JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

pubs.acs.org/jcim Article

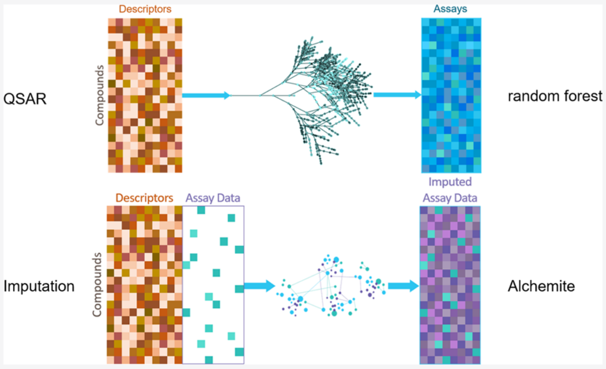
# Practical Applications of Deep Learning To Impute Heterogeneous Drug Discovery Data

Benedict W. J. Irwin,* Julian R. Levell, Thomas M. Whitehead, Matthew D. Segall,* and Gareth J. Conduit

Cite This: J. Chem. Inf. Model. 2020, 60, 2848−2857 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

**ABSTRACT:** Contemporary deep learning approaches still struggle to bring a useful improvement in the field of drug discovery because of the challenges of sparse, noisy, and heterogeneous data that are typically encountered in this context. We use a state-of-the-art deep learning method, Alchemite, to impute data from drug discovery projects, including multitarget biochemical activities, phenotypic activities in cell-based assays, and a variety of absorption, distribution, metabolism, and excretion (ADME) endpoints. The resulting model gives excellent predictions for activity and ADME endpoints, offering an average increase in $R^2$ of 0.22 versus quantitative structure−activity relationship methods. The model accuracy is robust to combining data across uncorrelated endpoints and projects with different chemical spaces, enabling a single model to be trained for all compounds and endpoints. We demonstrate improvements in accuracy on the latest chemistry and data when updating models with new data as an ongoing medicinal chemistry project progresses.



## ■ INTRODUCTION

Machine learning and, more recently, deep learning methods are becoming well-established and have been successful in a variety of scientific and commercial applications.[1,2] However, in the field of drug discovery, training on sparse and often noisy data requires extensive modification to existing algorithms to deliver useful results.[3−5] Recent advances are showing promise using deep learning to predict properties including solubility,[6,7] drug-induced liver injury,[8] target activities,[9,10] and many other endpoints.[11,12] Although each of these models may be individually good, they are tailored to predict only one specific endpoint or a group of closely related endpoints. A great deal of human time is also invested to optimize the hyperparameters[13] and architecture[4] of each model to prevent problems such as overfitting[11,14] and instability with different sizes of the data set.[15] Additionally, the training of deep neural networks can be slow[11,13] and may require significant investment in hardware.[9]

Many modern applications of deep learning in drug discovery are exploring new areas such as compound generation[16−18] and compound synthesis.[19] Meanwhile, realizing the goal of a fully generalized deep learning quantitative structure−activity relationship (QSAR) model that can be applied to general pharmaceutical project data, on both large and small scales, with minimal human intervention, has not received the same degree of attention. There are many pre-deep learning QSAR methods[20] including decision trees and random forests (RFs),[21−23] radial basis functions (RBFs),[24] support vector machines,[25,26] and Gaussian processes (GPs).[27−29] Intermediate neural network methods have a long history, including artificial neural networks[11,30] and general regression neural networks.[31]

So far, despite all this effort, attempts to apply traditional deep learning methods such as deep neural networks[9,10] and deep belief networks[7,32] to prediction of experimental drug discovery endpoints, in a practical way that helps a project progress, have resulted in only a small improvement over traditional QSAR modeling methods[33] such as RFs, with an average increase in $R^2$ coefficient of determination of only 0.043−0.051.[9] Most recently, increases have been seen in the case of graph convolutional networks,[34] which can add average increases of 0.14 to $R^2$ values.[35] Significant improvements over "conventional" machine learning are generally only seen in large data sets or in the case of multitask learning where there are strong correlations between the endpoints.[5] The reason this increase is not larger is likely due to challenges that arise when using pharmaceutical data in conventional approaches.

**Received:** April 28, 2020
**Published:** June 1, 2020

ACS Publications

https://dx.doi.org/10.1021/acs.jcim.0c00443

These are problems arising from sparse, noisy, heterogeneous, and dynamic data, which prohibit deep methods from adding their full value.

In this paper, we describe an application of a deep learning method for data imputation, Alchemite, to an ongoing drug discovery project. While originally developed and proved in the context of materials discovery,[36−39] success has been seen in an example application of this method to a challenging, public domain benchmark data set of kinase activity data.[40,41] In this benchmark, Alchemite was shown to outperform a range of QSAR methods, including a multitask deep neural network trained using TensorFlow[42] and collective matrix factorization.[43] Furthermore, this benchmark demonstrated Alchemite's ability to focus on the most confident predictions with a commensurate improvement in accuracy.

While applications to benchmarking data provide a proof of concept and a robust comparison with other methods, these data sets are not representative of the full range of data encountered in the context of drug discovery projects. In particular, the aforementioned kinase data set comprises only target activity data (expressed as $pIC_{50}$ values). In this work, we extend our previous work to apply the Alchemite algorithm to heterogeneous drug discovery data in a project-based context and explore the temporal evolution of data throughout the project to solve the challenges outlined above. We will briefly discuss the challenges in solving the practical issues encountered when modeling drug discovery data using other methods.

## PREDICTION AND IMPUTATION

There are distinct differences between the problems of predicting an endpoint based on a complete set of inputs, for example, a QSAR regression model, and imputing an endpoint with sparse data, for example, filling in the gaps in data for an experimental endpoint. Figure 1 shows a comparison of these two methods. A QSAR regression model is a function of a full set of complete inputs, that is, molecular descriptors that can be calculated for every compound. The sparsity of drug discovery data prevents assays and experimental values—which may not always be present— all from being used as inputs for this kind of model. The subset of compounds that has all experimental values present is generally quite small, and even if a model were to be trained on these data, new measurements must be made for all inputs in order to make a new prediction. On the contrary, an imputation model can take all existing data (both molecular descriptors and target experimental endpoints) as inputs to the model and fill in the missing values using whatever data that may be present. If the model is correctly designed, it does not suffer the same limitation from missing values as the prediction model. If data are present, they can be used, and if they are missing, they can be predicted.

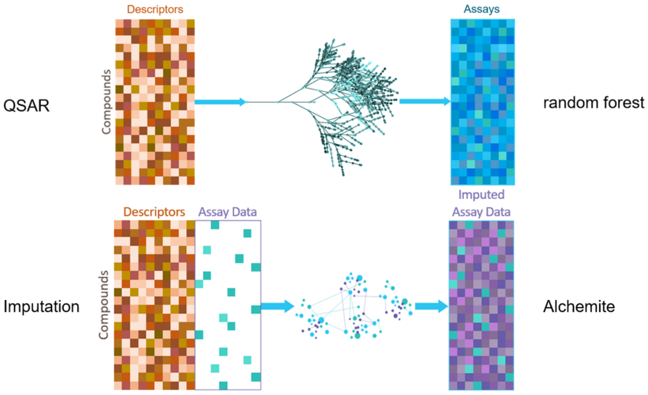


Figure 1. Comparison of a QSAR model (here, a RF) with the deep imputation process (Alchemite), which takes both complete descriptor columns and incomplete assay columns as input. These are used by the deep learning network to fill in the missing values in the assay data columns with an error bar for each data point.

