QSARtuna: an automated QSAR modelling platform for molecular property prediction in drug design

Lewis Mervin[1]*, Alexey Voronov[2], Mikhail Kabeshov[2], Ola Engkvist[2, 3]

[1]Molecular AI, Discovery Sciences, R&D, AstraZeneca, Cambridge, UK

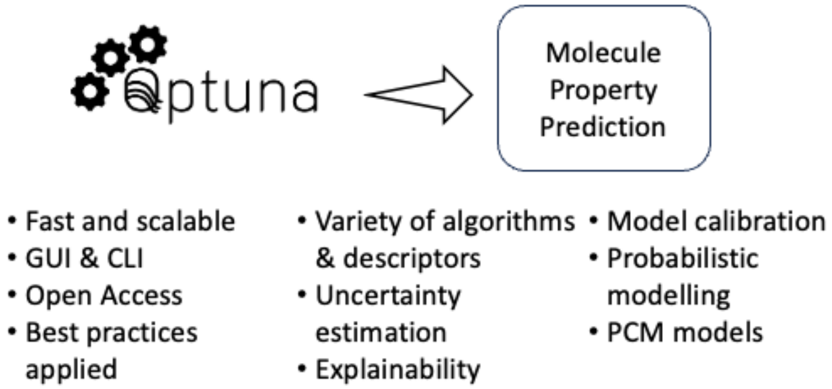
[2]Molecular AI, Discovery Sciences, R&D, AstraZeneca Gothenburg

[3]Department of Computer Science and Engineering, University of Gothenburg, Chalmers University of Technology, Gothenburg, Sweden

*lewis.mervin1@astrazeneca.com

Keywords: Molecular property prediction, target prediction, model building, hyperparameter optimization, QSAR, active learning, ChemProp, PCM

TOC:



- Fast and scalable
- GUI & CLI
- Open Access
- Best practices applied

- Variety of algorithms & descriptors
- Uncertainty estimation
- Explainability

- Model calibration
- Probabilistic modelling
- PCM models

Abstract

Machine-learning (ML) and Deep-Learning (DL) approaches to predict the molecular properties of small molecules are increasingly deployed within the design-make-test-analyse (DMTA) drug design cycle to predict molecular properties of interest. Despite this uptake, there are only a few automated packages to aid their development and deployment that also support uncertainty estimation, model explainability and other key aspects of model usage. This represents a key unmet need within the field and the large number of molecular representations and algorithms (and associated parameters) means it is non-trivial to robustly optimise, evaluate, reproduce, and deploy models. Here we present QSARtuna, a molecule property prediction modelling pipeline, written in Python and utilising the Optuna, Scikit-learn, RDKit and ChemProp packages, which enables the efficient and automated comparison between molecular representations and machine learning models. The platform was developed considering the increasingly important aspect of model uncertainty quantification and explainability by design. We provide details for our framework and provide illustrative examples to demonstrate the capability of the software when applied to simple molecular property, reaction/reactivity prediction and DNA encoded library enrichment analyses. We hope that the release of QSARtuna will further spur innovation in automatic ML modelling and provide a platform for education of best practises in molecular property modelling. The code to the QSARtuna framework is made freely available via GitHub.

Introduction

A typical drug design project consists of a design-make-test-analyse (DMTA) cycle aiming to optimise small molecules for activity against a desired protein target whilst at the same time maintaining a desirable absorption, distribution, excretion and toxicity (ADMET) profile, thereby improving chances for *in vivo* efficacy[1]. Since measuring such properties requires substance samples and is resource- and time-consuming, cycle times can be slow and compound prioritisation might be cumbersome[2, 3].

To address this, Machine learning (ML) and artificial intelligence (AI) approaches have been increasingly integrated into medicinal chemistry projects[4-6]. Here their routine use towards Quantitative Structure Activity Relationship prediction (QSAR) accelerates DMTA cycle times[7-9]. As shown in **Figure 1**, their application is designed to direct resources towards prospective screening experiments, and they have been used to screen extensive compound databases and to optimise efficacy[10-12]. *In silico* safety assessment can also minimize ethically concerned activities, such as animal or human experimentation[13, 14]. QSAR has also been combined with other fields such as molecular *de novo* design, where molecule property prediction is used to direct the objective function (capturing [un]-desired properties) of generative algorithms toward desirable chemical space[15-17], or coupled with active learning approaches to optimise free energy calculations[18]. Other applications include the prediction of chemical reaction yields, where reactants and yield are provided as training data[19].

The development of novel algorithms capable of rationalising complex relationships between chemical and biological information[20, 21], exponentially growing chemical and biological space added to molecule databases[22], falling cost of computational resources [23, 24], and MLOps systems for accessing production-level models[25] have spearheaded the development and use of QSAR models in practice[26-28]. Despite this, the assortment of workflows, algorithmic methods, and parameters means training and updating models is non-trivial and finding the relatively optimal modelling setup is a time-consuming task for data scientists. Consequently, there is a need to compare different models for specific properties reproducibly, efficiently, and robustly across different molecular representations and algorithms.

