Supporting Information

Feature Mining for Thermoelectric Materials based on Interpretable Machine Learning

Yiyu Liu, ‡[a]Zilong Mu, ‡[a]Peichao Hong, [a]Yun Yang, [a]Changxu Lin [ab]*

[a] State Key Laboratory of Physical Chemistry of Solid Surface, Xiamen, China

[b] Research Institute for Biomimetics and Soft Matter, Fujian Provincial Key Lab for Soft Functional Materials Research, Department of Physics, College of Physical Science and Technology, Xiamen University, Xiamen 361005, China

*E-mail: lincx@xmu.edu.cn

catalogue

1 Dataset

The feature information corresponding to each column in the raw data is shown in Table S1.

Table S1 The Feature Information Contained in the Original Data

Columns	Feature Description
composition	Chemical formula
crystallinity	Either single crystal, polycrystalline, or nanoparticles
synthesis	Brief string describing the synthesis method
spacegroup	Spacegroup number, if available
rho (ohm.cm)	Electrical resistivity, in ohm.cm
S [muV/K]	Seebeck coefficient, in microVolts/K, if available
PF [W/mK^2]	Thermoelectric power factor, $sigma * S^2$, in $[W/mK^2]$ if available
zT	Thermoelectric figure of merit, $PF * T/K$, if available
kappa [W/mK]	Thermal conductivity in $W/m * K$, if available
sigma [S/cm]	Electrical conductivity, in S/cm, if available
T [K]	Temperature in Kelvin at which these properties were obtained, if available
src	Original source of the recording.

To ensure the accuracy of preprocessing, statistical analysis was performed on the raw data, with specific results presented in Table S2 below:

Table S2 Statistical Analysis of Initial Data

	Space-group	rho (ohm.cm)	S [muV/K]	PF [W/mK^2]	zT	kappa [W/mK]	sigma [S/cm]	T [K]
Counts	1080.000	1093.000	1093.000	1093.000	714.000	714.000	1093.000	1082.000
Average value	148.763	22.800	-40.020	7.010E-04	0.208	4.190	1674.019	576.525
Standard deviation	77.491	450.286	193.103	1.061E-03	0.328	4.864	10533.184	264.424
Min	2.000	0.000	-752.200	1.766E-10	0.000	0.200	0.000	300.000
25%	62.000	0.002	-163.530	6.089E-05	0.017	1.752	50.000	300.000
50%	186.000	0.005	-67.600	2.154E-04	0.075	2.809	223.040	400.000
75%	220.000	0.023	100.000	8.497E-04	0.238	5.243	704.350	700.000
Max	227.000	14500.000	1235.430	6.728E-03	2.272	48.700	173720.000	1000.000

Table S3 Statistical Analysis After Data Preprocessing

	spacegroup	S [muV/K]	zT	kappa [W/mK]	sigma [S/cm]	T [K]
Counts	656.000	656.000	656.000	656.000	656.000	656.000
Average value	159.864	-41.560	0.221	4.059	587.232	562.957
Standard deviation	74.038	172.739	0.336	4.852	730.915	260.094
Min	2.000	-460.992	0.000	0.200	0.035	300.000
25%	64.000	-168.160	0.083	1.702	108.115	300.000
50%	204.000	-78.750	0.075	2.681	359.960	400.000
75%	221.000	104.250	0.251	4.970	799.235	700.000
Max	227.000	411.800	2.272	48.700	8196.700	1000.000

The HOMO_character, HOMO_element, LUMO_character and LUMO_element features generated by AtomicOrbitals characterization tool describe four groups of molecular orbital information, which are not included in the training matrix

Table S4 Descriptor Information Generated by Characterization

Name	Number of feature generation	Function Description
ElementProperty[1]	132	Elemental Property Descriptors
Meredig[2]	120	Thermodynamic Component Descriptors
BandCenter[3]	1	Calculation of Band Center Using Electronegativity
Stoichiometry[1]	6	Calculation Standards for Stoichiometric Feature Properties
AtomicPackingEfficiency[4]	5	Packing Efficiency Based on Amorphous Filler Geometry Theory
AtomicOrbitals[5]	7	Highest Occupied State/Lowest Unoccupied State
TMetalFraction[6]	1	Proportion of Magnetic Transition Metals in Components

2 Feature importance

Among the three models selected for importance analysis, we employed different methods. For the Ridge model, we prevent overfitting by adding an L2 regularization term to the loss function. We calculate feature importance using the weight coefficients of the linear model. In contrast, the GBDT model, as an ensemble learning method, builds a strong predictive model by combining multiple decision trees. For GBDT, we analyze feature importance using the tree model approach. The XGBoost model, on the other hand, utilizes SHAP values for feature importance analysis.

Following the described importance calculation methods, we conducted 100 rounds of cross-validation training. We collected normalized feature importance scores from each round, multiplied them by the R^2 scores, and summed them up as weighted cumulative feature importance. This result serves as the final ranking for each model.