## CHALLENGES OF MODELING DRUG DISCOVERY DATA

For an algorithm or method to get the most out of drug discovery data, it should address a few challenges with which common methods often struggle:

**Missing Data.** If one considers all of the compounds and assays in a large pharmaceutical company's corporate collection, typically, only a small fraction (<1%) of the possible compound-assay endpoint combinations have been measured in practice. Public domain databases are also sparsely populated; for example, the ChEMBL[44] data set is just 0.05% complete. Even in the context of an ongoing project, only a small proportion of compounds will be progressed for more detailed studies, such as measurement of absorption, distribution, metabolism, and excretion (ADME) properties. We have seen above that the design of an imputation model can use sparse experimental columns as inputs to a deep algorithm. One limiting factor for the application of deep learning is the lack of support for this kind of missing data in contemporary methods.[45,46] If inputs are not always present, simple implementations of common algorithms such as neural networks cannot give sensible answers without significant alteration.[46,47] Recent developments, such as the method presented in this study, have taken deep imputation a step further, working comfortably on data sets with <1% of data present.[40]

**Uncertainty and Confidence.** Experimental data are inherently noisy. Even good-quality pharmaceutical data may have up to 1 log unit of variability,[26] and some values could be incorrect because of experimental errors or artifacts.[48] Furthermore, a failure to take uncertainty from noisy predictions into account can lead to wasted time and missed opportunities through misdirection. Conversely, using uncertainties correctly can lead to optimized decisions and a mitigation of risk.[49] A practically useful algorithm should handle explicit uncertainties in the input experimental data and also give a measure of uncertainty in predictions they output.

**Heterogeneous Data.** In the course of drug discovery projects, data sets will be generated using a wide variety of assays which cover target and phenotypic activities, ADME properties, and toxicity and physicochemical properties of compounds of interest. Endpoints may be correlated if they are for the same target under different conditions, related targets, or measurements of the same property in different tissues. More complex assay endpoints, such as phenotypic responses in cell-based assays, may be correlated with multiple, simpler endpoints such as target activities, membrane permeability, solubility, and protein binding. When these mixed results are separated out into separate endpoints, the columns in the data matrix become increasingly sparse, making correlations harder to use without special techniques built for extremely sparse data, for example, by Whitehead et al.[40] Another method that has attempted this is the pQSAR 2.0 method of Martin et al.[41,50] However, previous methods such as pQSAR have focused on combining similar types of endpoints only, for example, all $pIC_{50}$ values. Few, if any, methods have yet attempted to make use of correlations from heterogeneous data with a variety of different scales and distributions, but this is solved automatically in an imputation model, as shown in Figure 1.

**Temporal Evolution of a Project.** Drug design projects evolve with time as the hit- and lead-optimization processes result in an exploration of the chemical space beyond the compounds for which data were previously available. The chemical space of interest may jump as series are discarded or focus during late lead optimization. Compound activity and other properties will improve as the project nears its goal, increasing the range of values. Specific assays may become concentrated and data-rich when an issue is being focused on, while other assays become sparser when an issue is presumed to have been addressed or is no longer relevant. If a model is to be deployed across an entire project data set or even across multiple projects, it should be able to handle a multiscale approach and seamlessly transition from early hit-based screening to lead development, retraining as more data become available.

The majority of machine learning methods are based around interpolation of training values. A successful method should continue to add value after the chemistry has evolved. Many models cannot handle temporally split test data,[51] and this is an important validation for whether a method can add real value to an ongoing project.

In the following Methods section, we will describe the Alchemite method and the data sets to which it was applied in this study. In the Results and Discussion section, we will present the results of applications in the context of an ongoing drug discovery project and the four challenges outlined above. Finally, we will draw some conclusions and discuss potential future work.

## ■ METHODS

The Alchemite method is a deep and iterative multiple imputation method that is a novel adaptation of a neural network in which all inputs are also outputs.[36−40] A detailed description of the underlying algorithm is given by Verpoort et al.[38] and, more recently, by Whitehead et al.[40] Additional information and description of the algorithm are given in the Supporting Information.

The goal is to solve for the weights and biases of a neural network where some outputs of the neural network in the first iteration(s) are potentially used as the inputs of subsequent iterations. This is solved iteratively in the context of a fixed-point equation $f(\boldsymbol{x}) = \boldsymbol{x}$. For the inputs to the first iteration, missing values are replaced by the mean of the available values of the corresponding endpoint. An iterative expectation maximization algorithm is applied[52] to converge the weights of the network.

In the applications described herein, the model will have $N$ inputs and outputs, of which $N = N_d + N_e$, where $N_d$ is the number of molecular descriptors and $N_e$ is the number of experimental assay endpoints. The matrix columns corresponding to the descriptor inputs will be complete because these can be computed in advance for any molecular structure. However, the assay endpoint columns may be sparsely occupied; some, or even most, of the potential experimental data may be missing. The output is a complete matrix of assay endpoints, in which the missing values have been imputed (the process is illustrated in Figure 1).

In this work, 200 networks are trained, with the data rows carrying different weights. This is substantially more than in previous work[36−38] and leads to an ensemble of predictions for each missing value in the data set. The mean of these 200 predictions can be used as the predicted value. The standard deviation of the 200 predictions is used as a measure of uncertainty in that value, giving an error bar for each predicted cell in the imputed matrix.

The hyperparameters of the network were optimized using a fivefold cross validation within the training set data only.[53] The tree-structured Parzen estimator[54] from the python library hyperopt[55] was used. The algorithm uses a combination of Bayesian inference and nonparametric density estimation to optimize the so-called expected improvement.[54,56] Hyperparameter optimization was applied to the number of inputs for each endpoint, the number of iteration layers (convergence loop in Figure S2), and the iterative mixing ratio alongside the hyperparameters of the neural network (Figure S1).

**Molecular Descriptors.** In this work, the number of molecular descriptors was $N_d = 330$. The descriptors used included whole-molecule properties such as molecular weight, lipophilicity, and polar surface area and structural fragments defined by SMARTS.[57] These descriptors were calculated with the Auto-Modeller module of the StarDrop software[58] and have previously been used to train successful QSAR models.[59] However, any set of numerical descriptors can be used as input.

**QSAR Methods for Comparison.** In this work, the Alchemite models will be compared against QSAR models generated with the Auto-Modeller module in StarDrop.[58] For each endpoint, individual models were trained using four common QSAR methods: partial least squares (PLSs), which describe the target property as a linear combination of latent variables;[60] RBF, a simple but effective data-driven method which approximates the target quantity as a linear combination of basis functions centered around the training points;[61] RF, which trains the split criteria for a collection of 100 randomized decision trees to minimize the variance in predictions;[62] and GP with fixed hyperparameters, a Bayesian method that draws models using the posterior distribution of a multivariate Gaussian with a parametric correlation matrix over the training set.[28]

**Data Sets.** Data cleaning was required. Qualified data (i.e., values containing the symbols >, <) were removed from the data set because preliminary investigations demonstrated that simple inclusion of these data with no qualifier symbol produced less-stable models. Some of the raw data were transformed onto scales and distributions more amenable to modeling: $IC_{50}$ values were transformed by taking the negative log of the $IC_{50}$ in molar concentration ($pIC_{50}$); percentage columns underwent a logit transform such that $\text{logit}(x) = \ln(x(1-x)^{-1})$; and the base 10 logarithm was taken of other ADME endpoints that varied over multiple orders of magnitude. Summary tables and series information are provided in the Supporting Information for compounds in all data sets. Distributions of experimental data and molecular characteristics are also provided along with experimental protocols for ADME endpoints.

