RegioML: Predicting the regioselectivity of electrophilic aromatic substitution reactions using machine learning

Nicolai Ree,[†] Andreas H. Göller,[*,‡] and Jan H. Jensen[*,†]

[†]*Department of Chemistry, University of Copenhagen, Universitetsparken 5, 2100 Copenhagen Ø, Denmark*

[‡]*Bayer AG, Pharmaceuticals, R&D, Computational Molecular Design, 42096 Wuppertal, Germany*

E-mail: andreas.goeller@bayer.com; jhjensen@chem.ku.dk

Abstract

We present RegioML, an atom-based machine learning model for predicting the regioselectivities of electrophilic aromatic substitution reactions. The model relies on CM5 atomic charges computed using semiempirical tight binding (GFN1-xTB) combined with the ensemble decision tree variant light gradient boosting machine (LightGBM). The model is trained and tested on 21,201 bromination reactions with 101K reaction centers, which is split into a training, test, and out-of-sample datasets with 58K, 15K, and 27K reaction centers, respectively. The accuracy is 93% for the test set and 90% for the out-of-sample set, while the precision (the percentage of positive predictions that are correct) is 88% and 80%, respectively. The test-set performance is very similar to the graph-based WLN method developed by Struble et al. (*React. Chem. Eng.* 2020, 5, 896) though the comparison is complicated by the possibility that some of the test and out-of-sample molecules are used to train WLN. RegioML out-performs our physics-based RegioSQM20 method (*J. Cheminform.* 2021, 13:10) where the precision is only 75%. Even for the out-of-sample dataset, RegioML slightly outperforms RegioSQM20. The good performance of RegioML and WLN is in large part due to the large datasets available for this type of reaction. However, for reactions where there is little experimental data, physics-based approaches like RegioSQM20 can

be used to generate synthetic data for model training. We demonstrate this by showing that the performance of RegioSQM20 can be reproduced by a ML-model trained on RegioSQM20-generated data.

Introduction

Many useful reactions are underutilised in synthetic organic chemistry because of an inability to predict the regioselectivity of the reaction[1] and there is thus an increasing interest in developing regioselectivity prediction methods for such reactions. Recent examples include nucleophilic[2,3] and electrophilic aromatic substitution reactions,[4–9] Diels-Alder reactions,[10,11] Heck reactions,[12] radical C-H functionalisation of heterocycles,[13] and reactions such as as alkylations, Michael additions, and aldol condensations that proceed through proton abstraction.[14] These methods have been based on either quantum chemical (QM) calculations,[2,5,6] machine learning (ML) trained on experimental data,[8,10–12] or a combination of the two where QM has either provided descriptors for the ML model[3,9] or used to augment the training data.[13,14] However, these approaches have rarely been compared on the same dataset.[9]

In this paper we present an ML model (RegioML) that predicts the regioselectivity of electrophilic aromatic substitution (EAS) reactions using QM charges. We compare the performance of RegioML to RegioSQM20[6] - a QM-based predictor for EAS regioselectivity - for the same dataset and discuss how QM-based predictors can be used to augment sparse experimental datasets. We focus in particular on the precision and recall of these methods for in- and out-of-sample datasets.

Methods

Dataset preparation

The reaction data are extracted from Reaxys using a set of queries (see supporting information), which results in a total of 30,368 bromination reactions. A thorough dataset curation is then performed to obtain a set of unique SMILES (simplified molecular input line entry system) of the reactants and their corresponding site of bromination, which reduces the total number of reactions to 21,896. For example, a reaction is discarded if there is not an exact one-to-one mapping between the heavy atoms of the reactant and the product excluding the reacting bromine(s), or if a reacting bromine forms a bond to something other than a cyclic sp^2 hybridized carbon atom (accounting for 5,314 reactions). Furthermore, reactions with unique reactions IDs in Reaxys but identical reactants are merged (accounting for 3,158 reactions).

Quantum chemical calculations

Recently, we published the RegioSQM20 method,[6] which predicts the regioselectivities of EAS reactions from semiempirical calculations of proton affinities. The single-tautomer

version of this method is applied to the 21,896 reactions to get proton affinities for all of the unique reaction sites. An extension of this method is also applied in which the RegioSQM20 calculations are followed by single point density functional theory (DFT) calculations in methanol (MeOH, dielectric = 33.6) using the PBEh-3c composite electronic structure method[15] and the conductor-like polarizable continuum model[16,17] (C-PCM) as implemented in the quantum chemistry program `ORCA` version 4.2.[18]

A few of the calculations resulted in extreme proton affinities corresponding to outliers in the dataset that complicated the development of regression models. However, the calculated proton affinities for both the original and extended RegioSQM20 calculations follow a Gaussian distribution (see supporting information), which enables the use of Chauvenet's criterion to remove these outliers in the dataset. In the Chauvenet's criterion the probability of the farthest point is calculated under the assumption of a Gaussian distribution. If this point is below some predefined value then the point is removed, and the procedure is repeated until no more points are removed. In our dataset molecules are removed if at least one atom in the molecule has a proton affinity corresponding to a probability below 1 %.

Atomic descriptors

We investigate seven different atomic descriptors as input to the ML models (details on the descriptors are given in Table S1 in the supporting information). The atomic descriptors are developed by Finkelmann *et al.*[19,20] and have been successfully applied to the prediction of site of metabolism,[20,21] hydrogen bond donor and acceptor strengths,[22,23] and Ames mutagenicity of primary aromatic amines.[24] Almost all of the descriptors depend on charge model 5 (CM5) atomic charges,[25] which are obtained from a single point calculation using GFN1-xTB as implemented in the open source semiempirical software package `xtb` version 6.4.0.[26] This particular charge scheme has been shown to be largely conformation-independent and to correctly reflect changes in the chemical environment i.e. substituents effects.[19] Hence, only a single conformer is generated for each molecule using ETKDG versions 3[27] with useSmallRingTorsions=True as implemented in RDKit version 2020.09.4.[28] This is the key to using quantum chemical derived descriptors as the computational cost is kept at a minimum (details about computational timings are provided in Results and discussion). The atomic descriptors are generated fully automatically from a SMILES representation of a given molecule.

From the screening of the seven atomic descriptors, we find that a charge shell descriptor with 5 shells and values sorted according to the Cahn-Ingold-Prelog (CIP) rules is particularly good for predicting the regioselectivty of bromination reactions (see Table S2 and Figure S4 in the supporting information). An illustration of this 485-dimensional descriptor can be seen in Figure 1.

