molecular informatics

DOI: 10.1002/minf.200((full DOI will be filled in by the editorial staff))

New pharmacophore fingerprints and Weight-Matrix Learning for virtual screening. Application to Bcr-Abl data.

Hajar Rehioui,[a] Bertrand Cuissart,[a] Abdelkader Ouali,[a] Alban Lepailleur,[b] Jean-Luc Lamotte,[a,b] Ronan Bureau,[b]* and Albrecht Zimmermann[a]

Abstract : We propose here to analyze the potential of a new type of pharmacophoric descriptors coupled with an original feature transformation technique, called Weight-Matrix Learning (WML, Feed Forward Neural Network). The application concerns virtual screening on a tyrosine kinase named BCR-ABL. Firstly, the compounds were described using three different families of descriptors: our new pharmacophoric descriptors and two radial-based fingerprints, ECFP4 and FCFP4. Then, each of these three original molecular depictions were transformed by using either an unsupervised WML method or a supervised one. Finally, using these transformed representations, K-Means clustering algorithm has been applied to partition the considered molecules. Combining our pharmacophoric descriptors to supervised Weight-Matrix Learning ($SWML_R$) leads to clearly superior results in terms of several quality measures.

Keywords : Pharmacophore, Molecular fingerprint, Feed Forward Neural Network, Clustering

1. Introduction

Recently, a new way to characterize a chemical data set has been defined based on Emerging Pharmacophores (EPs)[1]. An EP is a pharmacophore whose occurrence frequency inside one part of a data set is significantly greater than outside of it.

Each EP is associated to a growth rate value corresponding to the ratio of the frequencies between two categories associated to compounds (active *vs* inactive, for example). The first pharmacophore networks were constructed from representative EPs (called MMRFS) and the resulting pharmacophore spaces were based on similarities (graph edit distances) between the EPs [2].

Until now, we had never considered the Frequent Pharmacophores (FrPs) initially generated from a chemical data set, *i.e* before the calculation of EPs, as potential new descriptors. Thus, the aim of this publication is to take all the FrPs associated with a minimum number of compounds and to explore the structure-activity relationships towards a biological receptor starting from these descriptors. As a case study, we consider the same data set previously extracted from the ChEMBL database[2]. This data set relates to a tyrosine kinase named BCR-ABL, an oncogene involved in chronic myeloid leukemia.

To process this data, we rely on clustering techniques. Clustering is a machine learning method that groups objects considering how similar their descriptions are. Members of the same cluster must be similar to each other while being different from members of other clusters. Clustering algorithms associated with chemical data and chemoinformatics have been described in several publications[3–7]. The first step is to define a similarity matrix between the compounds. The latter matrix is usually computed from chemical fingerprints[7] such as circular-based fingerprints which consider the entire structure without pre-definition of fragments[8]. A circular-based fingerprint iteratively encodes features that represent each heavy atom in larger and larger structural neighborhoods, up to a given diameter. In this study, we chose to consider the global descriptors associated to the frequent pharmacophores (FrPs). For comparison purposes, we also used two circular-based fingerprints: ECFP4[9] for Extended Connectivity Fingerprints and FCFP4 for Functional Connectivity Fingerprints[8].

Feature transformations, like principal component analysis, are well-known techniques to optimize a machine learning process[10]. Here, besides considering the original data, the potential of two recent approaches is analyzed, approaches linked to a feed-forward neural network. The first one is called Weight-Matrix Learning (named WML) and is based on unsupervised learning metrics, identified as an off-center technique[11]. The underlying idea is to obtain a chemical representation in a new feature space so that ligands with similarity above a threshold are closer to each other and ligands with similarity below the same threshold are farther apart.

[a] *GREYC, Normandie Univ., UNICAEN, CNRS – UMR 6072, 14000 Caen, France*

[b] *Centre d'Etudes et de Recherche sur le Médicament de Normandie, Normandie Univ, UNICAEN, CERMN, 14000 Caen, France. *ronan.bureau@unicaen.fr: , +33231566820:*

Supporting Information for this article is available on the WWW under www.molinf.com

Based on WML, we propose a supervised way to transform the data. This second feature transformation method is called SWML for Supervised Weight-Matrix[11,12]. The main idea is to use the prior knowledge on the ligands (active *vs* inactive) by optimizing a given objective function, in our case categorical cross entropy. The learning process leads to a learned matrix such that the ligands from the same class are close to each other while those from different classes are moved away from each other.

The importance played by each feature in the previously discussed process is computed. The purpose of feature importance is to explain machine learning models and, in our case, to give an interpretation of the final clustering by providing the main descriptors involved in characterizing clusters. Several methods exist in this field such as Local Interpretable Model-agnostic Explanations (LIME)[13], Deep Learning Important FeaTures (DeepLIFT)[14], and SHapley Additive exPlanations (SHAP)[15]. The cited methods all belong to the post hoc interpretability[16] which means that they extract information from already learned models: it is a kind of `explanation-by-justification'. Recently, SHAP has been used to interpret relevant chemical features, and proved its effectiveness in the field of drug discovery[17–20]. Shapley values [21] were introduced in the 50s to measure the contributions of individual players to a collaborative game. This concept has been applied to feature contributions [15] by considering a team's success as an outcome (prediction), and each player's contribution as the feature importance. The idea behind the method is to create a parallel model g which aims to explain the predictions of a model f.

In summary, one compared three sets of descriptors (FrP, ECFP4, FCFP4) through their own capability to distinguish between active and inactive BCR ABL inhibitors. The comparison was done by analyzing both the performances obtained from the original data and the performances obtained from the data resulting of the application of a feature weight transformation. In addition, the predictive quality of the different models was analyzed by considering decoys linked to BCR-ABL ligands. The obtained results show that the SWML transformation applied to FrP is effective. Furthermore, a feature analysis based on the SHAP method was performed on the representative pharmacophores named MMRFS [14] for our chemical data set to identify key pharmacophores.

This paper is organized as follows : Section 2 introduces the materials and methods used to achieve our objective with a description of our methods of treatment of the chemical descriptors, Section 3 presents and discusses the main results and Section 4 concludes this work and draws up some perspectives.

2. Materials and Methods

2.1. Data set.

The BCR-ABL data set used in this work includes 1479 compounds described in our earlier publication[2]. Briefly, this ~~dataset~~ data set was collected from ChEMBL with the following restrictions: (i) only K_i and IC_{50} values expressed in nM units from biochemical assays reported in CHEMBL_24 (CHEMBL1862 : Target CHEMBL ID) were accepted as bioactivity data; (ii) measurements containing symbols such as ">" or "<" were not included unless they agreed with the threshold value (e.g., "<10 nM" would be retained as a means to identify an active molecule in the case of a 100 nM activity threshold); and (iii) if more than one bioactivity measurement was provided for the same molecule, we included the lowest K_i or the lowest IC_{50} value if no K_i was available. Duplicates were filtered and additional adjustments were performed (***e.g.***, compounds with molecular weight greater than or equal to 800 g/mol are eliminated, removal of salts, standardization of chemical functions, addition of hydrogens at the heteroatoms, and conversion to a two-dimensional [2D] spatial data file [SDF] format) using Pipeline Pilot (BIOVIA, San Diego, CA, USA) components. The molecules exhibiting K_i or IC_{50} values less than or equal to 100 nM were considered to be active compounds (n = 773); molecules with K_i or IC_{50} values greater than or equal to 1000 nM were considered to be inactive (n = 706). We created this substantial gap between active ***versus*** inactive molecules to maintain clear differentiation between the two groups.