*Initial Data.* Two real project data sets, project A and project B, were provided by Constellation Pharmaceuticals,[63]

Table 1. Summary of the Initial Data Received for Projects A and B[a]

| | | bioactivity assays | | cell assays | | ADME assays | |
|---|---|---|---|---|---|---|---|
| | number of compounds | number | % filled | number | % filled | number | % filled |
| project A | 1241 | 3 | 45 | 2 | 15 | 8 | 16 |
| project B | 338 | 5 | 55 | 0 | N/A | 8 | 3 |

[a] ADME assays were shared between the data sets. The number of endpoints of each type for each project is shown. The data for each endpoint were sparse, and the percentage filled of data points of each type that had been measured is also shown.

Table 2. Comparison of Alchemite Model Performance against Performance of Single-Endpoint Machine Learning Methods for QSAR on the Independent Test Set for the Initial Data Received from Constellation Pharmaceuticals[a]

| endpoint name (merged data set) | RF ($R^2$) | RBF ($R^2$) | GP ($R^2$) | PLS ($R^2$) | Alchemite ($R^2$) | $R^2$ boost over second-best method |
|---|---|---|---|---|---|---|
| CYP2D6 % inhibition | 0.26 | 0.37 | 0.40 | 0.08 | **0.63** | **+0.23** |
| CYP3A4 % inhibition | 0.26 | 0.24 | 0.21 | 0.15 | **0.3** | **+0.04** |
| HLM $Cl_{int}$ | 0.11 | 0.07 | −0.18 | −0.08 | **0.43** | **+0.32** |
| kinetic solubility | 0.44 | **0.54** | **0.54** | 0.40 | 0.50 | **−0.04** |
| MLM $Cl_{int}$ | 0.37 | 0.51 | 0.49 | 0.31 | **0.54** | **+0.03** |
| PAMPA permeability | 0.24 | 0.18 | **0.28** | 0.19 | 0.21 | **−0.07** |
| ADME PPB % human | 0.60 | 0.56 | 0.58 | 0.48 | **0.72** | **+0.12** |
| ADME PPB % mouse | 0.47 | 0.49 | 0.53 | 0.56 | **0.63** | **+0.07** |
| project A bio. 1 | 0.50 | 0.46 | 0.48 | 0.53 | **0.94** | **+0.41** |
| project A bio. 2 | 0.63 | 0.56 | 0.67 | 0.64 | **0.79** | **+0.12** |
| project A bio. 3 | 0.50 | 0.25 | 0.46 | 0.54 | **0.92** | **+0.38** |
| project A cell 1 | 0.62 | 0.72 | 0.71 | 0.73 | **0.84** | **+0.11** |
| project A cell 2 | −0.29 | −1.2 | −0.48 | −0.27 | **0.57** | **+0.84** |
| project B bio. 1 | 0.44 | 0.43 | 0.38 | 0.30 | **0.65** | **+0.21** |
| project B bio. 2 | 0.46 | 0.52 | 0.40 | 0.28 | **0.82** | **+0.30** |
| project B bio. 3 | 0.53 | 0.45 | 0.44 | 0.37 | **0.82** | **+0.29** |
| project B bio. 4 | 0.46 | 0.44 | 0.44 | 0.30 | **0.62** | **+0.16** |
| project B bio. 5 | 0.56 | 0.57 | 0.53 | 0.47 | **0.71** | **+0.14** |

[a] Bold result is the best method in the row.

including rows equating to anonymized compounds and columns containing sparse experimental data for a heterogeneous mixture of activity, cell, and ADME endpoints. Project A had already finished; no new data would be added. Project B was an ongoing project; the data were provided in batches and models iteratively trained as the project evolved. The targets for the project were unrelated, but some of the types of ADME data were present in both projects. After the modeling work was completed, more details have been published about project A which developed inhibitors for EP300/CBP histone acetyltransferase. Further details can be found in refs.[64−66]

The initial data are summarized in Table 1. The activity endpoints included three target bioactivity columns over two target isoforms, two cell-based assay columns for project A, and five bioactivity columns over three isoforms for project B. The targets of projects A and B were enzymes from unrelated protein families, and there should be no correlation between target activities or cross-target activity for compounds designed for each target. The ADME endpoints included kinetic solubility, permeability measured in a parallel artificial membrane permeability assay (PAMPA), human and mouse plasma protein binding (PPB), human and mouse liver microsome intrinsic clearance (HLM Clint, MLM Clint), and reversible cytochrome P450 (CYP) 2D6 and 3A4 inhibition.

The data were split into an 80% training set and a 20% independent test set. The split was stratified randomly over rows to find the set of training/test rows that had approximately equal data sparsity for all columns simultaneously. This was required because the ADME columns were so sparse that many purely random splits would leave an empty test column.

*Unified versus Individual Models.* To compare the stability of models under different partitioning of the data, the following additional models were trained for comparison with a single, unified model of all data across both projects:

(1) Only activity data from project A
(2) Only the activity data from project B
(3) All of the project A data
(4) All of the ADME data from project A and project B
(5) All of the data from both project A and project B

*Temporal Data.* At the start of the study, the project B data set contained 338 compounds. As the study progressed, another 874 compounds were added to project B, sorted by the date on which they were synthesized and registered in the database, which correlates with the measurement time of assay results. This allowed a temporal split to be made.[51] The new compounds were split into three blocks of ~300 compounds, with block 1 having the oldest and block 3 having the newest compounds in the project. The final block often had higher activities and more relevant ADME data.

Three data splits were generated to allow the construction of three temporal models: Model 1 which used all of the initial data (from Table 1) as a training set, model 2 which used all of the initial data and the first block of temporally split compounds, and model 3 which used all of the initial data and the first two blocks of temporally split compounds. All three models were validated against the final unseen block of compounds so that an independent comparison could be made.

**Model Assessment.** The quality of the models was assessed using the coefficient of determination ($R^2$) in the range $(-\infty,1]$ (N.B. This should not be confused with the Pearson correlation coefficient which is in the range $[-1,1]$). The coefficient of determination is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (f_i - y_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y})^2}$$

where $\bar{y}$ is the mean of the observed data points, $y_i$, and $f_i$ is the model prediction of data point $y_i$. In addition, the root-mean-squared error (RMSE) of the results for each endpoint is considered:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (f_i - y_i)^2}$$

## RESULTS AND DISCUSSION

**Initial Comparison with QSAR Methods.** We compared the multitarget Alchemite method with conventional QSAR models of single endpoints. The QSAR models are based only on molecular descriptors because they cannot use incomplete experimental data as input.