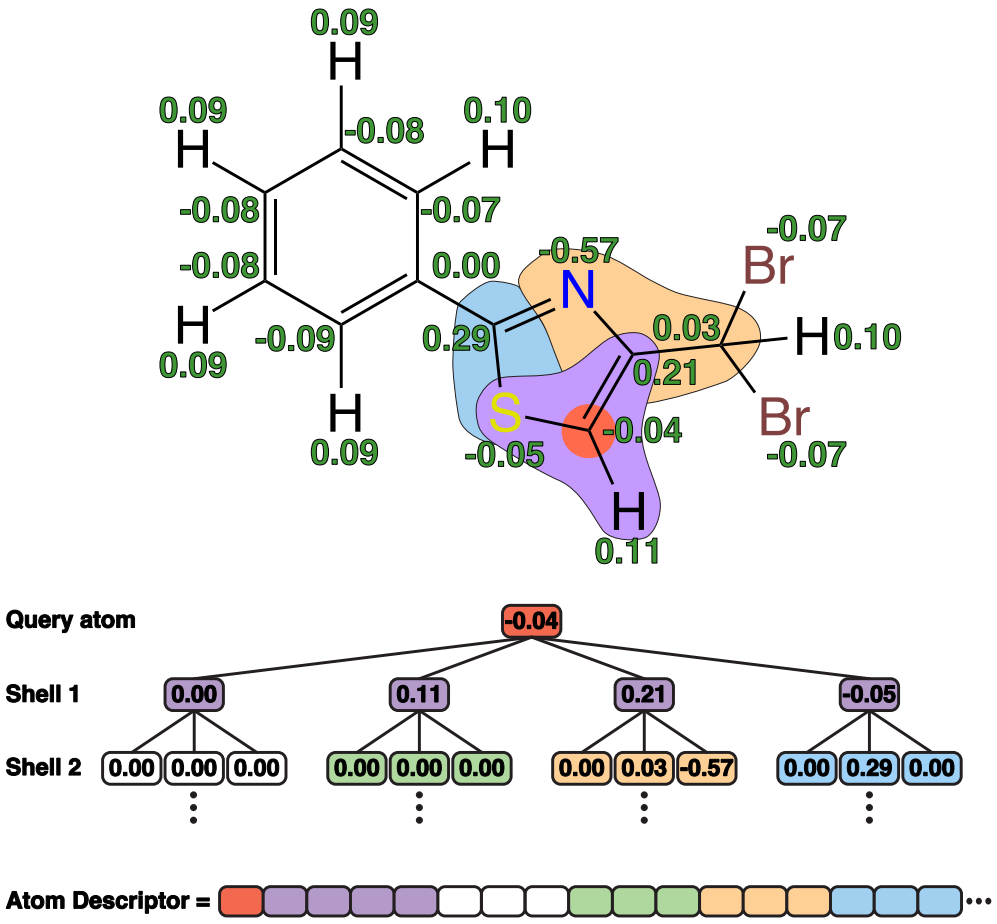

Figure 1: An illustration of the charge shell descriptor with values sorted according to the Cahn-Ingold-Prelog (CIP) rules. The green values correspond to the calculated CM5 charges using GFN1-xTB.

Dataset splitting

We utilize an unsupervised learning procedure similar to the one found in the MOLAN workflow by Sivaraman and Jackson *et al.*,[29] which resembles the ButinaSplitter from DeepChem. The procedure is as follows: SMILES representations of each molecule are converted into extended connectivity (Morgan) fingerprints[30] with a radius of 2 and 1,024 bits (ECFP4). The ECFP4 fingerprints are then used to construct a Tanimoto similarity matrix, which enables a clustering of the molecules using the Butina clustering algorithm[31] with a radial cutoff of 0.6 as implemented in RDKit.[28] Clusters with at least 7 molecules are included in the training/test set and otherwise in the out-of-sample set to explore how well the trained machine learning models generalize. For some molecules either the descriptor or RegioSQM20 calculations fail, or the molecules are excluded due to the Chauvenet's criterion, which left us with 21,201 reactions corresponding to 100,588 unique reaction sites. Thus, applying the above procedure results in a training/test set and an out-of-sample set of 15,246 and 5,955 molecules, which corresponds to 73,123 and 27,465 unique reaction sites, respectively.

Uniform stratified and random splits are then used to obtain a 80:20 ratio between the training and test sets resulting in 12,196 and 3,050 molecules corresponding to 58,384 and 14,739 unique reaction sites in each set, respectively. For the uniform stratified split, each of the individual clusters are randomly split and hereafter combined to ensure that both the training and test sets have similar representations of the underlying data distribution. On the other hand, the random split is indeed completely random with respect to all of the molecules obeying the cluster size cutoff.

As the strategy of this work is to learn and predict with atoms instead of molecules, all of the atomic descriptors for atoms in molecules belonging to the training, test, and out-of-sample sets are collected into different input sets and the corresponding proton affinities or classifications into different output sets.

Machine learning models

In order to learn and predict the regioselectivity of EAS reactions, we explore various regression and classification models with respect to both the experimental and calculated data described above. Initially, a screening of 17 regression models and 13 classification models using PyCaret version 2.3.2[32] is conducted (details can be found in the supporting information). This allows us to quickly find promising machine learning methods, which are then thoroughly examined in terms of finding optimal hyperparameters. The hyperparameter optimizations are carried out using a tree-structured Parzen estimator (TPE) as implemented in Optuna version 2.5.0.[33] All training and evaluation are done using either a normal or a stratified 5-fold cross-validation of the randomly shuffled training set in the case of the regression and classification models, respectively, and only the models with the best validation performance are saved for testing. As we show in Table S3 the best performance for both regression and classification is the ensemble decision tree variant called light gradient boosting machine (LightGBM) version 3.1.1[34] using the sorted-shell atomic descriptors with a shell radius of 5. We refer to this method as simply "LightGBM" hereafter.

Furthermore, we examine the imbalance in the dataset using a "Null model", where all sites are predicted to be non-reactive. And we employ a 1-nearest neighbors (1-NN) classifier as a baseline model using the brute-force search algorithm and the Jaccard metric as implemented in scikit-learn,[35] which corresponds to a perfect memorization of the training set.[36]

Table 1: Comparing different methods for predicting the reactivity of the 14,739 unique reaction sites in the test set and the 27,465 unique reaction sites in the out-of-sample set. The reported metrics are accuracy (ACC), Matthew's correlation coefficient (MCC), precision (PPV or positive predictive value), recall (TPR or true positive rate), specificity (TNR or true negative rate), and negative predictive value (NPV).

