Supporting Information

Recent progresses in the data-driven discovery of novel photovoltaic materials

Tian Lu[a], Minjie Li[b,*], Wencong Lu[a,b,*], Tongyi Zhang[a,*]

[a] *Materials Genome Institute, Shanghai University, Shanghai 200444, China*
[b] *Department of Chemistry, College of Sciences, Shanghai University, Shanghai 200444, China*

[*]Corresponding Authors.
E-mail addresses: minjieli@shu.edu.cn (M.J. Li); wclu@shu.edu.cn (W.C. Lu); zhangty@shu.edu.cn (T.Y. Zhang)

Table S1 Thirty classes of descriptors in Dragon software. We give a brief description for each type of descriptor and the difficult estimation of their interpretability.

Descriptor Type	Brief Description	Interpretability	Tool/Library/ Reference
Constitutional indices	Reflect chemical composition	Easy	Dragon/ PaDEL
Ring descriptor	Numerical quantities of rings	Easy	Dragon/PaDEL
Topological indices	Numerical quantifiers of molecular topology	Easy	Dragon/PaDEL
Walk and path counts	Graph representation based on walks and paths	Hard	Dragon/PaDEL
Connectivity indices	Numerical quantifiers of bonds	Easy	Dragon/PaDEL
Information indices	Numerical quantities of atom-based properties	Hard	Dragon
2D matrix-based descriptors	Numerical quantities based on graph-theoretical matrices	Hard	Dragon
2D autocorrelations	Describe how a considered property is distributed along with a topological molecular structure	Hard	Dragon/PaDEL
Burden eigenvalues	Eigenvalues in Burden matrix	Hard	Dragon/PaDEL
P-VSA-like descriptors	Amount of van der Waals surface area (VSA)	Hard	Dragon/PaDEL
ETA indices	Extended topological indices	Hard	Dragon
Edge adjacency indices	Information of edge adjacency matrix	Hard	Dragon/PaDEL
Geometrical descriptors	Reflect molecular shape	Hard	Dragon
3D matrix-based descriptors	3D version of 2D matrix-based descriptors	Hard	Dragon
3D autocorrelations	3D version of 2D autocorrelations	Hard	Dragon/PaDEL
RDF descriptors	Numerical quantities based on a radial distribution function	Hard	Dragon/PaDEL
3D-MoRSE descriptors	3D-molecule representation based on electron diffraction study	Hard	Dragon
WHIM descriptors	Geometrical descriptors based on the projections of the atoms along principal axes	Hard	Dragon/PaDEL
GETAWAY descriptors	Chemical structure descriptors derived from the molecular influence matrix	Hard	Dragon
Randic molecular profiles	Reflect the interatomic geometric distances of a molecule	Hard	Dragon

Functional groups count	The number of specific functional groups	Easy	Dragon
Atom-centered fragments	The number of specific atom types	Easy	Dragon
Atom-type E-state indices	Reflect number of groups combined with electro-topological state	Easy	Dragon
2D atom pairs	Numerical quantifiers of atom pairs in 2D	Easy	Dragon
3D atom pairs	Numerical quantifiers of atom pairs in 3D	Easy	Dragon
Charge descriptors	Reflect charge properties of a molecule	Easy	Dragon
Molecular properties	Physic-chemical and biological properties	Hard	Dragon
Drug-like indices	Dummy variables taking value equal to one when all the criteria of the consensus definition of a drug-like molecule are satisfied, 0 otherwise	Easy	Dragon
CATS 2D	Similar to 2D atom pairs but assigning atoms to defined pharmacophore point types	Easy	Dragon
CATS 3D	Similar to 3D atom pairs but assigning atoms to defined pharmacophore point types	Easy	Dragon
FP	Fingerprint	Hard	Dragon/Chemaxon/RDKit
ECFP	Extended connectivity fingerprint	Hard	Dragon/Chemaxon/CDK/Canvas
CHF	Chemical hashed fingerprint	Hard	Chemaxon
PF	Pharmacophore fingerprint	Hard	Chemaxon
RF	Reaction fingerprint	Hard	Chemaxon
MACCS-116 FP	MACCS-166 fingerprint	Hard	Chemaxon/OpenBabel/RDKit/Canvas
LINGO FP	LINGO fingerprint	Hard	OEChemTK/CDK
Daylight-like FP	Path fingerprint	Hard	OEChemTK/CDK/Canvas
Tree FP	Daylight-like with non-linear, “tree” fragments	Hard	OEChemTK/OpenBabel
Pharmacophoric FP	Pharmacophoric fingerprints	Hard	Chemaxon
MOLPRINT2D	MOLPRINT2D		OpenBabel
Acidic group count	/	/	PaDEL
ALOGP	/	/	PaDEL

APol	/	/	PaDEL
Aromatic atoms count	Count aromatic atoms	Easy	PaDEL
Aromatic bonds count	Count aromatic bonds	Easy	PaDEL
Atom count	Count atoms	Easy	PaDEL
Barysz matrix	/	/	PaDEL
Basic group count	Count groups	Easy	PaDEL
BCUT	/	/	PaDEL
Bond count	Count Bonds	Easy	PaDEL
BPol	/	/	PaDEL
Carbon types	Count carbon types	/	PaDEL
Chi chain	/	/	PaDEL
Chi cluster	/	/	PaDEL
Chi path cluster	/	/	PaDEL
Chi path	/	/	PaDEL
Crippen logP and MR	/	/	PaDEL
Detour matrix	/	/	PaDEL
Eccentric connectivity index	/	/	PaDEL
Atom type electrotopological state	/	/	PaDEL
Extended topochemical atom	/	/	PaDEL
FMFDescriptor	/	/	PaDEL
Fragment complexity	/	/	PaDEL
Hbond acceptor count	/	/	PaDEL
Hbond donor count	/	/	PaDEL
Hybridization ratio	/	/	PaDEL
Information content	/	/	PaDEL
Kappa shape indices	/	/	PaDEL
Largest chain	/	/	PaDEL
Largest Pi system	/	/	PaDEL
Longest aliphatic chain	/	/	PaDEL
Mannhold LogP	/	/	PaDEL
McGowan volume	/	/	PaDEL
Molecular distance edge	/	/	PaDEL
Molecular linear free energy relation	/	/	PaDEL
Petitjean number	/	/	PaDEL
Rotatable bonds count	/	/	PaDEL
Rule of five			
Wiener numbers	/	/	PaDEL
XLogP	/	/	PaDEL
Zagreb index	/	/	PaDEL

Gravitational index	/	/	PaDEL
Length over breadth	/	/	PaDEL
Moment of inertia	/	/	PaDEL
Petitjean shape index	/	/	PaDEL

Dragon: https://chm.kode-solutions.net/

Chemaxon: https://docs.chemaxon.com/display/docs/chemical-fingerprints.md

PaDEL: http://www.yapcwsoft.com/dd/padeldescriptor/

RDKit: https://www.rdkit.org/docs/source/rdkit.Chem.EState.Fingerprinter.html

OEChemTK: https://www.eyesopen.com/oechem-tk

OpenBabel: http://openbabel.org/wiki/Main_Page

CDK: https://sourceforge.net/projects/cdk/

Canvas: https://www.schrodinger.com/Canvas/

Section S1 Outlier detection algorithms

In this section, 4 outlier detection algorithms are firstly chosen to be introduced in details. Then the codes about how to employ the outlier detection algorithms in Python codes will be illustrated, considering the readers' interests of their utilizations. Finally, we will introduce our inhouse integrated module to utilize the currently outlier methods tools to perform a fast searching for the optimal outlier method and parameters.

Local outlier factor (LOF)

LOF method measures the so-called local density deviation of each data point with respect to its neighbors in a given data set. The outlier samples could be detected by inspecting their lower density than their neighbors.[1]

Firstly, the terms of *o, p, q* are denoted as sample points, and the term *C* displays a set of samples. The term $d(p,q)$ is the distance between samples *p* and *q*, while $d(p,C)$ stands for the minimum distance between sample *p* and the sample in *C*. Thence, according to the work of Breunig *et al.*[1], the notion of *k-distance* of sample *p*, denoted as *k-distance(p)*, is defined as the distance $d(p,o)$ between sample *p* and the sample *o* in data set (denoted as *D*), in which at least *k* samples $o' \in D \setminus \{p\}$ hold that $d(p,o') \leq d(p,o)$ and at most *k*-1 samples $o' \in D \setminus \{p\}$ hold that $d(p,o') < d(p,o)$. The term $D \setminus \{p\}$ indicates that the data set *D* excluding sample *p*. Given the notion of *k-distance(p)*, it could be noted that the *k-distance neighborhood* of sample *p* contains every sample whose distance from *p* is lower than *k-distance(p)*, which could be defined as equation 1.

$$N_k(p) = N_{k-distance(p)} = \{q \in D \setminus \{p\} \mid d(p,q) \leq dist_k(p)\} \#(1)$$

To simplify the notations, the $N_{k-distance(p)}$ is represeneted as $N_k(p)$, signaling the numeber of *k-distance neighborhood* of sample *p*, while $dist_k(p)$ is the shorthand of *k-distance(p)*.