From the results in Table 2, we can see that Alchemite adds significant predictive value over single-endpoint QSAR methods, when comparing the results on the 20% held-out test set for the initial data. On average, for an individual endpoint, Alchemite adds 0.2 to the $R^2$ value of the next leading method (range −0.07 to 0.84) and outperforms the best QSAR model on 16 out of 18 endpoints. Where there is no improvement, the performance is effectively equivalent to the best QSAR result.

Figure 2 shows the best QSAR model from the four types shown in Table 3 (N.B. It is, strictly speaking, unfair to compare the best of the test set results against Alchemite as it would not be known a priori which model was the best). Despite this, Alchemite is still significantly better than this result in almost all endpoints across both activities and ADME varieties. On average, the $R^2$ value for QSAR models is 0.44, and on average, the $R^2$ value for Alchemite models is 0.65.

In particular, we can see that the project A cell 2 (cell proliferation) results cannot be predicted with conventional QSAR methods; a negative $R^2$ indicates a performance that is worse than random (i.e., shuffling the test labels). This is likely because cell activity depends not only on target protein activity but also on the compound reaching the target which will be strongly influenced by physicochemical and ADME properties. However, assay−assay correlations are strong, so when the biochemical assay and ADME results, such as solubility and permeability, can be used as inputs to the model with Alchemite, there is a significant improvement in the ability to predict cell-based activity, even though the majority of data are not available for most compounds.

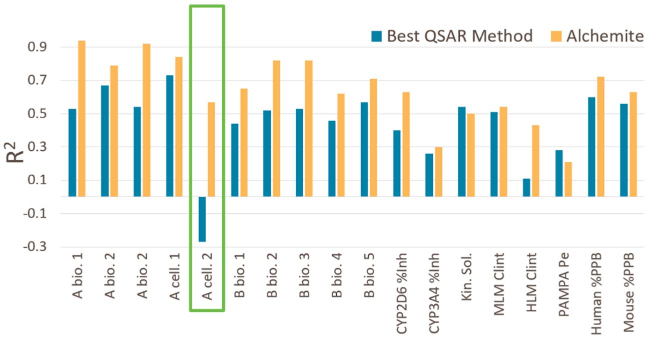


**Figure 2.** Comparison of the results on the independent test set for the best of the four QSAR methods (blue) with an Alchemite model (orange) built with all of the training data from the initial data set.

**Table 3. Summary of Five Model Types To Check How Robust the Algorithm Is to Data Partitioning**[a]

| model | ADME average $R^2$ | activity average $R^2$ | all average $R^2$ |
|---|---|---|---|
| project A activity | N/A | 0.81 | 0.81 |
| project B activity | N/A | 0.73 | 0.73 |
| project A all | 0.52 | 0.82 | 0.63 |
| all ADME data | 0.50 | N/A | 0.50 |
| all data | 0.50 | 0.77 | 0.65 |

[a] Cells with N/A represent combinations which cannot be measured because of the data split definition.

**Comparison of a Single, Unified Model with Individual Models.** Table 3 shows a breakdown of the $R^2$ performances of models constructed with different subsets of the initial data, as described under "Unified versus Individual Models" in the Data Sets section above. There is excellent agreement between models generated with different combinations of project data sets and endpoints, showing that it is not necessary to train individual models for different projects or objectives; the single model of both projects and all data performs equivalently to models built on the individual subsets.

The average coefficient of determination is particularly high on activity models with $R^2$ = 0.81 for the project which has complete lead optimization (project A) and $R^2$ = 0.73 for the new project which is in hit-to-lead (project B). The ADME $R^2$ values are good, considering the data sparsity (only 16% present) and complexity of the endpoints. The summary statistics for the model with all of the data are similar to the average of the two models.

We further drill down into the relative performance in Figure 3, where we compare the models built on individual data sets (i.e., only project A or only project B) versus a model constructed on both data sets simultaneously. We can see for cell and bioactivity assays that the predictive power of both types of model is virtually identical. On average, the quality of the models is also the same for ADME endpoints, although

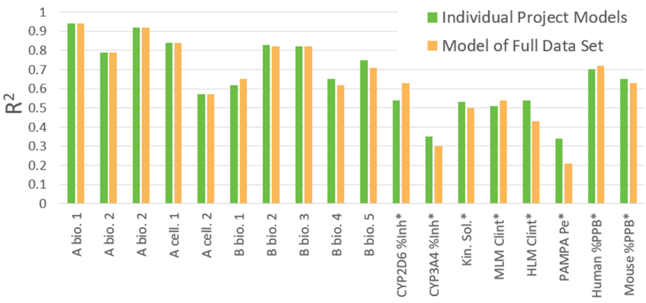


**Figure 3.** Breakdown of independent test $R^2$ values across endpoints in the initial data set. For endpoints marked with *, the individual project model for ADME properties was built and tested on project A only.

there is increased variability. It should be noted that the individual project model for ADME properties was only built and tested on project A because there were insufficient ADME data for project B with which to build and test an individual model, while the model built on all data is built and tested on both projects A and B. Therefore, these models are compared on different test sets.

**Selecting the Most Confident Predictions.** An ensemble of predictions is generated for each missing element of the data matrix, and the distribution of this ensemble can take many shapes. The mean and the standard deviation of this distribution give a unique prediction and error bar for each missing value, where the error bar represents one standard deviation about the mean. In the case where descriptor values or sparse experimental inputs for a new compound extrapolate beyond the training data, the error bar will grow to show that the algorithms have limited knowledge of that region of chemical space. Figure 4 shows an example scatter plot of the predicted versus observed activity, project B bioactivity 2 $pIC_{50}$, for the independent test set of the initial data. We can see the uncertainty estimates as error bars in the y-axis, which intersect with the identity line in almost all cases. The only significant outlier (red point) has correctly been assigned a large uncertainty, indicating that the model has determined this to be a low-confidence prediction.

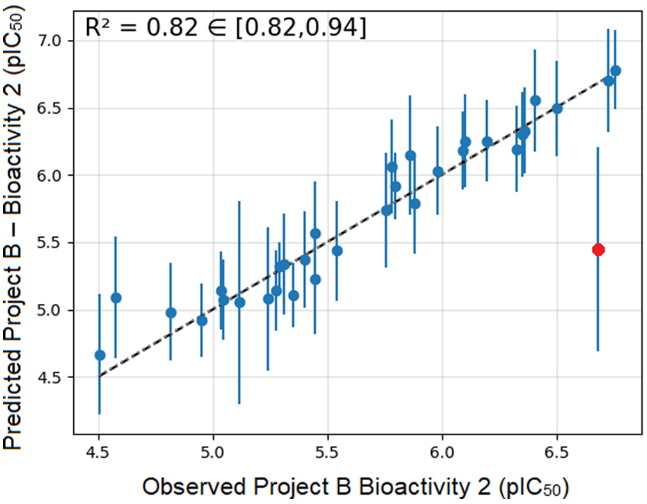


**Figure 4.** Plot of predicted vs observed project B bioactivity 2 values for the independent test set of the initial data predictions. The error bars show one standard deviation in the predicted value, and the dotted line shows the identity line of perfect fit. One clear outlier is highlighted in red, which is correctly assigned the highest uncertainty in prediction.

We can exploit our knowledge of the uncertainties in the predicted values by disregarding those with the highest uncertainty. We would expect the remaining, more confident, values to have a higher accuracy. In Figure 5, we analyze the impact of discarding the predictions in the increasing order of confidence (i.e., the predictions with the largest error bars will be discarded first). The RMSE is plotted on the *y*-axis of the graph, such that low values indicate more accurate predictions. The orange line shows that as the least confident predictions are removed, the RMSE falls sharply, confirming the expected behavior. For this model, we can predict around 80% of results with an RMSE of approximately 0.1 log units.

