Electronic Supplementary Material (ESI) for Digital Discovery.

Supporting Information

Table S1: The training, test, and out-sample sets contain, respectively, 372, 205, and 381 different aromatic rings and the latter two sets contain 48 and 183 rings not found in the training set. Similarly, the training, test, and out-sample sets contain, respectively, 1553, 620, and 1714 different substiuents and the latter two sets contain 210 and 1180 substiuents not found in the training set. The SMILES strings for all rings and substiuents are can be found in the data repository.

	Aromatic rings	Not in training set	Substiuents	Not in training set
Training set	372	0	1553	0
Test set	205	48	620	210
Out-of-sample	381	183	1714	1180

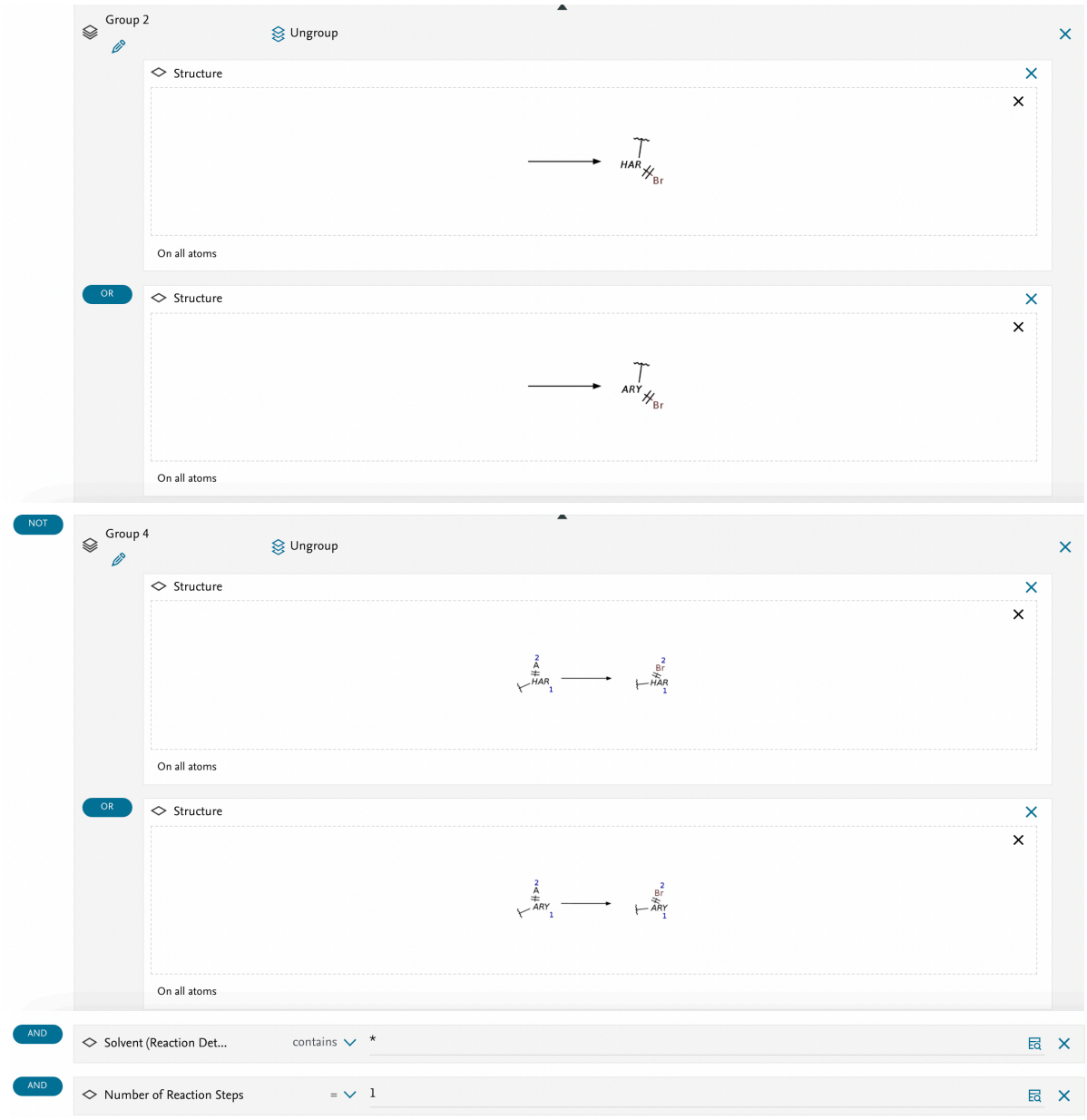

Figure S1: The set of queries used to extract the reaction data from Reaxys.