Then, the notion of *reachability distance* of sample *p* is defined as equation 2.

$$dist_k^{reach}(p,o) = \max\{dist_k(o), d(p,o)\} \#(2)$$

The *local reachability density* (*lrd*) of a sample *p* could be defined based on equation 2 as equation 3.

$$lrd_k(p) = \left(\frac{\sum_{o \in N_k(p)} dist_k^{reach}(p,o)}{|N_k(p)|} \right)^{-1} \#(3)$$

The *lrd* of a sample *p* could reflect the inverse of the average *reachability distance* of sample *p*. And at last, the *local outlier factor* (*lof*) of sample *p* is defined as equation 4.

$$lof_k(p) = \frac{\sum_{o \in N_k(p)} \frac{lrd_k(o)}{lrd_k(p)}}{|N_k(p)|} \#(4)$$

The *lof* of sample *p* is the average of the ratio of the local reachability density of *p* and those of the *k* neighbors of *p*. The lower *lrd* of *p*, the higher *lrd* of sample *o* will be and thence the higher *lof* of *p* will be, which indicates the higher probability of *p* to be an outlier.

Isolated forest (iForest)

By taking the advantages of two outliers' quantitative properties: i) they are the minority consisting of fewer instances and ii) they have attribute-values that are very different from normal samples, Liu *et al*.[2] argue that the outliers are more easily to be isolated closer to the root of a tree structure when applying the data set into a tree model, whereas the normal samples are isolated at the deeper end of the tree structure. Such the isolation characteristic of a tree model may help us to detect the outliers more effectively.

Given a data set of n samples $X = \{x_1, x_2, ..., x_n\}$, the samples X were recursively divided by randomly selecting an attribute and a split value, until the tree reaches a limited depth, or the remnant sample set could not be divided anymore. Thus, the *path length* $h(x)$ for a sample *x* could be defined as the number of edges *x* traverses a tree structure from the root node to a terminate node. The outlier score is then defined based on $h(x)$ as equation 5.

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \#(5)$$

$E(h(x))$ is the average of $h(x)$ from a collection of fitted tree structures, while $c(n)$ is calculated based on the data set of *n* samples as equation 6.

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \#(6)$$

$H(i)$ is the harmonic number and could be estimated by $\ln(i) + 0.5772156649$.

By sorting the samples by the order of outlier score in equation 5, the outliers could be easily detected according to the top of the ranked sample list.

One-class support vector machine (one-class SVM)

Similar to the binary support vector classification task (SVC), the idea of one-class SVM is determining a decision function to discriminate the binary sample classes, namely the normal and outlier samples, in which the decision function should cover the most normal samples and the least outlier samples as possible.[3] According to the work of Schölkopf *et al*.[3] and Chang *et al*.[4], given training vectors $\boldsymbol{x}_i \in R^n, i = 1, ..., l$ without classification labels, the primal problem of one-class SVM is to minimize the value of equation 7.

$$\min_{\boldsymbol{\omega}, \xi, \rho} \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} - \rho + \frac{1}{\nu l}\sum_{i=1}^{l}\xi_i \#(7)$$

$$subject\ to = \boldsymbol{\omega}^T\phi(\boldsymbol{x_i}) \geq \rho - \xi_i$$
$$= \xi_i \geq 0, i = 1, ..., l$$

In equation 1, $\phi(\boldsymbol{x_i})$ maps $\boldsymbol{x_i}$ into a higher-dimensional space. $\boldsymbol{\omega}$ and ρ are a weight vector and an offset parameterizing a hyperplane. Thence $\boldsymbol{\omega}^T\phi(\boldsymbol{x_i})$ determines the decision function in the feature space. ξ_i is slack variable that is used to introduce the "soft margin" of decision function, which aims allow for some errors. Parameter ν is introduced to control the size of soft margin. To solve equation 1, a Lagrangian could be introduced as equation 8, in which α, β are constants respectively.

$$L(\omega, \xi, \rho, \alpha, \beta) = \frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{\omega} - \rho + \frac{1}{\nu l}\sum_{i=1}^{l}\xi_i$$

$$= -\sum_{i=1}^{l}\alpha_i(\boldsymbol{\omega}^T\phi(\boldsymbol{x_i}) - \rho + \xi_i) - \sum_{i=1}^{l}\beta_i\xi_i \#(8)$$

The derivatives with respect to the primal variables, yielding equation 4.

$$\boldsymbol{\omega} = \sum_{i=1}^{l}\alpha_i\phi(\boldsymbol{x_i})$$

$$\alpha_i = \frac{1}{\nu l} - \beta_i \leq \frac{1}{\nu l} \#(9)$$

The decision function could be equation 10, in which the $sgn(z)$ equals 1 for $z \geq 0$ and -1 other wise.

$$sgn\left(\sum_{i=1}^{l}\boldsymbol{\omega}^T\phi(\boldsymbol{x_i}) - \rho\right) = sgn\left(\sum_{i=1}^{l}\alpha_i\,\phi(\boldsymbol{x_i})^T\phi(\boldsymbol{x_i}) - \rho\right) \#(10)$$

Thence, the outlier samples will be detected depending on the value of decision function

Extreme gradient boosting outlier detection (XGBOD)

Zhao *et al.*[5] proposed a supervised three-phase framework embedding extreme gradient boosting (XGBoost) algorithm, which is named XGBoost outlier detection (XGBOD).

In phase 1, given a data set of *n* samples and *d* features $X \in R^{n\times d}$, a set of unsupervised outlier detection methods are regarded as outlier scoring functions $\Phi(\cdot)$ for XGBOD, where each function $\Phi_i(\cdot)$ would output a vector consisting of the 1 or 0 values on data set X as the *transformed outlier scores* (*TOS*) to describe the outlying degree for each sample. Combing *k* outlier scoring functions, *k TOS* vectors could be constructed as new matrix $\Phi = [\Phi_1, ..., \Phi_k]$ which would be used as the input matrix for the next phase.

Phase 2 is to pick up *p* ($p<k$) *TOS* vectors from the matrix Φ, based on the consideration that not all the *TOS* vectors would contain useful information to detect the outliers. Two selection methods are considered. The first is the *accurate selection* that selects top *p* most accurate *TOS* vectors, in which the accuracy measurement could be chosen from any accuracy metrics such as the area under receiver operating characteristic curve (ROC). Let the $ACC_i(\cdot)$ as the accuracy metric function, each *TOC* vectors could be evaluated by using equation 11 along with the true labels *y* (this is the reason why XGBOD is a supervised method).

$$ACC_i = ROC(\Phi_i, y) \#(11)$$

The second selection method is *balance selection* that maintains the balance between diverse and accurate *TOS* vectors, in which the Pearson correlation $\rho(\Phi_i, \Phi_j)$ between two *TOC* vectors Φ_i and Φ_j is introduced into equation 11, as displayed in equation 12.

$$\Psi(\Phi_i) = \frac{ACC_i}{\sum_{j=1}^{S} \left|\rho\left(\Phi_i, \Phi_j\right)\right|} \#(12)$$
$$subject\ to\ \Phi_j \in \{S\}, ACC_i > 0$$

S means the selected *TOS* vector set while $\Psi(\Phi_i)$ is the defined discounted accuracy function. The Pearson correlation between a *TOS* of Φ_i (TOS_i) and the *TOS* in S (TOS_S) is aggregated as $\sum_{j=1}^{S} \left|\rho\left(\Phi_i, \Phi_j\right)\right|$. If TOS_i is highly related to TOS_S, the accuracy of TOS_i will be discounted, which thereby discouraging to select the similar *TOS* vector and increase the account of diverse *TOS* vector. After each *TOS* vector is iteratively evaluated by equation 12, the top p *TOS* vectors with the highest discounted accuracies will be selected for phase 3.

At last, an XGBoost classifier is applied to train the classification model based on the selected *TOS* matrix, in which the XGBoost could be replaced by other classifiers. Then the model could be used to predict the outlier sample for unknown data set.

