QPhAR – Quantitative Pharmacophore Activity Relationship: Method and Validation

Stefan M. Kohlbacher

University of Vienna <https://orcid.org/0000-0001-5477-1917>

Thierry Langer

University of Vienna <https://orcid.org/0000-0002-5242-1240>

Thomas Seidel (✉ thomas.seidel@univie.ac.at)

University of Vienna <https://orcid.org/0000-0002-9815-6577>

Methodology

Keywords: pharmacophore, QSAR, regression, quantitative-pharmacophore-model

Posted Date: April 22nd, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-426014/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at Journal of Cheminformatics on August 9th, 2021. See the published version at <https://doi.org/10.1186/s13321-021-00537-9>.

Abstract

QSAR methods are widely applied in the drug discovery process, both in the hit-to-lead and lead optimization phase, as well as in the drug-approval process. Most QSAR algorithms are limited to using molecules as input and disregard pharmacophores or pharmacophoric features entirely. However, due to the high level of abstraction, pharmacophore representations provide some advantageous properties for building quantitative SAR models. The abstract depiction of molecular interactions avoids a bias towards overrepresented functional groups in small datasets. Furthermore, a well-crafted quantitative pharmacophore model can generalise to underrepresented or even missing molecular features in the training set by using pharmacophoric interaction patterns only. This paper presents a novel method to construct quantitative pharmacophore models and demonstrates its applicability and robustness on more than 250 diverse datasets. 5-fold cross-validation on these datasets with default settings yielded an average RMSE of 0.62, with an average standard deviation of 0.18. Additional cross-validation studies on datasets with 15-20 training samples showed that robust quantitative pharmacophore models could be obtained. These low requirements for dataset sizes renders quantitative pharmacophores a viable go-to method for medicinal chemists, especially in the lead-optimisation stage of drug discovery projects.

Introduction

Quantitative structure-activity relationship (QSAR) studies were first introduced by Hansch et al. (1) in 1962 and were growing in popularity ever since. Starting with simple correlations studies of chemical and biological properties, such as logP and K_i values, QSAR has evolved into a sophisticated method applying complex machinelearning (ML) models(2) on vast amounts of chemical data(3), often using more than a few thousand descriptors. QSAR models are not only useful for internal assistance in the drug discovery process, but highly validated and robust models have even been built by the FDA to assist the drugapproval process(4).

Over the years, QSAR has been influenced heavily by advanced machine learning(2) and other data processing systems, which effectively allows the researcher to extract more complex relationships from their data. With the use of more capable models, more complex input data can be processed. Countless descriptors or fingerprints(5) have been derived from 2D molecular structures but do not take into account spatial information and molecular conformation. Spatial information becomes even more critical when dealing with stereoisomers(6).

The popular QSAR modelling algorithm CoMFA(7), developed in the '80s, uses 3D conformations of molecules as input, aligns them to each other, and then creates a predictive model from the molecules' calculated steric and electrostatic interaction fields. The concept has gained wide popularity but never extended to different input domains than molecules. The method PHASE(8) proposed by Schrödinger(9) has taken this approach a step further. In addition to, or instead of calculating electrostatic interaction fields of the molecules, it is possible to generate pharmacophore fields from the input molecules. The same MLalgorithm as used in CoMFA, PLS (partial least squares), is then applied to create a predictive

quantitative model. At the time, this has been a novelty since pharmacophores have only been used for qualitative virtual screening studies. Using pharmacophore fields derived from functional groups for quantitative modelling extends the CoMFA concept, using abstract 3D information of molecules for QSAR. Nevertheless, these pharmacophore fields are derived from molecules and a pure quantitative algorithm applied on pharmacophores has never been presented before.

Using pharmacophores as input in QSAR studies has several advantages: Due to the abstract nature of pharmacophores, they are less influenced by small spatial perturbations of molecular features characteristic for such interactions. For example, bioisosteres are often highly similar in their interaction profile. They might cover, however, entirely different functional groups and substructures. Building a QSAR model on such data inevitably introduces a bias towards the predominant bioisosteric form occurring in the dataset. Pharmacophores, on the other hand, transform different functional groups with the same interaction profile into an abstract chemical feature representation associated with a particular non-bonding interaction type, such as a π -stacking interaction or HBond donor/acceptor interaction. This generalisation makes quantitative models more robust and less dependent on the dataset being used. Primarily in biological assays, robust predictive models are essential to avoid modelling the experimental noise(10).

Virtual screening takes advantage of pharmacophores' abstract nature to achieve an effect known as "scaffoldhopping"(11). Here, pharmacophores help to overcome a structural molecular bias by only considering the interaction patterns but not the molecular structures. A carefully constructed quantitative pharmacophore model will build on these advantages and the scaffoldhopping ability to harness its strengths. Besides abstracting molecular structures, pharmacophores also abstract the exact steric location and orientation of interactions by introducing tolerance ranges. Losing information on the precise position of possible interactions might not be desired with highly conserved protein targets. In general, however, generalisation is considered positive and avoids overfitted models.

