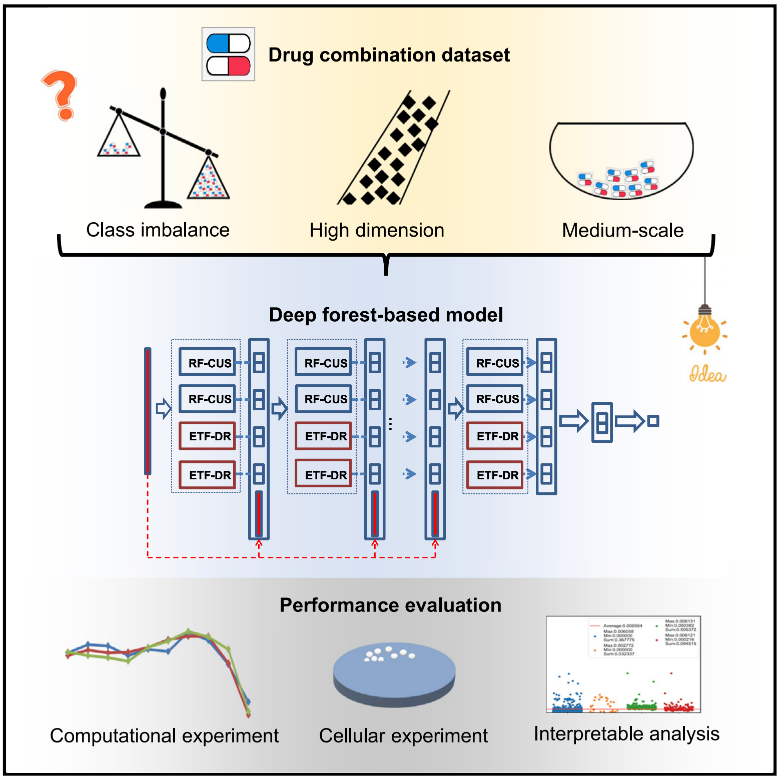
Cell Reports
Methods

Article

A hybrid deep forest-based method for predicting synergistic drug combinations

Graphical abstract



Highlights

- ForSyn is a deep forest-based method that predicts synergistic drug combinations
- ForSyn handles imbalanced, high-dimensional, and medium-scale datasets
- The predictive model is validated with cell-based assays
- Key genes can be extracted by the model for interpretable analysis

Authors

Lianlian Wu, Jie Gao, Yixin Zhang, ..., Kunhong Liu, Song He, Xiaochen Bo

Correspondence

lkhqz@xmu.edu.cn (K.L.),
hes1224@163.com (S.H.),
boxc@bmi.ac.cn (X.B.)

In brief

Wu et al. develop ForSyn, an improved, deep forest-based method that predicts synergistic drug combinations in cancer cell lines. ForSyn can effectively handle imbalanced and high-dimensional data in medium- and small-scale datasets. ForSyn predictions are validated with cell-based assays.

Wu et al., 2023, Cell Reports Methods *3*, 100411
February 27, 2023 © 2023 The Authors.
https://doi.org/10.1016/j.crmeth.2023.100411

Cell Reports Methods

CellPress
OPEN ACCESS

Article

A hybrid deep forest-based method for predicting synergistic drug combinations

Lianlian Wu,[1,2,5] Jie Gao,[4,5] Yixin Zhang,[2,5] Binsheng Sui,[3] Yuqi Wen,[2] Qingqiang Wu,[3] Kunhong Liu,[3,*] Song He,[2,*] and Xiaochen Bo[1,2,6,*]

[1]Academy of Medical Engineering and Translational Medicine, Tianjin University, Tianjin 300072, China
[2]Department of Bioinformatics, Institute of Health Service and Transfusion Medicine, Beijing 100850, China
[3]School of Film, Xiamen University, Xiamen 361005, China
[4]Department of Epidemiology and Health Statistics, School of Public Health, Fujian Medical University, Fuzhou 350122, China
[5]These authors contributed equally
[6]Lead contact
*Correspondence: lkhqz@xmu.edu.cn (K.L.), hes1224@163.com (S.H.), boxc@bmi.ac.cn (X.B.)

https://doi.org/10.1016/j.crmeth.2023.100411

MOTIVATION Combination therapy has shown promise as a treatment for complex diseases such as cancer. Synergistic drug combinations can offer increased therapeutic efficacy and reduce toxicity compared with single drugs. However, class imbalances in datasets have complicated the use of computational tools, such as deep learning, for synergistic drug prediction. We propose an improved deep forest-based model, ForSyn, to address the above problem on imbalanced, medium- or small-scale datasets with high dimensionality.

SUMMARY

Combination therapy is a promising approach in treating multiple complex diseases. However, the large search space of available drug combinations exacerbates challenge for experimental screening. To predict synergistic drug combinations in different cancer cell lines, we propose an improved deep forest-based method, ForSyn, and design two forest types embedded in ForSyn. ForSyn handles imbalanced and high-dimensional data in medium-/small-scale datasets, which are inherent characteristics of drug combination datasets. Compared with 12 state-of-the-art methods, ForSyn ranks first on four metrics for eight datasets with different feature combinations. We conduct a systematic analysis to identify the most appropriate configuration parameters. We validate the predictive value of ForSyn with cell-based experiments on several previously unexplored drug combinations. Finally, a systematic analysis of feature importance is performed on the top contributing features extracted by ForSyn. The resulting key genes may play key roles on corresponding cancers.

INTRODUCTION

There has been important progress in anticancer drugs, especially targeted therapies. However, many tumors inevitably become resistant to the single agents.[1–3] To overcome the limitations of monotherapy, combination therapy has been proposed as a new treatment approach. In combination therapy, multiple drugs can target multiple targets, subpopulations, or diseases simultaneously.[4,5] Compared with monotherapy, combination therapy can increase therapeutic efficacy, reduce toxic side effects, and slow down the development of drug resistance.[1,6–9] For these therapeutic benefits, combination therapy has become a standard clinical treatment strategy for several complex diseases including cancers.[7]