Minimum covariance determinant (MCD)

Minimum covariance determinant (MCD) method[6] determines an outlier in a given training vectors $\boldsymbol{x}_i \in R^n, i = 1, ..., l$ with p features by inspecting its *Mahalanobis distance* (MD):

$$MD(\boldsymbol{x}_i) = \sqrt{(\boldsymbol{x}_i - \overline{\boldsymbol{x}})^T \boldsymbol{S}^{-1} (\boldsymbol{x}_i - \overline{\boldsymbol{x}})} \#(13)$$

Herein, $\overline{\boldsymbol{x}}$ is the mean value of training vectors, $\boldsymbol{S}$ is the classical covariance matrix of the training set of l samples. To overcome the so-called *masking effect* that the resulting MD values are highly affected by the outlying points and no longer detect the outliers, the mean value and covariance matrix are usually calculated based on a subset of training set that includes as few outliers as possible. Considering a subset $\boldsymbol{H}$ of h training samples selected from l samples, the modified *Mahalanobis distance* is defined as:

$$MD(\boldsymbol{x}_i) = \sqrt{(\boldsymbol{x}_i - \overline{\boldsymbol{h}})^T \boldsymbol{S}_{\boldsymbol{h}}^{-1} (\boldsymbol{x}_i - \overline{\boldsymbol{h}})} \#(14)$$

Herein, $\overline{\boldsymbol{h}}$ is the mean value of h training vectors, $\boldsymbol{S}_{\boldsymbol{h}}$ is the covariance matrix of the training subset $\boldsymbol{H}$. The objective of MCD method is to find h samples whose classical covariance matrix has the lowest determinant, which corresponds to find the subset concentrating the most likely samples. To search an optimal subset $\boldsymbol{H}$ efficiently, Rousseeuw *et al.*[7] proposed a fast-MCD algorithms as followed:

1) Draw a random h-subset $\boldsymbol{H}$, and compute its $\overline{\boldsymbol{h}}$ and $\boldsymbol{S}_{\boldsymbol{h}}^{-1}$;

2) Compute the MD for each sample, and select the top h samples with lowest MD values to form a new h-subset $\boldsymbol{H}$;

3) Repeat step 1 ~ 2 for *e.g.*, 10 times (or the $\boldsymbol{S}_{\boldsymbol{h}}$ determinant reaches 0 or keeps unchanged), and store the results;

4) Repeat step 1 ~ 3 for *e.g.*, 30 times, and select *e.g.*, 10 h-subset $\boldsymbol{H}$ whose $\boldsymbol{S}_{\boldsymbol{h}}$ has the lowest determinants;

5) Iterate the selected 10 h-subset $\boldsymbol{H}$ as the starts to exert step 1 ~ 2 and store the results;

6) Select one h-subset $\boldsymbol{H}$ whose $\boldsymbol{S}_{\boldsymbol{h}}$ has the lowest determinant;

7) Output the corresponding MD values and determine the outliers.

Angle-based outlier detection (ABOD)

Angle-based outlier detection (ABOD)[8] utilizes the angles between difference vectors to pairs of other points to evaluate each sample point. As for a point within a cluster, the most angles might differ widely, and exhibit the higher variance. The angles of an outlier to most pairs of

points will be small since most normal points are clustered in some directions, showing the lower variance. The algorithm of ABOD could be summarized as followed:

1) Select a sample from the data set and calculate the variance of all the angles between difference vectors to pairs of other points;

2) Iterate all the samples;

3) Sort the variances and determine the outliers.

Performing outlier detection in code

Compared to the algorithm details of outlier detection, the usage on how to apply them into practice may be more appealing to us. As far as all we know, the Python tools PyOD and scikit-learn are the most commonly used packages to perform the outlier detection, which have involved the majority of algorithms (including the 4 algorithms that we have introduced above). Table S 2 list all the algorithms that are extracted from PyOD and scikit-learn, and Table S 3 displays the outlier detecting Python tools that we have known. Since PyOD has covered the most algorithms among the other Python packages, we will show examples mainly on it.

Figure S 1 exhibits the code to perform the outlier detection, including LOF, iForest and OCSVM. The outlier detection method could be chosen by switching the value of "outlier_detection". The toy data set is generated from the module "sklearn.datasets.make_regression" with 100 samples and 5 features as the original data set. The leaving-one-out cross-validation (LOOCV) determination coefficient (R^2) of original data set by using XGBoost algorithm is 0.792. The original data set is then transferred to the outlier module along with the hyper-parameter "contamination" that stands for the account of outlier samples and needs to be optimized by the users. The values of "contamination" are set 0.03 for OCSVM, 0.04 for LOF, and 0.04 for iForest respectively. Then we could obtain the pruned data set and evaluate it by using LOOCV validation and XGBoost, yielding the LOOCV R^2 of 0.820 for OCSVM, 0.813 for LOF, and 0.828 for iForest respectively.

Table S2 Outlier detection methods. Extracted from https://github.com/yzhao062/pyod and http://scikit-learn.org/stable/modules/outlier_detection.html.

Algorithm	Source
Minimum Covariance Determinant (MCD)	PyOD, scikit-learn
One-Class Support Vector Machines	PyOD, scikit-learn, libsvm
Deviation-based Outlier Detection	PyOD
Local Outlier Factor	PyOD, scikit-learn
Connectivity-Based Outlier Factor	PyOD
Memory Efficient Connectivity-Based Outlier Factor	PyOD
Clustering-Based Local Outlier Factor	PyOD
LOCI: Fast outlier detection using the local correlation integral	PyOD
Histogram-based Outlier Score	PyOD
Subspace Outlier Detection	PyOD
Rotation-based Outlier Detection	PyOD
Angle-Based Outlier Detection	PyOD
COPOD: Copula-Based Outlier Detection	PyOD
Fast Angle-Based Outlier Detection using	PyOD

approximation	
Median Absolute Deviation (MAD)	PyOD
Stochastic Outlier Selection	PyOD
Isolation Forest	PyOD, scikit-learn
Feature Bagging	PyOD
LSCP: Locally Selective Combination of Parallel Outlier Ensembles	PyOD
Extreme Boosting Based Outlier Detection (Supervised)	PyOD
Lightweight On-line Detector of Anomalies	PyOD
Fully connected AutoEncoder (use reconstruction error as the outlier score)	PyOD
Variational AutoEncoder (use reconstruction error as the outlier score)	PyOD
Variational AutoEncoder (all customized loss term by varying gamma and capacity)	PyOD
Single-Objective Generative Adversarial Active Learning	PyOD
Multiple-Objective Generative Adversarial Active Learning	PyOD
Deep One-Class Classification	PyOD

Table S3 Python tools for outlier detections. Extracted from https://github.com/yzhao062/anomaly-detection-resources.

Outlier detection tools	URL
PyOD	https://github.com/yzhao062/pyod
scikit-learn	http://scikit-learn.org/stable/modules/outlier_detection.html
PySAD	https://github.com/selimfirat/pysad
SUOD	https://github.com/yzhao062/suod
alibi-detect	https://github.com/SeldonIO/alibi-detect

```
# pip install fast-machine-learning pyod sklearn xgboost
from fml.data import DataObject
from fml.validates import Validate
from sklearn.datasets import make_regression
from xgboost import XGBRegressor

# import OCSVM, LOF, IForest
from pyod.models.ocsvm import OCSVM
from pyod.models.lof import LOF
from pyod.models.iforest import IForest

outlier_detection = IForest
model = XGBRegressor

# Create data
X, Y = make_regression(n_samples=100, n_features=5, noise=1.0, n_informative=5, random_state=0)
# Perform leaving-one-out cross-validation (LOOCV) by using fml.validates.Validate
result = Validate(model, DataObject(X, Y)).validate_loo().loo_result
# Print the original LOOCV R2 (determination coefficient)
# Original leaving-one-out R2 is 0.792
print(f"Original leaving-one-out R2 is {round(result['r2_score'], 3)}")

# Fit the class of OCSVM, or LOF, or IForest
# contamination is the hyper-parameter determining the outlier account
# The value of contamination is 0.03 (OCSVM), 0.04 (LOF), 0.04 (IForest)
clf = outlier_detection(contamination=0.04)
clf.fit(X)
y_train_pred = clf.labels_

# Obtain the remnant samples
X = X[(y_train_pred-1).astype(bool)]
Y = Y[(y_train_pred-1).astype(bool)]
# Print the remnant samples
# New X shape: 97x5 (OCSVM), 96x5(LOF), 96x5 (IForest)
print(f"New X shape: {X.shape}")

# Perform the LOOCV based on remnant samples
result = Validate(model, DataObject(X, Y)).validate_loo().loo_result
# New leaving-one-out R2 is 0.82 (OCSVM), 0.813 (LOF), 0.828 (IForest)
print(f"New leaving-one-out R2 is {round(result['r2_score'], 3)}")
```

Figure S1 Code to perform outlier detection.