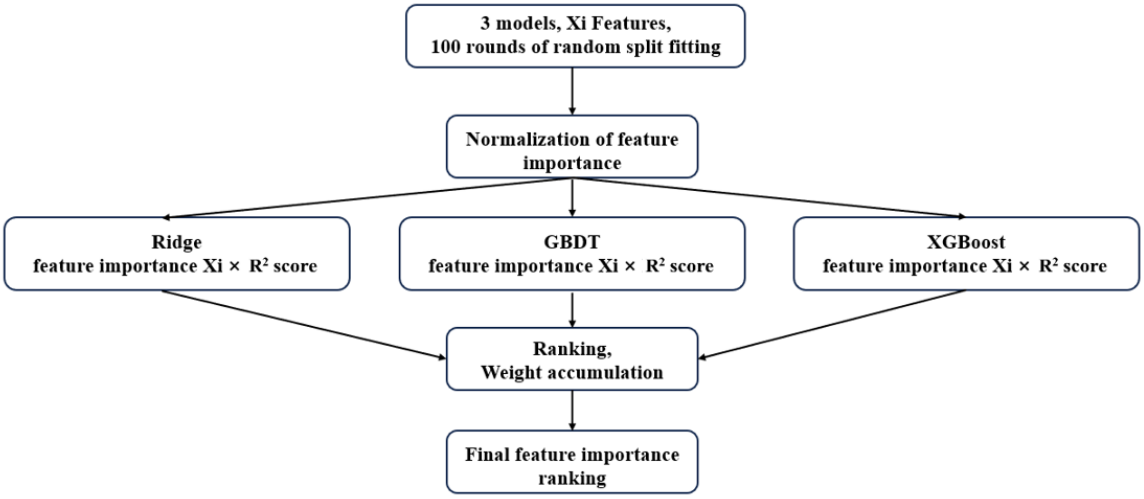

Figure S1 Feature importance-based feature selection.

We calculated the average performance evaluation indicators (R^2, RMSE, MAE) of each model in the 100 times CV process, and scored the overall performance of each model in the following way to allocate their own weight index for each model.

$$score_i = \frac{R2_{avg}}{RMSE_{avg} + MAE_{avg}} \#(1)$$

$$W_i = \frac{score_i}{\sum_{i=1}^{N} score_i} \#(2)$$

Table S5 The proportion of weights in the calculation results of each model during 100 rounds of training for lg(zT)

Rounds/100	Avg R^2	Avg RMSE	Avg MAE	Score	Weight
Ridge	0.739	0.969	0.620	0.465	17.26%
Gradient Boost	0.922	0.531	0.316	1.089	40.38%
XGBoost	0.928	0.510	0.302	1.142	42.36%

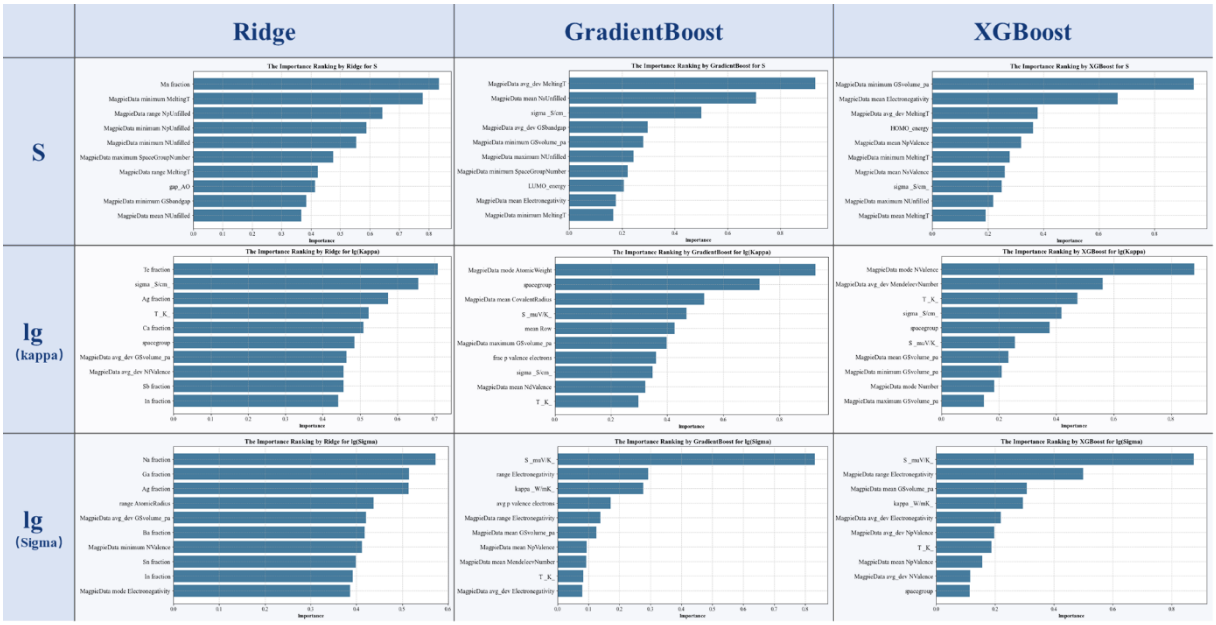

Figure S2 The distribution of the top ten important features under three importance calculation methods for a single feature.

