scientific reports

OPEN

Calculation of exact Shapley values for explaining support vector machine models using the radial basis function kernel

Andrea Mastropietro[1], Christian Feldmann[2] & Jürgen Bajorath[2]✉

Machine learning (ML) algorithms are extensively used in pharmaceutical research. Most ML models have black-box character, thus preventing the interpretation of predictions. However, rationalizing model decisions is of critical importance if predictions should aid in experimental design. Accordingly, in interdisciplinary research, there is growing interest in explaining ML models. Methods devised for this purpose are a part of the explainable artificial intelligence (XAI) spectrum of approaches. In XAI, the Shapley value concept originating from cooperative game theory has become popular for identifying features determining predictions. The Shapley value concept has been adapted as a model-agnostic approach for explaining predictions. Since the computational time required for Shapley value calculations scales exponentially with the number of features used, local approximations such as Shapley additive explanations (SHAP) are usually required in ML. The support vector machine (SVM) algorithm is one of the most popular ML methods in pharmaceutical research and beyond. SVM models are often explained using SHAP. However, there is only limited correlation between SHAP and exact Shapley values, as previously demonstrated for SVM calculations using the Tanimoto kernel, which limits SVM model explanation. Since the Tanimoto kernel is a special kernel function mostly applied for assessing chemical similarity, we have developed the Shapley value-expressed radial basis function (SVERAD), a computationally efficient approach for the calculation of exact Shapley values for SVM models based upon radial basis function kernels that are widely applied in different areas. SVERAD is shown to produce meaningful explanations of SVM predictions.

Machine learning (ML) is a key component of computer-aided drug discovery[1,2]. Fast-growing volumes of chemical and biological discovery data provide a sound basis for the derivation of ML models for practical applications. The data deluge also causes a need for predictive modeling in support of experimental programs. In early-phase drug discovery, many ML applications focus on prediction of candidate compounds with desired biological activity[3–5]. In interdisciplinary research, it is usually required to rationalize predictions for experimental design. However, with the exception of linear regression or simple decision tree models, most ML methods have a black-box character[6], that is, model decisions cannot be understood by humans, which often limits the impact of ML on experimental programs. Consequently, there is increasing interest in approaches to rationalize ML predictions, which belong to the spectrum of explainable artificial intelligence (XAI) methods[7,8]. For example, among other approaches, methods for model explanation often assess the contributions of input features and identify features that determine predictions[9–18]. While interest in XAI is steadily increasing, the field is far from being mature and relevant approaches are often still in early exploratory stages, which also applies to the chemical sciences and drug discovery[12–15]. Importantly, XAI approaches should not only help domain experts to rationalize predictions, but model explanations should also be accessible to non-expert investigators in interdisciplinary settings[14,15].

An exemplary generally applicable XAI method is *local interpretable model-agnostic explanations* (LIME)[10], which determines feature importance for an ML model by performing a local permutation of the input data and deriving a linear approximation. Furthermore, the Shapley value concept[19] from collaborative game theory has been adapted for quantifying feature importance in ML. The Shapley value formalism was originally designed

[1]Department of Computer, Control and Management Engineering "Antonio Ruberti", Sapienza University of Rome, 00185 Rome, Italy. [2]Department of Life Science Informatics and Data Science, B-IT, LIMES Program Unit Chemical Biology and Medicinal Chemistry, Rheinische Friedrich-Wilhelms-Universität, Friedrich-Hirzebruch-Allee 5/6, 53115 Bonn, Germany. ✉email: bajorath@bit.uni-bonn.de

to determine the contributions of individual players of a team to the performance in a game and divide the gain among players accordingly[19]. Therefore, Shapley values are calculated to quantify the contribution of each player by considering all possible ordered player coalitions. In the XAI adaptation of the Shapley value concept, players correspond to features and the game is the prediction of a test instance. Given the need to enumerate and calculate the marginal contribution of a feature in each possible coalition, the computational requirements of Shapley value calculations scale exponentially with increasing feature numbers. Hence, Shapley value calculations become infeasible for ML models based upon large feature sets. Therefore, corresponding to LIME[10], an approximation of the Shapley value approach has been introduced, termed *Shapley additive explanations* (SHAP)[20], representing another model-agnostic approach. For a given non-linear ML model, SHAP derives local linear models in the feature space vicinity of test instances to approximate Shapley values. To this end, Monte Carlo sampling strategies have also been employed[21] and the SHAP formalism has been extended to cover graph-structured data and graph neural networks[22,23]. Compared to other feature importance methods adapted for the interpretation of quantitative structure–activity relationship (QSAR) models in cheminformatics, a hallmark of the Shapley value/SHAP approach is that it can quantify contributions of features that are present or absent in test instances to their prediction. This is a distinguishing characteristic of the Shapley value/SHAP methodology. Furthermore, algorithms enabling the calculation of exact Shapley values for large feature sets provide a principal advantage compared to local approximation methods. However, another methodology has recently been introduced to adjust the Shapley value formalism for individual data sets[24], providing an alternative approach compared to the calculation of exact Shapley values in ML.

In XAI, calculation of exact Shapley values has thus far only been accomplished for deriving local explanations of decision tree models[25] such as random forests (RFs)[26] and for the support vector machine (SVM) algorithm[27] in combination with the Tanimoto kernel[28,29], as recently reported[30]. The decision tree- and SVM-based Shapley value approaches were termed TreeSHAP (or TreeExplainer)[21] and *Shapley value-expressed Tanimoto similarity* (SVETA)[30], respectively. Both RF and SVM are for long among the most popular ML methods in pharmaceutical research and other scientific fields, which often rival or exceed the performance of deep neural networks on sets of structured data with well-defined features[15,25], for example, in molecular property predictions[2,15]. Accordingly, rationalizing SVM black-box predictions is also of considerable interest. Notably, there was only limited correlation between exact Shapley values calculated for the SVM/Tanimoto kernel combination and corresponding SHAP values, indicating that the local approximation might not be suitable for reliable model explanation in this case. Given that the Tanimoto kernel is a special kernel function mostly applied to account for chemical similarity[29], we devised a methodology for calculating exact Shapley values for SVM models based upon the more generally applied radial basis function (RBF) kernels (including the popular Gaussian kernel). Herein, we report the development and proof-of-concept application of the *Shapley value-expressed radial basis function* (SVERAD) approach yielding exact Shapley values for the SVM/RBF combination in a computationally efficient manner (requiring quadratic computational time with respect to the number of input features). Comparison of SVERAD and SHAP values revealed limited correlation, hence reinforcing the need for calculation of exact Shapley values to explain SVM predictions. As a part of our study, the SVERAD code is made freely available.