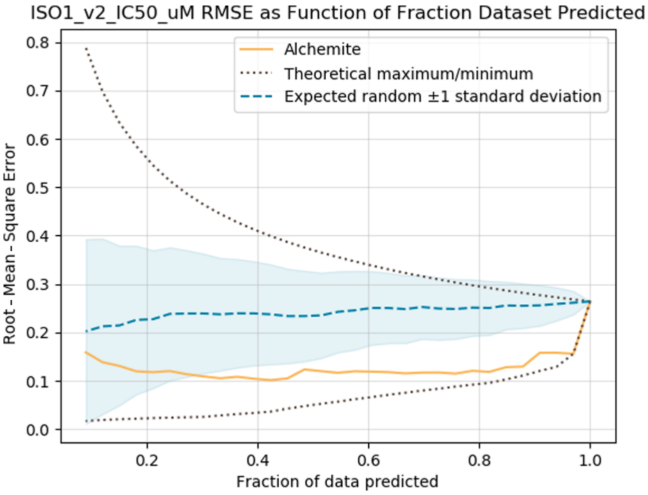


**Figure 5.** Plot of RMSE of predicted test results when predictions with lowest confidence are removed. The orange line shows the performance of the Alchemite model. For comparison, the black dotted lines show the minimum and maximum RMSE achievable as the least- and most-accurate results are removed, that is, the order which minimizes or maximizes the RMSE (N.B. in practice, this order is not known without measuring against the test set). The blue shaded region and dashed line indicate the expected results from randomly removing results. For this endpoint, Alchemite accurately identifies the least confident results, leading to a large improvement in RMSE when only discarding a few of the predictions.

**Temporal Learning and Validation.** We will now focus on the additional compounds provided from Constellation Pharmaceuticals as project B progressed. Results in this section correspond to the models trained on blocks of data as described under "Temporal Data" in the Data Sets section.

Figure 6 shows the average $R^2$ of models 1, 2, and 3 (bold, black line) and the individual endpoint $R^2$ values for the same models (fine, colored lines) for predictions on an independent test set corresponding to the most recent block of compounds and associated data. The average $R^2$ increases linearly, showing

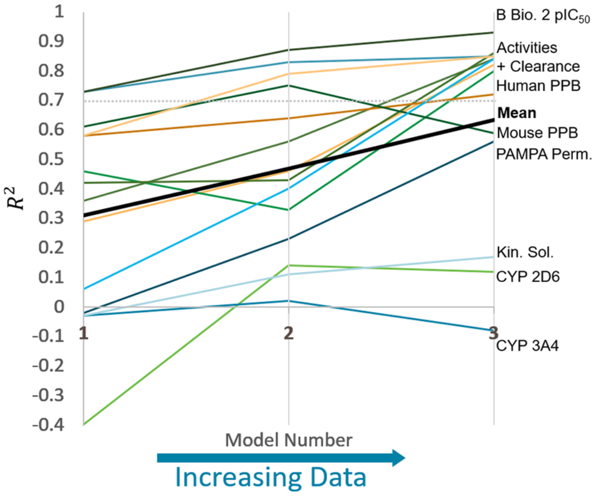


**Figure 6.** Coefficient of determination ($R^2$) of models 1, 2, and 3 on an independent test set corresponding to the most recent block of compounds and associated data (block 3), as more data are added temporally across the project. Bold, black: the average coefficient across all endpoints. Fine, colors: the coefficient for each endpoint with some examples given.

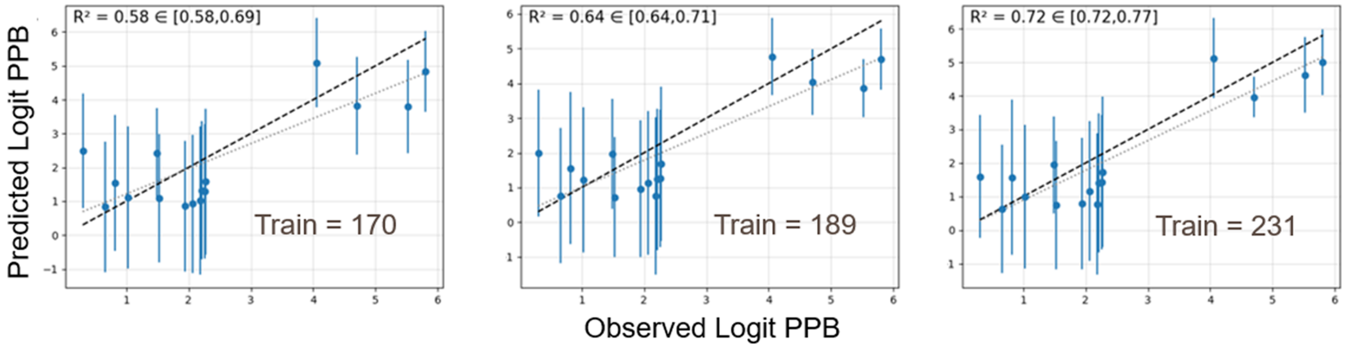


**Figure 7.** Plots of predictions with error bars by models 1, 2, and 3 (left to right) for human protein plasma binding on the independent test set corresponding to the most recent compounds and associated data (block 3). $R^2$ values, training set sizes, and the identity (black) and best fit (gray) lines are shown on each plot. The logit transform was applied to the percent bound data. Cleaning 12 compounds have a logit(PPB) ≤ 2 which corresponds to a PPB < 88%, and four compounds have a logit(PPB) > 4 which corresponds to a PPB > 98%. The highest two compounds have a logit(PPB) ≥ 5.5 which corresponds to a PPB > 99.6%.

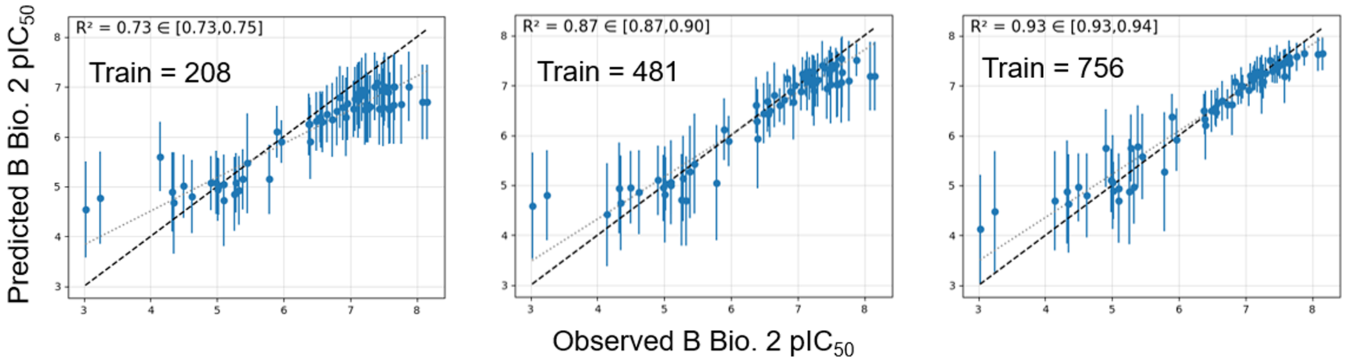


**Figure 8.** Plots of predictions with error bars by models 1, 2, and 3 (left to right) for the project B bioactivity 2 endpoint on the independent test set corresponding to the most recent compounds and associated data (block 3). $R^2$ values, training set sizes, and the identity (black) and best fit (gray) lines are shown on each plot.

constant improvement with additional project data. The breakdown shows a reduction in the variance of model performances and a general tendency for models to pass above the $R^2 = 0.7$ line (a threshold for a very good model). Initially, only activity models are above this line; by the third model, even ADME properties are being predicted with this high level of accuracy. A small number of endpoints do not increase in performance, notably the CYP inhibition endpoints that are some of the sparsest and most complex ADME endpoints in this data set.