3 Machine-Learning models and performances

Table S6 ML model parameters used by 11 regressors.

Regressor	Dependent package name	Parameters
Linear Regression	sklearn.linear_model. LinearRegression	Nan
Ridge	sklearn.linear_model. Ridge	alpha=55, solver='lsqr', max_iter=20000
Lasso	sklearn.linear_model.Lasso	alpha=0.003 , max_iter=20000 ,
ElasticNet	sklearn.linear_model.ElasticNet	alpha=0.015, l1_ratio=0.2,max_iter=20000
Decision Tree	sklearn.tree. DecisionTreeRegressor	max_depth=10 , max_features = 'auto'
Random Forest	sklearn.ensemble.RandomForestRegressor	max_depth=13 , max_features ='auto', n_estimators=50
AdaBoost	sklearn.ensemble.AdaBoostRegressor	learning_rate= 0.1, n_estimators= 100
Gradient Boost	sklearn.ensemble.GradientBoostingRegressor	learning_rate=0.15, max_depth=3, max_features='auto', n_estimators=1000
XGBoost	xgboost.sklearn.XGBRegressor	colsample_bytree=0.8, learning_rate=0.1, max_depth=3, n_estimators=500, subsample=0.8
K-nearest neighbor	sklearn.neighbors.KNeighborsRegressor	n_neighbors=3, weights='distance'
MLP	sklearn.neural_network.MLPRegressor	Activation='relu', alpha=0.1, hidden_layer_sizes=(100, 100), learning_rate='constant', learning_rate_init=0.01, max_iter=5000, solver='lbfgs'

* The parameters of the above models were obtained by GridSearchCV method. Other parameters not mentioned were used as default.

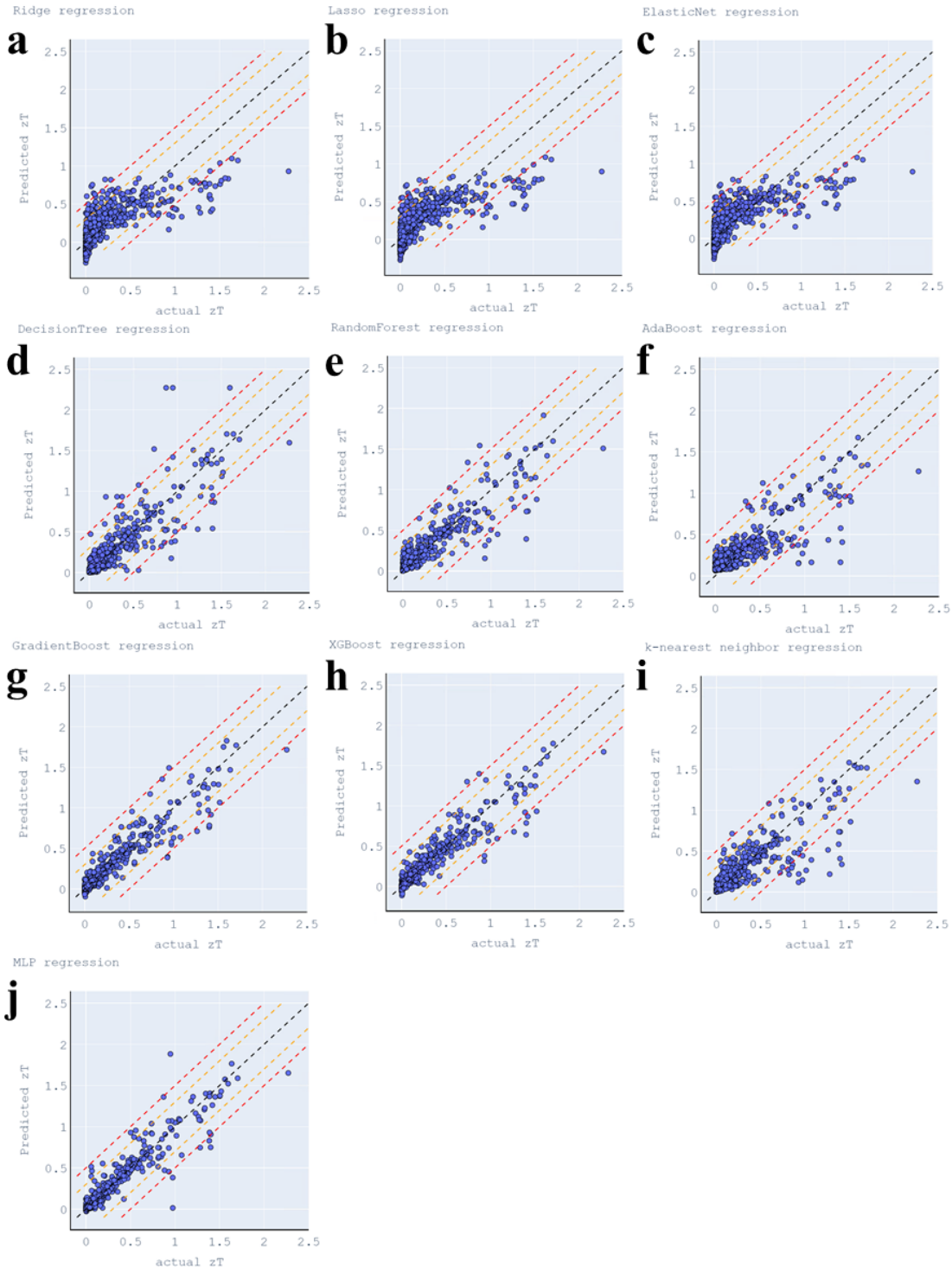

Figure S3 True-Predict Scatter Plot using the [656×274] feature matrix with 10 regressors.

