Supporting Information for:

Cuprate superconducting materials above liquid nitrogen temperature from machine learning

Yuxue Wang,[a,b,c,1] Tianhao Su,[a,b,c,1] Yaning Cui, [a,b,c] Xianzhe Ma, [a,b,c] Xue Zhou,[d]

Yin Wang, [a,b,c] Shunbo Hu, [a,b,c,*] and Wei Ren [a,b,c,*]

[a] *Department of Physics, Material Genome Institute, Institute for the Conservation of Cultural Heritage, Shanghai University, Shanghai 200444, China*

[b] *Shanghai Key Laboratory of High Temperature Superconductors, International Center for Quantum and Molecular Structures, Shanghai University, Shanghai 200444, China*

[c] *Zhejiang Lab, Hangzhou 311100, China*

[d] *Center for Spintronics and Quantum Systems, State Key Laboratory for Mechanical Behavior of Materials, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China*

[1] *Both authors contribute equally to this work.*

[*] renwei@shu.edu.cn

[*] shunbohu@shu.edu.cn

All our ML works were completed using Scikit-learn (version 0.23) and Tensorflow (version 2.0.0).[1,2] Accuracy, area under the receiver operating characteristic curve (AUC), F1 score and confusion matrix are adopted in the evaluation criteria of the classification task. We chose 77K (temperature of liquid nitrogen) as the dividing point between high superconducting critical temperature (high-T_c) and low superconducting critical temperature (low-T_c), and found the imbalance of data and different penalties for prediction errors, which is a cost-sensitive ML classification task. We set the cost function as: $cost = W_1C_1 + W_2C_2$, where W_1 and W_2 are the weights for judging the punishment of serious errors and general errors. We aimed to reduce serious errors without compromising the accuracy,[3,4] C_1 and C_2 can be expressed as $C_1 = \frac{C_{01}}{C_{11}} + C_{10}$, $C_2 = \frac{C_{10}}{C_{11}} + C_{01}$ where C_{mn} represent the data in the confusion matrix.

Table S1. The basic influencing factors and the corresponding basic descriptors.

Prior Knowledge	Basic influencing factors	Basic descriptors
Impact of Cu valence and Jahn-Teller	number of d orbitals' valence electrons of element Cu	space group, number of various orbitals' valence electrons, covalent radius, column in the periodic table
RVB theory, Zhang-Rice model and *t-J* model	bond length and bond angle, electronegativity, radius and magnetic moment of Cu ions	covalent radius, space group, electronegativity, magnetic moment, volume of elemental, pseudopotential radius
SO(5) super symmetry theory	electron doping concentration	covalent radius, electronegativity, number of various orbitals' valence electrons
Impact of electronic and magnetic structure	number of various orbitals' valence electrons, ionic radius	magnetic moment, number of various orbitals' valence electrons
Polarons and Plasmon	electron concentration, crystal structure, lattice of layered structure	covalent radius, electronegativity, number of various orbitals' valence electrons, space group, pseudopotential radius

Table S2. The setting conditions of the first part of creating a series of virtual samples according to the distribution of the training samples. Here we take the cuprate as an example and use the Hg-Pb-Ca-Ba-Cu-O element combination for virtual high-throughput sample screening.

Element	Interval	Sampling	Step
Hg	[0.00,0.25]	Uniformity	0.01
Pb	[0.09,0.33]	Uniformity	0.01
Ba	[0.00,0.30]90% [0.30,1.00]10%	Local uniformity	0.002 0.018
Ca	[0.00,0.27]76% [0.27,1.00]23%	Local uniformity	0.048 0.015
Cu	[0.20,0.50]	Uniformity	0.01
O	[1.00,1.00]	Uniformity	1.00

According to the distribution of these elements in the dataset, we construct the distribution range of each element and whether the distribution is uniform. We take the content of O element as a reference point, i.e., setting the stoichiometric number of O element to be always '1', for constructing a virtual sample in the first part; in the second part, we need to analyze the relationship between the ratio of metal elements and oxygen elements in the dataset. From the distribution of the data set, the ratio of metal elements to O elements should be greater than 0.7 and less than 2.7, so candidate materials out of this range were deleted. The basic priority order of element filling in this step is: **Hg→Pb→Ba→Ca→Cu→O**. When the chemical formula is generated, the priority of the element might be interchanged in the first place, while the rest of the priority order remains unchanged. For example, in the virtual high-throughput prediction with **Pb** as the independent variable, the order of element filling was that: **Pb→Hg→Ba→Ca→Cu→O**. Due to the different sequences of elements, virtual samples with different preferences will be generated.

