. However, in the prediction the model needs an initial value of the activities to be predicted in order to be taken by the algorithm in the iterations until an estimated value is obtained. Therefore, the goodness of the prediction depends on the goodness of these initial values.

Initial values can be set in many different ways. We can perform a PLS regression using any representational space (descriptors' matrix, similarity matrix, etc.). In this regression we can include or exclude the molecules whose activity we need to predict in order to generate a better fitness of the predicted activities.

The proposed method, described in this paper, allows us to carry out a finer prediction of unknown molecule activities, as follows:

- Initially, the tree structure is generated for the entire dataset containing molecules with and without known activity.
- A LOO regression is performed using the proposed model and only considering the set of molecules with known activity. This step generates an initial model that we use to obtain the estimated activity values for these molecules with unknown activity.
- Now, these estimated activities are considered as initial values and the process is again performed but considering the whole tree structure.
- Finally, finer prediction values are obtained for these molecules with unknown activity.

Table 5 shows the results of this process using the equations obtained in the regression, the initial activity values of the molecules 8-methoxyquinoline carboxylic acids derivatives data set. We have randomly selected a set of six molecules as the group of molecules with unknown activity: molecules **9**, **14**, **21**, **29**, **31** and **33**.

At first we built the whole tree structure, as Fig. 2 shows, considering the 34 molecules of the data set, those with known activity (28 molecules), and those with unknown activity (6 molecules).

Next, the model described in this paper is applied to only the 28 molecules. Thus, a PLS regression with LOO analysis is carried out and a correlation equation is obtained with the following statistical parameters: $slope = 1.29$, $bias = 0.07$, $Q^2 = 0.99$, SECV = 0.17 (without optimization), $slope = 1.15$, $bias = -0.11$, $Q^2 = 0.98$, SECV = 0.21 (using genetic algorithms), and $slope = 1.14$, $bias = -0.08$, $Q^2 = 0.99$, SECV = 0.17 (using differential evolution optimization). These equations are used for the estimation of the activity for the six selected molecules. Table 5 shows the values obtained for the initial estimated values of the activity for the six molecules.

In the following step, these estimated values are considered as initial values of activities, and the process is repeated considering, in this step, the 34 molecules; that is, the whole tree structure of Fig. 1. Thus, a new model is generated with the following parameters: $slope = 1.14$, $bias = 0.01$, $Q^2 = 0.99$, SECV = 0.14 (without optimization), $slope = 1.09$, $bias = 0.14$, $Q^2 = 0.98$, SECV = 0.14 (using genetic

algorithms), and *slope* = 0.99, *bias* = −0.07, $Q^2 = 0.99$, SECV = 0.08 (using differential evolution optimization).

Finally, using this model we can obtain new and finer predicted values for the six molecules. Refined predicted values for the activity of the six selected molecules are shown in Table 5. As we can observe in Table 5, these predicted values improve the initial estimated values obtained in the first step. With or without optimization algorithms, the predictions are improved obtaining activity values closer to the real ones.

Discussion and remarks

Prediction of activity of drugs using 2D-QSAR methods leads to many problems that are sometimes difficult to solve. The selection of an appropriate data set closely related to the characteristics of the molecules with unknown activity, the selection of the appropriate representational space, the set of molecular descriptors or the similarity approximation, as well as the statistical method used are some of the experimental conditions the researchers have to choose and adjust, often empirically.

In this paper we have proposed in 2D-QSAR a method based on the consideration of the structural characteristics of the molecule data set for the building of the representational space. Maximum common substructures extracted in the matching of pairwise molecules are used for the building of a hierarchical structure. This tree structure allows us to represent any pair of molecules as a path composed of the molecules and the non isomorphic substructures extracted in the same process as the building of the tree. Weighting these paths using a selected group of molecular descriptors enables the building of a 2D representational space that can be used, in a step-by-step process, to generate the QSAR model.

The use of optimization algorithms such as: genetic and differential evolution algorithms allow us to weigh up the contribution of the paths and molecular descriptors in the model, demonstrating an improvement in the model's robustness.

The proposed solution allows predicting unknown activity values easily, when regression is performed and even the estimated activity values can be later refined, showing the results obtained and a great improvement regarding the use of classical methods such as the descriptor's matrix.