In order to test the learning process, decoys were used. The decoys are generated from the original ligands and are all considered as inactive molecules. In our case we used the 10885 decoys associated to ABL1 data on DUD-E[22].

2.2. Molecular features

2.2.1. Pharmacophores.

The extraction of the frequent pharmacophores (FrP) is described in our previous publications[1,2]. Pharmacophore features correspond to generalized functionalities that are involved in favourable interactions between ligands and targets, including hydrogen-bond acceptors (A) and donors (D), negatively (N) and positively (P) charged ionizable groups, hydrophobic regions (H), and aromatic rings (R). Figure 1 represents the transformation of a molecule from its 2D molecular structure into a pharmacophore graph. When a pharmacophore graph is included into a molecule, one say that it occurs into the molecule. A FrP is a pharmacophore graph whose number of occurrences in the data set exceeds a given threshold. The minimal support for FrP was set to 10 in this study: a pharmacophore graph has to occur in at least 10 molecules to be considered as a FrP. We identified all FrPs with three to seven pharmacophoric features. With this setting, 111976 FrPs were generated.

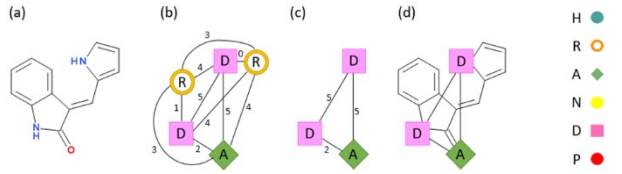


Figure 1. Transformation of a molecule (a) into its pharmacophore graph (b). (c) provides an example of a graph which occurs in the pharmacophore graph of (b). The corresponding fit of the three-point FrP in the molecule (d). The representative symbol for each pharmacophore feature is indicated on the right.

2.2.2. ECFP4 and FCFP4.

ECFP4 and FCFP4 fragments were extracted from a Pipeline Pilot protocol. From our data set, 7237 and 7100 descriptors were generated with ECFP4 and FCFP4, respectively.

2.2.3. Equivalence Classes (EC) for pharmacophores and fingerprints.

The goal of this pre-processing step is to eliminate redundancy and to retain only relevant data based on the standard Equivalence Classes (EC) techniques[23]. To this end, equivalent descriptors must be merged into a representative one. Herein, two descriptors are equivalent if they occur in the exact same set of molecules. With FrP, this method allowed us to reduce 86% of descriptors without losing any statistical information (from 111976 FrP to 15046 ECs). For molecular fingerprints, the reduction is lower with 3283 ECs for ECFP4 (45% of initial ECFP4) and 3269 ECs for FCFP4 (46% of initial FCFP4).

It should be emphasized that, in the rest of the article, all the operations will be carried out on the descriptions obtained after the latter EC pruning rather than on the initially computed chemical features (pharmacophores or fragments). In summary, from any of the three initial descriptions, exactly one descriptor per EC will be considered as a feature

2.3. WML and SWML feature transformation

All methods discussed in this subsection were implemented in Python.

2.3.1. WML

WML is a feature transformation method based on a feed-forward neural network (FFNN[24]). WML is an off-center technique: when its center is set to a similarity of 0.5, WML seeks for a transformation that brings two elements closer as soon as their similarity exceeds 0.5 while separating them otherwise. Such a transformation helps to reduce the difficulty of the learning task in the absence of labeled data. By this method, we aim to learn a matrix that transforms the chemical features of the original space into a new feature space. In the new feature space, ligands with a similarity larger than 0.5 will be closer to each other and ligands with a similarity smaller than 0.5 will be farther away.

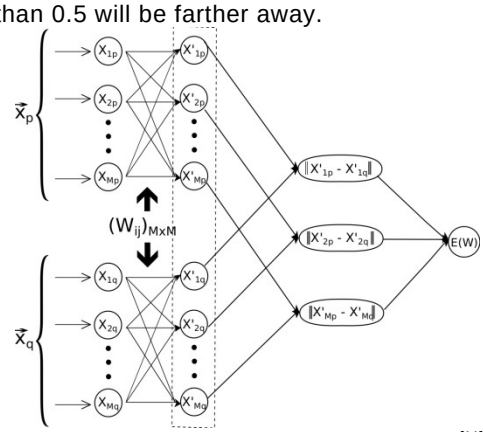


Figure 2. Network representation for WML[11]

The network structure of Figure 2 gives an explanation for WML. WML seeks for a linear transformation denoted as W. As the objective function called E(W) quantifies a loss, the latter must be minimized. For more information, see the main reference of Dasen et al. [11].

Given two ligands p and q whose initial representations are denoted $\vec{x}_p$ and $\vec{x}_q$, the calculation of their similarity is performed as in Equation 1:

$$\rho_{pq}^{(W)} = \frac{1}{1+\beta \cdot d_{pq}^{(W)}}$$

Equation 1

Where $\rho_{pq}^{(W)}$ is the similarity between the two ligands after the transformation W. $d_{pq}^{(W)}$ is a Mahalanobis distance[25] measure defined by Equation 2:

$$d_{pq}^{(W)} = \left(\vec{x}_p - \vec{x}_q\right)^T \left(W^T W\right)\left(\vec{x}_p - \vec{x}_q\right)$$

Equation 2

This distance corresponds to the squared euclidean distance when $W = I$, where I is an identity matrix.

β is a positive number calculated from Equation 3:

$$\frac{2}{N(N-1)} \sum_{q>p} \rho_{pq}^{(I)} = 0.5$$

Equation 3

N is the number of ligands in the data set, $\rho_{pq}^{(I)}$ is the value of $\rho_{pq}^{(w)}$ when $W = I$, i.e. the similarity before any transformation. The role of the parameter β is to balance the data distribution in order to have an average similarity of the sample around 0.5 (see Figure 3 for FrP).