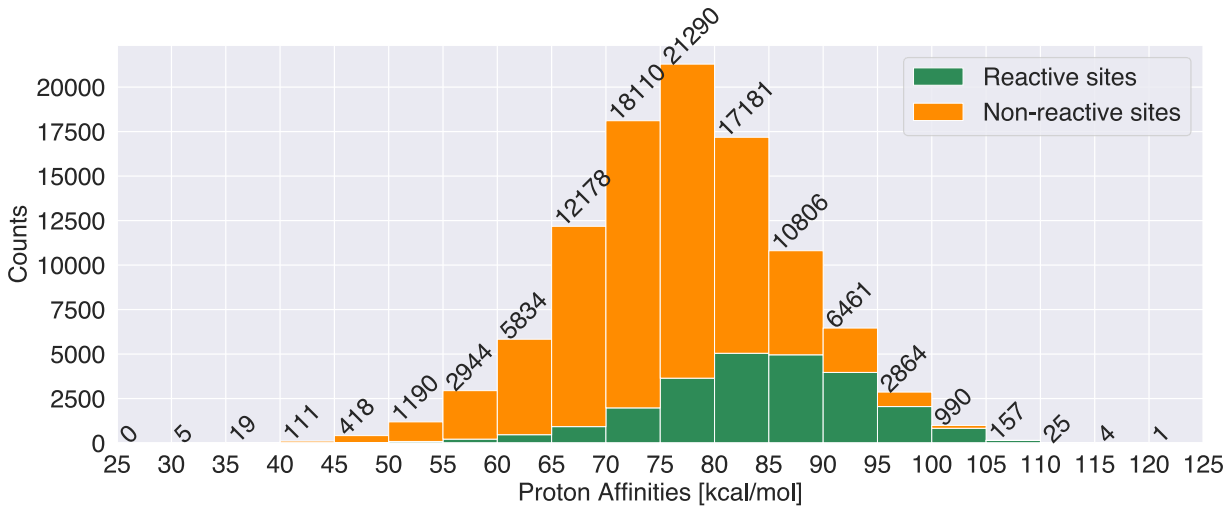

Figure S2: The distribution of calculated proton affinities for all of the collected data using the original version of RegioSQM20.

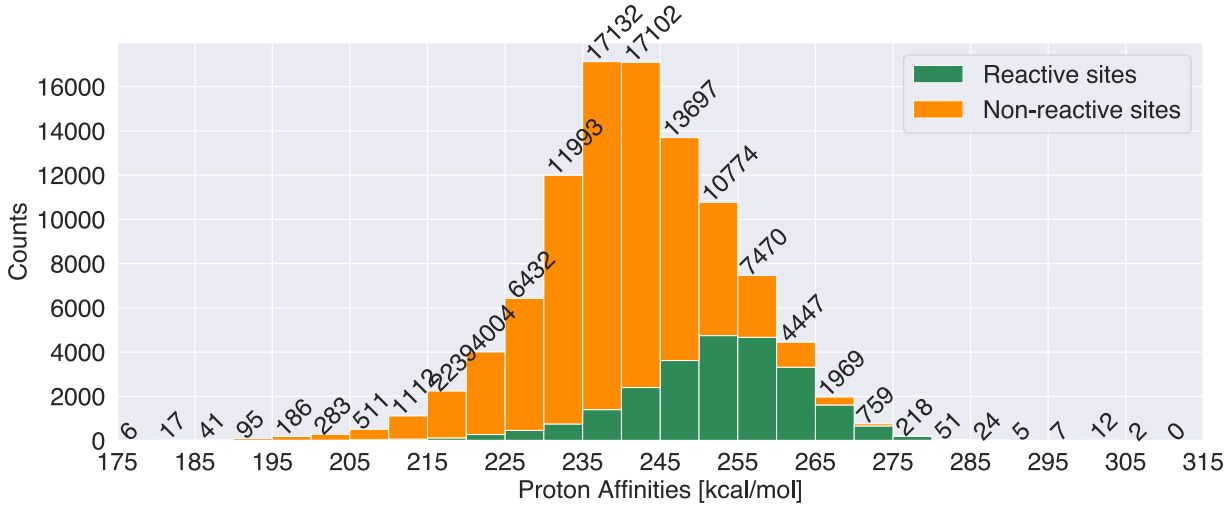

Figure S3: The distribution of calculated proton affinities for all of the collected data using the extended version of RegioSQM20.

Table S2: 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 S3: 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

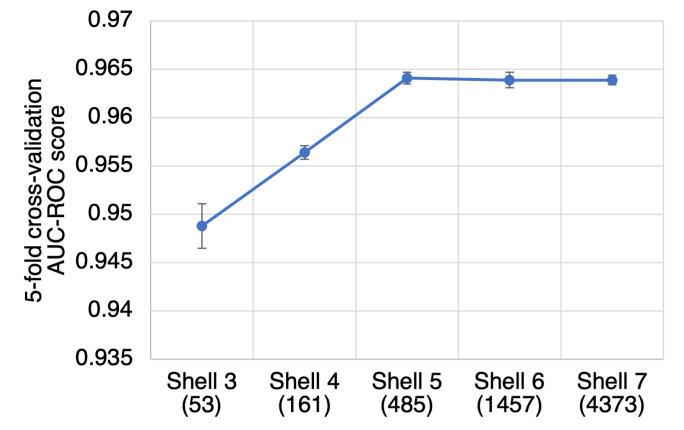

Figure S4: Increasing the number of included shells in the sorted-shell descriptor. The numbers in parenthesis correspond to the length of the feature vectors.