Systematic surveys of effective drug combinations *in vitro* have been proposed such as the high-throughput screening method.[10] However, it is insufficient for the large-scale experiments to search across such a large drug combination space.[11–13] To solve these problems, some computational approaches have been proposed such as network analysis[14–16] and mathematical models.[17] But most of them are often limited in the prior knowledge of biomedicine and the complexity of networks.[14] Alternatively, deep learning, as a data-driven computing method, has been widely used in drug combination prediction because of its generality, generalization and high prediction performance. Almost all deep learning methods used in drug combination prediction are based on deep neural networks (DNNs), including feedforward neural network,[18,19] deep belief network,[20] autoencoder,[21] transformer,[22] and graph neural network (GNN).[23] Although these methods have achieved high overall prediction performance, the problem of class imbalance is ignored. In drug combination dataset, the number of positive samples (minority class) involving synergistic drug combinations is usually small. Although most samples are negative samples (majority class) including antagonistic, additive and slightly synergistic drug combinations, which is usually more than ten times the number of positive samples. Most previous methods are based on the assumption that the distribution of training samples in each class is balanced. In the case of imbalanced data, the classification results are usually biased toward the majority class.[24,25] That is, the model tends to predict more samples as majority (negative) class to obtain higher overall prediction accuracy, while ignoring the prediction accuracy on minority (positive) class. Especially in DNN-based methods, it is prone to overfitting because of the samples in minority class are particularly rare. Anand et al.[26] explored the impact of imbalanced data on the neural network backpropagation algorithms. They showed that the majority class essentially dominates the gradient of the network and is responsible for the weight update of the model. The classification error of the majority class will rapidly decrease in the early iteration process, while the classification error of the minority class will increase and cause the network to fall into a slow convergence mode.

In addition, most previous studies only applied structural and physicochemical properties of drugs, and gene expression profiles of untreated cancer cell lines to construct the feature set. This may ignore the biological connection between drugs and cancer cells, as synergism is the response of cells to drugs.[5] The response of cancer cells to drugs should also be considered.[27–30] Once more informative feature types are applied, the samples with missing features should be removed. The number of samples will be reduced, and the dimension of each sample's feature will be increased. The DNN-based methods always rely on the large-scale training datasets, and it is difficult to maintain its prediction performance on a medium- or small-scale dataset. Small sample size dataset with high dimensionality has further aggravated the difficulty in drug combination prediction. This is also an inherent problem in many biomedical datasets with multi-view/multi-omics data.

Given the powerful performance of deep learning technology on classification tasks, it is of great importance to explore the application of non-neural network deep learning technology on imbalanced, medium- or small-scale datasets with high dimensionality. Zhou et al.[31] proposed the deep forest (DF) model, which can be regarded as an alternative to DNN. DF is a multilayer cascade structure, where each layer is composed of multiple tree-based forests. Each forest can be regarded as a unit in a cascade layer, similar to the neurons in the DNN. Compared with the DNN, the DF has the following advantages: suitable for datasets of different sizes, few hyper-parameters, and adaptive generation of model complexity.[32] The model complexity of DF can be adaptively determined under sufficient training. This advantage makes DF applicable to datasets of different scales, especially medium-sized datasets.[33] Because of its advantages, DF has been widely used in many fields, such as image retrieval,[34] cancer sub-category identification,[35] online financial cash-out monitoring,[36] etc. In the field of drug combination prediction, Zhang et al.[37] proposed a DF-based model, DCE-DForest, consisting of two components, a drug Bert[38] and a DF model. The Bert is a pretrained neural network to obtain the representations of drugs, and a DF is used to predict drug combinations. First, the drug representations extracted by Bert cannot fully represent the multi-view (physical, biological, etc.) information of drugs. Each dimension of the representations has no specific meaning and cannot be interpreted. Second, DCE-DForest uses the original DF framework and does not consider the case of data with imbalance and high feature dimension.

To solve the above problems, we first construct a feature set consisting of physical, chemical and biological properties of drugs, in which the key features can be evaluated through ForSyn. The feature types include drug molecular fingerprints (DMFs), drug physicochemical properties (DPPs), cell line-specific drug-induced gene expression profiles (DGEs), and gene expression profiles of untreated cell lines (CGEs). The cell line-specific DGE feature can not only capture biological connection between drugs and cancer cells, but also be generalized to the study of patients.[39] Each dimension of the curated feature types has a specific meaning, which can facilitate the interpretable analysis to find out the key features in prediction process. Faced with this imbalanced, high-dimensional and medium-sized dataset, an improved DF-based model, ForSyn, is proposed to predict synergistic drug combinations. Two novel forest units are designed to embed in ForSyn. One is an RF based on affinity propagation (AP) clustering[40] and stratified under-sampling, which is designed to deal with the problem of class imbalance. The other is an extreme tree forest (ETF) that based on data complexity dimension reduction dealing with the problem of high-dimensional data. Then, the application of ForSyn is systematically analyzed by comparing 12 algorithms in eight datasets. The ForSyn with all the feature types wins the best performance in most cases. The performance of different configurations of ForSyn are also explored. Then, cellular experimental validation performed on a set of previously untested drug combinations further confirms the predictive ability of ForSyn. Finally, a systematic interpretable analysis of the key features extracted by ForSyn is performed.

RESULTS

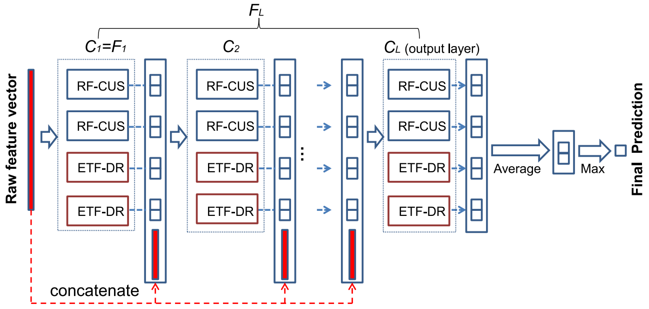
The framework of ForSyn

In this study, the drug combinations tested in different cancer cell lines are collected as the sample dataset. The effects of drug combinations can be classified as synergism and non-synergism. A total of 3,192 samples are obtained from the DrugComb,[41] DrugCombDB,[42] and AstraZeneca-Sanger Drug Combination Prediction[43] databases, and classified according to the scheme proposed by Malyutina et al.[44] Two hundred samples are regarded as the synergism class (minority class), and the remaining 2,992 samples are classified as non-synergism class (majority class). The imbalance rate is close to 15, which is defined as the ratio between the size of the majority class and that of the minority class. Meanwhile, feature set is composed of four feature types. The 881-dimensional DMF, 55-dimensional DPP, and 978-dimensional DGE are used as the feature of drugs, the 978-dimensional CGE are used to represent cancer cell lines. All the feature types have been proved to be effective on other drug-related prediction tasks.[33,45–49] Each dimension of the curated feature types has a specific meaning,

Table 1. Eight datasets used in this study

Dataset	Description	Dimension
Data 1	DMF + CGE	2,740
Data 2	DPP + CGE	1,088
Data 3	DGE	1,956
Data 4	DMF + DPP + CGE	2,850
Data 5	DMF + DGE	3,718
Data 6	DPP + DGE	2,066
Data 7	DMF + DPP + DGE	3,828
Data 8	DMF + DPP + DGE + CGE	4,806

In the eight datasets, the representation of each sample is the concatenation of pairwise drug feature and the cell line feature.