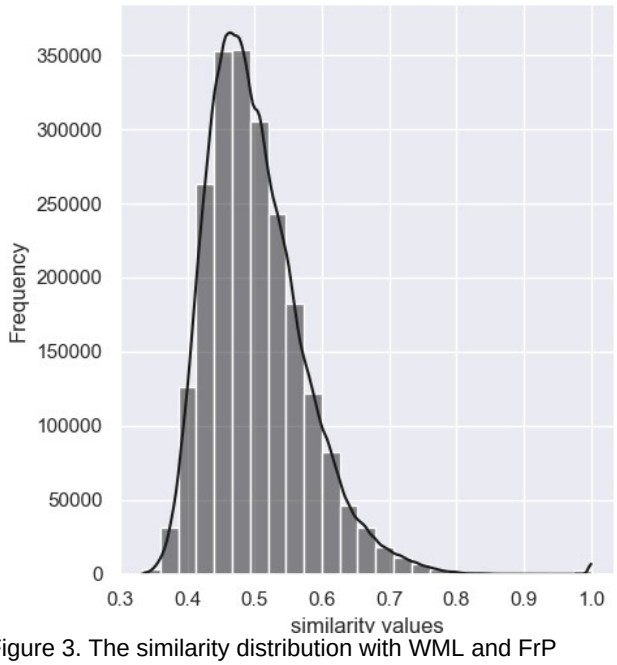


Figure 3. The similarity distribution with WML and FrP

Once the similarity matrix is calculated, the uncertainty of this similarity matrix is reduced by minimizing the loss function $E(W)$ in Equation 4:

$$E(W)=\frac{1}{N(N-1)}\sum_{q<p}\left[\rho_{pq}^{(W)}\left(1-\rho_{pq}^{(I)}\right)+\rho_{pq}^{(I)}\left(1-\rho_{pq}^{(W)}\right)\right]$$

Equation 4

2.3.2. SWML

SWML uses a supervised learning metrics also based on an FFNN. Here, the search for the best feature transformation is a supervised method that relies on the activity of ligands. The main idea is to use the activity of ligands for minimizing the *categorical crossentropy (CCE), a* loss function defined by Equation 5. In the latter, N is the number of ligands, C is the set of predefined labels (active and inactive), y_i^k is valued at 1 when the training example i belongs to class k, otherwise y_i^k is valued at 0. $model(x_i,k)$ represents the predicted class (the output of the model). The goal of SWML is to separate active ligands from inactive ones as much as possible, by seeking for a model that minimizes CCE.

$$CCE=-\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{C}y_i^k\log\left(model(x_i,k)\right)$$

Equation 5

The architecture of the SWML, as illustrated in Figure 5, is composed of one input layer and one hidden layer with *M-neurons* for the input layer and c-neurons for the output layer. The number of neurons corresponds to the dimension (number of properties) of the used representation, where the number of output neurons *c* corresponds to the number of classes (in our case two activity classes).

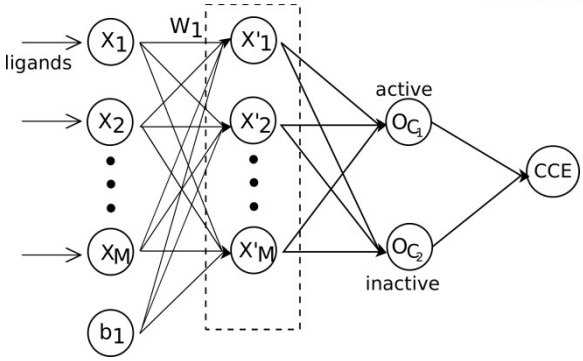


Figure 4. Network representation for SWML.

2.3.3. Feed forward Neural Network (FNN)

WML and SWML, use a FFNN implemented with TensorFlow[26] and Keras[27].

Both feature transformations seeks for a transformation W that converts the original space of N ligands described by M features into a new space. Subsequently, the learned weights and biases used in the training phase of the FFNN are used to perform the transformation of the initial feature representation (see Equation 6 where X' is the transformed data, X the original data, W the trained matrix of weights and b is the vector of bias).

$$X'=X.W+b$$

Equation 6

This transformation could increase, preserve or decrease the number of descriptors. We have chosen here to consider only the transformations that maintain the initial number of descriptors.

In the FFNN, an epoch denominates one forward pass and one backward pass of all the training examples of the learning process; the learning rate is a positive value, in the range between 0.0 and 1.0, controlling the convergence of the model by optimizing a loss function.

In this study the learning rate was set to 0.0005 and the maximum number of epochs was set to 100. To choose the optimal number of epochs we opted for the use of early stopping and the model check point techniques. As a stochastic gradient descent method, we choose Adam optimization[28], which are well suited for problems using large data, to reach the optimal value of the loss function. The sigmoid[29] was selected as the activation function, and the batch size was set to 256. The loss functions are described previously.

As SWML is a supervised method, overfitting is possible. We have therefore successively applied two particular methods. Dropout [30,31] is a technique that prevents overfitting by temporarily removing neurons (in input or hidden layers) in a neural network based on a rate p_r. In Figure 5, as the rate is set to 0.5, 50 % of the neurons are removed.

The choice of the dropped neurons is made randomly for each training step.

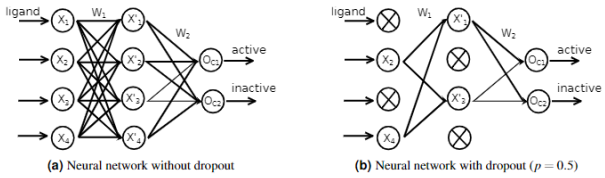


Figure 5 The neural network (a) before applying the dropout and (b) after applying the dropout with a rate p = 0.5.

The L2 regularization[32,33] is another method to overcome overfitting. The idea of L2 is to add a regularization term to the loss function so that $Loss_{total}$, defined in Equation 7, is minimized in the model training phase instead of the $Loss_{model}$.

$$Loss_{total} = Loss_{model} + \lambda Loss_{L2}$$

Equation 7

$Loss_{model}$. is the chosen loss function to train the model, $\lambda \in [0,1]$ is a regularization parameter that controls the trade-off between $Loss_{model}$ and $Loss_{L2}$, while the $Loss_{L2}$, as defined in Equation 8, represents the sum of squares of all weights w_i in the model with M for the number of features.

$$Loss_{L2} = \sum_{i=1}^{M} w_i^2$$

Equation 8

In our case, we use two dropout layers combined to the L2 regularization, an input dropout of 20% rate and a second dropout of 50% rate after the hidden layer [29].

The application of both methods leads to a new symbol that denotes SWML followed by the two regularizations: $SWML_R$.

2.4. Clustering with K-means

K-means [34] is one of the most used clustering algorithms. The strength of K-means lies in its simplicity and ability to group large data sets. Let *D* be the learning set. Each cluster is a subset of *D* represented by a centroid, initialized by a randomly chosen point of the data set. In order to fill the clusters, each element x of D is assigned to the nearest centroid c_k, belonging to $cluster_k$ as follows in Equation 9 :

$$x \in cluster_k \Longleftrightarrow dist(x, c_k) = argmin_{1 \le j \le K} dist(x, c$$

Equation 9

After each iteration, the centroids are recalculated and updated taking into account each new assignment.

We emphasize that we use K-means because of its simplicity and its wide use [35], but any clustering algorithm could be used instead of K-means. The Euclidean distances was chosen for the original and transformed data[36]. K-means and its evaluation metrics were implemented by using the Scikit-learn Python library[34].