Results

Scope of the analysis

We first develop the theory and mathematical foundations of SVERAD and then demonstrate the calculation of exact Shapley values using SVERAD based on a model system. In addition, compound activity predictions are carried out using SVM and RF models, and features determining the predictions were identified with SVERAD (SVM), KernelSHAP[20] (SVM, RF), the general applicable SHAP approximation, and TreeSHAP[25] (RF). These calculations enabled a direct comparison of SVERAD and SHAP and an additional comparison of corresponding SVM and RF predictions and their explanations. Furthermore, features prioritized for SVM and RF predictions were mapped onto the structures of correctly predicted test compounds to complement numerical analysis and compare chemically intuitive graphical explanations. Finally, XAI analysis is complemented by computational complexity analysis for SVERAD.

The Shapley value concept

Shapley values represent the weighted average marginal contribution of a feature to a prediction considering all the possible feature coalitions[19,20]. Let $\mathcal{F}$ be the complete set of features and $\mathcal{S}$ a coalition of features (subset of $\mathcal{F}\backslash\{f\}$). The contribution ϕ_f is computed by considering the difference in the value v of the coalition $\mathcal{S}$ with and without the assessed feature f, weighted by the inverse multinomial coefficient $\binom{|\mathcal{F}|}{1,|\mathcal{S}|,|\mathcal{F}|-|\mathcal{S}|-1}^{-1}$, which is calculated as the number of permutations of the coalition ($|\mathcal{S}|$) multiplied by the number of features not contained in the coalition ($|\mathcal{F}|-|\mathcal{S}|-1$) and divided by the number of all possible feature permutations ($|\mathcal{F}|!$). This must be repeated and summed for all possible subsets $\mathcal{S}$ of the $\mathcal{F}\backslash\{f\}$ features, obtaining the following equation:

$$\phi_f(v) = \sum_{\mathcal{S}\subseteq\mathcal{F}\backslash\{f\}} \frac{|\mathcal{S}|!(|\mathcal{F}|-|\mathcal{S}|-1)!}{|\mathcal{F}|!}\left(v\left(\mathcal{S}\cup f\right) - v(\mathcal{S})\right)$$

Radial basis function kernel

SVM relies on kernel functions for implicitly mapping data distributions into higher-dimensional feature space representations if linear separation of data with different class labels is not possible in a given feature space ("kernel trick")[31]. For this purpose, alternative kernel functions can be used, depending on the particular applications. Our methodology considers the widely used RBF kernel defined as

$$K\left(\boldsymbol{x},\boldsymbol{x}'\right)=e^{-\frac{d\left(\boldsymbol{x},\boldsymbol{x}'\right)^2}{2\sigma^2}}$$

where $d\left(\boldsymbol{x},\boldsymbol{x}'\right)$ is the Euclidean distance between vectors $\boldsymbol{x}$ and $\boldsymbol{x}'$:

$$d\left(\boldsymbol{x},\boldsymbol{x}'\right)=\|\boldsymbol{x}-\boldsymbol{x}'\|=\sqrt{\sum_i\left(x_i-x_i'\right)^2}$$

The parameter σ is a free parameter used to control the level of nonlinearity of the SVM model that will determine the decision boundary. An alternative definition of the RBF uses the parameter $\gamma=\frac{1}{2\sigma^2}$, obtaining the equivalent equation

$$K\left(\boldsymbol{x},\boldsymbol{x}'\right)=e^{-\gamma\|\boldsymbol{x}-\boldsymbol{x}'\|^2}$$

Larger values of γ will lead to a more complex decision boundary, while smaller values will render it smoother.

Notably, the RBF function considered is the Gaussian RBF, as it is the most common function employed in kernelized methods and has become a standard in SVM implementations[27]. RBFs are a family of functions with radial symmetry; the Gaussian one is expressed as

$$\varphi(r)=e^{-\gamma r^2}$$

where r is the radial distance, which usually corresponds to the Euclidean distance (as in our case).

In pharmaceutical research, SVM models are mostly derived for molecular property predictions based on chemical structures and therefore employ structural features of compounds as input. Structural features are conventionally encoded in a binary vector format (often termed fingerprints)[32], that is, a feature can be present or absent in test instance, corresponding to bit settings of 1 or 0, respectively. In the cheminformatics domain, SVMs are currently essentially exclusively employed with binary fingerprint descriptors. Moreover, binary input vectors are also common for other SVM modeling tasks. Therefore, we consider binary encoding of features as a basis for Shapley value calculations (for integer-based representations, adjustments are required). Furthermore, we define I the number of intersecting (common) features between the two feature vectors and D the number of features in the symmetric difference (present in either one vector or the other). N_i and N_d will be the number of intersecting and symmetric difference features in a given coalition, respectively. As we show below, the computation of Shapley values using SVERAD only relies on the number of intersecting and symmetric difference features.

Shapely values for the radial basis function kernel

In order to express feature contributions as Shapley values via the SVERAD formalism, we first need to assess the contribution of features to the Euclidean distance. We notice that features with the same value (intersecting or absent features) do not increase the distance, in fact $\left(x_i-x_i'\right)=0$ if $x_i=x_i'$. Of course, this is true also for non-binary features. Then, features with different values (features with symmetric difference) increase $d\left(\boldsymbol{x},\boldsymbol{x}'\right)^2$ by $\Delta_d=(0-1)^2=(1-0)^2=1$. This leads to having $d\left(\boldsymbol{x},\boldsymbol{x}'\right)=\sqrt{N_d}$ and $d\left(\boldsymbol{x},\boldsymbol{x}'\right)^2=N_d$, indicating that only features with symmetric difference determine the distance (and kernel) value:

$$e^{-\frac{d(\boldsymbol{x},\boldsymbol{x}')^2}{2\sigma^2}}=e^{-\frac{N_d}{2\sigma^2}}$$

This allows for a fast calculation of the kernel value.