References

1. Michielan L, Moro S (2010) Pharmaceutical perspectives of nonlinear QSAR strategies. J Chem Inf Model 50(6):961–978. doi:10.1021/ci100072z
2. Benigni R, Bossa C (2008) Predictivity of QSAR. J Chem Inf Model 48(5):971–980. doi:10.1021/ci8000088
3. Agrafiotis DK, Bandyopadhyay D, Wegner JK, van Vlijmen H (2007) Recent advances in chemoinformatics. J Chem Inf Model 47(4):1279–1293. doi:10.1021/ci700059g
4. Engel T (2006) Basic overview of chemoinformatics. J Chem Inf Model 46(6):2267–2277. doi:10.1021/ci600234z
5. Liu P, Agrafiotis DK, Rassokhin DN (2011) Power keys: a novel class of topological descriptors based on exhaustive subgraph enumeration and their application in substructure searching. J Chem Inf Model 51(11):2843–2851. doi:10.1021/ci200282z
6. Sun H, Shahane Shsng, Xia M, Austin CP, Huang R (2012) A Structure Based Model for the Prediction of Phospholipidosis Induction Potential of Small Molecules. Journal of Chemical Information and Modeling. doi:10.1021/ci3001875
7. Medina-Franco JL, Yongye AB, Pérez-Villanueva J, Houghten RA, Martínez-Mayorga K (2011) Multitarget structure–activity relationships characterized by activity-difference maps and consensus similarity measure. J Chem Inf Model 51(9):2427–2439. doi:10.1021/ci200281v
8. Su B-H, Y-s Tu, Esposito EX, Tseng YJ (2012) Predictive toxicology modeling: protocols for exploring hERG classification and tetrahymena pyriformis end point predictions. J Chem Inf Model 52(6):1660–1673. doi:10.1021/ci300060b
9. Hsieh J-H, Yin S, Wang XS, Liu S, Dokholyan NV, Tropsha A (2011) Cheminformatics meets molecular mechanics: a combined application of knowledge-based pose scoring and physical force field-based hit scoring functions improves the accuracy of structure-based virtual screening. J Chem Inf Model 52(1):16–28. doi:10.1021/ci2002507
10. Al-Sha'er MA, Taha MO (2010) Elaborate ligand-based modeling reveals new nanomolar heat shock protein 90α inhibitors. J Chem Inf Model 50(9):1706–1723. doi:10.1021/ci100222k
11. Urbano Cuadrado M, Luque Ruiz I, Gómez-Nieto MA (2006) Refinement and use of the approximate similarity in QSAR models for benzodiazepine receptor ligands. J Chem Inf Model 46(5):2022–2029
12. Sheridan RP (2012) Three useful dimensions for domain applicability in QSAR models using random forest. J Chem Inf Model 52(3):814–823. doi:10.1021/ci300004n
13. Petrone P, Simms B, Nigsch F, Lounkine E, Kutchukian P, Cornett A, Deng Z, Davies J, Jenkins J, Glick M (2012) Rethinking molecular similarity: comparing compounds on the basis of biological activity. ACS Chem Biol 7(8):1399–1409
14. Cronin MTD, Schultz TW (2003) Pitfalls in QSAR. J Mol Struct (Thoechem) 622(1–2):39–51. doi:10.1016/s0166-1280(02)00616-4
15. Sanders MPA, Barbosa AJM, Zarzycka B, Nicolaes GAF, Klomp JPG, de Vlieg J, Del Rio A (2012) Comparative analysis of pharmacophore screening tools. J Chem Inf Model 52(6):1607–1620. doi:10.1021/ci2005274
16. Zaretzki J, Rydberg P, Bergeron C, Bennett KP, Olsen L, Breneman CM (2012) RS-predictor models augmented with SMARTCyp reactivities: robust metabolic regioselectivity predictions for nine CYP isozymes. J Chem Inf Model 52(6):1637–1659. doi:10.1021/ci300009z
17. Rivera-Borroto OM, Marrero-Ponce Y, García-de la Vega JM, Grau-Ábalo RC (2011) Comparison of combinatorial clustering methods on pharmacological data sets represented by machine learning-selected real molecular descriptors. J Chem Inf Model 51(12):3036–3049. doi:10.1021/ci2000083
18. Ewing T, Baber JC, Feher M (2006) Novel 2D fingerprints for ligand-based virtual screening. J Chem Inf Model 46(6):2423–2431. doi:10.1021/ci060155b
19. Senese CL, Duca J, Pan D, Hopfinger AJ, Tseng YJ (2004) 4D-fingerprints, universal QSAR and QSPR descriptors. J Chem Inf Comput Sci 44(5):1526–1539. doi:10.1021/ci049898s

