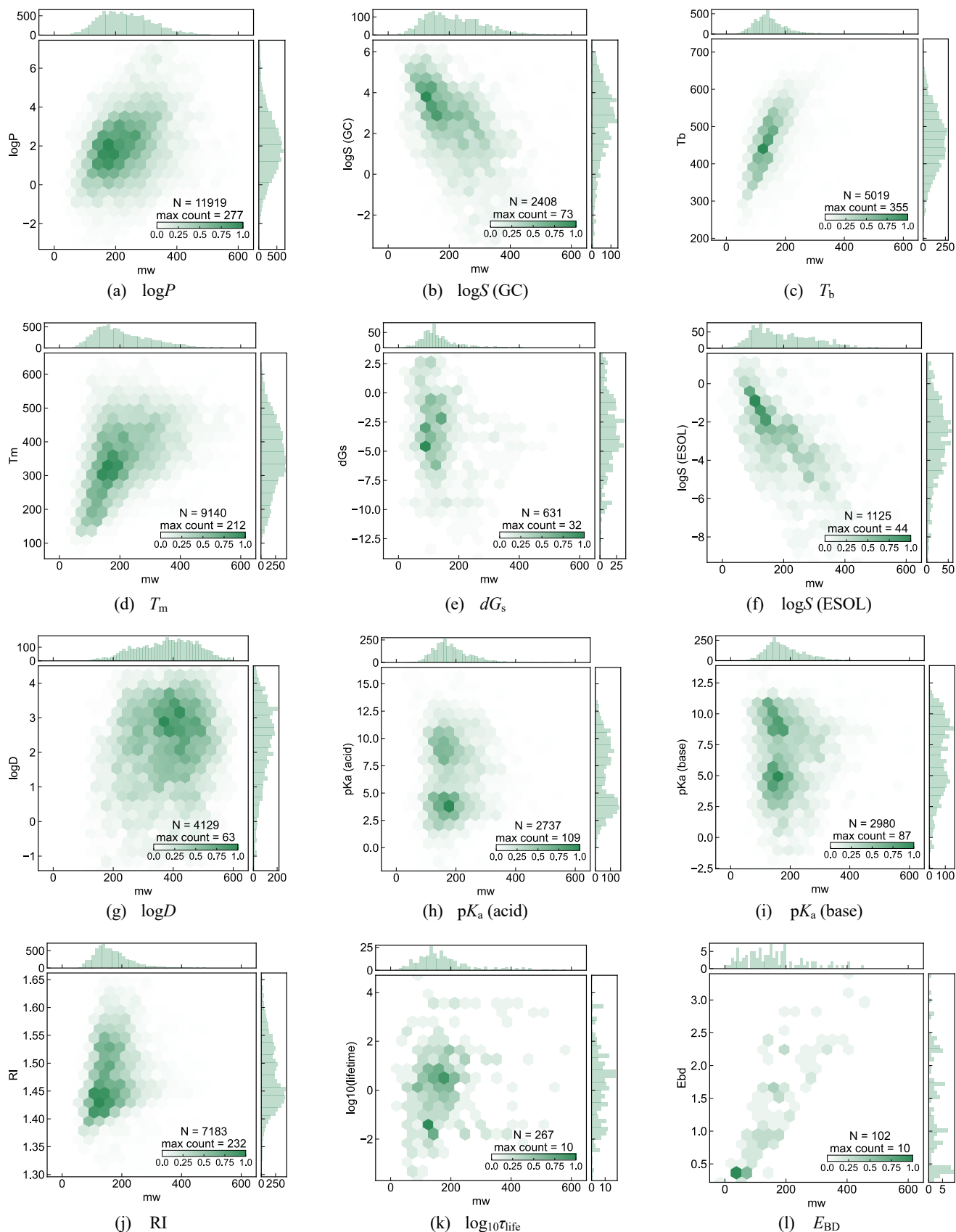


Supplementary Information

Extrapolative prediction of small-data molecular property using quantum mechanics-assisted machine learning

Hajime Shimakawa¹, Akiko Kumada¹, and Masahiro Sato¹

¹ Department of Electrical Engineering & Information Systems, School of Engineering,
The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, 113-8656, Tokyo, Japan



Supplementary Fig. 1. Density distribution between molecular weight (MW) and target property of each dataset. Molecular weight is limited within 600 MW for benchmark. Each property range is limited within $\mu \pm 3\sigma$ from raw data points, where μ is mean value and σ is standard deviation of property range to remove influences of outlier points. Number of data points N and colormap of density are presented in the upper right.

Supplementary Table 1. List of data points of molecules for property dataset and points contained in QMEx and QMugs datasets used for training.

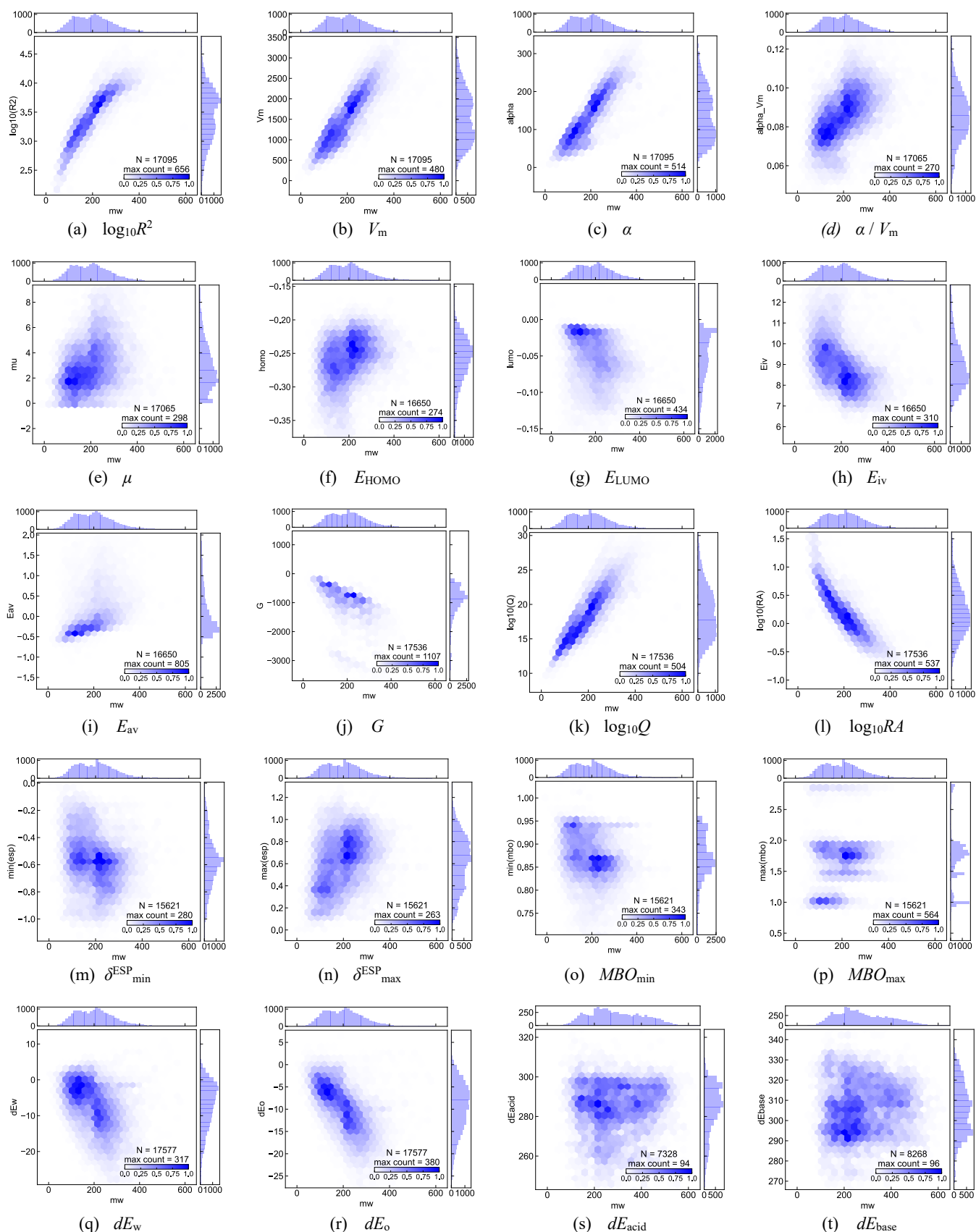
No.	Parameter	Data points for benchmark (after curation)	Data points contained in QMEx dataset out of 26k mols (rate of points for benchmark)	Data points contained in QMugs dataset out of 106k mols (rate of points for benchmark)
1	$\log P$	11,910	6160 (52%)	1580 (13%)
2	$\log S$ (GC)	2408	1507 (63%)	726 (30%)
3	T_b	5019	3001 (60%)	2109 (42%)
4	T_m	9140	4669 (51%)	2405 (26%)
5	dG_s	631	604 (96%)	331 (53%)
6	$\log S$ (ESOL)	1125	888 (79%)	465 (41%)
7	$\log D$	4129	3577 (87%)	464 (11%)
8	pK_a (acid)	2737	2317 (85%)	255 (9%)
9	pK_a (base)	2980	2161 (73%)	343 (12%)
10	RI	7183	2293 (32%)	1421 (20%)
11	$\log_{10} \tau_{life}$	267	225 (84%)	110 (41%)
12	E_{BD}	102	82 (80%)	48 (47%)
	Unduplicated total molecules	31,687	16,890 (53%)	4189 (13%)

Supplementary Table 2. List of QMEx descriptors calculated by RDkit and DFT calculations with Gaussian 16. The total number of unduplicated molecules for training QMEx descriptors is 25,650 (26k mols).

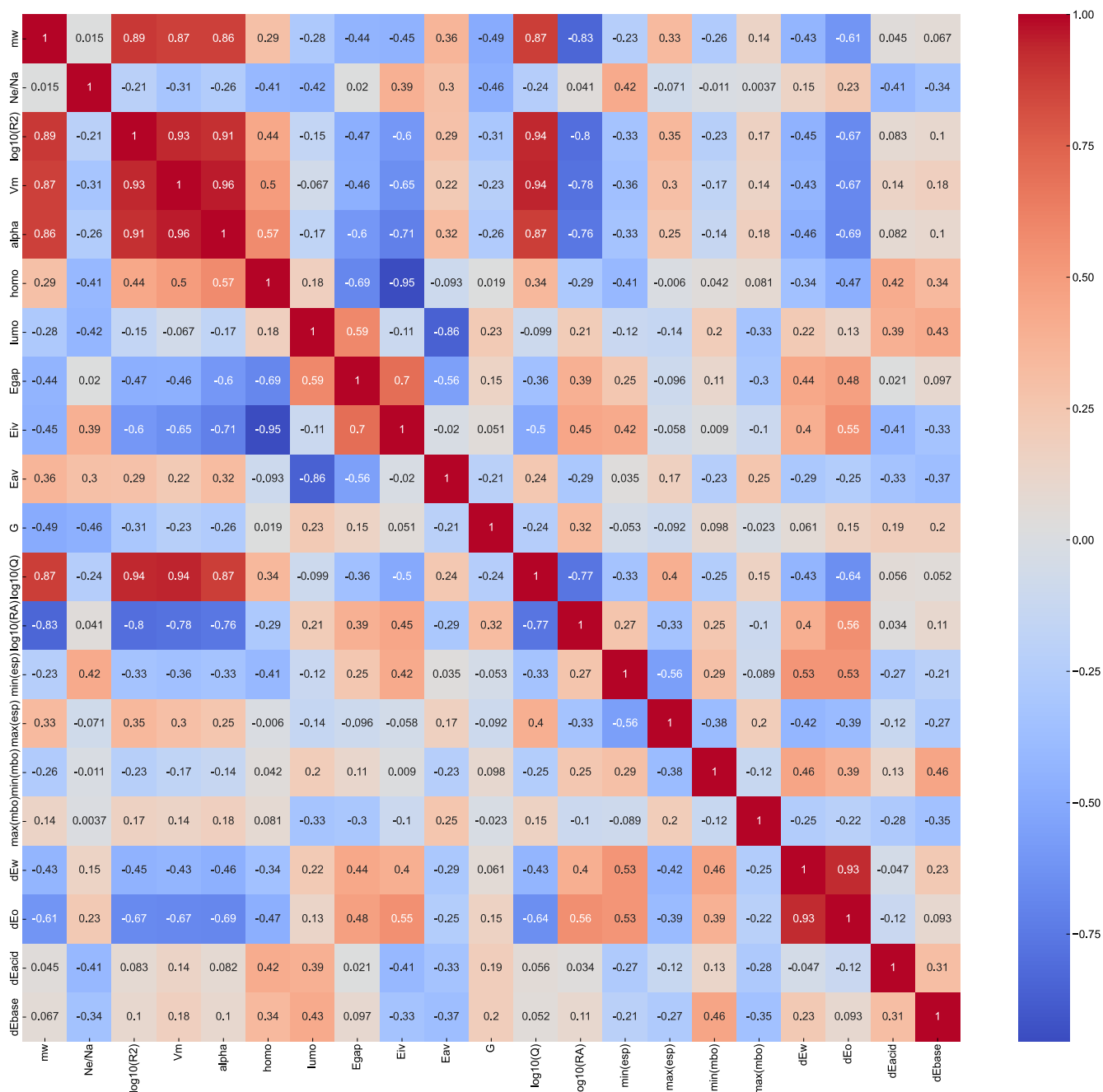
No.	Descriptor	Parameter	Unit	Data points	Calculation method	Surrogate-GINs accuracy		
						Model number	RMSE	R ²
1	Molecular weight	MW	1		RDkit			
2	Number of electrons per atom	N_e/N_a	1		RDkit			
3	Electronic spatial extent	$\log_{10} R^2$	1	17,095	DFT (B3LYP/aug-cc-pVDZ, scf)	1	0.052	0.987
4	Molecular volume	V_m	bohr ³	17,095	DFT (B3LYP/aug-cc-pVDZ, volume)	1	84.645	0.983
5	Polarizability	α	bohr ³	17,095	DFT (B3LYP/aug-cc-pVDZ, polar)	1	5.834	0.993
6	Ratio of polarizability to molecular volume	α / V_m	1	17,065	DFT (B3LYP/aug-cc-pVDZ, polar/volume)	2	0.006	0.831
7	Dipole moment	μ	D	17,065	DFT (B3LYP/aug-cc-pVDZ, polar)	2	1.149	0.699
8	Energy of HOMO	E_{HOMO}	E_h	16,650	DFT (B3LYP/aug-cc-pVDZ, scf)	3	0.010	0.925
9	Energy of LUMO	E_{LUMO}	E_h	16,650	DFT (B3LYP/aug-cc-pVDZ, scf)	3	0.011	0.896
10	Vertical ionization energy	E_{iv}	eV	16,650	DFT (B3LYP/aug-cc-pVDZ, scf [1])	3	0.305	0.943
11	Vertical electron affinity	E_{av}	eV	16,650	DFT (B3LYP/aug-cc-pVDZ, scf [1])	3	0.233	0.867
12	Free energy at 298.15 K	G	E_h	17,536	DFT (B3LYP/aug-cc-pVDZ, frequency)	4	52.832	0.996
13	Molecular partition function	$\log_{10} Q$	1	17,536	DFT (B3LYP/aug-cc-pVDZ, frequency)	4	0.413	0.989
14	Rotational constant	$\log_{10} RA$	1	17,536	DFT (B3LYP/aug-cc-pVDZ, frequency)	4	0.109	0.944
15	Minimum of ESP charge	δ_{min}^{ESP}	1	15,621	DFT (B3LYP/aug-cc-pVDZ, frequency)	5	0.077	0.845
16	Maximum of ESP charge	δ_{max}^{ESP}	1	15,621	DFT (B3LYP/aug-cc-pVDZ, population)	5	0.087	0.868
17	Minimum of mayer bond order	MBO_{min}	1	15,621	DFT (B3LYP/aug-cc-pVDZ, population)	5	0.025	0.799
18	Maximum of mayer bond order	MBO_{max}	1	15,621	DFT (B3LYP/aug-cc-pVDZ, population)	5	0.073	0.971
19	Water solvation energy	dE_w	kcal/mol	17,577	DFT (B3LYP/aug-cc-pVDZ, SMD, octanol) [2]	6	1.717	0.943
20	Octanol solvation energy	dE_o	kcal/mol	17,577	DFT (B3LYP/aug-cc-pVDZ, SMD, octanol) [2]	6	1.403	0.943
21	Protonation energy in water (to become conjugate acid)	dE_{acid}	kcal/mol	7328	DFT (B3LYP/6-31G(d), SMD, water) [2,3]	7	5.618	0.827
22	Deprotonation energy in water (to become conjugate base)	dE_{base}	kcal/mol	8268	DFT (B3LYP/6-31G(d), SMD, water) [2,3]	8	5.082	0.852

Supplementary Table 3. List of QMugs descriptors calculated by RDkit and extracted from datasets of QMugs [4] and CombiSolv-QM [5]. The two datasets are collectively referred to as the QMugs dataset for simplicity. The total number of unduplicated molecules for training QMugs descriptors is 106,401 (106k mols).

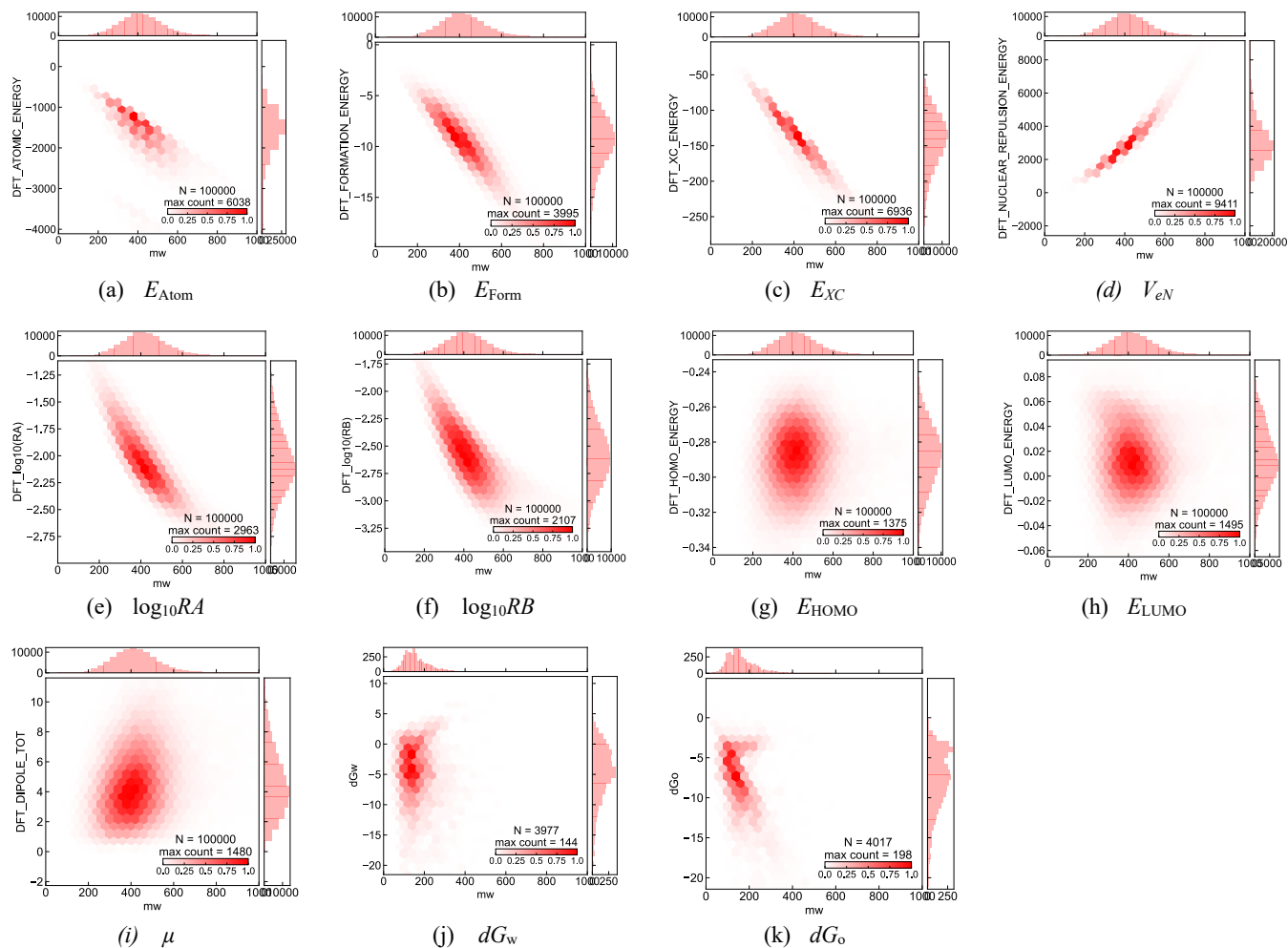
No.	Descriptor	Parameter	Unit	Data points	Calculation method	Surrogate-GINs accuracy		
						Model number	RMSE	R ²
1	Molecular weight	MW	1		RDkit			
2	Number of electrons per atom	N_e/N_a	1		RDkit			
3	Total internal atomic energy	E_{Atom}	E_h	100,000	DFT (ω B97X-D/def2-SVP)	1	39.454	0.998
4	Formation energy	E_{Form}	E_h	100,000	DFT (ω B97X-D/def2-SVP)	1	0.108	0.999
5	Exchange correlation energy	E_{XC}	E_h	100,000	DFT (ω B97X-D/def2-SVP)	1	1.778	0.999
6	Nuclear repulsion energy	V_{eN}	E_h	100,000	DFT (ω B97X-D/def2-SVP)	1	180.127	0.992
7	Rotational constants (A)	$\log_{10} RA$	1	100,000	DFT (ω B97X-D/def2-SVP)	1	0.107	0.884
8	Rotational constants (B)	$\log_{10} RB$	1	100,000	DFT (ω B97X-D/def2-SVP)	1	0.114	0.847
9	Energy of HOMO	E_{HOMO}	E_h	100,000	DFT (ω B97X-D/def2-SVP)	2	0.006	0.883
10	Energy of LUMO	E_{LUMO}	E_h	100,000	DFT (ω B97X-D/def2-SVP)	2	0.007	0.933
11	Total dipole moment	μ	D	100,000	DFT (ω B97X-D/def2-SVP)	3	1.794	0.397
12	Water solvation energy	dG_w	kcal/mol	3977	COSMOtherm [6]	4	1.929	0.875
13	Octanol solvation energy	dG_o	kcal/mol	4017	COSMOtherm [6]	5	1.319	0.910



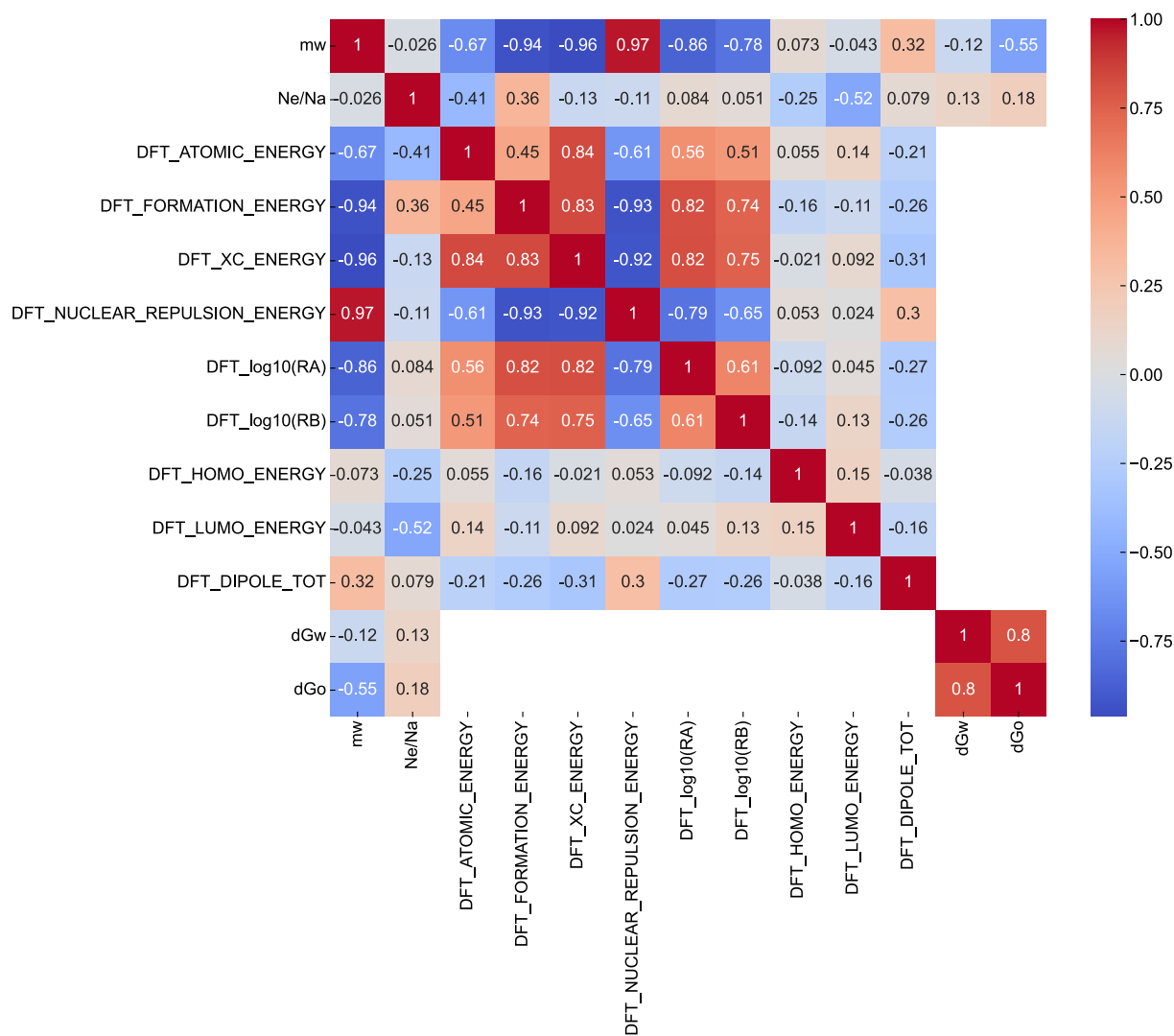
Supplementary Fig. 2. Density distribution between molecular weight (MW) and calculated value of QMEX descriptors. Molecular weight is limited within 600 MW, and each set of calculated values is limited within $\mu \pm 6\sigma$ from raw data points, where μ is mean value and σ is standard deviation of property range to remove influences of outlier points. Number of data points N and colormap of density are presented in the upper



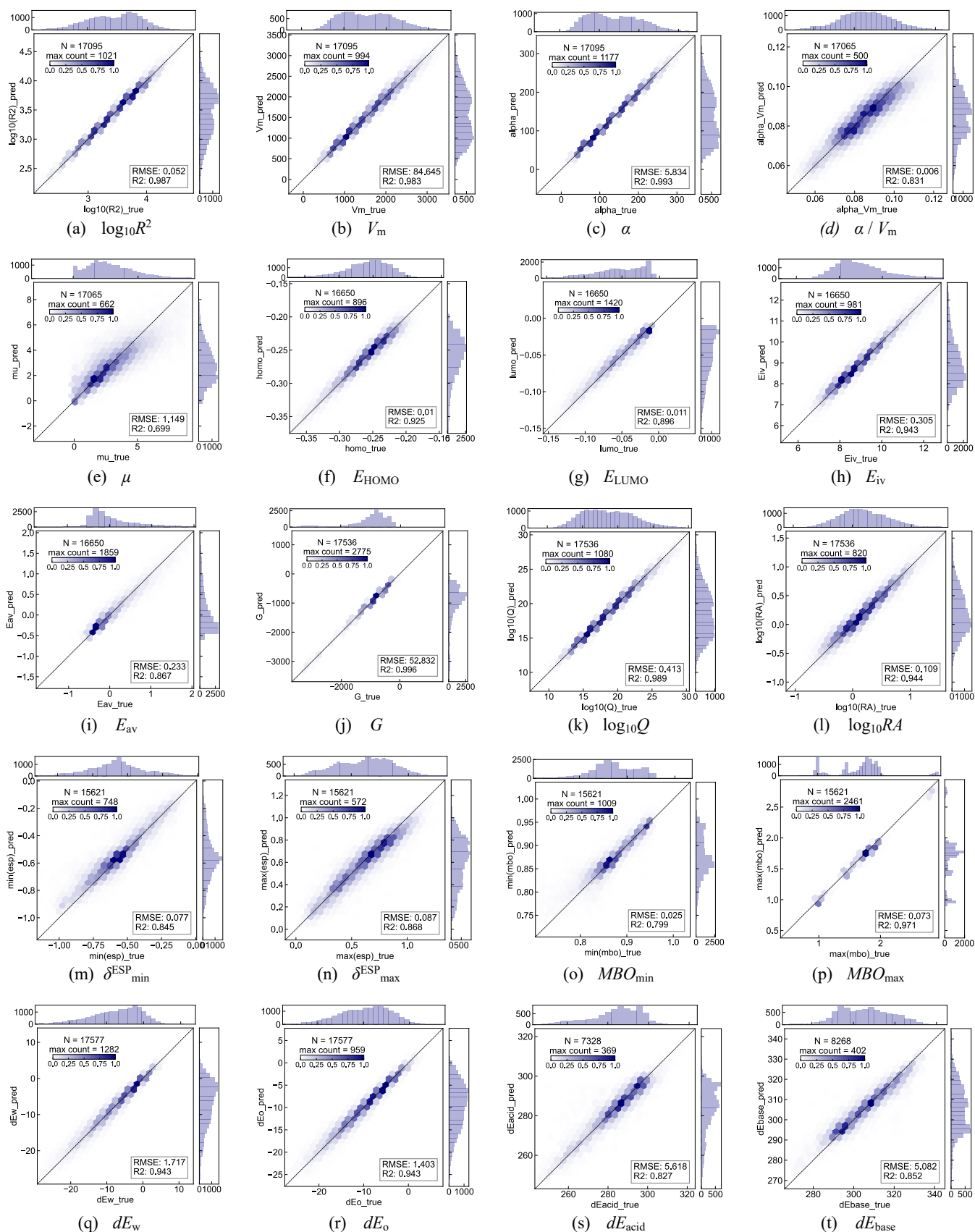
Supplementary Fig. 3. Correlation heatmap of QMex descriptors. The QMex descriptors have a limited number of highly correlated combinations due to the exclusion of highly correlated descriptors in the calculation results. The diversity of QMex descriptors enables the prediction of a wide range of molecular properties.