Selecting an optimal outlier detection in code

Dazzling by the various outlier methods in Figure S 2, we may feel confused to select a suitable method along with its parameter “contaminant”. Thence, we developed a simple integrated module in our inhouse package[9] to perform a fast searching for an optimal outlier method from the dazzling method choices provided from PyOD, which covers at least 17 unsupervised outlier methods.

The integral code has little difference from Figure S 1, except that:

a) we import the module “fml.outlier.HpOutlierDetect” instead of the methods provided from PyOD, which is marked in Figure S 2 a;

b) Use the “HpOutlierDetect” to perform the searching for the optimal outlier method and contamination value, which is marked in Figure S 2 b. The best result could be found in “best_param”;

c) Use the optimal outlier method and contamination value directly from the step b, which is marked in Figure S 2 c.

After searching automatically and quickly, we may find that the size of pruned data set is reduced to 88 samples along with the more favorable LOOCV R^2 and LOOCV RMSE values of 0.849 and 55.533.

```
# pip install fast-machine-learning pyod sklearn xgboost
from fml.data import DataObject
from fml.validates import Validate
from sklearn.datasets import make_regression
from xgboost import XGBRegressor

# Use the module fml.outlier.HpOuterlierDetect instead of the outlier method in PyOD
from fml.outlier import HpOuterlierDetect

model = XGBRegressor

# Create data
X, Y = make_regression(n_samples=100, n_features=5, noise=1.0, n_informative=5, random_state=0)
# Perform leaving-one-out cross-validation (LOOCV) by using fml.validates.Validate
result = Validate(model, DataObject(X, Y)).validate_loo().loo_result
# Print the original LOOCV R2 (determination coefficient)
# Original leaving-one-out R2 is 0.792
print(f"Original leaving-one-out R2 is {round(result['r2_score'], 3)}")
# Original leaving-one-out RMSE is 66.162
print(f"Original leaving-one-out RMSE is {round(result['rmse'], 3)}")

# Use class HpOuterlierDetect to search the optimal outlier method and contamination
hpoutlier = HpOuterlierDetect(verbose=True).fit(model, DataObject(X=X, Y=Y), cv=True)
# Print the best params: in our test, the result is [pyod.models.pca.PCA, 0.16565947420384478]
# The result might different since the searching is not global
best_params = hpoutlier.best_params
print(f"{best_params}")

# Use the optimal outlier method and contamination
clf = best_params[0](contamination=best_params[1])
clf.fit(X)
y_train_pred = clf.labels_

# Obtain the remnant samples
X = X[(y_train_pred-1).astype(bool)]
Y = Y[(y_train_pred-1).astype(bool)]
# Print the remnant samples
# New X shape: 88x5, the result might differ due to the searching randomness
print(f"New X shape: {X.shape}")

# Perform the LOOCV based on remnant samples
result = Validate(model, DataObject(X, Y)).validate_loo().loo_result
# New leaving-one-out R2 is 0.849
# New leaving-one-out RMSE is 55.533
# The result might differ to the searching randomness
print(f"New leaving-one-out R2 is {round(result['r2_score'], 3)}")
```

a

b

c

Figure S2 Code to perform a fast searching for the optimal outlier method and its parameter

Section S2 Illustration for Mor14p, Mor24m, R2s

Mor14p and Mor24m

Mor14p and Mor24m belong to the descriptor "3D-molecule representation based on electron diffraction study (MORSE)" from Dragon software. Their meanings might be hard to be understood. For the adequate comprehension for this feature, the background would be introduced tightly. The detail information could be found in the original papers.[10,11]

Taking Mor14p as the start, let's see the original definition as:

$$\mathrm{Mor14p} = \sum_{i=2}^{N}\sum_{j=1}^{i=1} P_i P_j \frac{sin\left(13r_{ij}\right)}{13r_{ij}} \quad \#(13)$$

where P is carbon-scaled polarizability, r is the Euclidean distance between ith and jth atoms. Each summand in this descriptor mainly depends on distance r and thence could be considered as a radial basis function itself, which could be treated as:

$$f(r) = P_1 P_2 \frac{\sin\left(13r\right)}{13r} \#(14)$$

where $f(r)$ represents the radial basis function of atomic pairs as well as the summand in Mor14p.

Hence, Mor14p could be treated as the sum of contributions by different atomic pairs, which could be rewrote as (taking a system of C, N as the example):

$$\mathrm{Mor14p} = \sum P_C P_C \frac{\sin(13r)}{13r} + \sum P_C P_N \frac{\sin(13r)}{13r} + \sum P_N P_N \frac{\sin(13r)}{13r} \#(15)$$

Then for each atomic pair, there are different distances:

$$\sum P_C P_C \frac{\sin(13r)}{13r} = P_c P_c \times \left(\frac{\sin(13r_1)}{13r_1} + \frac{\sin(13r_2)}{13r_2} + \ldots \right) \#(16)$$

Considering the data set (targeted to power convention efficiency PCE) in our work[12], three atomic pairs of C-S, C-C and C-O are mainly influencing Mor14p. Thence the three determined radial basis functions f(r) are plotted in Figure S 3. We could deduce that the positive values of functions f(r) would lead to positive contribution for Mor14p and the contrary for PCE values. Thus, the most favorable atomic pairs are located at the distances about 1.329, 1.812 Å while the most detrimental pairs are located at 1.087, 1.571 Å.

There is the same condition for Mor24m which belongs to the vertical model. Mor24m could be defined as:

$$\mathrm{Mor24m} = \sum_{i=2}^{N} \sum_{j=1}^{i=1} m_i m_j \frac{sin\left(23r_{ij}\right)}{23r_{ij}} \#(17)$$

where m is carbon-scaled atomic mass and the others are the same in Mor14p. Similarly, the summand could be treated as a radial basis function:

$$f(r) = m_1 m_2 \frac{sin23r}{23r} \#(18)$$

where the $f(r)$ is the summand in Mor23m.

In our data set[12], two atomic pairs of C-S and C-O are considered. The determined radial basis functions $f(r)$ are presented in Figure S 4. Due to the positive coefficient of Mor24m, the most favorable atomic pairs are located at the distance around 1.161, 1.434, 1.707, 1.981 Å, while the most detrimental pairs are at 1.024, 1.298, 1.571, 1.844 Å.

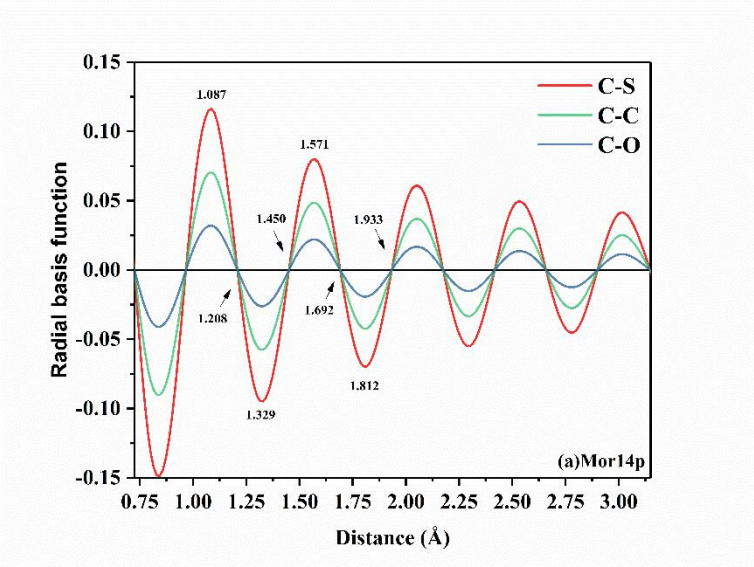

Figure S 3 Radial basis functions of Mor14p descriptor corresponding to different atomic pairs

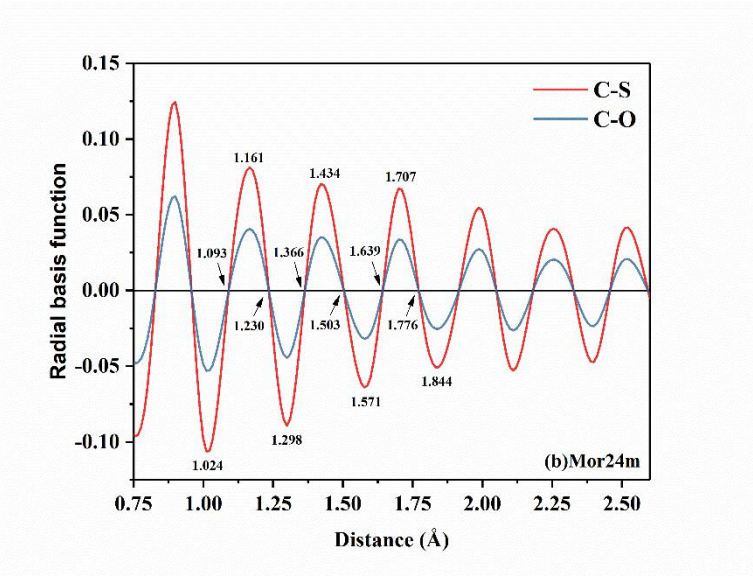

Figure S 4 Radial basis functions of Mor24m descriptor corresponding to different atomic pairs