Initial ML screening with PyCaret

PyCaret version 2.2.0[33] 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,[36] and the light gradient boosting machine (LightGBM) model version 3.1.1[35] (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 S4.

	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 S4: 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 S5: 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

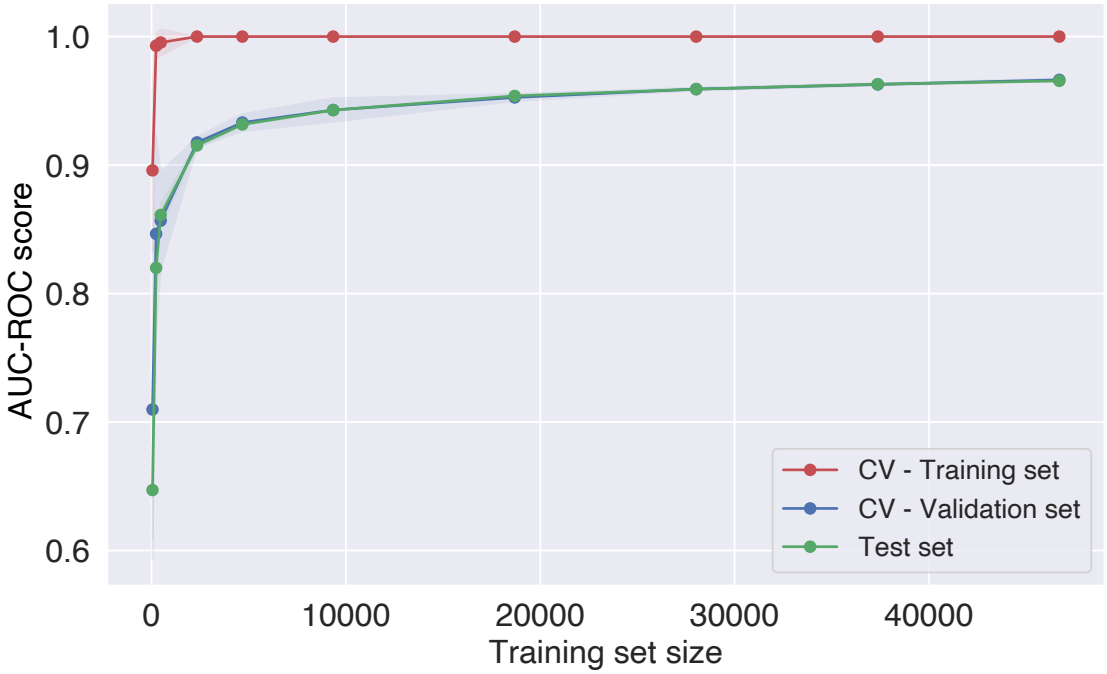

Figure S7: Learning curve for the LightGBM model. The training set size corresponds to the number of unique reaction sites as the model is trained on atoms instead of molecules.

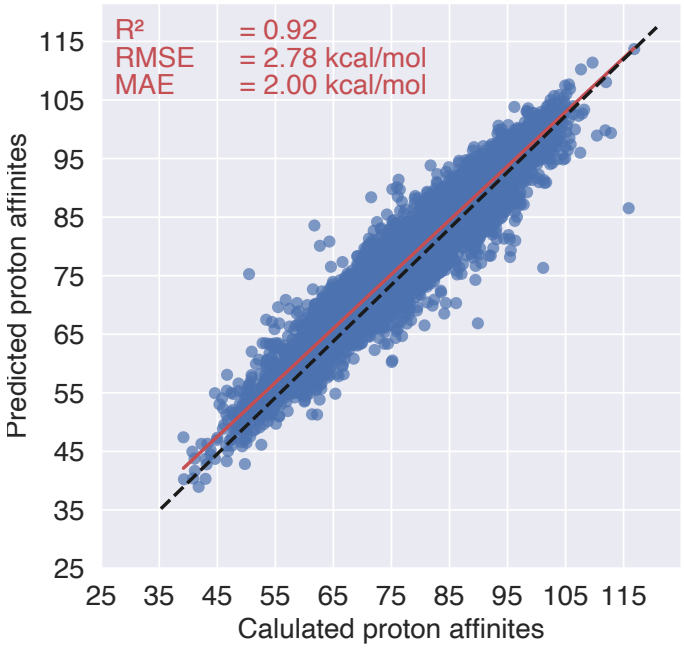

Figure S8: Performance of the LightGBM RegioSQM20 regression model showing the predicted proton affinities versus the calculated proton affinities for the test set.