A platform offering this functionality should maintain and update QSAR models throughout their "life cycle" and needs to involve the standard steps critical for reliable model building in a temporal setting. The automatic evaluation of the ML stack (including the sequential steps of data ingestion, pre-processing and model training) is a distinct area identified as AutoML[29]. The application of AutoML toward the field of molecular property prediction has only partially addressed despite the early attempts to attract attention towards this unmet need[30]. There remains a lack of robust, modular and scalable platforms for QSAR modelling, though some open-source tools have been presented (see **Table 1** for an overview). SL Dixon, J Duan, E Smith, CD Von Bargen, W Sherman and MP Repasky [31] developed a machine-learning application (AutoQSAR) for automated QSAR modelling. eTOXlab [32] and offers an alternative automated QSAR framework, but is no longer maintained and requires advanced Python programming skills. An online alternative OCHEM[33] is available, however the cloud-based infrastructure renders the software unsuitable for private or sensitive data. R Cox, DV Green, CN Luscombe, N Malcolm and SD Pickett [30] designed a Pipeline Pilot web application (QSAR Workbench) although this is restricted to Pipeline Pilot users. Automated Predictive Modeling[34] is also available but demands expert technical skills and significant resources for model development and maintenance. More recently, TranScreen provides a transfer-learning setup based on graph convolutional neural networks and focuses on small imbalanced data sets, though algorithmic choice is restricted to only deep-learning methods[35]. S Kausar and AO Falcao [36] also proposed an automated framework for QSAR model building, but this is based on KNIME and requires expert knowledge for their implementation with a complicated interface. AMPL[37] was also developed as a modelling pipeline as an open-source software suite, allowing users to build models for a wide array of molecular properties. It extends the open source DeepChem library, supports an array of ML and molecular featurization tools and offers uncertainty quantification. Despite this, only four sets of algorithms are available, a restricted number of four descriptors are available (the MOE descriptors also require a license), and no GUI is provided. PREdictive modeling FramEwoRk (PREFER) was recently proposed by J Lanini, G Santarossa, F Sirockin, R Lewis, N Fechner, H Misztela, S Lewis, K Maziarz, M Stanley, M Segler, et al. [38]. In this package, popular libraries are used for hyperparameter optimization, with the authors stating the most important factor being the ability to customise the framework. AutoSklearn is supported by an active community, but the package relies on notebooks and requires a detailed four step installation process. Uni-QSAR was recently published[39], though this software must combine 1D tokens, 2D topology graphs, and 3D conformers to generate learnt representations and does not offer the same level of functionality or ease of use compared to other packages. Other approaches towards automated QSAR procedures are also available but are tailored to specific settings such as blood-brain barrier penetration and aqueous solubility[40], Leishmania High-Throughput Screening Data[41] or Gaussian Processes[42], which limits applicability. Molflux[43] was also recently released as a foundational package for molecular predictive modelling, though this platform does not optimise for algorithm hyperparameters. A variety of data mining and automation tools could offer the ability to develop custom pipelines, such as Pipeline Pilot[44], KoNstanz Information MinEr (KNIME)[45], Orange[46], Taverna[47], Kepler[48] and the Loni Pipeline[49]. However, workflow managers require specific competency to design or run pre-existing configurations, and developing custom workflows requires time and effort. Ideally a platform should provide both a CLI and GUI, without the need for proprietary software or licenses, expert knowledge or complicated installation steps. It should be developed with a popular maintainable programming language, with ability to use the state-of-the-art (well-maintained) open access packages for additional functionality. Uncertainty estimation and model explainability should also be considered during the design, given the increased focus placed onto these aspects of modelling[2, 4, 50], and since this will facilitate decision making when models are used in production.

In this vein, we have developed QSARtuna; a platform which employs, to the best of our knowledge, all best practices from the field to deploy predictive QSAR models into production. The platform deals with data input, molecule standardardisation, deduplication, splitting, hyperparameter optimization and deployment in an easy to use, modular way. The platform is released as open source under a permissive license for educational purposes, and to facilitate further innovation. It is intended to be a living project with continuous updates and new features. Here we outline the platform structure showing the workflow, implementation and the additional functions offered. We provide easy-to-follow examples using the tool toward three different types of applications reflecting the breath of modelling tasks a modern ML platform needs to handle; aqueous solubility prediction, probabilistic reactivity prediction and calibrated DEL enrichment classification. We consider they represent a diverse range and reflects the emerging landscape of popular QSAR tasks and demonstrates the versatility of the platform.

Implementation

The workflow is structured around three steps:

1. *(Bayesian) Hyperparameter Optimization:* Train many models with different parameters using Optuna. Only the training dataset is used using cross-validation.
2. *Build (Training):* Select the optimal model from Optimization, and optionally evaluate its performance on the test dataset.

3. *“Producition build (Re-training):* Re-train the best-performing model on the merged training and test datasets with the drawback that there no data remains. for evaluating the resulting model, but a large benefit that the final model is trained on all available data.

A detailed overview of the standardised protocol toward automated QSAR modelling is shown in the **Supporting Information Figure 1**.

The QSARtuna workflow starts with data preparation including the import of molecular structures and corresponding biological activity data for a specified molecule property prediction task. Several sanity checks are performed on the input data, including valid response values and input molecules. An optimisation protocol is next initiated, where internal validation is used to develop a QSAR model by following a rigorous internal and external validation process. Here, an initial split of data partitions training instances into internal and external validation, to avoid data leakage. This is a critical step, where many different splitting strategies are available to afford a more realistic evaluation of model performance in practice. Hyper-parameter optimisation is performed on the internal split using the Optuna package; a framework performing Bayesian hyperparameter optimisation across a set of molecular descriptors and algorithms. Finally, a selected model can be created by initiating a “production build”, which can comprise both internal and external training instances (model trained on all available data with the caveat of no performance assessment). Hence, our open-source automated workflow embeds all the tools and steps necessary to perform all steps of the QSAR life cycle by following best practices. The workflow is easily applied without having expertise in ML or programming.

Data Preparation

One of the most important procedures in building QSAR models is the appropriate pre-processing of the data prior training[51-53]. This section describes how the steps implemented in QSARtuna to ensure best practice in this regard.

QSARtuna expects inputs to be in the form of a CSV or SDF file and the resources required to reproduce the results in this work are provided in the **Supporting Information File 1**. The pipeline provides an opportunity for automation since queries can be polled for continuous updates from project teams and is hence intended to cater for a variety of approaches.

Input data is retrieved and processed retaining only the requested property of interest records, and any information related to chemical structures and assays (for example co-variates corresponding to time/date/protein) or side-information tasks for use in multi-task learning. Since the objective of QSAR is to quantify a ligand–molecular property values, any response column value may be utilised and related to the algorithm for training. Validation also includes the identification of missing data, duplicates and dealing with several forms of the same molecule (including salt groups).