To deliver further insights, we now focus on the model predictions for human PPB (Figure 7). There are two classes of compounds in the test set: (1) many moderate binders and (2) four strong binders. Model 1 has limited ability to distinguish between these two classes, with a great deal of overlap in the error bars. With only 19 more training points in model 2, the predictions for the strong binders improve, and the error bars allow the compounds to be more confidently distinguished. By the third model, with 42 further training points, the $R^2$ value has increased significantly, and the model can distinguish all four compounds.

We now focus on the data-rich project B bioactivity 2 endpoint, as shown in Figure 8. There are more training points for this activity column, and models 1, 2, and 3 progressively improve from $R^2 = 0.73$ through to an excellent model with $R^2 = 0.93$. The uncertainties in the predictions for activities reduce greatly by the third model because of the large amount of training data. There were very few examples of a training activity greater than 8; thus, the model begins to extrapolate effectively on the far right-hand side of the plot.

Figure 9 shows the breakdown of the accuracy of model predictions on an independent test set for models generated and tested with all of the data received. For a consistent comparison with the initial model, an 80:20 stratified split was applied, as for the initial data set. The average $R^2$ value from the best of the four QSAR methods for each of the endpoints was now 0.50, which had improved from the previous value of

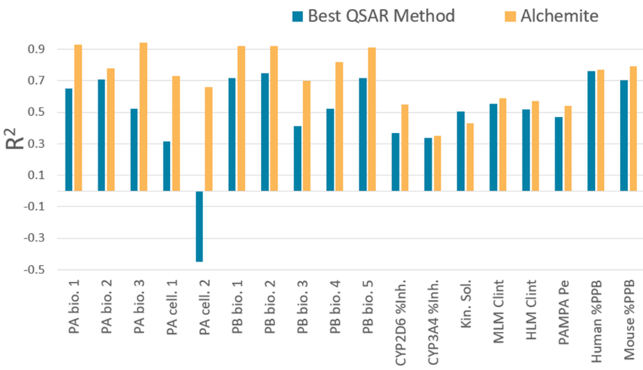


**Figure 9.** Comparison of the results on the independent test set for the best of the four QSAR methods (blue) with an Alchemite model (orange) built with all of the training data using an 80:20 stratified random split on the final data set. This plot can be compared to Figure 2 to inspect the improvement in models with more data.

0.44. This shows that the QSAR methods had used the additional information to improve the model quality. The final Alchemite average $R^2$ value was 0.72, which had improved from 0.65 for the initial set, providing an average improvement of 0.22 over QSAR models on this final data set.

Notably, there are now five bioactivity models at or above the excellent $R^2 = 0.9$ threshold. Alchemite has retained strong models for project A endpoints as more data are added for project B.

## ■ CONCLUSIONS

We have demonstrated a flexible deep learning algorithm that can be used for wide-scale and general-purpose data imputation in the context of an ongoing drug discovery project. It can handle multiple, potentially unrelated inputs and build stable models that outperform conventional QSAR methods by using incomplete experimental data as input to learn transferrable assay—assay correlations. It is also notable that this method still outperforms QSAR in the limit of a smaller data set, representative of a medicinal chemistry project. This contrasts with other deep learning methods which have seen more marginal improvements and generally require much larger data sets.

We considered the application of this method in relation to the challenges of dealing with sparse, noisy, and heterogeneous data in the context of an evolving drug discovery project.

We have seen that an Alchemite model can be trained for data spanning multiple projects and a variety of diverse endpoints, and the quality of predictions was very similar to that of separate models. This shows promise in its ability to capture information at multiple levels of resolution in a single model. The most notable examples where imputation added much greater value over QSAR were for complex endpoints, such as cell-based assays, that likely required a combination of experimental and descriptor inputs to make a meaningful model.

Furthermore, we showed that the confidence estimates in individual predictions enable the most accurate predictions to be identified for individual endpoints. This outcome has now been seen in both homogeneous data[40] and heterogeneous data in this study.

Finally, we illustrated the application of Alchemite to evolving project data, demonstrating that as more data become available, the model can be retrained, resulting in rapidly improving accuracy on the most recent chemistry and experimental data. This enables the application of these models to augment an ongoing project and guide the next most valuable experiment to perform in order to yield the maximum possible benefit.

## ■ ASSOCIATED CONTENT

### SI Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jcim.0c00443.

> Description of the data set in terms of chemical diversity, chemical series, distributions and common chemical properties, and assay values (PDF)

## ■ AUTHOR INFORMATION

### Corresponding Authors

**Benedict W. J. Irwin** — *Optibrium Limited, Cambridge CB25 9PB, U.K.; Cavendish Laboratory, University of Cambridge, Cambridge CB3 0HE, U.K.*; orcid.org/0000-0001-5102-7439; Email: ben@optibrium.com

**Matthew D. Segall** — *Optibrium Limited, Cambridge CB25 9PB, U.K.*; orcid.org/0000-0002-2105-6535; Email: matt@optibrium.com

### Authors

**Julian R. Levell** — *Constellation Pharmaceuticals Inc., Cambridge, Massachusetts 02142, United States*; orcid.org/0000-0002-6171-3819

**Thomas M. Whitehead** — *Intellegens Limited, Cambridge CB4 3AZ, U.K.*

**Gareth J. Conduit** — *Intellegens Limited, Cambridge CB4 3AZ, U.K.; Cavendish Laboratory, University of Cambridge, Cambridge CB3 0HE, U.K.*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.0c00443

### Notes

The authors declare the following competing financial interest(s): BWJI and MDS are employees of Optibrium Ltd. which produce the StarDrop software. TMW and GJC are employees of Intellegens Ltd. JRL is an employee of Constellation Pharmaceuticals Inc.

## ■ REFERENCES

(1) Lecun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436.

(2) Schmidhuber, J. Deep Learning in Neural Networks: An Overview. *Neural Networks* **2015**, *61*, 85−117.

(3) Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T. The rise of deep learning in drug discovery. *Today* **2018**, *23*, 1241−1250.

(4) Ramsundar, B.; Liu, B.; Wu, Z.; Verras, A.; Tudor, M.; Sheridan, R. P.; Pande, V. Is Multitask Deep Learning Practical for Pharma? *J. Chem. Inf. Model.* **2017**, *57*, 2068−2076.

(5) Mayr, A.; Klambauer, G.; Unterthiner, T.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Clevert, D.-A.; Hochreiter, S. Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chem. Sci.* **2018**, *9*, 5441−5451.