RF-CUS : Random Forest based on Clustering and Stratified Under-sampling.
ETF-DR : ETF based on data complexity Dimension Reduction.
C_l : stands for one layer in the cascade, where l=1...L, L is the number of layers.
F_l : stands for the cascade which incorporate multiple layers.

Figure 1. The overall framework of ForSyn

which can facilitate the interpretable analysis to find out the key features in prediction process. More specifically, each dimension in DMF, DPP, and DGE represents a substructure, physicochemical property, and gene expression values of drugs respectively.

To further investigate the influence of different representations in the classification process, eight different datasets including different feature combinations are generated (Table 1). In the training dataset of this study, a sample represent a drug combination on a particular cancer cell line (i.e., a drug combination-cell line pair). The same drug combination on different cell lines will have different effects. It is important to distinguish the drug combinations on different cell lines. In order to make the model to gain the distinguished ability, the representation of each sample is consisting of the drug feature and a cell line-specific feature. The drug feature includes DGE, DMF, and DPP. The cell line-specific features are DGE and CGE. According to the principle, eight datasets are generated and listed in Table 1.

Faced with the imbalanced, high-dimensional and medium-sized datasets, we propose ForSyn, which is a multi-layer cascade structure (Figure 1). Two novel forest types are embedded as the unit in each cascade layer. One is the RF based on clustering and stratified under-sampling (RF-CSU) dealing with imbalanced data. The other is an ETF based on data complexity dimension reduction (ETF-DR) dealing with high feature dimension (details are provided in STAR Methods).

RF is one of the representative algorithms of ensemble learning. It performs bootstrap sampling and random feature selection in the induction process of the base classifier. The perturbation of the feature space and the sample space ensures the diversity of the ensemble system. However, as with most traditional machine learning algorithms, the RF cannot effectively process imbalanced data. To deal with the problem of imbalanced data, the most common method is to rebalance the training set, such as randomly under-sampling the majority class. But this method always loses useful information. Some training samples that may play a key role in the classification process may be lost in the under-sampling process. To overcome this defect, we design an under-sampling method on the basis of AP clustering and stratified under-sampling, to rebalance the training set and minimize the information loss caused by random sampling. The proposed under-sampling method is combined with the standard RF framework to rebalance the training set of each decision tree.

The ETF can be regarded as a variant of RF. Different from the RF, the ETF uses all the features as candidates, and then randomly selects a feature as the split node of the tree. The tree will continuously grow until each leaf node contains samples of the same class.[32] According to the properties, the ETF performs better to the imbalanced data. The pure leaf node that stores minority samples can effectively identify unknown minority samples. However, the high feature dimension and random selection of features would deepen the depth of the tree and cause over-fitting. To overcome the problem of ETF, we propose a greedy dimension reduction method, which combines a data complexity metric with the greedy algorithm. Data complexity, such as the shape of the decision boundary and the overlap between classes, is always used to describe the characteristics of the data.[50] The data complexity metrics would closely affect the predictive performance of the classifier.[51] In this study, the data complexity metric is defined as the tail overlap of the conditional distribution between two classes[50] (details are provided in STAR Methods).

Performance evaluation

In this experiment, ForSyn is compared with 12 advanced algorithms on five metrics. The comparison algorithms include eight state-of-the-art deep learning-based algorithms in drug combination prediction, and four advanced machine learning algorithms. The deep learning-based algorithms are four DNN-based methods (DeepSynergy,[18] MatchMaker,[19] TranSynergy,[22] and SynPathy[52]), two DF-based methods (original DF[32] and DCE-DForest[37]), and two GNN-based methods (DeepDDS-GCN and DeepDDS-GAT[23]). The machine learning algorithms are two ensemble learning methods (XGBoost[53] and RF), and two imbalance learning methods (RUSBoost[54] and balanced bagging[55]). The evaluation metrics include F1 score, AUPR (area under the precision-recall curve), recall, MCC (Matthews correlation coefficient), and G-mean[24]; the F1 value is regarded as the main evaluation metric.

The results of all algorithms on the five metrics are shown in Tables S1–S5. The performance results are the mean value of ten-time 5-fold cross-validation (CV). In addition to the

DeepDDS, other 11 algorithms are tested on eight datasets (data 1–8) composed of different feature types. The performance of DeepDDS-GCN and DeepDDS-GAT based on graph data is shown in separate rows in Tables S1–S5. In Tables S1–S5, the performance of 11 algorithms based on data 1–8 is ranked, and the ranking values are shown in parentheses. The smaller ranking value indicates better performance. Then, the Friedman test and the Nemenyi test[56] are used to analyze the performance difference among the 11 algorithms. The Friedman test compares the performance differences of multiple algorithms on multiple datasets, while the Nemenyi test is performed between pairwise algorithms. According to Equations 15 and 16 (provided in STAR Methods), the Friedman statistical values and the corresponding p values in Tables S1–S5 are 16.40 ($p = 2.638 \times 10^{-8}$), 55.35 ($p = 5.200 \times 10^{-11}$), 37.30 ($p = 1.480 \times 10^{-10}$), 28.10 ($p = 1.823 \times 10^{-9}$), and 33.34 ($p = 3.419 \times 10^{-10}$), respectively ($N = 8$, $K = 11$). The distribution of F_F is based on the F distribution with 10 and 70 degrees of freedom. The critical value of F_F is 1.969 (Equation 16) with a 95% confidence level. The statistical results and p values on all the metrics reflect that there is a significant performance difference among the 11 algorithms. Next, according to Equation 17 and Table S6, CD = 5.338 is calculated with the 95% confidence level in this study. Figures 2A–2E visually show the Nemenyi test results for Tables S1–S5. The average rank of the algorithms in data 1−8 is shown as the red dot in Figures 2A–2E.