Next, deduplication of distinct compound replicates (based on the canonicalized SMILES of user inputs) is performed, where the current options are:

- Keep First and Keep Last: keep the first or last occurrence
- Keep Random (with a seed): keep a random observation
- Keep Minimum and Keep Maximum: keep min or max
- Keep Average: take the average
- Keep Median (default in QSARtuna): take the median
- Keep All: all observations are retained

The default option is Keep Median, which is recommended due the ability to utilise all experimental data in one value (accounting for experimental variability across replicates), whilst being robust to outliers.

Response value transformations

Scaling or transforming user response columns to normalise highly varying values in raw data is a common practice for proper training of a predictive model. QSARtuna may transform input labels so that log-scaled or irregularly distributed data can be transformed to a normal distribution as required for many ML inputs. Data can be transformed with different logarithmic functions, but this is deactivated by default, assuming data is already normalised.

Data partitioning

To facilitate external predictive performance assessment, input data is divided into the internal training set and external validation set using different options. By default, the platform applies a stratified (real-valued) shuffle split. For classification, data is split ensuring the same distribution of classes. For regression, data is split according to a binning scheme of response values, ensuring that the (binned) distribution of regression values for modelling are consistent between test and training sets. This split is robust for both classification and regression settings and provides a good baseline for most cases. A variety of other splits, including a scaffold-based split (to emulate when models may be used for scaffold hopping) are also available. Hence there are a wide array of splitting strategies capturing most user applications. Next, the internal training set is further split using a K-fold cross validation process (either stratified or random) for internal hyperparameter optimisation, evaluation, and selection. The external split is never used for any feature selection or model training procedure, to avoid leakage. The full list of splitting strategies in QSARtuna are as follows:

- Random
- Stratified (real-value) shuffle
- Temporal
- Scaffold-based
- Predefined (from a user column)

Descriptor calculation

QSARtuna calculates several molecular descriptors which are parallelised and cached to reduce trial runtime. Users can submit precalculated molecular descriptors using the precomputed descriptors option. The full complement of descriptors currently includes:

- RDkit circular fingerprints (Morgan-like)
- RDkit circular fingerprints (Morgan-like) with counts
- RDKit physchem descriptors
- Avalon[54]
- MACCS[55]
- Jazzy[56]
- Composite descriptors (concatenate any combination of descriptors together)
- Predefined descriptors
- Scaled descriptors (ensures custom descriptors are scaled)

Model Selection

A variety of different algorithms for classification or regression are provided. We apply many popular ML approaches, such as neural networks (ChemProp[57]), support vector machines (SVMs), random forests (RF). We also provide an implementation of the Probabilistic Random Forest (PRF)[58] for use with the probabilistic data transform, which has been shown to improve uncertain bioactivity predictions[59]. Other algorithms are easily integrated given the modular nature of QSARtuna.

Each QSARtuna trial is evaluated via the primary performance metric (this is ROC-AUC or negated Mean Squared Error (MSE) by default) which is customisable. QSARtuna also calculates other Scikit-learn metrics in addition to BEDROC (implemented via RDKit)[60]. All metrics are available for review by the user though only the primary metric is used as an objective function in Optuna trials. The user may (optionally) specify multi-parameter optimisation for minimisation of the standard deviation of primary performance scores across the folds, thereby suggesting descriptor and algorithm pairs that are more generalisable across splits (and therefore in production). External validation is finally performed for the realised model on the external test set.

Functions offered by QSARtuna compared to other platforms

Probabilistic modelling transformation

Since molecule properties derived from experiments have reproducibility limits due to experimental errors, models based on this data have such unavoidable error influencing performance. This should ideally be factored into modelling and consequently a probabilistic transform of the activity scale is available in QSARtuna, based on the approach performed here[59].

With this setting enabled, QSARtuna treats compound response labels as probability distribution functions (rather than deterministic values) on a per-threshold basis based on the cumulative distribution function (CDF) of a normal distribution. The activity values become represented in a framework in-between a classification and regression architecture (given dataset experimental variability), with philosophical differences from either approach. Compared to classification, this enables better representation of factors increasing/decreasing inactivity. Conversely, one can utilize all data (even delimited/operand/censored data far from a cut-off) at the same time as considering the granularity around the decision boundary, compared to a conventional regression framework. Enabling this setting thereby combines characteristics from both classification and regression settings.

Probability calibration

Probability calibration methods are provided via the Calibrated Classifier with Cross Validation option (based on the inductive cross-validated method in Scikit-learn). The available functions are Sigmoid, Isotonic regression and VennABERS[61], and a review of those calibration methods for QSAR has been performed here[62]. Calibration can improve probabilities by representing the ground truth and should be useful for making decisions under uncertainty.

Uncertainty estimation

QSARtuna offers uncertainty via three different methods:

1. VennABERS discordance, based on the "Uses for the Multipoint Probabilities from the VA Predictors" from [62]
2. Ensemble uncertainty (ChemProp models trained with random initialisations).
3. Dropout uncertainty at inference time (ChemProp models)
4. Model Agnostic Prediction Interval Estimator (MAPIE)[63] (uncertainty for regression)

Model Explainability

Model explainability is incorporated into QSARtuna using two different approaches that focus on the input descriptors for molecules. Each depend on the algorithm chosen:

1. SHapley Additive exPlanations (SHAP)[64] (available for all models)
2. ChemProp interpret (available for ChemProp models and based on the interpret function in the original package)

Results and Discussion

This section demonstrates three diverse and relevant use cases for QSARtuna:

1.) ESOL aqueous solubility regression[65]

2.) Probabilistic reactivity prediction (evaluated via regression metrics)[66]

3.) DNA encoded library (DEL) enrichment classification[67]

Each seeks to exemplify platform capabilities reflecting the latest prediction task trends in QSAR. The datasets are provided in the **Supporting Information File 1** for reproducibility.