(6) Lusci, A.; Pollastri, G.; Baldi, P. Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-like Molecules. *J. Chem. Inf. Model.* **2013**, *53*, 1563−1575.

(7) Li, H.; Yu, L.; Tian, S.; Li, L.; Wang, M.; Lu, X. Deep Learning in Pharmacy: The Prediction of Aqueous Solubility Based on Deep Belief Network. *Autom. Control Comput. Sci.* **2017**, *51*, 97−107.

(8) Xu, Y.; Dai, Z.; Chen, F.; Gao, S.; Pei, J.; Lai, L. Deep Learning for Drug-Induced Liver Injury. *J. Chem. Inf. Model.* **2015**, *55*, 2085−2093.

(9) Ma, J.; Sheridan, R. P.; Liaw, A.; Dahl, G. E.; Svetnik, V. Deep Neural Nets as a Method for Quantitative Structure-Activity Relationships. *J. Chem. Inf. Model.* **2015**, *55*, 263−274.

(10) Xu, Y.; Ma, J.; Liaw, A.; Sheridan, R. P.; Svetnik, V. Demystifying Multitask Deep Neural Networks for Quantitative Structure-Activity Relationships. *J. Chem. Inf. Model.* **2017**, *57*, 2490−2504.

(11) Baskin, I. I.; Winkler, D.; Tetko, I. V. A Renaissance of Neural Networks in Drug Discovery. *Expert Opin. Drug Discovery* **2016**, *11*, 785−795.

(12) Halberstam, N. M.; Baskin, I. I.; Palyulin, V. A.; Zefirov, N. S. Neural networks as a method for elucidating structure-property relationships for organic compounds. *Russ. Chem. Rev.* **2003**, *72*, 629−649.

(13) Hessler, G.; Baringhaus, K.-H. Artificial Intelligence in Drug Design. *Molecules* **2018**, *23*, 2520.

(14) Tetko, I. V.; Livingstone, D. J.; Luik, A. I. Neural Network Studies. 1. Comparison of Overfitting and Overtraining. *J. Chem. Inf. Model.* **1995**, *35*, 826−833.

(15) Varnek, A.; Gaudin, C.; Marcou, G.; Baskin, I.; Pandey, A. K.; Tetko, I. V. Inductive Transfer of Knowledge: Application of Multi-Task Learning and Feature Net Approaches to Model Tissue-Air Partition Coefficients. **2009**, *49*, 133−144. DOI: 10.1021/ci8002914

(16) Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Cent. Sci.* **2018**, *4*, 268−276.

(17) Segler, M. H. S.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ACS Cent. Sci.* **2018**, *4*, 120−131.

(18) De Cao, N.; Kipf, T. MolGAN: An Implicit Generative Model for Small Molecular Graphs. **2018**, arXiv:abs/1805.11973.

(19) Segler, M. H. S.; Preuss, M.; Waller, M. P. Planning Chemical Syntheses with Deep Neural Networks and Symbolic AI. *Nature* **2018**, *555*, 604−610.

(20) Lo, Y.-C.; Rensi, S. E.; Torng, W.; Altman, R. B. Machine Learning in Chemoinformatics and Drug Discovery. *Drug Discovery Today* **2018**, *23*, 1538−1546.

(21) Gao, C.; Cahya, S.; Nicolaou, C. A.; Wang, J.; Watson, I. A.; Cummins, D. J.; Iversen, P. W.; Vieth, M. Selectivity Data: Assessment, Predictions, Concordance, and Implications. *J. Med. Chem.* **2013**, *56*, 6991−7002.

(22) Schürer, S. C.; Muskal, S. M. Kinome-Wide Activity Modeling from Diverse Public High-Quality Data Sets. *J. Chem. Inf. Model.* **2013**, *53*, 27−38.

(23) Christmann-Franck, S.; van Westen, G. J. P.; Papadatos, G.; Beltran Escudie, F.; Roberts, A.; Overington, J. P.; Domine, D. Unprecedently Large-Scale Kinase Inhibitor Set Enabling the Accurate Prediction of Compound-Kinase Activities: A Way toward Selective Promiscuity by Design? *J. Chem. Inf. Model.* **2016**, *56*, 1654−1675.

(24) Zakharov, A. V.; Peach, M. L.; Sitzmann, M.; Nicklaus, M. C. A New Approach to Radial Basis Function Approximation and Its Application to QSAR. *J. Chem. Inf. Model.* **2014**, *54*, 713−719.

(25) Shahlaei, M.; Fassihi, A. QSAR analysis of some 1-(3,3-diphenylpropyl)-piperidinyl amides and ureas as CCR5 inhibitors using genetic algorithm-least square support vector machine. **2013**, *22*, 4384−4400. https://doi.org/10.1007/s00044-012-0430-2. DOI: 10.1007/s00044-012-0430-2

(26) Barrett, S. J.; Langdon, W. B. Advances in the Application of Machine Learning Techniques in Drug Discovery , Design and Development SVM Applications in Pharmaceuticals Research. *Applications of Soft Computing*; Springers, 2004.

(27) Burden, F. R. Quantitative Structure−Activity Relationship Studies Using Gaussian Processes. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 830−835.

(28) Obrezanova, O.; Csányi, G.; Gola, J. M. R.; Segall, M. D. Gaussian Processes: A Method for Automatic QSAR Modeling of ADME Properties. *J. Chem. Inf. Model.* **2007**, *47*, 1847−1857.

(29) Obrezanova, O.; Segall, M. D. Gaussian Processes for Classification: QSAR Modeling of ADMET and Target Activity. *J. Chem. Inf. Model.* **2010**, *50*, 1053−1061.

(30) Myint, K.-Z.; Wang, L.; Tong, Q.; Xie, X.-Q. Molecular Fingerprint-Based Artificial Neural Networks QSAR for Ligand Biological Activity Predictions. *Mol. Pharm.* **2012**, *9*, 2912−2923.

(31) Shahlaei, M.; Sabet, R.; Ziari, M. B.; Moeinifard, B.; Fassihi, A.; Karbakhsh, R. QSAR Study of Anthranilic Acid Sulfonamides as Inhibitors of Methionine Aminopeptidase-2 Using LS-SVM and GRNN Based on Principal Components. *Eur. J. Med. Chem.* **2010**, *45*, 4499−4508.

(32) Ghasemi, F.; Mehridehnavi, A.; Fassihi, A.; Pérez-Sánchez, H. Deep Neural Network in QSAR Studies Using Deep Belief Network. *Appl. Soft Comput.* **2018**, *62*, 251−258.

(33) Dearden, J. C. The History and Development of Quantitative Structure-Activity Relationships (QSARs). *Int. J. Quant. Struct.−Prop. Relat.* **2017**, *2*, 36−46.

(34) Feinberg, E. N.; Sur, D.; Wu, Z.; Husic, B. E.; Mai, H.; Li, Y.; Sun, S.; Yang, J.; Ramsundar, B.; Pande, V. S. PotentialNet for Molecular Property Prediction. *ACS Cent. Sci.* **2018**, *4*, 1520−1530.

(35) Feinberg, E. N.; Sheridan, R.; Joshi, E.; Pande, V. S.; Cheng, A. C. Step Change Improvement in ADMET Prediction with PotentialNet Deep Featurization. **2019**, arXiv:abs/1903.11789. arXiv preprint.