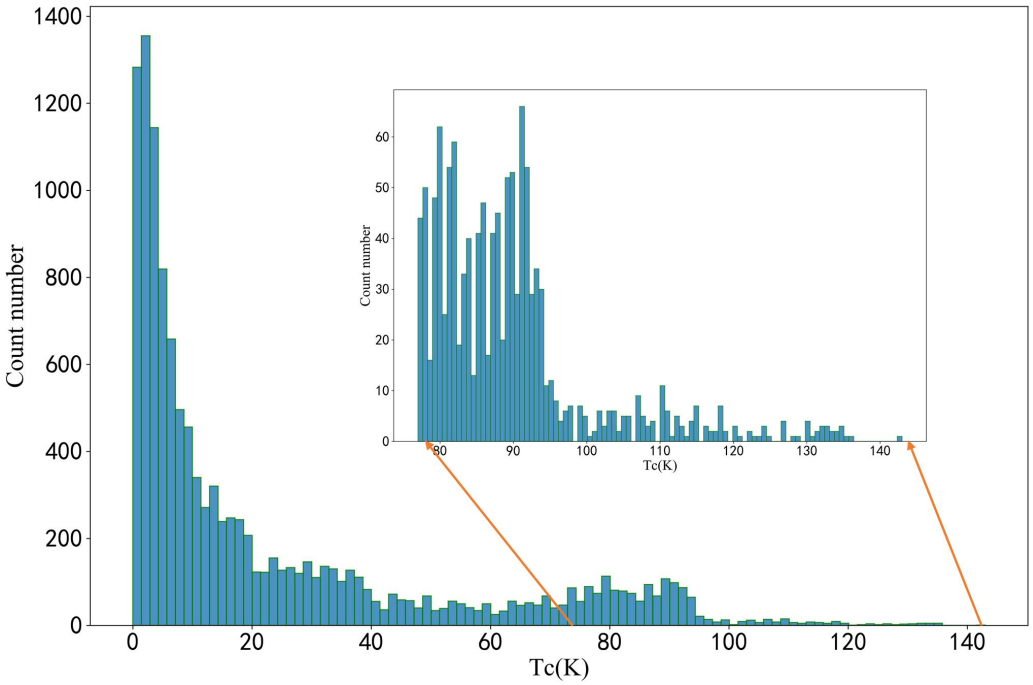
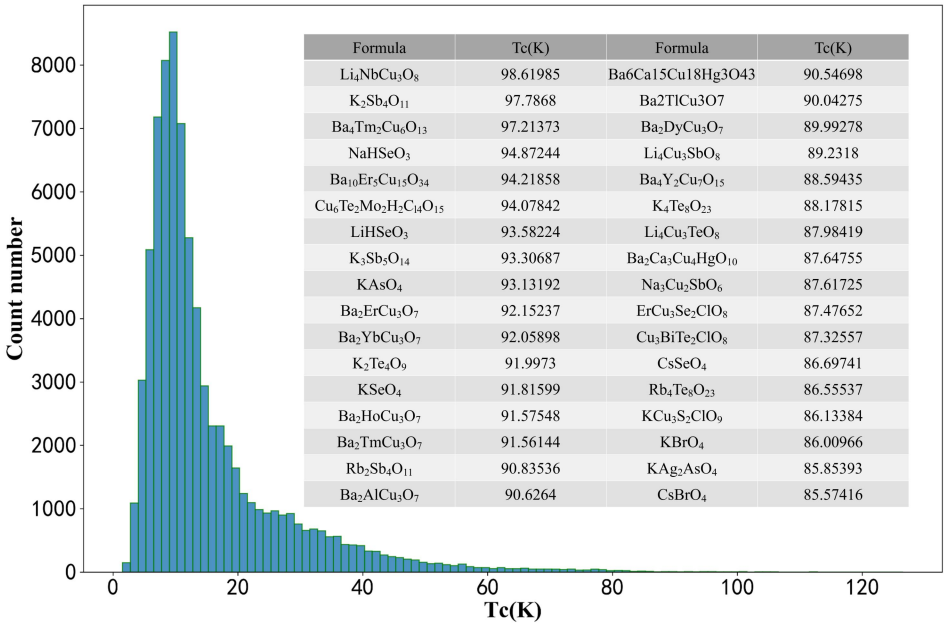