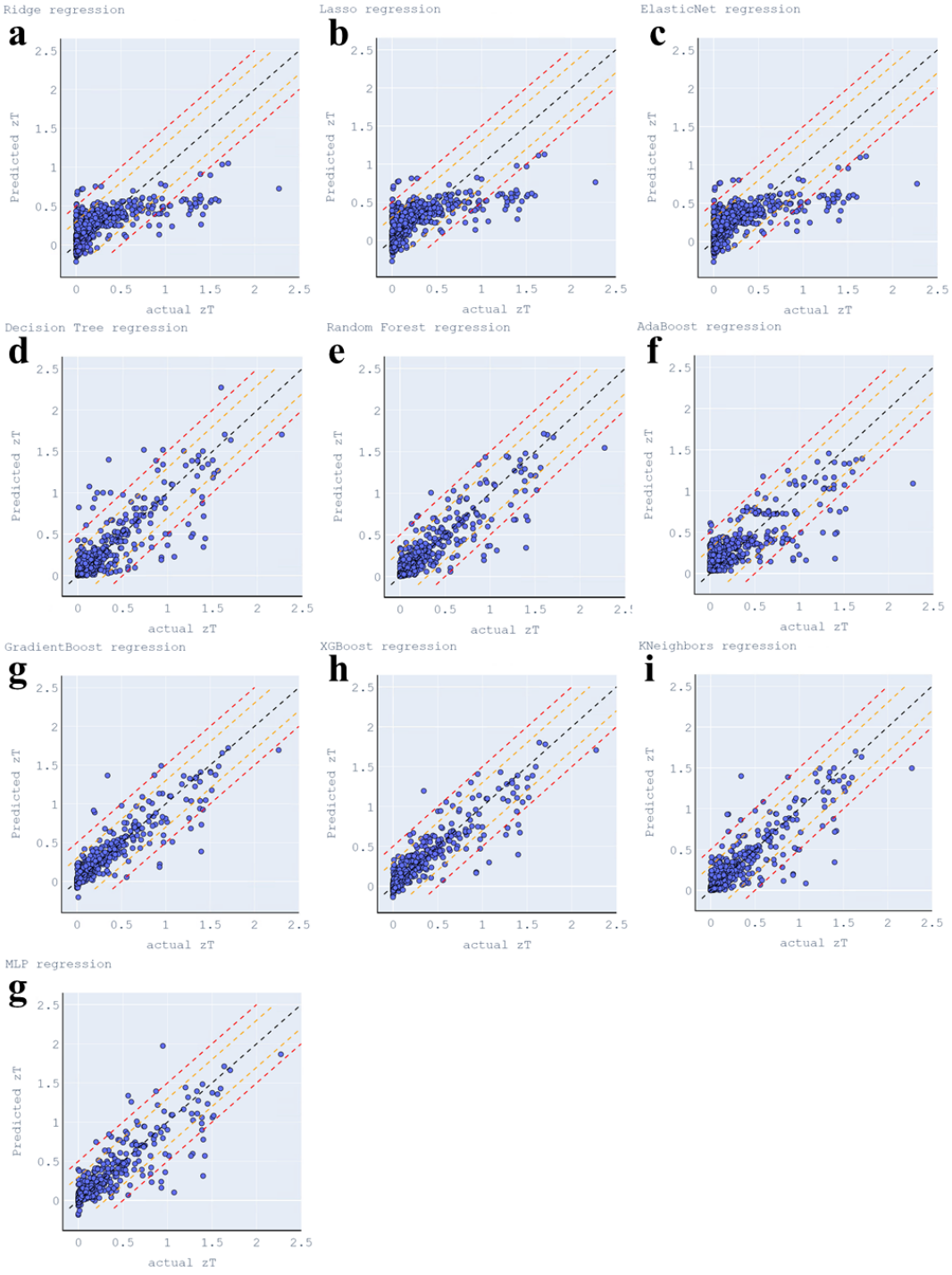

Figure S4 True-Predict Scatter Plot using the [656×5] feature matrix with 10 regressors.

The model performance differences in Table S7 are explained in detail. When the data set has many features, the multicollinearity problem between the features is easy to occur, and the standard Linear Regression method results in overfitting the model, with an R^2 of -77.462, which is poor. Ridge and Lasso solve this problem by adding L2 regularization terms and L1 regularization terms, respectively, so the performance is improved compared to the Linear Regression model with an R^2 of 0.523 and 0.519, respectively. Ridge regression, however, does not apply to feature selection; it considers all features. When features are highly correlated, Lasso regression selects some features at random. Moreover, it is difficult for them to choose regularization parameters, which can easily cause bad effects on the model. ElasticNet regression combines the advantages of Ridge and Lasso as well as their disadvantages, with an R^2 of 0.516.