Solubility modelling

We first applied QSARtuna toward the ESOL solubility dataset which represents a typical regression task based on an assay readout important in early drug discovery. An overview of external performance is provided in **Table 2**. Results show a marked improvement during scaffold-based testing when using QSARtuna over conventional approaches; with an improvement in Pearson correlation from 0.264 to 0.636 (margin of 0.372) between the simple RF & ECFP (No optimisation) baseline compared to a full QSARtuna run (150 start-up trials, proper 300 trials) optimising for low standard deviation across hyper-parameter folds. Optimising for folds improved performance by a Pearson correlation margin of 0.130, indicating this approach can be used for better selection of hyperparameters in the analysis presented here. To our knowledge minimising for standard deviation across folds in an automated multi-parameter optimisation in this manner is not available in alternative open-source AutoML platforms. Results for the RF grid search also highlight the clear benefit for performing proper optimisation within QSARtuna, since the grid optimised RF achieved a Pearson of only 0.297.

The stratified split also showed benefit in performing optimisation over a baseline, with improvements from 0.725 to 0.907 for the RF and ECFP model and obtained QSARtuna models, respectively. QSARtuna identified the same optimal (start-up) trial for this splitting evaluation approach, so there is no benefit to activating the multi-parameter optimisation approach for standard deviation for this analysis. The RF grid search present only modest performance gains over the baseline model, with a Pearson 0.763, further highlighting the importance of fully optimising both algorithm and descriptor spaces.

Taken together, results highlight the benefit in performing hyper-parameter optimisation using the QSARtuna package for a solubility dataset and present evidence for usefulness of the unique functionality offered by our package. Although some additional latency is introduced by the time taken for optimisation, we consider this is mitigated by substantial performance gains as observed for the ESOL dataset.

Reactivity modelling

QSARtuna was applied to a Buchwald-Hartwig reactivity prediction dataset[66] to demonstrate its application to a different molecule property prediction endpoint. Probabilistic thresholding of the regression scale was implemented to outline this functionality, which to our knowledge is not offered by alternative software. In this procedure, reactivity response values were discretised using an activity threshold boundary of 5 and provided a standard deviation of 2, thereby accounting for experimental variability of reactivity assays within the modelling procedure and representing the reactivity prediction task in a probabilistic framework. In this setting, a yield of 5 is assigned a likelihood score of 50%, whereas scores of 2.5 or 7.5 would be assigned scores of 10.6% and 89.4%, respectively. Yields below or above the standard deviation range converge to the minimum and maximum values of 0% and 100%, respectively, thereby allowing the use of even delimited (qualified) values of “<” or “>”.

Results are shown in **Table 3** and demonstrate that QSARtuna with probabilistic modelling combined with PRF performs with the most relatively optimal performance of any of the approaches evaluated; with an improvement in Pearson correlation from 0.880 and 0.967 (margin of 0.087) between the simple RF & ECFP (No optimisation or probabilistic modelling) baselines when compared to a full QSARtuna run (15 start-up trials, proper 15 trials). This finding highlights the clear benefits for representing the reactivity scale in this manner and accounting for uncertainty near the decision boundary, which to our knowledge, is a unique option offered by QSARtuna.

DEL modelling

In this section we chose to evaluate QSARtuna performance for a DEL enrichment dataset from KS Lim, AG Reidenbach, BK Hua, JW Mason, CJ Gerry, PA Clemons and CW Coley [67], since this task type represents a more recently popularised prediction problem, comprising a highly imbalanced classification set with large numbers of enrichment response values. This provides an opportunity to not only benchmark the software on a larger, more noisy data set, but also to demonstrate the calibration methods available in QSARtuna, to obtain better probability estimates representing the ground truth. This is an important aspect of model behaviour to consider since the outputs from poorly calibrated models can be misleading and not always actionable.

Results from our DEL classification analysis is presented in **Table 4**. The findings highlight the clear benefit for using QSARtuna with the VennABERS calibration approach, since the VennABERS scaling has the most relatively optimal ROC AUC whilst also maintaining the highest negated Brier score loss (which indicates superior calibration performance), with 0.906 and -0.003, respectively. To our knowledge, this approach is a unique option offered by QSARtuna.

We next analysed how well calibrated the VennABERS (optimal QSARtuna run) is compared to a (uncalibrated) QSARtuna model obtained without the VennABERS functionality activated, for a stratified subset of 3,800 test set compounds. Results provided in **Supporting Information Figure 2** illustrate a reliability plot (a common method to evaluate model calibration) relating the ground truth likelihood of compounds obtaining a positive prediction as a function of different probability bins. Findings clearly demonstrate the superior calibration performance of the model obtained by the VennABERS predictor over the uncalibrated baseline (a higher proportion of compounds are assigned estimates closer to the ideal) as outlined by markers near to the diagonal (ideal) line. Again, this represents a key benefit for QSARtuna over alternative software (when considering model calibration).

Discussion & Conclusion

In this work, we present a robust, modular, and extendable platform designed as a QSAR modelling pipeline to obtain robust predictive models for molecule property prediction tasks. The pipeline can perform fully automated QSAR modelling to assist all users including those not an expert in the ML field or those which have limited knowledge in data preparation and QSAR best practices.

Since the training of a most relatively optimal model is reliant of many critical and time-consuming steps (including data collection and processing, data representation via descriptors, model training, hyper-optimisation, and validation), this workflow completely automates these laborious processes. The following are the main advantages of the QSARtuna framework:

- Automatically deployable in a three-step framework
- Data ingestion (selecting only the property of interest) offering classification and regression
- Deduplication, removing invalid/missing data
- Descriptors calculation across a wide range of state-of-the-art options
- Data normalization, standardisation and transformation (including probabilistic transformation for probabilistic modelling)
- Best practice validation procedure using internal and external splits
- State-of-the-art interpretation or explainability methods available
- Model calibration using inductive methods
- Uncertainty quantification options depending on the algorithm selected
- Support for model architectures utilising auxiliary domain information (e.g. Proteochemometric [PCM] modelling, dose, timepoint, etc.)