20. Pan D, Iyer M, Liu J, Li Y, Hopfinger AJ (2004) Constructing optimum blood brain barrier QSAR models using a combination of 4D-molecular similarity measures and cluster analysis. J Chem Inf Comput Sci 44(6):2083–2098. doi:10.1021/ci0498057
21. Sciabola S, Morao I, de Groot MJ (2006) Pharmacophoric fingerprint method (TOPP) for 3D-QSAR modeling: application to CYP2D6 metabolic stability. J Chem Inf Model 47(1):76–84. doi:10.1021/ci060143q
22. Cerruela García G, Luque Ruiz I, Gómez-Nieto MAn (2011) Analysis and study of molecule data sets using snowflake diagrams of weighted maximum common subgraph trees. J Chem Inf Model 51(6):1216–1232. doi:10.1021/ci100484z
23. Urbano Cuadrado M, Luque Ruiz I, Gómez-Nieto MÁ (2006) A steroids QSAR approach based on approximate similarity measurements. J Chem Inf Model 46(4):1678–1686
24. Cuadrado MU, Ruiz IL, Gómez-Nieto MA (2007) QSAR models based on isomorphic and nonisomorphic data fusion for predicting the blood brain barrier permeability. J Comput Chem 28(7): 1252–1260
25. Varnek A, Baskin I (2012) Machine learning methods for property prediction in chemoinformatics: quo vadis? J Chem Inf Model 52(6):1413–1437. doi:10.1021/ci200409x
26. Baroni M, Costantino G, Cruciani G, Riganelli D, Valigi R, Clementi S (1993) Generating optimal linear PLS estimations (GOLPE): an advanced chemometric tool for handling 3D-QSAR problems. Quant Struct-Act Relat 12(1):9–20. doi:10.1002/qsar.19930120103
27. O'Hara-Mays P (1997) Genetic algorithms in molecular modeling. In: James Devillers (eds) Principles of QSAR and Drug Design, vol 1. Academic Press, Harcourt Brace & Company: New York, 1996. 327 pp. ISBN 0-12-213810-4. $55.00. Journal of Chemical Information and Computer Sciences 37 (6):1204-1205. doi:10.1021/ci970394m
28. Hao M, Li Y, Wang Y, Yan Y, Zhang S (2011) Combined 3D-QSAR, molecular docking, and molecular dynamics study on piperazinyl-glutamate-pyridines/pyrimidines as potent P2Y12 antagonists for inhibition of platelet aggregation. J Chem Inf Model 51(10):2560–2572. doi:10.1021/ci2002878
29. Mercader AG, Duchowicz PR, Fernández FM, Castro EA (2011) Advances in the replacement and enhanced replacement method in QSAR and QSPR theories. J Chem Inf Model 51(7):1575–1581. doi:10.1021/ci200079b
30. Polanski J, Bak A, Gieleciak R, Magdziarz T (2005) Modeling robust QSAR. J Chem Inf Model 46(6):2310–2318. doi:10.1021/ci050314b
31. Nicolotti O, Carotti A (2005) QSAR and QSPR studies of a highly structured physicochemical domain. J Chem Inf Model 46(1):264–276. doi:10.1021/ci0502931
32. Mwense M, Wang XZ, Buontempo FV, Horan N, Young A, Osborn D (2004) Prediction of noninteractive mixture toxicity of organic compounds based on a fuzzy set method. J Chem Inf Comput Sci 44(5):1763–1773. doi:10.1021/ci0499368
33. Ghosh P, Bagchi MC (2009) QSAR modeling for quinoxaline derivatives using genetic algorithm and simulated annealing based feature selection. Curr Med Chem 16(30):4032–4048. doi:10.2174/092986709789352303
34. Leach AG, Jones HD, Cosgrove DA, Kenny PW, Ruston L, MacFaul P, Wood JM, Colclough N, Law B (2006) Matched molecular pairs as a guide in the optimization of pharmaceutical properties; a study of aqueous solubility, plasma protein binding and oral exposure. J Med Chem 49(23):6672–6682. doi:10.1021/jm0605233
35. Papadatos G, Alkarouri M, Gillet VJ, Willett P, Kadirkamanathan V, Luscombe CN, Bravi G, Richmond NJ, Pickett SD, Hussain J, Pritchard JM, Cooper AWJ, Macdonald SJF (2010) Lead optimization using matched molecular pairs: inclusion of contextual information for enhanced prediction of hERG inhibition, solubility, and lipophilicity. J Chem Inf Model 50(10):1872–1886. doi:10.1021/ci100258p
36. Prajapati K, Singh S, Pathak AK, Mehta P (2011) QSAR analysis on some 8-methoxy quinoline derivatives as H37RV (MTB) inhibitors. Int J ChemTech Res 3(1):408–422
37. Bagchi MC, Maiti BC, Bose S (2004) QSAR of anti tuberculosis drugs of INH type using graphical invariants. J Mol Struct (Thoechem) 679(3):179–186. doi:10.1016/j.theochem.2004.04.013
38. Price K, Storn RM, Lampinen JA (2005) Differential evolution: a practical approach to global optimization (natural computing series). Springer, New York
39. Hussain J, Rea C (2010) Computationally efficient algorithm to identify matched molecular pairs (MMPs) in large data sets. J Chem Inf Model 50(3):339–348. doi:10.1021/ci900450m
40. Raymond JW, Watson IA, Mahoui A (2009) Rationalizing lead optimization by associating quantitative relevance with molecular structure modification. J Chem Inf Model 49(8):1952–1962. doi:10.1021/ci9000426
41. Sheridan RP, Hunt P, Culberson JC (2005) Molecular transformations as a way of finding and exploiting consistent local QSAR. J Chem Inf Model 46(1):180–192. doi:10.1021/ci0503208
42. Birch AM, Kenny PW, Simpson I, Whittamore PRO (2009) Matched molecular pair analysis of activity and properties of glycogen phosphorylase inhibitors. Bioorg Med Chem Lett 19(3):850–853. doi:10.1016/j.bmcl.2008.12.003
43. Vargyas M, Csizmadia F (2008) Hierarchical clustering of chemical structures by maximum common substructures. Noordwijkerhout, The Netherlands, pp 1–5
44. Daylight Toolkit v4.94. Daylight Chemical Information Services Inc. http://www.daylight. 2010
45. Cerruela García G, Luque Ruiz I, Gómez-Nieto MA (2004) Step-by-step calculation of all maximum common substructures through a constraint satisfaction based algorithm. J Chem Inf Comput Sci 44(1):30–41
46. Fechner N, Jahn A, Hinselmann G, Zell A (2009) Atomic local neighborhood flexibility incorporation into a structured similarity measure for QSAR. J Chem Inf Model 49(3):549–560. doi:10.1021/ci800329r
47. Steffen A, Kogej T, Tyrchan C, Engkvist O (2009) Comparison of molecular fingerprint methods on the basis of biological profile data. J Chem Inf Model 49(2):338–347. doi:10.1021/ci800326z
48. Pandey G, Saxena AK (2006) 3D QSAR studies on protein tyrosine phosphatase 1B inhibitors: comparison of the quality and predictivity among 3D QSAR models obtained from different conformer-based alignments. J Chem Inf Model 46(6):2579–2590. doi:10.1021/ci600224n
49. Roy K, Leonard JT (2005) QSAR analyses of 3-(4-Benzylpiperidin-1-yl)-N-phenylpropylamine derivatives as potent CCR5 antagonists. J Chem Inf Model 45(5):1352–1368. doi:10.1021/ci050205x
50. Cuissart B, Touffet F, Crémilleux B, Bureau R, Rault S (2002) The maximum common substructure as a molecular depiction in a supervised classification context: experiments in quantitative structure/biodegradability relationships. J Chem Inf Comput Sci 42(5):1043–1052. doi:10.1021/ci020017w
51. Dimitrov S, Dimitrova G, Pavlov T, Dimitrova N, Patlewicz G, Niemela J, Mekenyan O (2005) A stepwise approach for defining the applicability domain of SAR and QSAR models. J Chem Inf Model 45(4):839–849. doi:10.1021/ci0500381
52. Shi LM, Fang H, Tong W, Wu J, Perkins R, Blair RM, Branham WS, Dial SL, Moland CL, Sheehan DM (2000) QSAR models using a large diverse set of estrogens. J Chem Inf Comput Sci 41(1):186–195. doi:10.1021/ci000066d
53. Maggiora GM, Johnson MA, Lajiness MS, Miller AB, Hagadone TR (1988) Looking for buried treasures: the search for