R2s

R2s belongs to GETAWAY descriptors, which was defined by Consonni *et al.*[13,14] Before understanding the meaning of R2s, more backgrounds should be acknowledged. The first should be introduced is the molecular influence matrix **H** (MIM), which is defined as:

$$\mathbf{H} = \mathbf{M}\,(\mathbf{M^T}\,\mathbf{M})^{-1}\,\mathbf{M}^{\mathrm{T}}\#(19)$$

where **M** is the molecular matrix constituted by A rows corresponding to the atoms in a molecule (hydrogen included) and three columns corresponding to the Cartesian coordinates x, y, z of each atom in optimized molecular structure. Atomic coordinates are assumed to be calculated with respect to the geometrical center of the molecule to obtain translational invariance. The matrix **H** is a symmetric $A \times A$ matrix, where A represents the number of atoms, with the following mathematical properties:

$$0 \le h_{ii} \le 1;\ \sum_{i=1}^{A} h_{ii} = 3\ (only\ for\ 3D - molecules)\#(20)$$

where h denotes the elements of the MIM. The diagonal elements h_{ii}, called leverages, encode atomic information and represent the "influence" of each molecule atom in determining the whole shape of the molecule. The mantle atoms always have higher h_{ii} values than the atoms near the molecule geometric center.

Then, based on the MIM, the influence/distance matrix **R** is determined as following:

$$[\mathbf{R}]_{ij} \equiv \left[\frac{\sqrt{h_{ii}h_{jj}}}{r_{ij}}\right]_{ij}\ i \neq j\#(21)$$

where r_{ij} is the geometric distance of ith and jth atoms. The diagonal elements of the matrix **R** are zero. In one molecule, the largest values of the matrix elements derive from the most external atoms that are simultaneously next to each other in the molecular space. The lower values of $[\mathbf{R}]_{ij}$ is dependent on the atoms near molecule geometric center.

Then we could define R2u descriptor as:

$$R2(u) = \sum_{i-1}^{A-1} \sum_{j>i} [\mathbf{R}]_{ij} \delta(2; d_{ij}) \#(22)$$

where u means unweighted, d_{ij} is the topological distance between atoms i and j, $\delta(2; d_{ij})$ is Dirac-delta function (δ=1 if d_{ij} =2, zero otherwise). Apparently, R2u is the sum of the elements in **R** matrix, where the values are determined by the atomic leverages and interatom distances. To reduce R2u values, the atom density near molecule geometric center should be higher and the mantle density is prone to be lower.

It's very close to the subjected descriptor R2s, where s substitutes u. The s represents intrinsic state of the ith atom (I-state) which is defined as:

$$s_i = \frac{\left(\frac{2}{L_i}\right)^2 \cdot \delta^v + 1}{\delta_i} \#(23)$$

where L is the principal quantum number, δ^v is the number of valence electrons, δ is the number of sigma electrons in the H-depleted molecular structure. Disregarding the bond types, the I-state values of the elements in the periodic table tend to be larger, such as R-**C**(2.5)≡R, R≡**N**(6.0), R-**F**(8.0), R-**Si**(1.389)≡R. In the terms of bond types, the higher bonds, the higher the I-state would be, such as R-**C**(2.5)≡R, R-**C**(2)H=R, R-**C**$_{ar}$(1.667), R-**C**(1.5)H2=R. Since the I-state is only defined for non-H atoms, Dragon fixes the value of 1 for hydrogen.

So, R2s is the combination of R2u and I-state, which is formulated as:

$$R2(s) = \sum_{i-1}^{A-1} \sum_{j>i} [\mathbf{R}]_{ij} \cdot s_i \cdot s_j \cdot \delta(2; d_{ij}) \#(24)$$

In our work[12], to compress the effect of R2s, the proportion of atoms with high I-state should be lowered, such as R=**O**(7.0), R-**F**(8.0), R≡**N**(6.0) while the density of geometric centered atoms should be increased and the vice versa.

Section S3 Atomic parameters

Table S 4 lists the ionic radii for various ions from the work of Pauling[15], Sanderson[16], Белов[17], Ahrens[17], Kordes[17], Goldschmidt[18], Quill[19], Zachariasen[20], and Chen[21]. Table S 5 lists covalent radii from the work of Белов[17]. Table S 6 ~ Table S 10 list ionization energies, metal radii, valence electrons to covalent radius ratios, electronegativities and equivalent conductance from the work of Chen[21].

Table S4 Ionic radius

Ion	Ionic radius (Å)	Source
H^-	2.08	Pauling
Li^+	0.60	Pauling
	0.69	Sanderson
	0.68	Белов-Бокий
	0.68	Ahrens
	0.71	Kordes
Be^{2+}	0.31	Pauling
	0.49	Sanderson
	0.34	Белов-Бокий

	0.35	Ahrens
	0.38	Kordes
B^{3+}	0.20	Pauling
	0.41	Sanderson
	0.20	Белов-Бокий
	0.23	Ahrens
	0.25	Kordes
C^{4+}	0.15	Pauling
	0.34	Sanderson
	0.20	Белов-Бокий
	0.16	Ahrens
	0.18	Kordes
C^{4-}	2.60	Pauling
N^{5+}	0.11	Pauling
	0.31	Sanderson
	0.15	Белов-Бокий
	0.13	Ahrens
	0.14	Kordes
N^{3-}	1.71	Pauling
O^{6+}	0.09	Pauling
O^{2-}	1.40	Pauling
F^{7+}	0.07	Pauling
F^{-}	1.36	Pauling
Na^{+}	0.95	Pauling
	0.99	Sanderson
	0.98	Белов-Бокий
	0.97	Ahrens
	0.95	Kordes
Mg^{2+}	0.65	Pauling
	0.75	Sanderson
	0.74	Белов-Бокий
	0.66	Ahrens
	0.66	Kordes
Al^{3+}	0.50	Pauling
	0.62	Sanderson
	0.57	Белов-Бокий
	0.51	Ahrens
	0.52	Kordes
Si^{4+}	0.41	Pauling
	0.62	Sanderson
	0.57	Белов-Бокий
	0.51	Ahrens
	0.52	Kordes

Si^{4-}	2.71	Pauling
P^{5+}	0.34	Pauling
	0.48	Sanderson
	0.35	Белов-Бокий
	0.35	Ahrens
	0.34	Kordes
P^{-3}	2.12	Pauling
S^{6+}	0.29	Pauling
	0.43	Sanderson
	0.29	Белов-Бокий
	0.30	Ahrens
	0.31	Kordes
S^{2-}	1.84	Pauling
Cl^{7+}	0.26	Pauling
Cl^{-}	1.81	Pauling
K^{+}	1.33	Pauling
	1.33	Sanderson
	1.33	Белов-Бокий
	1.33	Ahrens
	1.33	Kordes
Ca^{2+}	0.99	Pauling
	1.00	Sanderson
	1.04	Белов-Бокий
	0.99	Ahrens
	0.99	Kordes
Sc^{3+}	0.81	Pauling
	0.79	Sanderson
	0.83	Белов-Бокий
	0.81	Ahrens
	0.81	Kordes
Ge^{4+}	0.53	Pauling
Ge^{4-}	2.72	Pauling
As^{5+}	0.47	Pauling
	0.65	Sanderson
	0.47	Белов-Бокий
	0.46	Ahrens
	0.46	Kordes
As^{3-}	2.22	Pauling
Se^{6+}	0.42	Pauling
	0.60	Sanderson
	0.35	Белов-Бокий
	0.42	Ahrens
	0.41	Kordes

Se^{2-}	1.98	Pauling
Br^{7+}	0.39	Pauling
Br^{-}	1.95	Pauling
Ti^{4+}	0.68	Pauling
	0.69	Sanderson
	0.64	Белов-Бокий
	0.68	Ahrens
	0.68	Kordes
V^{5+}	0.59	Pauling
	0.62	Sanderson
	0.40	Белов-Бокий
	0.59	Ahrens
	0.59	Kordes
Cr^{6+}	0.52	Pauling
	0.58	Sanderson
	0.35	Белов-Бокий
	0.52	Ahrens
	0.52	Kordes
Cu^{+}	0.96	Pauling
	0.96	Sanderson
	0.98	Белов-Бокий
	0.96	Ahrens
	0.93	Kordes
Zn^{2+}	0.74	Pauling
	0.84	Sanderson
	0.83	Белов-Бокий
	0.74	Ahrens
	0.72	Kordes
Ga^{3+}	0.62	Pauling
	0.75	Sanderson
	0.62	Белов-Бокий
	0.62	Ahrens
	0.60	Kordes
Rb^{+}	1.48	Pauling
	1.46	Sanderson
	1.49	Белов-Бокий
	1.47	Ahrens
	1.47	Kordes
Sr^{2+}	1.13	Pauling
	1.09	Sanderson
	1.20	Белов-Бокий
	1.12	Ahrens
	1.15	Kordes