Method	**Test set**						**Out-of-sample set**					
	ACC	MCC	PPV	TPR	TNR	NPV	ACC	MCC	PPV	TPR	TNR	NPV
Null model	0.76	0.00	0.00	0.00	1.00	0.76	0.76	0.00	0.00	0.00	1.00	0.76
1-NN	0.86	0.62	0.71	0.72	0.91	0.91	0.81	0.49	0.59	0.64	0.86	0.88
RegioML	0.93	0.81	0.88	0.83	0.96	0.95	0.90	0.72	0.80	0.76	0.94	0.93
WLN (not retrained)	0.93	0.80	0.92	0.78	0.98	0.93	0.92	0.78	0.88	0.78	0.96	0.93
RegioML (all reactions)	0.99	0.98	0.99	0.99	1.00	1.00	0.99	0.98	0.99	0.98	1.00	0.99
RegioSQM20	0.89	0.70	0.75	0.81	0.91	0.94	0.88	0.69	0.73	0.80	0.91	0.94
LightGBM (532 reactions)	0.84	0.52	0.84	0.43	0.97	0.84	0.84	0.51	0.78	0.46	0.96	0.85
LightGBM RegioSQM20	0.88	0.69	0.75	0.78	0.92	0.93	0.86	0.62	0.69	0.74	0.89	0.92
RegioSQM20 PBEh-3c	0.90	0.73	0.81	0.78	0.94	0.93	0.90	0.72	0.79	0.79	0.93	0.93
LightGBM RegioSQM20 PBEh-3c	0.90	0.72	0.81	0.76	0.94	0.93	0.87	0.65	0.74	0.73	0.92	0.91
LightGBM RegioSQM20 regression	0.87	0.65	0.74	0.73	0.92	0.92	0.86	0.61	0.70	0.71	0.90	0.91

Results and discussion

The results we present here only involve the random splitting of the training and test set as similar performances are observed for both the stratified and random splits as seen in Table S4 in the supporting information. Unless otherwise noted all machine learning models are classifiers that output a value between 0 and 1 for each atom, where a value greater than 0.5 indicates that an atom should be reactive.

Data-driven machine learning classifiers

In this section, we train and evaluate machine learning classifiers on experimental data collected from Reaxys consisting of 58,384, 14,739, and 27,465 unique reaction sites in the training, test, and out-of-sample sets, respectively. The experimental data often contain just single or a few reported reactive sites among all reaction sites in the reactant, i.e. there are significantly more negatives (N) than positives (P) in the dataset. Consequently the accuracy (the proportion of correct predictions, ACC $=\frac{\mathrm{TP+TN}}{\mathrm{P+N}}$) can be a misleading metric. For example, a "Null model", where all sites are predicted to be non-reactive achieves a

respectable accuracy of 76 % (Table 1) for both the test and out-of-sample sets, but this just reflects the fact that 76% of the sites in both the datasets are unreactive. The Matthews correlation coefficient[37] (MCC) is a more robust metric to assess the model performance, since it also considers false positives (FP) and false negatives (FN) in addition to true positives (TP), true negatives (TN).

$$\mathrm{MCC} = \sqrt{\mathrm{PPV} \times \mathrm{TPR} \times \mathrm{TNR} \times \mathrm{NPV}} - \sqrt{(1-\mathrm{PPV}) \times (1-\mathrm{TPR}) \times (1-\mathrm{TNR}) \times (1-\mathrm{NPV})} \quad (1)$$

where $\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$, $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$, $\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{N}} = \frac{\mathrm{TN}}{\mathrm{TN+FP}}$, and $\mathrm{NPV} = \frac{\mathrm{TN}}{\mathrm{TN+FN}}$ are also known as precision, recall, specificity, and negative predictive value, respectively.
The MCC values for both the test and out-of-sample sets are zero, which clearly shows that the Null model lacks any real predictive power.

As a baseline model, we trained a 1-nearest neighbors (1-NN) classifier corresponding to a perfect memorization of the training set.[36] The data shows that an impressive-looking 86% accuracy can be achieved for the test set by simple memorisation of the training set. In contrast, the MCC value is only 0.62 for the test set and considerably lower (0.49) for the out-of-sample set. These values primarily reflect a low precision where only 71% and 59% of the positive predictions are actually correct.

Our best machine learning model (LightGBM) achieves a considerably better precision of 88% and 80%, respectively. Note that while there is only a 3% drop in accuracy on going from the test set to the out-of-sample set, there is an 8% drop in the precision (and a concomitant drop in MCC). Hereafter, we refer to this method (i.e. LightGBM trained on experimental data) as RegioML.

The test set MCC of RegioML is virtually identical to the Weisfeiler-Lehman neural network (WLN) architecture specifically trained to predict the regioselectivity of EAS reactions by Struble et al.[8]. While the precision is 4% higher for WLN, the recall (the fraction of positives that are predicted correctly) is 5% lower leading to an nearly identical overall performance. WLN performs better on the out-of-sample set, with an MCC value that is nearly identical to that off the test set. However, it should be noted that many of the molecules in these two sets are likely included in the set used to train the WLN method, which is likely to inflate the WLN MCC values. For example, we are able to achieve MCC values of 0.98 on both the test and out-of-sample sets by training the LightGBM model on the entire collection of data using 10-fold cross-validation (the MCC value is for the best performing model).

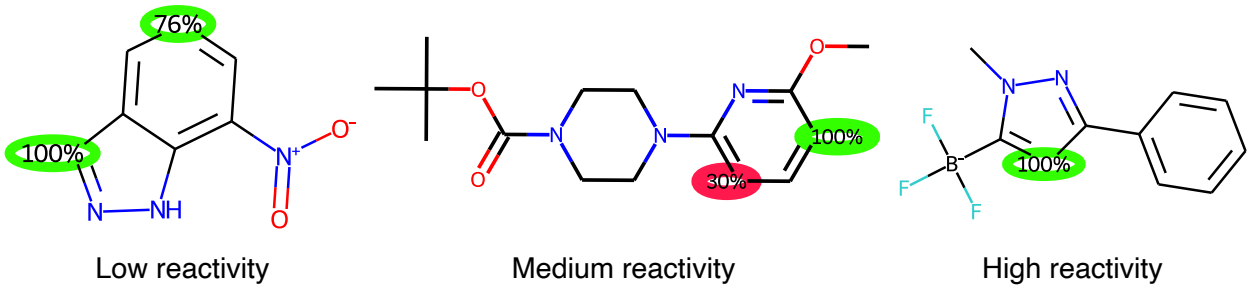

Figure 2: Examples of the output of RegioML. The scores are obtained by a LightGBM classification model, where values above 50% indicates that an atom should be reactive (green circles). However, atoms with scores above 5% are also highlighted (red circles). The predicted low, medium, or high reactivity are based on the highest proton affinity within the molecule obtained by a LightGBM regression model.

Comparison to RegioSQM20

RegioSQM20 predicts the regioselectivity of EASs by finding the reaction center with the highest proton affinity. For computational efficiency the proton affinities are computed using the semiempirical tight binding method GFN1-xTB and a continuum solvent model of methanol. The centers with proton affinities within 1 kcal/mol of the maximum are considered reactive. The method thus has only a handful of hyperparameters (choice of computational method, solvent, energy cutoff, conformational search method) and these are chosen based on a dataset of 532 experimental measurements, some of which are included in the current training set.