Pharmacophore modelling is often used in combination with virtual screening to find novel hits. Deciding on the best pharmacophore model for virtual screening runs is often a tedious process relying on a large dataset of mostly artificially generated decoys and some truly active compounds. In addition to requiring large amounts of data, this evaluation process relies on the binaryclassification of molecules into active and inactive ones. Molecules with similar activity values close to the cutoff are classified differently, although they demonstrate a quite similar experimental behaviour.

A quantitative pharmacophore model would be able to score other pharmacophore models and assign an estimated non-binary activity to these pharmacophores. The (biological) activity of pharmacophores can be interpreted as the expected activity of molecules matching such a pharmacophore. In the context of virtual screening, it is expected that the scored pharmacophore will retrieve molecules from a database with similar activity values. Therefore, the quantitative pharmacophore model can be easily applied as a ranking method to prioritise pharmacophore models generated by a researcher.

Despite the possible advantages hardly any research was done on quantitative pharmacophores and accompanied methods. In contrast, QSAR applied on molecular structures has fostered plenty of research and a googlescholar search for the query “quantitative structure-activity relationships” yields close to 5 million results. Nevertheless, two commercially available tools have been released which are able to relate pharmacophores quantitatively to biological activity or other specified properties.

PHASE is a commercially available tool implemented in Maestro(8). Besides pharmacophore perception, it allows for quantitative rationalisation of activity data based on 3D pharmacophore fields obtained from a set of ligands. Pharmacophores are created for each aligned ligand, whereas the alignment is not done automatically and needs to be considered by the user. The aligned pharmacophores are placed into a vectorised box, each voxel containing information about the value of the pharmacophore fields in that location. The box is used as input for a PLSalgorithm to regress the pharmacophore fields against a set of activity values. As output, the user gets a model displaying favourable as well as unfavourable regions contributing to the activity values. Additionally, the activity of new ligands can be predicted by feeding them to the model after alignment.

Even though PHASE provides one of two available QSAR methods for pharmacophores, it still relies on molecules as input for alignment and model building. Due to this shortcoming, aposite derived pharmacophores or pharmacophores obtained from ligand-based modelling can only be predicted via workarounds. Therefore, pharmacophore QSAR within PHASE is similar to atom-based QSAR, except that an additional step for calculating the pharmacophore fields is carried out.

The second available method for pharmacophore QSAR is the Hypogen(12) algorithm implemented in the Catalyst program, which now is part of BioVia's(13) Discovery Studio(14). The Hypogen algorithm works utterly different than the PHASE algorithm, directly operating on pharmacophore features instead of using grids as a proxy. First, a subset of the most active compounds is chosen. All possible pharmacophore hypotheses from the two most active compounds are enumerated and must fit a minimum subset of the remaining compounds in the most active subset to be considered by the algorithm. From this generated set of pharmacophore hypotheses, the ones matching a group of inactive compounds are removed in a follow-up phase. In a third final phase, small perturbations are introduced to the remaining hypotheses, which are then scored based on the RSME of predictions against the training set. The Hypogen refine algorithms extends this method by adding exclusion volumes and introducing another term in the loss function.

In contrast to PHASE, Hypogen is operating directly on pharmacophores without the need to provide the underlying molecules. However, a drawback of this method is still that it builds the quantitative models from a selected subset of highly active compounds. Even though refinement considers lessactive compounds, we would expect predictions for pharmacophores obtained from less active molecules to be worse due to missing domain knowledge. After the modelbuilding is done, no single quantitative model is selected, but a set of possible solutions is provided to the user, adding some ambiguity about the model's quality.

Having in mind the potential advantages of a quantitative method that is based on pure pharmacophoric representations, we developed and herein present a novel approach for the generation of quantitative pharmacophore models. Based on a small dataset of molecules and/or pharmacophores, the proposed algorithm will first find a consensus pharmacophore (merged pharmacophore) from all training samples. The input pharmacophores, or pharmacophores generated from the input molecules, will then be aligned to the merged pharmacophore. For each aligned pharmacophore, information regarding its position relative to the merged pharmacophore is extracted. This information is then used as input to a simple machine learning algorithm which derives a quantitative relationship of the merged-pharmacophores' features with biological activities.

Methods

The aim of this study is to develop a novel quantitative pharmacophore algorithm which can be applied on small datasets usually found in SAR studies of new drug targets. The algorithm allows the prediction of the pharmacophore's activity without the need for molecules in the alignment process. This allows the prediction of pure pharmacophore, for example when pharmacophores are obtained from a site pharmacophore modelling.

Model robustness

Data used to test the model robustness with cross-validation (CV) was pulled from ChEMBL(15). A list of popular QSAR targets was obtained from Cortés-Ciriano(16) et al. and was selected as the data source for target selection. The Uniprot-ID of these targets was used as query to get a list of compounds, along with the biological activity of these compounds on the target of interest, from ChEMBL by using the chemblwebresourceclient(15) python package. Retrieved compounds were filtered according to the following parameters:

standard_type: 'IC50' or 'Ki'

standard_units: 'nM'

standard_relation: '='

assay_type: 'B'

target_organism: 'Homo Sapiens'

Compounds were further grouped by 'target_chembl_id' and 'assay_chembl_id'. Molecular structures were generated from the canonical SMILES representation. The field 'standard_value', in the following referred to as 'activity value' or 'activity', was saved and used as an endpoint in the QSAR studies.