From the average rank of performance results on data 1–8 (Figures 2A–2E; Tables S1–S5), it is observed that the ForSyn ranks first on four metrics, F1 score, AUPR, MCC and G-mean, showing its superior prediction performance. In addition, ForSyn performs better than the two DeepDDS algorithms on almost all datasets (Tables S1–S5). The deep learning-based algorithms, original DF, DCE-DForest, DeepSynergy, MatchMaker, TranSynergy, SynPathy and DeepDDS, have no module for imbalanced data processing, so the performance results on the five typical evaluation metrics of imbalanced data are relatively low. For the metric of recall, the performance of ForSyn ranks second, slightly lower than that of balanced bagging (Figure 2C; Table S3). Actually, the recall metric cannot fully reflect the performance of the model, and it often conflicts with precision. According to Figure 2A and Table S1, the F1 score of balanced bagging is low. It can be inferred that the algorithm greatly sacrifices the recognition rate of the majority class samples in exchange for an improvement in the recognition rate of the minority class samples. In addition to ForSyn, the other two DF-based algorithms, original DF and DCE-DForest, have similar ranks in all metrics and get the middle rank. This shows that the innovative design of ForSyn has brought great performance improvement. Figure 2F show the performance difference between ForSyn and other algorithms on the main metric (F1 score) more intuitively. From Figure 2F, it is observed that only three algorithms, XGBoost, random forest, and balanced bagging, have slightly better performance than ForSyn on data 1, 2 and 4. In addition to the three algorithms, ForSyn outperforms other comparison algorithms on all datasets. Similar results exist in other metrics. The performance difference on other metrics is shown in Figure S1.

Next, to evaluate the generalization performance on novel unseen cell lines, drugs and drug combinations, three cross-validation strategies are performed. The training and test sets are shuffled by cell lines, drugs, or drug combinations, which are described as leave-cell-line-out CV, leave-drug-out CV and leave-drug-combination-out CV. The performance results are listed in Table S7. The result of the leave-drug-combination-out CV of all algorithms is inferior to random 5-fold cross-validation. For the leave-drug-out and leave-cell-line-out CV, the results are similar to those mentioned by Preuer et al.[18] That is, all methods yield low predictive performance and thus do not generalize well on novel drugs or novel cell lines, while ForSyn has achieved the best performance in F1 score, AUPR, and MCC, followed by TranSynergy. On the metric recall, RUSBoost is still the best, which is similar to the results discussed in random CV.

Parameter analysis

Each layer of DF is the ensemble of multiple individual forests. In ForSyn, the RF-CSU unit dealing with data imbalance and the ETF-DR unit dealing with high-dimensional features are designed. This subsection will analyze the parameters that affect the performance of the RF-CUS, ETF-DR units, and ForSyn, respectively.

In the RF-CUS unit, the major parameter is the under-sampling ratio for the majority class, which is the ratio between the number of samples in the majority class before sampling and after sampling. We explore the effect of the number of base classifiers and under-sampling ratios on the performance of the RF-CSU unit. As shown in Figure 2G, the performance of the RF-CSU unit wins the best performance when the under-sampling ratio is 0.4. In addition, the increase in the number of decision trees dose not bring a significant improvement in model performance. Therefore, in the ForSyn, the number of decision trees in the RF-CSU unit is set to 100, and the under-sampling ratio for the majority class is set to 0.4.

Figure 2H shows the parameters that affect the performance of the ETF-DR unit. The longest dimension of the samples is 4,806 (data 8). We first sort all features by data complexity, and then perform a greedy backward shrinkage to iteratively reduce the feature dimension, with a step size of 300. From Figure 2H, the performance of the ETF-DR unit wins the best performance when the retained dimension is 1,806. In addition, when the number of decision trees is set to 100, the model has the best performance. Therefore, in the ForSyn, we first reduce the feature dimension of the training sample to 1,806, then train the ETF-DR unit, and set the number of base classifiers of the ETF-DR unit to 100.

Table 2 shows the F1 score of the ForSyn under different configurations. It is observed that the average ranks of ForSyn$^{(RFC\star2+ETFD\star2)}$ and ForSyn$^{(RFC\star3+ETFD\star3)}$ are the same, and the average rank of ForSyn$^{(RFC\star4+ETFD\star4)}$ model is slightly lower. It is inferred that as the unit number increases, the performance of the model does not increase obviously. According to the principle of Occam's razor (''entities should not be multiplied unnecessarily''), ForSyn$^{(RFC\star2+ETFD\star2)}$ is chosen as the best configuration for the proposed model. In addition, by observing the performance of ForSyn$^{(RFC\star4)}$ and ForSyn$^{(ETFD\star4)}$, it can be inferred that the ETF-DR unit has more advantages than the RF-CUS unit when processing drug combination dataset. If ForSyn

