Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal

Cite this: DOI: 10.1039/d5dd00398a

When machine learning models learn chemistry I: quantifying explainability with matched molecular pairs

Kerrin Janssen,[a] Jan M. Wollschläger,[b] Jonny Proppe*[a] and Andreas H. Göller*[c]

Explainability methods in machine learning-driven research are increasingly being used, but it remains challenging to assess their reliability without deeply investigating the specific problem at hand. In this work, we present a Python-based Workflow for Interpretability Scoring using matched molecular Pairs (WISP). This workflow can be applied to assess the performance of explainability methods on any given dataset containing SMILES and is model-agnostic, making it compatible with any machine learning model. Evaluation on two physics-based datasets demonstrates that the explanations reliably capture the predictions of the respective machine learning models. Furthermore, our workflow reveals that explainability methods can only meaningfully reflect the property of interest when the underlying models achieve high predictive accuracy. Therefore, the explainability performance on a test set can function as a quality measure of the underlying model. To ensure compatibility with any model type, we developed an atom attributor, which generates atom-level attributions for any model using any descriptor that can be obtained using SMILES representations. This method can also be applied as a standalone explainability tool, independently of WISP. WISP enables users to interpret a wide range of machine learning models in the chemical domain and gain valuable insights into how these models operate and the extent to which they capture underlying chemical principles.

Received 5th September 2025
Accepted 5th December 2025

DOI: 10.1039/d5dd00398a

rsc.li/digitaldiscovery

1 Introduction

Machine learning is a powerful tool to support decision-making across many areas of the life sciences.[1–3] It can save time and resources for laboratory scientists, for example in the development of new bioactive compounds through quantitative structure–activity relationship (QSAR) models, the optimization of drug properties such as absorption, distribution, metabolism, excretion, and toxicity (ADMET), or in computer-aided synthesis planning (CASP).[4–10] By incorporating model explanations[11–13] into the workflow, the 'black-box' nature of these models can be reduced, trust in the predictions can be increased, and molecular design can be enhanced when combined with chemical intuition.[14,15]

Visualization techniques, such as heatmaps of model explanations, can help clarify model behavior and inspire new research directions.[4–6,9,16–19] In this context, the model explanations refer to attributions assigned to each atom by the respective explainability method, indicating the contribution of each atom to the predicted property of interest. Such visual tools can be valuable for both machine learning experts and non-experts.[6] Machine learning experts can use the heatmaps as a sanity check to visually verify what their models have learned.[20] For non-experts, these heatmaps provide an accessible tool to guide molecular design decisions without requiring detailed knowledge of the underlying machine learning model.

Various approaches for interpretability have already been described in the literature.[21–23] For example, the XSMILES approach by Heberle *et al.* assigns attributions to each character in the input SMILES and presents them in an interactive format.[15] Besides understanding the behavior of ML models, Humer *et al.* highlighted the challenge of comparing different explainable AI (XAI) methods as an important open research question.[6] Being able to compare different XAI methods and their performance gives users the opportunity to choose the most suitable explainability approach for the problem at hand. Humer *et al.* addressed these tasks through an interactive two-dimensional visualization of molecules with their respective heatmaps, as well as a table view summarizing model performances.[6] Building on this work, we introduce a workflow for interpretability scoring using matched molecular pairs (WISP) and a descriptor- and model-agnostic chemical explainability method — the atom attributor. Humer *et al.* also highlighted the lack of a connection between performance metrics and explainability in interactive tools, which is precisely one of the key aspects WISP is designed to address.[6]

[a]TU Braunschweig Institute of Physical and Theoretical Chemistry, Gauss Str 17, 38106 Braunschweig, Germany. E-mail: j.proppe@tu-braunschweig.de
[b]Bayer AG Pharmaceuticals, R&D, Machine Learning Research, 13353 Berlin, Germany
[c]Bayer AG Pharmaceuticals, R&D, Computational Molecular Design, 42096 Wuppertal, Germany. E-mail: andreas.goeller@bayer.com

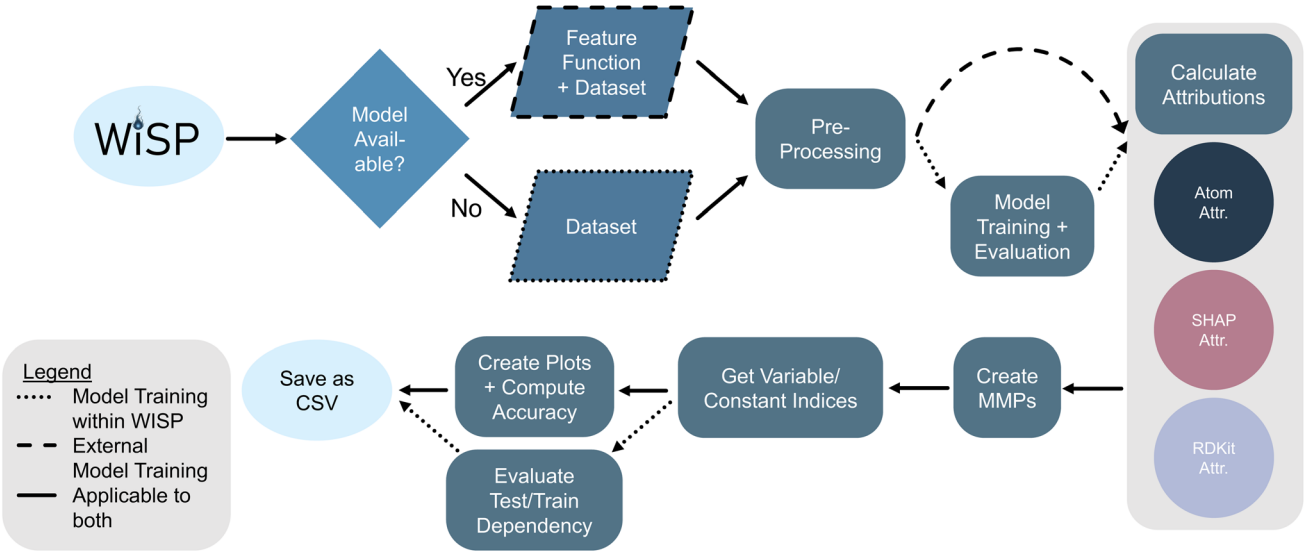

Fig. 1 Workflow diagram of WISP. The feature function refers to the process that transforms SMILES strings into the input features used by the machine learning model. Functionalities such as the ability to input a pre-trained model, as well as the inclusion of the SHAP and RDKit attributors, are described in the second part of this series. As a result, the user obtains the corresponding performance plots and accuracy values. If the model training was carried out within WISP, these plots are provided separately for the training and test sets.

To assess the performance of different explainability methods, we made use of matched molecular pairs (MMPs). MMPs are pairs of chemically similar molecules that differ by only a small, well-defined structural change, such as the substitution of a functional group.[24] The portion of the molecule that changes is referred to as the variable part, while the unchanged portion is called the constant part. Because only a single structural modification separates the pair, changes in molecular properties can often be directly linked to this specific difference.[25] This connection between structural differences and the resulting property change in the MMP can be used to quantify explainability methods, since an effective explainability method should be able to link the predicted property change to the relevant chemical motif.[26] This concept is similar to the approach used by Wellawatte *et al.*, who employed counterfactuals to explain the influence of functional group changes on model outcomes.[22] Counterfactuals describe the minimal modification required to change an outcome, a concept rooted in both philosophical reasoning and mathematical analysis.[13,14,27–29] Likewise, Vangala *et al.* used MMPs to evaluate their explainability method pBRICS, which determines fragment importances.[30] With WISP, we are now able to quantitatively assess the performance of explainability methods, providing broader and more robust insights than relying solely on the analysis of specific MMPs within a dataset. By providing a quantitative evaluation, these performance measures indicate how well the explainability methods account for both the predicted outcomes and the property of interest, *i.e.*, the extent to which the model has learned the underlying chemistry of the experimental data.