For the test set, the recall of RegioSQM20 is similar to RegioML (81% vs 83%) but the precision is significantly worse (75% vs 88%). For the out-of-sample set, the recall is somewhat better for RegioSQM20 (80% vs 76%) but the precision is still worse (73% vs 80%), leading to a slightly smaller MCC value of 0.69 compared to 0.72 for RegioML. In contrast to RegioML, the overall performance of RegioSQM20 is very similar for the test and out-of-sample set, as one would expect from a more physics based method. However, RegioSQM20 does not offer an advantage over RegioML for the out-of-sample dataset, while being computationally much more demanding (see below).

The main advantage of the RegioSQM20 approach is that it may offer an accuracy similar to RegioML based on a much smaller training set. Indeed a LightGBM model trained on the same 532 reactions used to develop RegioSQM20 results in MCC values around 0.5 for both the test and out-of-sample set. While the precision is quite good for this model, the recall is less than 50% due to a large proportion of false negatives. Thus, in cases where little experimental data is available physics-based methods like RegioSQM20 is likely to outperform ML based methods, even if the latter rely on quantum descriptors such as atomic charges.

The computational expense of the physics-based methods can be mitigated by using them to generate synthetic data for the machine learning model. Indeed, a LightGBM classifier

trained on RegioSQM20 predictions for the large training set of 58K reaction centers offers the same performance as RegioSQM20 for the test set. Of course, the performance is worse for the out-of-sample set just like for RegioML, but the training dataset can now easily be expanded to ensure a better coverage of chemical space. Furthermore, since RegioSQM20 is not used to offer real-time predictions to a user, more accurate and computationally expensive methods can be explored. For example, the precision of RegioSQM20 can be increased by 6% by using PBEh-3c single point calculations to compute the proton affinities - an increase that is reflected in the corresponding ML model. The overall performance of RegioSQM20 PBEh-3c is now identical to the RegioML for the out-of-sample dataset, with MCC values of 0.72.

We also explore whether it is better to predict the proton affinities using regression and use them to identify reactive centers, rather the classification approach. Although the LightGBM RegioSQM20 regression model is able to achieve a mean absolute error (MAE) of 2.00 kcal/mol on the test set, the accuracy is not good enough to distinguish between reactive and non-reactive sites compared to the LightGBM RegioSQM20 model, as evidence by the low recall values of 71%-73%. However, the LightGBM RegioSQM20 regression model can be used to predict low, medium, or high reactivity as we showed in the RegioSQM20 paper.[6] In fact, by combining the classification model and the regression model one gets both regioselectivity predictions and a qualitative prediction of the reactivity of a molecule with almost no additional cost as the atomic descriptors only have to be calculated once. Examples of the output of RegioML can be seen in Fig. 2.

Table 2: Timings of the RegioSQM20 method, the RegioML model, and the WLN architecture for predicting the regioselectivity of the 3,050 molecules in the test set given a SMILES representation as input. For the RegioML model and the WLN architecture, the timings include descriptor creation and model prediction for all reaction sites in the given reactant.

Method	Median CPU time (s)	Mean CPU time (s)
RegioSQM20[a]	48	130
RegioML[b]	0.46	0.69
WLN (not retrained)[b]	0.03	0.03

[a] 4 cores/molecule (Intel(R) Xeon(R) CPU E5-2643 v3 @ 3.40GHz)
[b] 1 core/molecule (Intel(R) Xeon(R) CPU X5550 @ 2.67GHz)

Timings

In Table 2 we compare the timings of the RegioSQM20 method, the RegioML model, and the WLN architecture by Struble *et al.*[8] for the 3,050 molecules in the test set. We report the median CPU time and the mean CPU time for predicting the regioselectivity of a molecule given a SMILES representation as input. For the RegioML model and the WLN architecture,

the timings cover descriptor creation as well as model prediction for all reaction sites in the given reactant.

The results show that the median CPU time requirements of the RegioSQM20 method is 48 s per molecule on four Intel(R) Xeon(R) CPU E5-2643 v3 @ 3.40GHz cores. The RegioML model is almost 100 times faster on just a single Intel(R) Xeon(R) CPU X5550 @ 2.67GHz core with a median CPU time of less than half a second per molecule. The WLN architecture is able to achieve a mean CPU time of just 0.03 s per molecule on the single Intel(R) Xeon(R) CPU X5550 @ 2.67GHz core. The main reason for the slower performance of RegioML is the GFN1-xTB single point calculation needed to compute the atomic charges.

Conclusions and outlook

We present RegioML, an atom-based machine learning model for predicting the regioselectivities of electrophilic aromatic substitution (EAS) reactions. The model relies on ultra fast quantum chemical descriptor calculations combined with an ensemble decision tree variant called light gradient boosting machine (LightGBM). The atomic descriptors are based on CM5 atomic charges obtained from a single conformer embedding with RDKit [28] and a single point calculation using the open source semiempirical tight binding method GFN1-xTB.[26] The model is trained and tested on 21,201 bromination EAS reactions and 101K reaction centers, which is split into a training, test, and out-of-sample datasets with 58K, 15K, and 27K reaction centers, respectively. The accuracy is 93% and 90% for the test and out-of-sample set, respectively, but this is not a good measure of performance due to the preponderance of non-reactive sites. For example, the precision (the percentage of positive predictions that are correct) is 88% for the test set, but only 80% for the out-of-sample set. The final RegioML model released to users is trained on the entire data set and we expect similar performance for molecules in-sample and out-of-sample for this large dataset. The test-set performance is very similar to the graph-based WLN method developed by Struble et al.[8] though the comparison is complicated by the possibility that some of the test and out-of-sample molecules are used to train WLN. RegioML out-performs our physics-based RegioSQM20 method[6] where the precision is only 75%. Even for the out-of-sample dataset, RegioML slightly outperforms RegioSQM20.

The good performance of RegioML and WLN is in large part due to the large datasets available for this type of reaction. For example, if we retrain the RegioML model on the same 532 reaction we used to develop RegioSQM20, the performance is much worse due to a large increase in the false negative rate leading to a recall (the percentage of positives that are predicted correctly) below 50% compared to 80% for RegioSQM20. Thus, one use of physics-based approaches like RegioSQM20 is to generate synthetic data for ML model for reactions where there is little experimental data. We demonstrate this by showing that the performance of RegioSQM20 can be reproduced by a ML model trained on RegioSQM20-generated data.

Acknowledgement

Not applicable.

Author's contributions

AG and JHJ developed the idea and lead the project. NR wrote all the code and performed all the calculations. All authors contributed to the data analysis. All authors read and approved the final manuscript.

Funding

This work was supported by Bayer AG.

Availability of data and materials

RegioML is freely available under the MIT open source license at:
https://github.com/jensengroup/RegioML.

Additional code for dataset curation and machine learning training etc. are available at:
https://sid.erda.dk/sharelink/HypB1igzDl.

Competing interests

The authors declare that there are no competing interests.