For tree models, Decision Tree is easy to be disturbed by noisy data when data is divided, and it is sensitive to feature scaling, and the selection of feature scale will affect the structure of decision tree. Therefore, in the case of multiple features, the model performance will be affected when dealing with complex data sets, and the R^2 is 0.741. Random Tree makes the final prediction by integrating the prediction of multiple decision trees. The data and features of each tree are selected randomly, which greatly reduces the possibility of overfitting and improves the performance of the model with an R^2 of 0.823. While Gradient Boosting optimizes the model by gradually fitting residuals, each iteration of the model corrects the errors of the previous one, and can more accurately identify which features contribute most to the improvement of the model, performing best in the tree model with an R^2 of 0.891.

AdaBoost is particularly sensitive to noisy data and extreme values, and in each iteration, AdaBoost increases the weight of samples with large errors. If these samples are difficult to predict due to noise, the model will pay excessive attention to these samples,

resulting in overfitting the noisy data and decreasing model performance, with an R^2 of 0.692. XGBoost directly optimizes loss functions, adds regularization terms, and builds multiple complex decision trees capable of capturing nonlinear relationships in the data. When dealing with large, complex or noisy data sets, XGBoost performs better, with an R^2 of 0.887.

When making regression prediction, KNN takes K sample data from the training set that is closest to the sample to be tested, and the average value of these K sample data is considered to be the value of the sample to be tested. Because the data is sparse at the boundary, the effect of K-nearest neighbor model on the boundary data is poor, and the R^2 is 0.745.

MLP is often used in unstructured data and is more sensitive to the feature correlation of structured data. Redundant or irrelevant features may affect the performance of the model. Therefore, in the case of many features, the training process may be unstable and prone to overfitting, with an R^2 of 0.788.

Table S7 Training Results of Preselected Models

	R^2		RMSE		MAE	
	Training	**10-fold**	**Training**	**10-fold**	**Training**	**10-fold**
Linear Regression	0.705	-77.462	0.182	1.303	0.132	0.337
Ridge	0.641	0.523	0.201	0.225	0.139	0.158
Lasso	0.625	0.519	0.206	0.226	0.144	0.16
ElasticNet	0.621	0.516	0.207	0.227	0.144	0.159
Decision Tree	0.996	0.741	0.02	0.159	0.01	0.081
Random Forest	0.976	0.823	0.052	0.134	0.026	0.071
AdaBoost	0.814	0.692	0.145	0.179	0.11	0.121

Gradient Boost	1	0.891	0.002	0.106	0.001	0.057
XGBoost	0.998	0.887	0.017	0.108	0.012	0.058
K-nearest neighbor	1	0.745	0	0.164	0	0.085
MLP	0.998	0.788	0.015	0.109	0.006	0.047

4 Feature selection algorithm

Backward Elimination: In this study, we employed a backward elimination approach for preliminary feature selection. Backward elimination is a feature selection method and an essential step in feature engineering. The basic principle of backward elimination starts with a complete model containing all features. In each iteration, the least relevant feature is removed from the model, and the remaining features are used for training. At each step, the feature that results in the smallest performance drop upon removal is eliminated. This process continues until removing any feature significantly degrades model performance, leaving the most important features.

The main advantage of backward elimination is its consideration of interactions between features. However, when dealing with many features, the computational cost of backward elimination can become significant. Therefore, in this study, we improved the backward elimination process by removing the least important feature from the feature set based on the previously calculated feature importance ranking.

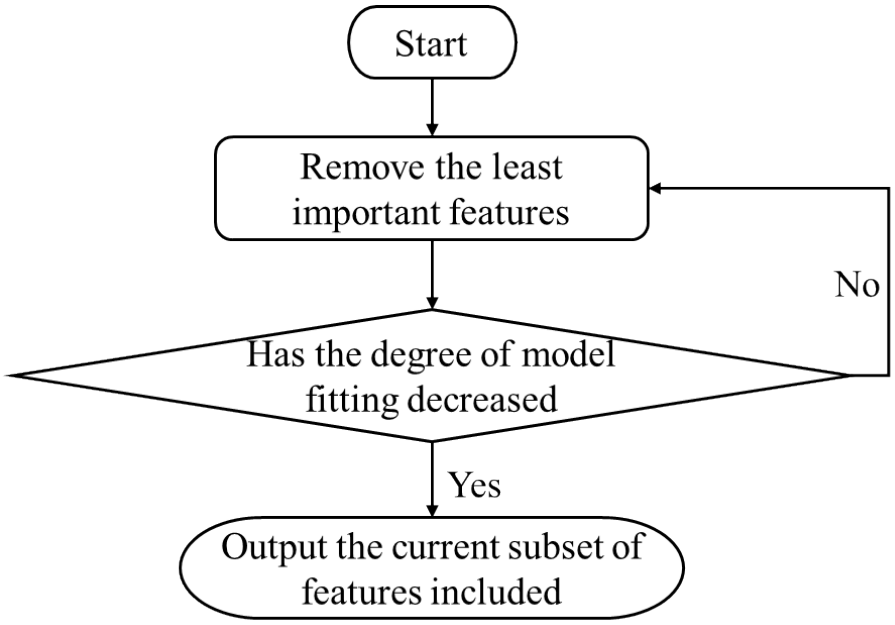

Figure S5 The flowchart of the backward deletion algorithm used in this study.