2.4.1. Predictive Clustering

For predictive clustering of a new ligand after an initial clustering with K-means, the ligand is assigned to the cluster that minimizes the similarity between its centroid and the ligand. By adding the points in each iteration, the centroids are recalculated and updated taking into account each new point. The presented K-means process is applied on the training data after its transformation as presented in Equation 10.

$$T(x_{train_i}) \in cluster_{train_k} \Longleftrightarrow dist(T(x_{train_i}), c_{train_k}) = argmin_{1 \le j \le K}$$

Equation 10

Where $x_{train_i} \in D_{train}, D_{train}$ is the training partition of data D, $T(x_{train_i})$ is the transformation of x_{train_i} by the model T, c_{train_k} is the centroid belonging to $cluster_{train_k}$.

In order to predict the membership of a new ligand to an already constructed clusters, this ligand is assigned to the cluster which minimizes the similarity between its centroid and this point[37] as defined in Equation 11.

$$T(x_{test_i}) \in cluster_{train_k} \Longleftrightarrow dist(T(x_{test_i}), c_{train_k}) = argmin_{1 \le j \le}$$

Equation 11

Where $x_{test_i} \in D_{test}, D_{test}$ is the training partition of data D, $T(x_{test_i})$ is the transformation of x_{test_i} by the model T, c_{train_k} is the centroid belonging to $cluster_{train_k}$.

We emphasize that predictive clustering is used to cluster test folds in the cross validation process and also to cluster decoys.

2.4.2. Evaluation metrics

In addition to a classical confusion matrix[38], two evaluation metrics, Normalized mutual information[39] (NMI) and Silhouette[40], are computed for each partition considered during the clustering process. NMI is a quality measure that compares the resulting clusters with a given classification, the latter being considered as a ground truth. The results vary between 0 (no mutual information) and 1 (perfect correlation). Silhouette is a function that averages intra-cluster distances and inter-clusters distances. The best value is 1 and the worst is -1. Values close to 0 indicate that the clusters overlap. Negative values usually indicate that samples was assigned to the wrong cluster.

2.5. Feature importance: SHAP[15]

The purpose of feature importance is to explain machine learning models. As defined in Equation 12 the sum of the contributions of all the features must be equal to the prediction (output) of the model *f(x)*.

$$f(x) = g(x') = \varnothing_0 + \sum_{i=1}^{M} \varnothing_i x_i'$$

Equation 12

Where M is the number of input features, $x' \in \{0,1\}^M$ is a binary vector indicating the presence or absence of a feature, x' is called the interpretable representation of *x*. The base value $\varnothing_0$ represents the mean value of the predictions of the model *f*. In the SHAP algorithm, the contribution of a feature i, denoted $\varnothing_i$, is a real value, commonly used to identify how much a feature i influenced a model's prediction.

The **articles** of Lundberg and Lee[15] provides ~~a~~ precise definition and ~~a detailled~~ computation of $\varnothing_i$.

SHAP values are calculated by using the SHAP Python library.[15][43]

3. Results and Discussion

3.1. Descriptors and clustering

To have a first view of the evolution of the initial ~~dataset~~ data set and the similarities between compounds, Figure 6 shows the Multidimensional Scaling[44] projection of the three situations (Original, WML, $SWML_R$) and for the three descriptors (FrP, ECFP4, FCFP4).

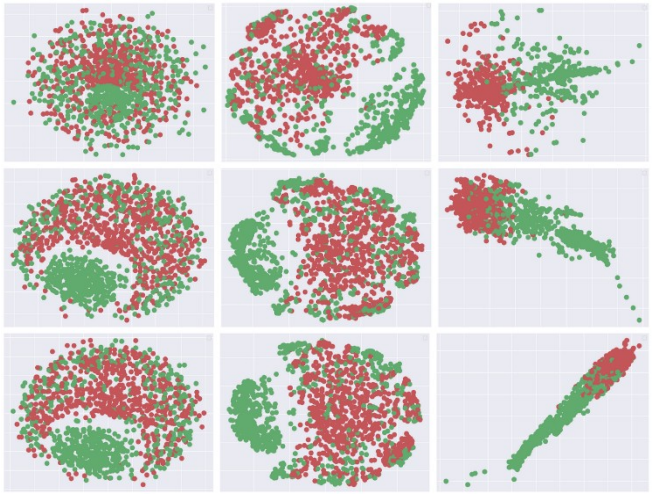


Figure 6. MDS projections for the three descriptors (FrP (up), ECFP4 (middle), FCFP4 (down)). Left : Original representation. Middle: WML. Right: $SWML_R$. Active compounds are in red, inactive in green.

$SWML_R$ leads to a more compact representation and a clearer separation between the two classes. The Table 1 records the results of full trained data based on original data, WML and $SWML_R$ transformations. $SWML_R$ leads to the best clustering whatever the descriptors. FrP outperforms ECFP4 and FCFP4 and the best configurations were obtained from k=2 to 5 (NMI and misclassified compounds) with an optimum for k = 3 (see Table 1, orange cells). The stability of $SWML_R$ was analyzed by a 5-fold cross validation technique by splitting the data into 80 % of training folds and 20 % of testing folds (see Table 2, k = 3). Misclassified compounds represent 12% in this case, compared to 6,5% for the full data. A comparison of SWML and $SWML_R$ has only been shown for k=3 in Table 1. SWML performs better for misclassified compounds but not for silhouette. Also, an overfitting is clearly possible with SWML. Along with this result, this phenomenon was analyzed on decoys (see below). In the end, we chose to keep $SWML_R$/FrP with k=3 (K-means) in the following studies.

3.2. Composition of the clusters

The composition of the clusters (SWMLR/FrP with k=3) is summarized in Table 3. Cluster0 is associated with inactive compounds (91% inactive). Cluster1 is an active cluster and only the compound **1** (trained data, FRP, ECFP4, FCFP4) is inactive in this subset (IC_{50} = 1000 nM). The closest compound of **1** in this cluster is the compound **2** with an IC_{50} of 62 nM (see Figure 7). This activity cliff has already been described and discussed in our previous publication.[1]

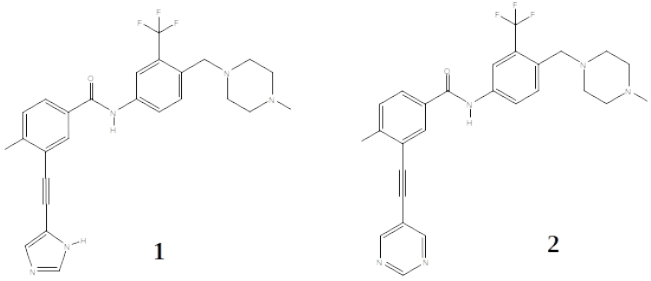


Figure 7. Inactive *vs* active compound in cluster1.

Cluster2 is an active cluster (90% active) with 31 compounds misclassified. Compound **3** is a typical example of these misclassifications with an IC_{50} of 11000 nM compared to a close derivative in this cluster, the compound **4**, with an IC_{50} of 70 nM (see Figure 8).