new drug leads in large chemical databases. Math Comput Model 11:626–629. doi:10.1016/0895-7177(88)90568-7
54. Deb K (2000) An efficient constraint handling method for genetic algorithms. Comput Methods Appl Mech Eng 186(2–4):311–338. doi:10.1016/s0045-7825(99)00389-8
55. Tsoulos IG (2008) Modifications of real code genetic algorithm for global optimization. Appl Math Comput 203(2):598–607. doi:10.1016/j.amc.2008.05.005
56. Andre J, Siarry P, Dognon T (2001) An improvement of the standard genetic algorithm fighting premature convergence in continuous optimization. Adv Eng Softw 32(1):49–60. doi:10.1016/s0965-9978(00)00070-3
57. Deep K, Singh KP, Kansal ML, Mohan C (2009) A real coded genetic algorithm for solving integer and mixed integer optimization problems. Appl Math Comput 212(2):505–518. doi:10.1016/j.amc.2009.02.044
58. JChem, version 5.3.7. Chemaxon Ltd (2010)
59. Palacios-Bejarano B, Luque-Ruiz I, Gomez-Nieto MA An Open Environment to Support the Development of Computational Chemistry Solutions im AIP Conference Proceedings. In: AIP Conference Proceedings, 2009. vol 1. pp 519–522