Now, we consider a coalition of features $\mathcal{S}$ whose value $v(\mathcal{S})$ is the RBF kernel value. If $\mathcal{S}$ contains intersecting features only ($N_d=0$) we have $v(\mathcal{S})=e^{-\frac{N_d}{2\sigma^2}}=e^0=1$. This is true for any value of I (size of the intersection). Differently, for a coalition with features with symmetric difference only (or with a mixture of intersecting and symmetric difference features), the value $v(\mathcal{S})=e^{-\frac{N_d}{2\sigma^2}}$ must be calculated given N_d and σ (or γ) , as aforementioned. Finally, for the empty coalition $\mathcal{S}=\varnothing$, we set $v(\mathcal{S})=0$, conforming to the Shapley value formalism for the empty set[16,17].

To obtain the Shapley values for the RBF kernel, we need to compute the change in the kernel value when a feature from the intersection f_+, or a feature from the symmetric difference f_-, are added to the coalition $\mathcal{S}$ with N_i intersecting features and N_d features with symmetric difference. For f_+ we have

$$\Delta v_{f_+}(N_i,N_d)=e^{-\frac{N_d}{2\sigma^2}}-e^{-\frac{N_d}{2\sigma^2}}=0$$

Adding a feature from the intersection does not change the distance and thus not the kernel value. This is always true except if f_+ is added to the empty coalition ($v(\varnothing)=0$). In this case, the kernel value when adding the features becomes 1 ($N_d=0$) and so

$$\Delta v_{f_+}(0,0)=1$$

Then, for a symmetric difference feature f_- we have

$$\Delta v_{f_-}(N_i, N_d) = e^{-\frac{N_d+1}{2\sigma^2}} - e^{-\frac{N_d}{2\sigma^2}}$$

When adding a feature with symmetric difference, the squared Euclidean distance increases by 1 (as shown). The change in the kernel value must be calculated consequently. When the subtracted term represents the empty coalition, its value is set to 0.

Once we have computed the value change, we need to calculate the number of occurrences for each possible coalition with N_i intersecting features and N_d features with symmetric difference. For f_+ we thus consider all possible combinations of N_i elements in a set of $I - 1$ elements (the assessed feature is not a part of the coalition) and N_d elements in a set of D elements:

$$C_{f_+}(N_i, N_d) = \binom{I-1}{N_i}\binom{D}{N_d}$$

Likely, for f_- we consider all possible combinations of N_i elements in a set of I elements and N_d elements in a set of $D - 1$:

$$C_{f_-}(N_i, N_d) = \binom{I}{N_i}\binom{D-1}{N_d}$$

Once we have all the elements, we can compute the Shapley values as the sum of the products of Δv_f, C_f and the inverse multinomial coefficient.

For an intersecting feature, the Shapley value (ϕ_f) for the RBF kernel will be computed as

$$\phi_{f_+} = \sum_{N_i=0}^{I-1}\sum_{N_d=0}^{D} \Delta v_{f_+}(N_i, N_d) \cdot C_{f_+}(N_i, N_d) \cdot \binom{I+D}{1, N_i + N_{d,}, I + D - N_i - N_d - 1}^{-1}$$

As previously shown, $\Delta v_{f_+}(N_i, N_d)$ is always 0, except if f_+ is added to the empty coalition ($N_i = 0$ and $N_d = 0$). In this case, the kernel value changes from 0 to 1, thus $\Delta v_{f_+}(0, 0) = 1$. So, we can easily compute the Shapley value considering only the addition to the empty coalition:

$$\begin{aligned}
\phi_{f_+} &= \Delta v_{f_+}(0,0) \cdot C_{f_+}(0,0) \cdot \binom{I+D}{1, N_i + N_d,\ I + D - N_i - N_d - 1}^{-1} \\
&= 1 \cdot 1 \cdot \frac{(N_i + N_d)!(I + D - N_i - N_d - 1)!}{(I + D)!} \\
&= \frac{(I + D - 1)!}{(I + D)!}
\end{aligned}$$

Analogously, for a symmetric difference feature, we obtain

$$\begin{aligned}
\phi_{f_-} &= \sum_{N_i=0}^{I}\sum_{N_d=0}^{D-1} \Delta v_{f_-}(N_i, N_d) \cdot C_{f_-}(N_i, N_d) \cdot \binom{I+D}{1, N_i + N_{d,},\ I + D - N_i - N_d - 1}^{-1} \\
&= \sum_{N_i=0}^{I}\sum_{N_d=0}^{D-1} \left(e^{-\frac{N_d+1}{2\sigma^2}} - e^{-\frac{N_d}{2\sigma^2}}\right) \cdot \binom{I}{N_i}\binom{D-1}{N_d} \cdot \frac{(N_i + N_d)!(I + D - N_i - N_d - 1)!}{(I + D)!} \\
&= \sum_{N_i=0}^{I}\sum_{N_d=0}^{D-1} \left(e^{-\frac{N_d+1}{2\sigma^2}} - e^{-\frac{N_d}{2\sigma^2}}\right) \cdot \frac{I!}{(I - N_i)!N_i!} \cdot \frac{(D-1)!}{(D - N_d - 1)!N_d!} \cdot \frac{(N_i + N_d)!(I + D - N_i - N_d - 1)!}{(I + D)!}
\end{aligned}$$

The computation can be further simplified by aggregating common factors. The possible coalitions to which f_- can be added include the empty coalition ($N_i = N_d = 0$), coalitions with intersecting features only ($N_d = 0$ and $v(\mathcal{S}) = 1$), and coalitions with intersecting and symmetric difference features, or with symmetric difference features only ($N_i \in [0, I]$ and $N_d \in [1, D - 1]$). We thus obtain

$$\begin{aligned}
\phi_{f_-} = {} & e^{-\frac{1}{2\sigma^2}} \cdot \frac{(I + D - 1)!}{(I + D)!} + \left(e^{-\frac{1}{2\sigma^2}} - 1\right) \cdot \sum_{N_i=1}^{I} \binom{I}{N_i} \cdot \frac{N_i!(I + D - N_i - 1)!}{(I + D)!} \\
& + \sum_{N_i=0}^{I}\sum_{N_d=1}^{D-1} \left(e^{-\frac{N_d+1}{2\sigma^2}} - e^{-\frac{N_d}{2\sigma^2}}\right) \cdot \binom{I}{N_i}\binom{D-1}{N_d} \cdot \frac{(N_i + N_d)!(I + D - N_i - N_d - 1)!}{(I + D)!}
\end{aligned}$$

Proof-of-concept

To establish initial proof-of-concept for the approach, we calculate Shapley values for the RBF kernel and two exemplary small model vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ using SVERAD:

$$\boldsymbol{x} = [10010], \boldsymbol{y} = [10111]$$

Notably, these vectors are only used to illustrate the SVERAD calculations and do not represent (high-dimensional) molecular fingerprints. The model vectors share two features (set to 1, intersecting features), so $I = 2$, have a unique feature each (set to 0 and 1, respectively, symmetric difference), so $D = 2$, and lack a feature (set to 0). For the exemplary calculation, we set $\sigma = 1$. Tables 1 and 2 show the steps needed to compute the Shapley values for intersecting and symmetric difference features, respectively.