Y^{3+}	0.93	Pauling
	0.88	Sanderson
	0.97	Белов-Бокий
	0.92	Ahrens
	0.96	Kordes
Sn^{4+}	0.71	Pauling
	0.77	Sanderson
	0.67	Белов-Бокий
	0.71	Ahrens
	0.71	Kordes
Sn^{4-}	2.94	Pauling
Sb^{5+}	0.62	Pauling
	0.72	Sanderson
	0.62	Белов-Бокий
	0.62	Ahrens
	0.63	Kordes
Sb^{3-}	2.45	Pauling
Te^{6+}	0.56	Pauling
	0.67	Sanderson
	0.56	Белов-Бокий
	0.56	Ahrens
	0.57	Kordes
Te^{2-}	2.21	Pauling
I^{7+}	0.50	Pauling
I^{-}	2.16	Pauling
Zr^{4+}	0.80	Pauling
	0.76	Sanderson
	0.82	Белов-Бокий
	0.79	Ahrens
	0.83	Kordes
Nb^{5+}	0.70	Pauling
	0.68	Sanderson
	0.66	Белов-Бокий
	0.69	Ahrens
	0.73	Kordes
Mo^{6+}	0.62	Pauling
	0.64	Sanderson
	0.65	Белов-Бокий
	0.62	Ahrens
	0.65	Kordes
Ag^{+}	1.26	Pauling
	1.08	Sanderson
	1.13	Белов-Бокий

	1.26	Ahrens
	1.21	Kordes
Cd^{2+}	0.97	Pauling
	0.94	Sanderson
	0.99	Белов-Бокий
	0.97	Ahrens
	0.96	Kordes
In^{3+}	0.81	Pauling
	0.85	Sanderson
	0.92	Белов-Бокий
	0.81	Ahrens
	0.81	Kordes
Cs^{+}	1.69	Pauling
	1.64	Sanderson
	1.65	Белов-Бокий
	1.67	Ahrens
	1.74	Kordes
Ba^{2+}	1.35	Pauling
	1.24	Sanderson
	1.38	Белов-Бокий
	1.34	Ahrens
	1.37	Kordes
La^{3+}	1.15	Pauling
	0.95	Sanderson
	1.04	Белов-Бокий
	1.14	Ahrens
	1.16	Kordes
Pb^{4+}	0.84	Pauling
	0.83	Sanderson
	0.76	Белов-Бокий
	0.84	Ahrens
	0.84	Kordes
Bi^{5+}	0.74	Pauling
	0.77	Sanderson
	0.74	Белов-Бокий
	0.74	Ahrens
	0.75	Kordes
Au^{+}	1.37	Pauling
	1.11	Sanderson
	1.37	Белов-Бокий
	1.37	Ahrens
	1.37	Kordes
Hg^{2+}	1.10	Pauling

	0.97	Sanderson
	1.12	Белов-Бокий
	1.10	Ahrens
	1.10	Kordes
Tl^{+}	0.95	Pauling
	0.90	Sanderson
	1.05	Белов-Бокий
	0.95	Ahrens
	0.95	Kordes
Mn^{4+}	0.55	Sanderson
	0.46	Белов-Бокий
	0.46	Ahrens
	0.46	Kordes
Fe^{2+}	0.81	Sanderson
	0.80	Белов-Бокий
	0.74	Ahrens
	0.76	Kordes
Fe^{3+}	0.73	Sanderson
	0.67	Белов-Бокий
	0.64	Ahrens
	0.64	Kordes
Co^{2+}	0.80	Sanderson
	0.78	Белов-Бокий
	0.72	Ahrens
	0.74	Kordes
Co^{3+}	0.72	Sanderson
	0.64	Белов-Бокий
	0.63	Ahrens
	0.63	Kordes
Ni^{2+}	0.79	Sanderson
	0.74	Белов-Бокий
	0.69	Ahrens
	0.73	Kordes
Ni^{3+}	0.72	Sanderson
Ge^{4+}	0.68	Sanderson
	0.44	Белов-Бокий
	0.53	Ahrens
	0.52	Kordes
Ru^{2+}	0.85	Sanderson
Ru^{4+}	0.71	Sanderson
	0.62	Белов-Бокий
	0.67	Ahrens
	0.66	Kordes

Rh^{4+}	0.71	Sanderson
	0.65	Белов-Бокий
Rh^{3+}	0.78	Sanderson
	0.75	Белов-Бокий
	0.68	Ahrens
	0.72	Kordes
Pd^{2+}	0.88	Sanderson
	0.80	Ahrens
	1.01	Kordes
Pd^{4+}	0.73	Sanderson
	0.64	Белов-Бокий
	0.65	Ahrens
	0.74	Kordes
Hf^{4+}	0.81	Sanderson
	0.82	Белов-Бокий
	0.78	Ahrens
	0.82	Kordes
Ta^{5+}	0.72	Sanderson
	0.66	Белов-Бокий
	0.68	Ahrens
	0.74	Kordes
W^{6+}	0.68	Sanderson
	0.65	Белов-Бокий
	0.62	Ahrens
	0.68	Kordes
Re^{4+}	0.64	Sanderson
	0.57	Ahrens
Os^{2+}	0.89	Sanderson
Os^{4+}	0.75	Sanderson
	0.65	Белов-Бокий
	0.69	Ahrens
	0.69	Kordes
Ir^{2+}	0.89	Sanderson
Ir^{4+}	0.75	Sanderson
	0.65	Белов-Бокий
	0.68	Ahrens
	0.69	Kordes
Pt^{2+}	0.90	Sanderson
	0.80	Ahrens
	1.12	Kordes
Pt^{4+}	0.76	Sanderson
	0.64	Белов-Бокий
	0.65	Ahrens

	0.87	Kordes
Ce^{3+}	1.14	Quill
	1.18	Goldschmidt
Ce^{4+}	1.01	Quill
Pr^{3+}	1.12	Quill
	1.16	Goldschmidt
Nd^{3+}	1.10	Quill
	1.15	Goldschmidt
Pm^{3+}	1.08	Quill
Sm^{3+}	1.07	Quill
	1.13	Goldschmidt
Sm^{2+}	1.16	Quill
	1.26	Goldschmidt
Eu^{3+}	1.05	Quill
	1.12	Goldschmidt
Eu^{2+}	1.14	Quill
	1.24	Goldschmidt
Gd^{3+}	1.03	Quill
	1.11	Goldschmidt
Tb^{3+}	1.02	Quill
	1.09	Goldschmidt
Dy^{3+}	1.00	Quill
	1.07	Goldschmidt
Ho^{3+}	0.99	Quill
	1.05	Goldschmidt
Er^{3+}	0.98	Quill
	1.03	Goldschmidt
Th^{3+}	0.96	Quill
	1.01	Goldschmidt
Yb^{3+}	0.95	Quill
	1.00	Goldschmidt
Tl^{+}	1.49	Goldschmidt
	1.44	Pauling
Mn^{+}	0.91	Goldschmidt
	0.80	Pauling
Fe^{2+}	0.83	Goldschmidt
	0.76	Pauling
Co^{2+}	0.82	Goldschmidt
	0.69	Pauling
Cu^{2+}	0.80	Goldschmidt
Pd^{2+}	0.89	Goldschmidt
Pb^{2+}	1.32	Goldschmidt
	1.21	Pauling

Sn^{2+}	1.10	Quill
V^{2+}	0.82	Quill
W^{2+}	0.87	Quill
Mo^{2+}	0.83	Quill
Ga^{2+}	0.76	Quill
Tl^{2+}	1.19	Quill
Ti^{2+}	0.85	Quill
Ti^{3+}	0.69	Goldschmidt
	0.64	Quill
V^{3+}	0.65	Goldschmidt
	0.69	Quill
Cr^{3+}	0.64	Goldschmidt
	0.62	Quill
Mn^{3+}	0.70	Goldschmidt
	0.66	Quill
Fe^{3+}	0.67	Goldschmidt
	0.64	Quill
Co^{3+}	0.63	Quill
Ru^{3+}	0.72	Quill
Rh^{3+}	0.68	Goldschmidt
	0.72	Quill
Pd^{3+}	0.74	Quill
Pt^{3+}	0.83	Quill
P^{3+}	0.55	Quill
As^{3+}	0.69	Quill
Sb^{3+}	0.92	Quill
Bi^{3+}	1.08	Quill
V^{4+}	0.61	Goldschmidt
	0.50	Pauling
Mn^{4+}	0.52	Goldschmidt
	0.50	Pauling
Nb^{4+}	0.69	Goldschmidt
	0.67	Pauling
Mo^{4+}	0.68	Goldschmidt
	0.66	Pauling
W^{4+}	0.68	Goldschmidt
	0.66	Pauling
Ru^{4+}	0.65	Goldschmidt
	0.63	Pauling
Os^{4+}	0.67	Goldschmidt
	0.65	Pauling
Ir^{4+}	0.66	Goldschmidt
	0.64	Pauling