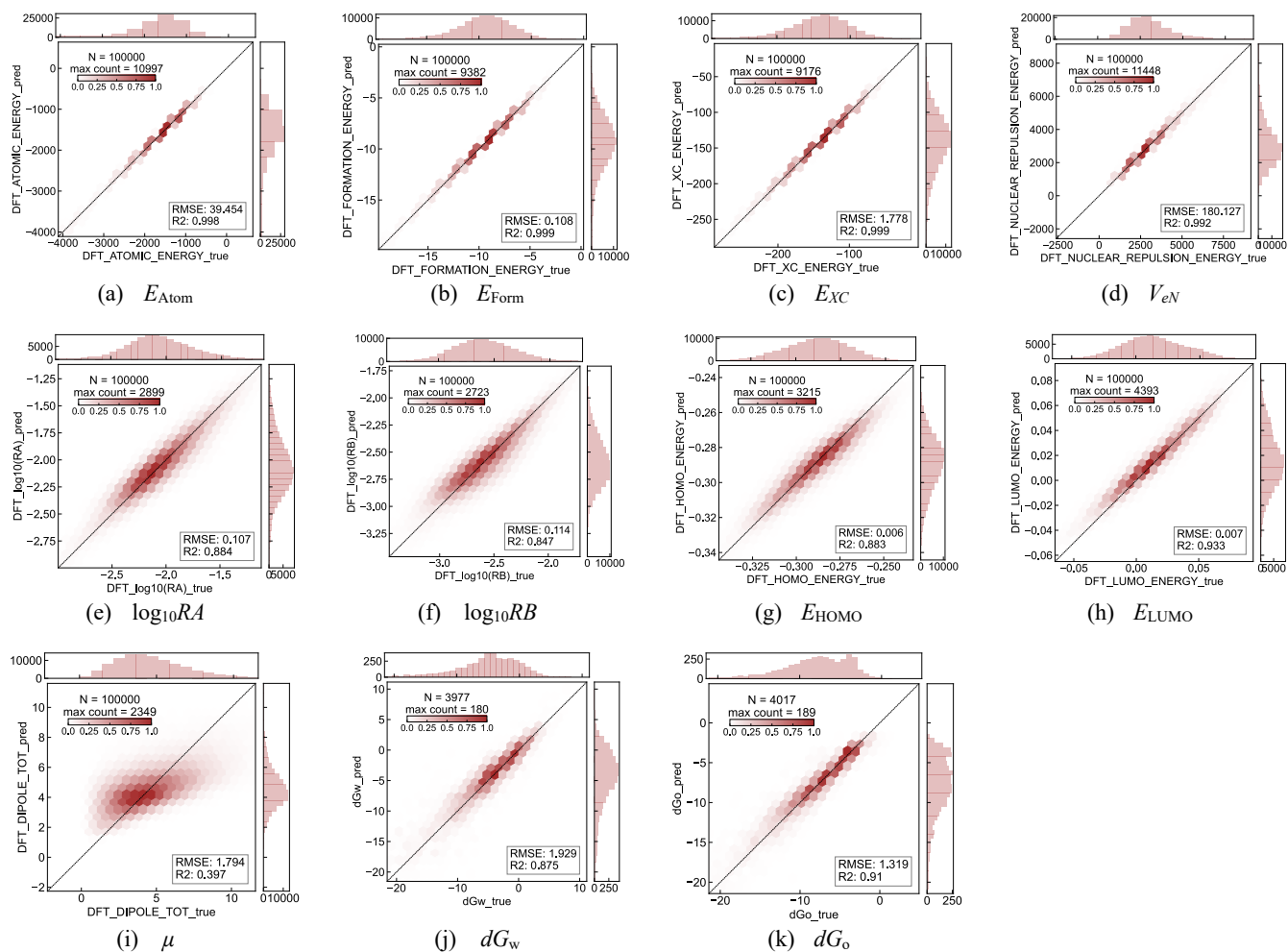
Supplementary Fig. 4. Density distribution between molecular weight (MW) and value of QMugs descriptors in datasets [4,5]. Molecular weight is ranged within 1000 MW, and number of data points N and colormap of density are presented in the upper right. To reduce the computational cost of GIN learning, a random sampling of 100k molecules was performed from DFT dataset of QMugs [4], which originally consisted of 660k molecules.



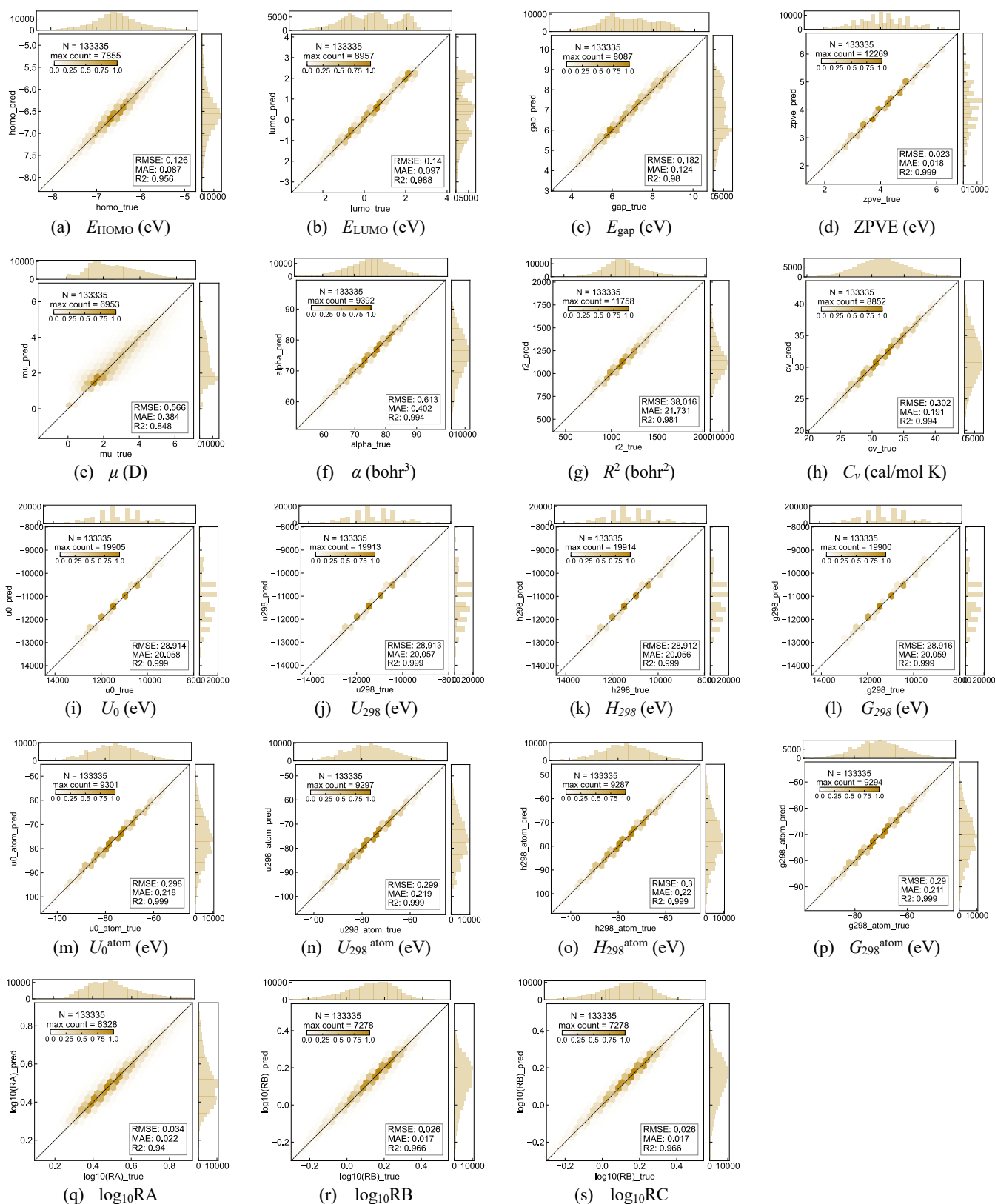
Supplementary Fig. 5. Correlation heatmap of QMugs descriptors. The exclusion of highly correlated descriptors was applied to the DFT dataset of QMugs [4,5]. The lack of correlation indicates that molecules possessing both descriptors were not present in the dataset.



Supplementary Fig. 6. Density distribution of parity plots representing predicted results of QMex descriptors with surrogate-GINs.



Supplementary Fig. 7. Density distribution of parity plots representing predicted results of QMugs descriptors with surrogate-GINs.



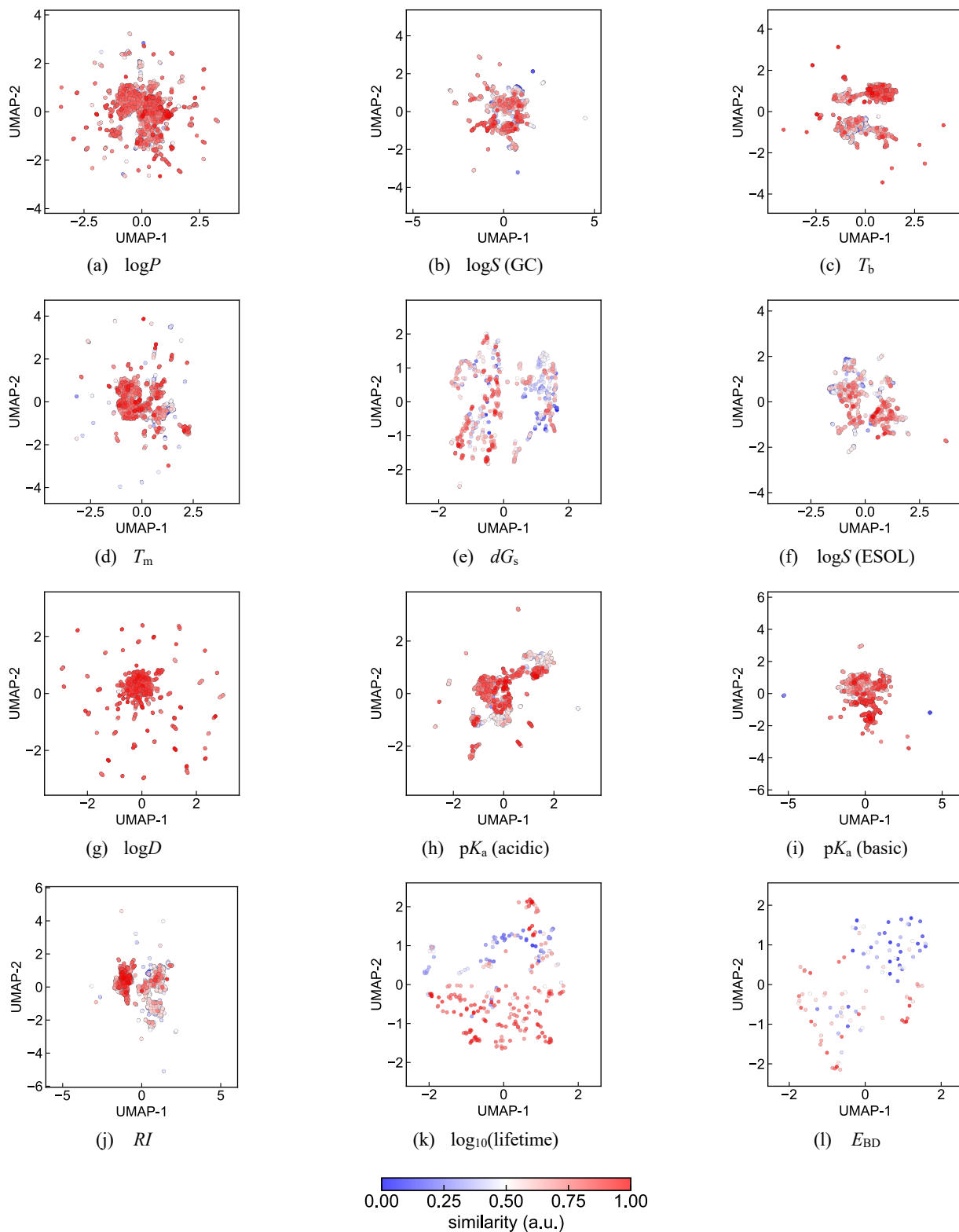
Supplementary Fig. 8. Density distribution of parity plots representing predicted results of QM9 descriptors with surrogate-GINs. MAE is calculated to compare the performance of surrogate-GINs with other benchmark results. The surrogate-GINs achieve the superior performance to MolCLR_{GIN}, that is based on contrastive learning of molecular graphs [7] and the comparable performance to MoLF_{ORMER}, that is based on transformer encoder model of SMILES sequences [8].

Supplementary Table 4. List of rdkit descriptors, chemical categories, and modules for generation.

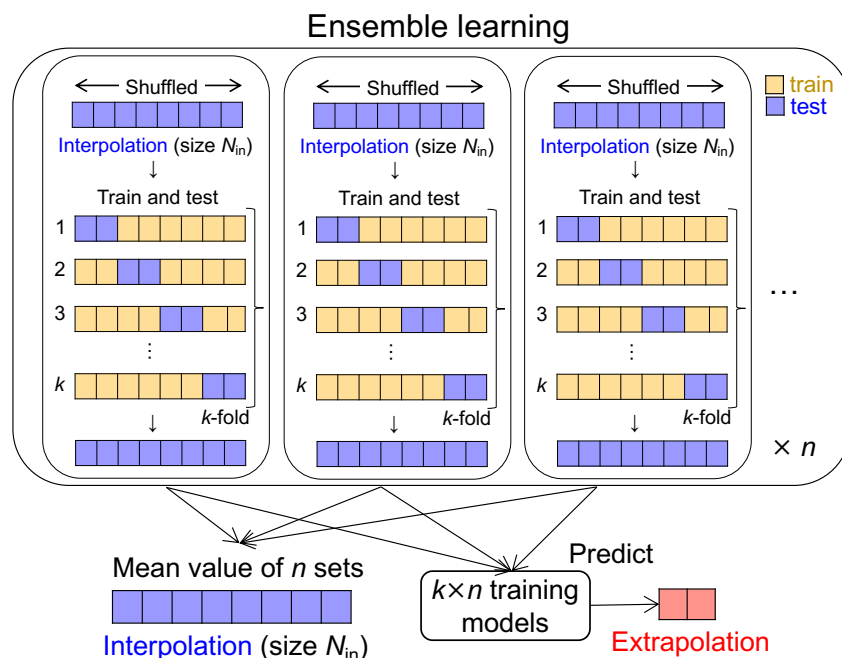
No.	Name	Description	Module list for calculation of chemical categories with Rdkit [9], pybel [10], and pymatgen [11] ([atom], [bond], and [mol] represent Rdkit object. [SMILES] and [SMARTS] represent string input.)
	ECFP	Extended-Connectivity Fingerprint	AllChem.GetMorganFingerprintAsBitVect([mol], radius=2, nBits=256)
	2DFP	2D fingerprint	Chem.Descriptors.descList
Chemical categories			
1	Polar	Has polarity based on point group	<code>mymol=pybel.readstring([SMILES])</code> <code>gif=GaussianInput.from_string(mymol.write('gif'))</code> <code>pg=PointGroupAnalyzer(gif.molecule).sch_symbol</code> (GaussianInput and PointGroupAnalyzer are modules of pymatgen [9]) <code>polar = 1 if pg in</code> <code>[None,'C','C1','C2','C3','Cs','Ci','C*v','C2v','C3v','C4v','C5v','C6v'] else 0</code>
2	Acid	Contains acidic substructures	Using Wu's code on GitHub repository [12, 13]
3	Base	Contains basic substructures	Using Wu's code on GitHub repository [12, 13]
4	With H	Contains H atom	<code>[atom].GetSymbol()</code>
5	With B	Contains B atom	<code>[atom].GetSymbol()</code>
6	With C	Contains C atom	<code>[atom].GetSymbol()</code>
7	With N	Contains N atom	<code>[atom].GetSymbol()</code>
7	With O	Contains O atom	<code>[atom].GetSymbol()</code>
8	With F	Contains F atom	<code>[atom].GetSymbol()</code>
9	With Si	Contains Si atom	<code>[atom].GetSymbol()</code>
10	With P	Contains P atom	<code>[atom].GetSymbol()</code>
11	With S	Contains S atom	<code>[atom].GetSymbol()</code>
12	With Cl	Contains Cl atom	<code>[atom].GetSymbol()</code>
13	With Br	Contains Br atom	<code>[atom].GetSymbol()</code>
14	With I	Contains I atom	<code>[atom].GetSymbol()</code>
15	Argon	Contains He, Ne, Ar, Kr, Xe, or Rn	<code>[atom].GetSymbol()</code>
16	Halogen	Contains F, Cl, Br, I, or At	<code>[atom].GetSymbol()</code>
17	Other	Contains Se, As, or Te	<code>[atom].GetSymbol()</code>
18	Metal	Contains metal atoms (other than those listed above)	<code>[atom].GetSymbol()</code>
19	Cation	Is cation	<code>[atom].GetFormalCharge()</code>
20	Anion	Is anion	<code>[atom].GetFormalCharge()</code>
21	SINGLE	Contains single bonds	<code>[bond].GetBondType()</code>
22	DOUBLE	Contains double bonds	<code>[bond].GetBondType()</code>
23	TRIPLE	Contains triple bonds	<code>[bond].GetBondType()</code>
24	AROMATIC	Is aromatic	<code>[bond].GetBondType()</code>
25	Pi	Contains pi electron	<code>fp = Pairs.GetAtomPairFingerprint([mol])</code> <code>[Pairs.ExplainPairScore(k) for k in fp.GetNonzeroElements().keys()]</code>
26	Conjugated	Is conjugated	<code>[bond].GetIsConjugated()</code>
27	Ring	Contains ring structures	<code>[mol].GetRingInfo().NumRings()</code>
28	Branch	Contains branch structures	<code>[mol].count("(")</code>
29	Rot	Contains rotatable bonds	<code>AllChem.CalcNumRotatableBonds([mol])</code>
30	Spiro	Contains spiro structures	<code>AllChem.CalcNumSpiroAtoms([mol])</code>
31	HBA	Contains hydrogen bond acceptors	<code>AllChem.CalcNumHBA([mol])</code>
32	HBD	Contains hydrogen bond donors	<code>AllChem.CalcNumHBD([mol])</code>
33	O single	Contains X-O-X groups ("*O*")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
34	O double	Contains X=O groups ("*=O")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
35	N triple	Contains X#N groups ("*#N")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
36	Amine	Contains amine groups ("([NX3;H2,H1,H0;!\$(NC=[!#6])])")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
37	Alcohol	Contains alcohol groups ("([OX2H])([!#6X4])")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
38	Ether	Contains ether groups ("([OD2])([!#6X4])[!#6X4])")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
39	Cyano	Contains cyano groups ("C#N")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
40	Carboxylic	Contains carboxylic acids ("([#6]([CX3](=O)[OX2H1])")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
41	Carbonyl	Contains carbonyl group ("([CX3]=[OX1]),\$(CX3+)-[OX1-])")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>
42	Amide	Contains amide groups ("([NX3][CX3](=[OX1])[!#6])")	<code>[mol].HasSubstructMatch(Chem.MolFromSmarts([SMARTS]))</code>

Supplementary Table 5. List of hyperparameters of ML/DL models.

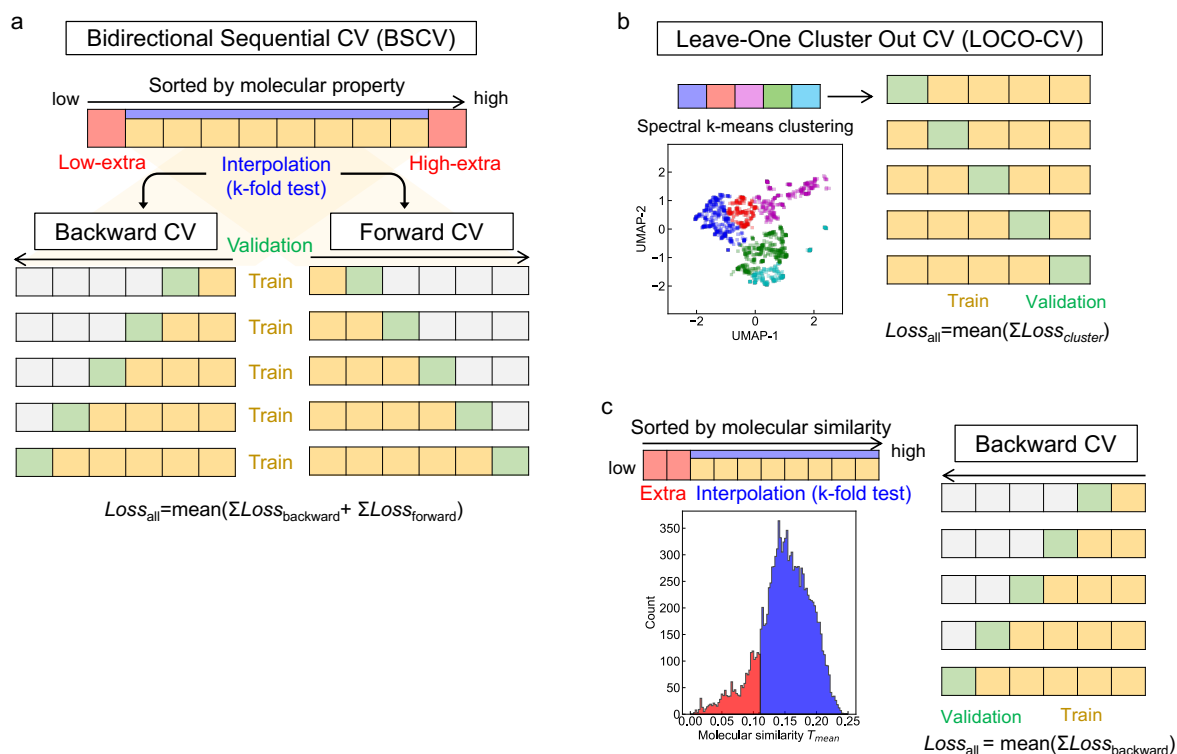
ML/DL model	Parameter	Value, list, or range of parameter (List and range are used for optimization.)	Method of optimization
PLS (QMex-LR, QMugs-LR)	<i>n_components</i>	[1, 2, 3, 4, 5, 6, 7, 8]	Optuna [14] with 10 trials
PLS (Mol-GCR)	<i>n_components</i>	[1, 2, 3, 4, 6, 8, 10, 12, 16, 20, 24, 28, 32, 36]	Optuna [14] with 14 trials
PLS (QMex-ILR, QMugs-ILR)	<i>n_components</i>	[1, 2, 3, 4, 6, 8, 10, 12, 16, 20, 24, 28, 32, 36]	Optuna [14] with 56 trials
	<i>degree</i>	[-2, -1, 0, 1]	
KRR (QM-NLR, QMugs-NLR, 2DFP-NLR, ECFP-NLR)	<i>alpha</i>	10^{-6} – 10^1 (log scale)	Optuna [14] with 50 trials
	<i>gamma</i>	10^{-5} – 10^0 (log scale)	
	<i>kernel</i>	‘rbf’	
GCN, GIN (Mol-GCN, Mol-GIN)	<i>GNN layer num.</i>	[2, 4]	Brute force search of $2*2*3*3*2*3 = 216$ cases
	<i>GNN emb. dim.</i>	[50, 100]	
	<i>Pool</i>	['mean', 'sum', 'max']	
	<i>MLP layer num.</i>	[0, 1, 2]	
	<i>Epoch</i>	[50, 200]	
	<i>Learning rate</i>	$[2\times 10^{-3}, 5\times 10^{-4}, 1\times 10^{-4}]$	
GIN (QM-surrogate models for QMex descriptors)	<i>GNN layer num.</i>	[2, 4]	Brute force search of $2*3*3*2*1*3 = 108$ cases
	<i>GNN emb. dim. (N_{emb})</i>	[50, 100, 200]	
	<i>Pool</i>	['mean', 'sum', 'max']	
	<i>MLP layer num.</i>	[1, 2]	
	<i>Epoch</i>	[200]	
	<i>Learning rate</i>	$[5\times 10^{-2}, 1\times 10^{-2}, 2\times 10^{-3}]$	
GIN (QM-surrogate models for descriptors in QMugs dataset [4])	<i>GNN layer num.</i>	[3, 6]	Brute force search of $2*2*3*3*2*1*2 = 48$ cases
	<i>GNN emb. dim. (N_{emb})</i>	[100, 500]	
	<i>Pool</i>	['mean', 'sum', 'max']	
	<i>MLP layer num.</i>	[2, 4]	
	<i>Epoch</i>	[200]	
	<i>Learning rate</i>	$[1\times 10^{-2}, 2\times 10^{-3}]$	
GIN (QM-surrogate models for descriptors in Vermeire’s dataset [5])	<i>GNN layer num.</i>	[2, 4]	Brute force search of $2*2*3*3*2*4 = 288$ cases
	<i>GNN emb. dim. (N_{emb})</i>	[100, 200]	
	<i>Pool</i>	['mean', 'sum', 'max']	
	<i>MLP layer num.</i>	[0, 2, 4]	
	<i>Epoch</i>	[50, 200]	
	<i>Learning rate</i>	$[1\times 10^{-2}, 2\times 10^{-3}, 5\times 10^{-4}, 1\times 10^{-4}]$	
GCN, GIN (Mol-GCN, Mol-GIN, QM-surrogate models)	<i>MLP layer dim.</i>	(N_{emb} ,), (N_{emb} , $N_{emb}/2$,), and (N_{emb} , $N_{emb}/2$, $N_{emb}/4$,) when <i>MLP layer num.</i> is 0, 1, and 2, respectively, where N_{emb} represents <i>GNN emb. dim.</i> The last number of dimensions works as number of input dimension for fully-connected layer just before output values.	Fixed or dependent on other values
	<i>Activation</i>	‘relu’	
	<i>Batch size</i>	$\min(200, 0.1*N_{train})$	
	<i>Drop ratio</i>	0.10	
	<i>Weight decay</i>	10^{-6}	