Forward Selection: The forward selection algorithm is employed in this study for further feature selection. The goal is to narrow down the range of physically important features that indirectly affect the zT value. The forward selection algorithm is a feature selection method that starts with a zero-feature model and gradually adds one feature at a time. After adding each feature, a model is constructed, and the most impactful feature in terms of performance improvement is selected, thus determining the optimal feature combination. This approach integrates feature selection and model training, allowing simultaneous selection of the best feature subset and training of the optimal model.

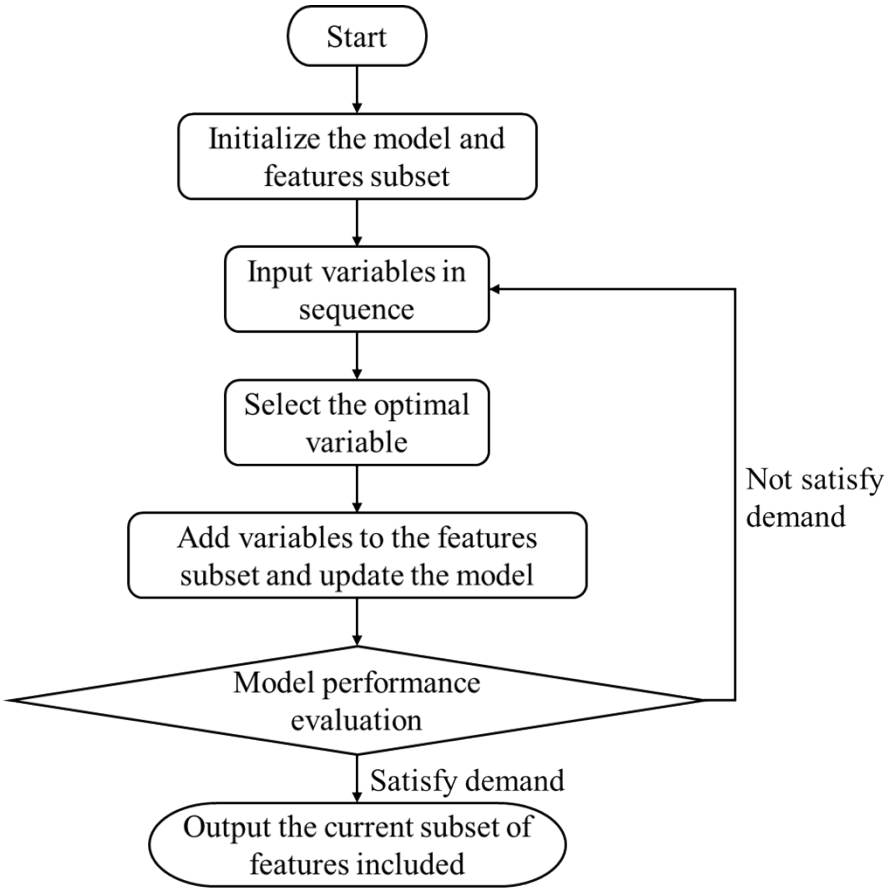

Figure S6 The flowchart of the Forward Selection algorithm used in this study.

Table S8 The Forward Selection Process of lg (zT)

Feature Names	RMSE	MAE	R^2
range Electronegativity	1.644	1.211	0.308
T _K_	1.192	0.744	0.637
MagpieData avg_dev MendeleevNumber	0.821	0.484	0.830
MagpieData mode NValence	0.764	0.462	0.853
MagpieData minimum GSvolume_pa	0.726	0.453	0.867
MagpieData mean GSvolume_pa	0.724	0.440	0.866
MagpieData mean NsUnfilled	0.718	0.441	0.868
Te fraction	0.705	0.435	0.873
MagpieData range Electronegativity	0.705	0.435	0.873
Mn fraction	0.711	0.441	0.871
MagpieData range NpUnfilled	0.712	0.442	0.870
spacegroup	0.706	0.434	0.874
mean Row	0.700	0.434	0.875
MagpieData minimum SpaceGroupNumber	0.701	0.435	0.875
MagpieData mean Electronegativity	0.693	0.437	0.878
MagpieData mode AtomicWeight	0.703	0.439	0.874
MagpieData mean CovalentRadius	0.706	0.440	0.874
HOMO_energy	0.729	0.443	0.865
MagpieData avg_dev NpValence	0.729	0.449	0.865
MagpieData maximum GSvolume_pa	0.736	0.445	0.862
MagpieData avg_dev GSbandgap	0.756	0.457	0.855
MagpieData minimum MeltingT	0.751	0.451	0.856
MagpieData avg_dev Electronegativity	0.758	0.457	0.853
MagpieData avg_dev MeltingT	0.774	0.455	0.847
MagpieData maximum NUnfilled	0.791	0.454	0.838
MagpieData mean NpValence	0.794	0.468	0.838

* The above calculation results are retained with three significant digits.