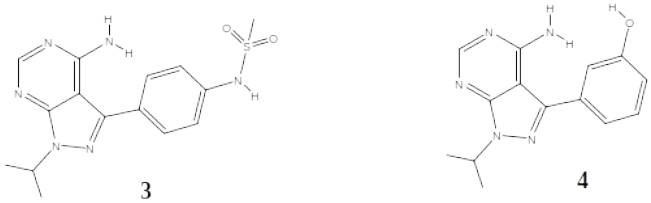


Figure 8. Inactive vs active compound in cluster2

Representative compounds of the two active clusters were analyzed with Maximum dissimilarity clustering (clusters with members similarity inferior or equal to 0.45 (distance Tanimoto, ECFP4)). We got 43 clusters (see Figure 9, for 25 cluster centers with a minimum number of compounds of 5 in each cluster among the 43 clusters) for the cluster1 and 95 clusters for cluster2 with a lower number of compounds compared to cluster1 (see Figure 10, for 22 cluster centers with a minimum numbers of compounds of 5 in each cluster among the 95 clusters). This reflects the greater diversity of compounds for cluster2 compared to cluster1.

Table 1. Quality data for original and feature transformation data in function of the type of descriptors.

k clusters	data	FrP			ECFP4			FCFP4		
		NMI	Silhouette	misclassified	NMI	Silhouette	misclassified	NMI	Silhouette	misclassified
k=2	Original	0.280	0.035	449	0.328	0.081	394	0.328	0.082	379
	WML	0.318	0.398	393	0.315	0.338	396	0.320	0.351	396
	$SWML_R$	0.558	0.553	177	0.450	0.708	261	0.421	0.708	290
k=3	Original	0.297	0.055	395	0.292	0.102	339	0.271	0.097	394
	WML	0.253	0.383	394	0.263	0.355	374	0.277	0.345	341
	SWML	0.678	0.349	38	0.623	0.294	76	0.589	0.281	108
	$SWML_R$	0.546	0.438	96	0.440	0.603	171	0.430	0.599	179
k=4	Original	0.252	0.085	387	0.252	0.113	338	0.256	0.112	337
	WML	0.257	0.411	331	0.222	0.355	369	0.236	0.344	353
	$SWML_R$	0.516	0.450	108	0.414	0.489	195	0.392	0.529	173
k=5	Original	0.257	0.081	395	0.270	0.119	329	0.270	0.119	318
	WML	0.229	0.407	331	0.210	0.353	364	0.207	0.359	359
	$SWML_R$	0.497	0.465	110	0.379	0.456	191	0.362	0.475	189
k=6	Original	0.256	0.087	381	0.238	0.126	319	0.236	0.123	317
	WML	0.225	0.384	290	0.206	0.367	364	0.212	0.372	348
	$SWML_R$	0.454	0.364	117	0.342	0.389	178	0.345	0.468	185
k=7	Original	0.224	0.091	357	0.240	0.138	320	0.226	0.087	329
	WML	0.221	0.401	291	0.196	0.367	327	0.202	0.393	350
	$SWML_R$	0.397	0.344	128	0.325	0.374	179	0.312	0.413	192
k=8	Original	0.213	0.107	372	0.221	0.091	283	0.213	0.086	314
	WML	0.211	0.434	293	0.190	0.331	324	0.213	0.399	331
	$SWML_R$	0.394	0.281	129	0.318	0.351	172	0.314	0.385	177
k=9	Original	0.291	0.004	318	0.221	0.091	249	0.214	0.100	281
	WML	0.198	0.408	293	0.189	0.354	323	0.170	0.346	331
	$SWML_R$	0.370	0.305	137	0.318	0.352	179	0.305	0.378	177

Table 2. Crossvalidation data on $SWML_R$.

k clusters	Metrics	fold0	fold1	fold2	fold3	fold4	Average (folds)
k=3	NMI	0.438	0.470	0.459	0.440	0.387	0.439
	Silhouette	0.516	0.369	0.436	0.516	0.554	0.478
	Misclassified	43	28	29	42	46	38

Table 3. Repartition of the compounds into the three clusters (k = 3) for the initial data set and for the decoys with FRP, ECFP4 and FCFP4

	ligands	Initial data set			Decoys		
		cluster0	cluster1	cluster2	cluster0	cluster1	cluster2
FRP	Inactive	674	1	31	10864	0	20
	Active	64	426	283	0	0	0
ECFP4	Inactive	661	1	44	698	4031	6156
	Active	132	421	220	0	0	0
FCFP4	Inactive	663	1	42	214	7760	2911
	Active	125	424	224	0	0	0

Figure 9. View of representative compounds of cluster1 (5 compounds by cluster minimum).