WISP enables us to assess whether a machine learning model genuinely captures underlying chemical relationships, or whether it merely learns numerical patterns without reflecting meaningful chemistry. We aim to quantify the chemical understanding of the method's performance and use it to assess the explainability performance for any given dataset (Fig. 1). WISP allows users to either evaluate the explainability of an existing model on a dataset or train a new model within its workflow, making it accessible to both experts and non-experts. After preprocessing and model evaluation, WISP computes attributions using model- and descriptor-agnostic methods, allowing users to apply different explainers (*e.g.*, the atom attributor, RDKit or SHAP). This aligns with the findings of Li *et al.*, who recommend employing multiple explainability methods for comparative evaluation.[31] Matched molecular pairs (MMPs) are generated to quantitatively assess attribution accuracy through parity plots and metrics, providing a measure of how well explanations generalize to new data.

This workflow (Fig. 1) enables users to gain valuable insights into chemical data and model behavior, and to select models that align more closely with chemical intuition. Reflecting on negative results can further help identify patterns or factors that may contribute to inaccurate explanations—and, by extension, to unreliable predictions. We also aim to evaluate how accurate models need to be in order to reliably reflect underlying chemical relationships, providing valuable guidance for future model development and application.

In this work, we describe the design and functionality of WISP, detailing how each component of the code contributes to quantifying different explainability methods (Sec. 2). We then evaluate the model performances (Sec. 3.1) and apply WISP to the Crippen log P (Sec. 3.2), experimental log P (Sec. 3.3), and solubility datasets (Sec. 3.4) to demonstrate the workflow's outcomes and illustrate how these results can be interpreted and used. Whether you are facing challenges for structural changes in molecular design, want to evaluate the quality of

your machine learning model, or seek systematic ways to improve it, WISP provides the necessary insights and tools to support these tasks.

2 Methods

2.1 WISP workflow

To quantify the performance of the explainability methods, we examined the attributions in the context of matched molecular pairs (MMPs). To achieve this, we summed the attributions of each atom (attr_i) of the variable part (N_1 and N_2) for each molecule and calculated the difference between these sums for each matched molecular pair. Details on how the attributions were calculated can be found in Section 2.4.

$$\Delta\text{Attributions MMP} = \sum_{i=1}^{N_1} \mathrm{attr}_{1,i} - \sum_{i=1}^{N_2} \mathrm{attr}_{2,i} \tag{1}$$

This difference is then compared with the difference in the model predictions ($\mathrm{pred}_{1;2}$) for the pair,

$$\Delta\text{Predictions MMP} = \mathrm{pred}_1 - \mathrm{pred}_2 \tag{2}$$

Next, the squared Pearson correlation coefficient (eqn (3)) and the accuracy (eqn (4)) can be calculated for both differences.

$$r^2 = \frac{\left(\sum_{j=1}^{K} \left(\mathbf{x}_j - \bar{\mathbf{x}}\right)\left(y_j - \bar{y}\right)\right)^2}{\sum_{j=1}^{K} \left(\mathbf{x}_j - \bar{\mathbf{x}}\right)^2 \sum_{j=1}^{K} \left(y_j - \bar{y}\right)^2} \tag{3}$$

$$\text{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{4}$$

For the squared Pearson correlation coefficient, the sums iterate over the dataset with K datapoints. The index j refers to a single datapoint, while $\bar{y}$ and $\bar{\mathbf{x}}$ denote the respective means. The accuracy indicates the percentage of molecules for which the attributions are correctly assigned in terms of sign. In this context, true positives (TP) and true negatives (TN) represent molecules where the sign of the attribution matches the sign of the prediction. Conversely, false positives (FP) and false negatives (FN) correspond to molecules where this sign agreement is not present. This allows the quantification of how well the explainability methods capture the changes in the model's predictions.

Additionally, we created a histogram of the ΔAttributions MMP for the constant part of each pair. Here, each data point represents the change in the attributions within the constant part of a single MMP.

For (mostly) group-additive tasks like Crippen log P, a robust explainability method should yield a Δ of zero for the constant part. By analyzing this histogram, one can assess how much variability the constant part introduces and to what extent this affects the reliability of the explainability method. In cases where no intermolecular interactions between the molecules of the dataset play a role, this metric should correlate with the variability in the constant part and thus with the quality of the machine learning model.

In WISP, users can choose whether they want to evaluate the explainability performance of an existing model on a given dataset (Fig. 1, top row) or whether they wish to provide only a dataset and have WISP train a model within the workflow (Fig. 1, bottom row). This flexibility also makes WISP accessible to users who may not be familiar with machine learning. After preprocessing, the attributions for the input dataset are computed (Sec. 2.4) and, if no machine learning model is supplied, one will be trained and evaluated (Fig. 1, right). Because the atom attributor is model- and descriptor-agnostic, it is always applied within the WISP framework. If the trained model uses RDKit fingerprints or Morgan fingerprints, the RDKit attributions are also generated. SHAP attributions are calculated only if the input features are Morgan fingerprints and the model is compatible with the SHAP explainer. This aspect is not discussed in detail here but can be found in part II of this paper series. Subsequently, MMPs are generated from the input data, and the atom indices of variable and constant parts of the molecules are determined to enable quantitative evaluation of the attributors, including parity plots and accuracy scores (eqn (4)).

If the model is trained within the workflow, WISP also computes the explanations on the training and test sets separately. This provides insights into how well the explanations generalize to unseen data.

2.2 Datasets and preprocessing

We evaluated WISP and the atom attributor by applying them to two different datasets covering three endpoints (*i.e.*, properties of interest). These endpoints represent a broad range of prediction tasks and varying levels of difficulty. Additional endpoints such as yield prediction, pK_i values for the inhibition of coagulation Factor Xa, and AMES mutagenicity can be found in part II of this series.

We tested the validity of the workflow using the Crippen log P as a property of interest.[32] Crippen log P is defined as a purely additive property, which is why we considered it the simplest evaluation task. Since the Crippen log P model assigns contributions to hydrogen atoms within molecules, these must be taken into account in the present investigation.[33] However, the inclusion and evaluation of hydrogen atoms is not part of the standard WISP workflow in order to reduce the computing times and was therefore only used to evaluate the calculated Crippen log P values. The next prediction task is the experimental log P, which is inherently noisier due to systematic and random measurement effects and thus more challenging to learn and explain than the Crippen log P.[32] The solubility dataset (solubility in water) should also be well-suited for explanation by the interpretability methods, since the underlying interactions are comparatively simple.[32] This stands in contrast to more challenging tasks such as binding to a biological receptor, where complex protein–ligand interactions must be considered rather than only solvent–solute interactions.

The preprocessing of the datasets was performed using a module based on RDKit's (version 2024.09.6) `rdMolStandardize`.[34] The settings were configured to process molecules with up to 1000 atoms, consider a maximum of 10 tautomers during tautomer canonicalization, retain only the largest fragment when one SMILES contained multiple fragments, and apply normalization and sanitization. This step ensures that molecules represented differently are treated equally throughout the workflow. Duplicate SMILES with different property-of-interest entries due to one of the previous steps were removed, and in cases of duplicates with identical property-of-interest entries, only one entry was retained.