Fig. S1. The T_c distribution statistics from the Supercon database.[5]



Formula	Tc(K)	Formula	Tc(K)
$Li_4NbCu_3O_8$	98.61985	Ba6Ca15Cu18Hg3O43	90.54698
$K_2Sb_4O_{11}$	97.7868	Ba2TlCu3O7	90.04275
$Ba_4Tm_2Cu_6O_{13}$	97.21373	$Ba_2DyCu_3O_7$	89.99278
$NaHSeO_3$	94.87244	$Li_4Cu_3SbO_8$	89.2318
$Ba_{10}Er_5Cu_{15}O_{34}$	94.21858	$Ba_4Y_2Cu_7O_{15}$	88.59435
$Cu_6Te_2Mo_2H_2C_{14}O_{15}$	94.07842	$K_4Te_8O_{23}$	88.17815
$LiHSeO_3$	93.58224	$Li_4Cu_3TeO_8$	87.98419
$K_3Sb_5O_{14}$	93.30687	$Ba_2Ca_3Cu_4HgO_{10}$	87.64755
$KAsO_4$	93.13192	$Na_3Cu_2SbO_6$	87.61725
$Ba_2ErCu_3O_7$	92.15237	$ErCu_3Se_2ClO_8$	87.47652
$Ba_2YbCu_3O_7$	92.05898	$Cu_3BiTe_2ClO_8$	87.32557
$K_2Te_4O_9$	91.9973	$CsSeO_4$	86.69741
$KSeO_4$	91.81599	$Rb_4Te_8O_{23}$	86.55537
$Ba_2HoCu_3O_7$	91.57548	$KCu_3S_2ClO_9$	86.13384
$Ba_2TmCu_3O_7$	91.56144	$KBrO_4$	86.00966
$Rb_2Sb_4O_{11}$	90.83536	KAg_2AsO_4	85.85393
$Ba_2AlCu_3O_7$	90.6264	$CsBrO_4$	85.57416

Fig. S2. Predicted T_c distribution and some of the best candidates in the Materials Project Database. [6]