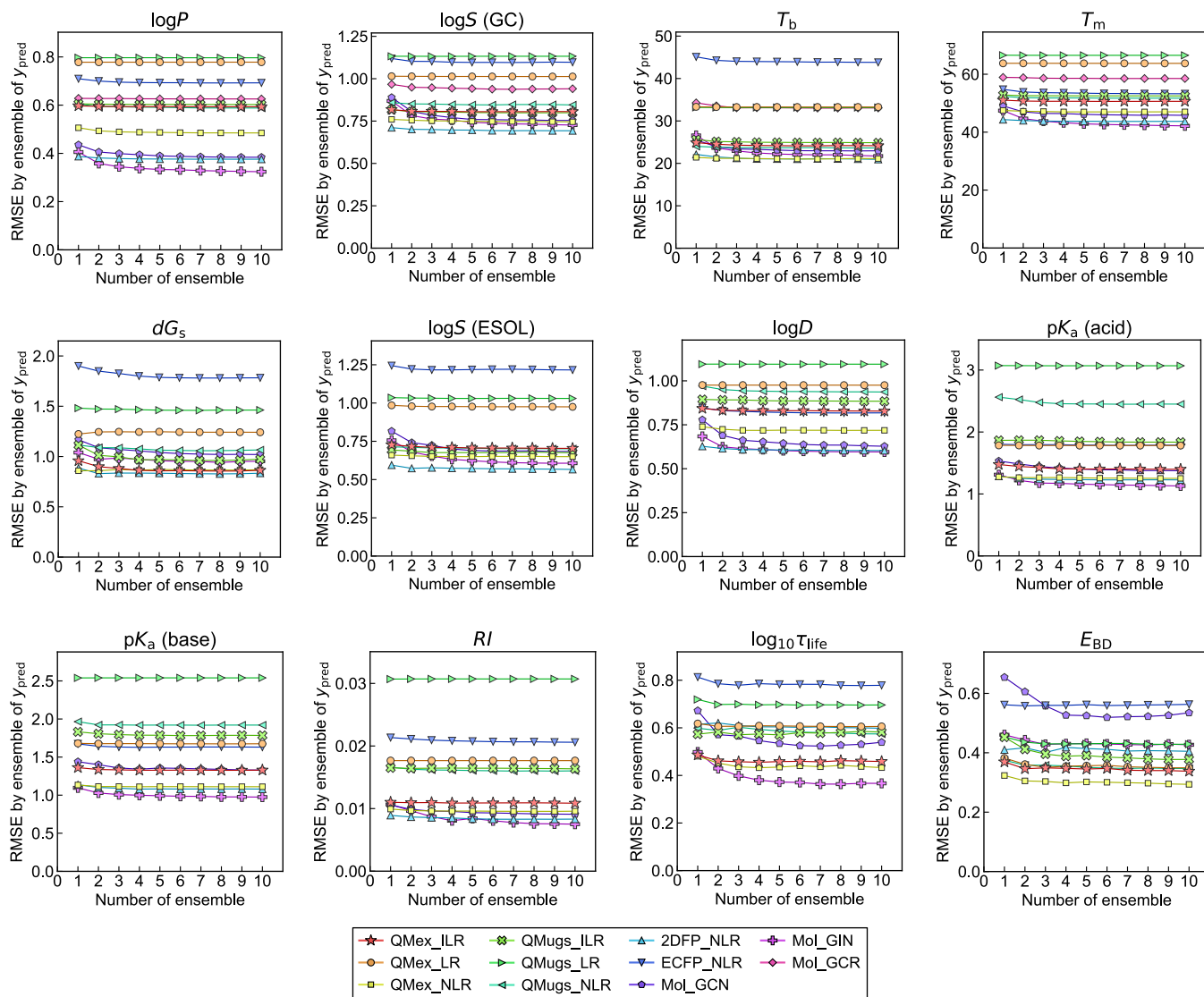
Supplementary Fig. 9. 2D-UMAP distribution of each dataset used for cluster extrapolation. 2D-UMAP components were calculated using parameters of $n_components=2$, $n_neighbors=15$, $densmap=True$, and $dens_lambda=5.0$ to extract local similarity of molecular structures while preserving density information in high dimensional spaces of input data. Input data consisted of 64-dimensional PCA components of 2048-bit ECFP (radius=2) of molecular structures. Colormap represents molecular similarity calculated by mean value of Tanimoto index, which aims to measure global similarity of molecular structures. Molecules with high similarity are distributed not only in dense regions but also in sparse regions, which indicates a discrepancy between the local similarity based on the density of 2D-UMAP, evaluated through cluster extrapolation, and the global similarity evaluated by similarity extrapolation.



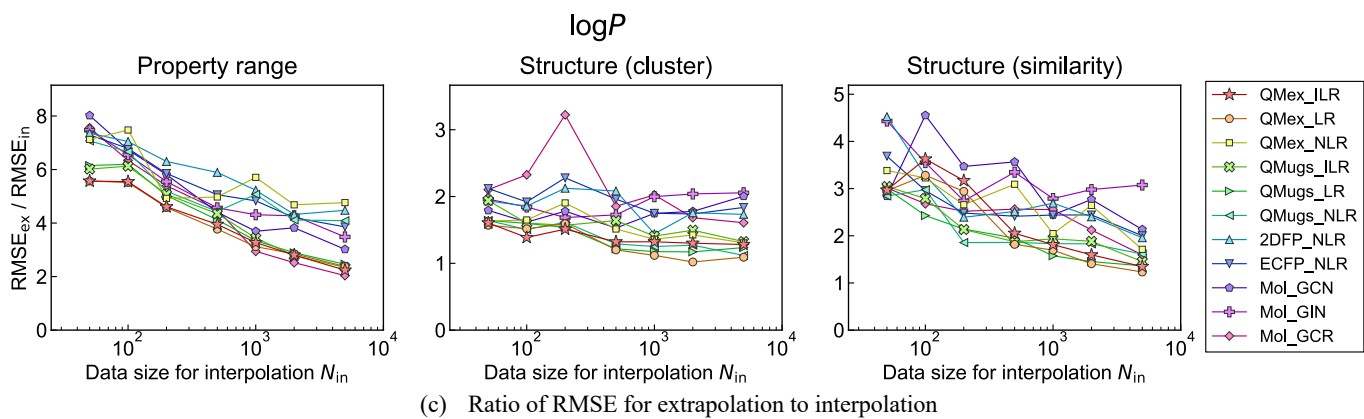
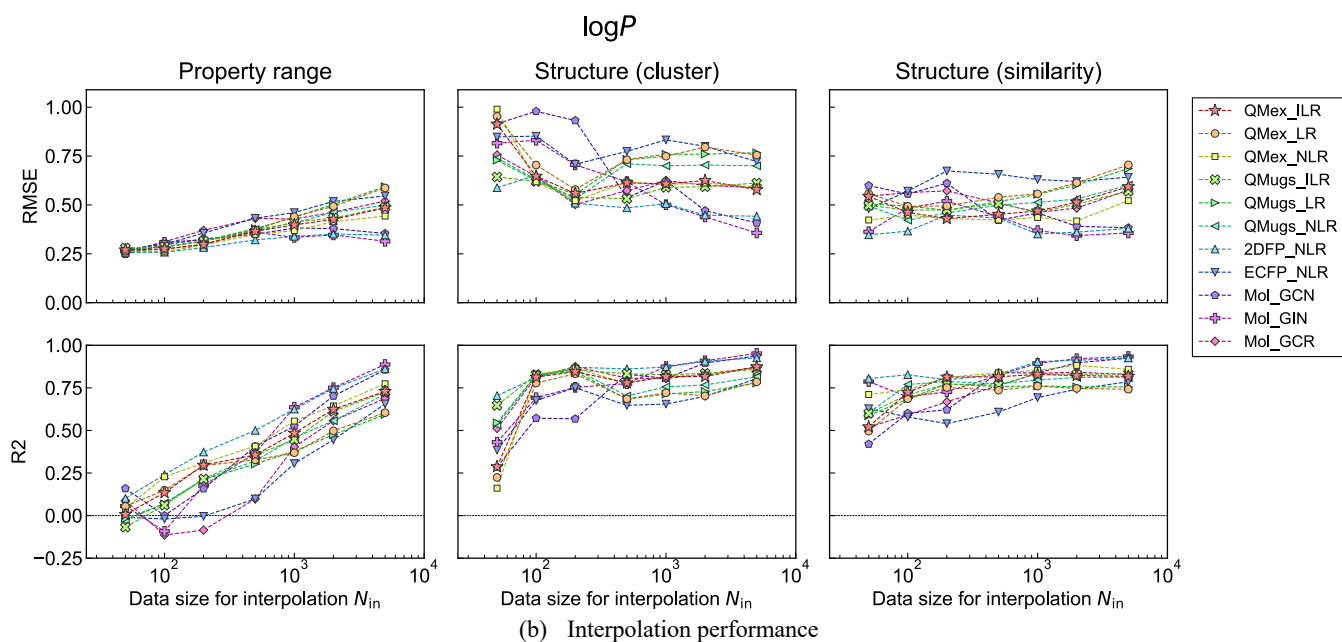
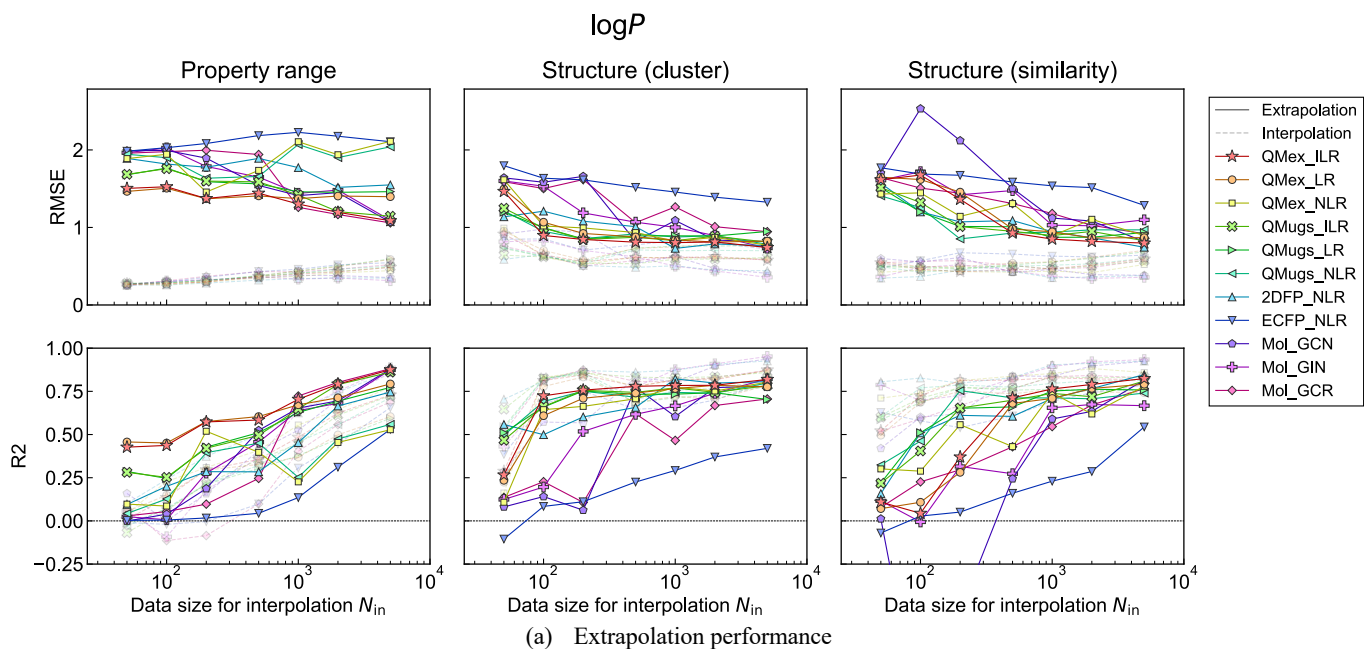
Supplementary Fig. 10. Ensemble learning scheme to reduce ML/DL accuracy bias. For each learning, interpolation data is shuffled and used for k -fold test, resulting k sets of training models and one set of test values. Interpolation performance is evaluated by mean value of n sets of test values, and extrapolation performance is evaluated by mean value of $k \times n$ sets of predicted values of extrapolation data. In this study, the ensemble iteration n and test folds k were set to $k=5$ and $n=10$ for training surrogate GINs and evaluation of interpolation and extrapolation.



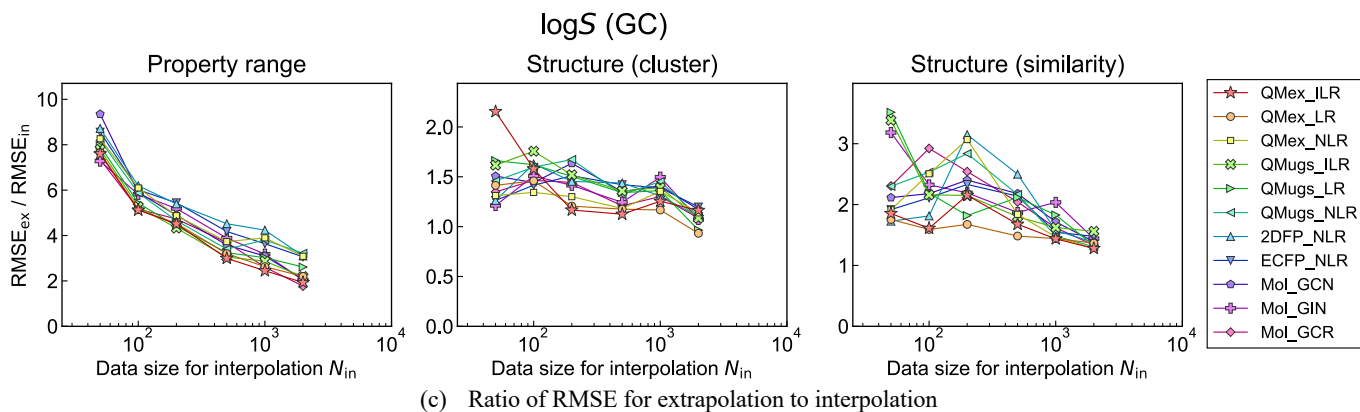
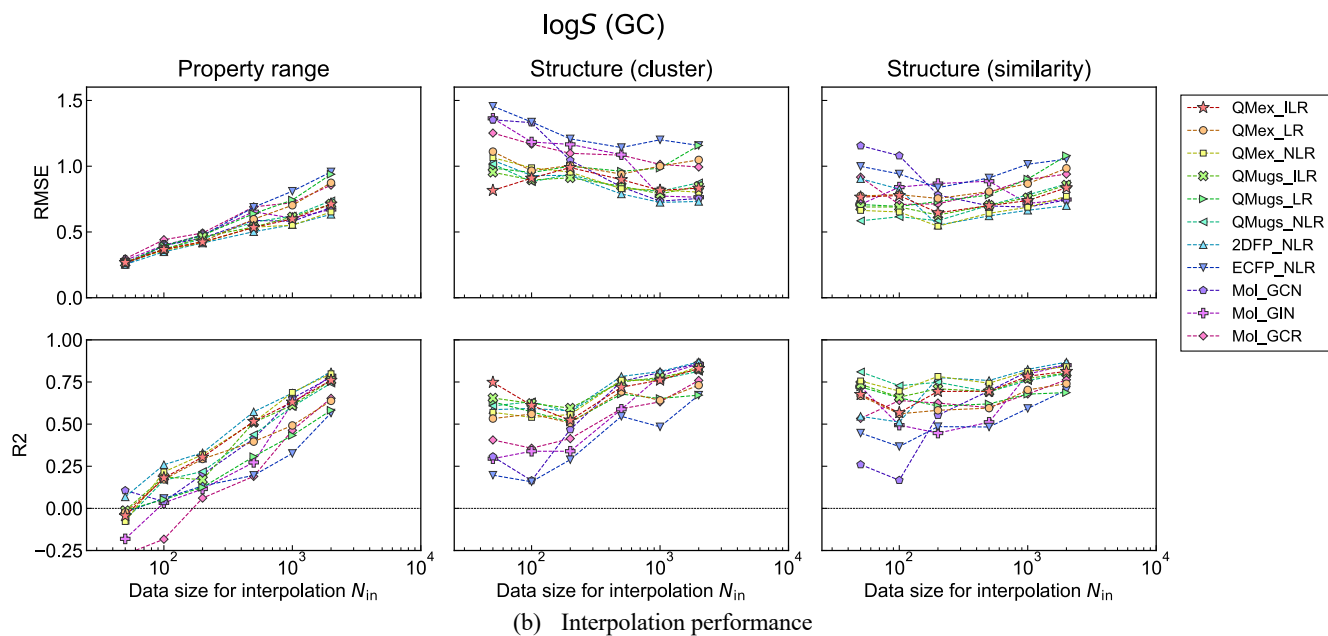
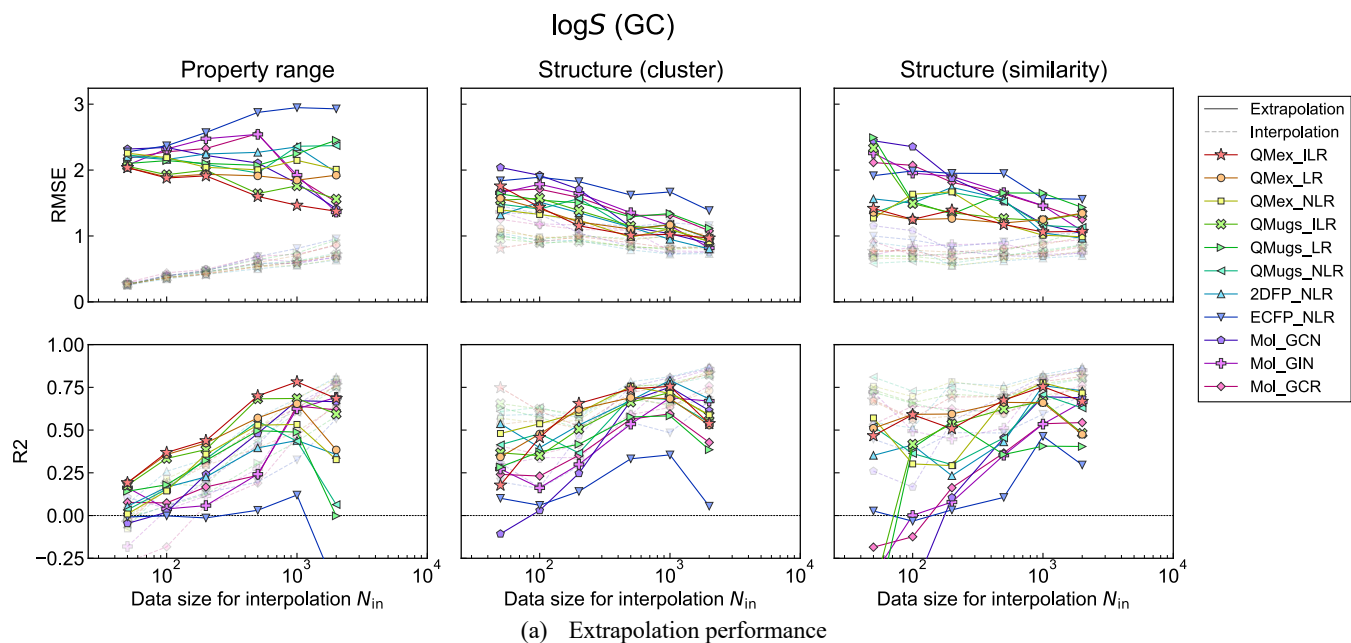
Supplementary Fig. 11. Description of validation methods designed for property, cluster, and similarity extrapolation, respectively. a, Bidirectional Sequential Cross-Validation (BSCV) designed for improving performance of property extrapolation. During k -fold test, a training data is sorted by target property and separated into k -fold backward CV and forward CV datasets, respectively. Validation loss is calculated by mean of $2k$ sets of losses of backward CV and forward CV. b, Leave One Cluster Out Cross-Validation (LOCOCV) designed for improving performance of cluster extrapolation. During k -fold test, a training data is separated into k sets of clusters for CV datasets by using spectral k -means clustering on UMAP. Validation loss is calculated by mean of k sets of losses. c, Backward CV designed for improving performance of similarity extrapolation. During k -fold test, a training data is sorted by molecular similarity and separated into k -fold backward CV datasets. Validation loss is calculated by mean of k sets of losses of backward CV.



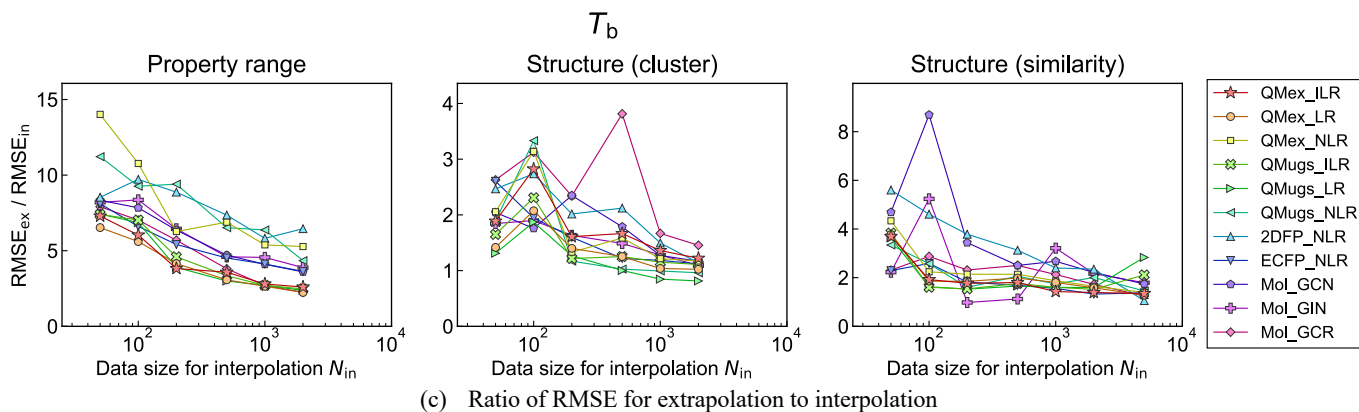
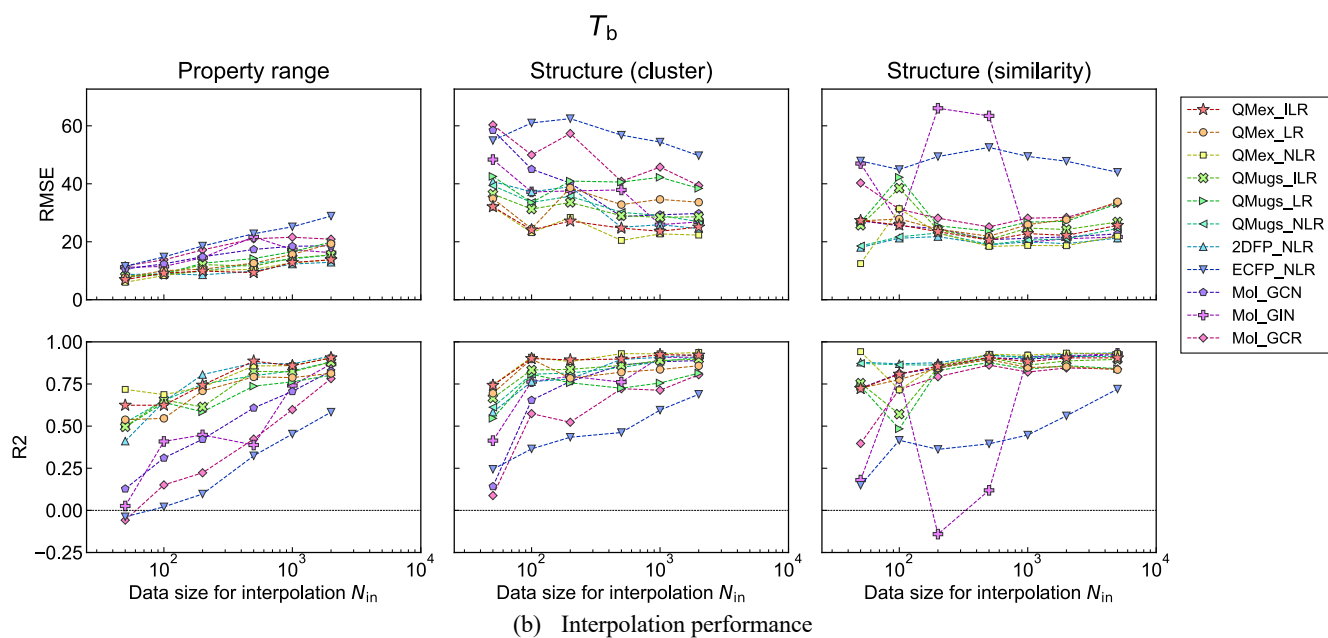
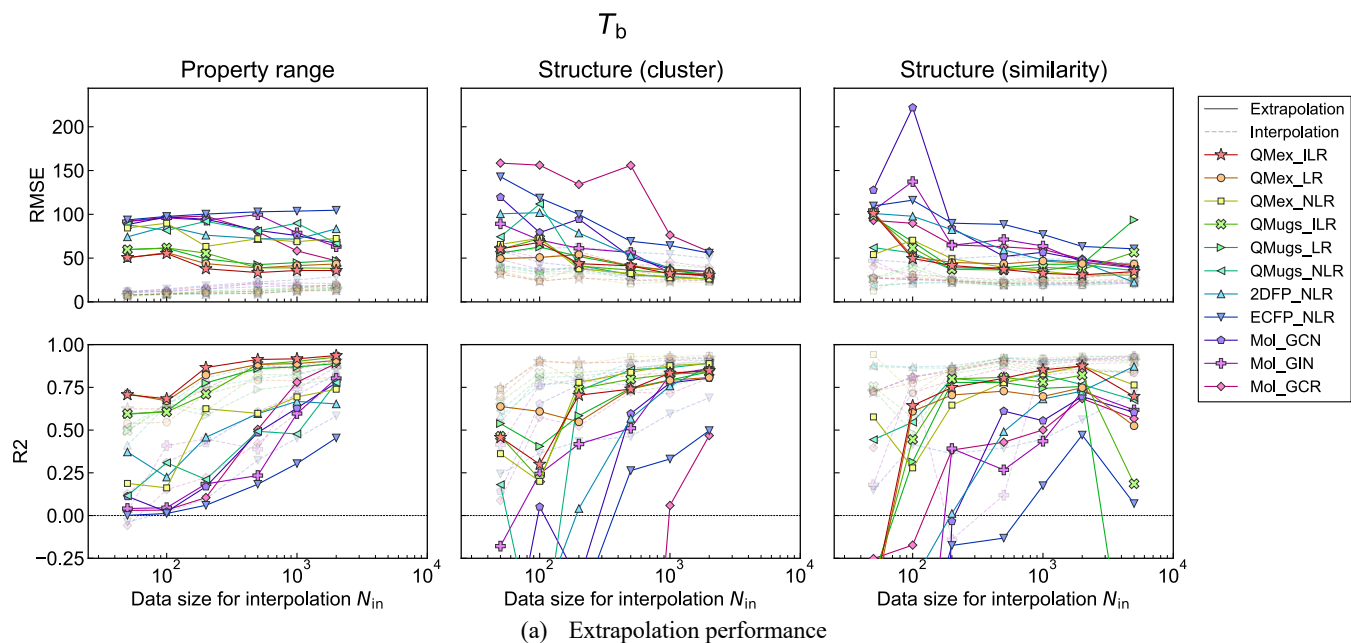
Supplementary Fig. 12. Variation of RMSE during ensemble learning of interpolation test using all data points for each property dataset. Each RMSE was calculated by the mean values of predicted values up to each ensemble iteration. It is observed that the RMSE saturates after ten iterations, suggesting that ten iterations are sufficient for ensemble learning to reproduce the final predicted values.