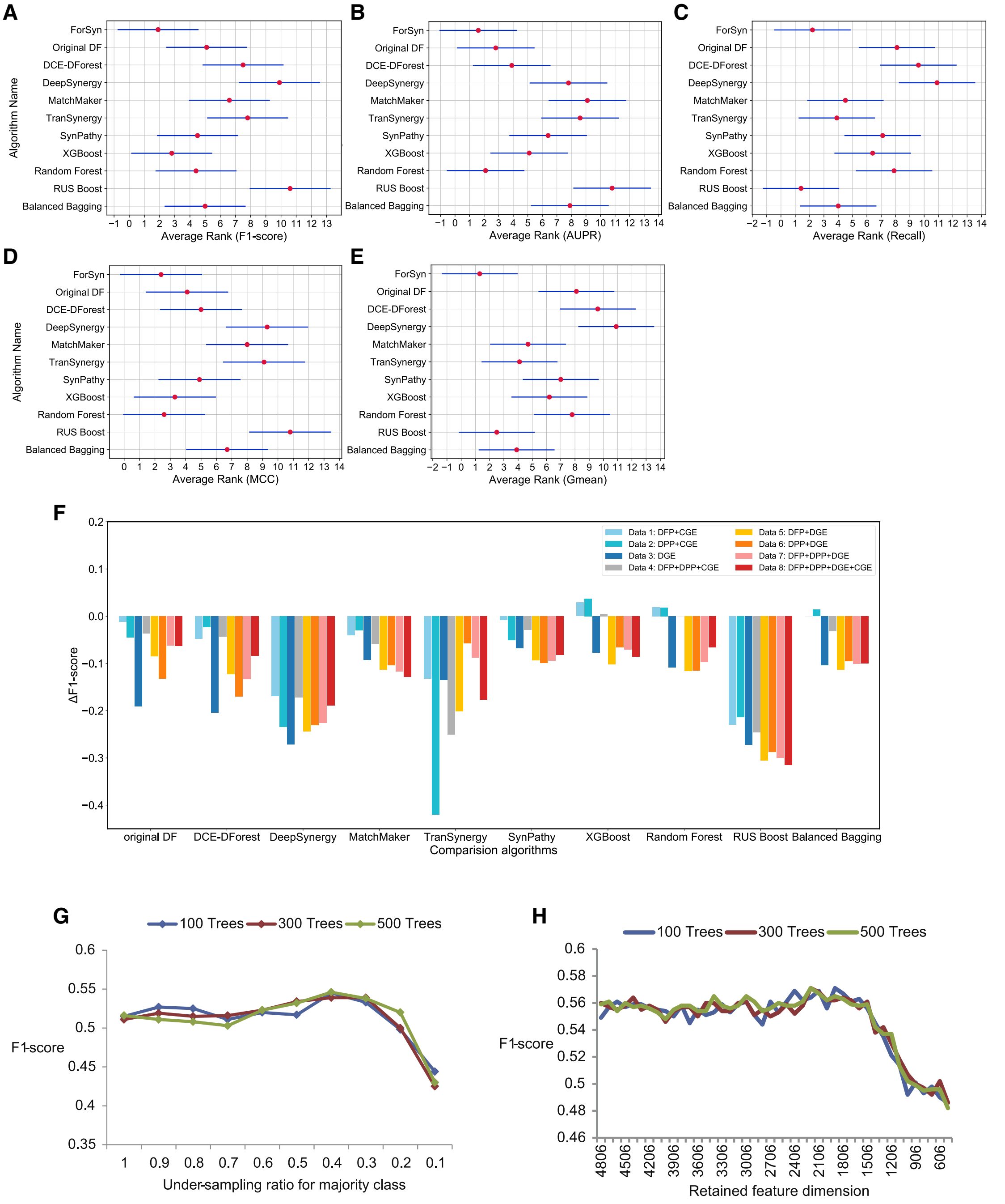


Figure 2. Performance evaluation of ForSyn

(A–E) According to Nemenyi test, the average rank of all algorithms tested on data 1–8 and five metrics: (A) F1 score, (B) AUPR, (C) recall, (D) MCC, and (E) G-mean. The average rank of each algorithm in eight datasets is marked as a red dot, and a horizontal line crossing the red dot indicates the range of CD value in Nemenyi test. The smaller the overlap between two horizontal bars, the more significant the difference between the two algorithms.

(F) The performance difference between ForSyn and other algorithms on F1 score under data 1–8. The y axis denotes ΔF1 between ForSyn and other comparison algorithms, $\Delta F1 = F1_{\text{comparison algorithms}} - F1_{\text{ForSyn}}$. A positive number indicates that the performance value of the comparison algorithm exceeds ForSyn, while a negative number indicates that ForSyn is superior to the comparison algorithm.

(G) The impact of the number of base classifiers and the under-sampling ratio on performance of ForSyn's RF-CSU unit. The y axis represents the F1 score, and the x axis represents the under-sampling ratio for the majority class with a value range of 0.1–1. The blue, red, and green lines represent the RF-CUS unit containing 100, 300, and 500 decision trees, respectively.

(H) The impact of the number of base classifiers and the retained feature dimension on performance of ForSyn's ETF-DR unit.

Table 2. Performance of ForSyn in different configurations under F1 score

	ForSyn$^{(RFC\star2+ETFD\star2)}$	ForSyn$^{(RFC\star3+ETFD\star3)}$	ForSyn$^{(RFC\star4+ETFD\star4)}$	ForSyn$^{(RFC\star4)}$	ForSyn$^{(ETFD\star4)}$
Data 1	0.499$_{(2.5)}$	0.491$_{(4.0)}$	0.499$_{(2.5)}$	0.341$_{(5.0)}$	0.510$_{(1.0)}$
Data 2	0.496$_{(3.0)}$	0.525$_{(1.0)}$	0.501$_{(2.0)}$	0.364$_{(5.0)}$	0.477$_{(4.0)}$
Data 3	0.519$_{(3.0)}$	0.543$_{(1.5)}$	0.543$_{(1.5)}$	0.327$_{(5.0)}$	0.460$_{(4.0)}$
Data 4	0.529$_{(1.0)}$	0.524$_{(2.0)}$	0.519$_{(3.0)}$	0.349$_{(5.0)}$	0.475$_{(4.0)}$
Data 5	0.568$_{(3.0)}$	0.575$_{(1.5)}$	0.575$_{(1.5)}$	0.335$_{(5.0)}$	0.497$_{(4.0)}$
Data 6	0.551$_{(1.5)}$	0.539$_{(3.0)}$	0.551$_{(1.5)}$	0.345$_{(5.0)}$	0.473$_{(4.0)}$
Data 7	0.564$_{(1.5)}$	0.564$_{(1.5)}$	0.547$_{(3.0)}$	0.354$_{(5.0)}$	0.493$_{(4.0)}$
Data 8	0.572$_{(1.0)}$	0.556$_{(2.0)}$	0.547$_{(4.0)}$	0.339$_{(5.0)}$	0.551$_{(3.0)}$
Average rank	2.1	2.1	2.4	5.0	3.5

The value in parentheses represents the ranking value of the corresponding performance. Taking data 8 as an example, the ForSyn$^{(RFC\star2+ETFD\star2)}$ on this dataset has the best performance (0.572) and is assigned a ranking value of 1.0. In data 6, the performance of ForSyn$^{(RFC\star2+ETFD\star2)}$ and ForSyn$^{(RFC\star4+ETFD\star4)}$ are the same (0.551), and they occupy the first and second positions, respectively, so their ranking values are uniformly assigned 1.5 ([1.0 + 2.0]/2). The average rank of each algorithm is defined as the average of its ranks on all datasets. RFC, RF-CUS unit; ETFD, ETF-DR unit; and the number behind each unit represents the number of units of this type on each cascade layer. For example, ForSyn$^{(RFC\star2+ETFD\star2)}$ means that each cascade layer is placed with two RF-CUS units and two ETF-DR units.

only uses a single type of unit (for example, ForSyn$^{(ETFD\star4)}$), it will cause the ensemble diversity of the cascade layer to decrease, which in turn leads to a decrease in the performance of the layer. However, the combination of different type units will promote the diversity of the cascade layer, which further improves the performance of the overall model.