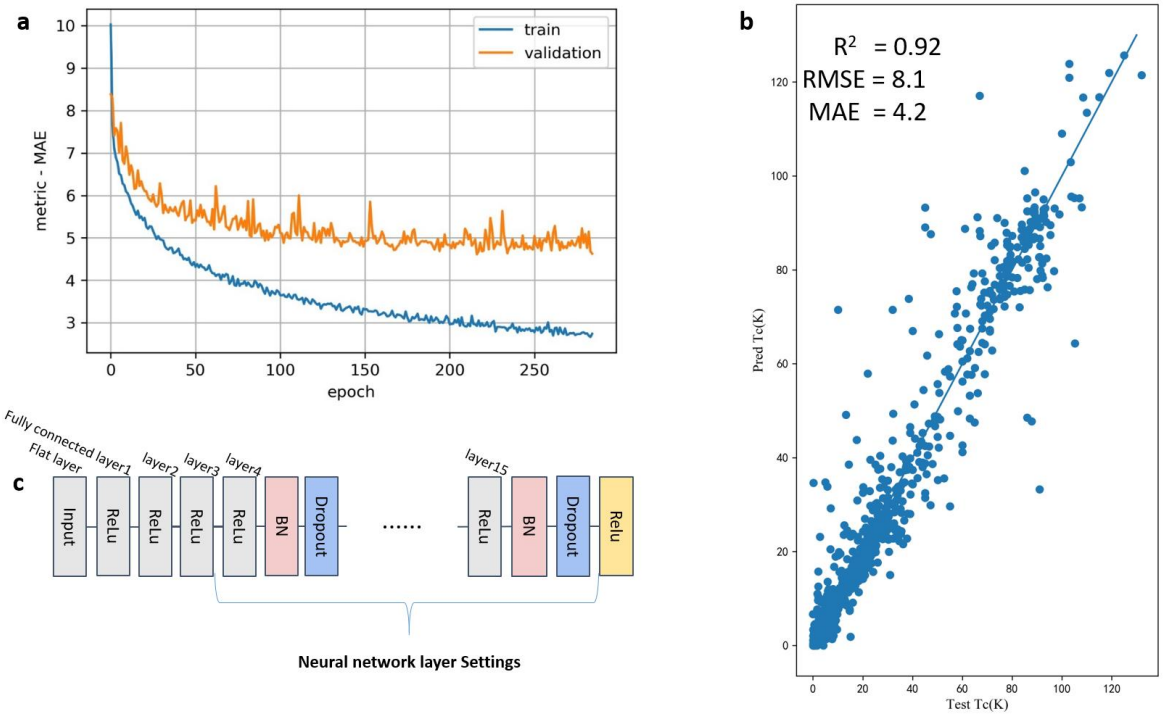


Fig. S3. Neural network training, testing and setup. **a** The convergence of a model training. **b** Performance of the DNN trained by AFS-2 on a random test, the score of R^2, RMSE and MAE on the test set are used as the performance indicator of the

model. **c** Deep neural network layer setting.

The first three layers have the activation function Rectified Linear Unit (ReLu) fully connected layer, and these three layers are frozen, the only thing that can be optimized is the number of neurons on each layer. Starting from the fourth layer, adding the Batch Normalization (BN) layer and Dropout layer, until the output layer of the last layer, the activation function of the output layer also selected ReLu and these unfrozen layers are optimized not only neuron parameters but the number of layers. Data are divided into three parts: training set, validation set and test set. The training set and validation set were used for model training, and the test set was used to evaluate model performance. In contrast, the validation set got converged if the performance fluctuation is less than 10^{-5}, and the training of the model will be stopped for more than 50 times of such convergence condition.

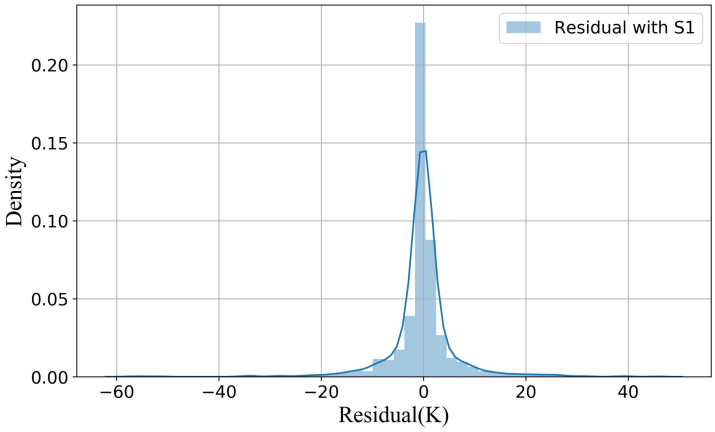


Fig. S4. Residual of DNN trained by AFS-1(S1), the absolute error is mostly within 5K and it is distributed within 20K.

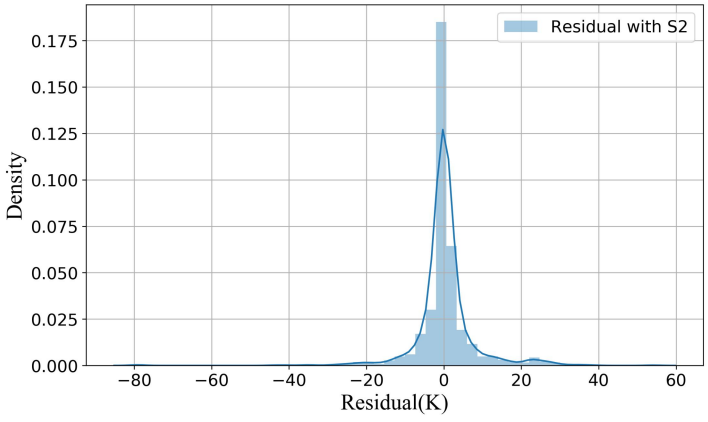


Fig. S5. Residual of DNN trained by AFS-2(S2), the absolute error is mostly within

10K, and it is distributed within 25K.

The trained models were saved as S1model, S2model folders, which can be loaded via Tensorflow2.0, to get complete residual information.