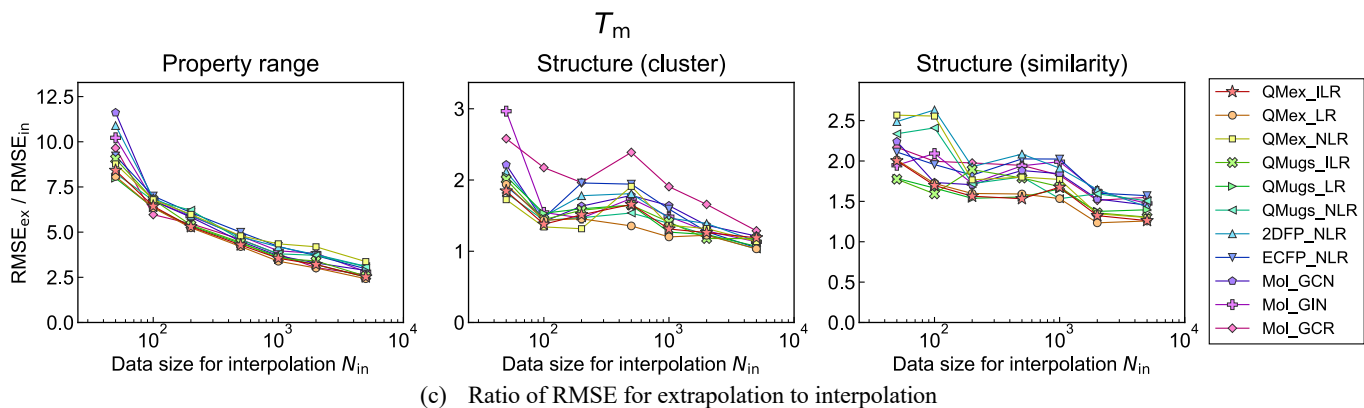
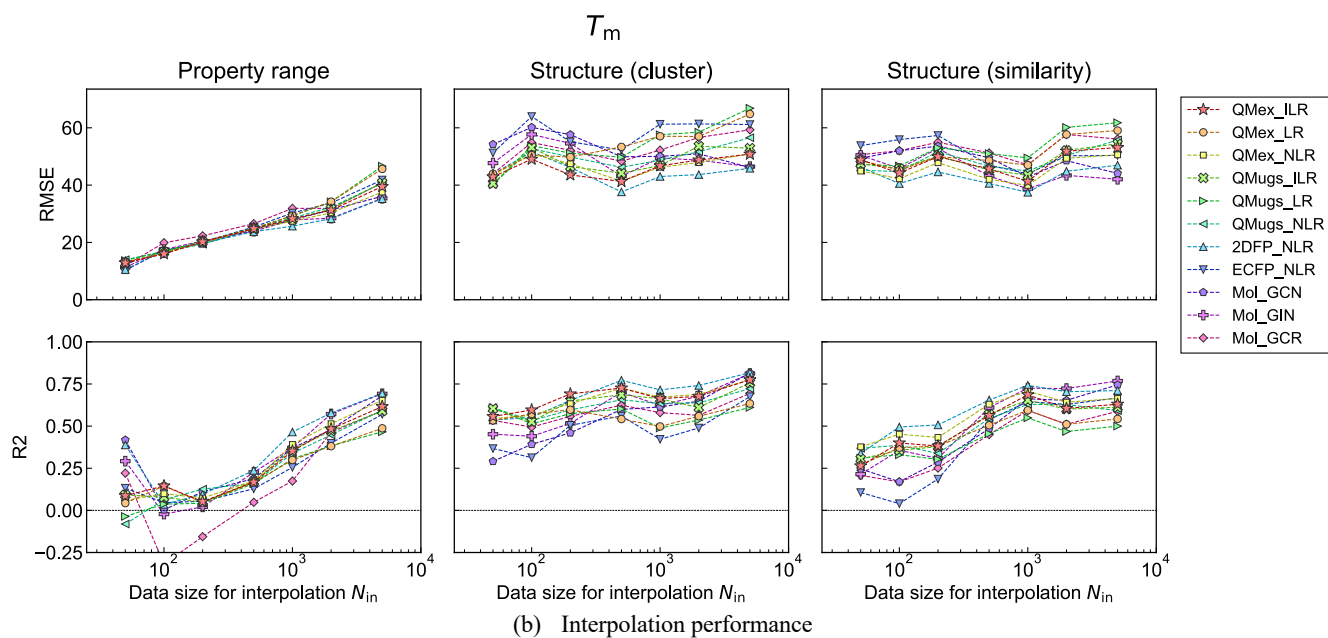
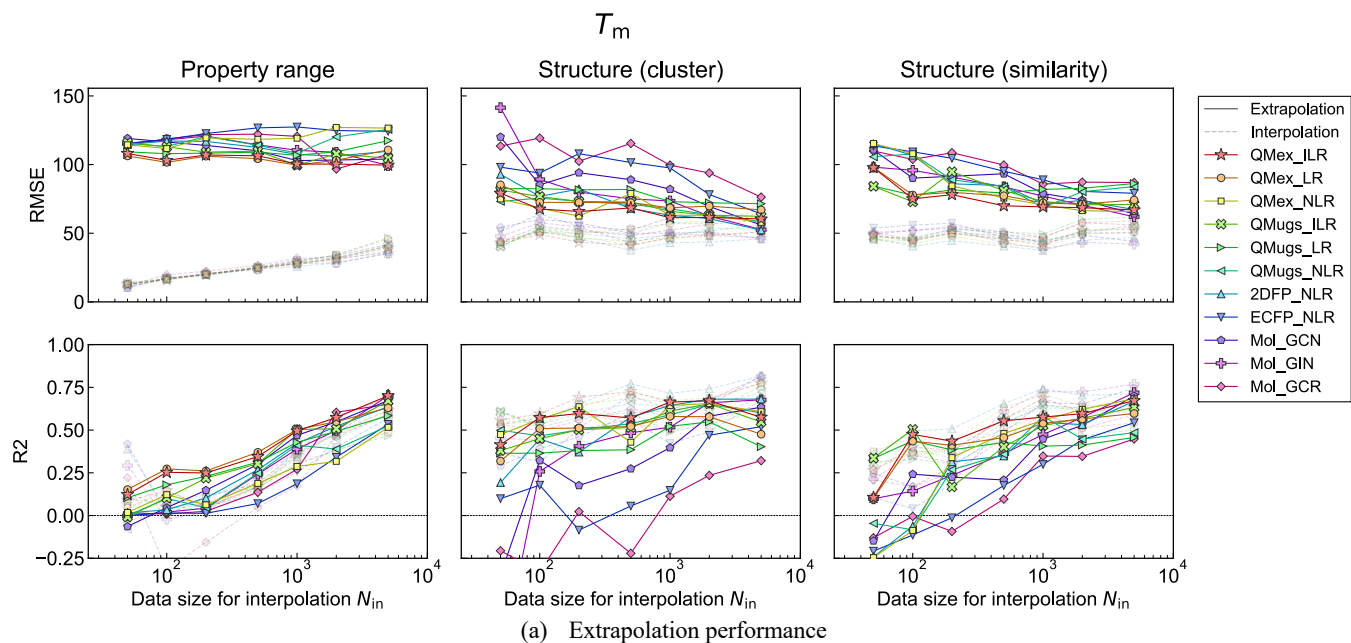
Supplementary Fig. 13. Benchmark result of $\log P$.



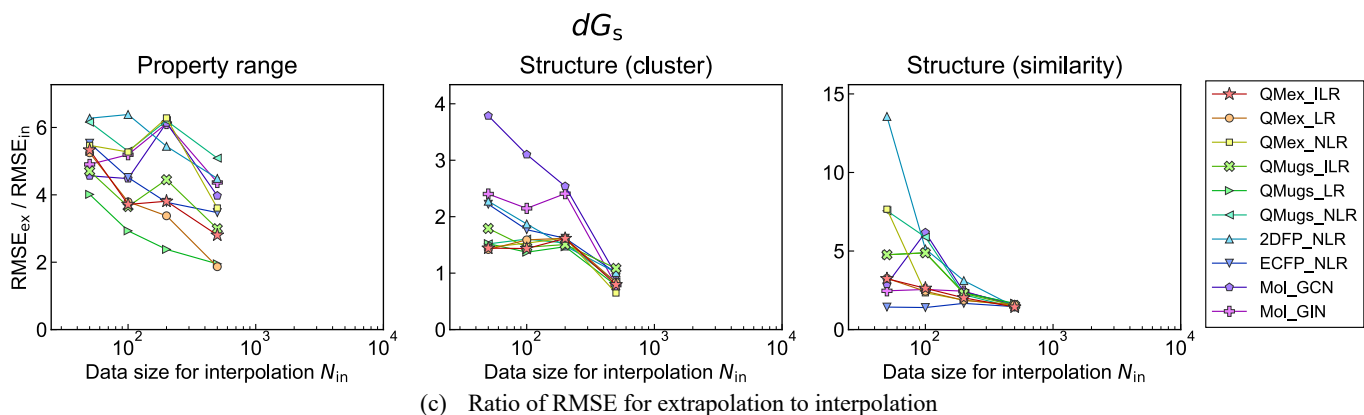
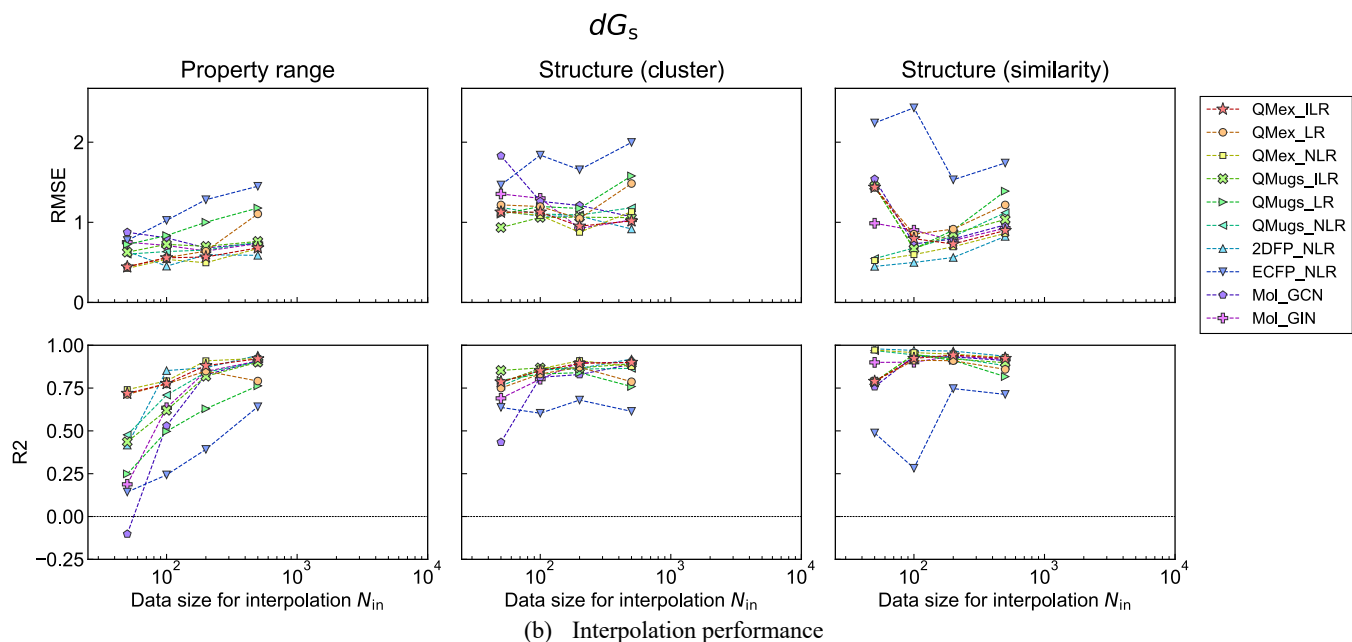
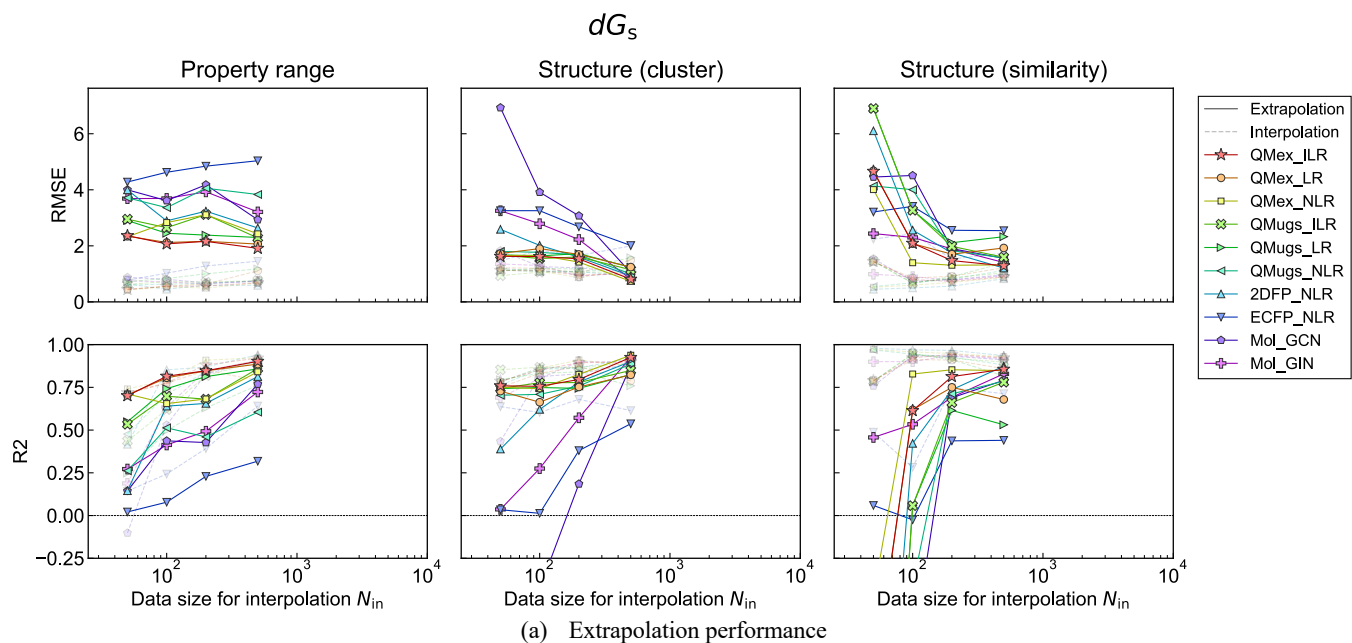
Supplementary Fig. 14. Benchmark result of logS (GC).



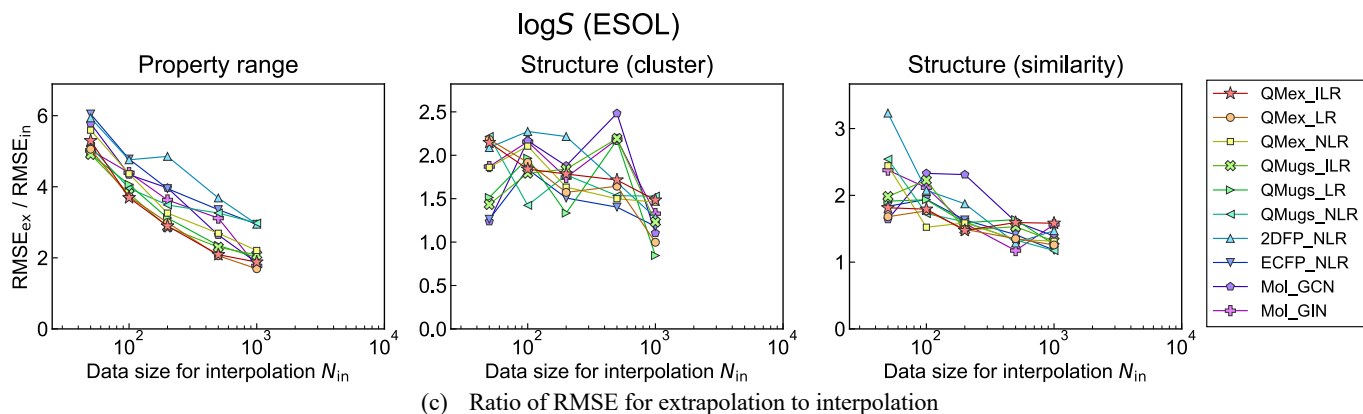
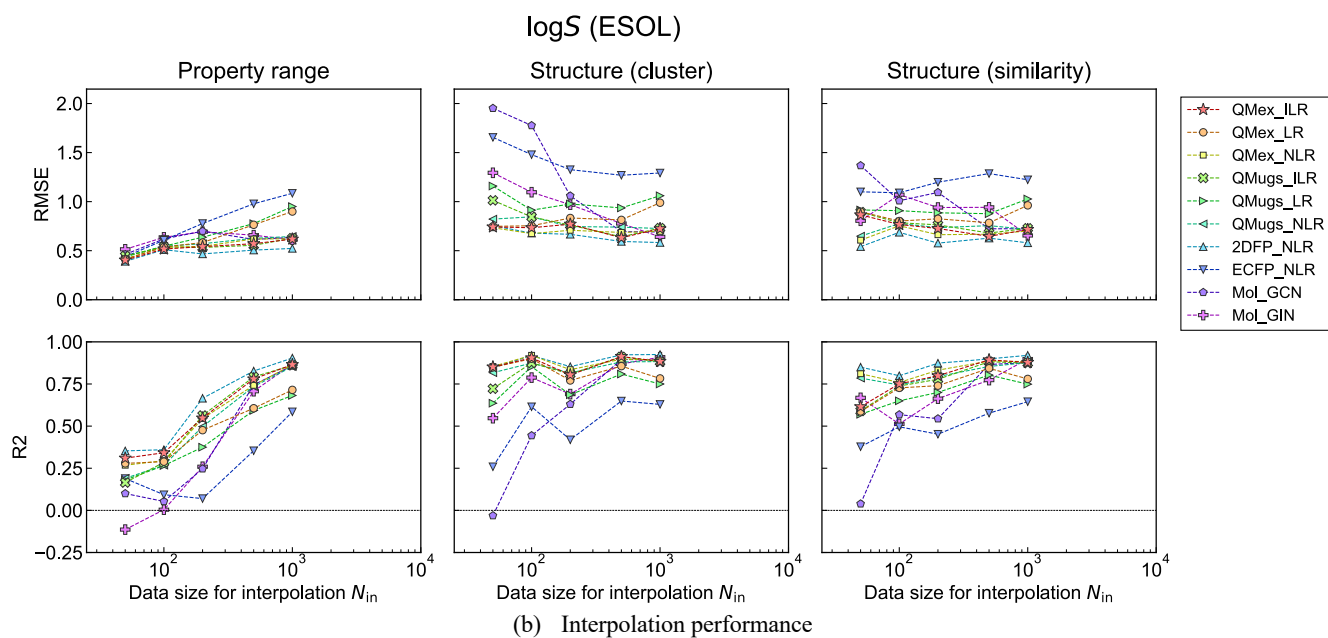
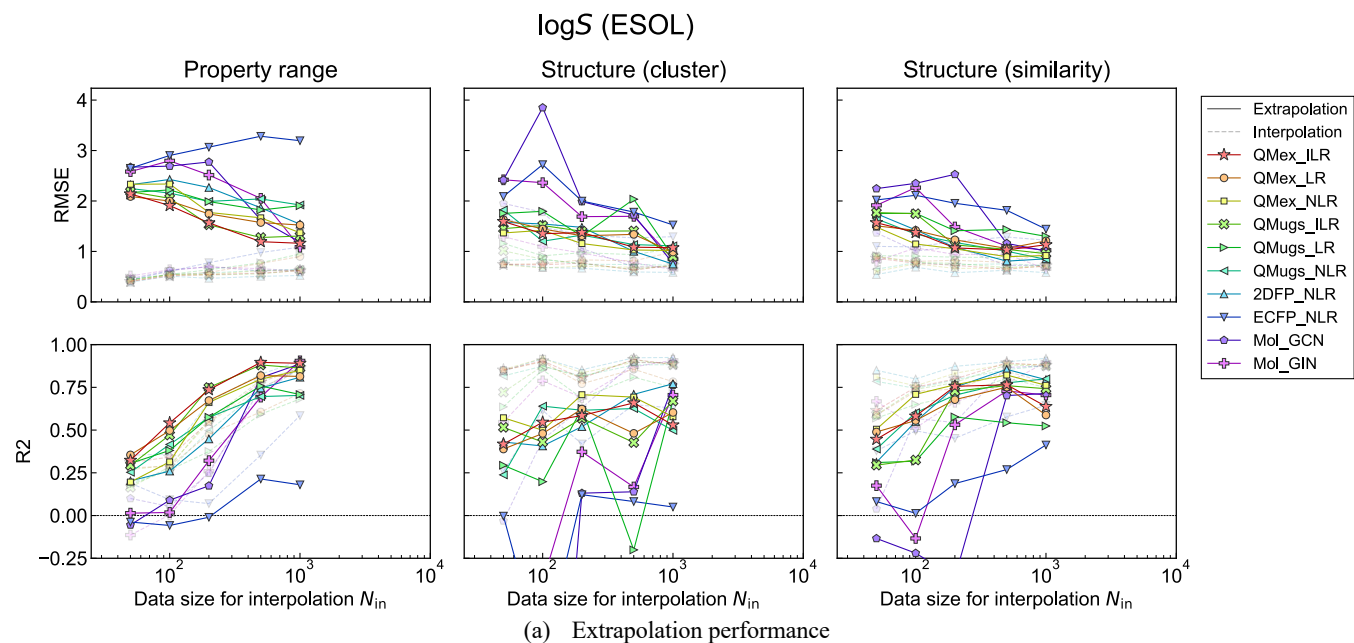
Supplementary Fig. 15. Benchmark result of T_b .



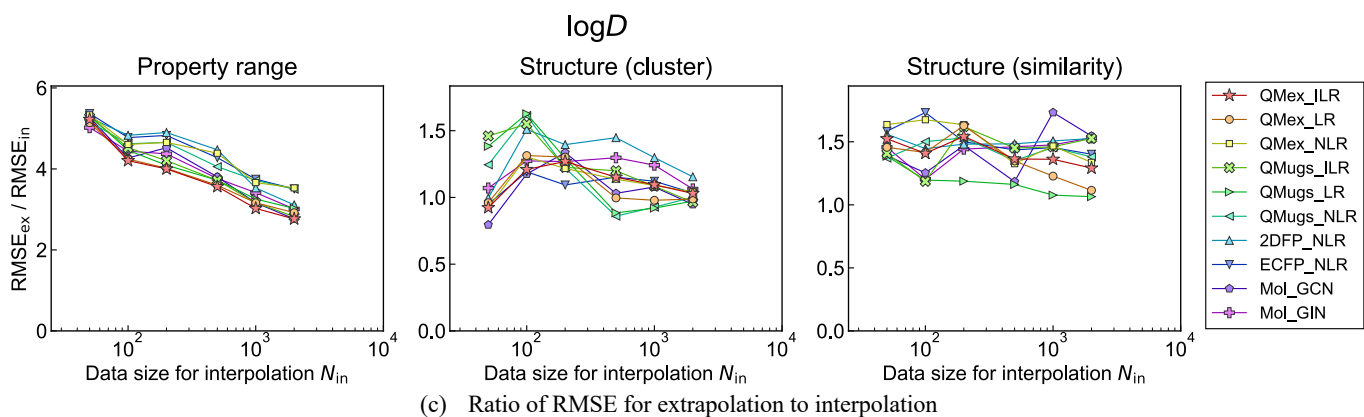
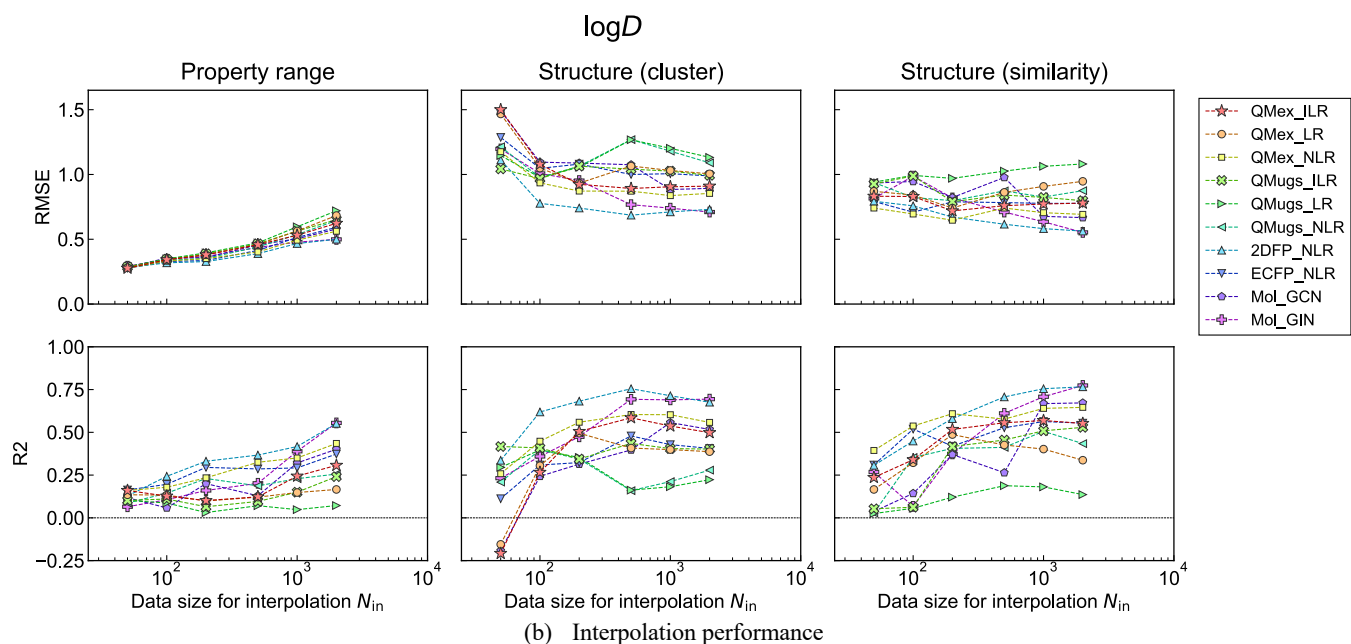
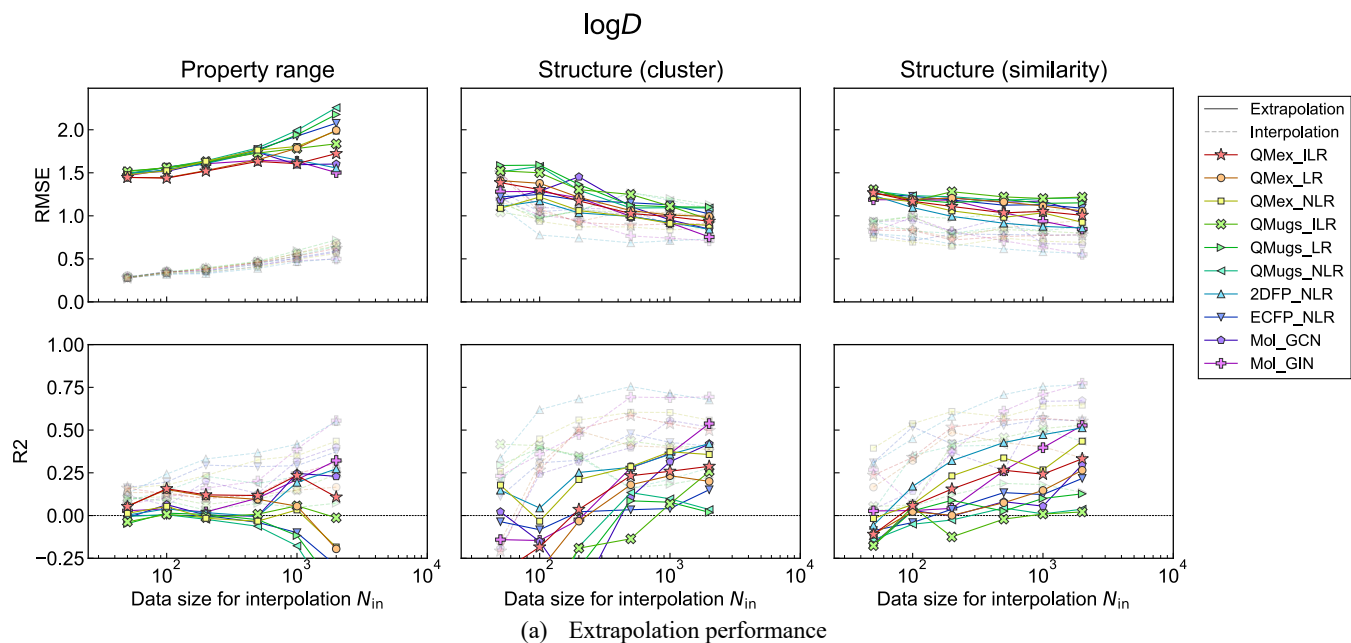
Supplementary Fig. 16. Benchmark result of T_m .