References

(1) Hartenfeller, M.; Eberle, M.; Meier, P.; Nieto-Oberhuber, C.; Altmann, K.-H.; Schneider, G.; Jacoby, E.; Renner, S. *J. Chem. Inf. Model.* **2011**, *51*, 3093–3098.

(2) Liljenberg, M.; Brinck, T.; Herschend, B.; Rein, T.; Tomasi, S.; Svensson, M. *J. Org. Chem.* **2012**, *77*, 3262–3269.

(3) Jorner, K.; Brinck, T.; Norrby, P.-O.; Buttar, D. *Chem. Sci.* **2021**, *12*, 1163–1175.

(4) Kruszyk, M.; Jessing, M.; Kristensen, J. L.; Jørgensen, M. *J. Org. Chem.* **2016**, *81*, 5128–5134.

(5) Kromann, J. C.; Jensen, J. H.; Kruszyk, M.; Jessing, M.; Jørgensen, M. *Chem. Sci.* **2018**, *9*, 660–665.

(6) Ree, N.; Göller, A. H.; Jensen, J. H. *J. Cheminformatics* **2021**, *13*.

(7) Tomberg, A.; Johansson, M. J.; Norrby, P.-O. *J. Org. Chem.* **2019**, *84*, 4695–4703.

(8) Struble, T. J.; Coley, C. W.; Jensen, K. F. *React. Chem. Eng.* **2020**, *5*, 896–902.

(9) yanfei guan,; Coley, C.; wu, H.; Ranasinghe, D.; esther heid,; Struble, T. J.; Pattanaik, L.; Green, W. H.; Jensen, K. F. **2020**,

(10) Beker, W.; Gajewska, E. P.; Badowski, T.; Grzybowski, B. A. *Angew. Chem. Int. Ed Engl.* **2019**, *58*, 4515–4519.

(11) Moskal, M.; Beker, W.; Szymkuć, S.; Grzybowski, B. A. *Angew. Chem. Int. Ed Engl.* **2021**,

(12) Wang, L.; Zhang, C.; Bai, R.; Li, J.; Duan, H. *Chem. Commun.* **2020**, *56*, 9368–9371.

(13) Li, X.; Zhang, S.; Xu, L.; Hong, X. *Angew. Chem. Int. Ed.* **2020**, *59*, 13253–13259.

(14) Roszak, R.; Beker, W.; Molga, K.; Grzybowski, B. A. *J. Am. Chem. Soc.* **2019**, *141*, 17142–17149.

(15) Grimme, S.; Brandenburg, J. G.; Bannwarth, C.; Hansen, A. *J. Chem. Phys.* **2015**, *143*, 054107.

(16) Cossi, M.; Barone, V. *J. Chem. Phys.* **1998**, *109*, 6246–6254.

(17) Garcia-Ratés, M.; Neese, F. *J. Comput. Chem.* **2019**, *40*, 1816–1828.

(18) Neese, F. *WIREs Comput. Mol. Sci.* **2017**, *8*.

(19) Finkelmann, A. R.; Göller, A. H.; Schneider, G. *Chem. Commun.* **2016**, *52*, 681–684.

(20) Finkelmann, A. R.; Göller, A. H.; Schneider, G. *ChemMedChem* **2017**, *12*, 606–612.

(21) Finkelmann, A. R.; Goldmann, D.; Schneider, G.; Göller, A. H. *ChemMedChem* **2018**, *13*, 2281–2289.

(22) Bauer, C. A.; Schneider, G.; Göller, A. H. *Mol. Inform.* **2018**, *38*, 1800115.

(23) Bauer, C. A.; Schneider, G.; Göller, A. H. *J. Cheminformatics* **2019**, *11*.

(24) Kuhnke, L.; ter Laak, A.; Göller, A. H. *J. Chem. Inf. Model.* **2019**, *59*, 668–672.

(25) Marenich, A. V.; Jerome, S. V.; Cramer, C. J.; Truhlar, D. G. *J. Chem. Theory Comput.* **2012**, *8*, 527–541.

(26) Grimme, S.; Bannwarth, C.; Shushkov, P. *J. Chem. Theory Comput.* **2017**, *13*, 1989–2009.

(27) Wang, S.; Witek, J.; Landrum, G. A.; Riniker, S. *J. Chem. Inf. Model.* **2020**, *60*, 2044–2058.

(28) RDKit: Open-source cheminformatics. `http://www.rdkit.org` (version 2020.09.4).

(29) Sivaraman, G.; Jackson, N. E.; Sanchez-Lengeling, B.; Vázquez-Mayagoitia, Á.; Aspuru-Guzik, A.; Vishwanath, V.; de Pablo, J. J. *Machine Learning: Science and Technology* **2020**, *1*, 025015.

(30) Rogers, D.; Hahn, M. *J. Chem. Inf. Model.* **2010**, *50*, 742–754.

(31) Butina, D. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 747–750.

(32) Ali, M. PyCaret: An open source, low-code machine learning library in Python. 2020; PyCaret version 2.3.2.

(33) Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A next-generation hyperparameter optimization framework. Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2019.

(34) Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A highly efficient gradient boosting decision tree. Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook, NY, USA, 2017; p 3149–3157.

(35) Pedregosa, F. et al. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

(36) Wallach, I.; Heifets, A. *J. Chem. Inf. Model.* **2018**, *58*, 916–932.

(37) Matthews, B. *Biochimica et Biophysica Acta (BBA) - Protein Structure* **1975**, *405*, 442–451.