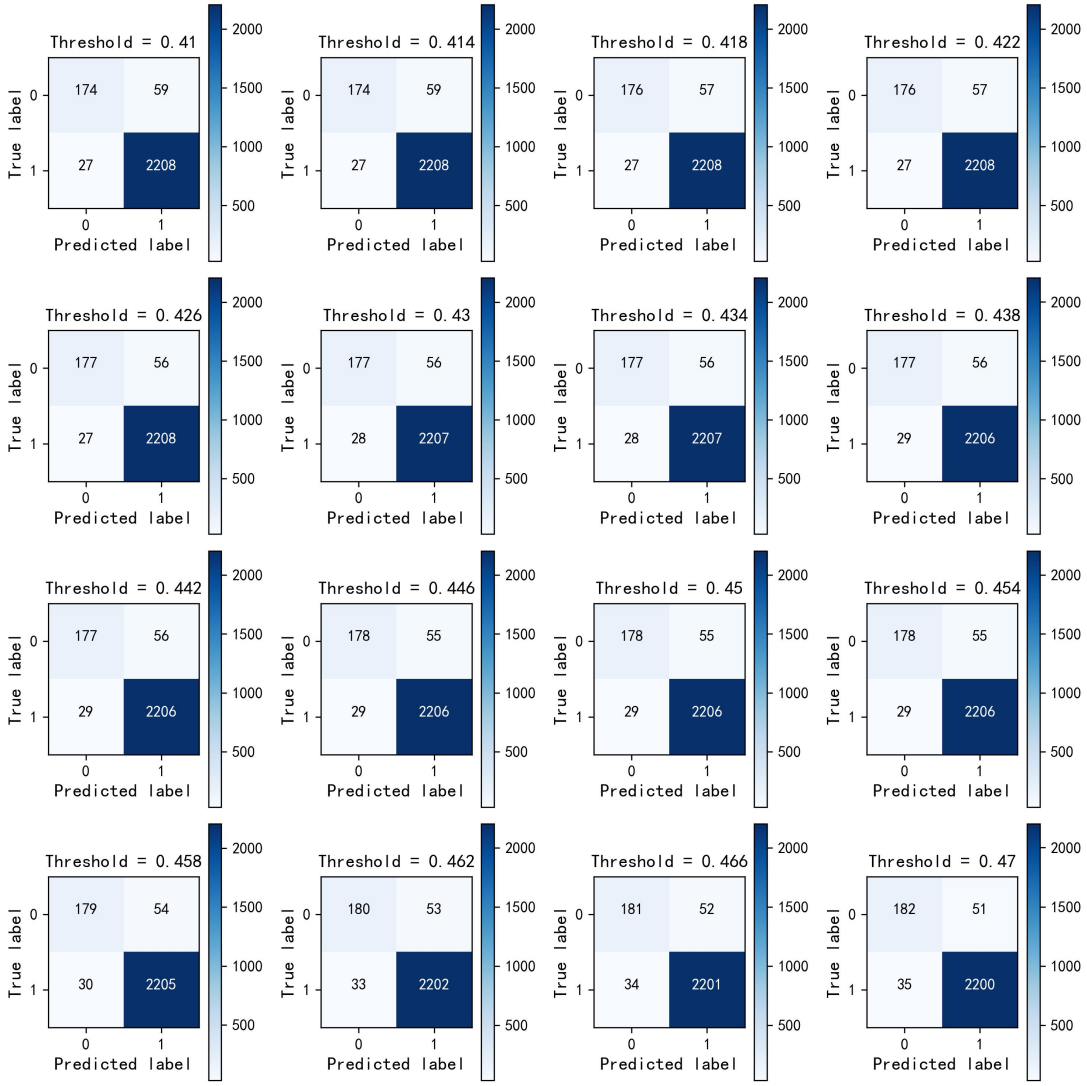


Fig. S6. In the interval of 0.44 ± 0.03, each threshold sliding step is set to 0.004. By observing the confusion matrix, we find that by adjusting the threshold, the frequency of serious errors can be reduced while maintaining other scores, thereby reducing the model's serious errors in the classification task.