After pulling the data from ChEMBL and basic filtering, the datasets, comprised of molecules from a single assay, were further filtered for their applicability for QSAR. The rationale behind this is that

datasets should span a specific range of target values, here activity, for being analysed by QSAR. Here a cutoff of at least 3 logunits difference between the minimum and maximum activity value was applied; otherwise, the dataset got dismissed. Furthermore, a certain degree of heterogeneity in the distribution of activity values is necessary for successful QSAR analysis. The heterogeneity of datasets is often interpreted differently by multiple investigators. Therefore, the datasets were required to be as evenly distributed over the range of activity values as possible. Applying this constraint to a clearly defined measurement, the *KL* divergence of all datasets against a uniform distribution over the same range was calculated. The closer the *KL* divergence was to one, the more heterogeneous and the less clustered around a particular activity value the datasets were perceived to be. As a cut-off, 0.75 was used. All datasets with a *KL* divergence above the cutoff value were not considered for our studies.

The datasets were split into training and validation sets for cross-validation. The most and least active compound was hand-selected and put into the training set for all splits. Handselecting these two compounds was done to ensure the trained models only interpolate validation and test data, making sure the data was in the domain of the training set. Similar measures need to be taken during inference since the quality of predictions cannot be guaranteed for out of domain samples. In cross-validation, handpicking these two samples introduces a small bias, following which the individual folds are not 100% unique at each iteration. Apart from this slight bias, all samples were randomly separated into different datasets.

Datasets for cross-validation were generated randomly with a 5fold CVsplit, whereas each training split consisted of 20% of the data and the remaining 80% were used for validation (2080 split). In addition, a second CVsplit was conducted with 80% training and 20% validation data, which is the norm in machine learning (80 – 20 split). The 2080 split was done to mimic and evaluate performance in a typical SAR setting of a medicinal chemist, where the researcher might only have access to a meagre number of data points. Due to the small dataset sizes, training sets in the 2080 split typically consisted of 1015 samples.

Finally, for all molecules in the datasets, conformations were generated. For conformer generation, the commandline tool 'iconfgen'(17) from included in LigandScout(18) (Inte:Ligand GmbH(19), Vienna, Austria) was used. The maximum number of generated conformers was set to 25, all other settings were kept at their default values.

KullbackLeibler divergence

The KullbackLeibler (*KL*) divergence for each dataset was calculated according to Eq. 1, whereas *P* and *Q* denote discrete uniform distributions over activity values from a given dataset. *P* represents the estimated uniform distribution of the datasets, *Q* resembles the reference uniform distribution over activity values, *a* being the minimum and *b* the maximum. *P* is estimated by binning the activity values into *N* (sample size) bins. Each *P*(*x*) is defined by the frequency of activity values in each bin *x*.

$$D(P||Q) = KL(P, Q) = \sum_{x \in X} P(x) * \log \frac{P(x)}{Q(x)}$$

Equation 1: KL Divergence

PHASE

Data for performance evaluation alongside PHASE was obtained from a paper published by Dixon(8) et al., initially published by Debnath(20) et al. in 2002. Dixon et al. compared the 3D pharmacophore QSAR method implemented in PHASE against the Hypogen algorithm implemented in Catalyst. These results will be used as a baseline for evaluating the quantitative pharmacophore QSAR in this paper.

Conformations were created using iconfgen that is part of LigandScout. Splitting of training and test data was done as reported in the article by Dixon et al(8). No other modifications or filters were applied on the dataset, except for removing molecule number 67, for which no experimental activity value was provided by Dixon et al.

Baselines

Two general baselines were used to estimate the improvement of the quantitative pharmacophore over elementary QSAR models. The first baseline was a QSAR model built on the number of pharmacophore features per sample in the training set. The second model was constructed from a few standard physicochemical properties (number of HBond Donors / Acceptors, number of rotatable bonds, molecular weight, number of heavy atoms, cLogP(21), TPSA(22)). The baseline models were trained and tested on the same data splits as the quantitative pharmacophores. All baselines were trained with the same machine learning algorithm and the same parameters as the quant. pharmacophores to guarantee a fair comparison. Therefore, differences in performance between the quantitative pharmacophores and the baselines originate solely from the representation of molecules or pharmacophores.

Quantitative Pharmacophore Algorithm

The quantitative pharmacophore generation procedure is divided into two parts. At first, the merged pharmacophore model is generated, which serves as the basis for training a machine learning model for the prediction of new samples' activity.

Alignment template: As a starting point, a template is required to align all the other training samples. It can be either a pharmacophore or a molecule, although they are treated slightly different during the

initialisation:

Pharmacophore: Pharmacophores should be used as a template if the user has additional information about its target or has reason to believe that the template pharmacophore is of high importance to the user's project. This might be the case if the researcher has access to a crystal structure of the investigated target. For example, an interaction pharmacophore of a cocrystallised ligand in the protein binding site could yield valuable information. If this information is perceived to be relevant, then such an interaction pharmacophore could be used as the quantitative pharmacophore's initial template. All training samples are then aligned to the template via pharmacophore alignment. Samples can be either pharmacophores or molecules, in which case conformer ensembles for these molecules are required. Pharmacophores are generated for all conformations of a molecule. The conformation with the best-aligned pharmacophore is chosen.