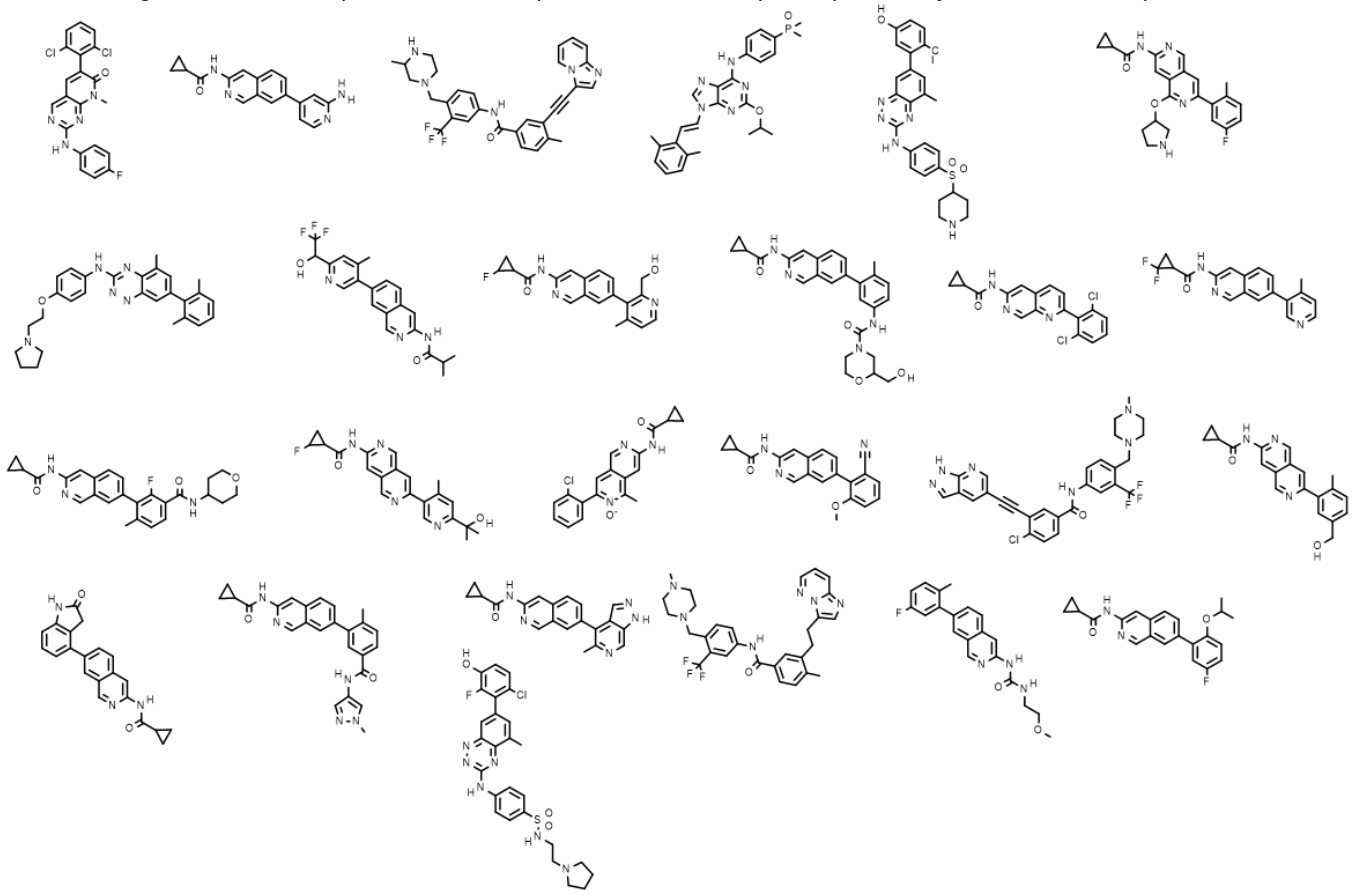
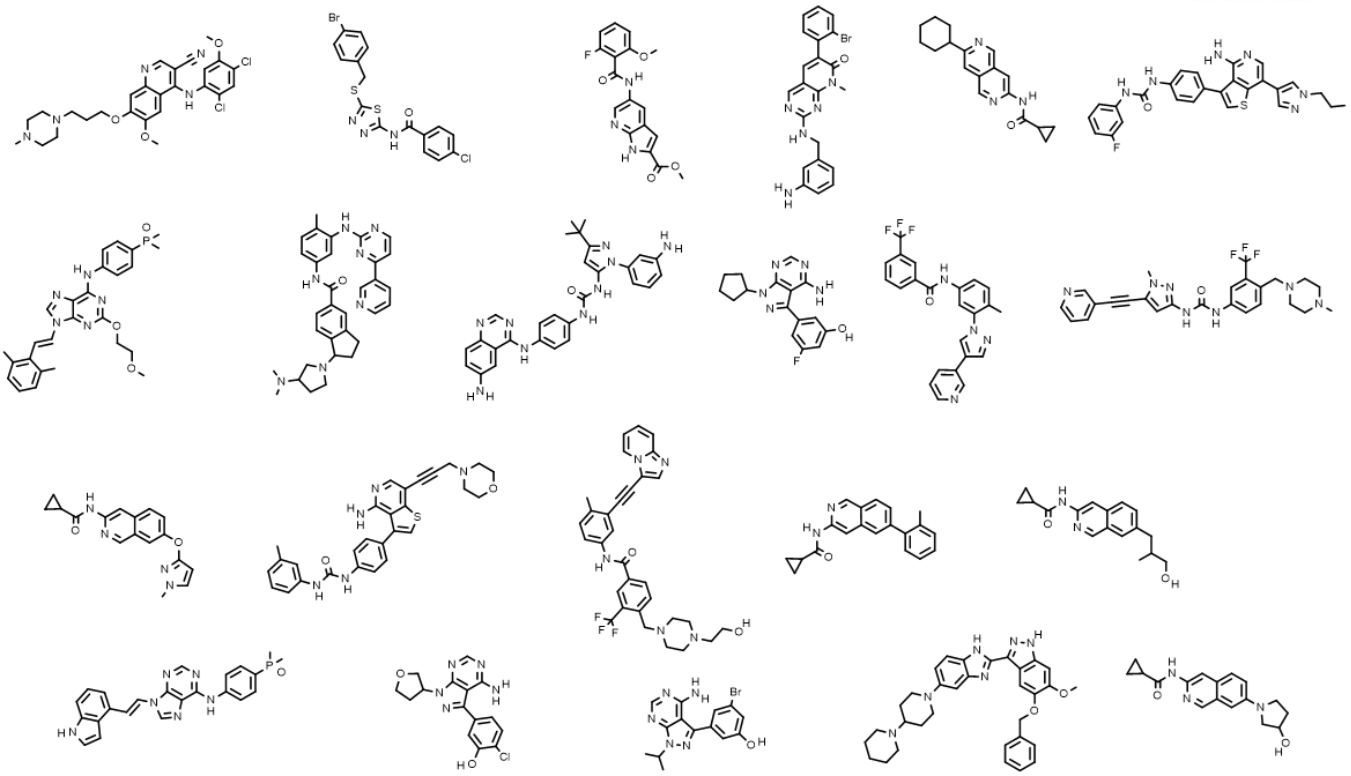


Figure 10. View of representative compounds of cluster2 (5 compounds by cluster minimum)



The Minimum (MinDistance), maximum (MaxDistance) and average distances (AvgDistance) for all pairs of molecules in the active cluster (ECFP4, Tanimoto Distance) were calculated (see Table 4). With an average distance of 0.65, the chemical diversity of cluster1 is lower than cluster2.

Table 4.

	MinDistance	MaxDistance	AvgDistance
Cluster 1	0.02	0.92	0.65
Cluster2	0.02	0.96	0.81

3.3. Feature importance

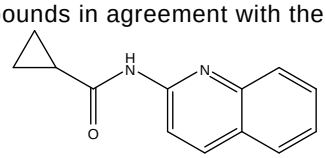
The definition of the main pharmacophores associated with the three clusters was explored with the notion of feature importance *via* SHAP. To clarify the interpretation of these results, we added a label named Growth Rate (GR) to FrP, label in agreement with our previous studies[1,2]. The GR of a pharmacophore corresponds to the ratio between its frequency of fit within the active molecules and its frequency of fit within the inactive molecules. From the 15046 FrP, the extraction of the main features (pharmacophores) with SHAP is difficult (see Erreur : source de la référence non trouvée with $M = 15046$) but feasible. We obtained a pharmacophore ranking for each cluster but the first (among the 15046 FrPs) seems insignificant (differentiation between the clusters) considering the average GR values for each cluster and the small extensions (number of compounds) associated with each pharmacophore. To decrease the number of pharmacophores for the SHAP analysis and also in agreement with our previous studies, we calculated representative pharmacophores named MMRFS[2] with three to seven pharmacophoric features for our chemical data set. For this definition, we have considered both classes (actives *vs* inactives and the inverse[2]) because GR is the main basis for the selection of MMRFS pharmacophores (initial classification of pharmacophores in function of the GR values and the extension[1]). These processes lead to 312 MMRFS pharmacophores. To verify the performance of these new 312 descriptors, $SWML_R$ was applied followed by a clustering with k = 3, as for FrP. The results are close to $SWML_R$/FrP (see Table 5).

The first three MMRFS pharmacophores selected with SHAP, for each cluster, have an active recall value (the part of active molecules (IC_{50} <= 100 nM) which are predicted as active) of 0.46 for cluster0, 0.60 for cluster1 and 0.94 for cluster2. The same three MMRFS pharmacophores, reversing the notion of active compounds, have an inactive recall value (the part of inactive molecules (IC_{50} >= 1000 nM) which are predicted as inactive) of 0.75 for cluster0, 0.0057 for cluster1 and 0.56 for cluster2. These results help to understand the differences between the three clusters (see Table 6) and their specific classes (active or inactive). If we consider the first 50 MMRFS pharmacophores, we observe the same trend by calculating the average values of the normalized GR (GR_{Nor}).