To further investigate the impact of primary features (Seebeck coefficient, electrical conductivity, and thermal conductivity) on the properties of thermoelectric materials, this study conducted a SHAP value analysis for each feature within an XGBoost model trained across the full feature space (encompassing all training features, Figure S7). The features were ranked according to their importance as perceived by the model, with the ranking approximated by the average magnitude of the SHAP values in descending order, placing higher importance on features closer to the top. The horizontal axis represents the magnitude of the SHAP values for specific data points, consistent with the main text analysis, where the sign of the SHAP value correlates positively or negatively with zT, and a larger magnitude indicates a stronger correlation. The scatter points correspond to each actual data point in the training data, with redder colors indicating higher zT values.

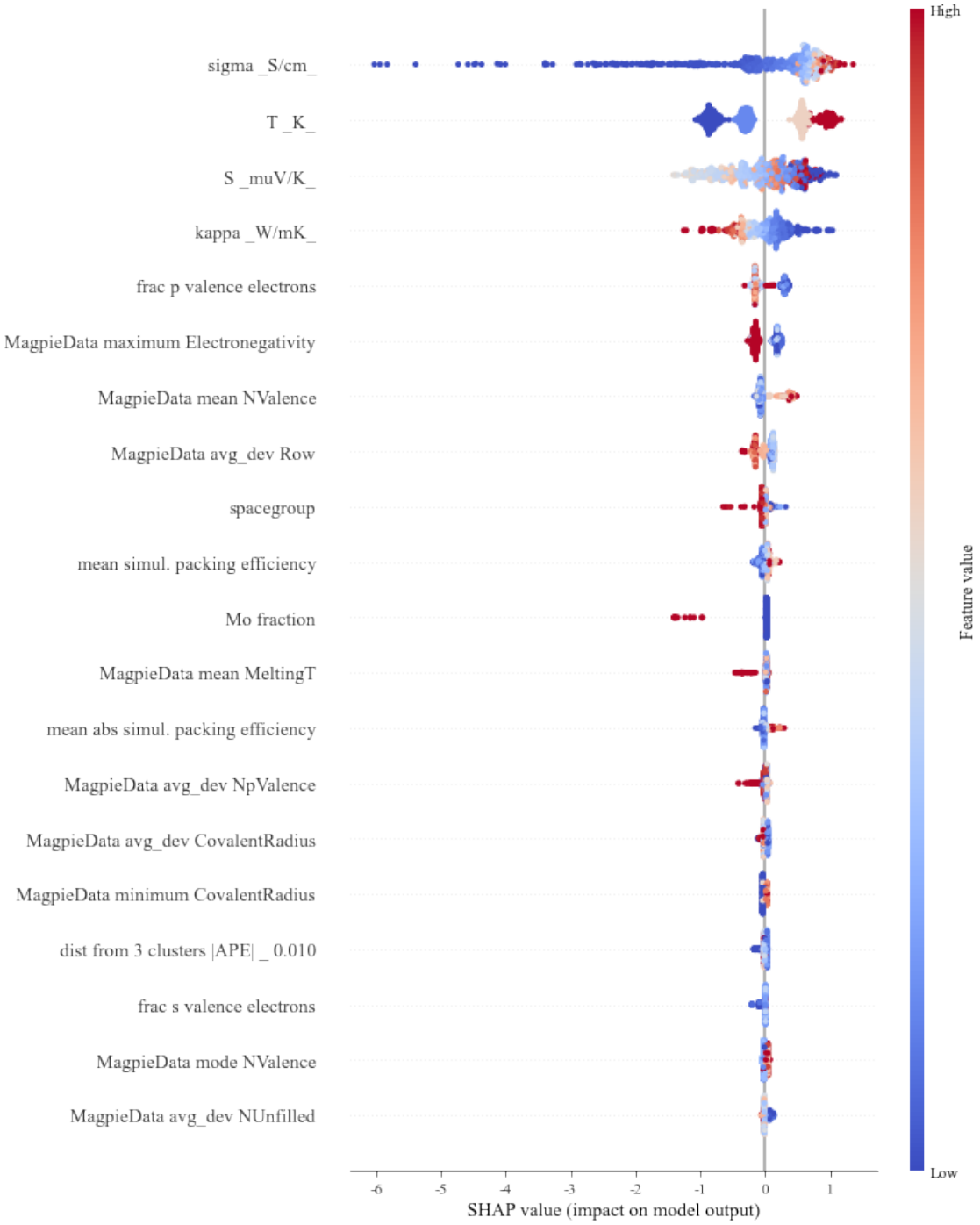

Figure S7 SHAP value analysis of each sample point and feature by the XGBoost model trained in the full feature space. The SHAP analysis in the full feature space validates the accuracy of our model, confirming that, as described in the definition of zT, the performance of thermoelectric materials is jointly determined by electrical

conductivity, temperature, Seebeck coefficient, and thermal conductivity. It also reveals that under the current data distribution, electrical conductivity has a more significant impact on material performance relative to temperature, Seebeck coefficient, and thermal conductivity.