(36) Conduit, B. D.; Jones, N. G.; Stone, H. J.; Conduit, G. J. Design of a Nickel-Base Superalloy Using a Neural Network. *Mater. Des.* **2017**, *131*, 358−365.

(37) Conduit, B. D.; Jones, N. G.; Stone, H. J.; Conduit, G. J. Probabilistic Design of a Molybdenum-Base Alloy Using a Neural Network. *Scr. Mater.* **2018**, *146*, 82−86.

(38) Verpoort, P. C.; MacDonald, P.; Conduit, G. J. Materials Data Validation and Imputation with an Artificial Neural Network. *Comput. Mater. Sci.* **2018**, *147*, 176−185.

(39) Santak, P.; Conduit, G. Predicting Physical Properties of Alkanes with Neural Networks. *Fluid Phase Equilib.* **2019**, *501*, 112259.

(40) Whitehead, T. M.; Irwin, B. W. J.; Hunt, P.; Segall, M. D.; Conduit, G. J. Imputation of Assay Bioactivity Data Using Deep Learning. *J. Chem. Inf. Model.* **2019**, *59*, 1197−1204.

(41) Martin, E. J.; Polyakov, V. R.; Tian, L.; Perez, R. C. Profile-QSAR 2.0: Kinase Virtual Screening Accuracy Comparable to Four-Concentration IC50s for Realistically Novel Compounds. *J. Chem. Inf. Model.* **2017**, *57*, 2077−2088.

(42) Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; Kudlur, M.; Levenberg, J.; Monga, R.; Moore, S.; Murray, D. G.; Steiner, B.; Tucker, P.; Vasudevan, V.; Warden, P.; Wicke, M.; Yu, Y.; Zheng, X.; Brain, G.; Osdi, I.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; Kudlur, M.; Levenberg, J.; Monga, R.; Moore, S.; Murray, D. G.; Steiner, B.; Tucker, P.; Vasudevan, V.; Warden, P.; Wicke, M.; Yu, Y.; Zheng, X. *TensorFlow: A System for Large-Scale Machine Learning*, 2016.

(43) Singh, A. P.; Gordon, G. J. Relational Learning via Collective Matrix Factorization. *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.

(44) Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P. The ChEMBL Bioactivity Database: An Update. *Nucleic Acids Res.* **2014**, *42*, D1083−D1090.

(45) Rubin, D. B. Inference and Missing Data. *Biometrika* **1976**, *63*, 581−592.

(46) Smieja, M.; Struski, Ł.; Tabor, J.; Zieliński, B.; Spurek, P. Processing of Missing Data by Neural Networks. *Adv. Neural Inf. Process. Syst.* **2018**, *2018*, 2719−2729.

(47) Tresp, V.; Ahmad, S.; Neuneier, R. Training Neural Networks with Deficient Data. *Adv. Neural Inf. Process. Syst.* **2002**, *6*, 128.

(48) Yang, J. J.; Ursu, O.; Lipinski, C. A.; Sklar, L. A.; Oprea, T. I.; Bologa, C. G. Badapple: Promiscuity Patterns from Noisy Evidence. *J. Cheminf.* **2016**, *8*, 29.

(49) Segall, M. D.; Champness, E. J. The Challenges of Making Decisions Using Uncertain Data. *J. Comput. Aided Mol. Des.* **2015**, *29*, 809−816.

(50) Martin, E. J.; Polyakov, V. R.; Zhu, X.-W.; Mukherjee, P.; Tian, L.; Liu, X. All-Assay-Max2 PQSAR: Activity Predictions as Accurate as 4-Concentration $IC_{50}$s for 8558 Novartis Assays. *J. Chem. Inf. Model.* **2019**, *59*, 4450−4459.

(51) Sheridan, R. P. Time-Split Cross-Validation as a Method for Estimating the Goodness of Prospective Prediction. *J. Chem. Inf. Model.* **2013**, *53*, 783−790.

(52) Mclachlan, G.; Krishnan, T. *The EM Algorithm and Extensions*, 2nd ed.; Wiley, 2008.

(53) Marron, J. S. A Comparison of Cross-Validation Techniques in Density Estimation. *Ann. Stat.* **1987**, *15*, 152−162.

(54) Bergstra, J.; Bardenet, R.; Bengio, Y.; Kégl, B. Algorithms for Hyper-Parameter Optimization. *Advances in Neural Information Processing Systems*, 2011; pp 2546−2554.

(55) Bergstra, J.; Komer, B.; Eliasmith, C.; Yamins, D.; Cox, D. D. Hyperopt: A Python Library for Model Selection and Hyperparameter Optimization. *Comput. Sci. Discov.* **2015**, *8*, 014008.

(56) Jones, D. R. A Taxonomy of Global Optimization Methods Based on Response Surfaces. **2001**, *21*, 345−383. DOI: 10.1023/a:1012771025575

(57) Daylight SMARTS. https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html (accessed Dec 16, 2019).

(58) StarDrop, https://www.optibrium.com/stardrop/. (accessed Dec 16, 2019).

(59) Hunt, P. A.; Segall, M. D.; Tyzack, J. D. WhichP450: A Multi-Class Categorical Model to Predict the Major Metabolising CYP450 Isoform for a Compound. *J. Comput. Aided Mol. Des.* **2018**, *32*, 537−546.

(60) Wold, S.; Sjostrom, M.; Eriksson, L. PLS Method. In *The Encyclopedia of Computational Chemistry*; Schleyer, P., Allinger, N., Clark, T., Gasteiger, J., Kollman, P. S., Eds.; John Wiley and Sons.: Chichester, U.K., 1999; pp 1−16.

(61) Introduction. In *Radial Basis Functions: Theory and Implementations*; Buhmann, M. D., Ed.; Cambridge Monographs on Applied and Computational Mathematics; Cambridge University Press: Cambridge, 2003; pp 1−10.

(62) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5−32.

(63) Constellation Pharmaceuticals. https://www.constellationpharma.com/ (accessed Dec 16, 2019).

(64) Gardberg, A. S.; Huhn, A. J.; Cummings, R.; Bommi-Reddy, A.; Poy, F.; Setser, J.; Vivat, V.; Brucelle, F.; Wilson, J. Make the Right Measurement: Discovery of an Allosteric Inhibition Site for P300-HAT. *Struct. Dyn.* **2019**, *6*, 054702.

(65) Wilson, J. E.; Huhn, A.; Gardberg, A. S.; Poy, F.; Brucelle, F.; Vivat, V.; Patel, G.; Patel, C.; Cummings, R.; Sims, R.; Levell, J.; Audia, J. E.; Bommi-Reddy, A.; Cantone, N. Early Drug Discovery Efforts Towards the Identification of EP300/CBP Histone Acetyltransferase (HAT) Inhibitors. *ChemMedChem* **2020**, *15*, 955.

(66) Wilson, J. E.; Patel, G.; Patel, C.; Brucelle, F.; Huhn, A.; Gardberg, A. S.; Poy, F.; Cantone, N.; Bommi-Reddy, A.; Sims, R. J.; Cummings, R. T.; Levell, J. R. Discovery of CPI-1612: A Potent, Selective, and Orally Bioavailable EP300/CBP Histone Acetyltransferase Inhibitor. *ACS Med. Chem. Lett.* **2020**, *11*, 1324−1329.