As discussed, calculation of the kernel value only depends on the number of features with symmetric difference, resulting in equation

$$K(\boldsymbol{x}, \boldsymbol{y}) = e^{-\frac{N_d}{2}} = e^{-1} = 0.368$$

The sum of the Shapley values for all features yields the kernel value. To compute the Shapley value for a feature in the intersection and a feature with symmetric difference, Δv_f is multiplied by the number of coalitions and the inverse multinomial coefficient and the sum over all possible coalitions is calculated. Given that any feature from the same set (intersection or symmetric difference) makes the same contribution to the kernel value, we need to multiply the Shapley value obtained for one representative feature of each set by I and D to obtain the total contribution of the intersecting and symmetric difference features, respectively. In our example, the Shapley value for an intersecting feature is 0.25 and for a feature with symmetric difference it is -0.066. The set of intersecting features ($I = 2$) yields a sum of Shapley values of 0.5 while symmetric difference features ($D = 2$) contribute to the kernel value for -0.132. The sum of these values is 0.368, which is exactly the kernel value.

As an additional proof-of-concept calculation, we consider 20 random binary vectors with a small number of features ($|F| = 15$), so that Shapley values can be computed explicitly by enumerating all possible coalitions[19]. SVERAD yields the same Shapley values as produced by the exhaustive computation, thus demonstrating the validity of the method. This is also evident in Table 3, which shows a comparison of SVERAD Shapley values with the SHAP approximation (for the calculations, we set $\gamma = \frac{1}{2\sigma^2} = 1$).

The correlation coefficient of 1 for SVERAD Shapley values and exact Shapley values confirms that both calculations return the same values (the associated error resulting from the imprecision in the representation of very small numbers is smaller than 10^{-10}). This differs from exact Shapley values vs. SHAP, for which a Fisher-transformed correlation coefficient of 0.72 ± 0.43 is obtained, reflecting the underlying local approximation of SHAP values.

N_i	N_d	$v(\mathcal{S})$	$v(\mathcal{S} \cup f_+)$	Δv_f	# coalitions	Inverse multinomial coefficient	Δv_f · # coalitions · inv. mult. coeff
0	0	0	1	1	1 · 1 = 1	¼ = 0.25	0.25
0	1	$e^{-1/2}$	$e^{-1/2}$	0	1 · 2 = 2	1/12 = 0.083	0
0	2	e^{-1}	e^{-1}	0	1 · 1 = 1	1/12 = 0.083	0
1	0	1	1	0	1 · 1 = 1	1/12 = 0.083	0
1	1	$e^{-1/2}$	$e^{-1/2}$	0	1 · 2 = 2	1/12 = 0.083	0
1	2	e^{-1}	e^{-1}	0	1 · 1 = 1	¼ = 0.25	0

Table 1. Calculation of the Shapley value for an intersecting feature.

N_i	N_d	$v(\mathcal{S})$	$v(\mathcal{S} \cup f_-)$	Δv_f	# coalitions	Inverse multinomial coefficient	Δv_f · # coalitions · inv. mult. coeff
0	0	0	$e^{-1/2}$	$e^{-1/2}$	1 · 1 = 1	¼ = 0.25	0.1516
0	1	$e^{-1/2}$	e^{-1}	$e^{-1} - e^{-1/2}$	1 · 1 = 1	1/12 = 0.083	−0.0199
1	0	1	$e^{-1/2}$	$e^{-1/2} - 1$	2 · 1 = 2	1/12 = 0.083	−0.0656
1	1	$e^{-1/2}$	e^{-1}	$e^{-1}-e^{-1/2}$	2 · 1 = 2	1/12 = 0.083	−0.0398
2	0	1	$e^{-1/2}$	$e^{-1/2} - 1$	1 · 1 = 1	1/12 = 0.083	−0.0328
2	1	$e^{-1/2}$	e^{-1}	$e^{-1}-e^{-1/2}$	1 · 1 = 1	¼ = 0.25	−0.0597

Table 2. Calculation of the Shapley value for a symmetric difference feature.

	Exact Shapley values	SVERAD	SHAP
Exact Shapley values	1.0 ± 0.0	1.0 ± 0.0	0.72 ± 0.43
SVERAD	1.0 ± 0.0	1.0 ± 0.0	0.72 ± 0.43
SHAP	0.72 ± 0.43	0.72 ± 0.43	1.0 ± 0.0

Table 3. Comparison of exact Shapley values, SVERAD, and SHAP values using Pearson's *r* correlation coefficient with standard deviations.

It also follows that the predictive performance of original SVM models is not affected through the Shapley value modification because it exactly accounts for the RBF kernel value, as demonstrated above, and the SVM computational classification criteria do not change.

Shapley values for support vector machine predictions

In an SVM model, the distance of a vector $\boldsymbol{x}$ from the separating hyperplane is defined by the support vectors $\boldsymbol{V_n}$ and is given by

$$dist(\boldsymbol{x}) = b + \sum\nolimits_{n=0}^{N_v-1} y_n w_n K(\boldsymbol{x}, \boldsymbol{V_n})$$

where N_v is the number of support vectors, y_n (-1 or 1) is the class label of the support vector $\boldsymbol{V_n}$, w_n is the weight by which the class label is scaled and $K(\boldsymbol{x}, \boldsymbol{V_n})$ is the kernel value comparing the support vector and the predicted instance $\boldsymbol{x}$. Finally, b is a bias value.