Molecule: Instead of a pharmacophore, a molecule may be given as the initial template. Pharmacophores will be generated from each conformation of the molecule. Without additional information, none of these pharmacophores can be deemed more important than the others. However, alignment to any other molecule in the training set will result in a single best solution for the given pair of molecules. The conformation with the highest alignment score is chosen as the initial template. All other training samples will be aligned with this template. Depending on the counterpart molecule selected, the quality of the template and following the model will deviate. Therefore, templates and models are built for each sample in the training set. A supplied validation set will be used to select the best performing model. Due to multiple models trained in this step, model building will take considerably longer when choosing molecules over pharmacophores as the initial template. Nevertheless, training of all models is usually finished within a few minutes.

After a template was selected, the remaining samples from the training set are aligned with the template to create the merged pharmacophore. Instead of sequentially merging aligned pharmacophores and building the merged pharmacophore, all features of aligned pharmacophores are first added to a single pharmacophore data structure, also referred to as container in the following. Each feature is assigned the activity of its parent pharmacophore and stored alongside the feature. Furthermore, information about the orientation of directed interactions like Hbonding is disregarded, and all features modelling such interactions are represented by spheres. Once the training set got aligned to the template, the pharmacophore features collected in the container are clustered. A minimum distance hierarchical clustering algorithm is applied, whereas the cutoff can be set as a hyperparameter. The default value is equal to the default tolerance radius of pharmacophore features, 1.5 Å. Clusters are formed separately for each feature type: hydrophobic (H), aromatic (AR), positive/negative ionisable (PI, NI), HBond donor/acceptor (HBD, HBA).

Features belonging to a single cluster are then merged into a single feature, if possible, which represents the cluster. If no single feature can be created to represent the cluster, multiple features are placed. Clusters are merged with the priority of placing features as listed below:

Clusters containing a single feature: Clusters containing a single feature are represented by that one feature alone.

Clusters containing multiple features: For clusters containing various features, the goal is to find the smallest number of features required to represent the cluster. Ideally, a single feature is sufficient to represent the entire cluster. Finding multiple features to represent a cluster is not straightforward and has no objective best solution. In practice it was found that in the vast majority of cases a single feature is sufficient to represent a cluster.

A feature is declared to represent other features if the distance between their two centres is smaller or equal to the feature radius of one of the pharmacophore features. If their radii are different sizes, the smaller one is chosen as cutoff. Merged features are assigned the following properties:

List of activity values: The activity values of each feature merged into the representative. These values will be used to determine the importance of each feature.

Number of merged features: Indicates how many features were merged into the representative. This information will be used to determine the confidence in each feature.

Representative features are found in the following order. If at any point a representative was found, the algorithm goes on to the next cluster:

Selecting one of the features as representative: Each feature is checked whether it overlaps with all other features in the cluster. If so, this feature is chosen as the representative of the cluster. If none of the features fulfils the requirements, a new feature will be created instead.

New feature at the centroid: A new feature is created and placed at the centroid of all features. The new feature has no associated activity value and simply serves the purpose of merging the other features. All features are probed whether they are represented by the new feature.

New feature at the centre: If the new feature at the centroid does not satisfy all feature's requirements, a new feature is placed at the centre of all features. Once again, a check is run whether all features match the new feature.

Placing multiple features: Multiple representative features are only considered when all options to place a single feature were exhausted. As already mentioned, there is no single solution for placing multiple features to represent a cluster of features. Our algorithm iteratively finds multiple representatives, whereas each representative should merge as many features into the cluster as possible. Therefore, the most connected feature, determined by the number of overlapping features, is selected as a representative at each iteration. This process is repeated until all features in the cluster were merged. Each representative is assigned the list of activity values merged and the number of merged features.

A post-processing step is applied to the merged pharmacophore after clustering the features and creating a merged pharmacophore from all training samples. The post-processing step aims to remove noise and features with a widespread list of activity values from the quantitative pharmacophore model. For example, two molecules with the same scaffold but different residues will have the same pharmacophore features for the scaffold, but different peripheral features. Merging these two pharmacophores will be easy due to the shared features from the scaffold. However, suppose these two pharmacophores have different activity values. In that case, the scaffold's features will not add any useful information since they are included in the low and highly active sample. Only information obtained from peripheral features, which are unambiguous in their activity, are utilised by the model. Therefore, features are probed for their ambiguity in activity values, and only unambiguous features are retained, whereas ambiguous features are removed from the model. The following criteria determine the ambiguity of features:

A minimum number of activity values per feature is required. Thereby, only features the model is confident in are added.

The absolute difference in activity values of merged features must not be greater than the absolute difference in activity values of all features in the merged pharmacophore.