2.3 Model training

To enable robust evaluation of the explainability methods, WISP includes a model training routine when no pre-trained model is provided by the user. By default, an 80/20 train/test split is applied to the data. For model training, we integrated both `scikit-learn` algorithms and the `chemprop` framework to ensure that the best-performing model can be selected for each prediction task based on the training MAE. `chemprop` offers access to deep learning models, which are widely recognized as state-of-the-art for molecular property prediction.[35,36] For instance, the portfolio of `chemprop` includes an implementation of directed message-passing neural networks (D-MPNNs), making cutting-edge deep learning approaches accessible to users of any level of expertise.[36] D-MPNNs have been shown to outperform baseline models like random forests trained on Morgan fingerprints in 9 out of 15 benchmark datasets.[36,37] However, since they do not consistently outperform simpler models in every case and bear the risk of overfitting, we ensured that WISP supports a diverse range of model types to cover various use cases and data characteristics.

2.3.1 Scikit-learn models. The default features for this step include RDKit fingerprints with a maximum path length of 7 and 2048 bits, Morgan fingerprints with a radius of 2 and 2048 bits, and MACCS fingerprints. These fingerprints were generated using RDKit version 2024.09.6.[34] The subsequent hyperparameter optimization is performed on the training set *via* a grid search over multiple model types, including Linear Least Squares Regression (`LinearRegression()`); LASSO Regression (`Lasso()`); Bayesian Ridge Regression (`BayesianRidge()`); Random Forest Regression (`RandomForestRegressor()`); Gradient Boosting Regression (`GradientBoostingRegressor()`); Support Vector Regression (`SVR()`); Gaussian Process Regression (`GaussianProcessRegressor()`) with the `Matern()` kernel and Multi-layer Perceptron Regression (`MLPRegressor()`), all implemented in `scikit-learn` (version 1.6.1).[38] For each model and feature combination, a `HalvingRandomSearchCV()` hyperparameter search was conducted using five-fold cross-validation on the training data. The parameter grid comprised a total of 82 hyperparameter combinations, while random seeds were kept constant. The results of the grid search can be found in Table SI-2. The mean absolute error (MAE, eqn (5)) was calculated for each fold, and the average MAE across all folds was used to compare model–feature combinations. The combination with the lowest average MAE was selected as the best model. The optimized model was then retrained on the entire training set and subsequently evaluated on the test set. Evaluation metrics included the squared Pearson correlation coefficient (r^2, eqn (3)), the mean absolute error (MAE, eqn (5)), the root mean squared error (RMSE, eqn (6)), and the maximum absolute error (AE_{max}, eqn (7)). Here, $\mathbf{x}$ refers to the target property, and y refers to its predicted value. The summation is carried out over all K datapoints, with j indexing each datapoint individually. The term $\bar{y}$ stands for the average of the predictions.

$$\mathrm{MAE} = K^{-1} \sum_{i=1}^{K} \left| y_j - \mathbf{x}_j \right| \tag{5}$$

$$\mathrm{RMSE} = \sqrt{K^{-1} \sum_{j=1}^{K} \left(y_j - \mathbf{x}_j \right)^2} \tag{6}$$

$$\mathrm{AE}_{\max} = \max\{|y_j - \mathbf{x}_j|\}_{j=1}{}^{K} \tag{7}$$

2.3.2 Chemprop. We integrated the functionalities of the `chemprop` package (version 2.2.0) into the WISP workflow to enable the training and evaluation of deep learning models.[36] For model training, we implemented a workflow where the predefined training set is split internally, using an 80/20 split to create a validation set during fitting. We employed the `SimpleMoleculeMolGraphFeaturizer` with `BondMessagePassing` and `MeanAggregation`. For the feed-forward component, we used the `RegressionFFN` module. After training for 50 epochs (default), the MAE on the entire training set was determined and compared to the `scikit-learn` models to select the best-performing model type.

2.4 Explainability methods

To enable comparison of all attribution methods across different datasets, we normalize the attributions by dividing each atom's attribution by the standard deviation of all attributions within the respective dataset. WISP supports the inclusion of any attribution method applicable to molecules. By default, it integrates the atom attributor, the RDKit attributor, and a SHAP-based attributor. Since the atom attributor is the only model— and descriptor—agnostic method among these, it is the primary focus in this work. The RDKit and SHAP attributors are described in detail in part II of this series.

2.4.1 Atom attributor. This attribution method is based on the atom attribution approach by Zhao *et al.*[39] To assign attributions to each atom (eqn (8)), it is systematically replaced with other elements such as hydrogen, boron, carbon, nitrogen, oxygen, fluorine, silicon, phosphorus, sulfur, chlorine, bromine, or iodine. This generates multiple mutated SMILES per atom, each of which undergoes a validity check. The atom attributor considers up to 12 mutations per atom. With a mean molecular size of 27 atoms in the MoleculeNet log P dataset, this amounts to approximately 324 mutant predictions in order to

attribute each molecule. The number of valid mutations per atom is denoted by G. Valid mutated SMILES are then featurized and passed to the model to predict the property of interest ($\text{pred}_{\text{mutated},h}$). The attribution for each atom is calculated as the average difference between the model's prediction on all mutated SMILES and the original SMILES prediction,

$$\text{Atom attribution} = \frac{\sum_{h=1}^{G} \text{pred}_{\text{original}} - \text{pred}_{\text{mutated},h}}{G} \quad (8)$$

Building on Zhao *et al.*'s work, we adapted our atom attributor to be descriptor-independent, enabling its use beyond models trained on CDDD embeddings as in the original implementation.[39] Additionally, we introduced a validity check for mutated SMILES, which was not present in the original code, and focused on attributing atoms rather than every character of a SMILES string.

2.5 MMP generation

The matched molecular pairs (MMPs) were generated using the `mmpdb` tool version 3.1.1.[40] The process involved fragmenting and subsequently indexing the molecules to create a database of MMPs. The fragmentation followed these rules: a maximum of 100 heavy atoms per molecule (`max_heavies`), up to 10 rotatable bonds (`max_rotatable_bonds`), and exactly one cut in the variable fragment part (`num_cuts`). Chirality was preserved during fragmentation (`method`), the RDKit standard salt remover was applied (`salt_remover`), and the maximum number of "up" enumerations was set to 1000 (`max_up_enumerations`), which controls stereochemistry enumeration. For indexing, default settings were used with some parameters explicitly set: a maximum of 10 heavy atoms in the variable fragment (`max_variable_heavies`), an environment radius between 0 and 5 (`min_radius` and `max_radius`), a maximum ratio of 0.2 for the variable part heavy atoms (non-hydrogen atoms) relative to the whole molecule heavy atoms (`max_variable_ratio`), and all transformations were retained (`smallest_transformation_only` set to `False`). In this work, we primarily used the default settings of the `mmpdb` tool, except for setting the number of cuts to 1 and limiting the maximum ratio of the variable part to 0.2.[40] These modifications were introduced to align with the definition of a matched molecular pair (small, well-defined structural change). To illustrate the impact of these settings, we performed an additional WISP run with the maximum variable ratio constraint disabled. The corresponding results are provided in the SI (Table SI-1). After creating the MMP database, the property of interest was loaded into the database using the `loadprops` function. Finally, duplicate MMP entries were removed, retaining only the pair with the largest number of atoms in the constant part. This results in 920 to 2544 MMPs for the databases considered in this study (Table 1).