To compute the Shapley value for the distance for each feature f, we first substitute the kernel value with the sum of Shapley values of the features for the RBF kernel between vector $\boldsymbol{x}$ and support vector $\boldsymbol{V_n}$ ($\phi_{f,n}$) and scale the sum by the label and the weight. Then, we consider the bias as an additional feature whose value b is its Shapley value:

$$\begin{aligned} dist(\boldsymbol{x}) &= b + \sum_{n=0}^{Nv-1} y_n w_n K(\boldsymbol{x}, \boldsymbol{V_n}) = b + \sum_{n=0}^{Nv-1} y_n w_n \sum_{f=0}^{|F|-1} \phi_{f,n} \\ &= b + \sum_{f=0}^{|F|-1} \sum_{n=0}^{Nv-1} y_n w_n \phi_{f,n} = \phi_b + \sum_{f=0}^{|F|-1} \phi_f \end{aligned}$$

Finally, given the additivity property of Shapley values, the Shapley value for a feature f is obtained by summing up the Shapley values of f for the RBF kernel values comparing vector $\boldsymbol{x}$ with all the support vectors:

$$\phi_f = \sum\nolimits_{n=0}^{Nv-1} y_n w_n \phi_{f,n},$$

which gives the contribution of feature f with respect to the distance from the separating hyperplane.

Expressing feature contributions as log odds values

The distance from the hyperplane can be transformed into probability estimates using Platt scaling[33]:

$$p(\boldsymbol{x}) = \frac{1}{1 + e^{A \cdot dist(\boldsymbol{x}) + B}}$$

Given that Shapley values for probabilities cannot be calculated from Shapley values for the distance from the hyperplane, we need to compute the logits (log odds):

$$logit\left(p(\boldsymbol{x})\right) = log\left(\frac{p(\boldsymbol{x})}{1 - p(\boldsymbol{x})}\right) = \cdots = log\left(\frac{1}{e^{A \cdot dist(\boldsymbol{x}) + B}}\right) = -A \cdot dist(\boldsymbol{x}) - B$$

We can express $dist(\boldsymbol{x})$ as the sum of the Shapley values for the distance:

$$logit\left(p(\boldsymbol{x})\right) = -A \cdot \left(\phi_b + \sum\nolimits_{f=0}^{|F|-1} \phi_f\right) - B = -(A \cdot \phi_b + B) - \sum\nolimits_{f=0}^{|F|-1} A \cdot \phi_f$$

Logits are a linear transformation of the distance. Hence, the Shapley values for the logits are obtained as a linear transformation of the Shapley values for the distance (scaling by $-A$). Moreover, by scaling ϕ_b by $-A$ and offsetting it by $-B$ the Shapley value for the additional feature is obtained, analogously to the Shapley value for the distance bias b, previously calculated. The term $-(A \cdot \phi_b + B)$ is regarded as an expected value since it does not depend on other features. The sum of the Shapley values $-\sum_{f=0}^{|F|-1} A \cdot \phi_f$ represents the difference between the actual value and the expected value, conforming to the Shapley value formalism[19,20].

Feature contributions to the radial basis function kernel

For a direct comparison, SVERAD and SHAP values were calculated for 50 randomly selected adenosine receptor A3 ligands (A3 ligands) that we also used for compound activity predictions (see Methods). Compounds were represented using topological structural features, that is, systematically calculated pathways originating from atoms with a constant bond radius (see Methods). The RBF kernel was computed for all possible compound pairs and for each pair, exact Shapley values calculated using SVERAD were compared to corresponding SHAP values from KernelSHAP. For a value $\gamma = \frac{1}{2\sigma^2} = 0.005$, a mean Pearson's r correlation coefficient after Fisher transformation of 0.36 ± 0.18 was obtained. The low correlation indicated that the SHAP approximation was limited in its ability to explain RBF-based similarities and that calculation of exact Shapley values was preferred.

Rationalizing compound activity predictions

To apply the SVERAD approach to pharmaceutically relevant predictions and compare model explanations for different Shapley value/SHAP calculation variants, we derived SVM and RF classification models based to distinguish A3 ligands from other randomly selected compounds (see Methods). The SVM and RF classifiers achieved comparably high prediction accuracy of 93% and 92%, respectively. We then analyzed these predictions in detail.

Feature contributions to classification models

For SVM predictions, Shapley values and SHAP values were calculated with SVERAD and KernelSHAP and for RF predictions with TreeExplainer and KernelSHAP, respectively. In Table 4, median Pearson's *r* correlation coefficients are reported for feature contributions and all combinations of classification models and corresponding Shapley value/SHAP calculation methods. In addition, Fig. 1 shows the corresponding distributions of correlation coefficients.

There was only very low correlation between SVERAD Shapley values and SHAP values (0.120), which reflected the apparent limited ability of SHAP calculations to approximate Shapley values for SVM. Notably, the correlation for the SVM/RBF combination was much lower than previously determined for the SVM/Tanimoto kernel combination (0.682)[30], which reinforced the need for calculating exact Shapley values if the widely applied RBF kernel is used. By contrast, for RF, there was nearly perfect correlation between KernelSHAP and TreeExplainer (0.994), which uses exact Shapley/SHAP values for deriving local explanations. When comparing exact Shapley/SHAP values from SVERAD and TreeExplainer for corresponding predictions, essentially no correlation was observed (-0.040), indicating that different features were determining SVM and RF predictions in the presence of comparably high prediction accuracy. However, in this case, potential correlation was also principally limited because the calculations were based on different metrics (log odds scores for SVM vs. class probabilities for RF). Furthermore, SHAP values for SVM and RF displayed relatively high correlation (0.758). Taken together, the results indicated that SVERAD values were more accurate for SVM using the RBF kernel than the SHAP approximation, whereas TreeExplainer and KernelSHAP values were strongly correlated for RF.

Model explanations and feature mapping

For the SVM and RF predictions, SVERAD and TreeExplainer values were calculated, respectively, and separately analyzed for features that there were present or absent in correctly predicted test compounds. Figure 2 shows the distribution of cumulative Shapley values for these features in test compounds for log odds scores from SVERAD and probabilities from TreeExplainer.

The analysis explained model decisions and revealed different prediction characteristics for SVM and RF. For SVM, features present in active compounds made strong positive contributions to correct predictions, whereas

	SVM – SVERAD	SVM – KernelSHAP	RF – TreeSHAP	RF – KernelSHAP
SVM – SVERAD	1.000	0.120	−0.040	−0.010
SVM – KernelSHAP	0.120	1.000	0.758	0.750
RF – TreeSHAP	−0.040	0.758	1.000	0.994
RF – KernelSHAP	−0.010	0.750	0.994	1.000

Table 4. Median Pearson's *r* correlation coefficient between feature contributions from different models and explanation strategies.