Following this, a ML model is trained on the aligned samples to predict the pharmacophores' activity values. In order to reason from the merged pharmacophore to activity values, appropriate features need to be extracted from the aligned samples. For this, the features of the aligned samples are matched against the merged pharmacophore, whereas a match is defined by an overlap of the query and reference pharmacophore feature tolerance spheres. For each match, the distance to the corresponding feature in the merged pharmacophore is calculated. The inverse of the distance value is used as input to the ML model, whereas a maximum value of the inverse distance can be set as a hyperparameter. Using the inverse value of the distance keeps a zero value for features without matches. It is important to note that the ML model will not consider features from samples without a matching feature in the merged pharmacophore.

Given that there are no corresponding features in the merged pharmacophore, it is acceptable to disregard potentially missing features during prediction since no information about their contribution to activity in that location is known. Therefore, including these features at inference would only increase noise and weaken the model's confidence in the prediction(23).

Once all distances are calculated, a vector of m features is obtained (m being the total number of features in the merged pharmacophore). Each sample is represented by such a vector describing its alignment with the merged pharmacophore. These vectors are used as input to the ML model. In order to comply with tiny dataset sizes, the choice of ML models is restricted to simple models like regularised linear regression or slightly more complex models like random forest trees. For random forest trees, their parameters are set to train only a few shallow trees to avoid overfitting. Optionally, a PCA might be performed beforehand on the input data, extracting only a few highly relevant features from the vector.

This process aims to increase the ratio from the number of samples to the number of features, thereby reducing the possibility of overfitting.

Before training the model, weights may be applied to the input data and the type of weight can be set as a hyperparameter. The following options are available:

No weights at all: The distances are treated as binary information. (least recommended option)

Weighted by distance: Explained above in the feature extraction step. This is the default option.

Weighted by the number of merged features: The third option emphasises features obtained from a higher number of pharmacophore features that have been merged into the respective feature. However, due to the post-processing step, features which the model is not confident in were already removed. Therefore, this type of feature weighting is optional and should only be considered by the user if particular emphasis wants to be given to that information.

The current version of the algorithm is implemented in Python using the chemical data processing (CDP) toolkit(24) for molecule and pharmacophore representations. Machine learning models are trained using the scikitlearn package(25). The code and all datasets are available at <https://github.com/StefanKohlbacher/QuantPharmacophore>.

Results And Discussion

The quantitative pharmacophore model is obtained by first creating a merged pharmacophore. Data from the merged pharmacophore and the training set is then used to fit a machine learning model. Training of the ML model is carried out with the same dataset as the merged pharmacophore was created from. Therefore, the dataset is required to have known activity values for each sample. Creating a merged pharmacophore as the underlying model has several advantages. For one, it keeps the model explainable and straightforward, unlike many other blackbox ML algorithms. Second, due to the familiar merged pharmacophore concept and representation, the model can quickly be adopted by scientist already familiar with such tools. The steep learning curve allows a medicinal chemist to iterate through ideas quickly.

As mentioned before, there are only a few tools currently available to the scientific community allowing scientists to perform QSAR from pharmacophores. Here we do not directly compare against these methods since the quantitative method described in this paper expands to domains not accessible by previous algorithms. Nevertheless, we show that the quantitative pharmacophore performs similar to the PHASE algorithm on molecule datasets. Furthermore, based on a broad set of commonly used protein targets for QSAR, we prove that the method shows robust performance over a wide variety of datasets.

The paper published by Dixon (8) et al. in 2006 describing the PHASE algorithm compares its method against an even earlier published paper(20) using the Hypogen algorithm to predict the activities of a

dataset. The quantitative pharmacophore was trained on the same training set, 20 samples, as described in the paper by Debnath(20) et al.. It was then evaluated on the holdout test set containing 57 molecules (originally 58, but one sample had no reported activity value). The reported RMSE and R^2 values on the test set of the PHASE algorithm were 0.822 and 0.407 (Fig. 1A), respectively. In contrast, the quantitative pharmacophore model could achieve an RMSE of 0.85 and an R^2 of 0.365 (Fig. 1B). These two models are comparable to each other in terms of quality, although the PHASE algorithm is still slightly better than the quantitative pharmacophores. However, it is important to keep in mind that both models were trained and evaluated on molecules, which is not the main focus of the method described here. Prediction of pharmacophores not obtained from molecules is one of the unique strengths of the quantitative pharmacophores. Alignment and prediction of such pharmacophores has not been possible before and prediction of pharmacophores with other methods still relied on molecules for alignment. Therefore, the method described here does not aim to compete against previous algorithms but rather expands the toolbox available to researchers.

Besides comparing our method against existing methods, CV was carried out on more than 250 distinct datasets to test the quantitative pharmacophores' general applicability across a wide range of datasets. All trainingvalidation runs used default parameters to gauge the quantitative pharmacophores' effectiveness when used outofthe box. Simple baseline models were built to demonstrate superior behaviour over standard methods. As baselines, the number of pharmacophore features in the training set was regressed against the activity endpoint. Furthermore, a set of simple physicochemical properties was calculated and used as a second baseline model input. To ensure a fair comparison, all baseline models used the same machine learning algorithm with default parameters as the quant. pharmacophores. Cross-validation runs were evaluated by calculating the mean RMSE and the standard deviation of the RMSE over the five individual runs (Fig. 2).