Due to its modular nature, QSARtuna is transparent in comparison to alternative black-box solutions available from other platforms. Our extensible and highly customisable package will aid the development of robust predictive models and provide an ideal framework for a predictive model life cycle. It ensures the same protocol is used for updating models as new data becomes available, thereby improving reproducibility. By integrating the latest explainability and uncertainty quantification, we intend for the realised models to have more impactful and actionable predictions when used in production. QSARtuna is made open source as an automated QSAR modelling framework to spur further innovation in the field. We hope that the most important aspects of QSAR modelling are addressed and consistently applied when using QSARtuna.

Acknowledgements

None

Figures

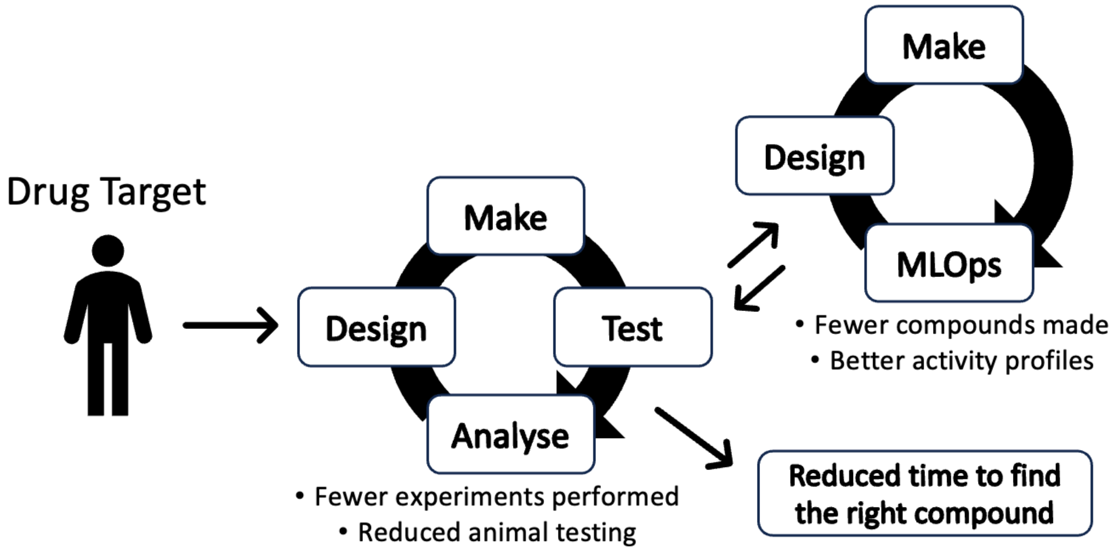

Drug Target
Make
Design
Test
Analyse
• Fewer experiments performed
• Reduced animal testing
Make
Design
MLOps
• Fewer compounds made
• Better activity profiles
Reduced time to find the right compound

Figure 1. Importance of integrating well-trained models into the drug design process. A well-established infrastructure of model hosting (MLOps) and re-training of models is required for effective model deployment. The principal way to impact the cycle via modelling approaches is to make the best up-to-date models available to all scientists at the point of Design

Tables

Table 1. QSARtuna comparison with alternative open-source software for molecule property prediction.

Software	Dataset modellability/ pre-modelling analysis	Custom Splitting techniques	Number of descriptors	Composite descriptors	Custom descriptors?	Custom train/test splits?	Shallow models	Neural network-based algorithms	Inductive model calibration	Uncertainty estimation	Explainability	Multi-parameter optimisation?	Probabilistic transform
QSARtuna	No	Yes	8	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
AMPL	No	No	4	No	No	No	Yes	Yes	No	Yes	No	No	No
PREFER	No	No	4	No	No	No	Yes	Yes	No	No	No	No	No
Uni-QSAR	No	No	5+	No	No	No	Yes	No	No	No	No	No	No

Table 2. ESOL prediction performance demonstrates the value of optimising parameters. Hyperparameter optimisation obtains better models regardless of split method considering all six performance metrics evaluated.

Run no.	Modelling Approach	Split Methods		Time			External Performance					
		External	Internal (hyper-parameter)	Optimisation	Build	Total	Explained Variance	Max Error	Negated Mean Absolute Error	Negated Mean Squared Error*	Negated Median Absolute Error	Pearson correlation
1	RF & ECFP (No optimisation)	Scaffold	-	-	00:00:29	**00:00:29**	0.347	-4.13	-1.1	-2.274	-0.693	0.264
2	RF grid optimisation & ECFP		Stratified	**00:28:06**	**00:00:28**	00:28:34	0.362	-3.907	-1.095	-2.174	-0.764	0.297
3	QSARtuna		Stratified	09:28:26	00:08:30	09:36:56	0.533	-3.601	-0.867	-1.527	-0.709	0.506
4	QSARtuna Min Std.Dev		Stratified	02:44:56	00:01:11	02:46:07	**0.675**	**-3.496**	**-0.698**	**-1.124**	**-0.553**	**0.636**
5	RF & ECFP (No optimisation)	Stratified	-	-	00:00:27	**00:00:27**	0.727	-3.972	-0.819	-1.172	-0.631	0.725
6	RF & ECFP (grid optimisation)		Random	**00:18:25**	**00:00:22**	00:18:47	0.766	-3.964	-0.745	-1.009	-0.585	0.763
7	QSARtuna		Random	04:14:41	00:01:29	04:16:10	**0.907**	**-3.587**	**-0.448**	**-0.398**	**-0.326**	**0.907**
8	QSARtuna Min Std.Dev		Random	02:47:08	00:01:16	02:48:24	**0.907**	**-3.587**	**-0.448**	**-0.398**	**-0.326**	**0.907**

Table 3. Probabilistic modelling for reactivity prediction best considers experimental variability. QSARtuna with probabilistic modelling provides the most optimal setup for modelling the probabilistic likelihood of a successful reaction (considering experimental variability), obtaining the highest external performance.