Therefore, the optimal configuration of ForSyn is to place two RF-CUS units and two ETF-DR units in each cascade layer. Each RF-CUS unit contains 100 decision trees, and the under-sampling ratio is set to 0.4. Before training the ETF-DR unit, the proposed dimensionality reduction method is used to sort the feature space, and the first 1,806 dimensions of the sorted features are retained as the training set. The number of base classifiers in the ETF-DR unit is set to 100.

In addition, other tree-based forests are tested as the unit of ForSyn, including ADAboost (ADA), BAGging (BAG), and gradient boosting classifier (GBC). The base classifier of these models is the decision tree, and the parameters use default settings. Table 3 shows the performance comparison between the ForSyn and these derivative models. Under five evaluation metrics, the performance of the proposed ForSyn with two RF-CUS units and two ETF-DR units wins the best performance.

Subsequently, an ablation experiment on ForSyn is performed (Table S8). First, five different type units are placed on each cascade layer of DF, such as DF$^{(ADA\star1+BAG\star1+GBC\star1+RF\text{-}CUS\star1+ETF\text{-}DR\star1)}$, the performance of this model can be regarded as a benchmark for ablation experiment (0.562). Then the units will be removed to observe the change of performance. As shown in Table S8, when the ETF-DR unit is removed, the model performance drops the most, followed by the RF-CUS unit. It can be inferred that the two units we designed are more suitable as units in the cascade framework than other decision tree ensembles.

Cellular experiments of novel drug combinations

To confirm the efficacy of ForSyn, we further apply ForSyn to predict novel synergistic drug combination that have not been tested before. The cellular experiment is carried out on the predicted novel drug combinations. All drugs are combined in pairs, and the reported samples are removed. The remaining unmeasured samples are regarded as the novel drug combinations. According to the predicted probability of synergism class, eight drug combinations in the HT29 colorectal cell line with top predicted probability (Table S9) are selected to perform the cellular experiment. The synergistic potentials are observed on four drug combinations in the HT29 cell line, including erlotinib hydrochloride and AZD1775, erlotinib hydrochloride and MK-5108, etoposide and gefitinib, and erlotinib hydrochloride and dinaciclib (Figures 3A–3D).

Erlotinib hydrochloride is an inhibitor of the epidermal growth factor receptor tyrosine kinase (EGFR-TK). The EGFR has become an important therapeutic target for a variety of cancers.[57,58] The alterations of EGFR lead to cell growth, invasion, angiogenesis, and metastases. In colorectal cancer, 25%–77% of tumors overexpress EGFR.[59,60] There have been various EGFR inhibitors, such as erlotinib, an EGFR-TK inhibitor. Erlotinib has demonstrated efficacy against a range of solid tumor types including non-small-cell lung cancer (NSCLC), with more modest effects in colorectal cancer in phase I and II clinical trials.[61–63] Although the response rate of erlotinib is not satisfactory when used as monotherapy.[64] The combination therapy of erlotinib with other anticancer therapies should be more explored.

AZD1775 is a WEE1 inhibitor. It has been proved that the WEE1 gene could repair the DNA damage, which would limit the efficacy of DNA-damaging treatments in cancer cells.[65] The erlotinib has been found to suppress DNA damage repair in tumor cells.[64] The combination erlotinib and AZD1775 may enhance the sensitivity of tumor cells. MK-5108 is an Aurora-A kinase inhibitor. The synergistic effect has been observed in combined inhibition of the EGFR and Aurora-A pathways in cancer cells.[66] Aurora kinase inhibitors are active in combination with EGFR inhibition in a number of EGFR-mutant cell lines. Dinaciclib is a CDK inhibitor for CDK2, CDK5, CDK1, and CDK9. It has been reported that combined inhibition of EGFR and CDK9 resulted in reduced cell proliferation, accompanied by induction of apoptosis, G2-M cell-cycle arrest, inhibition of DNA

Table 3. Performance comparison of deep forest embedding different units based on data 8

Configuration	F1 score	AUPR	Recall	MCC	G-mean
$DF^{(ADA\star 2+BAG\star 2)}$	0.500	0.582	0.384	0.509	0.614
$DF^{(ADA\star 2+GBC\star 2)}$	0.484	0.559	0.350	0.508	0.588
$DF^{(BAG\star 2+GBC\star 2)}$	0.493	0.561	0.365	0.512	0.598
$DF^{(RF\text{-}CUS\star 2+ETF\text{-}DR\star 2)}$	0.572	0.591	0.537	0.535	0.722

DF, deep forest; ADA, ADAboost; BAG, BAGging; GBC, gradient boosting classifier.

replication and abrogation of CDK9-mediated transcriptional elongation, in contrast to monotherapy.[67]

In addition, giving gefitinib together with etoposide may kill more tumor cells (https://clinicaltrials.gov, NCT00483561). The phase II trial is studying how well giving gefitinib and etoposide works in treating patients with advanced prostate cancer that did not respond to hormone therapy. Gefitinib may stop the growth of tumor cells by blocking some of the enzymes needed for cell growth and by blocking blood flow to the tumor. Etoposide works in different ways to stop the growth of tumor cells, either by killing the cells or by stopping them from dividing.

Moreover, to investigate the potential false negatives of ForSyn, we also pick up five drug combinations in the HT29 cell line predicted as negative (non-synergistic) with highest probability to perform the same cellular experiment. The experimental results are shown in Figures 3E–3I. It is observed that all the five samples predicted by ForSyn as negative samples are verified as negative by cellular experiments. The CI of three drug combinations even exceeded 100, indicating the strong potential of non-synergistic. This further demonstrates the prediction accuracy of ForSyn in the negative samples.

Interpretable analysis of feature importance

Model interpretation is of paramount importance in machine learning-based biomedical studies. In this study, ForSyn can evaluate the importance of each feature in the prediction process. ForSyn quantify the global relationship between each feature and the output by evaluating the feature importance value (FIV). Then, the FIVs extracted by ForSyn is analyzed from three aspects, the association with prediction process, the contribution of feature types, and the biological analysis of key features. All the FIVs are calculated on data 8 because it contains all the feature types.

Association with prediction process

First, two experiments are performed to show the relationship between FIVs and prediction process from the global and local perspectives, including the layer-by-layer error correction, and the difference of FIVs between different layers.