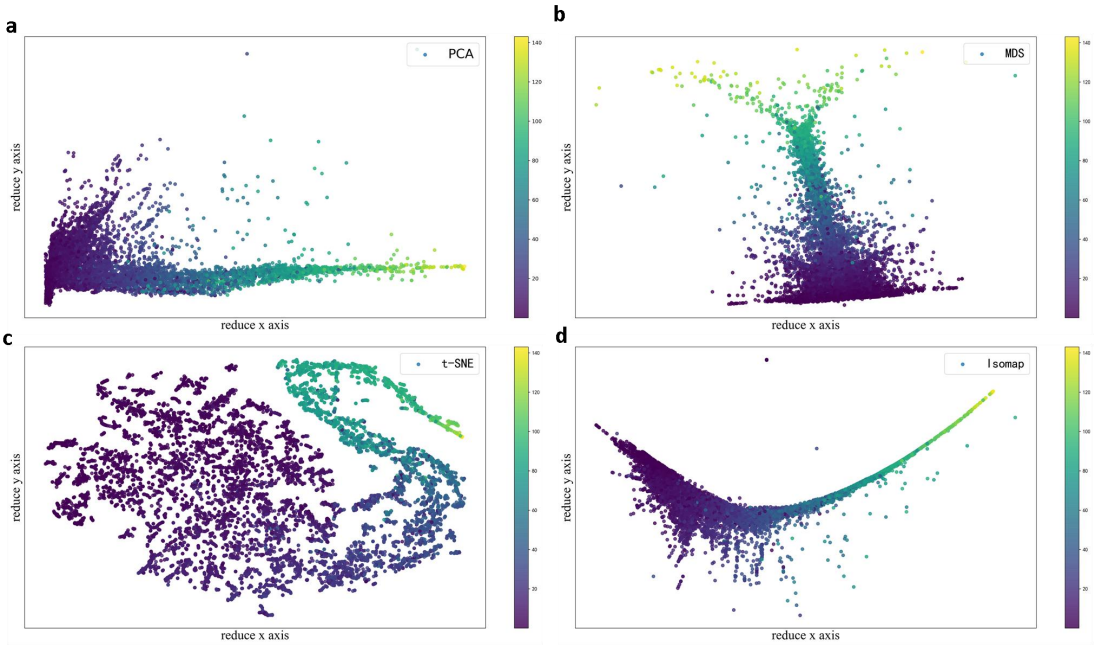


Fig. S7. According to the order of **a**→**b**→**c**→**d**, the manifold learning methods of principal component analysis (PAC), multidimensional scaling (MDS), t-distributed stochastic neighbor embedding (t-SNE) and isometric feature mapping (Isomap) show the dimensionality reduction visualization of the previous layer of the output layer of the deep neural network.[7–9]