Run no.	Modelling Approach	Split Methods		Time			External Performance					
		External	Internal (hyper-parameter)	Optimisation	Build	Total	Explained Variance	Max Error	Negated Mean Absolute Error	Negated Mean Squared Error*	Negated Median Absolute Error	Pearson correlation
1	RF & ECFP (No optimisation & no probabilistic modelling)	Stratified	-	-	**00:01:40**	**00:01:40**	0.880	-0.710	-0.078	-0.017	-0.040	0.880
2	RF grid search & ECFP (No probabilistic modelling)		Random	**00:22:49**	00:01:59	00:24:48	0.905	-0.688	-0.064	-0.013	-0.025	0.905
3	QSARtuna (No probabilistic modelling)		Random	01:56:09	00:05:27	02:01:36	0.953	-0.565	-0.042	-0.007	-0.010	0.953
4	QSARtuna (Probabilistic modelling)		Random	01:25:41	00:26:34	01:52:15	**0.967**	**-0.480**	**-0.035**	**-0.005**	**-0.004**	**0.967**

Table 4. VennABERS calibration (scaling) for optimally calibrated DEL enrichment models. QSARtuna with VennABERS scaling provides the most optimal setup during modelling, with the relatively optimal balance between ROC AUC (objective performance) whilst being well calibrated (indicated via negated Brier score loss).

Run no.	Modelling Approach	Split Methods		Time			External Performance							
		External	Internal (hyper-parameter)	Optimisation	Build	Total	AUC PR Calibrated	Average precision (AUC PR)	BEDROC	F1 (macro)	Negated brier score loss	Precision macro	Recall macro	ROC AUC
1	RF & ECFP (No optimisation or scaling	Stratified	Stratified	-	02:22:48	**02:22:48**	0.367	0.021	0.341	0.519	-0.08	0.514	0.647	0.801
2	RF & ECFP (No optimisation & VennABERS scaling)			**-**	02:31:33	02:31:33	0.331	0.017	0.295	0.499	**-0.003**	0.498	0.5	0.802
3	RF grid search & ECFP (No scaling)			1-03:10:26	02:24:55	1-05:35:21	**0.508**	0.051	0.424	0.499	-0.08	0.498	0.5	**0.906**
4	QSARtuna (No scaling)			**1-02:22:20**	**00:11:19**	1-02:33:39	0.486	**0.226**	**0.467**	**0.553**	-0.122	**0.553**	**0.793**	0.874
5	QSARtuna (VennABERS scaling)			3-21:56:44	02:24:55	1-00:21:39	0.499	0.033	0.437	0.499	**-0.003**	0.498	0.5	**0.906**

Data and Software Availability statement

The source code to QSARtuna is made freely available via GitHub at https://github.com/MolecularAI/QSARtuna/tree/master and distributed under an Apache-2.0 license open-source license. The GitHub repository contains all the new methods presented here with clear instructions on setup. Additional instructions on how to install QSARtuna, detailed documentation and further usage examples are available at the GitHub pages located at https://molecularai.github.io/QSARtuna/. The solubility, DEL and reactivity datasets are made available here in the Supporting Information file "Supporting_Information_File_1.zip". Each dataset is provided via the .csv file in each example folder. The accompanying JSON files contain the JSON configurations necessary to reproduce each of the QSARtuna runs for all results in this work.

References

1. Patronov A, Papadopoulos K, Engkvist O: **Has Artificial Intelligence Impacted Drug Discovery?** In: *Artificial Intelligence in Drug Design.* Edited by Heifetz A. New York, NY: Springer US; 2022: 153-176.

2. Thomas M, Boardman A, Garcia-Ortegon M, Yang H, de Graaf C, Bender A: **Applications of Artificial Intelligence in Drug Design: Opportunities and Challenges**. *Methods Mol Biol* 2022, **2390**:1-59.

3. Vijayan RSK, Kihlberg J, Cross JB, Poongavanam V: **Enhancing preclinical drug discovery with artificial intelligence**. *Drug Discov Today* 2022, **27**(4):967-984.

4. De P, Kar S, Ambure P, Roy K: **Prediction reliability of QSAR models: an overview of various validation tools**. *Arch Toxicol* 2022, **96**(5):1279-1295.

5. Kolluri S, Lin J, Liu R, Zhang Y, Zhang W: **Machine Learning and Artificial Intelligence in Pharmaceutical Research and Development: a Review**. *AAPS J* 2022, **24**(1):19.

6. Dara S, Dhamercherla S, Jadav SS, Babu CM, Ahsan MJ: **Machine Learning in Drug Discovery: A Review**. *Artif Intell Rev* 2022, **55**(3):1947-1999.

7. Ren F, Ding X, Zheng M, Korzinkin M, Cai X, Zhu W, Mantsyzov A, Aliper A, Aladinskiy V, Cao Z: **AlphaFold Accelerates Artificial Intelligence Powered Drug Discovery: Efficient Discovery of a Novel CDK20 Small Molecule Inhibitor**. *Chemical Science* 2023.

8. Zheng S, Tan Y, Wang Z, Li C, Zhang Z, Sang X, Chen H, Yang Y: **Accelerated rational PROTAC design via deep learning and molecular simulations**. *Nature Machine Intelligence* 2022, **4**(9):739-748.

9. Karaman B, Sippl W: **Computational Drug Repurposing: Current Trends**. *Curr Med Chem* 2019, **26**(28):5389-5409.

10. Ferreira LT, Borba JVB, Moreira-Filho JT, Rimoldi A, Andrade CH, Costa FTM: **QSAR-Based Virtual Screening of Natural Products Database for Identification of Potent Antimalarial Hits**. *Biomolecules* 2021, **11**(3).

11. Zaki MEA, Al-Hussain SA, Masand VH, Akasapu S, Bajaj SO, El-Sayed NNE, Ghosh A, Lewaa I: **Identification of Anti-SARS-CoV-2 Compounds from Food Using QSAR-Based**

Virtual Screening, Molecular Docking, and Molecular Dynamics Simulation Analysis. *Pharmaceuticals (Basel)* 2021, **14**(4).

12. Yang S, Lee KH, Ryu S: **A comprehensive study on the prediction reliability of graph neural networks for virtual screening**. *arXiv preprint arXiv:200307611* 2020.

13. Williams DP, Lazic SE, Foster AJ, Semenova E, Morgan P: **Predicting Drug-Induced Liver Injury with Bayesian Machine Learning**. *Chem Res Toxicol* 2020, **33**(1):239-248.