2.6 Creation of heatmaps

Heatmaps were created to visualize the atom attributions directly on the molecular structures. This was done using the `GetSimilarityMapFromWeights` function in RDKit. To ensure comparability between different heatmaps in one dataset, they were scaled so that the maximum color intensity reflects the 70th percentile of the absolute atom attributions in the entire dataset. This approach, inspired by Harren *et al.*, ensures that the atom coloring maintains a sufficiently high visual intensity for meaningful interpretation.[5]

3 Results and discussion

3.1 Model performances

The chemprop model trained on the Crippen log *P* is the best-performing model in this study (Table 2) and is therefore most suited to quantify the error introduced by the machine learning model compared to the exact, rule-based reference. To further investigate the impact of model performance on explanation quality, we trained two models on the experimental log *P* dataset. First, by disabling the GNN functionality, we derived the best possible model architecture available within the `scikit-learn` library (Section 2.3.1), which in this case was a linear, an SVR and a GBR model (Table 2). In parallel, we trained the best-performing model for this task, *i.e.*, a `chemprop` graph neural network. The performance difference between these two models is substantial: The SVR model achieves an r^2 of 0.49, while the `chemprop` model reaches an r^2 of 0.74 (Table 2). A *t* test evaluating the significance of the model performances is provided in Table SI-3. Overall, across all regression tasks in this work, where the training was done by WISP, models based on the `chemprop` architecture consistently outperform the models trained with `scikit-learn`.

3.2 Proof of concept: Crippen log *P*

In this work, we specifically compare the exact, rule-based Crippen log *P*—which is perfectly explainable—to a machine-learned log *P* prediction. This comparison allows us to estimate the error introduced by the machine learning model in the explanations.

3.2.1 Calculated Crippen log *P*. As demonstrated by Rasmussen *et al.*, the Crippen log *P* serves as an effective

Table 1 Datasets and resulting MMPs used for the evaluation

Dataset	Type	#	# After prep	# MMPs	Source
MoleculeNet crippen	Calculated crippen log *P*	4200	4102	2544	32
MoleculeNet log *P*	Exp. log *P* at pH 7.4	4200	4028	2400	32
MoleculeNet ESOL	Water solubility (log mol L^{-1})	1128	1109	920	32

Table 2 Performance on the test set of the models used for the evaluation

Property of interest	Model type	r^2	R^2	MAE	RMSE	AE_{max}	Model source
Crippen log *P*	MolGraph; chemprop	0.93	0.93	0.24	0.38	2.84	WISP
Exp log *P*	MolGraph; chemprop	0.74	0.74	0.47	0.63	3.12	WISP
Solubility	MolGraph; chemprop	0.89	0.89	0.52	0.72	3.78	WISP
Crippen log *P*	Morgan fingerprint; Bayesian Ridge	0.72	0.72	0.52	0.73	5.06	WISP (no GNN)
Exp log *P*	MACCS fingerprint; SVR	0.49	0.49	0.66	0.88	4.19	WISP (no GNN)
Solubility	MACCS fingerprint; Gradient Boosting	0.77	0.77	0.73	1.06	5.39	WISP (no GNN)

benchmark for heatmap-based interpretability approaches.[19] The Crippen log *P* is an estimated log *P* value calculated by summing fixed contributions from different atom types, yielding the calculated value P_{calc} (eqn (9)).[33]

$$P_{calc} = \sum_i n_i a_i \quad (9)$$

Here, the number of atoms of one specific type i is denoted by n, while a represents the contribution of the atom type.[33] This makes the Crippen log *P* an excellent proof of concept for the WISP workflow, as the Δ values from eqn (1) ideally correspond exactly to the Δ in the calculated Crippen log *P*.

As expected, the Δ in contributions from the entire molecule is in perfect agreement ($r^2 = 1.00$) with the Δ in the Crippen log *P* (Fig. 2a). However, when considering only the variable part of the molecule, the squared Pearson correlation coefficient between its contribution Δ and the Crippen log *P* Δ decreases to 0.93 (Fig. 2b). This reduction is attributable to the fact that Crippen atom contributions are dependent on the local chemical environment. For example, a carbonyl group adjacent to an aromatic system contributes +0.11, whereas the same group in an aliphatic environment contributes −0.15 to the Crippen log *P*.[33] This neighborhood dependency also explains the outliers observed in the difference of the constant part (Fig. 2d), where ideally the contribution difference should be zero, as the constant part should remain unchanged between matched molecular pairs (MMPs). To further investigate this issue, we included in the analysis a neighboring atom of the variable part that originally belonged to the constant part (Fig. 2c), resulting in a significantly improved correlation. This finding confirms that the reduced correlation in the variable part arises from the dependency of atom contributions on their immediate chemical surroundings. Still remaining deviations from the ideal correlation can be resolved by including a second neighboring atom, further supporting this conclusion.

3.2.2 Machine learning on Crippen log *P*. To quantify the influence of the machine learning model and the attribution method on explainability performance, a machine learning model was trained to predict the Crippen log *P* (Table 2). The resulting model explainability demonstrates a significantly higher squared Pearson correlation coefficient ($r^2 = 0.77$) (Fig. 4b), than the second-best SVR model ($r^2 = 0.45$) (Fig. 7b) on the test set. Since the model operates on molecular graphs, only

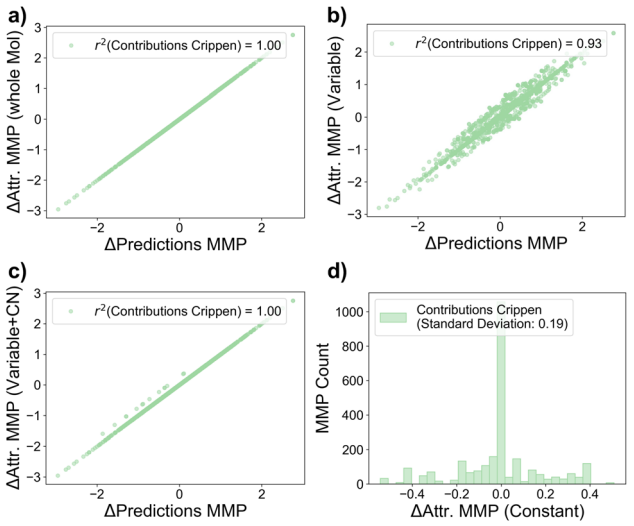

Fig. 2 Shown are the contributions to the Crippen log *P* of the entire molecule (a), the variable part (b), and the variable part including one closest neighbor (CN) atom (c). Additionally, a histogram of the variance in contributions from the constant part of the molecules is presented (d). In all cases, the predictions refer specifically to the Crippen log *P*.

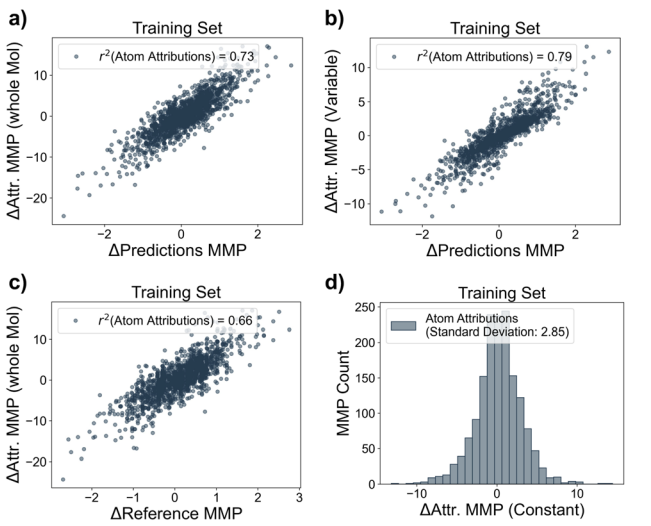

Fig. 3 Shown are the attributions for the entire molecule (a) and for the variable part (b) with respect to the predicted Crippen log *P* training set. Additionally, the correlation between the attributed contributions and the true Crippen log *P* is presented (c), along with a histogram of the variance in the contributions of the constant part of the molecules (d). All values are derived from the model's training set.