Supporting Information

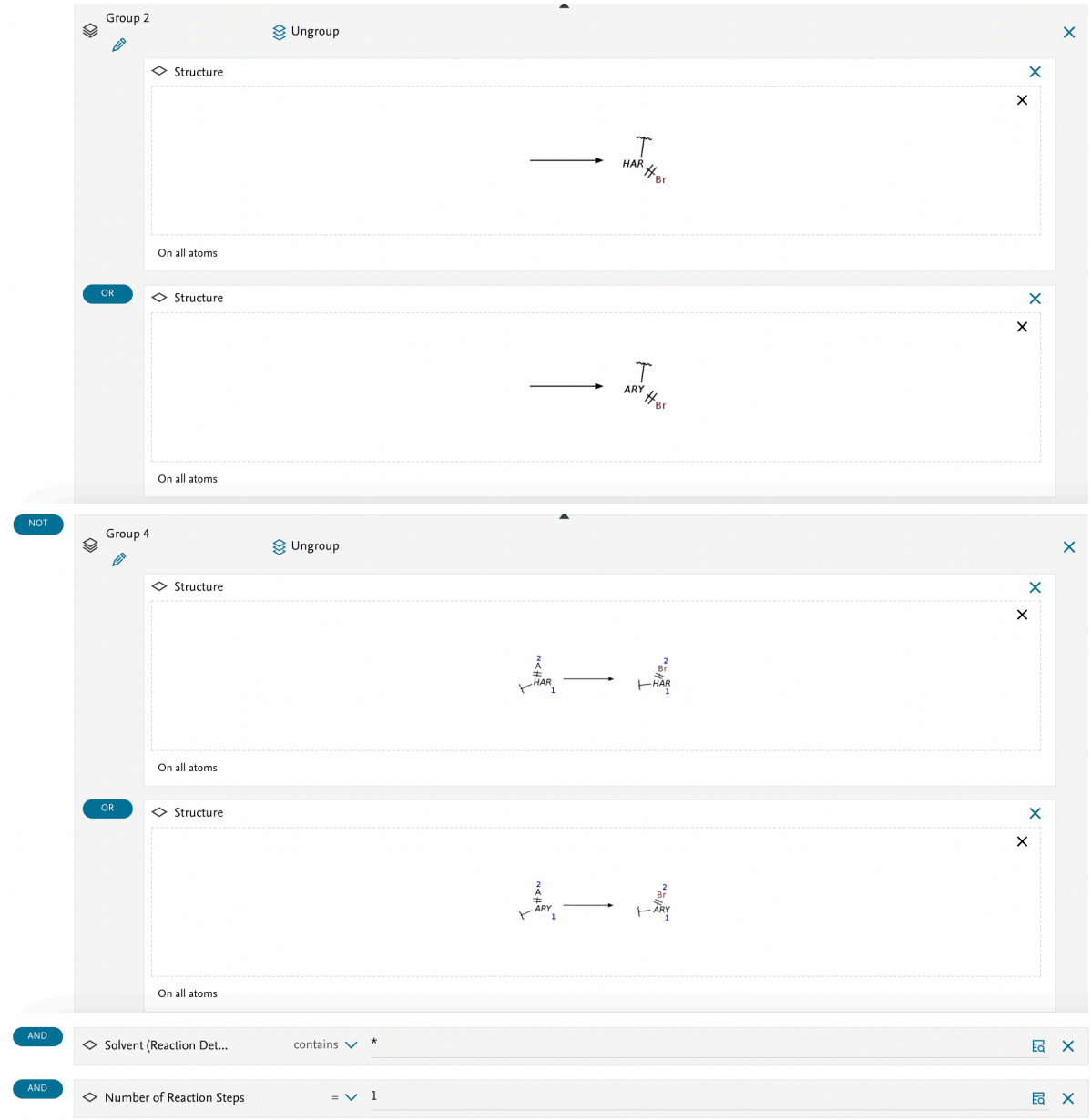

Figure S1: The set of queries used to extract the reaction data from Reaxys.

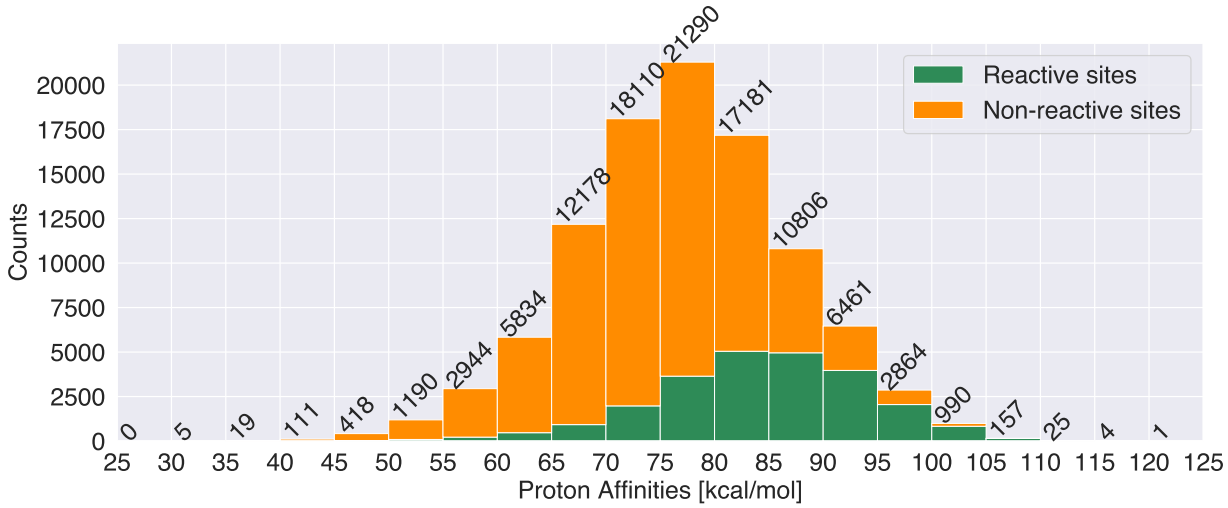

Figure S2: The distribution of calculated proton affinities for all of the collected data using the original version of RegioSQM20.

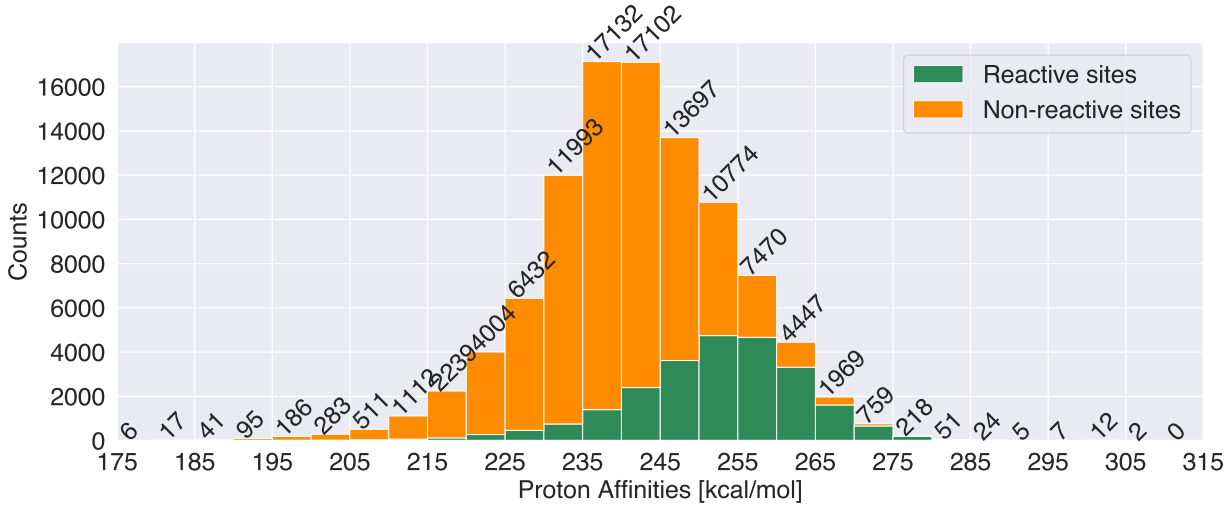

Figure S3: The distribution of calculated proton affinities for all of the collected data using the extended version of RegioSQM20.

Table S1: A description of the different atomic descriptors used for machine learning. The descriptors are based on the CM5 charge scheme.