ForSyn is a deep learning method with multiple layers, which can be adaptively expanded according to the performance gain. In this section, we trained a ForSyn model with three layers, the classification results and FIVs of each layer in ForSyn are analyzed. In the first experiment, the layer-by-layer error correction capability of ForSyn is visualized in the feature space through FIVs (Figures 4A–4C). The positive samples (synergistic drug combinations) that are wrongly classified at each layer are extracted. Then the top two features on the basis of the FIVs are used to project the mis-classified samples into a two-dimensional space. Figures 4A–4C shows the error correction result of each layer. The blue dots represent the mis-classified positive samples by the first layer of ForSyn. The red "+" represents the samples that are correctly classified at the second and last layers of ForSyn. From Figures 4A and 4B, the number of red plus signs appears more, indicating that the growth of the layer brings a significant performance improvement. In Figures 4B and 4C, the number of red plus signs increases slightly, indicating that the layer stops growing and the performance gradually converges. In addition, there are samples that cannot be corrected in the final layer, some of which may be related to the correctness of labels in the dataset. There may still be several incorrectly labeled noisy samples in the dataset because of the existence of experimental noise, as mentioned by Malyutina et al.[68]

In the local analysis of the association with prediction process, the difference of FIVs between different layers is evaluated. The FIV of each feature in the *l*th layer is calculated according to Equation 12 in STAR Methods. Then a rank vector is generated by sorting the FIVs of all features, so as to generate the rank vectors of three layers of ForSyn. Finally, the Wilcoxon signed rank test[69] is used to evaluate the significant differences between the three rank vectors. The p value for layer 1 vs. layer 2 is 0.958, and that for layer 2 vs. layer 3 is 0.972. The original hypothesis of this test is that there is no significant difference between paired vectors. Both p values are greater than 0.05, failing to reject the original hypothesis. That is, there is no significant difference between the paired FIVs' rank vectors in the layers of ForSyn.

Contribution of feature types

The key features based on FIVs are then analyzed. The most contributing feature type is first investigated. The feature set is composed of four feature types, DMF, DPP, DGE, and CGE. When analyzing the FIVs, it should be noted that not all features participate in the whole prediction process. In the ETF-DR unit of ForSyn, a greedy dimension reduction method is applied to select 1,806-dimensional (see Parameter analysis) features to achieve the prediction task. Therefore, only the 1,806 features participate in the whole prediction process, including 1,037 DMFs, 31 DPPs, 600 DGEs, and 138 CGEs. The FIV of each feature is shown in Figure 4D. The red line in Figure 4D represents the average FIV of all features, which is 0.000554 (1/1,806). Figure 4E divides the features into two groups, the features that are greater than and less than the average FIV. It further shows the contribution of each feature type in the two groups. The contribution is calculated by summing the FIVs of features in a feature type. From Figures 4D and 4E, 768 features are greater than the average FIV, and the contributions of the 768 features are accounted for 74%. Therefore, we believe that these 768 features are top contributing features for prediction process. Among the 768 features, there are 107 DMFs, 17 DPPs, 582 DGEs, and 62 CGEs, with contributions of 15.35%, 2.86%, 49.70%, and 6.09%, respectively (Figure 4E). The results show that DGE plays a key role in the prediction process. Although there are many DMFs among 1,806 features, the contribution of most DMF features is lower than the average FIV (Figures 4D and 4E).

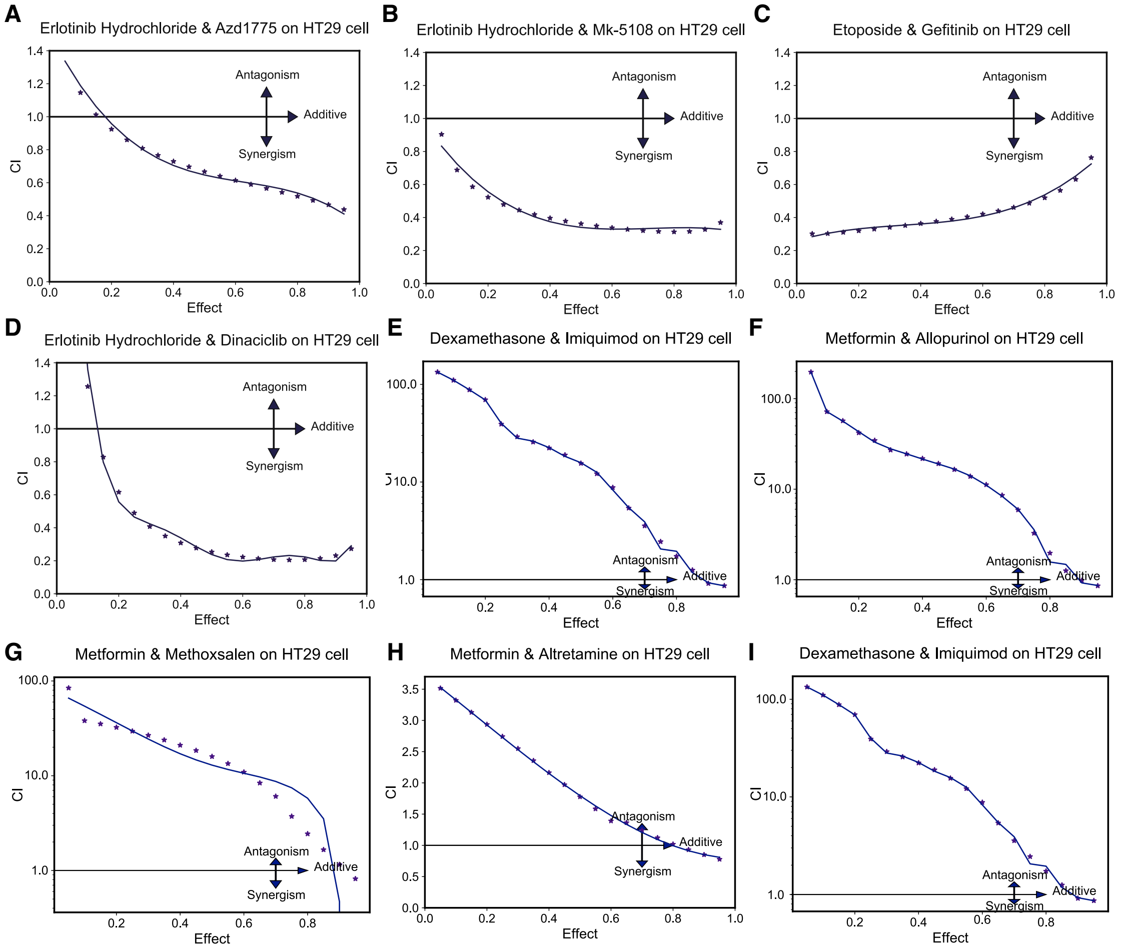


Figure 3. The result of cellular experiment of ForSyn

(A–D) The effect-CI plot of top predicted synergistic drug combinations tested in the HT29 cell line. CI < 1 indicates that the drug combination has synergistic effect, while CI > 1 indicates the non-synergistic effect.