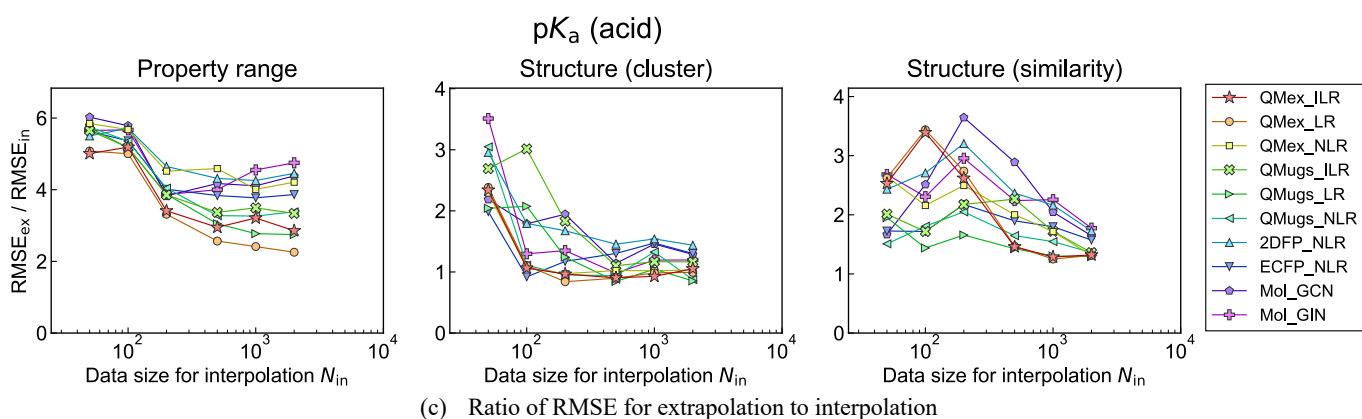
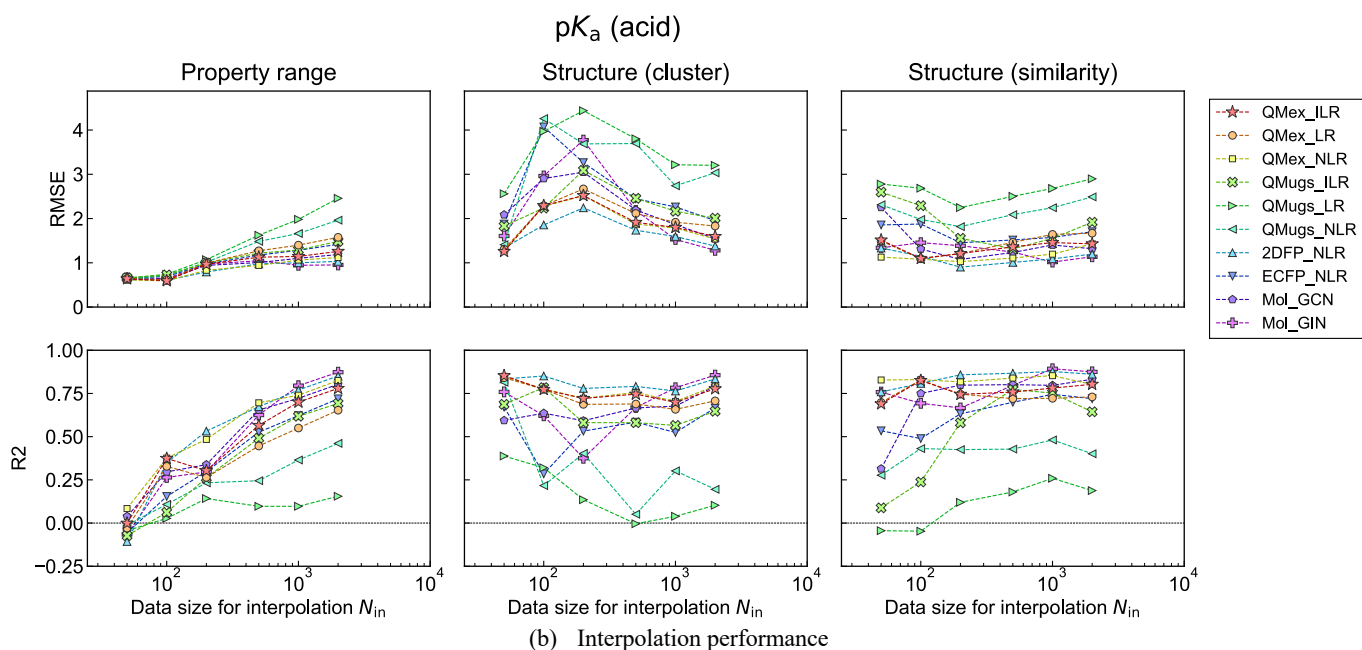
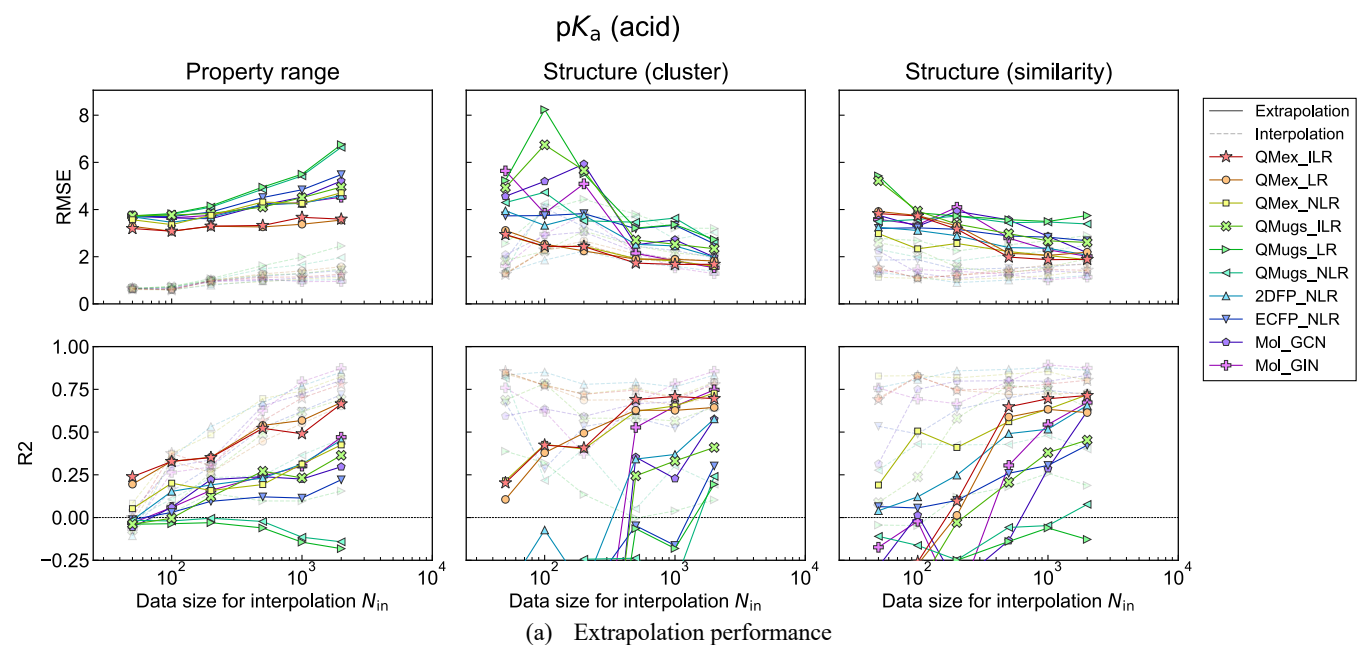
Supplementary Fig. 17. Benchmark result of dG_s .



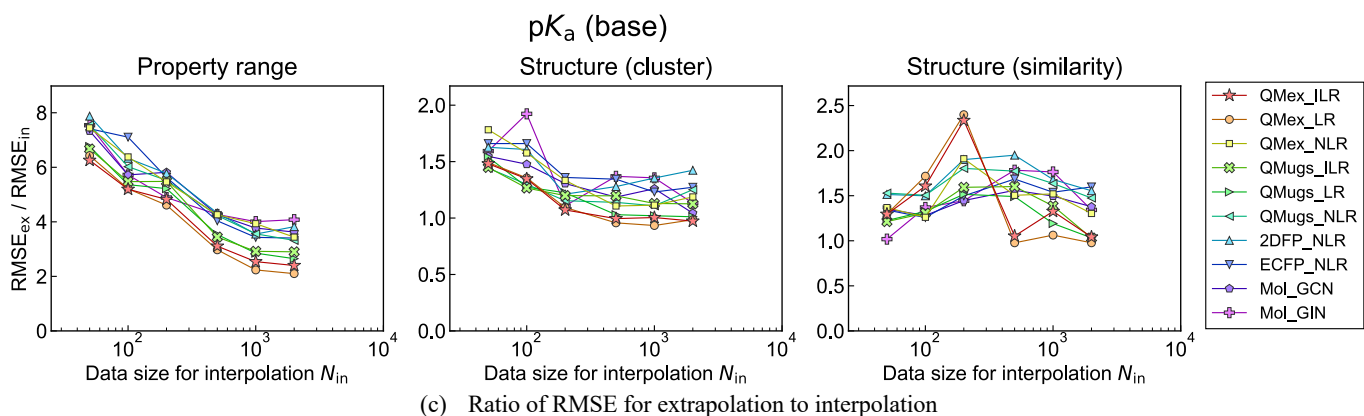
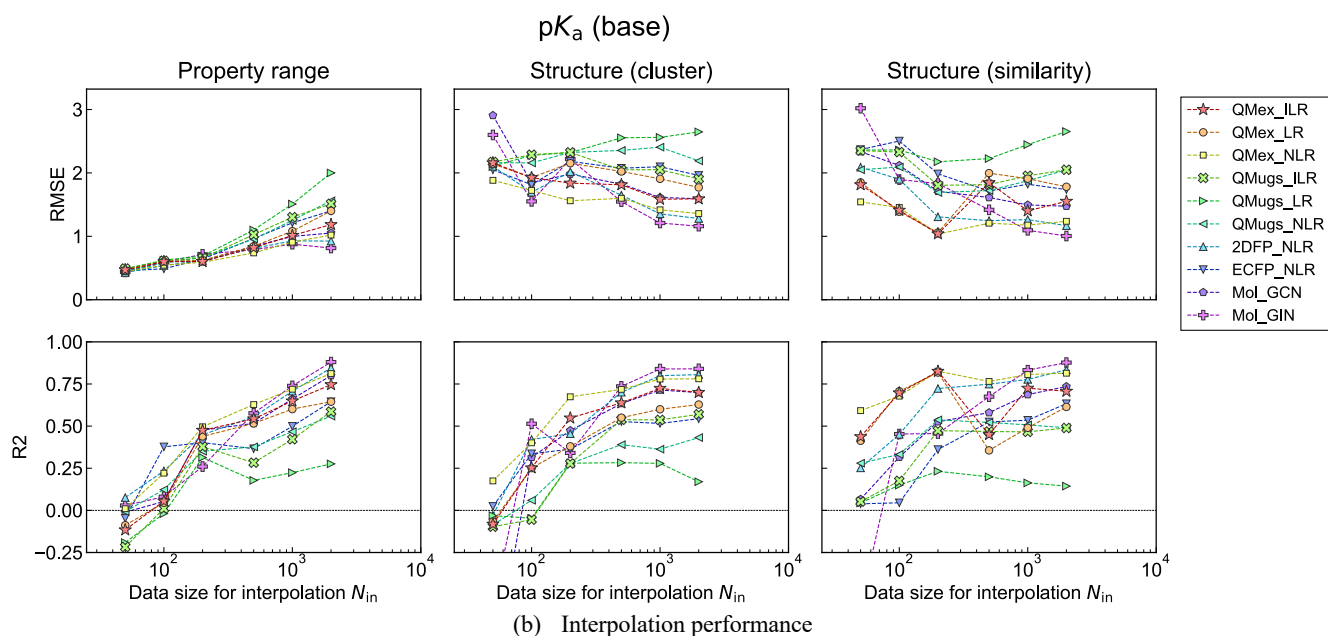
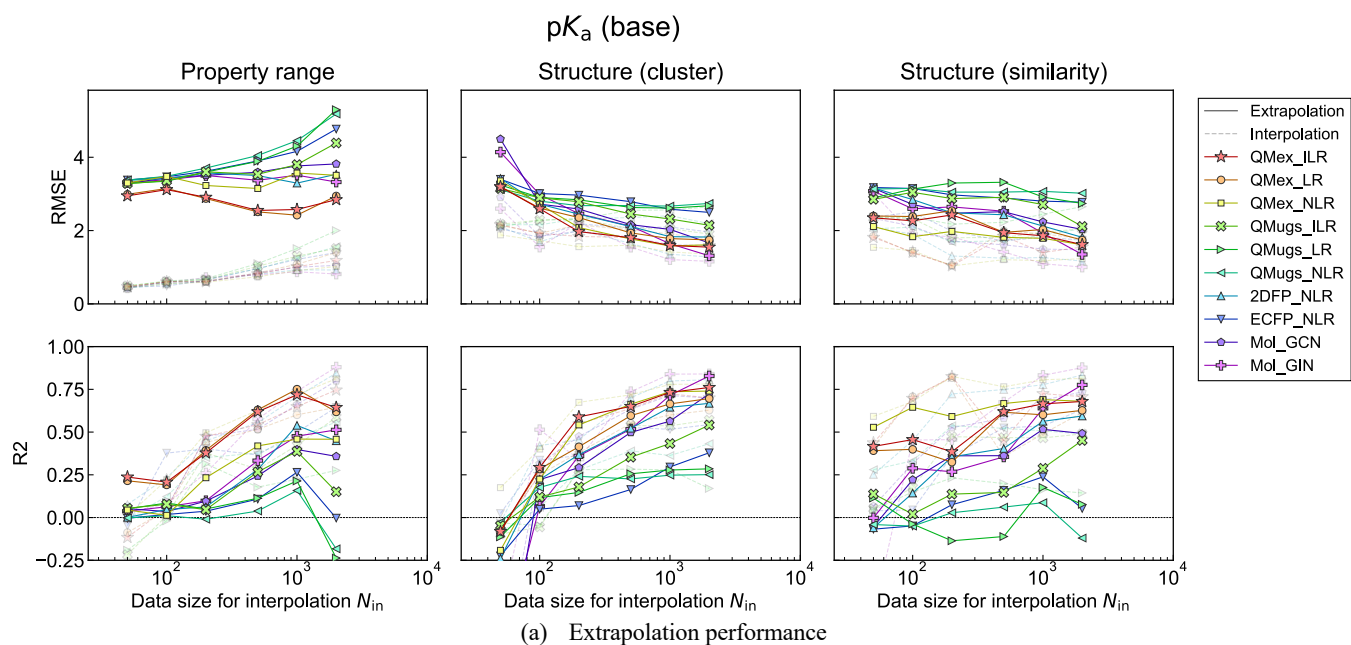
Supplementary Fig. 18. Benchmark result of logS (ESOL).



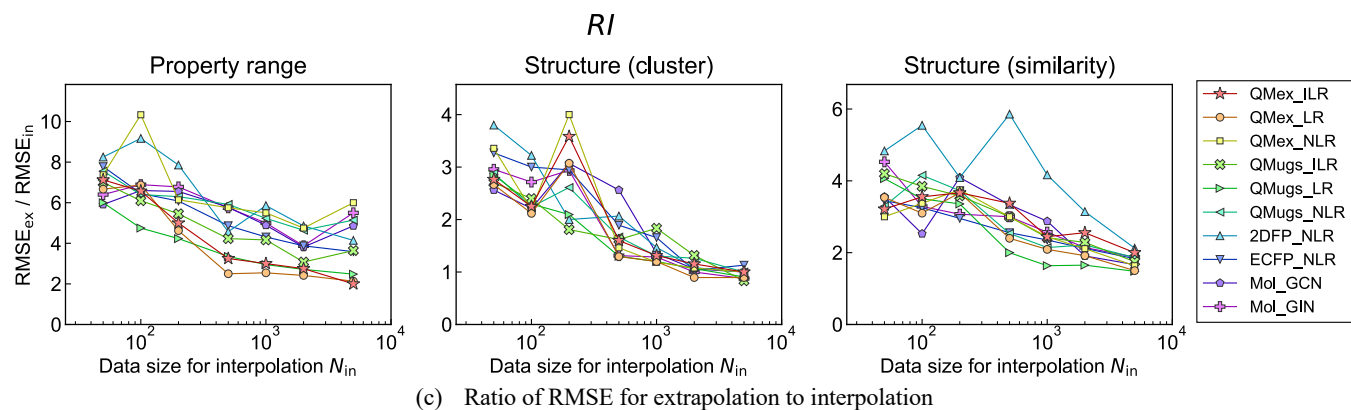
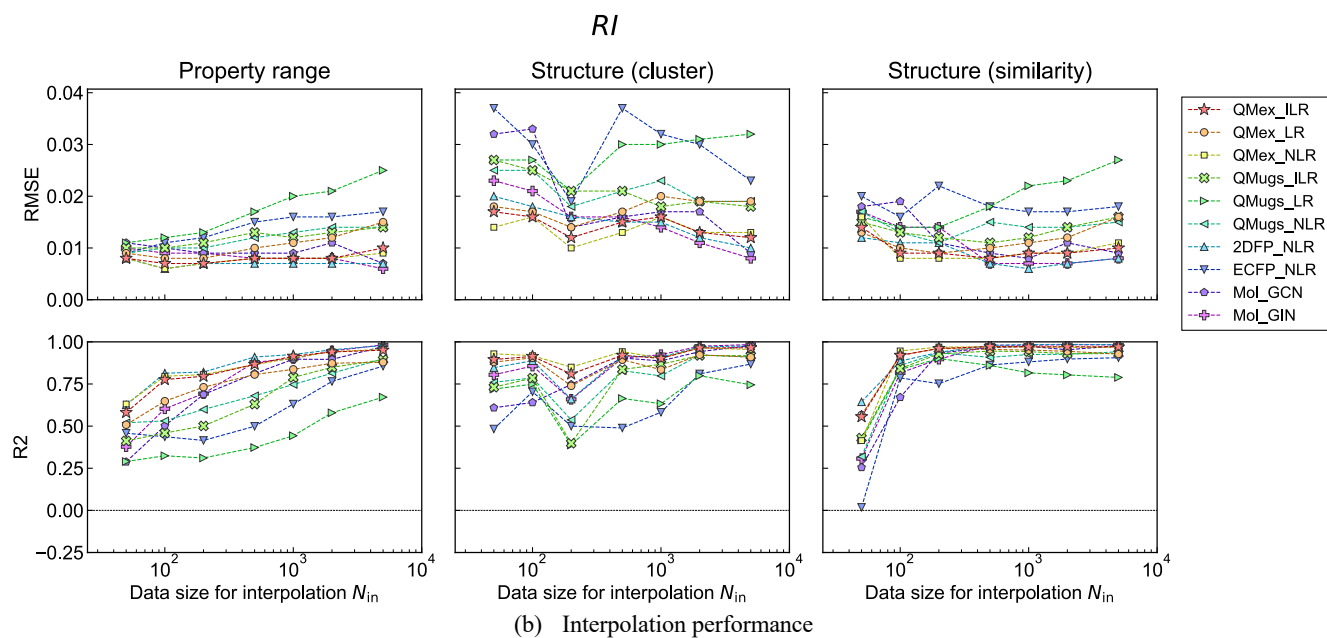
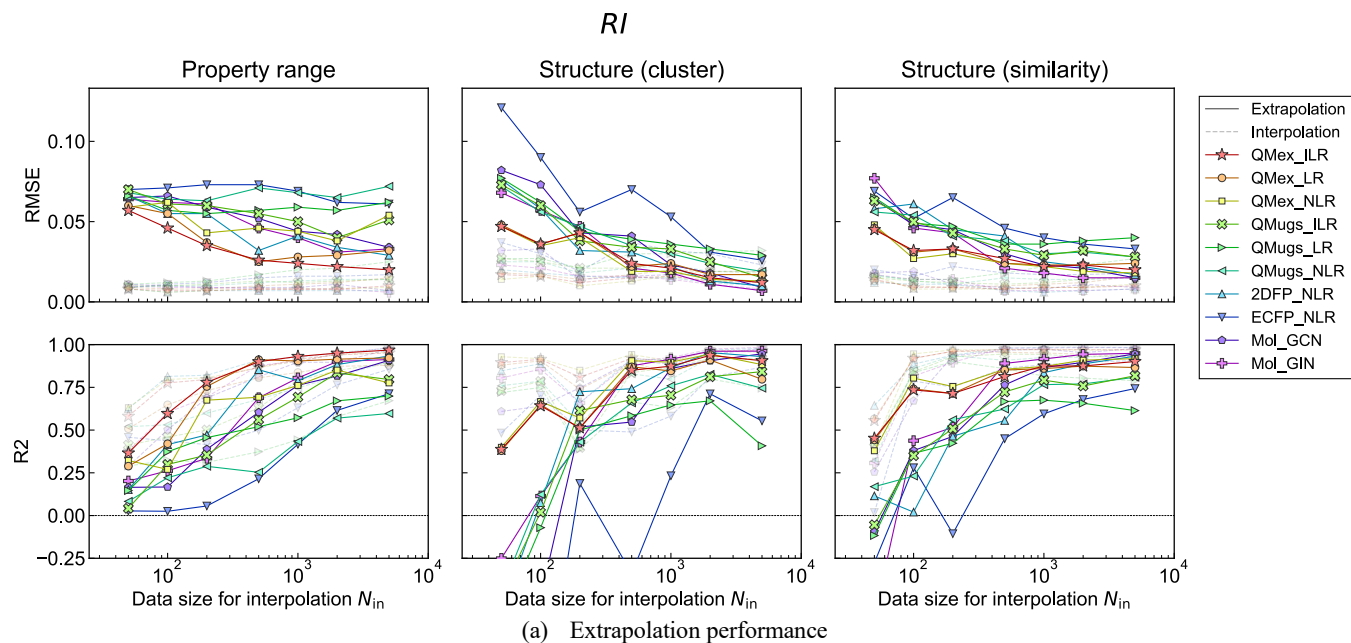
Supplementary Fig. 19. Benchmark result of $\log D$.



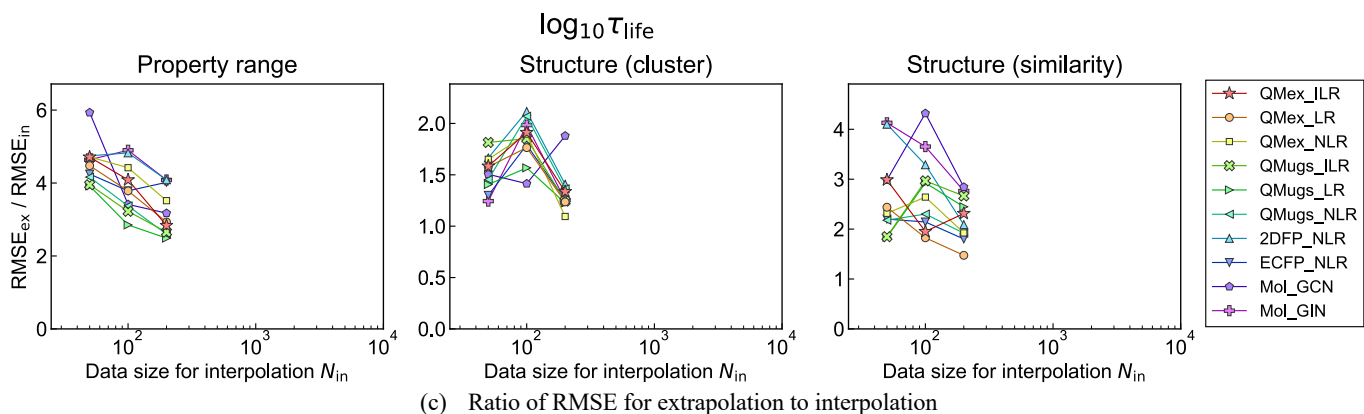
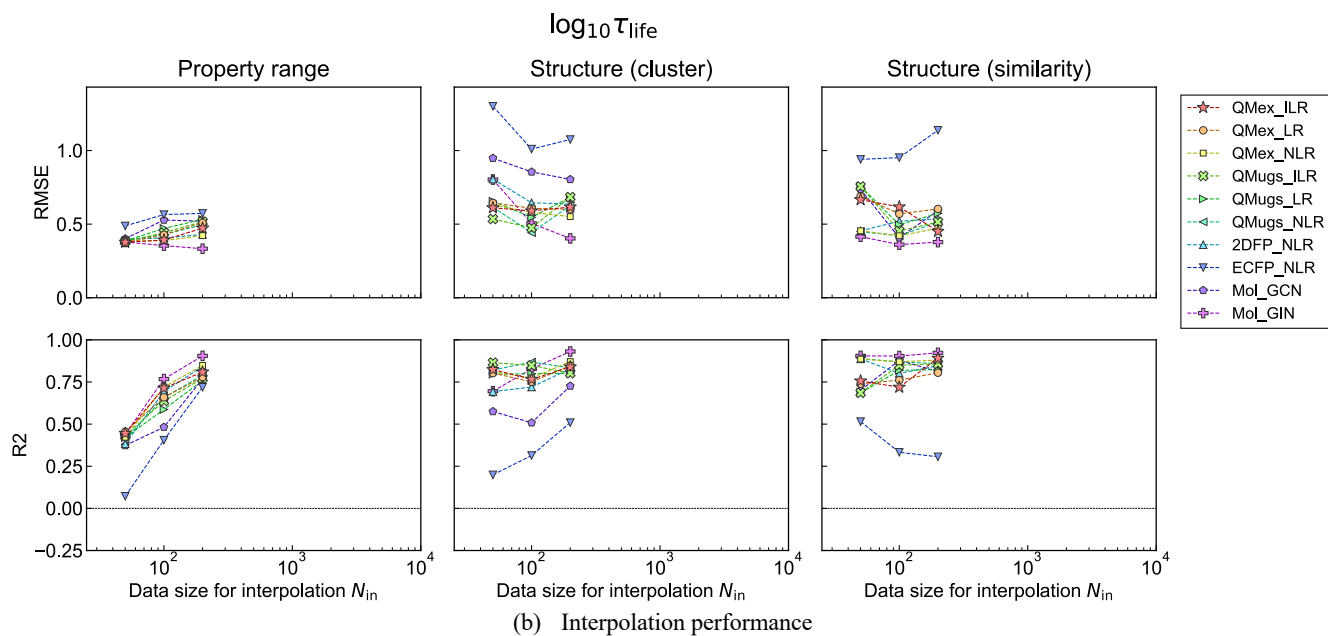
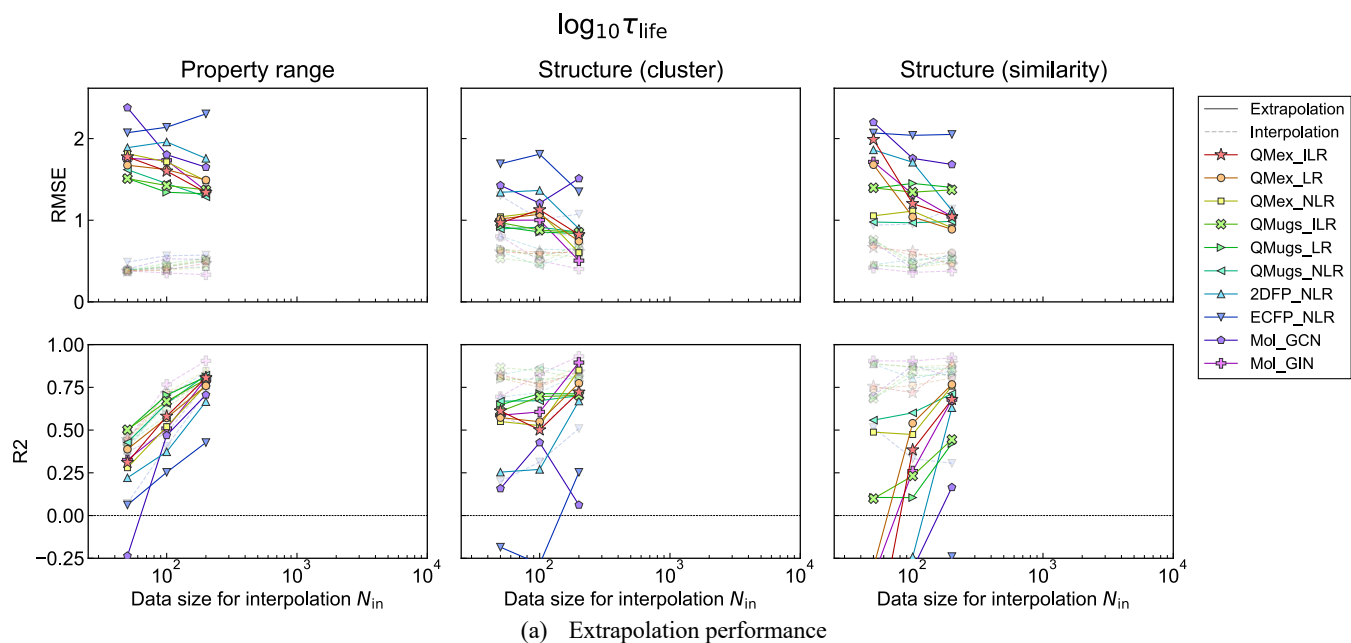
Supplementary Fig. 20. Benchmark result of pK_a (acid).