Mean RMSE values of the CV (8020 split; Fig. 2A) range from 0.19 to 1.31, with an average over all datasets of 0.62. Generally, an average mean RMSE of 0.62 over all datasets is perceived as quite good, considering that lower RMSE values are a strong indication of modelling the experimental noise in the datasets. With that in mind, the small number of datasets with mean RMSE values of 0.19 from CV are very likely to overfit. On the other hand, the worst RMSE of 1.31 was obtained without parameter optimisation, which is usually applied in praxis and could sharply increase the model's performance. Along with mean RMSE values, the standard deviation of the model's CV performance was calculated, which on average, was 0.18 across the five folds CV.

Evaluation of the 2080 split (Fig. 2B) yielded an average RMSE of 0.83 over all datasets, with a minimum RMSE of 0.40 and a maximum of 1.61. As expected, these models' performance is not as high as the average performance from models trained on the 8020 split. These results agree with the general notion that more data will improve a machine learning model's quality. However, considering that the training sets contained only 10 to 15 samples, the model's performances are respectably high. A valid concern often raised with such low training set sizes is the potential overfitting of the models. Here, we can exclude overfitting since the trained models were evaluated on validation sets four times larger than the

training sets. If models would overfit, the performance on the validation sets would be considerably worse and therefore not be in agreement with obtained results. Furthermore, the standard deviation over all datasets of the 2080 CV is much lower than the 8020 CV, Fig. 2D and C respectively. The low standard deviation further strengthens the point that the models are not overfitting any of the CV splits. Nevertheless, the small standard deviation compared to the 8020 split is surprising since smaller datasets are expected to increase the model's performance variance. Achieving a smaller variance during CV with smaller datasets further boosts confidence in robust quantitative pharmacophores.

In direct comparison to the baselines, the quantitative pharmacophore model was superior in ~ 9/10 cases measured by the RMSE of CV (8020 split as well as 2080 split). In the 8020 split CV, the quantitative pharmacophore could improve the mean RMSE by 34% over the pharmacophore features baseline and 27% over the physicochemical properties baseline. Similar results can be seen on the 2080 split, where 20% and 12% improvement was achieved, respectively.

Conclusion

Pharmacophores are widely applied in a qualitative manner for hit identification in virtual screening experiments and hardly any information can be found on the quantitative use of pharmacophore models. PHASE and Hypogen, only accessible in commercial packages, currently provide the only two algorithms which allow for quantitative insights on pharmacophore models. Targeting their drawbacks, such as alignment, the requirement of molecules for training, and user-friendliness, we present a novel quantitative pharmacophore generation algorithm for QSAR studies. The algorithm first creates a merged-pharmacophore from a given set of molecules and/or pharmacophores. Information obtained from aligning the training set to the merged-pharmacophore is then used to train a machine-learning model. We performed extensive cross-validation on a large variety of datasets and could show that quantitative pharmacophore models generated by our methods generalise well to many different datasets even without performing hyperparameter optimisation. The trained models achieved a mean RMSE of 0.61 during CV over >250 datasets. The datasets used for CV resemble sizes typically encountered in SAR settings by medicinal chemists. We could also demonstrate the robustness of our algorithm which is insensitive to small perturbations during training by achieving small variance in RMSE over 5-fold CV. Furthermore, on more than 90% of datasets, the generated quantitative pharmacophore models outperformed tested baselines, thus making our method a reasonable first approach for any researcher looking to get quantitative SAR insights on his data.

Declarations

Availability of data and materials

The datasets supporting the conclusions of this article are available in the Github repository <https://github.com/StefanKohlbacher/QuantPharmacophore>.

Competing interests

The Authors declare no competing interests.

Funding

This work was funded by the NeuroDeRisk project. The NeuroDeRisk project has received funding from the Innovative Medicines Initiative 2(26) Joint Undertaking under grant agreement No 821528. This Joint Undertaking receives support from the European Union's Horizon 2020(27) research and innovation programme and EFPIA(28).

Authors' contributions

The method was developed, implemented and validated by SK. The paper was jointly written by TS, TL and SK.

Acknowledgements

We acknowledge the NeuroDeRisk(29) consortium for supporting this project.