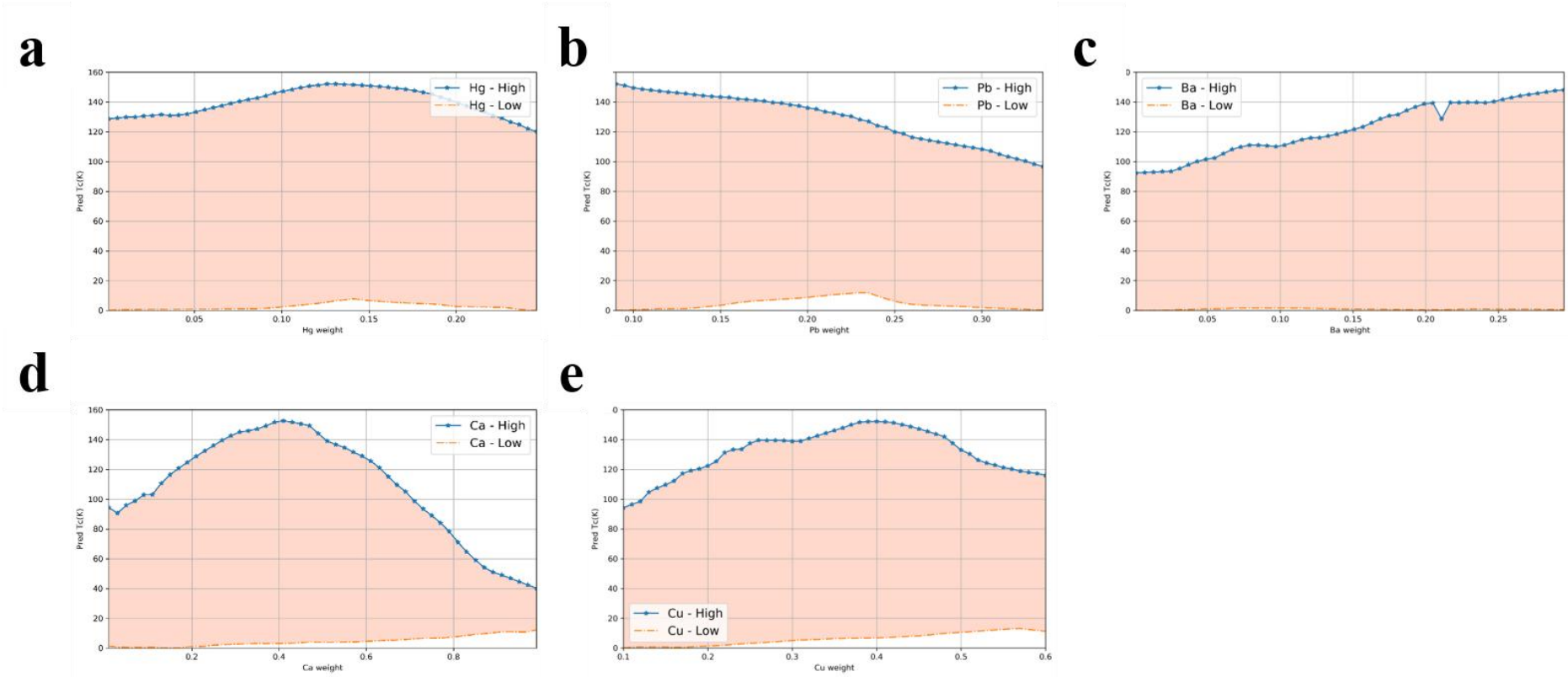


Fig. S8. The virtual sample prediction results with **a** Hg, **b** Pb, **c** Ca, **d** Ba, and **e** Cu as independent variables by DNN (trained by AFS-1), the blue and orange curves represent the highest and lowest predicted values, respectively.

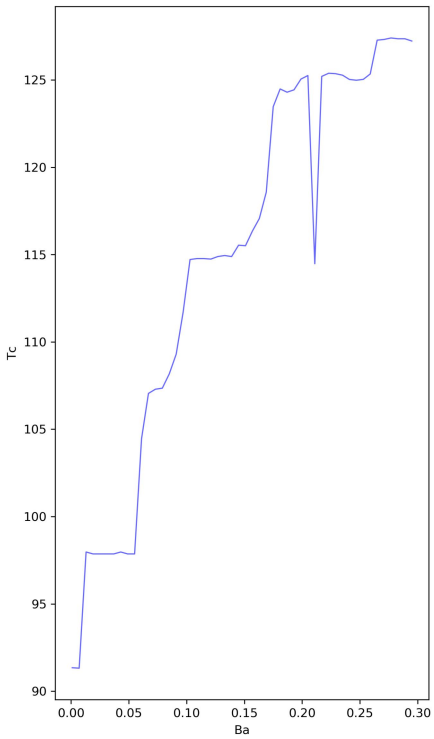


Fig. S9. The virtual sample prediction the highest T_c with Ba by RF model (trained by AFS-1), it also found a dip in the 0.2~0.25 interval of Ba weight, there may be a mysterious physical effect or possible wrong data in the data set.

In our restricted interval, the highest critical temperature increases as the content of the Pb element decreases and increases with the increase of Ba element; for Hg, Ca and Cu elements, there is a suitable proportion of components that corresponds to the highest T_c. Interestingly, there is a dip in the curve of the highest value of the Ba element. It is considered that there could be an outlier in the data, and an outlier is also found in the Isomap (Fig S7d) dimensionality reduction visualization in manifold learning. This outlier will not affect the performance of our model because of the complex inter-layer weights of the DNN, weakening the sensitivity to outliers. We can also use it to test whether there are suspicious data in other superconducting experimental data.