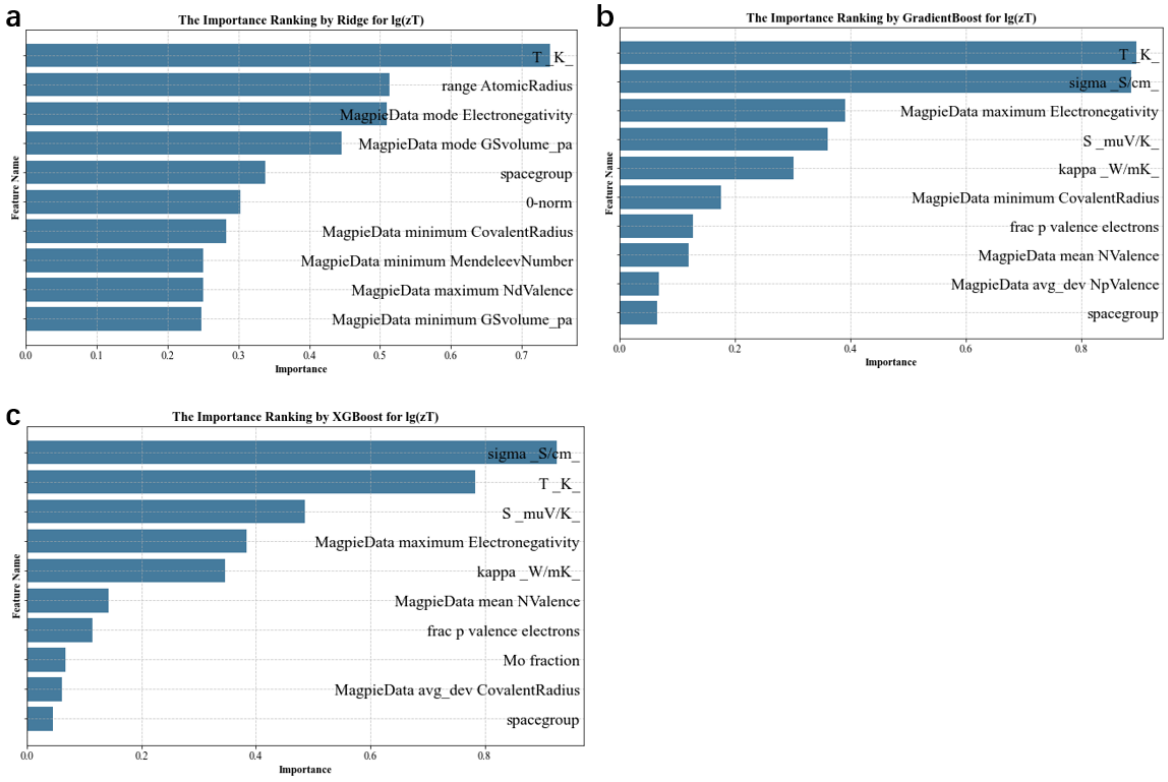

Figure S8 Feature importance ranking for lg(zT) by (a)Ridge, (b)GBDT, and (c)XGBoost models using SHAP method.

5 Reference

(1) Ward, L.; Agrawal, A.; Choudhary, A.; Wolverton, C. A general-purpose machine learning framework for predicting properties of inorganic materials. *Npj Computational Materials* **2016**, *2*. DOI: 10.1038/npjcompumats.2016.28.
(2) Meredig, B.; Agrawal, A.; Kirklin, S.; Saal, J. E.; Doak, J. W.; Thompson, A.; Zhang, K.; Choudhary, A.; Wolverton, C. Combinatorial screening for new materials in unconstrained composition space with machine learning. *Physical Review B* **2014**, *89* (9). DOI: 10.1103/PhysRevB.89.094104.

(3) Butler, M.; Ginley, D. Prediction of flatband potentials at semiconductor‐electrolyte interfaces from atomic electronegativities. *Journal of the Electrochemical Society* **1978**, *125* (2), 228.
(4) Laws, K. J.; Miracle, D. B.; Ferry, M. A predictive structural model for bulk metallic glasses. *Nature Communications* **2015**, *6*. DOI: 10.1038/ncomms9123.
(5) Kotochigova, S.; Levine, Z. H.; Shirley, E. L.; Stiles, M. D.; Clark, C. W. Local-density-functional calculations of the energy of atoms (vol 55, pg 191, 1997). *Physical Review A* **1997**, *56* (6), 5191-5192. DOI: 10.1103/PhysRevA.56.5191.2.
(6) Deml, A. M.; O'Hayre, R.; Wolverton, C.; Stevanovic, V. Predicting density functional theory total energies and enthalpies of formation of metal-nonmetal compounds by linear regression. *Physical Review B* **2016**, *93* (8). DOI: 10.1103/PhysRevB.93.085142.