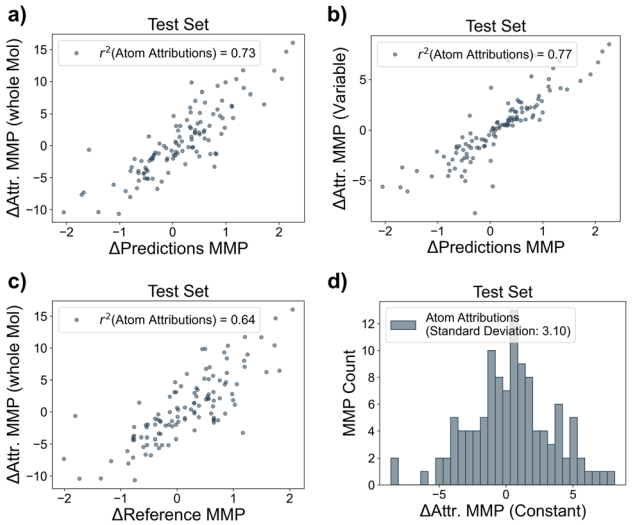

Fig. 4 Shown are the attributions for the entire molecule (a) and for the variable part (b) with respect to the predicted Crippen log *P* test set. Additionally, the correlation between the attributed contributions and the true Crippen log *P* is presented (c), along with a histogram of the variance in the contributions of the constant part of the molecules (d). All values are derived from the model's test set.

the atom attributor from the attributors used in this work is applicable for generating explanations in this case.

The explainability performance on the test set closely mirrors that of the training set (Fig. 3). This consistency is expected, given the model's high predictive performance (Table 2), which suggests it is equally capable of providing meaningful explanations for both training and test data. This finding underscores that accurate predictions are a prerequisite for generating meaningful explanations on unseen data.

When comparing the explainability performance of the machine learning-based approach (Fig. 3) with the direct Crippen log *P* calculation (Fig. 2), the former scores significantly worse in terms of the r^2 value across all cases. This demonstrates how essential the influence of the `chemprop` machine learning model as well as the attribution method is on the final explanations, and suggests that an r^2 of 0.79 on the MMPs may represent the practical upper limit for this setup (Fig. 3b).

Examining individual example structures from the training and test sets (Fig. 5, left) shows that the machine-learned Crippen model correctly captures the overall trend and correctly attributes the amino group and fluorine, which is the variable part of the matched molecular pair. However, when comparing the ground-truth Crippen heatmap (Fig. 5, top left) to the heatmap produced by the machine-learned model (Fig. 5, bottom left), it becomes clear that the latter attributes the aromatic atoms with higher values than the ground truth but still with near constant attributions of the constant part.

To qualitatively assess how well the explanations reflect the model predictions, we calculated the accuracy of the ΔAttributions MMP (eqn (1)) in relation to the differences in predicted values, *i.e.*, if the directions of the machine predictions and the attributions for the pairs are consistent (Table 3). Interestingly, the accuracy appears to improve when only the

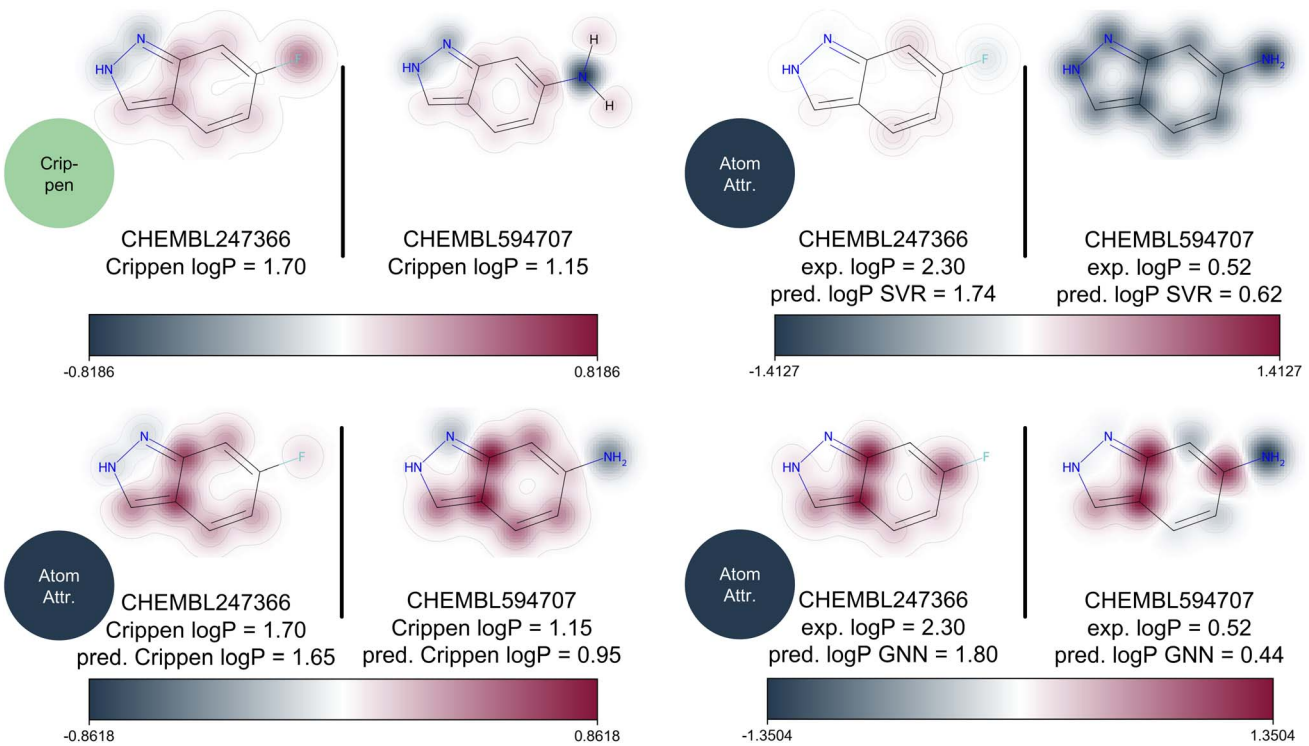

Fig. 5 Comparison of heatmaps for the matched molecular pair consisting of CHEMBL247366 (test set) and CHEMBL594707 (training set), generated using WISP.

Table 3 Overview of the performance of the explainability methods. The best results in each column are shown in bold, and the least favorable in italics. The top part refers to the best performing models while the second part refers to the second best performing models of the respective dataset. POI stands for property of interest, *e.g.*, target property

	Training set					Test set				
Property of interest (POI)	r^2 whole molecule to pred.	r^2 variable part to pred.	r^2 whole molecule to POI	Std constant part	Accuracy variable part to pred.	r^2 whole molecule to pred.	r^2 variable part to pred.	r^2 whole molecule to POI	Std constant part	Accuracy variable part to pred.
Crippen log *P*	0.73	0.79	0.66	**2.85**	0.90	0.73	0.77	0.64	**3.10**	0.93
Exp log *P* GNN	0.61	0.61	0.49	4.95	0.83	*0.61*	0.63	0.41	3.71	0.79
Solubility	0.85	**0.90**	0.78	4.32	**0.94**	0.91	**0.96**	**0.77**	5.59	**0.95**
Crippen log *P* linear	0.63	*0.49*	*0.30*	4.39	0.79	0.74	0.55	*0.05*	5.30	0.80
Exp log *P* SVR	**0.96**	0.50	**0.87**	*20.99*	*0.76*	**0.96**	*0.45*	0.17	*16.03*	*0.70*
Solubility GBR	*0.57*	0.86	0.39	5.19	0.87	0.65	0.96	0.60	3.44	0.93

variable part of the molecule is considered. However, this observation is not consistent with the quantitative analysis presented in Fig. 3. One possible explanation lies in the presence of small variations around zero, which can increase the likelihood of false positives or negatives. In this example, the overall heatmap coloring—i.e., the reliability of the attributions—can be trusted with an accuracy of 80% on the training set and 84% on the test set.