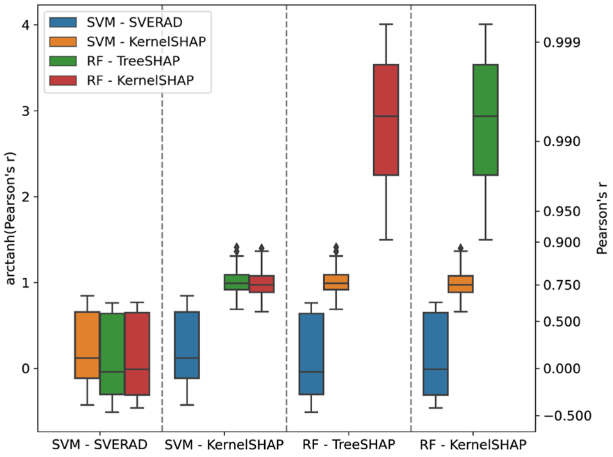

Figure 1. Distributions of Pearson's *r* correlation coefficient. Box plots represent the distributions of correlation coefficients for feature contributions from different models and explanation strategies.

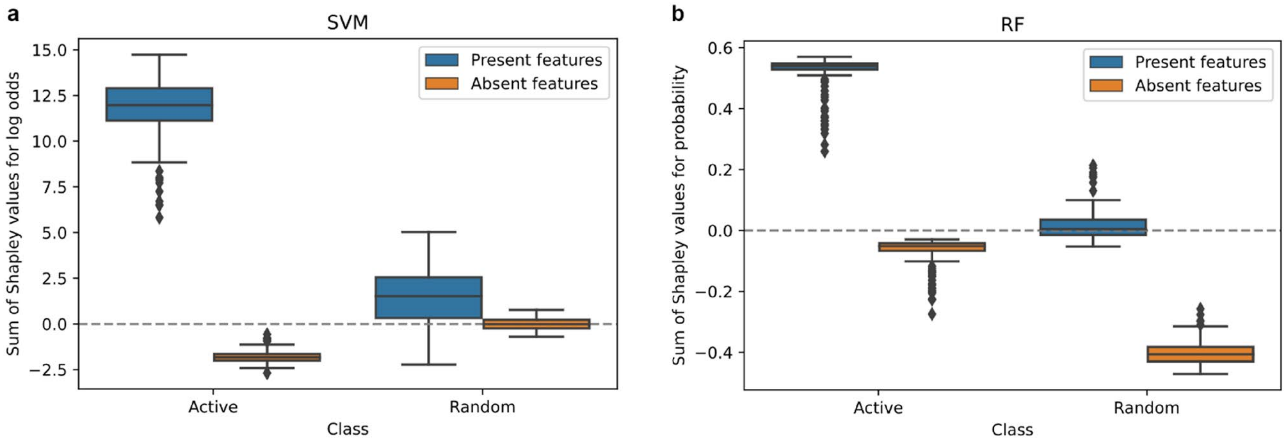

Figure 2. Distribution of feature contributions. Box plots show the distributions of cumulative Shapley values of features present or absent in correctly predicted test instances for SVERAD/SVM (**a**) and TreeExplainer/RF (**b**).

absent features made only minor contributions to incorrect predictions. For random compounds, present features made small contributions to incorrect predictions (of activity) while absent features made essentially no contributions (with cumulative Shapley values close to zero). Hence, correct predictions of inactive compounds can only be rationalized taking the expected values into account, as discussed below. For RF, features present in active compounds determined their correct predictions, while the absence of these features in random/inactive compounds was decisive for their correct predictions. By contrast, features absent in active and present in inactive compounds made only very little or no contributions.

Overall, for active compounds, the average sum of the SVERAD Shapley values for SVM of present features was 11.65, indicating strong positive contributions to predictions far beyond the expected value (– 4.61). On the contrary, absent features, with an average sum of Shapley values of – 1.79, made small negative contributions. RF displayed a similar behavior for active, but not for inactive compounds. Here, the average sum of the Shapley values for present and absent features was 0.51 and – 0.07, respectively, and the expected value was 0.49. Accordingly, for inactive compounds, SVM predictions were largely determined by the expected value, given that features present in these compounds slightly opposed correct predictions (with average positive contributions of 1.46) while the effect of absent features was negligible (– 0.008). By contrast, for RF, absent features made strong contributions (– 0.40 with respect to the expected value), while the average contribution of present features was only modest (0.018).

Features with largest contributions to predictions were visualized by mapping them on the corresponding atoms in correctly predicted test compounds, as shown in Fig. 3.

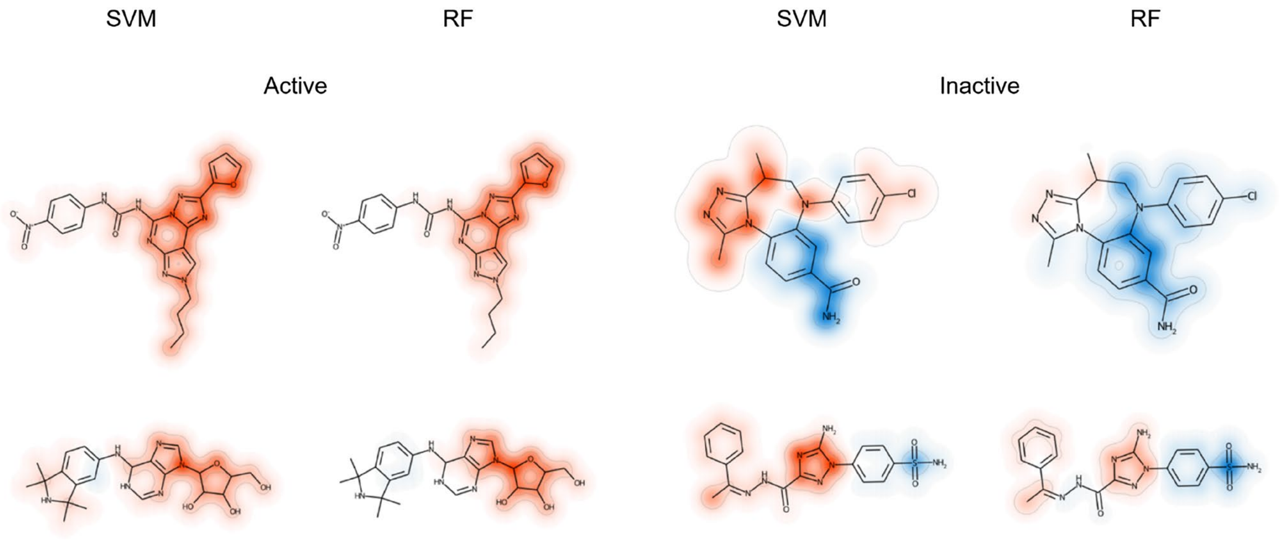

Figure 3. Feature mapping. Features present in exemplary active and random compounds correctly predicted by SVM and RF models are mapped on corresponding atoms. Red and blue coloring indicates positive and negative contributions towards prediction of activity and randomness, respectively.