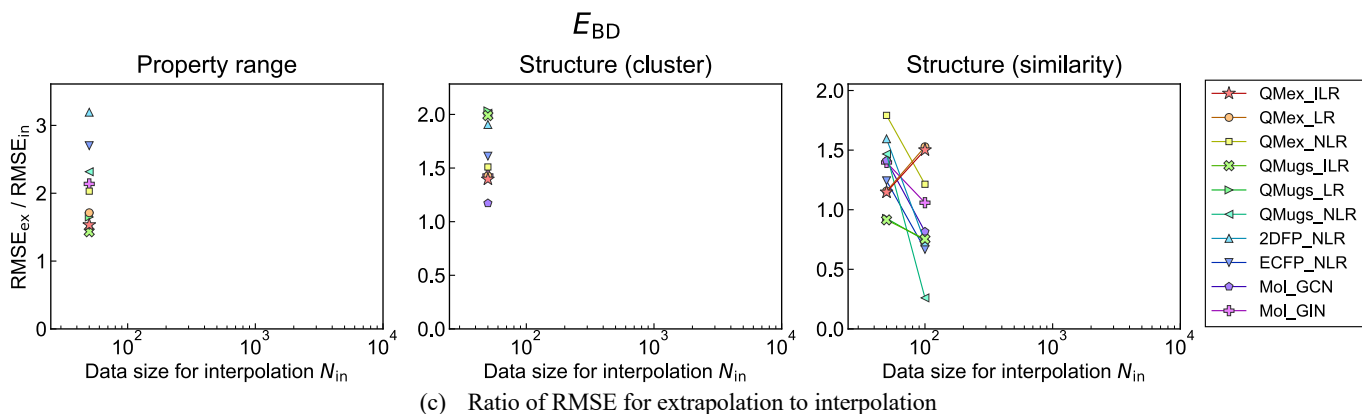
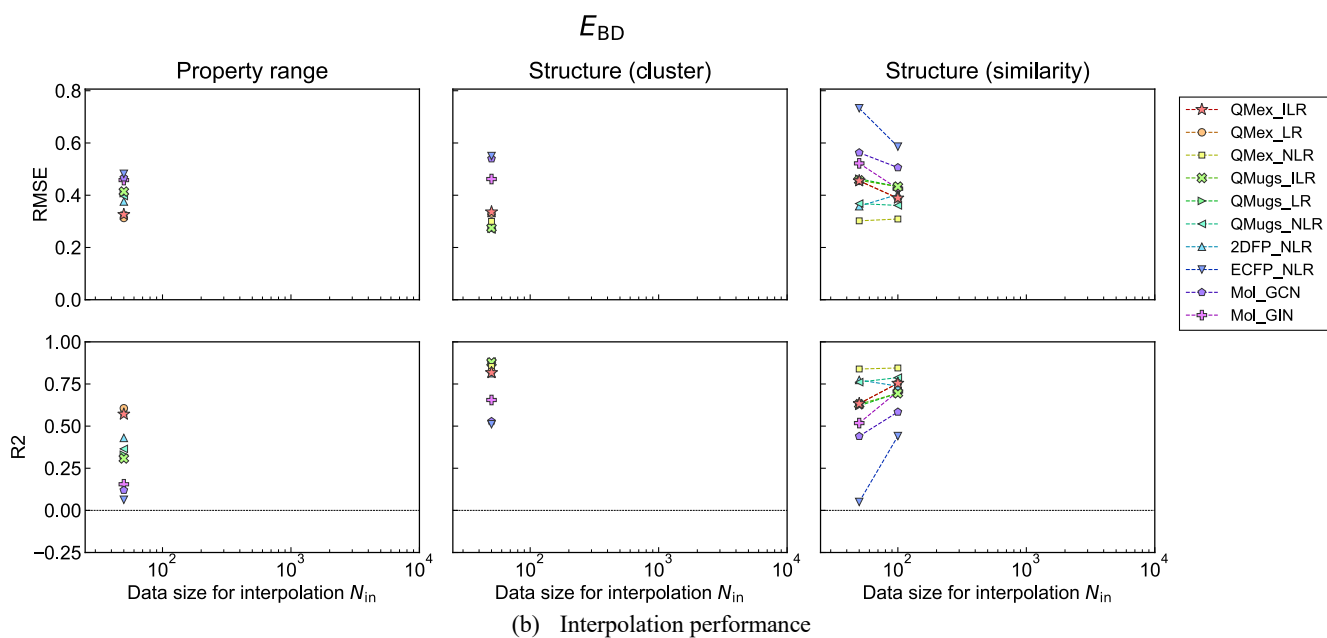
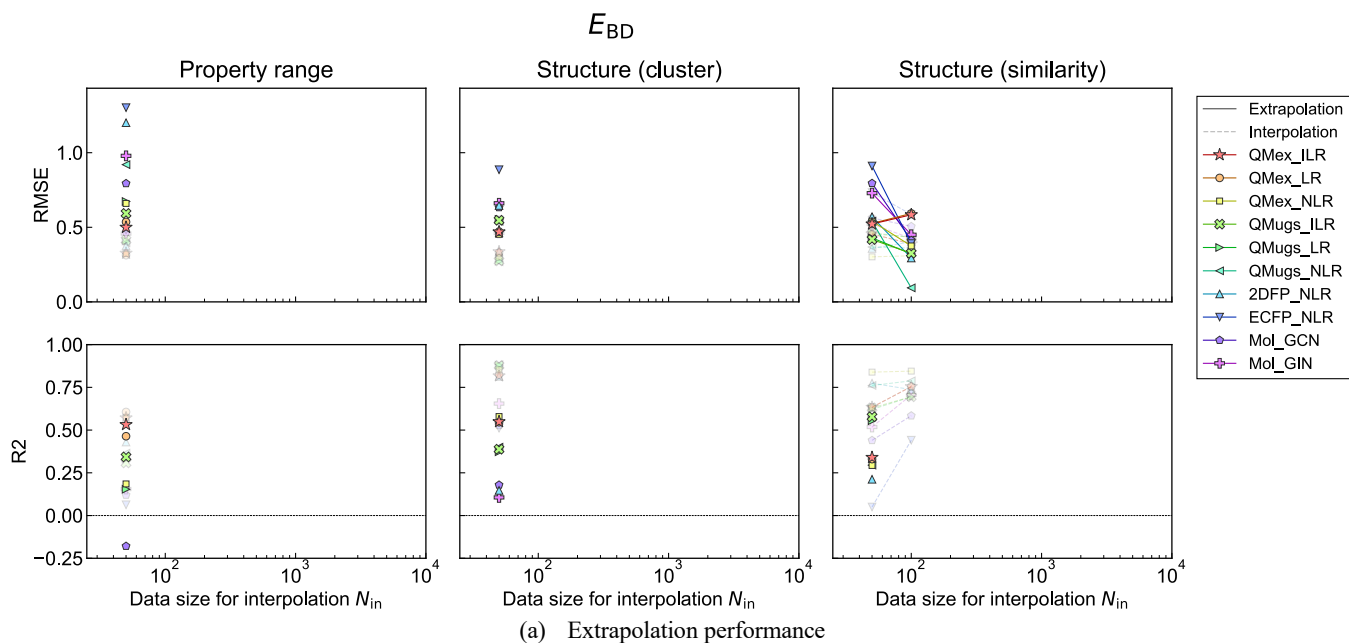
Supplementary Fig. 21. Benchmark result of pK_a (base).



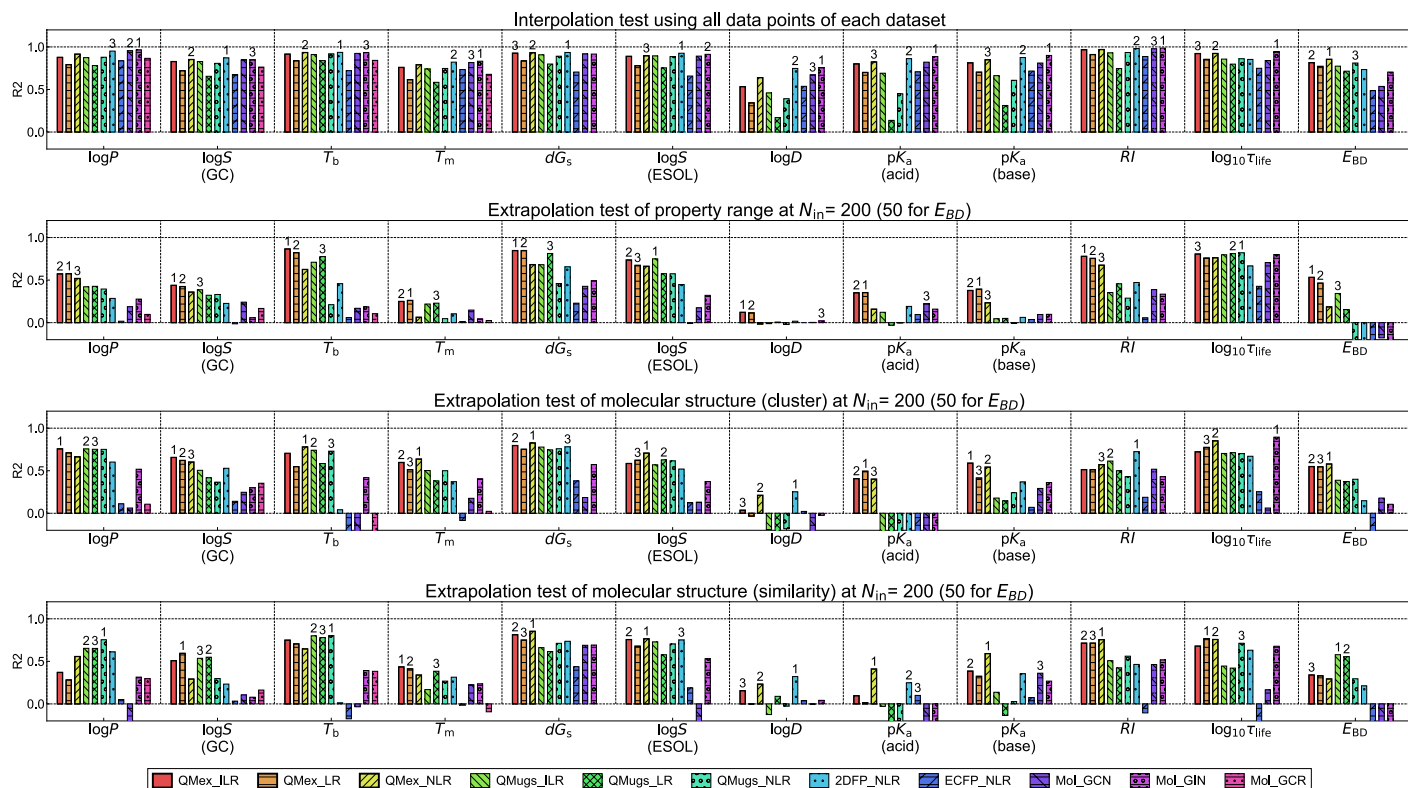
Supplementary Fig. 22. Benchmark result of *RI*.



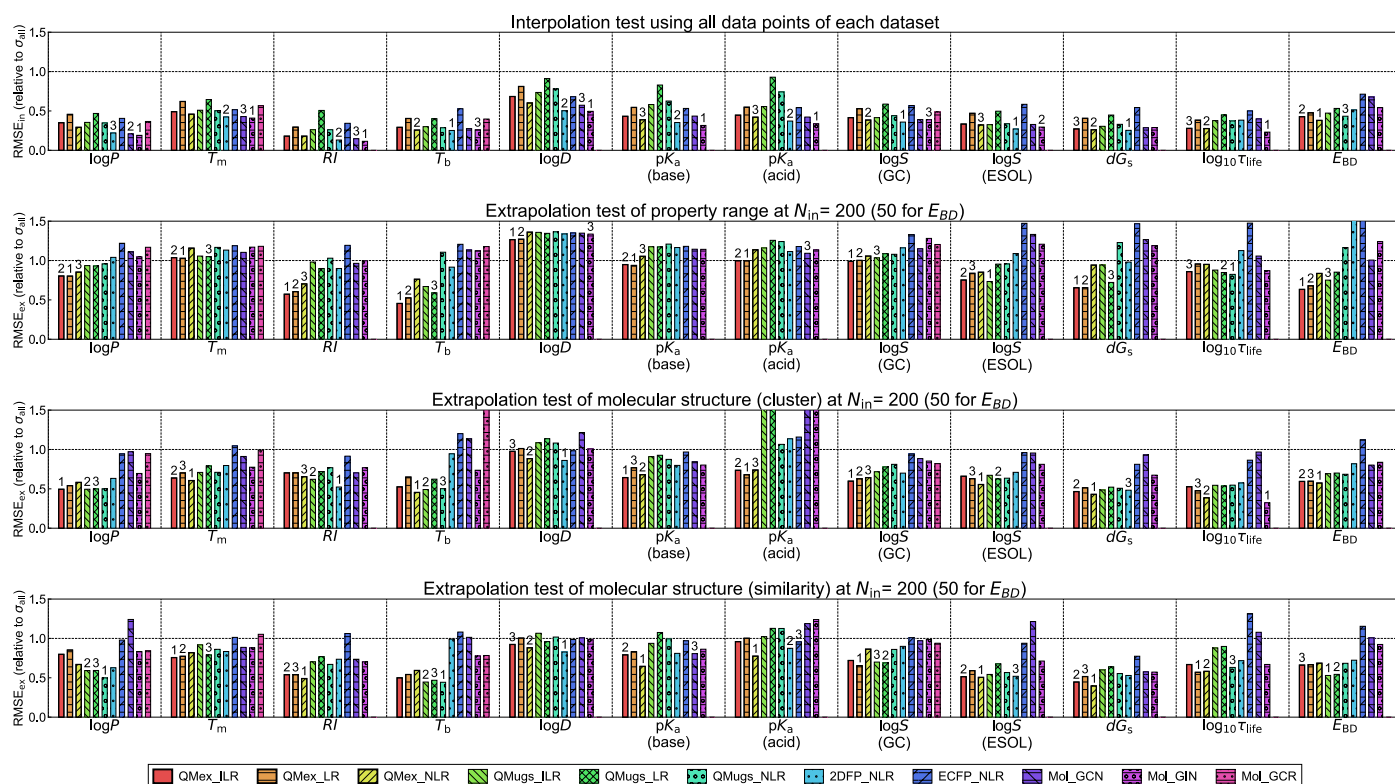
Supplementary Fig. 23. Benchmark result of $\log_{10} \tau_{\text{life}}$.



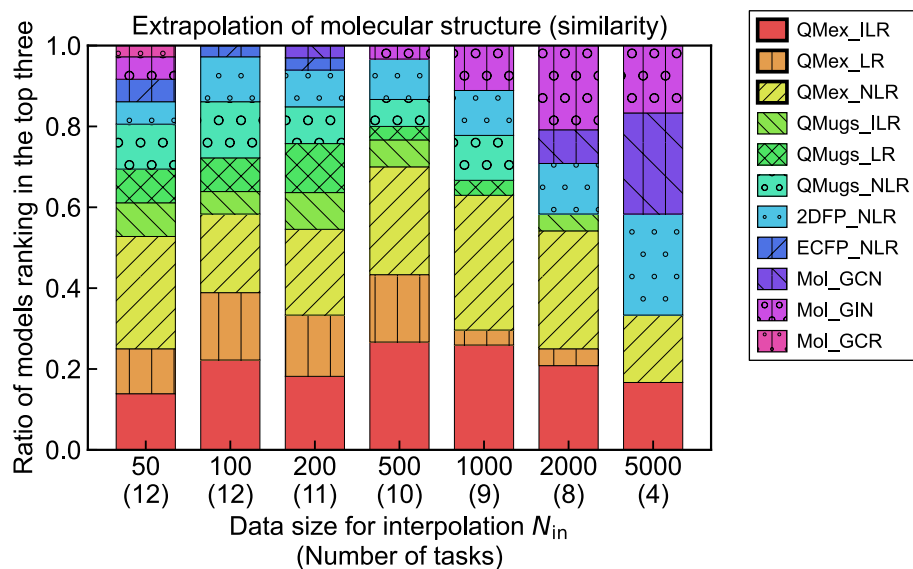
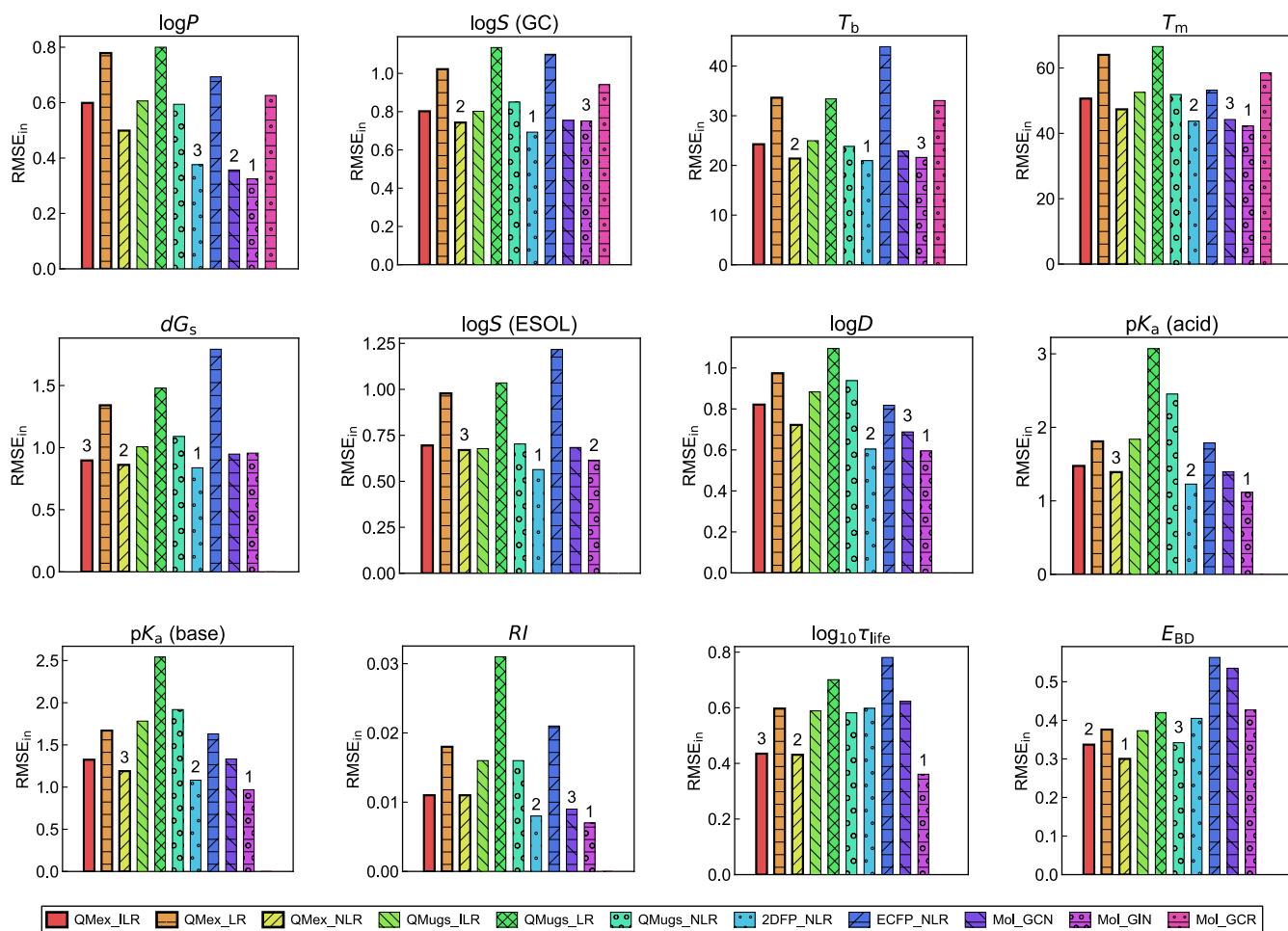
Supplementary Fig. 24. Benchmark result of E_{BD} .



Supplementary Fig. 25. Evaluation results of the interpolation and extrapolation tests with R^2 , corresponding to Fig. 4. The metrics were calculated using predicted values that were the average of predictions from 10 ensemble models. The numbers above bars represent rankings of model performance in the top three.

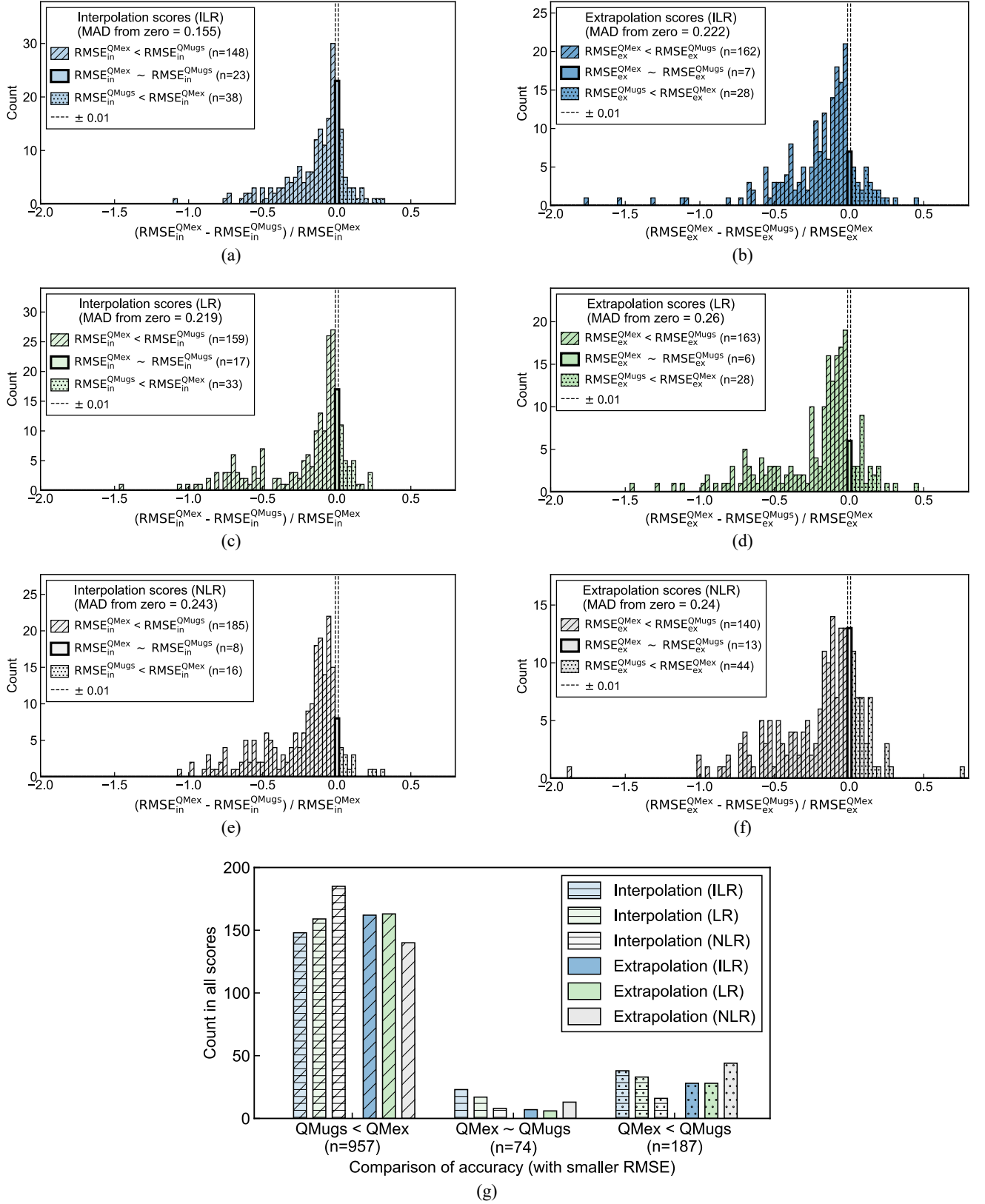


Supplementary Fig. 26. Evaluation results of the interpolation and extrapolation tests with RMSE relative to σ_{all} , where σ_{all} represents the standard deviation of each dataset as listed in Table 1. This corresponds to Fig. 4, arranged in order of increasing size of property datasets. The metrics were calculated using predicted values that were the average of predictions from 10 ensemble models. The numbers above bars represent rankings of model performance in the top three.