3.3 Experimental log *P*

3.3.1 SVR model. This model is the worst performing model of the ones created in this study (Table 2).The atom attribution method reflects the experimental values surprisingly well for the log *P* model training set (Fig. 6c), despite the model's relatively low performance (Table 2). The squared Pearson correlation coefficient for the explainability of the entire molecule is 0.96 on both the training and test sets (Fig. 6a and 7a). However, the explainability performance for the variable part of the MMP is considerably lower, with squared Pearson correlation coefficients of 0.50 for the training set and 0.47 for the test sets (Fig. 6b and 7b). The variation in the constant part is substantial, with standard deviations of 20.99 and 16.03 for the training and test set, respectively, indicating that the constant part introduces significant variability to the attributions. The largest discrepancy between the explainability of the training and test sets lies in the model's ability to explain the experimental values. For the training set, the model demonstrates reasonable explanatory power, with an r^2 of 0.87 for the whole molecule. In contrast, on the test set, the r^2 drops sharply to 0.17. These results suggest that the model fails to learn the underlying chemistry and, consequently, heatmaps generated from the test data should not be used to guide

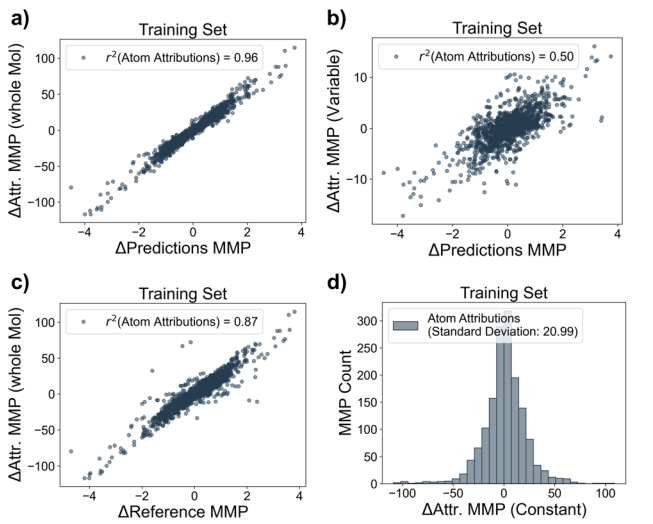

Fig. 6 Shown are the attributions for the entire molecule (a) and for the variable part (b) with respect to the predicted log *P* with the SVR model. Additionally, the correlation between the attributed contributions and the experimental log *P* is presented (c), along with a histogram of the variance in contributions from the constant part of the molecules (d). All values are derived from the model's training set.

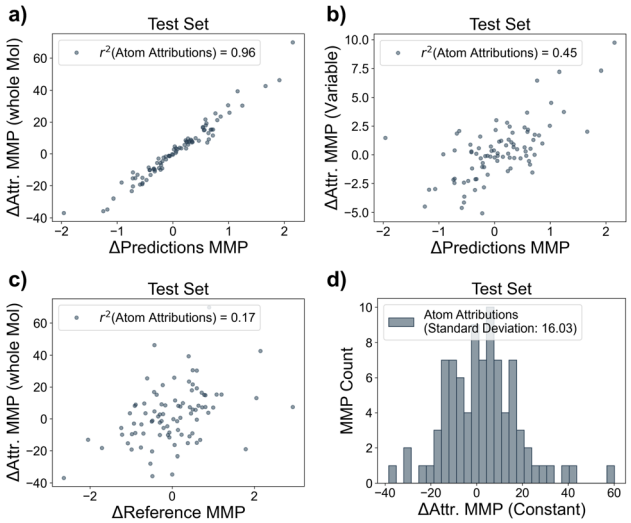

Fig. 7 Presented are the attributions for the entire molecule (a) and for the variable part (b) with respect to the predicted log *P* with the SVR model. Additionally, the correlation between the attributed contributions and the experimental log *P* is shown (c), along with a histogram of the variance in contributions from the constant part of the molecules (d). All values are derived from the model's test set.

experimental decisions. Therefore, a drop in explainability performance on the test set can be regarded as a quality measure for the underlying model, which can be used systematically to improve the model or the training data, ultimately enabling the development of models that truly capture chemical relationships.

3.3.2 GNN model. For the better-performing GNN log *P* model, explainability on the training set does not improve for either the whole molecule or the variable part (Fig. 8) compared to the SVR model. However, the variability in the constant part is significantly reduced (Fig. 8d), with the standard deviation decreasing from 20.99 in the SVR model (Fig. 6d) to 4.95 in the GNN model (Fig. 8d). Additionally, the explainability performance for both the variable part and the property of interest on the test set increases substantially. Consequently, this model (Fig. 9) is considerably better suited to explain future data compared to the SVR model.

Examining the accuracy of the variable part reveals a similar trend (Table 3). The explainability of the GNN model appears superior in capturing the trends of the variable part, whereas the SVR model demonstrates strong explainability for the whole molecule, consistent with the observations described above.

This difference between the SVR and GNN models also becomes evident when examining an example MMP (Fig. 5, right). For the SVR model (Fig. 5, top right), both molecules are almost uniformly colored, indicating that the model does not capture any meaningful structure–property relationships. In contrast, the heatmap derived from the GNN model is more detailed and shows that the nitrogen atoms in the right molecule are generally assigned a slightly negative contribution (Fig. 5, bottom right), reflecting the expected effect of heteroatoms decreasing lipophilicity. This more nuanced attribution is absent in the left molecule. A likely reason for this difference is that the left molecule is part of the test set, whereas the right molecule was included in the training set.

Different MMPs from this dataset are also discussed in the work of Humer *et al.*[6] Here, Class Attribution Maps (CAMs) were used for atom-level attribution.[6,41] In comparison, our heatmaps resemble their "base model" explanations, which are similarly more uniform than those from their more complex "XAI

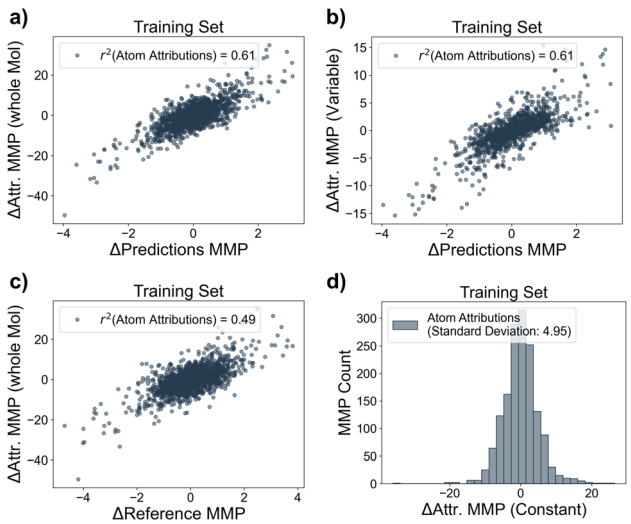

Fig. 8 Shown are the atom attributions for the entire molecule (a) and the variable part (b) with respect to the predicted log *P* of the GNN model. Additionally, the correlation between the attributed contributions and the experimental log *P* is presented (c), along with a histogram of the variance in contributions from the constant part of the molecules (d). All values are derived from the model's training set.