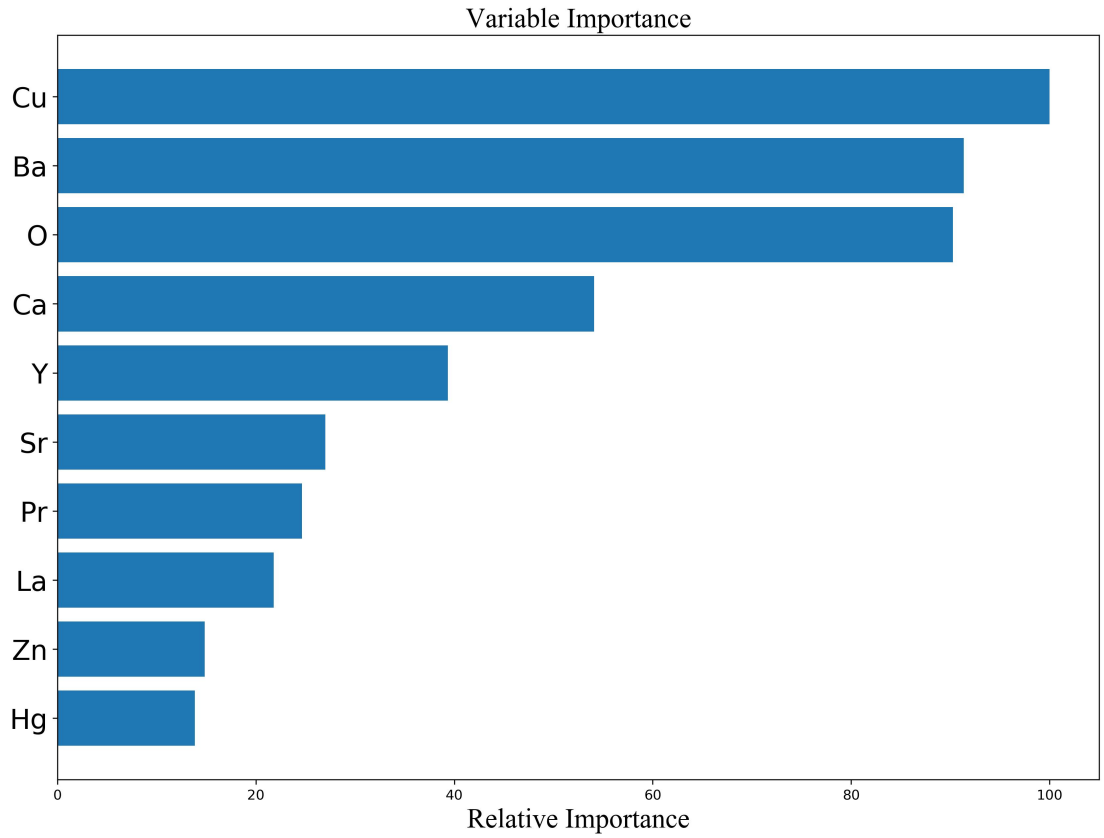


Fig. S10. The importance ranking of the AFS-1 descriptors extracted from the RF model. The richness of the elements is shown in the high-resolution picture "elem.jpg".

The source of the basic physical characteristics of the elements is collected from the Materials Agnostic Platform for Informatics and Exploration (MAGPIE),[10] see 'FillFeature.csv' for detail.

References

1 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos and D. Cournapeau, Scikit-learn: Machine Learning in Python, *MACHINE LEARNING IN PYTHON*, 6.

2 M. Ali, PyCaret: An open source, low-code machine learning library in Python, *PyCaret version*.

3 F. Pernkopf, Bayesian network classifiers versus selective k-NN classifier, *Pattern Recognition*, 2005, **38**, 1–10.

4 C. Elkan, The Foundations of Cost-Sensitive Learning, 6.

5 National Institute for Materials Science | Tsukuba, Japan | NIMS, https://www.researchgate.net/institution/National-Institute-for-Materials-Science/de

partments, (accessed March 1, 2022).
6 A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder and K. A. Persson, Commentary: The Materials Project: A materials genome approach to accelerating materials innovation, *APL Materials*, 2013, **1**, 011002.
7 L. van der Maaten and G. Hinton, Viualizing data using t-SNE, *Journal of Machine Learning Research*, 2008, **9**, 2579–2605.
8 M. Balasubramanian and E. L. Schwartz, The isomap algorithm and topological stability, *Science*, 2002, **295**, 7.
9 S. T. Roweis and L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, *Science*, 2000, **290**, 2323–2326.
10 L. Ward, A general-purpose machine learning framework for predicting, *npj Computational Materials*, 2016, 7.