Descriptor abbreviation	**Description**
Sorted-shell	Charge shell descriptor with values sorted by Cahn-Ingold-Prelog rules
CS	Charge shell descriptor with average charge per shell
CRDF	Spatial charge radial distribution function
CACF	Spatial charge autocorrelation function (split into positive and negative parts)
MS	Mass shell; the elements are the sums of the masses of each shell
GACF	Topological charge autocorrelation function
Combinatorial	Combination of shorted-shell, CACF, and CS

Table S2: 5-fold cross-validation AUC-ROC scores for the seven different atomic descriptors using the LightGBM model on the randomly split training set. AUC-ROC corresponds to the area under the curve of the receiver operating characteristic curve.

Descriptor	**Settings**	**Dimensions**	**5-fold cross-validation AUC-ROC score**
Sorted-shell	shells: 3	53	0.949 ± 0.002
CS	shells: 3	4	0.891 ± 0.003
CRDF	r_{min}: 1, r_{max}: 6, β: 20, step size: 0.2	25	0.931 ± 0.002
CACF	r_{min}: 1, r_{max}: 8, step size: 0.5	28	0.921 ± 0.002
MS	shells: 2	3	0.782 ± 0.005
GACF	r_{min}: 1, r_{max}: 3	6	0.898 ± 0.004
Combinatorial	sorted-shell (shells: 2), CACF (r_{min}: 1, r_{max}: 10, step size: 0.2), CS (shells: 7)	115	0.946 ± 0.002

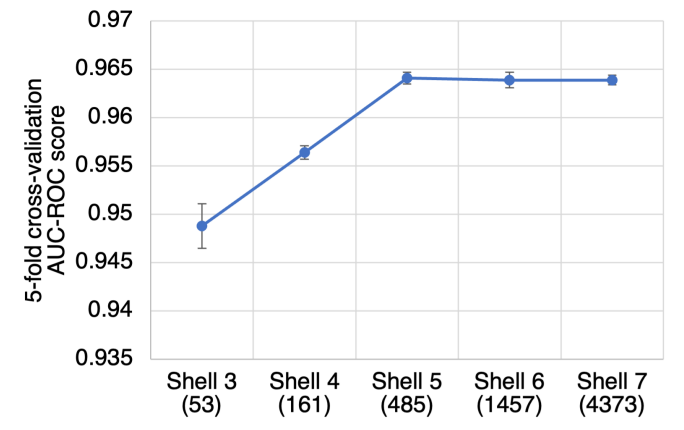

Figure S4: Increasing the number of included shells in the sorted-shell descriptor. The numbers in parenthesis correspond to the length of the feature vectors.

Initial ML screening with PyCaret

PyCaret version 2.2.0[32] is used as an initial screening of 17 regression models and 13 classification models to find promising models. To evaluate the performance of the models, we use a 5-fold cross-validation scheme of the atomic data for atoms in molecules belonging to the randomly split training set. The atomic data consist of the sorted-shell descriptor with 5 shells in combination with either a binary label corresponding to whether or not a bromination reaction has been experimentally observed on the specific site or the calculated proton affinity obtained by the original RegioSQM20 method. The sorted-shell descriptor with 3 shells and the combinatorial descriptor were also tested in this initial screening, but the ranking of the different models were similar to those presented in Figures S5 and S6.

The top-3 performing models for both tasks are the extra-trees and random forest models as implemented in scikit-learn 0.24.2,[35] and the light gradient boosting machine (LightGBM) model version 3.1.1[34] (see Figures S5 and S6).

Due to the good performance of the LightGBM model, we also tested a similar model called extreme gradient boosting (XGBoost) version 1.4.0, and a new deep neural network architecture for tabular data called TabNet by Google Cloud AI, which has recently outperformed several gradient boosting algorithms on different tasks. In the case of the latter, we used a pyTorch implementation of TabNet by DreamQuark. The results of the different optimized machine learning models can be found in Table S3.

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
et	Extra Trees Classifier	0.9204	0.9549	0.7655	0.8919	0.8239	0.7729	0.7766	14.456
rf	Random Forest Classifier	0.9164	0.9509	0.7520	0.8869	0.8139	0.7604	0.7648	13.166
lightgbm	Light Gradient Boosting Machine	0.9098	0.9463	0.7490	0.8619	0.8015	0.7435	0.7466	2.332
knn	K Neighbors Classifier	0.8891	0.9145	0.7256	0.7998	0.7609	0.6889	0.6902	223.358
gbc	Gradient Boosting Classifier	0.8849	0.9114	0.6581	0.8338	0.7355	0.6633	0.6709	39.038
dt	Decision Tree Classifier	0.8650	0.8197	0.7316	0.7185	0.7250	0.6355	0.6356	3.532
ada	Ada Boost Classifier	0.8571	0.8891	0.5968	0.7646	0.6702	0.5808	0.5882	9.654
lr	Logistic Regression	0.8045	0.8019	0.3475	0.6970	0.4636	0.3601	0.3928	12.014
ridge	Ridge Classifier	0.8028	0.0000	0.3092	0.7206	0.4326	0.3356	0.3798	0.470
lda	Linear Discriminant Analysis	0.8026	0.8023	0.3687	0.6720	0.4760	0.3669	0.3921	4.776
svm	SVM - Linear Kernel	0.8010	0.0000	0.2907	0.7304	0.4138	0.3203	0.3705	1.312
qda	Quadratic Discriminant Analysis	0.7569	0.5010	0.0028	0.5162	0.0056	0.0031	0.0235	3.484
nb	Naive Bayes	0.2583	0.5070	0.9893	0.2456	0.3936	0.0063	0.0382	0.354

Figure S5: 5-fold cross-validation results on the randomly split training set using PyCaret w.r.t. classification.

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
et	Extra Trees Regressor	2.1592	9.8424	3.1367	0.8991	0.0416	0.0286	78.342
rf	Random Forest Regressor	2.5242	12.9645	3.5998	0.8671	0.0475	0.0334	117.438
lightgbm	Light Gradient Boosting Machine	2.7021	13.2370	3.6373	0.8643	0.0476	0.0355	1.998
gbr	Gradient Boosting Regressor	3.8416	24.9357	4.9930	0.7444	0.0658	0.0509	38.392
knn	K Neighbors Regressor	3.4953	26.5942	5.1566	0.7274	0.0682	0.0466	113.402
dt	Decision Tree Regressor	3.7041	29.2604	5.4091	0.7000	0.0710	0.0489	1.950
ada	AdaBoost Regressor	5.4477	45.7395	6.7616	0.5311	0.0892	0.0734	26.258
br	Bayesian Ridge	5.6272	51.7361	7.1923	0.4696	0.0925	0.0737	2.690
ridge	Ridge Regression	5.6454	51.9586	7.2077	0.4674	0.0927	0.0740	0.238
omp	Orthogonal Matching Pursuit	5.7171	53.2138	7.2944	0.4545	0.0945	0.0750	0.284
huber	Huber Regressor	5.6202	53.3203	7.3017	0.4534	0.0935	0.0730	21.912
par	Passive Aggressive Regressor	6.1183	62.6318	7.8882	0.3581	0.1012	0.0787	1.406
en	Elastic Net	7.6909	96.5902	9.8279	0.0098	0.1293	0.1034	0.176
lasso	Lasso Regression	7.7308	97.5523	9.8767	-0.0001	0.1300	0.1040	0.158
llar	Lasso Least Angle Regression	7.7308	97.5523	9.8767	-0.0001	0.1300	0.1040	0.284
lr	Linear Regression	6.7012	9163.8243	70.4332	-93.4221	0.1129	0.0890	0.732

Figure S6: 5-fold cross-validation results on the randomly split training set using PyCaret w.r.t. regression.