14. Toh TS, Dondelinger F, Wang D: **Looking beyond the hype: Applied AI and machine learning in translational medicine**. *EBioMedicine* 2019, **47**:607-615.

15. Xie W, Wang F, Li Y, Lai L, Pei J: **Advances and Challenges in De Novo Drug Design Using Three-Dimensional Deep Generative Models**. *Journal of Chemical Information and Modeling* 2022, **62**(10):2269-2279.

16. Grisoni F, Huisman BJH, Button AL, Moret M, Atz K, Merk D, Schneider G: **Combining generative artificial intelligence and on-chip synthesis for de novo drug design**. *Sci Adv* 2021, **7**(24).

17. Merk D, Friedrich L, Grisoni F, Schneider G: **De Novo Design of Bioactive Small Molecules by Artificial Intelligence**. *Mol Inform* 2018, **37**(1-2).

18. Thompson J, Walters WP, Feng JA, Pabon NA, Xu H, Maser M, Goldman BB, Moustakas D, Schmidt M, York F: **Optimizing active learning for free energy calculations**. *Artificial Intelligence in the Life Sciences* 2022, **2**:100050.

19. Kwon Y, Lee D, Choi YS, Kang S: **Uncertainty-aware prediction of chemical reaction yields with graph neural networks**. *J Cheminform* 2022, **14**(1):2.

20. Chen H, Engkvist O, Wang Y, Olivecrona M, Blaschke T: **The rise of deep learning in drug discovery**. *Drug Discov Today* 2018, **23**(6):1241-1250.

21. Wu Z, Zhu M, Kang Y, Leung EL, Lei T, Shen C, Jiang D, Wang Z, Cao D, Hou T: **Do we need different machine learning algorithms for QSAR modeling? A comprehensive assessment of 16 machine learning algorithms on 14 QSAR data sets**. *Brief Bioinform* 2021, **22**(4).

22. Humbeck L, Koch O: **What Can We Learn from Bioactivity Data? Chemoinformatics Tools and Applications in Chemical Biology Research**. *ACS Chem Biol* 2017, **12**(1):23-35.

23. Zhu H: **Big Data and Artificial Intelligence Modeling for Drug Discovery**. *Annu Rev Pharmacol Toxicol* 2020, **60**:573-589.

24. Chan HCS, Shan H, Dahoun T, Vogel H, Yuan S: **Advancing Drug Discovery via Artificial Intelligence**. *Trends Pharmacol Sci* 2019, **40**(8):592-604.

25. Ghiandoni GM, E. E, J. RD, C. T, C. RP: **Augmenting DMTA using predictive AI modelling at AstraZeneca**. *Submitted*.

26. Coley CW, Eyke NS, Jensen KF: **Autonomous Discovery in the Chemical Sciences Part I: Progress**. *Angew Chem Int Ed Engl* 2020, **59**(51):22858-22893.

27. Dimitrov T, Kreisbeck C, Becker JS, Aspuru-Guzik A, Saikin SK: **Autonomous Molecular Design: Then and Now**. *ACS Appl Mater Interfaces* 2019, **11**(28):24825-24836.

28. Schneider G: **Automating drug discovery**. *Nat Rev Drug Discov* 2018, **17**(2):97-113.

29. He X, Zhao K, Chu X: **AutoML: A survey of the state-of-the-art**. *Knowledge-Based Systems* 2021, **212**:106622.

30. Cox R, Green DV, Luscombe CN, Malcolm N, Pickett SD: **QSAR workbench: automating QSAR modeling to drive compound design**. *J Comput Aided Mol Des* 2013, **27**(4):321-336.

31. Dixon SL, Duan J, Smith E, Von Bargen CD, Sherman W, Repasky MP: **AutoQSAR: an automated machine learning tool for best-practice quantitative structure-activity relationship modeling**. *Future Med Chem* 2016, **8**(15):1825-1839.

32. Carrio P, Lopez O, Sanz F, Pastor M: **eTOXlab, an open source modeling framework for implementing predictive models in production environments**. *J Cheminform* 2015, **7**:8.

33. Sushko I, Novotarskyi S, Korner R, Pandey AK, Rupp M, Teetz W, Brandmaier S, Abdelaziz A, Prokopenko VV, Tanchuk VY *et al*: **Online chemical modeling environment (OCHEM): web platform for data storage, model development and publishing of chemical information**. *J Comput Aided Mol Des* 2011, **25**(6):533-554.

34. Green D, Pickett S, Keefer C, Bizon C, Woody N, Chakravorty S: **Automated predictive modelling: modeller's utopia or fools' gold**. In.; 2008.

35. Salem M, Khormali A, Arshadi AK, Webb J, Yuan J-S: **Transcreen: transfer learning on graph-based anti-cancer virtual screening model**. *Big Data and Cognitive Computing* 2020, **4**(3):16.

36. Kausar S, Falcao AO: **An automated framework for QSAR model building**. *J Cheminform* 2018, **10**(1):1.

37. Minnich AJ, McLoughlin K, Tse M, Deng J, Weber A, Murad N, Madej BD, Ramsundar B, Rush T, Calad-Thomson S *et al*: **AMPL: A Data-Driven Modeling Pipeline for Drug Discovery**. *J Chem Inf Model* 2020, **60**(4):1955-1968.

38. Lanini J, Santarossa G, Sirockin F, Lewis R, Fechner N, Misztela H, Lewis S, Maziarz K, Stanley M, Segler M *et al*: **PREFER: A New Predictive Modeling Framework for Molecular Discovery**. *J Chem Inf Model* 2023, **63**(15):4497-4504.

39. Gao Z, Ji X, Zhao G, Wang H, Zheng H, Ke G, Zhang L: **Uni-QSAR: an Auto-ML Tool for Molecular Property Prediction**. *arXiv preprint arXiv:230412239* 2023.