For SVM and RF, features present in active compounds mostly had large positive Shapley values (red) and hence supported correct predictions (despite different value distributions, as discussed above). By contrast, for random compounds, different contributions of present features were observed. While some features supported correct predictions (blue), others opposed them (red). In active test compounds, present features supporting correct predictions with SVM and RF delineated very similar substructures.

Computational complexity

For SVERAD, the computation of the Shapley values for a given instance has at most $O\left(|F|^2\right)$ complexity, with $|F|$ being the total number of features.

We consider U the number features present in the union of the explained sample with a support vector ($U = I + D$). No computation is needed for features not present in either the intersection nor the difference ($\phi_f = 0$). However, in the worst case, all features are present in the union, so $U = |F|$ for all the support vectors.

For each support vector, we need to compute the Shapley value for one feature from the intersection and one feature from the symmetric difference. This computation requires $O(1)$ for the intersection (hence, one only needs to calculate the inverse multinomial coefficient, as shown above), and $O(D \cdot (I + 1)) = O(D \cdot I)$ for the symmetric difference. Here, $D \cdot (I + 1)$ represents the size of the Cartesian product describing unique combinations of intersecting and symmetric difference features, also considering the empty coalition. Highest complexity would result from $D = \frac{|F|}{2}$ and $I = \frac{|F|}{2}$, leading to $O\left(\frac{|F|}{2} \cdot \frac{|F|}{2}\right) = O\left(\frac{|F|^2}{4}\right) = O(|F|^2)$.

The step above must be repeated and summed up for each support vector, hence the complexity becomes $O\left(|F|^2\right) \cdot N_v$. Assuming the number of support vectors N_v to be a constant and given that the rest of the operations are products and sums with constant values, the final complexity will be $O(|F|^2)$.

Notably, for sparse input vectors such as for the calculations reported herein, the number of features in the union U was on average two orders of magnitude smaller than the total number of features $|F|$. Accordingly, in such cases, highest possible complexity is unlikely to occur. In this case, considering U as the average number of features in the union between the input sample and the support vectors, the computations require on average $O\left(U^2\right)$.

It follows that SVERAD has at most quadratic time requirements with respect to the number of features $|F|$ instead of exponential computation typically required for systematic Shapley value calculations.

Conclusions

In this work, we have introduced SVERAD, a novel methodology for the computationally efficient calculation of exact Shapley values for SVM predictions with RBF kernels. The study follows and further extends a previous investigation determining exact Shapley values for the SVM/Tanimoto kernel combination, which is preferentially used for applications focusing on the assessment of chemical similarity. The SVM/RBF kernel combination (including the Gaussian kernel) is more widely applied. In the XAI field, the Shapley value concept experiences increasing interest for rationalizing predictions of ML models. Due to the complexity of explicit Shapley value calculations, approximations are typically required, for which the SHAP approach has been a pioneering development. However, low correlation between exact Shapley values calculated with SVERAD for the RBF kernel and SHAP values clearly indicated the need to use exact Shapley values for explaining SVM predictions, in marked contrast to RF. Comparative Shapley value/SHAP analysis also revealed that highly accurate SVM and RF compound predictions were determined by different relative contributions of features present or absent in active and random test compounds. However, features present in active test compounds that consistently contributed to correct predictions with both algorithms delineated corresponding substructures. Taken together, the results reported herein indicate that SVERAD substantially aids in rationalizing SVM predictions in pharmaceutical research and other scientific fields. Therefore, SVERAD is made freely available as a part of our study.

Methods

Compounds

For compound-based Shapley value calculations and activity predictions, we used a set of 287 A3 ligands from ChEMBL[34] with curated high-confidence activity annotations, as reported previously.[30] As negative data, an equally sized set of other ChEMBL compounds was randomly selected.

Molecular representation

Compounds were represented as a keyed Morgan fingerprint with bond radius 2 (that is, a binary feature vector in which each bit position represents a unique feature)[32,35] calculated using RDKit[36]. The fingerprint comprises compound-specific numbers of layered atom environments, which represent topological structural features[35]. Each compound is described using 5487 possible binary features.

Machine learning models

Compounds activity predictions were carried out using SVM and RF models derived using the Scikit-learn library for Python[37]. The data set comprising active and random compounds was divided into training (50%) and test (50%) sets. The training set was then used for grid search hyperparameter optimization via cross-validation by randomly partitioning the compounds 10 times into training (50%) and validation (50%) subsets.

Support vector machine

Hyperparameters *gamma* (values were searched in [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]) and C (values search in [0.1, 1, 10, 50, 100, 200, 400, 500, 750, 1000]) were optimized. Parameter *gamma* corresponds to the γ value in the RBF kernel, as discussed above, and C controls the applied regularization. Smaller values of C favor generalization but increase the risk of training errors. Large values lead to a harder margin and strict misclassification penalties instead, thereby improving classification accuracy of training samples but potentially limiting the generalization ability. After grid search optimization, the best model with *gamma* = 0.01 and $C = 10$ produced an accuracy of 93% on the test set.

Random forest

The hyperparameters *n_estimators* (10, 100, 250, 500), *min_samples_split* (2, 3, 5, 7, 10), and *min_samples_leaf* (1, 2, 5, 10) were optimized. These parameters account for the number of decision trees, the minimum number of samples required to split an internal node, and minimum number of samples required to reach a leaf node, respectively. The last parameters control overfitting and model complexity. Best hyperparameter values selected via grid search were 500, 2, and 1, respectively. The final model reached an accuracy of 92% on the test set.

Shapley additive explanations

The Python SHAP[20] package was used for KernelSHAP and TreeExplainer calculations. For both SVM and RF, the KernelSHAP background sample was composed of 100 randomly selected training instances. For TreeExplainer, the entire training set was used as background sample and interventional feature perturbation was used to control input feature correlation[38].

Data availability

The code and data generated in this study are freely available on GitHub at: https://github.com/AndMastro/SVERAD.