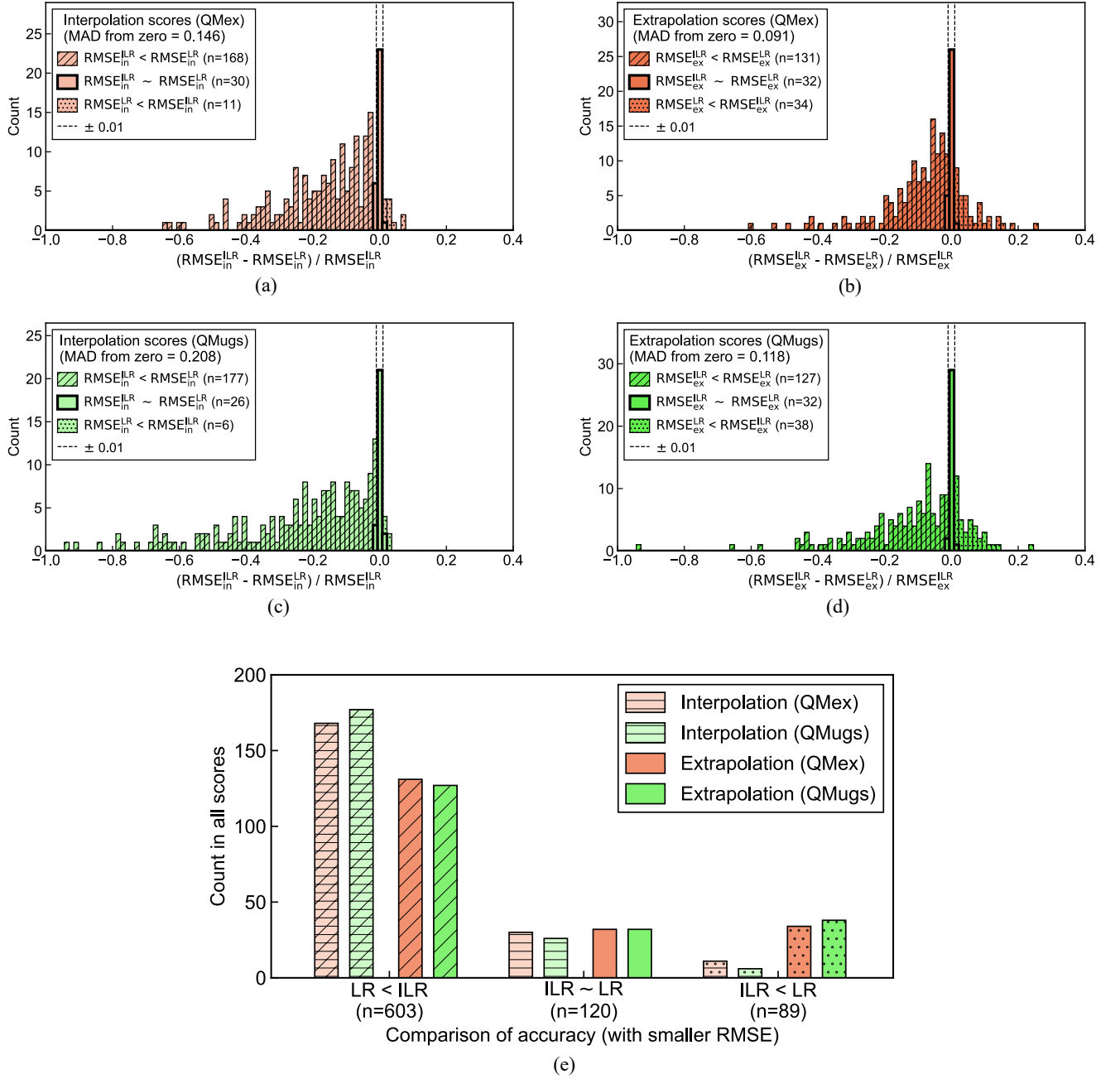




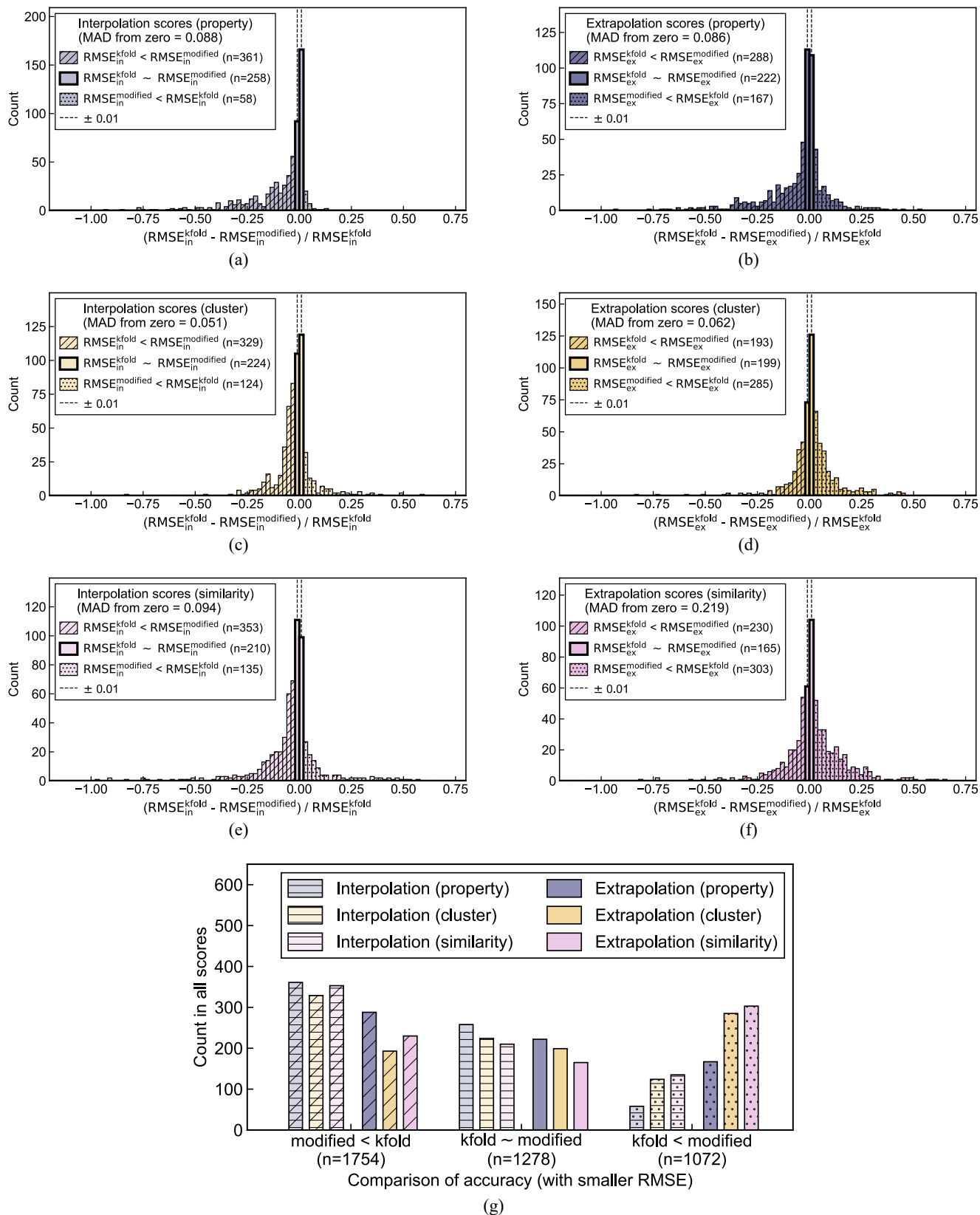
Supplementary Fig. 29. Comparison results of extrapolation performance between modified-CV and kfold-CV methods for the three extrapolation tests at data size for interpolation $N_{in}=200$. The results with modified-CV and kfold-CV are illustrated by dark-color bars and light-color bars, respectively. The title above each figure represents the rate of accurate scores between modified-CV and kfold-CV.



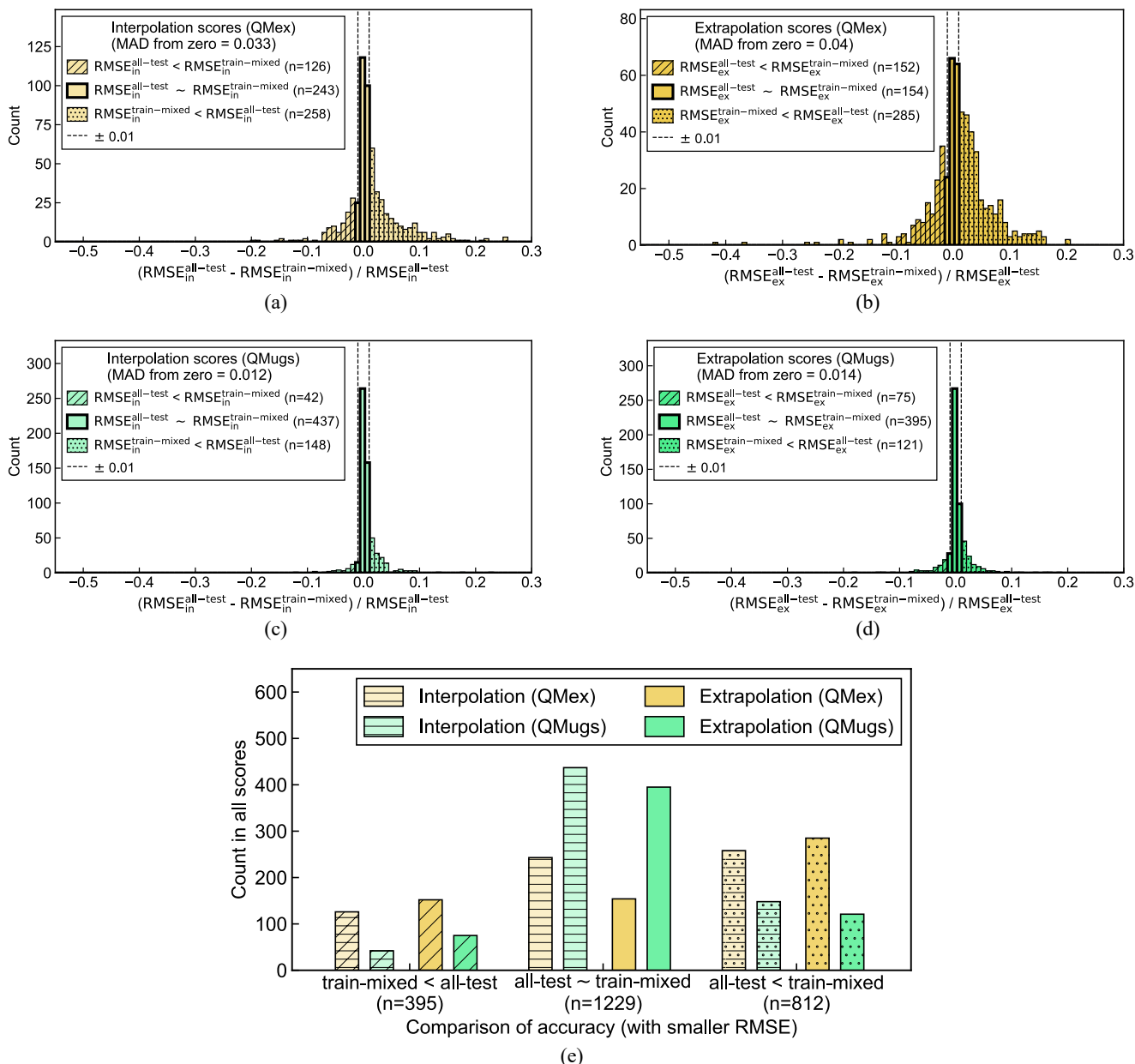
Supplementary Fig. 30. Statistical evaluation for performance comparison between QMex and QMugs-based models. (a)–(f) Distribution of difference in $RMSE_{in}$ and $RMSE_{ex}$ between QMex and QMugs-based models with QM-based ILR, LR, and NLR models, respectively, where $RMSE_{in}$ and $RMSE_{ex}$ represents the interpolation and extrapolation performance by RMSE. The superiority or inferiority between QMex and QMugs-based models is evaluated as shown in hatches, assuming equivalent performance when difference is within $\pm 1\%$. The MAD (mean absolute deviation) from zero serves as estimation of the spread of a distribution relative to zero. (g), Comparison of the interpolation and extrapolation performance of QMex and QMugs-based models based on the count of their superiority.



Supplementary Fig. 31. Statistical evaluation for performance comparison between QM-based ILR and LR models. (a)–(d) Distribution of difference in $RMSE_{in}$ and $RMSE_{ex}$ between ILR and LR with QMEx and QMugs-based models, respectively, where $RMSE_{in}$ and $RMSE_{ex}$ represents the interpolation and extrapolation performance by RMSE. The superiority or inferiority between ILR and LR is evaluated as shown in hatches, assuming equivalent performance when difference is within $\pm 1\%$. The MAD (mean absolute deviation) from zero serves as estimation of the spread of a distribution relative to zero. (e), Comparison of the interpolation and extrapolation performance of QM-based ILR and LR models based on the count of their superiority.



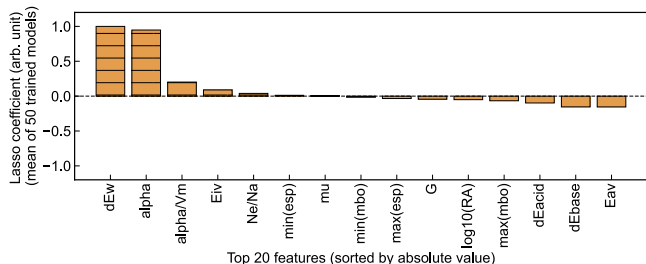
Supplementary Fig. 32. Statistical evaluation for performance comparison between kfold-CV and modified-CV. (a)–(f) Distribution of difference in $RMSE_{in}$ and $RMSE_{ex}$ between kfold-CV and modified-CV in the three extrapolation tests, respectively, where $RMSE_{in}$ and $RMSE_{ex}$ represents the interpolation and extrapolation performance by RMSE. The superiority or inferiority between kfold-CV and modified-CV is evaluated as shown in hatches, assuming equivalent performance when difference is within $\pm 1\%$. The MAD (mean absolute deviation) from zero serves as estimation of the spread of a distribution relative to zero. (g), Comparison of the interpolation and extrapolation performance of kfold-CV and modified-CV based on the count of their superiority.



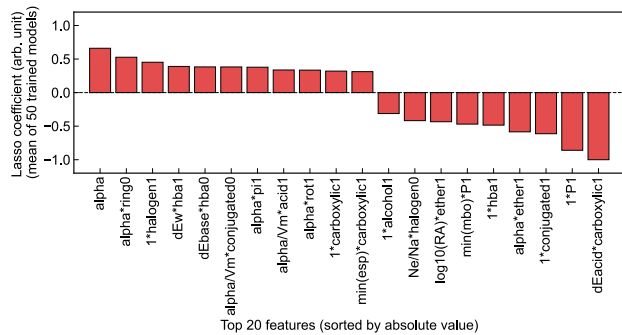
Supplementary Fig. 33. Statistical evaluation for performance comparison of all-test and train-mixed surrogates for QM descriptors. The all-test surrogates were derived from all test values of surrogate GINs, while the train-mixed surrogates included training values. (a)–(d) Distribution of difference in $RMSE_{in}$ and $RMSE_{ex}$ between all-test and train-mixed QMEx and QMugs-based models, respectively, where $RMSE_{in}$ and $RMSE_{ex}$ represents the interpolation and extrapolation performance by RMSE. The superiority or inferiority between all-test and train-mixed surrogates is evaluated as shown in hatches, assuming equivalent performance when difference is within $\pm 1\%$. The MAD (mean absolute deviation) from zero serves as estimation of the spread of a distribution relative to zero. (e) Comparison of the interpolation and extrapolation performance of all-test and train-mixed surrogates based on the count of their superiority.

Impact of QM surrogate accuracy on benchmark performance

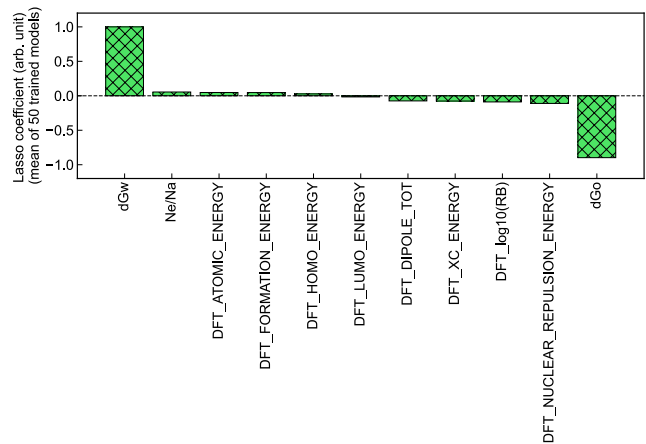
We compared the performance between cases where QM descriptors were solely derived from the test values of surrogate GINs (all-test surrogates, used in the benchmark) and cases where they included training values (train-mixed surrogates). The results in Supplementary Fig. 33 show that more train-mixed surrogates perform better than all-test surrogates, particularly when utilizing QMEx descriptors, as train-mixed surrogates were able to predict QM descriptors more accurately for DFT results. Nonetheless, the observed disparity does not appear to be significant, as evidenced by the small MADs from zero, all measuring below 0.04. This suggests that the QMEx dataset provides sufficient training data for surrogate GINs, and their prediction errors have only a modest impact on benchmark performance.



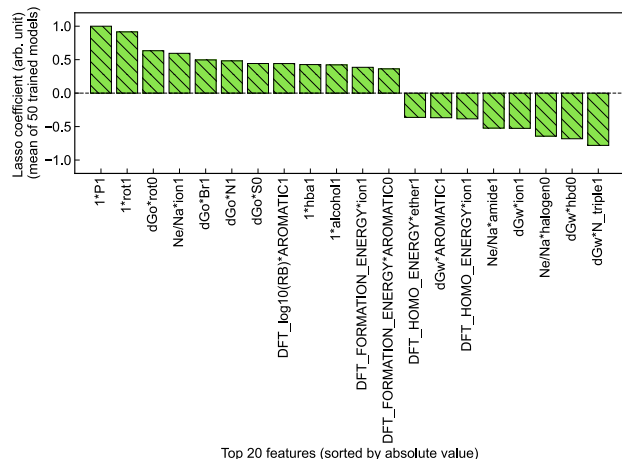
(a1) $\log P$ (QMex-LR)



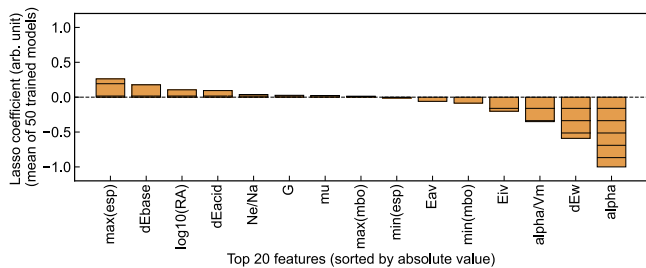
(a2) $\log P$ (QMex-ILR)



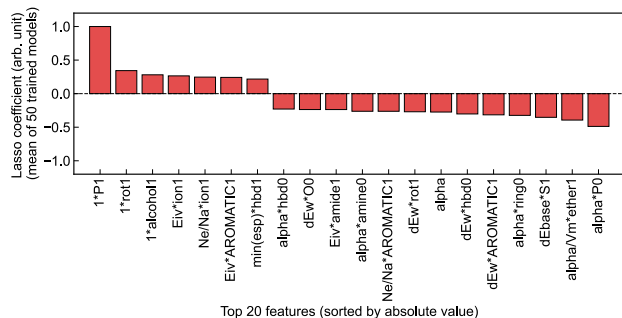
(a3) $\log P$ (QMugs-LR)



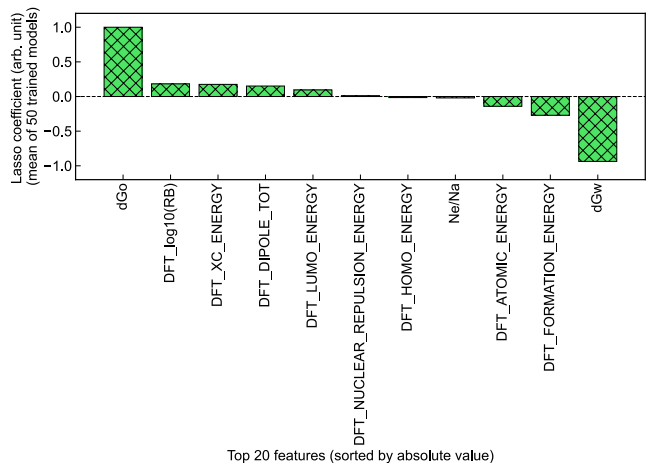
(a4) $\log P$ (QMugs-ILR)



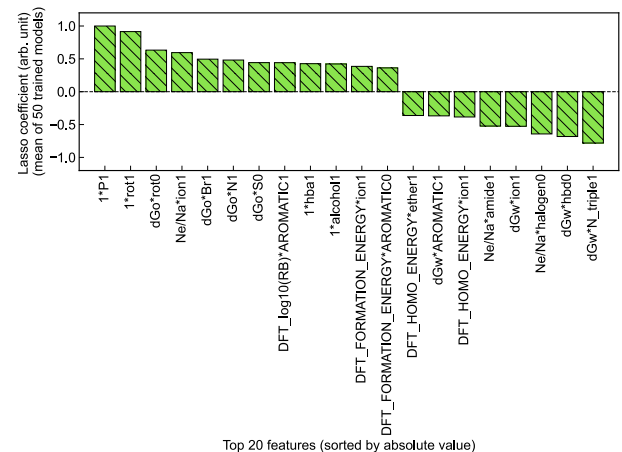
(b1) $\log S$ (GC) (QMex-LR)



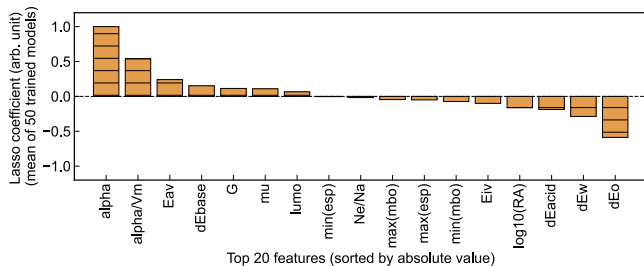
(b2) $\log S$ (GC) (QMex-ILR)



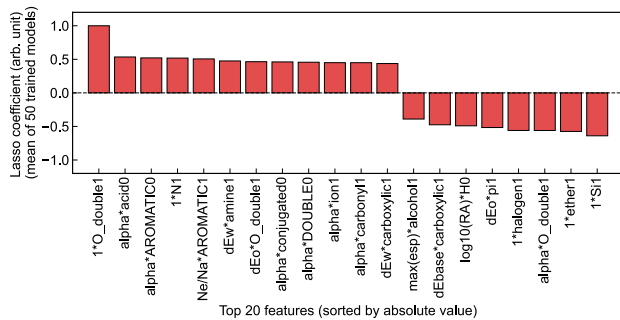
(b3) $\log S$ (GC) (QMugs-LR)



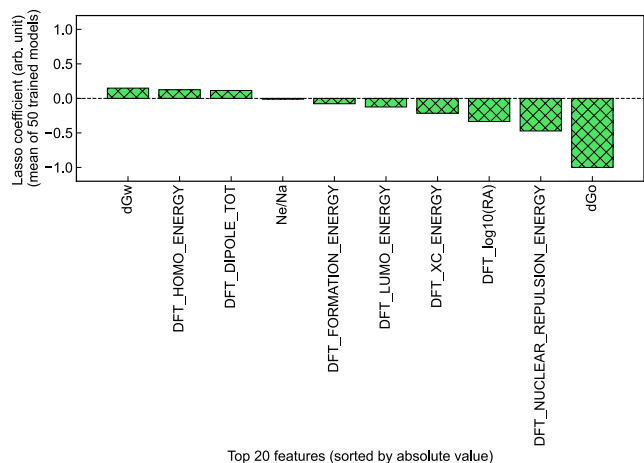
(b4) $\log S$ (GC) (QMugs-ILR)



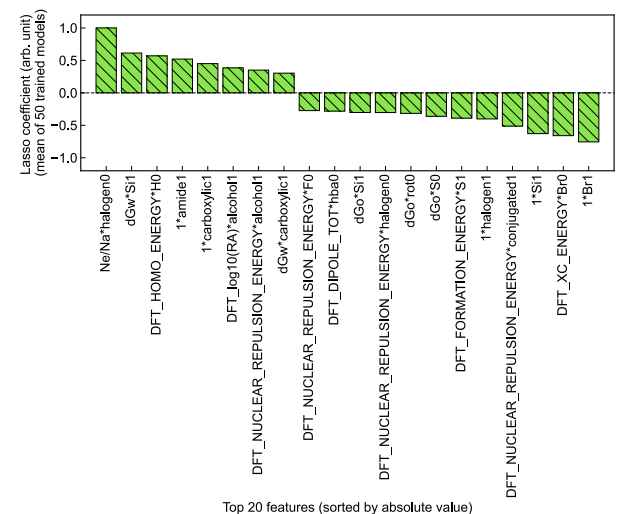
(c1) T_b (QMex-LR)



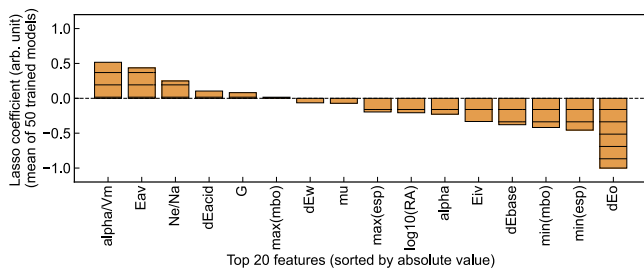
(c2) T_b (QMex-ILR)



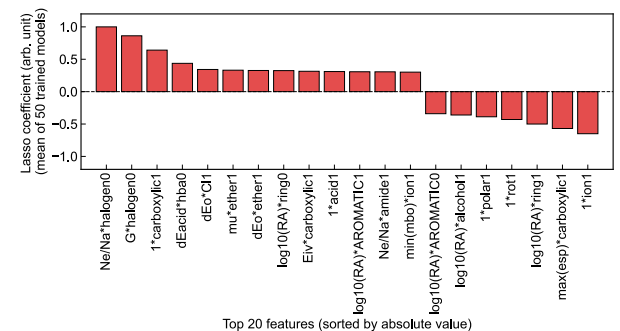
(c3) T_b (QMugs-LR)



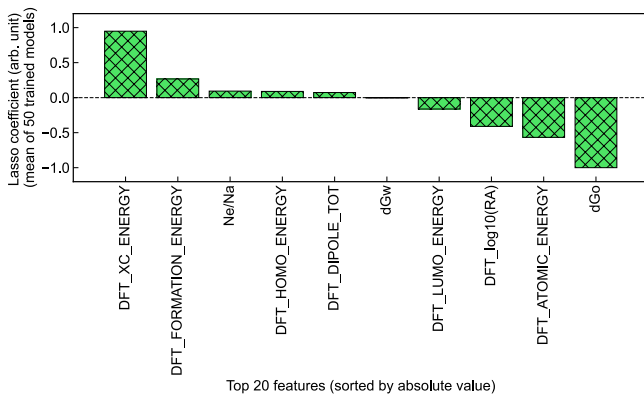
(c4) T_b (QMugs-ILR)



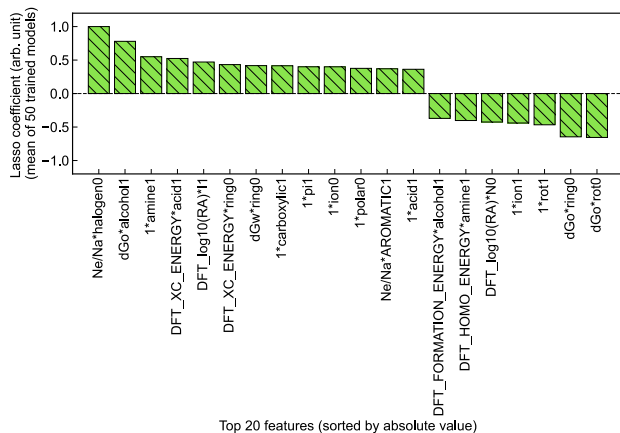
(d1) T_m (QMex-LR)



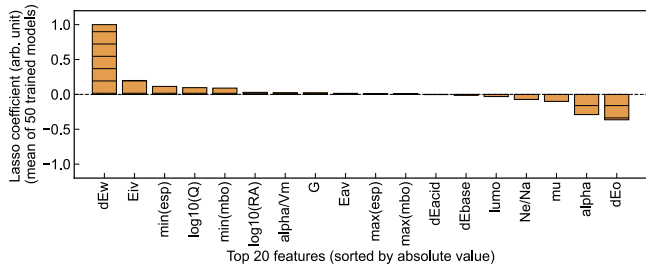
(d2) T_m (QMex-ILR)