40. Obrezanova O, Gola JM, Champness EJ, Segall MD: **Automatic QSAR modeling of ADME properties: blood-brain barrier penetration and aqueous solubility**. *J Comput Aided Mol Des* 2008, **22**(6-7):431-440.

41. Casanova-Alvarez O, Morales-Helguera A, Cabrera-Perez MA, Molina-Ruiz R, Molina C: **A Novel Automated Framework for QSAR Modeling of Highly Imbalanced Leishmania High-Throughput Screening Data**. *J Chem Inf Model* 2021, **61**(7):3213-3231.

42. Obrezanova O, Csanyi G, Gola JM, Segall MD: **Gaussian processes: a method for automatic QSAR modeling of ADME properties**. *J Chem Inf Model* 2007, **47**(5):1847-1857.

43. **Molflux** [https://github.com/Exscientia/molflux]

44. Stevenson JM, Mulready PD: **Pipeline Pilot 2.1 By Scitegic, 9665 Chesapeake Drive, Suite 401, San Diego, CA 92123-1365. www. scitegic. com. See Web Site for Pricing Information**. In.: ACS Publications; 2003.

45. Berthold MR, Cebron N, Dill F, Gabriel TR, Kötter T, Meinl T, Ohl P, Thiel K, Wiswedel B: **KNIME-the Konstanz information miner: version 2.0 and beyond**. *AcM SIGKDD explorations Newsletter* 2009, **11**(1):26-31.

46. Demšar J, Curk T, Erjavec A, Gorup Č, Hočevar T, Milutinovič M, Možina M, Polajnar M, Toplak M, Starič A: **Orange: data mining toolbox in Python**. *the Journal of machine Learning research* 2013, **14**(1):2349-2353.

47. Hull D, Wolstencroft K, Stevens R, Goble C, Pocock MR, Li P, Oinn T: **Taverna: a tool for building and running workflows of services**. *Nucleic Acids Res* 2006, **34**(Web Server issue):W729-732.

48. Altintas I, Berkley C, Jaeger E, Jones M, Ludascher B, Mock S: **Kepler: an extensible system for design and execution of scientific workflows**. In: *Proceedings 16th International Conference on Scientific and Statistical Database Management, 2004: 2004*. IEEE: 423-424.

49. Rex DE, Ma JQ, Toga AW: **The LONI pipeline processing environment**. *Neuroimage* 2003, **19**(3):1033-1048.

50. Mervin LH, Johansson S, Semenova E, Giblin KA, Engkvist O: **Uncertainty quantification in drug design**. *Drug Discov Today* 2021, **26**(2):474-489.

51. Fourches D, Muratov E, Tropsha A: **Trust, but verify: on the importance of chemical structure curation in cheminformatics and QSAR modeling research**. *J Chem Inf Model* 2010, **50**(7):1189-1204.

52. Tropsha A: **Best Practices for QSAR Model Development, Validation, and Exploitation**. *Mol Inform* 2010, **29**(6-7):476-488.

53. Ambure P, Cordeiro MNDS: **Importance of data curation in QSAR studies especially while modeling large-size datasets**. *Ecotoxicological QSARs* 2020:97-109.

54. Bernhard R: **Avalon Cheminformatics Toolkit**. In.

55. Koutsoukas A, Paricharak S, Galloway WR, Spring DR, Ijzerman AP, Glen RC, Marcus D, Bender A: **How diverse are diversity assessment methods? A comparative analysis and benchmarking of molecular descriptor space**. *J Chem Inf Model* 2014, **54**(1):230-242.

56. Ghiandoni GM, Caldeweyher E: **Fast calculation of hydrogen-bond strengths and free energy of hydration of small molecules**. *Sci Rep* 2023, **13**(1):4143.

57. Yang K, Swanson K, Jin W, Coley C, Eiden P, Gao H, Guzman-Perez A, Hopper T, Kelley B, Mathea M *et al*: **Analyzing Learned Molecular Representations for Property Prediction**. *J Chem Inf Model* 2019, **59**(8):3370-3388.

58. Reis I, Baron D, Shahaf S: **Probabilistic random forest: A machine learning algorithm for noisy data sets**. *The Astronomical Journal* 2018, **157**(1):16.

59. Mervin LH, Trapotsi M-A, Afzal AM, Barrett IP, Bender A, Engkvist O: **Probabilistic Random Forest improves bioactivity predictions close to the classification threshold by taking into account experimental uncertainty**. *J Cheminform* 2021, **13**(1):1-17.

60. Truchon JF, Bayly CI: **Evaluating virtual screening methods: good and bad metrics for the "early recognition" problem**. *J Chem Inf Model* 2007, **47**(2):488-508.

61. Buendia R, Engkvist O, Carlsson L, Kogej T, Ahlberg E: **Venn-Abers predictors for improved compound iterative screening in drug discovery**. In: *Conformal and Probabilistic Prediction and Applications: 2018*. 201-219.

62. Mervin L, Afzal AM, Engkvist O, Bender A: **A Comparison of Scaling Methods to Obtain Calibrated Probabilities of Activity for Ligand-Target Predictions**. 2020.

63. Taquet V, Blot V, Morzadec T, Lacombe L, Brunel N: **MAPIE: an open-source library for distribution-free uncertainty quantification**. *arXiv preprint arXiv:220712274* 2022.

64. Lundberg SM, Lee S-I: **A unified approach to interpreting model predictions**. *Advances in neural information processing systems* 2017, **30**.

65. Delaney JS: **ESOL: estimating aqueous solubility directly from molecular structure**. *J Chem Inf Comput Sci* 2004, **44**(3):1000-1005.

66. Heravi MM, Kheilkordi Z, Zadsirjan V, Heydari M, Malmir M: **Buchwald-Hartwig reaction: An overview**. *J Organomet Chem* 2018, **861**:17-104.

67. Lim KS, Reidenbach AG, Hua BK, Mason JW, Gerry CJ, Clemons PA, Coley CW: **Machine Learning on DNA-Encoded Library Count Data Using an Uncertainty-Aware Probabilistic Loss Function**. *J Chem Inf Model* 2022, **62**(10):2316-2331.