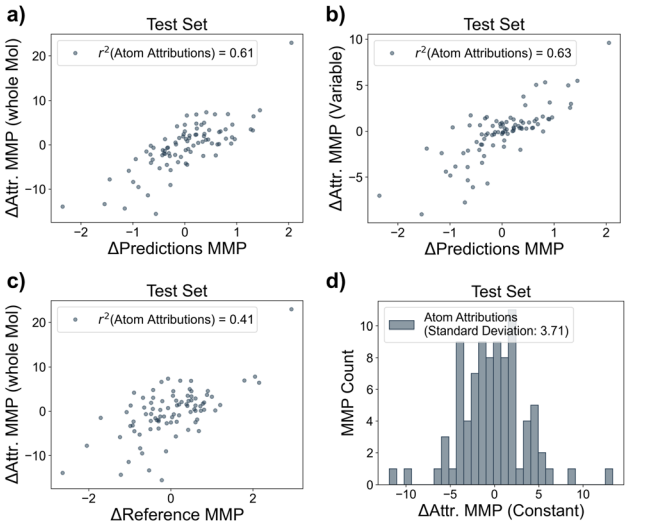

Fig. 9 Shown are the atom attributions for the entire molecule (a) and the variable part (b) with respect to the predicted log *P* of the GNN model. Additionally, the correlation between the attributed contributions and the experimental log *P* is presented (c), along with a histogram of the variance in contributions from the constant part of the molecules (d). All values are derived from the model's test set.

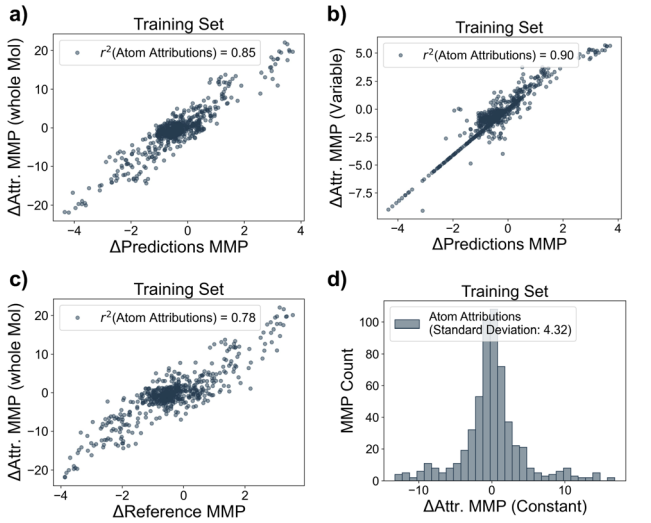

Fig. 10 Shown are the atom attributions for the entire molecule (a) and the variable part (b) with respect to the predicted solubility. Additionally, the correlation between the attributed contributions and the experimental solubility is presented (c), along with a histogram of the variance in contributions from the constant part of the molecules (d). All values are derived from the model's training set.

model".[6] Notably, their heatmaps show considerable variability in the constant parts of the molecules, whereas the results of this work fluctuate less—a marker of reliability (Fig. 5, right).

3.4 Solubility

The solubility model is the second-best performing model after the Crippen log *P* model (Table 2). This is also reflected in the performance of its explanations: The correlation for the variable part partly forms a near–perfect correlation line (Fig. 10 and 11b panels). The MMPs contributing to this near–perfect correlation mostly involve relatively small variable regions, often consisting of the exchange of a single substituent on an aromatic ring. Moreover, it appears feasible to explain not only the predictions but also the solubility itself, as indicated by the high squared Pearson correlation coefficient of up to 0.77 on the test set (Fig. 11c). This represents the best test-set performance among all experiments conducted in this study and highlights the high quality of this model. Importantly, no significant drop in explainability performance between the training and test sets was observed, further supporting this conclusion, as discussed in Section 3.3.1. The slight improvements observed in the performance on the test set may be due to its small size: while the training set contains 574 MMPs, the test set includes only 44 MMPs. In the histogram of the constant part of the molecule (Fig. 10 and 11, panel d), 'shoulders' appear around a Δ of −10 and 10. These features arise from MMPs where, for example, a methyl group or halogen is exchanged for another small group, or a hydroxy group is replaced by a hydrogen atom.

Qualitatively, the accuracy for the variable part is the highest across all experiments performed in this work. Accordingly, the user can rely on the attribution coloring of the variable part for 94% of the MMPs in the training set and 95% in the test set (Table 3).

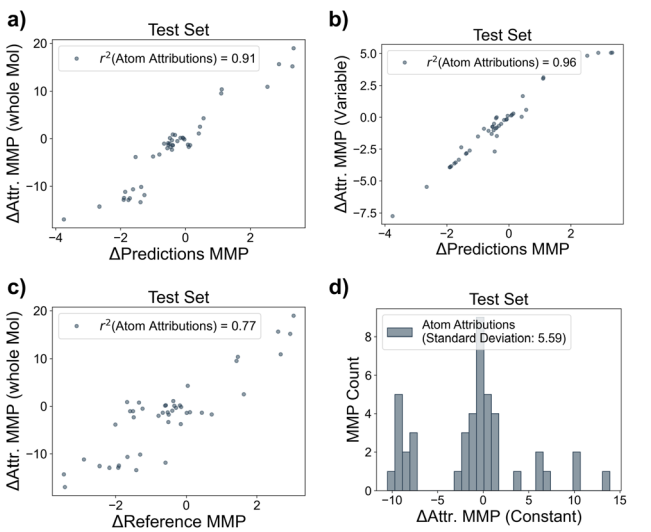

Fig. 11 Shown are the attributions for the entire molecule (a) and the variable part (b) with respect to the predicted solubility. Additionally, the correlation between the attributed contributions and the experimental solubility is presented (c), along with a histogram of the variance in contributions from the constant part of the molecules (d). All values are derived from the model's test set.

4 Conclusion

In this work, we present WISP, a framework designed to qualitatively and quantitatively assess molecular model explainability, along with a novel, model- and descriptor-agnostic atom attributor. The matched molecular pair (MMP) analysis proved invaluable in assessing whether local structural changes are correctly reflected in the heatmaps, providing an objective sanity check alongside global correlation metrics. Our results show that the performance of explainability methods depends strongly on the underlying model quality.

Our findings demonstrate that when a machine learning model achieves high predictive performance, it is usually capable of providing meaningful and reliable explanations for previously unseen data. The Crippen log *P* served as a benchmark to define the upper bound for explainability when the true atom contributions to the property of interest are known, highlighting how model imperfections inevitably introduce systematic attribution errors (Table 3). Our results for the experimental log *P* dataset highlight how strongly model performance influences explainability performance. Here, the impact of model quality on the r^2 of the variable part, the standard deviation of the constant part, and the r^2 for the explanations on the test set becomes clear. On the test set, the r^2 for the variable part is improved by 0.18 units with the better-performing model, while the standard deviation of the constant part decreases by 12.32 units. Since the explanation of unseen data is a key goal, having a well-performing model is essential to achieve this. While not an implication—see, *e.g.*, the SVR log *P* model with high variability in the constant parts of the MMPs—we find evidence for a strong potential of high

explainability on well-performing models. Consequently, there is a clear need to continue developing and validating robust, high-performing models to enable explanations that truly reflect underlying chemical relationships. A drop in explainability performance on the test set can serve as a valuable quality measure for the model's lacking ability to generalize and capture real chemical effects, guiding targeted improvements to both the model architecture and the training data.

In summary, WISP provides an accessible, systematic way to scrutinize and compare model explanations, helping to identify where models succeed, where they fail, and how they can be improved. Our atom attributor extends explainability beyond specific embeddings or descriptors, offering a flexible approach for diverse molecular modeling tasks. Together, these contributions move us closer to the goal of truly interpretable and reliable AI-driven predictions in chemistry and drug discovery.

Author contributions

JMW + AHG: conceptualization, KJ + JMW: data curation, KJ: formal analysis, JP + AHG: funding acquisition, KJ: investigation, KJ + JMW: methodology, JP + AHG: project administration, JP + AHG: resources, KJ + JMW: software, JMW + JP + AHG: supervision, KJ: validation, KJ: visualization, KJ: writing – original draft, KJ + JMW + JP + AHG: writing – review & editing.