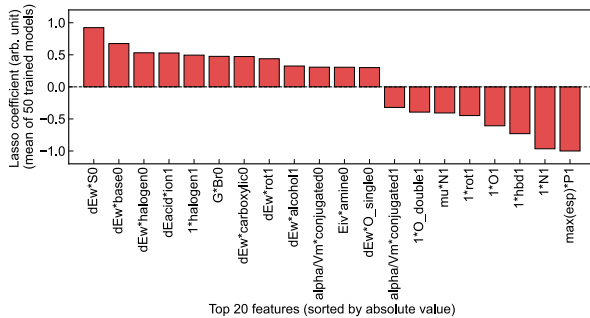
(d3) T_m (QMugs-LR)



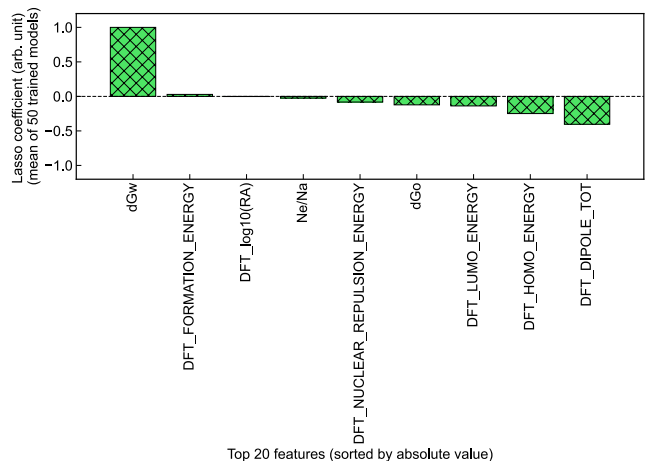
(d4) T_m (QMugs-ILR)



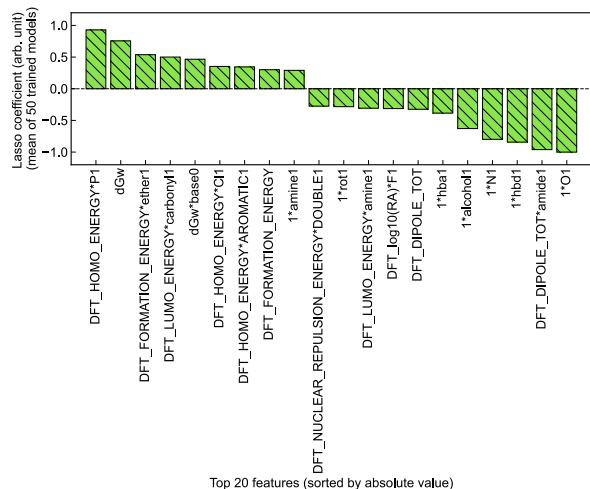
(e1) dG_s (QMex-LR)



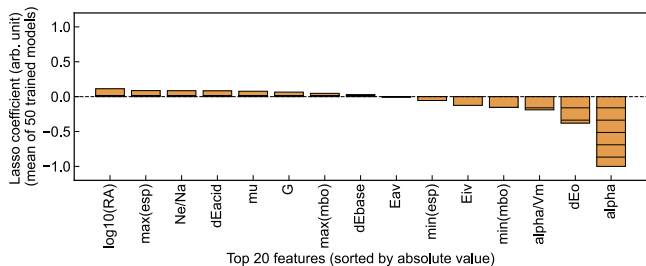
(e2) dG_s (QMex-ILR)



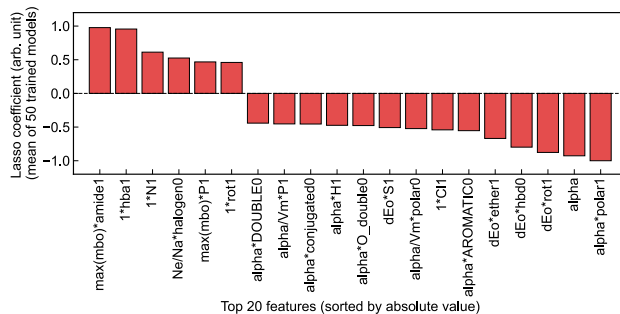
(e3) dG_s (QMugs-LR)



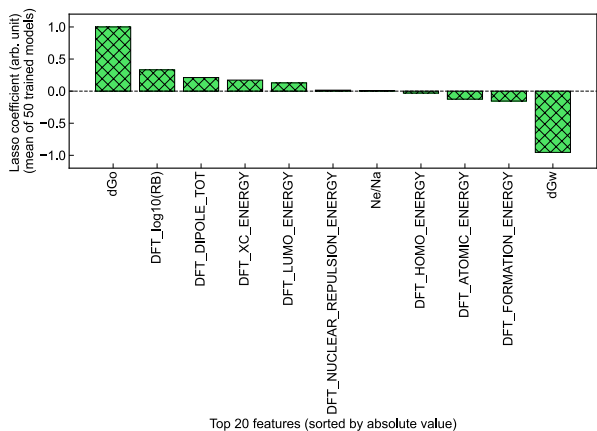
(e4) dG_s (QMugs-ILR)



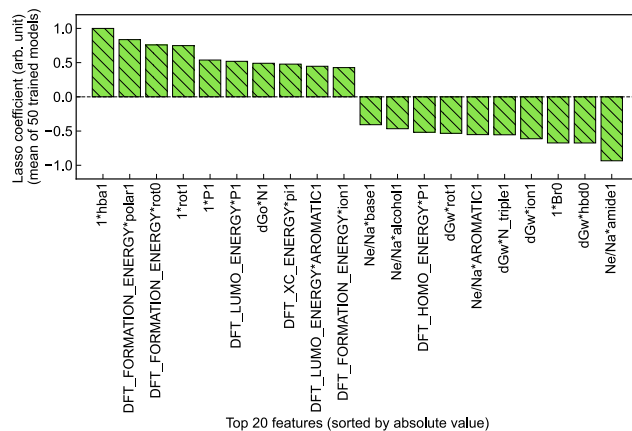
(f1) $\log S$ (ESOL) (QMex-LR)



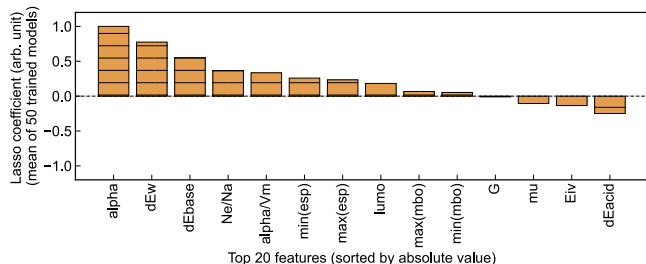
(f2) $\log S$ (ESOL) (QMex-ILR)



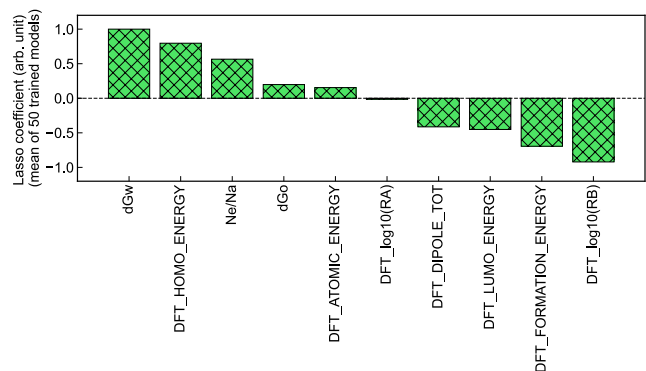
(f3) $\log S$ (ESOL) (QMugs-LR)



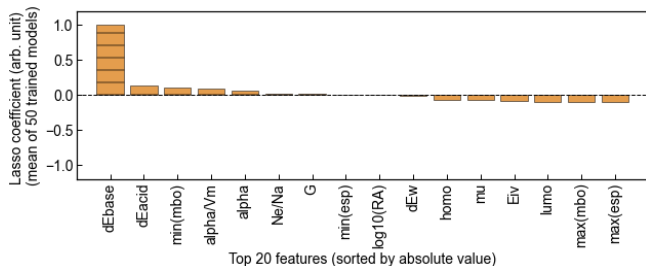
(f4) $\log S$ (ESOL) (QMugs-ILR)



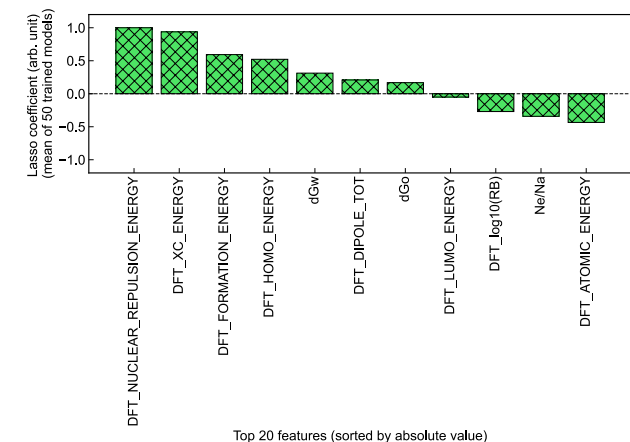
(g1) $\log D$ (QMex-LR)



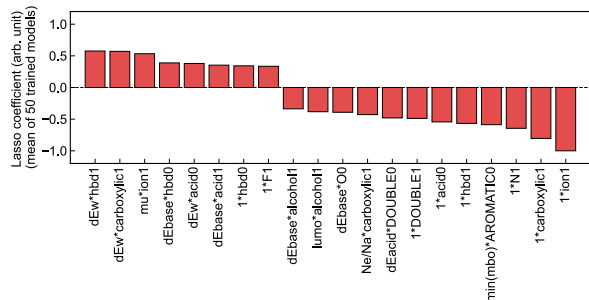
(g3) $\log D$ (QMugs-LR)



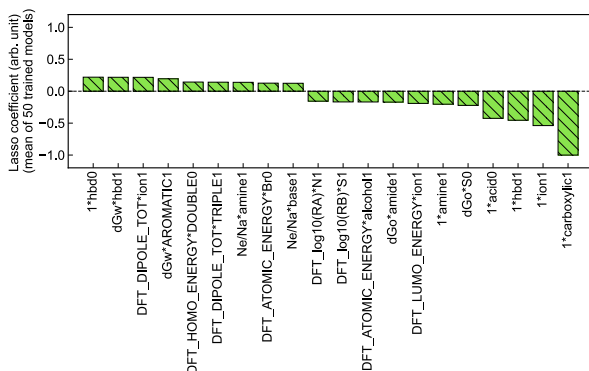
(h1) pK_a (acid) (QMex-LR)



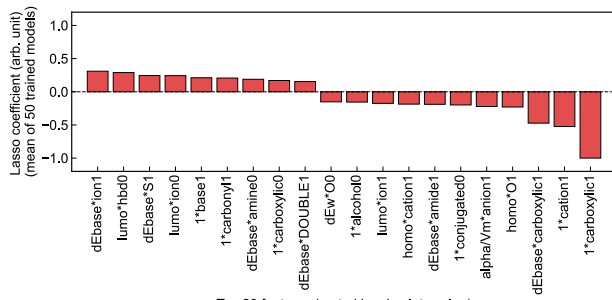
(h3) pK_a (acid) (QMugs-LR)



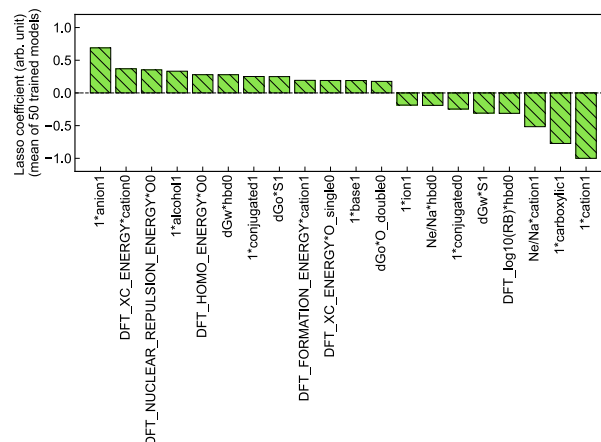
(g2) $\log D$ (QMex-ILR)



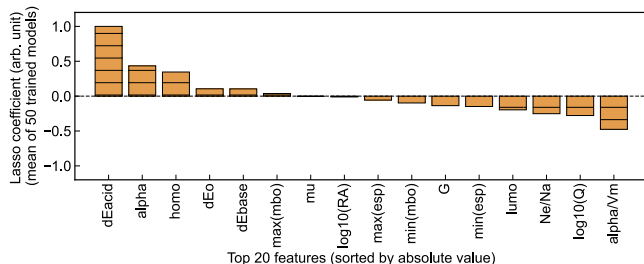
(g4) $\log D$ (QMugs-ILR)



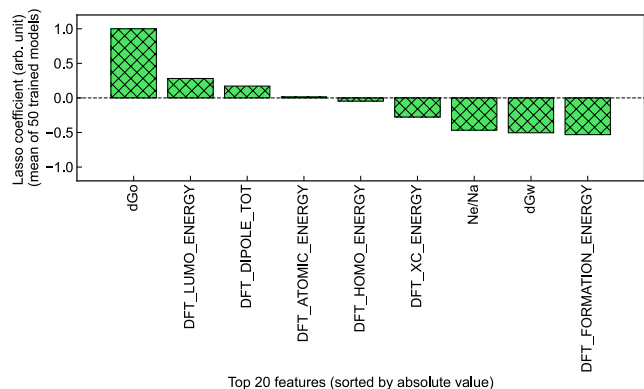
(h2) pK_a (acid) (QMex-ILR)



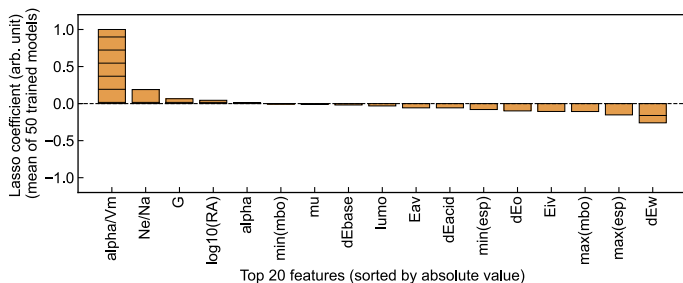
(h4) pK_a (acid) (QMugs-ILR)



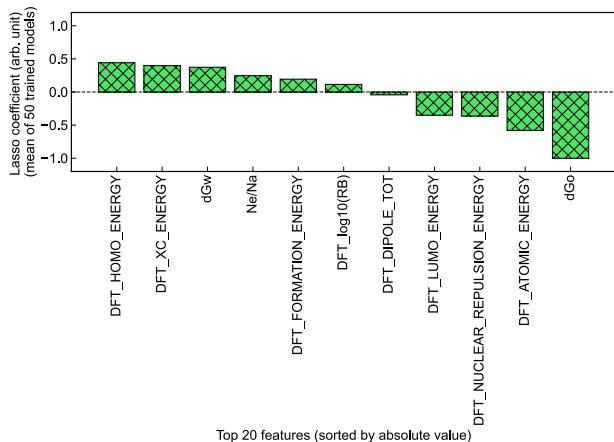
(i1) pK_a (base) (QMex-LR)



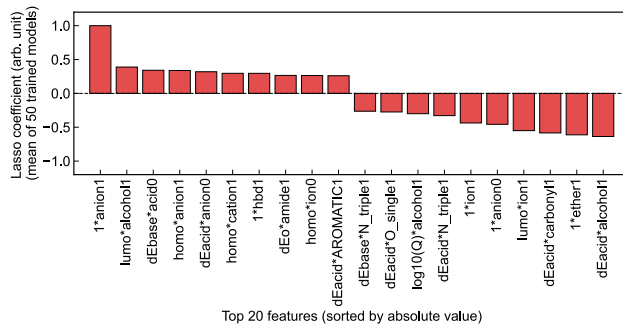
(i3) pK_a (base) (QMugs-LR)



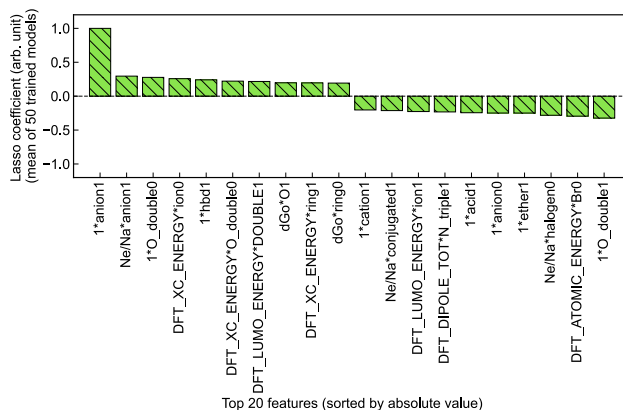
(j1) RI (QMex-LR)



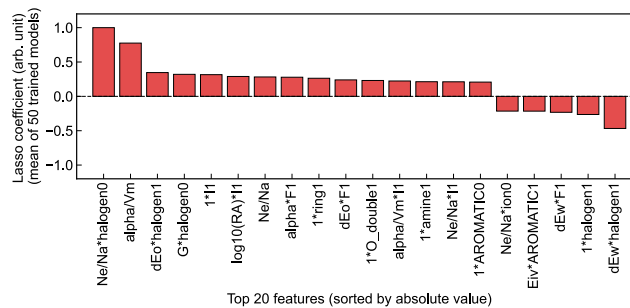
(j3) RI (QMugs-LR)



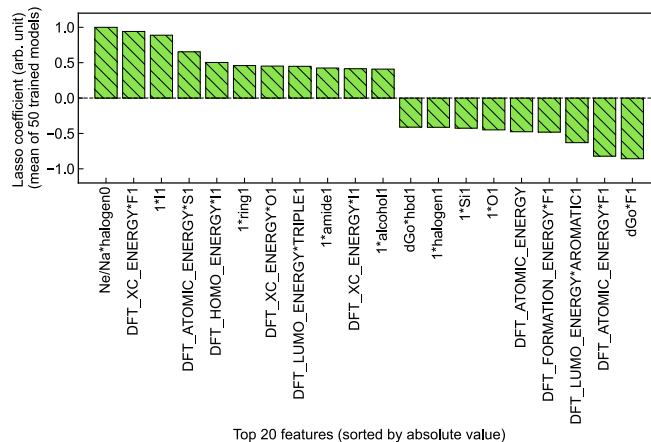
(i2) pK_a (base) (QMex-ILR)



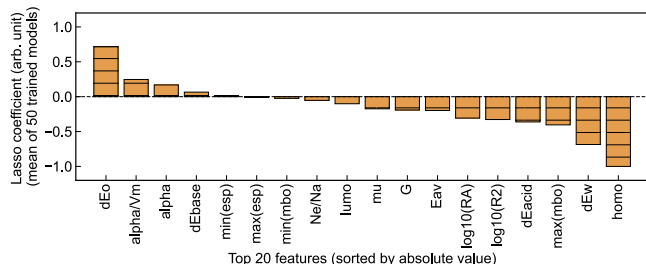
(i4) pK_a (base) (QMugs-ILR)



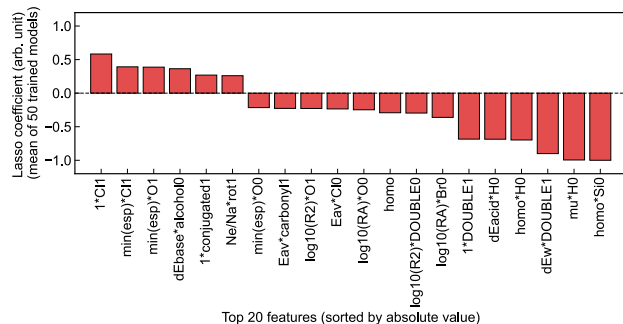
(j2) RI (QMex-ILR)



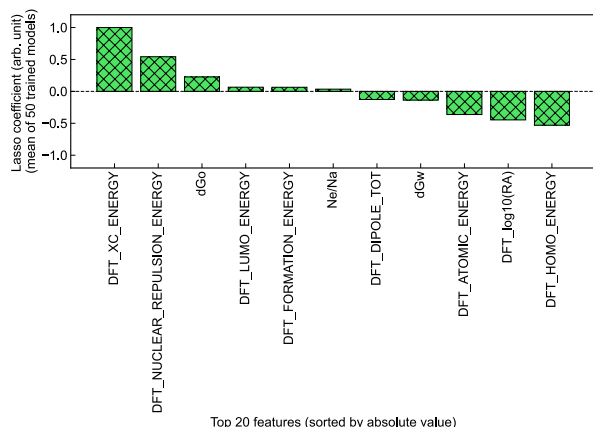
(j4) RI (QMugs-ILR)



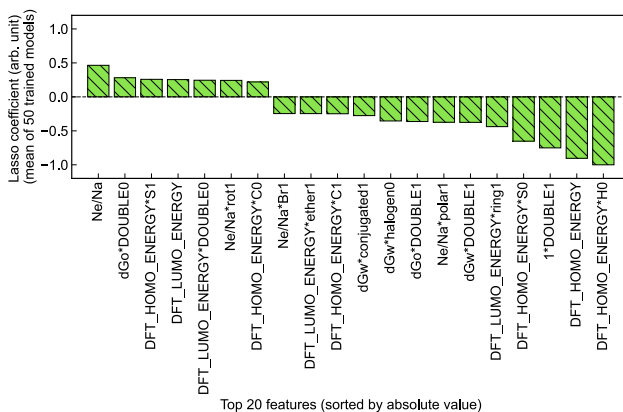
(k1) $\log_{10}\tau_{\text{life}}$ (QMex-LR)



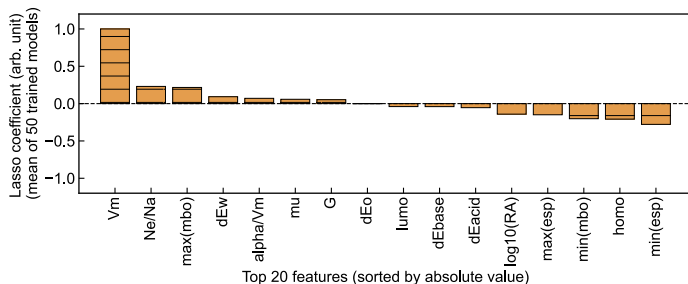
(k2) $\log_{10}\tau_{\text{life}}$ (QMex-ILR)



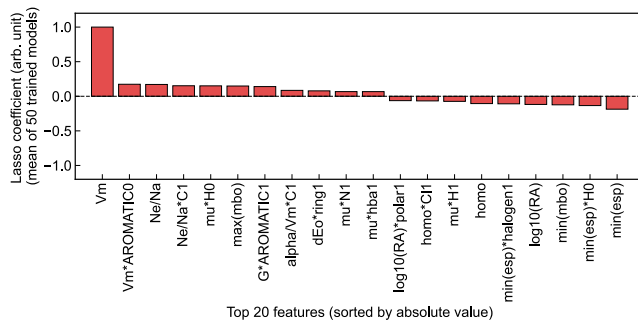
(k3) $\log_{10}\tau_{\text{life}}$ (QMugs-LR)



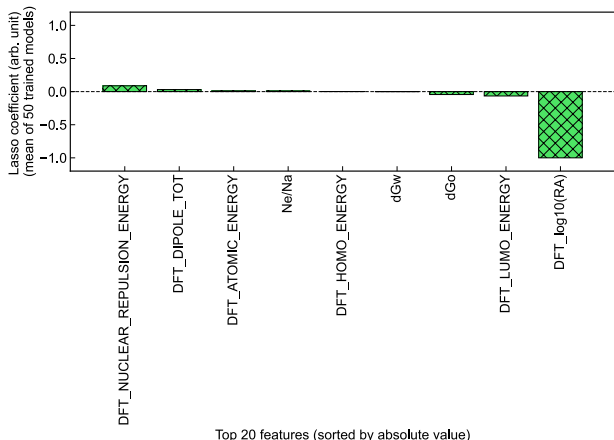
(k4) $\log_{10}\tau_{\text{life}}$ (QMugs-ILR)



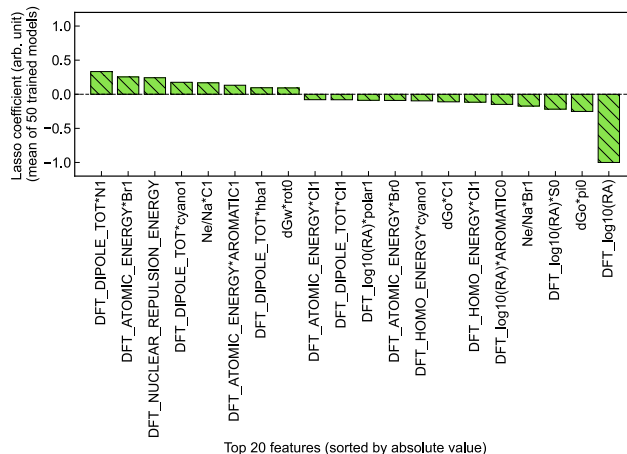
(11) E_{BD} (QMex-LR)



(12) E_{BD} (QMex-ILR)



(13) E_{BD} (QMugs-LR)



(14) E_{BD} (QMugs-ILR)

Supplementary Fig. 34. Results of feature importance evaluation for QMex-LR, QMex-ILR, QMugs-LR, and QMugs-ILR using trained Lasso coefficients as indicators of importance. Prior to training, multicollinearity and sparsity of features are removed as described in the Method section, and the mean of trained Lasso coefficients is calculated through ensemble learning of 10-times 5-fold tests to eliminate fluctuations.

- [1] Zhan, C.-G., Nichols, J. A. & Dixon, D. A. Ionization Potential, Electron Affinity, Electronegativity, Hardness, and Electron Excitation Energy: Molecular Properties from Density Functional Theory Orbital Energies. *J. Phys. Chem. A* 107, 4184–4195 (2003)
- [2] Marenich, A. V., Cramer, C. J. & Truhlar, D. G. Universal solvation model based on solute electron density and on a continuum model of the solvent defined by the bulk dielectric constant and atomic surface tensions. *J. Phys. Chem. B* 113, 6378–6396 (2009)
- [3] Bochevarov, A. D., Watson, M. A., Greenwood, J. R. & Philipp, D. M. Multiconformation, Density Functional Theory-Based pKa Prediction in Application to Large, Flexible Organic Molecules with Diverse Functional Groups. *J. Chem. Theory Comput.* 12, 6001–6019 (2016)
- [4] Iser, C., Atz, K., Jimenez-Luna, J. & Schneider, G. QMugs, quantum mechanical properties of drug-like molecules. *en. Sci Data* 9, 273 (June 2022)
- [5] Vermeire, F. H. & Green, W. H. Transfer learning for solvation free energies: From quantum chemistry to experiments. *Chem. Eng. J.* 418, 129307 (Aug. 2021)
- [6] Dassault Systemes, IBOVIA cosmothrm, 2020.
- [7] Wang, Y., Wang, J., Cao, Z. & Barati Farimani, A. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence* 4, 279–287 (2022)
- [8] Ross, J. et al. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence* 4, 1256–1264 (2022)
- [9] RDKit: Open-source cheminformatics. <https://www.rdkit.org>
- [10] O’Boyle, N. M., Morley, C. & Hutchison, G. R. Pybel: a Python wrapper for the OpenBabel cheminformatics toolkit. *Chem. Cent. J.* 2, 5 (2008)
- [11] Ong, S. P. et al. Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Comput. Mater. Sci.* 68, 314–319 (2013)
- [12] Wu, J. et al. MF-SuP-pKa: Multi-fidelity modeling with subgraph pooling mechanism for pKa prediction. *Acta Pharmaceutica Sinica B*, 13, 6, 2572–2584, (2023)
- [13] GitHub repository: <https://github.com/wujialu/MF-SuP-pKa>
- [14] Akiba, T., Sano, S., Yanase, T., Ohta, T. & Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining 2623–2631 (Association for Computing Machinery, 2019).