$GR_{N\vee ¿=(GR/(GR+1))¿}$

Indeed, GR_{Nor} is 0.34 for cluster0, 0.83 for cluster1 and 0.63 for cluster2. Considering this last data and previous results for prediction, cluster1 is associated with pharmacophores of higher orders, thus leads to larger pharmacophoric/structural constraints than cluster2.

For the MMRFS pharmacophores selected with SHAP, number one (SHAP1) is the same MMRFS for cluster0 and cluster2. It covers 745 molecules (56% of the inactive molecules) and it is often associated with an aromatic group linked to an amide or amidine function. For cluster1, SHAP1 is a pharmacophore close to the main pharmacophore of cluster C1[2] and pharmacophore P1[1] in our previous publications. It covers 374 molecules (4 inactive compounds). Always in cluster1, SHAP2 is associated with a typical subfamily of our training set with a very close scaffold for 289 compounds in agreement with the scaffold **7**.



7

The last MMRFS pharmacophore for cluster1 (SHAP3), associated to 378 molecules, gives only 4 new active molecules (and new scaffolds) compared to SHAP1. SHAP1 and SHAP2 of cluster1 correspond to SHAP2 and 3 of cluster2. SHAP1-3 of cluster2 covers 94% of active molecules but with 56% coverage of inactive molecules.

3.4. Prediction on decoys

In order to better assess the predictive capability of the clustering and the supervised learning of the $SWML_R$, the classification of the 10885 decoys (all potentially inactives) associated to ABL1 data (DUD-E[22]), were analyzed for each of the three different descriptors. The results are depicted in Table 3, Table 5 (MMRFS) and Figure 11.

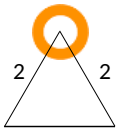
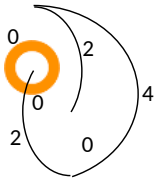
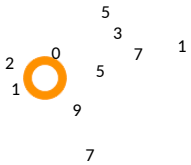
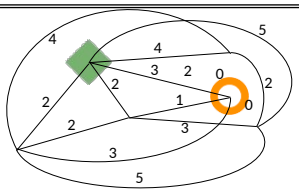
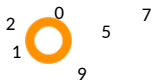
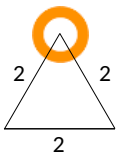
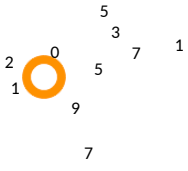
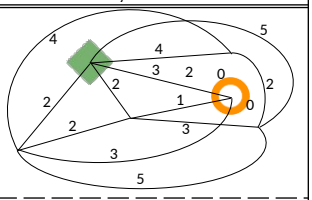
FrP outperforms ECFP4 and FCFP4. Only 20 decoys are associated with an active cluster with FrP which is very surprising compared to the other descriptors. SWML and $SWML_R$ lead globally to the same results (see Figure 11, n = 96 for misclassified compounds with SWML/FrP). For an explanation, FrP are more precise than a structural fragment with, for each FrP, at least three pharmacophoric features with precise distances. The other point is the fact than DUDE decoys were defined to analyze the quality of docking studies. They must physically resemble to ligands but be topologically dissimilar. In the final decoy definition procedure, ECFP4 fingerprints were generated for real ligands and potential decoys. Decoys were sorted based on their maximum similarity to any ligand, and the most dissimilar 25% were retained through this dissimilarity filter[22].

.

Table 5. Repartition of the compounds into the three clusters (k = 3) for the initial data set and for the decoys with MMRFS.

	ligands	Initial data set			Active for decoys		
		cluster0	cluster1	cluster2	cluster0	cluster1	cluster2
MMRFS	Inactive	642	0	64	9991	0	894
	Active	112	384	277	0	0	0

Table 6. First three MMRFS pharmacophores for each cluster with SHAP (see Figure 1 for an explanation of the symbols). The recall value indicated corresponds to the contribution of each pharmacophore to the recall for the three pharmacophores. The recall values are additives from the first pharmacophore to the third.

	SHAP1	SHAP2	SHAP3
Cluster0			
Recall_active	0.45	0.46	0.46
Recall_inactive	0.56	0.72	0.75
Cluster1			
Recall_active	0.48	0.59	0.60
Recall_inactive	0,0057	0,0057	0,0057
Cluster2			
Recall_active	0.45	0.82	0.94
Recall_inactive	0.56	0.5637	0.5637

We think that a higher dissimilarity value, for the definition of decoys, could lead to a different result in our study with ECFP4 and FCFP4 for our analysis. It is surprising to see the distribution of the decoys into cluster1 and cluster2 in a large majority (only 209 in cluster0 for FCFP4).

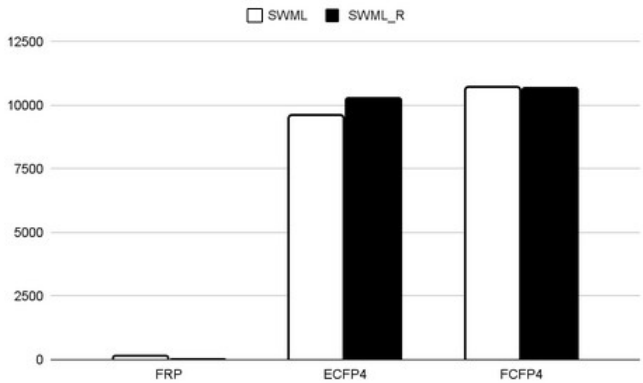


Figure 11. Number of misclassified compounds SWML (White) and SWMLR for decoys with the three descriptors.

Decoy prediction was also performed with MMRFS in conjunction with SHAP (see Table 5). The performances compared to FrP are logically reduced but still remains very good (759 compounds misclassified). An example is shown in Figure 12 where compound **5** (ZINC49918351), one of the 20 misclassified compounds with $SWML_R$/FrP, is close to an active compound **6** (CHEMBL1164265, IC_{50} = 100 nM) in cluster2.

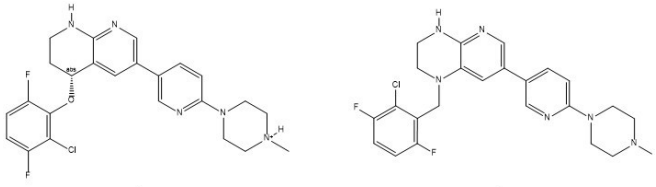


Figure 12. ZINC49918351 *vs* CHEMBL1164265

4. Conclusions

The definition of a new type of pharmacophoric descriptors called FrP leads to convincing results with machine learning techniques associating feature transformations with FNN and classical clustering. In our dataset, supervised feature transformation with regularization ($SWML_R$) performs best. The definition of three clusters (FRP / SWMLR), for BCR-ABL data, allows to obtain good predictive results with our initial data set and with decoys. In the end, on the three clusters, the two active clusters are different in terms of constraints with FrP. SHAP analysis (defining feature importance) shows that most MMRFS pharmacophores associated with cluster1 have high values for GR (more discriminating pharmacophores). The results by the same techniques with ECFP4 and FCFP4 are disappointing especially for decoys. This new and original approach with FrP is promising for virtual screening of chemical databases. This work is a first step toward an interactive learning process, in which we will focus on discovering the capabilities of our pharmacophores to understand the behaviour of chemicals.