Conflicts of interest

There are no conflicts to declare.

Data availability

All the code for the presented work, as well as the URL to the public web app, are available on GitHub at https://github.com/kerjans/ml-XAI. An archived version of the whole code is provided at zenodo at https://doi.org/10.5281/zenodo.17055142.

Supplementary information (SI) is available. See DOI: https://doi.org/10.1039/d5dd00398a.

Acknowledgements

JP acknowledges funding by Germany's joint federal and state program supporting early-career researchers (WISNA) established by the Federal Ministry of Education and Research (BMBF). The authors acknowledge Philipp Held (TU Braunschweig) for providing explanatory Jupyter notebooks on WISP. During the preparation of this work, the authors employed ChatGPT/OpenAI to improve phrasing and assist with coding. All generated content was carefully reviewed and edited by the authors, who take full responsibility for the accuracy and content of the publication.

References

1 E. N. Muratov, J. Bajorath, R. P. Sheridan, I. V. Tetko, D. Filimonov, V. Poroikov, T. I. Oprea, I. I. Baskin, A. Varnek, A. Roitberg, O. Isayev, S. Curtalolo, D. Fourches, Y. Cohen, A. Aspuru-Guzik, D. A. Winkler, D. Agrafiotis, A. Cherkasov and A. Tropsha, *Chem. Soc. Rev.*, 2020, **49**, 3525–3564.
2 L. Zhang, J. Tan, D. Han and H. Zhu, *Drug Discovery Today*, 2017, **22**, 1680–1685.
3 K. Jorner, A. Tomberg, C. Bauer, C. Sköld and P.-O. Norrby, *Nat. Rev. Chem.*, 2021, **5**, 240–255.
4 P. Polishchuk, *J. Chem. Inf. Model.*, 2017, **57**, 2618–2639.
5 T. Harren, H. Matter, G. Hessler, M. Rarey and C. Grebner, *J. Chem. Inf. Model.*, 2022, **62**, 447–462.
6 C. Humer, H. Heberle, F. Montanari, T. Wolf, F. Huber, R. Henderson, J. Heinrich and M. Streit, *J. Cheminf.*, 2022, **14**, 21.
7 A. H. Göller, L. Kuhnke, F. Montanari, A. Bonin, S. Schneckener, A. ter Laak, J. Wichard, M. Lobell and A. Hillisch, *Drug Discovery Today*, 2020, **25**, 1702–1709.
8 K. Janssen and J. Proppe, *J. Chem. Inf. Model.*, 2025, **65**, 1862–1872.
9 Y. Sushko, S. Novotarskyi, R. Körner, J. Vogt, A. Abdelaziz and I. V. Tetko, *J. Cheminf.*, 2014, **6**, 48.
10 M. Eckhoff, J. V. Diedrich, M. Mücke and J. Proppe, *J. Phys. Chem. A*, 2024, **128**, 343–354.
11 D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf and G.-Z. Yang, *Sci. Robot.*, 2019, **4**, eaay7120.
12 J. D. Lee and K. A. See, *Hum. Factors*, 2004, **46**, 50–80.
13 T. Miller, *AI*, 2019, **267**, 1–38.
14 J. Jiménez-Luna, F. Grisoni and G. Schneider, *Nat. Mach. Intell*, 2020, **2**, 573–584.
15 H. Heberle, L. Zhao, S. Schmidt, T. Wolf and J. Heinrich, *J. Cheminf.*, 2023, **15**, 2.
16 J. Jiménez-Luna, M. Skalic, N. Weskamp and G. Schneider, *J. Chem. Inf. Model.*, 2021, **61**, 1083–1094.
17 A. Chatzimparmpas, R. M. Martins, I. Jusufi and A. Kerren, *IV*, 2020, **19**, 207–233.
18 G. Marcou, D. Horvath, V. Solov'ev, A. Arrault, P. Vayer and A. Varnek, *Mol. Inform.*, 2012, **31**, 639–642.
19 M. H. Rasmussen, D. S. Christensen and J. H. Jensen, *SciPost Che*, 2023, **2**, 002.
20 M. Matveieva and P. Polishchuk, *J. Cheminf.*, 2021, **13**, 41.
21 T. Janela and J. Bajorath, *J. Chem. Inf. Model.*, 2023, **63**, 7032–7044.
22 G. P. Wellawatte, H. A. Gandhi, A. Seshadri and A. D. White, *J. Chem. Theory Comput.*, 2023, **19**, 2149–2160.
23 D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen and X. Zhang in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA, 2020, pp. 19620–19631.
24 J. Hussain and C. Rea, *J. Chem. Inf. Model.*, 2010, **50**, 339–348.
25 D. Stumpfe, H. Hu and J. Bajorath, *ACS Omega*, 2019, **4**, 14360–14368.
26 J. Jiménez-Luna, M. Skalic and N. Weskamp, *J. Chem. Inf. Model.*, 2022, **62**, 274–283.
27 J. Woodward and C. Hitchcock, *Nos*, 2003, **37**, 1–24.
28 A. Reutlinger, *Philos. Sci*, 2016, **83**, 733–745.
29 D. Kahneman and D. T. Miller, *Psychol. Rev*, 1986, **93**, 136–153.
30 S. R. Vangala, S. R. Krishnan, N. Bung, R. Srinivasan and A. Roy, *J. Chem. Inf. Model.*, 2023, **63**, 5066–5076.

31 S. Li, X. Wang and A. Barnard, *Mach. Learn.: Sci. Technol*, 2025, **6**, 013002.
32 Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing and V. Pande, *Chem. Sci.*, 2018, **9**, 513–530.
33 S. A. Wildman and G. M. Crippen, *J. Chem. Inf. Comput. Sci.*, 1999, **39**, 868–873.
34 G. Landrum, P. Tosco, B. Kelley, R. Rodriguez, D. Cosgrove, R. Vianello, P. sriniker, P. Gedeck, G. Jones, N. Schneider, E. Kawashima, D. Nealschneider, A. Dalke, M. Swain, B. Cole, S. Turk, A. Savelev, C. Tadhurst, A. Vaucher, M. Wójcikowski, I. Take, V. F. Scalfani, R. Walker, K. Ujihara, D. Probst, J. Lehtivarjo, H. Faara, G. Godin, A. Pahl and J. Monat, *rdkit/rdkit: 2024_09_5 (Q3 2024) Release*, 2025.
35 H. Chen, O. Engkvist, Y. Wang, M. Olivecrona and T. Blaschke, *Drug Discovery Today*, 2018, **23**, 1241–1250.
36 E. Heid, K. P. Greenman, Y. Chung, S.-C. Li, D. E. Graff, F. H. Vermeire, H. Wu, W. H. Green and C. J. McGill, *J. Chem. Inf. Model.*, 2024, **64**, 9–17.
37 K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea, A. Palmer, V. Settels, T. Jaakkola, K. Jensen and R. Barzilay, *J. Chem. Inf. Model.*, 2019, **59**, 5304–5305.
38 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and É. Duchesnay, *J. Mach. Learn Res.*, 2011, **12**, 2825–2830.
39 L. Zhao, F. Montanari, H. Heberle and S. Schmidt, *Artif. Intell. Life Sci.*, 2022, **2**, 100047.
40 A. Dalke, J. Hert and C. Kramer, *J. Chem. Inf. Model.*, 2018, **58**, 902–910.
41 P. E. Pope, S. Kolouri, M. Rostami, C. E. Martin and H. Hoffmann in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Long Beach, CA, USA, 2019, pp. 10764–10773.