Bi^{4+}	0.88	Quill
Te^{4+}	0.89	Goldschmidt
	0.81	Pauling
Th^{4+}	1.10	Goldschmidt
	1.02	Pauling
	0.99	Zachariasen
Pa^{4+}	0.89	Sanderson
	0.96	Zachariasen
Pa^{5+}	0.84	Sanderson
	0.90	Zachariasen
U^{3+}	0.96	Sanderson
	1.03	Zachariasen
U^{4+}	0.89	Sanderson
	0.93	Zachariasen
U^{5+}	0.83	Sanderson
	0.87	Zachariasen
U^{6+}	0.79	Sanderson
	0.83	Zachariasen
Np^{3+}	0.96	Sanderson
	1.01	Zachariasen
Np^{4+}	0.88	Sanderson
	0.92	Zachariasen
Np^{5+}	0.83	Sanderson
	0.88	Zachariasen
Np^{6+}	0.78	Sanderson
	0.82	Zachariasen
Pu^{3+}	0.95	Sanderson
	1.00	Zachariasen
Pu^{4+}	0.88	Sanderson
	0.90	Zachariasen
Pu^{5+}	0.82	Sanderson
	0.87	Zachariasen
Pu^{6+}	0.78	Sanderson
	0.81	Zachariasen
Am^{3+}	0.95	Sanderson
	0.99	Zachariasen
Am^{4+}	0.88	Sanderson
	0.89	Zachariasen
Am^{6+}	0.86	Sanderson
	0.80	Zachariasen
CN^{-}	1.82	Chen
CNS^{-}	1.95	Chen
NO_2^{-}	1.55	Chen

HCO_2^-	1.58	Chen
$CH_3CO_2^-$	1.59	Chen
IO_3^-	1.82	Chen
NO_3^-	1.89	Chen
BrO_3^-	1.91	Chen
ClO_3^-	2.00	Chen
ClO_4^-	2.36	Chen
MnO_4^-	2.40	Chen
IO_4^-	2.49	Chen
NH_2^-	1.30	Chen
OH^-	1.40	Chen
HCO_3^-	1.63	Chen
$OH•H_2O$	1.75	Chen
HS^-	1.95	Chen
HSO_4^-	2.06	Chen
CI_3^{2-}	1.85	Chen
SO_4^{2-}	2.30	Chen
CrO_4^-	2.40	Chen
SeO_4^{2-}	2.43	Chen
MoO_4^{2-}	2.54	Chen
TeO_4^{2-}	2.54	Chen
WO_4^{2-}	2.57	Chen
BO_4^{3-}	1.91	Chen
PO_4^{3-}	2.38	Chen
AsO_4^{3-}	2.48	Chen
SbO_4^{3-}	2.60	Chen
SiO_4^{4-}	2.40	Chen
NH_4^+	1.43	Chen
PH_4^+	1.80	Chen
$N(CH_3)_4^+$	3.00	Chen
$N(C_2H_5)_4^+$	3.63	Chen

Table S5 Covalent radius from Белов-Бокий. The Roman numeral in bracket is the coordination number.

Atom	Radius
H	0.37
He	0.93
Li	1.32
Be	0.96
B	0.84
C	0.77
N	0.74
O	0.74

F	0.72
Ne	1.31
Na	1.58
Mg	1.38
Al	1.26
Si	1.17
P	1.10
S	1.04
Cl	1.00
Ar	1.74
K	2.02
Ca	1.74
Sc	1.44
Ti	1.32
V	1.21
Cr	1.19
Mn	1.19
Fe	1.20
Co	1.18
Ni	1.17
Cu	1.25
Zn	1.27
Ga	1.25
Ge	1.22
As	1.22
Se	1.17
Br	1.14
Kr	1.89
Rb	2.15
Sr	1.91
Y	1.61
Zr	1.45
Nb	1.32
Mo	1.32
Tc	1.20
Ru	1.26
Rh	1.27
Pd	1.30
Ag	1.42
Cd	1.43
In	1.46
Sn	1.40
Sb	1.41

Te	1.37
I	1.33
Xe	2.09
Cs	2.33
Ba	1.97
La	1.68
Ce	1.62
Pr	1.62
Nd	1.62
Pm	1.67
Sm	1.64
Eu	1.62
Gd	1.60
Tb	1.59
Dy	1.58
Ho	1.57
Er	1.56
Tu	1.55
Yb	1.70
Lu	1.55
Hf	1.44
Ta	1.32
W	1.30
Re	1.25
Os	1.27
Ir	1.28
Pt	1.30
Au	1.43
Hg	1.45
Tl	1.50
Pb	1.50
Bi	1.49
Rn	2.14
Fr	2.46
Ra	2.35
Ac	1.79
Th	1.58
Pa (IV)	1.64
Pa (V)	1.52
U (IV)	1.62
U (V)	1.50
U (VI)	1.42
Np (IV)	1.60

Np (V)	1.49
NP (VI)	1.41
Pu (IV)	1.58
Pu (V)	1.48
Pu (VI)	1.40
Am (IV)	1.57
Am (V)	1.47
Am (VI)	1.39

Table S6 Ionization energy.

	Degree							
Atom	I	II	III	IV	V	VI	VII	VIII
H	13.595							
He	24.581	54.403						
Li	5.390	75.619						
Be	9.320	18.206	153.850					
B	8.296	25.149	37.920	259.298				
C	11.256	24.376	47.871	64.476	391.986			
N	14.53	29.593	47.426	77.450	97.863	551.925		
O	13.614	35.105	54.886	77.394	113.873	138.080	739.114	
F	17.418	34.98	62.646	84.14	114.214	157.117	185.139	953.60
Na	5.138	47.29						
Mg	7.644	15.031	80.12					
Al	5.984	18.823	28.44	119.96				
Si	8.149	16.34	33.46	45.13	166.73			
P	10.484	19.72	30.156	51.351	65.007	220.414		
S	10.357	23.4	35.0	47.29	72.5	88.029	280.99	
Cl	13.01	23.80	33.9	53.5	67.80	96.79	114.27	348.3
K	4.339	31.81						
Ca	6.111	11.868	51.21					
Sc	6.54	12.80	24.75	73.9				
Ti	6.82	13.57	27.47	43.24	99.8			
V	6.74	14.65	29.31	48	65	129		
Cr	6.764	16.49	30.95	50	73	91	161	
Mn	7.432	15.636	33.69		76		119	196
Fe	7.87	16.18	30.643					
Co	7.86	17.05	33.49					
Ni	7.633	18.15	35.16					
Cu	7.724	20.29	36.83					
Zn	9.391	17.96	39.70					
Ga	6.00	20.51	30.70	64.2				
Ge	7.88	15.93	34.21	45.7	93.4			
As	9.81	18.63	28.34	50.1	62.6	127.5		

Se	9.75	21.5	32	43	68	82	155	
Br	11.84	21.6	35.9	47.3	59.7	88.6	103	193
Rb	4.176	27.5						
Sr	5.692	11.027		57				
Nb	6.88	14.32	25.04	38.3	50	103		
Mo	7.10	16.15	27.13	46.4	61.2	68	126	
Ru	7.364	16.76	28.46					
Rh	7.46	18.07	31.05					
Pd	8.33	19.42	32.92					
Ag	7.574	21.48						
Cd	8.991	16.904	37.47					
In	5.785	18.86	28.03	54.4				
Sn	7.342	14.628	30.49	40.72	72.3			
Sb	8.639	16.5	25.3	44.1	56	108		
Te	9.01	18.6	31	38	60	72	137	
Cs	3.893	25.1						
Ba	5.210	10.001						
La	5.61	11.43	19.17					
Au	9.22	20.5						
Hg	10.43	18.751	34.2					
Tl	6.106	20.42	29.8	50.7				
Pb	7.415	15.028	31.93	42.31	68.8			
Bi	7.287	16.68	25.56	45.3	56.0	88.3		

Table S7 Metal radius (Coordination 12)

Metal atom	Radius
Li	1.58
Be	1.12
Na	1.92
Mg	1.60
Al	1.43
K	2.38
Ca	1.97
Sc	1.66
Ti	1.47
V	1.35
Cr	1.29
Mn	1.37
Fe	1.26
Co	1.25
Ni	1.25
Cu	1.28
Zn	1.37

Ga	1.53
Ge	1.39
Rb	2.53
Sr	2.15
Y	1.82
Zr	1.60
Nb	1.47
Mo	1.40
Tc	1.35
Ru	1.34
Rh	1.34
Pd	1.37
Ag	1.44
Cd	1.52
In	1.67
Sn	1.58
Sb	1.61
Cs	2.72
Ba	2.24
La	1.82
Hf	1.59
Ta	1.47
W	1.41
Re	1.37
Os	1.35
Ir	1.69
Pt	1.39
Au	1.44
Hg	1.55
Tl	1.71
Pb	1.75
Bi	1.82

Table S8 Valence electrons to covalent radius ratio.