Received: 1 June 2023; Accepted: 7 November 2023
Published online: 10 November 2023

References

1. Baum, Z. J. *et al.* Artificial intelligence in chemistry: Current trends and future directions. *J. Chem. Inf. Model.* **61**, 3197–3212 (2021).
2. Bajorath, J. State-of-the-art of artificial intelligence in medicinal chemistry. *Future Sci. OA* **7**, FSO702 (2021).
3. Varnek, A. & Baskin, I. Machine learning methods for property prediction in chemoinformatics: Quo vadis?. *J. Chem. Inf. Model.* **52**, 1413–1437 (2012).
4. Lavecchia, A. Machine-learning approaches in drug discovery: Methods and applications. *Drug Discov. Today* **20**, 318–331 (2015).
5. Chen, H., Engkvist, O., Wang, Y., Olivecrona, M. & Blaschke, T. The rise of deep learning in drug discovery. *Drug Discov. Today* **23**, 12411250 (2018).
6. Castelvecchi, D. Can we open the black box of AI?. *Nature* **538**, 20–23 (2016).
7. Gunning, D. *et al.* XAI-explainable artificial intelligence. *Sci. Rob.* **4**, eaay7120 (2019).
8. Belle, V. & Papantonis, I. Principles and practice of explainable machine learning. *Front. Big Data* **4**, 39 (2021).
9. Sundararajan, M., Ankur T. & Qiqi Y. Axiomatic attribution for deep networks. *Proc. International conference on machine learning.* 70, 3319–3328 (2017).
10. Ribeiro, M. T., Singh, S. & Guestrin, C. Why should I trust you?: Explaining the predictions of any classifier. *Proc. of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144, (2016).
11. Shrikumar, A., Greenside, P. & Kundaje, A. Learning important features through propagating activation differences. *Proc. Mach. Learn. Res.* **70**, 3145–3153 (2017).
12. Vamathevan, J. *et al.* Applications of machine learning in drug discovery and development. *Nat. Rev. Drug Discov.* **18**, 463–477 (2019).
13. Jiménez-Luna, J., Grisoni, F. & Schneider, G. Drug discovery with explainable artificial intelligence. *Nat. Mach. Intell.* **2**, 573–584 (2020).
14. Feng, J., Lansford, J. L., Katsoulakis, M. A. & Vlachos, D. G. Explainable and trustworthy artificial intelligence for correctable modeling in chemical sciences. *Sci. Adv.* **6**, eabc3204 (2020).
15. Rodríguez-Pérez, R. & Bajorath, J. Explainable machine learning for property predictions in compound optimization. *J. Med. Chem.* **64**, 17744–17752 (2021).
16. Marcou, G. *et al.* Interpretability of SAR/QSAR models of any complexity by atomic contributions. *Mol. Inf.* **31**, 639–642 (2012).
17. Polishchuk, P. Interpretation of quantitative structure–activity relationship models: Past, present, and future. *J. Chem. Inf. Model.* **57**, 2618–2639 (2017).
18. Mariia, M. & Pavel, P. Benchmarks for interpretation of QSAR models. *J. Cheminf.* **13**, 41 (2021).
19. Shapley, L. S. A value for n-person games. In *Contributions to the Theory of Games* (eds Kuhn, H. W. & Tucker, A. W.) 307–317 (Princeton University Press, 1953).
20. Lundberg, S. M., & Lee, S. I. A unified approach to interpreting model predictions. *Proc. of the 31st International Conference on Neural Information Processing Systems*, 4768–4777, (2017).
21. Strumbelj, E. & Kononenko, I. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.* **41**, 647–665 (2014).
22. Mastropietro, A., Pasculli, G., Feldmann, C., Rodríguez-Pérez, R. & Bajorath, J. EdgeSHAPer: Bond-centric Shapley value-based explanation method for graph neural networks. *iScience* **25**, 105043 (2022).
23. Mastropietro, A., Pasculli, G. & Bajorath, J. Protocol to explain graph neural network predictions using an edge-centric Shapley value-based approach. *STAR Protoc.* **3**, 101887 (2022).
24. Schoch, S., Xu, H. & Ji, Y. CS-Shapley: class-wise Shapley values for data valuation in classification. *Adv. Neural Inf. Proc. Syst.* **35**, 34574–34585 (2022).
25. Lundberg, S. M. *et al.* From local explanations to global understanding with explainable AI for trees. *Nat. Mach. Intell.* **2**, 56–67 (2020).
26. Breimann, L. Random forests. *Mach. Learn.* **45**, 5–32 (2001).
27. Cortes, C. & Vapnik, V. Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).

28. Tanimoto, T. T. *An Elementary Mathematical Theory of Classification and Prediction. IBM Report* (1958).
29. Ralaivola, L., Swamidass, S. J., Saigo, H. & Baldi, P. Graph kernels for chemical informatics. *Neural Netw.* **18**, 1093–1110 (2005).
30. Feldmann, C. & Bajorath, J. Calculation of exact Shapley values for support vector machines with Tanimoto kernel enables model interpretation. *iScience* **25**, 105023 (2022).
31. Boser, B. E., Guyon, I. M., & Vapnik, V. N. A training algorithm for optimal margin classifiers. *Proc. of the Fifth Annual Workshop on Computational Learning Theory*, 144–152 (1992).
32. Maggiora, G., Vogt, M., Stumpfe, D. & Bajorath, J. Molecular similarity in medicinal chemistry. *J. Med. Chem.* **57**, 3186–3204 (2014).
33. Böken, B. On the appropriateness of Platt scaling in classifier calibration. *Inf. Syst.* **95**, 101641 (2021).
34. Gaulton, A. *et al.* ChEMBL: A large-scale bioactivity database for drug discovery. *Nucl. Acids Res.* **40**, D1100-1107 (2012).
35. Rogers, D. & Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **50**, 742–754 (2010).
36. RDKit: Cheminformatics and machine learning software. http://www.rdkit.org/ (Accessed on 10 Mar 2023).
37. Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
38. Janzing, D., Minorics, L. & Blöbaum, P. Feature relevance quantification in explainable AI: A causal problem. *Proc. Mach. Learn. Res.* **108**, 2907–2916 (2020).

Acknowledgements

This work was partially supported (A.M.) by the EC H2020RIA project "SoBigData++" (871042) and the PNRR MUR project IR0000013-SoBigData.it.

Author contributions

All authors contributed to designing and conducting the study, analyzing the results, and preparing the manuscript.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.B.