References

1. Hansch C, Maloney PP, Fujita T, Muir RM (1962) Correlation of Biological Activity of Phenoxyacetic Acids with Hammett Substituent Constants and Partition Coefficients. *Nature* 194(4824):178
2. Lo Y-C, Rensi SE, Torng W, Altman RB (2018) Machine learning in chemoinformatics and drug discovery. *Drug Discov Today* 23(8):1538–1546
3. Tetko IV, Engkvist O (2020) From Big Data to Artificial Intelligence: chemoinformatics meets new challenges. *J Cheminformatics* 12(1):74
4. Hong H, Chen M, Ng HW, Tong W (2016) QSAR Models at the US FDA/NCTR. *Methods Mol Biol Clifton NJ* 1425:431–459
5. Rogers D, Hahn M. Extended-Connectivity, Fingerprints (2010) *J Chem Inf Model* 50(5):742–754
6. Golbraikh A, Tropsha A (2003) QSAR Modeling Using Chirality Descriptors Derived from Molecular Topology. *J Chem Inf Comput Sci* 43(1):144–154
7. Cramer RD, Patterson DE, Bunce JD (1988) Comparative molecular field analysis (CoMFA). 1. Effect of shape on binding of steroids to carrier proteins. *J Am Chem Soc* 110(18):5959–5967
8. Dixon SL, Smondyrev AM, Knoll EH, Rao SN, Shaw DE, Friesner RA (2006) PHASE: a new engine for pharmacophore perception, 3D QSAR model development, and 3D database screening: 1. Methodology and preliminary results. *J Comput Aided Mol Des* 20(10–11):647–671
9. Schrödinger | Schrödinger is the scientific leader in developing state-of-the-art chemical simulation software for use in pharmaceutical, biotechnology, and materials research. [Internet]. [accessed 2021 Mar 25]

10. Kramer C, Dahl G, Tyrchan C, Ulander J (2016) A comprehensive company database analysis of biological assay variability. *Drug Discov Today* 21(8):1213–1221
11. Hu Y, Stumpfe D, Bajorath J (2017) Recent Advances in Scaffold Hopping. *J Med Chem* 60(4):1238–1246
12. Li H, Sutter J, Hoffman R. HypoGen: An Automated System for Generating 3D Predictive Pharmacophore Models. In: *Pharmacophore Perception, Development and Use in Drug Design*; Guner, O, Ed; International University Line: La Jolla, CA,. 2000. p. 171–89
13. 3D Design & Engineering Software - Dassault Systèmes® [Internet]. [accessed 2021 Mar 25]
14. Guner O, Clement O, Kurogi Y (2004) Pharmacophore Modeling and Three Dimensional Database Searching for Drug Design Using Catalyst: Recent Advances. *Curr Med Chem* 11(22):2991–3005
15. Davies M, Nowotka M, Papadatos G, Dedman N, Gaulton A, Atkinson F et al (2015) ChEMBL web services: streamlining access to drug discovery data and utilities. *Nucleic Acids Res* 43(W1):W612–W620
16. Cortés-Ciriano I, Škuta C, Bender A, Svozil D (2020) QSAR-derived affinity fingerprints (part 2): modeling performance for potency prediction. *J Cheminformatics* 12(1):41
17. Poli G, Seidel T, Langer T (2018) Conformational Sampling of Small Molecules With iCon: Performance Assessment in Comparison With OMEGA. *Front Chem* 6:229
18. Wolber G, Langer T (2005) LigandScout: 3-D Pharmacophores Derived from Protein-Bound Ligands and Their Use as Virtual Screening Filters. *J Chem Inf Model* 45(1):160–169
19. Inte:Ligand: Your partner for in-silico drug discovery [Internet] (2019) [accessed 2019 Feb 21]
20. Debnath AK (2002) Pharmacophore Mapping of a Series of 2,4-Diamino-5-deazapteridine Inhibitors of *Mycobacterium avium* Complex Dihydrofolate Reductase. *J Med Chem* 45(1):41–53
21. Viswanadhan VN, Ghose AK, Revankar GR, Robins RK (1989) Atomic physicochemical parameters for three dimensional structure directed quantitative structure-activity relationships. 4. Additional parameters for hydrophobic and dispersive interactions and their application for an automated superposition of certain naturally occurring nucleoside antibiotics. *J Chem Inf Comput Sci* 29(3):163–172
22. Prasanna S, Doerksen R (2009) Topological Polar Surface Area: A Useful Descriptor in 2D-QSAR. *Curr Med Chem* 16(1):21–41
23. Sutton C, Boley M, Ghiringhelli LM, Rupp M, Vreeken J, Scheffler M (2020) Identifying domains of applicability of machine learning models for materials science. *Nat Commun* 11(1):4428
24. Seidel T. Chemical Data Processing Toolkit, GitHub repository: <https://github.com/aglanger/CDPKit> [Internet]. 2021 [accessed 2021 Mar 19]
25. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O et al (2011) Scikit-learn: Machine Learning in Python. *J Mach Learn Res* 12(85):2825–2830
26. Homepage [Internet]. IMI Innovative Medicines Initiative. [accessed 2021 Mar 17]
27. Horizon (2020 [Internet]) Horizon 2020 - European Commission. [accessed 2021 Mar 17]

28. Homepage EFPIA [Internet]. [accessed 2021 Mar 17]

29. NeuroDeRisk. NeuroDeRisk | Neurotoxicity De-Risking in Preclinical Drug Discovery [Internet] (2019) [accessed 2021 Mar 17]