Atom	Ratio
H	2.7
Li	0.76
Be	2.08
B	3.57
C	5.19
N	6.67
O	8.11
F	9.72

Na	0.63
Mg	1.45
Al	2.38
Si	3.42
P	4.55
S	5.77
Cl	7.00
K	0.50
Ca	1.15
Sc	2.08
Ti	3.04
V	4.13
Cr	5.04
Mn	5.88
Fe	2.50
Co	1.69
Ni	1.71
Cu	0.80
Zn	1.57
Ga	2.40
Ge	3.28
As	4.10
Se	5.13
Br	6.14
Rb	0.47
Sr	1.05
Y	1.86
Zr	2.76
Nb	3.79
Mo	4.55
Tc	5.83
Ru	3.17
Rh	3.15
Pd	3.08
Ag	0.70
Cd	1.40
In	2.05
Sn	2.86
Sb	3.55
Te	4.38
I	5.26
Ca	0.43
Ba	1.02

La	1.79
Hf	2.78
Ta	3.79
W	4.62
Re	5.60
Os	3.15
Ir	3.13
Pt	3.08
Au	0.70
Hg	1.38
Tl	2.00
Rb	2.67
Bi	3.36
Fr	0.41
Ra	0.85
Ac	1.68
Th	2.53
U	4.22
Ce	1.85
PrNd	1.85
Pm	1.85
Sm	1.80
Eu	1.83
Gd	1.88
Tb	1.89
Dy	1.90
Ho	1.91
Er	1.92
Tu	1.94
Yb	1.76
Lu	1.94

Table S9 Electronegativity

Atom	Pauling electronegativity	Белов-Бокий electronegativity
H		2.15
Li	1.0	0.95
Be	1.5	1.5
B	2.0	2.0
C	2.5	2.6
N	3.0	3.0
O	3.5	3.5
F	4.0	3.9
Na	0.9	0.9

Mg	1.2	1.2
Al	1.5	1.5
Si	1.8	1.9
P	2.1	2.1
S	2.5	2.6
Cl	3.0	3.1
K	0.8	0.8
Ca	1.0	1.0
Sc	1.3	1.3
Ti	1.5	1.1
V	1.6	1.4
Cr	1.6	1.4
Mn	1.5	1.4
Fe	1.8	1.7
Co	1.8	1.7
Ni	1.8	1.8
Cu	1.9	1.8
Zn	1.6	1.6
Ga	1.6	1.6
Ge	1.8	2.0
As	2.0	2.0
Se	2.4	2.4
Br	2.8	2.9
Rb	0.8	0.8
Sr	1.0	1.0
Y	1.2	1.2
Zr	1.4	1.5
Nb	1.6	1.7
Mo	1.8	1.6
Tc	1.9	1.9
Ru	2.2	2.0
Rh	2.2	2.1
Pd	2.2	2.1
Ag	1.9	1.9
Cd	1.7	1.7
In	1.7	1.7
Sn	1.8	1.7
Sb	1.9	1.8
Te	2.1	2.1
I	2.5	2.6
Cs	0.7	0.75
Ba	0.9	0.9
La		1.2

La-Lu	1.1-1.2	
Hf	1.3	1.4
Ta	1.5	1.3
W	1.7	1.6
Re	1.9	1.8
Os	2.2	2.1
Ir	2.2	2.1
Pt	2.2	2.2
Au	2.4	2.3
Hg	1.9	1.8
Tl	1.8	1.4
Pb	1.8	1.6
Bi	1.9	1.8
Po	2.0	2.0
At	2.2	2.2
Fr	0.7	0.7
Ra	0.9	0.9
Ac	1.1	
Th	1.3	1.0
Pa	1.5	
U	1.7	1.4
Np		1.4
Np-No	1.3	
Ce		1.2
Pr		1.2
Nd		1.3
Pm		1.3
Sm		1.3
Eu		1.2
Gd		1.3
Tb		1.3
Dy		1.3
Ho		1.3
Er		1.3
Tu		1.3
Yb		1.2
Lu		1.3
Pa		1.3
Pu		1.3
Am		1.3
Cm		1.3
Bk		1.3
Cf		1.3

Table S10 Equivalent conductance for molten chloride.

Chloride	Equivalent conductance ($\Omega^{-1}cm^{-2}$)
HCl	$1*10^{-6}$
LiCl	166
$BeCl_2$	0.086
BCl_3	0
CCl_4	0
NaCl	133.5
$MgCl_2$	28.8
$AlCl_3$	$1.5*10^{-6}$
$SiCl_4$	0
PCl_5	0
KCl	103.5
$CaCl_2$	51.9
$ScCl_3$	15
$TiCl_4$	0
VCl_4	0
RbCl	78.2
$SrCl_2$	55.7
YCl_3	9.5
$ZrCl_4$	
$NbCl_5$	$2*10^{-7}$
$MoCl_5$	$1.8*10^{-6}$
CsCl	66
$BaCl_2$	65.5
$LaCl_3$	29
$HfCl_4$	
$TaCl_5$	$3*10^{-7}$
WCl_5	$2*10^{-6}$
$ThCl_4$	16
UCl_4	0.34

References:

[1] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, ACM SIGMOD Record **29** (2), 93 (2000).

[2] F. T. Liu, K. M. Ting, and Z.-H. Zhou, ACM Trans. Knowl. Discov. Data **6** (1), Article 3 (2012).

[3] B. Schölkopf, J. C. Platt, J. S. Taylor, A. J. Smola, and R. C. Williamson, Neural Computation **13** (7), 1443 (2001).

[4] C. Chang and C. Lin, ACM Trans. Intell. Syst. Technol. **2** (3), Article 27 (2011).

[5] Y. Zhao and M. K. Hryniewicki, arXiv e-prints, arXiv:1912.00290 (2019).

[6] P. J. Rousseeuw, Journal of the American Statistical Association **79** (388), 871 (1984).

[7] P. J. Rousseeuw and K. V. Driessen, Technometrics **41** (3), 212 (1999).

[8] X. Li, J. C. Lv, and D. Cheng, presented at the Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems, Volume 1, Cham, 2015 (unpublished).

[9] T. Lu, fast machine learning (2021), https://pypi.org/project/fast-machine-learning/.

[10] O. Devinyak, D. Havrylyuk, and R. Lesyk, J. Mol. Graphics Modell. **54**, 194 (2014).

[11] J. H. Schuur, P. Selzer, and J. Gasteiger, Journal of Chemical Information and Computer Sciences **36** (2), 334 (1996).

[12] T. Lu, M. Li, Z. Yao, and W. Lu, Journal of Materiomics **7** (4), 790 (2021).

[13] V. Consonni, R. Todeschini, and M. Pavan, Journal of Chemical Information and Computer Sciences **42** (3), 682 (2002).

[14] V. Consonni, R. Todeschini, M. Pavan, and P. Gramatica, Journal of Chemical Information and Computer Sciences **42** (3), 693 (2002).

[15] L. Pauling, J. Am. Chem. Soc. **53** (4), 1367 (1931).

[16] R. T. Sanderson, J. Chem. Educ. **65** (2), 112 (1988).

[17] С. С. Белов and В. И. Дураков, Журнал структурной химии **1**, 353 (1960).

[18] G. Beskow, Geologiska Föreningen i Stockholm Förhandlingar **46** (6-7), 738 (1924).

[19] L. L. Quill, J. Chem. Educ. **27** (10), 583 (1950).

[20] W. H. Zachariasen, Zeitschrift für Kristallographie - Crystalline Materials **80** (1-6), 137 (1931).

[21] N. Chen, *Bond parameter function and application (In Chinese)*, 1st ed. (CHINA SCIENCE PUBLISHING & MEDIA LTD, Beijing, China, 1976).