Table S3: Comparing different optimized machine learning methods using the random split of the molecular data to obtain the training and test sets.

Method	Test set			Out-of-sample set		
	AUC-ROC	ACC	MCC	AUC-ROC	ACC	MCC
Stratified split						
Random Forest	0.96	0.92	0.78	0.93	0.89	0.68
Extra Trees	0.96	0.93	0.79	0.93	0.89	0.68
XGBoost	0.96	0.93	0.80	0.93	0.89	0.70
TabNet	0.94	0.90	0.73	0.87	0.84	0.57
LightGBM	0.97	0.93	0.81	0.94	0.90	0.72
LightGBM RegioSQM20	0.92	0.88	0.69	0.90	0.86	0.62
LightGBM RegioSQM20 PBEh-3c	0.92	0.90	0.72	0.90	0.87	0.65

Table S4: Comparing the use of either a stratified or random split of the molecular data to obtain the training and test sets. Note that the test sets are different between the stratified and random split, but the out-of-sample set is identical.

Method	Test set			Out-of-sample set		
	AUC-ROC	ACC	MCC	AUC-ROC	ACC	MCC
Stratified split						
LightGBM	0.97	0.93	0.81	0.93	0.89	0.71
LightGBM RegioSQM20	0.92	0.88	0.69	0.90	0.85	0.62
LightGBM RegioSQM20 PBEh-3c	0.92	0.90	0.72	0.90	0.87	0.65
Random split						
LightGBM	0.97	0.93	0.81	0.94	0.90	0.72
LightGBM RegioSQM20	0.92	0.88	0.69	0.90	0.86	0.62
LightGBM RegioSQM20 PBEh-3c	0.92	0.90	0.72	0.90	0.87	0.65

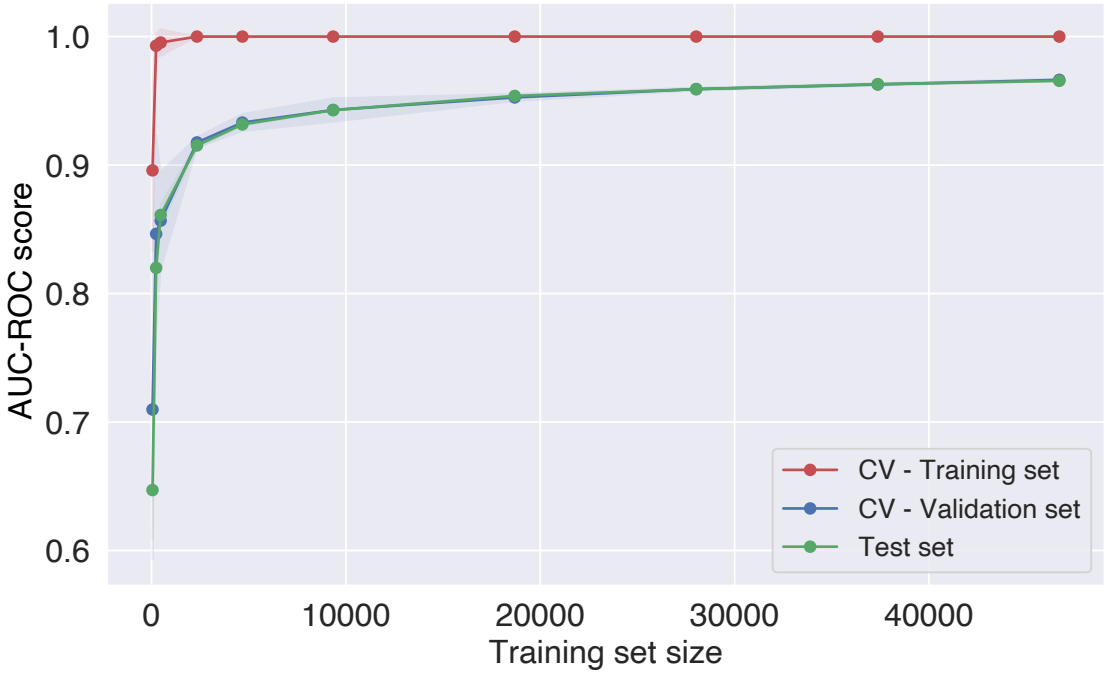

Figure S7: Learning curve for the LightGBM model. The training set size corresponds to the number of unique reaction sites as the model is trained on atoms instead of molecules.

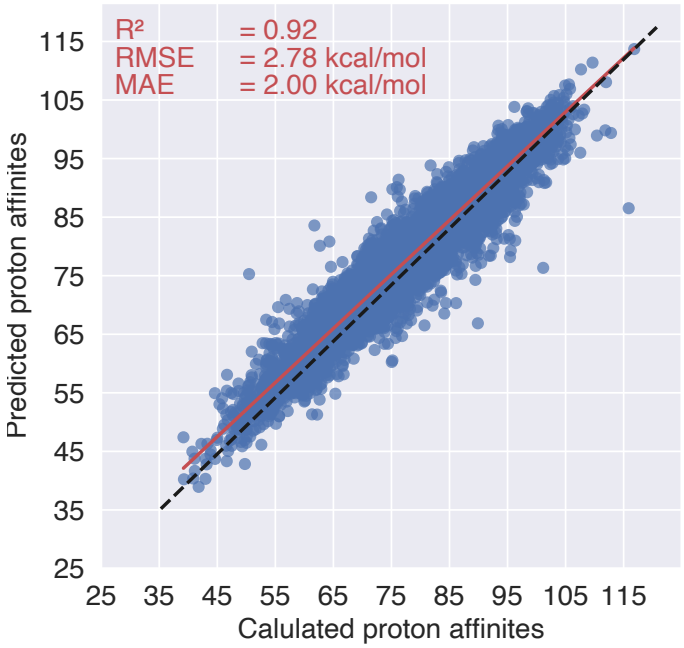

Figure S8: Performance of the LightGBM RegioSQM20 regression model showing the predicted proton affinities versus the calculated proton affinities for the test set.

Graphical TOC Entry