Tables

Table 1: Target list used for model validation

Target	Uniprot ID	CHEMBL ID
Alpha-2a adrenergic receptor	P08913	CHEMBL1867
Tyrosine-protein kinase ABL	P00519	CHEMBL1862
Acetylcholinesterase	P22303	CHEMBL220
Androgen receptor	P10275	CHEMBL1871
Serine/threonine-protein kinase Aurora-A	O14965	CHEMBL4722
Serine/threonine-protein kinase B-raf	P15056	CHEMBL5145
Cannabinoid CB1 receptor	P21554	CHEMBL218
Carbonic anhydrase II	P00918	CHEMBL205
Caspase-3	P42574	CHEMBL2334
Thrombin	P00734	CHEMBL204
Cyclooxygenase-1	P23219	CHEMBL221
Cyclooxygenase-2	P35354	CHEMBL230
Dihydrofolate reductase	P00374	CHEMBL202
Dopamin D2 receptor	P14416	CHEMBL217
Norepinephrine transporter	P23975	CHEMBL222
Epidermal growth factor receptor erbB1	P00533	CHEMBL203
Estrogen receptor alpha	P03372	CHEMBL206
Glucocorticoid receptor	P04150	CHEMBL2034
Glycogen synthase kinase-3 beta	P49841	CHEMBL262
HERG	Q12809	CHEMBL240
Tyrosine-protein kinase JAK2	O60674	CHEMBL2971
Tyrosine-protein kinase LCK	P06239	CHEMBL258
Monoamine oxidase A	P21397	CHEMBL1951
Mu opioid receptor	P35372	CHEMBL233
Vanilloid receptor	Q8NER1	CHEMBL4794

The hyper-parameters of the best model were as the following:

Table 2: Hyperparameters of trained quantitative pharmacophore model on datasets from Debnath et. al.

Parameter	Value
fuzzy	True
modelType	RandomForest
threshold	1
weightType	Distance
maxDepth (of ML-model)	3
nEstimators (of ML-model)	20

The predictions of the model on the training and test set can be found in the following tables:

Table 3: Predictions on test set of quantitative pharmacophore model

Index	pIC50 exp.	pIC50 pred.	Index	pIC50 exp.	pIC50 pred.
<i>0</i>	5.64	7.13	<i>35</i>	8.57	8.10
<i>1</i>	6.6	6.66	<i>37</i>	7.74	7.02
<i>2</i>	6.07	6.75	<i>38</i>	6	6.78
<i>3</i>	6.43	7.24	<i>42</i>	8.17	7.86
<i>4</i>	6.92	6.94	<i>43</i>	6.3	6.22
<i>5</i>	6.52	6.77	<i>44</i>	5.47	7.71
<i>6</i>	6.56	6.60	<i>45</i>	7.17	6.13
<i>7</i>	7.16	8.11	<i>47</i>	6.36	6.08
<i>8</i>	7.77	7.59	<i>48</i>	8.44	7.06
<i>9</i>	7.72	8.14	<i>49</i>	8.15	8.11
<i>10</i>	6	7.36	<i>52</i>	6.85	8.34
<i>11</i>	5.72	7.70	<i>53</i>	8.23	8.11
<i>12</i>	8.09	6.75	<i>54</i>	5.47	6.28
<i>13</i>	8.21	8.04	<i>55</i>	8.37	8.02
<i>14</i>	7.44	6.33	<i>59</i>	7.82	7.63
<i>15</i>	7.48	6.75	<i>61</i>	8.21	7.70
<i>18</i>	8.39	8.34	<i>62</i>	5.57	6.24
<i>19</i>	6.82	7.01	<i>63</i>	8.5	8.28
<i>20</i>	8	8.11	<i>64</i>	8.21	8.34
<i>21</i>	6.17	8.04	<i>65</i>	5.89	7.18
<i>22</i>	7.36	6.75	<i>66</i>	5.6	6.06
<i>24</i>	8.3	8.08	<i>67</i>	8.37	8.34
<i>25</i>	8.55	7.51	<i>68</i>	6	6.16
<i>26</i>	8.06	6.20	<i>69</i>	6.38	6.55
<i>27</i>	8.28	8.13	<i>70</i>	7.24	6.88
<i>28</i>	5.92	6.22	<i>71</i>	7.85	8.22
<i>30</i>	8.26	8.04	<i>72</i>	8.34	8.28
<i>32</i>	8.17	8.04	<i>76</i>	5.14	6.65

Table 4: Predictions of quantitative model training set

Index	pIC50 exp.	pIC50 pred.
16	7.82	7.48
17	7.57	7.25
23	6.09	5.98
29	7.70	7.53
31	8.33	7.46
33	8.00	7.71
36	4.51	6.78
39	5.21	6.10
40	8.43	8.10
41	8.42	8.34
46	8.29	7.76
50	5.41	6.38
51	6.23	6.46
56	6.07	6.19
57	6.00	6.23
58	6.15	6.47
60	8.40	8.25
73	5.59	6.65
74	8.70	8.42
75	6.36	6.22

Figures

Figure 1

Test set performance of PHASE (A) and the quantitative pharmacophore model (B)

Figure 2

Aggregated CV performance over >250 datasets from quantitative pharmacophores and two baseline methods. A: Mean RMSE values of models in 80-20 split. B: Mean RMSE values of models in 20-80 split. C: Mean standard deviation of models in 80-20 split. D: Mean standard deviation of models in 20-80 split.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Appendix.docx](#)
- [Additionalfile1.csv](#)
- [Additionalfile2.csv](#)