5. Acknowledgements

Our work is part of the European project “SCHISM”, funded by the European Union within the framework of the Operational Programme ERDF/ESF 2014-2020. We thank the Regional Council of Normandie for financial support.

6. References

[1] J.-P. Métivier, B. Cuissart, R. Bureau, A. Lepailleur, *J. Med. Chem.* **2018**, *61*, 3551–3564.

[2] D. Geslin, A. Lepailleur, J.-L. Manguin, N.-V. Vo, J.-L. Lamotte, B. Cuissart, R. Bureau, *J. Chem. Inf. Model.* **2022**, *62*, 678–691.

[3] P. Willett, J. M. Barnard, G. M. Downs, *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 983–996.

[4] J. M. Barnard, G. M. Downs, *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 644–649.

[5] V. J. Gillet, D. J. Wild, P. Willett, J. Bradshaw, *Comput. J.* **1998**, *41*, 547–558.

[6] G. M. Downs, P. Willett, W. Fisanick, *J. Chem. Inf. Comput. Sci.* **1994**, *34*, 1094–1102.

[7] J. D. MacCuish, N. E. MacCuish, *WIREs Comput. Mol. Sci.* **2014**, *4*, 34–48.

[8] A. Cereto-Massagué, M. J. Ojeda, C. Valls, M. Mulero, S. Garcia-Vallvé, G. Pujadas, *Methods* **2015**, *71*, 58–63.

[9] D. Rogers, M. Hahn, *J. Chem. Inf. Model.* **2010**, *50*, 742–754.

[10] Y.-C. Lo, S. E. Rensi, W. Torng, R. B. Altman, *Drug Discov. Today* **2018**, *23*, 1538–1546.

[11] D. Yan, X. Zhou, X. Wang, R. Wang, *Inf. Sci.* **2019**, *503*, 635–651.

[12] D. S. Yeung, X. Z. Wang, *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 556–561.

[13] M. T. Ribeiro, S. Singh, C. Guestrin, *ArXiv160204938 Cs Stat* **2016**.

[14] A. Shrikumar, P. Greenside, A. Kundaje, *ArXiv170402685 Cs* **2019**.

[15] S. Lundberg, S.-I. Lee, *ArXiv170507874 Cs Stat* **2017**.

[16] F. K. Došilović, M. Brčić, N. Hlupić, in *2018 41st Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO*, **2018**, pp. 0210–0215.

[17] C. Esposito, S. Wang, U. E. W. Lange, F. Oellien, S. Riniker, *J. Chem. Inf. Model.* **2020**, *60*, 4730–4749.

[18] J. Jiménez-Luna, F. Grisoni, G. Schneider, *Nat. Mach. Intell.* **2020**, *2*, 573–584.

[19] R. Rodríguez-Pérez, J. Bajorath, *J. Med. Chem.* **2020**, *63*, 8761–8777.

[20] “Improving Docking-Based Virtual Screening Ability by Integrating Multiple Energy Auxiliary Terms from Molecular Docking Scoring | Journal of Chemical Information and Modeling,” can be found under https://pubs.acs.org/doi/10.1021/acs.jcim.9b00977, **n.d.**

[21] *Contributions to the Theory of Games (AM-28), Volume II*, **1953**.

[22] M. M. Mysinger, M. Carchia, John. J. Irwin, B. K. Shoichet, *J. Med. Chem.* **2012**, *55*, 6582–6594.

[23] P. Fournier-Viger, T. Gueniche, S. Zida, V. S. Tseng, in *Adv. Intell. Data Anal. XIII* (Eds.: H. Blockeel, M. van Leeuwen, V. Vinciotti), Springer International Publishing, Cham, **2014**, pp. 108–119.

[24] T. Gupta, “Deep Learning: Feedforward Neural Network,” can be found under https://towardsdatascience.com/deep-learning-feedforward-neural-network-26a6705dbdc7, **2018**.

[25] *Sankhya A* **2018**, *80*, 1–7.

[26] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, *ArXiv160304467 Cs* **2016**.

[27] F. Chollet, others, “Keras,” can be found under https://github.com/fchollet/keras, **2015**.

[28] D. P. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization*, ArXiv, **2017**.

[29] S. Sharma, S. Sharma, A. Athaiya, *Int. J. Eng. Appl. Sci. Technol.* **2020**, *04*, 310–316.

[30] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. R. Salakhutdinov, *ArXiv12070580 Cs* **2012**.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

[32] B. J. McCaffrey, 10/05/2017, “Neural Network L2 Regularization Using Python -,” can be found under https://visualstudiomagazine.com/articles/2017/09/01/neural-network-l2.aspx, **n.d.**

[33] E. Phaisangittisagul, in *2016 7th Int. Conf. Intell. Syst. Model. Simul. ISMS*, **2016**, pp. 174–179.

[34] J. MacQueen, *Proc. Fifth Berkeley Symp. Math. Stat. Probab. Vol. 1 Stat.* **1967**, *5.1*, 281–298.

[35] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, D. Steinberg, *Knowl. Inf. Syst.* **2007**, *14*, 1–37.

[36] E. Martin, E. Cao, *J. Comput. Aided Mol. Des.* **2015**, *29*, 387–395.

[37] B. Yang, X. Fu, N. D. Sidiropoulos, M. Hong, *ArXiv161004794 Cs* **2017**.

[38] S. Visa, B. Ramsay, A. L. Ralescu, E. Van Der Knaap, *MAICS* **2011**, *710*, 120–127.

[39] P. A. Estevez, M. Tesmer, C. A. Perez, J. M. Zurada, *IEEE Trans. Neural Netw.* **2009**, *20*, 189–201.

[40] P. J. Rousseeuw, *J. Comput. Appl. Math.* **1987**, *20*, 53–65.

[41] S. M. Lundberg, G. G. Erion, S.-I. Lee, *Consistent Individualized Feature Attribution for Tree Ensembles*, ArXiv, **2019**.

[42] "From local explanations to global understanding with explainable AI for trees | Nature Machine Intelligence," can be found under https://www.nature.com/articles/s42256-019-0138-9, **n.d.**

[43] "Welcome to the SHAP documentation — SHAP latest documentation," can be found under https://shap.readthedocs.io/en/latest/index.html, **n.d.**

[44] A. Mead, *J. R. Stat. Soc. Ser. Stat.* **1992**, *41*, 27–39.

Received: ((will be filled in by the editorial staff))
Accepted: ((will be filled in by the editorial staff))
Published online: ((will be filled in by the editorial staff))