(E–I) The effect-CI plot of top predicted non-synergistic drug combinations tested in the HT29 cell line.

Biological analysis of key features

Next, the biological analysis is performed on the key DGE features extracted by ForSyn. The 479 genes (with duplication removed) involved in the 582 DGE features that are greater than the average FIV are extracted. A global analysis on the extracted genes is carried out, including two kinds of gene enrichment analysis on Kyoto Encyclopedia of Genes and Genomes (KEGG) pathway and Gene Ontology Biological Process (GOBP). Enrichment results show that these genes are significantly enriched in 67 KEGG pathways and 518 GOBPs (adjusted $p < 0.01$). The top 20 enrichment results are shown in Figures 4F and 4G. KEGG pathway enrichment result shows multiple significant biological pathways that are closely related to cancer (Figure 4F). According to the characteristics of these pathways, they can be divided into four categories: specific cancer pathways (colorectal cancer, pancreatic cancer, etc.), regulation process of cancer (cellular senescence, cell cycle, etc.), oncogenic virus infection (Kaposi sarcoma-associated herpesvirus infection, etc.) and immune inflammation (lipid and atherosclerosis, etc.). For enrichment result of top 20 GOBP (Figure 4G), the key genes are more concentrated in the response to stimulus, especially the response to oxidative stress.

After the global analysis of the key genes, the cancer-specific key genes in DGE in different cell lines are further investigated. Four cell lines (HT29, A549, MCF7, and PC3) with more than 500 samples are selected to train the ForSyn respectively. Then the key DGE features with top FIVs of four cell lines are obtained. The top 10 genes involved in these key DGE features may play a key role in corresponding cancer cell lines, as shown in Table S10. For example, in A549 lung cancer cell line, CCND3 and TSPAN14 genes are identified as the top contributing genes. Song et al.[70] proposed that CCND3 could serve as potential biomarkers and provide a theoretical basis for the pathogenesis of lung adenocarcinoma. And TSPAN14 gene is also proposed as an indicator of NSCLC metastasis and progression.[71] In the HT29 colorectal cell line, the

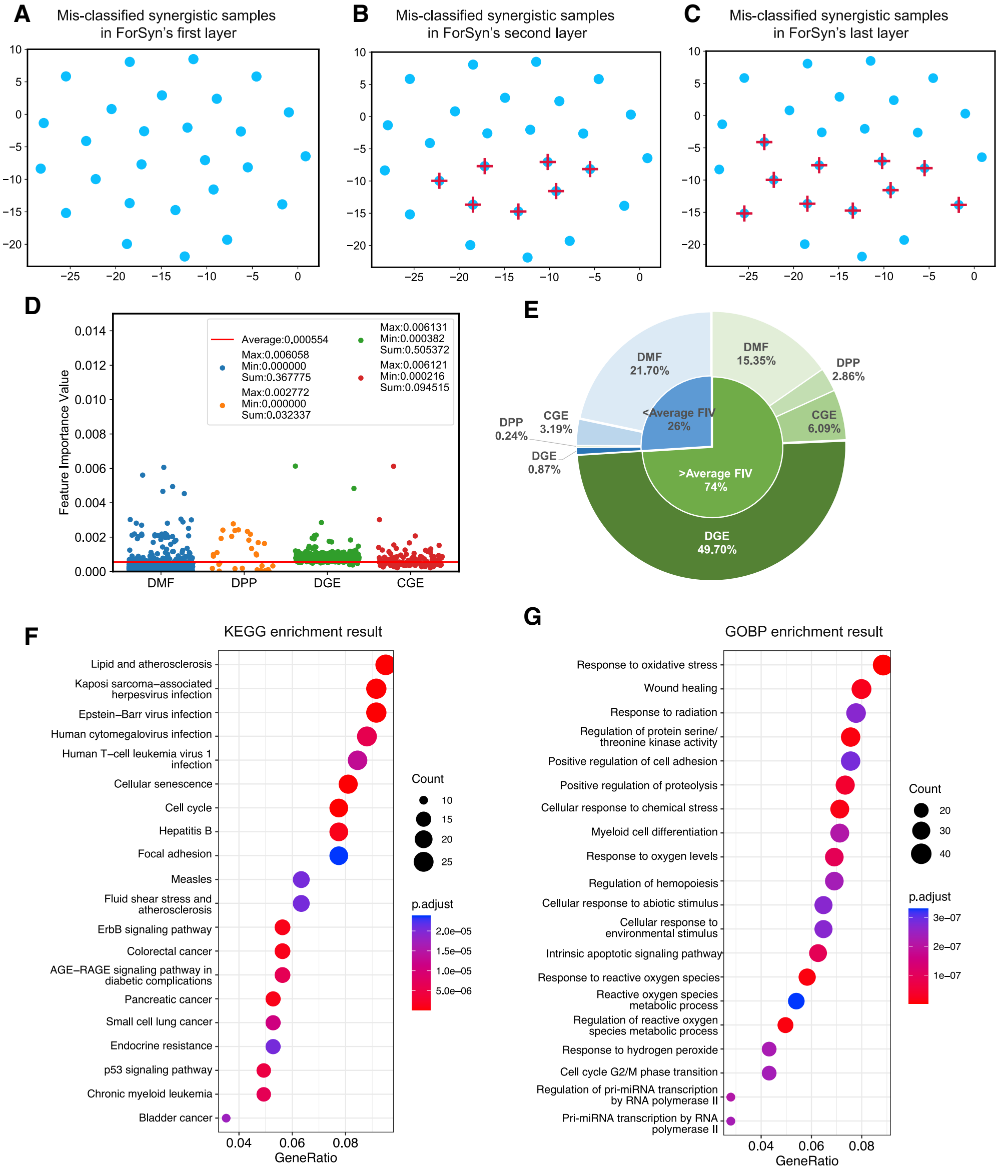


Figure 4. The interpretable analysis result of ForSyn

(A–C) The top two features sorted by FIVs are used to visualize the ForSyn's layer-by-layer error correction of mis-classified positive (synergistic) samples. The blue dots represent the mis-classified positive samples by the first layer of ForSyn. The red plus sign represents the samples that are correctly classified at the second and last layers of ForSyn.

(D) The FIV of each feature in four feature types. The red line indicates the average FIV all features.

(E) The contribution of each feature type in two groups, which are the features that are greater than and less than the average FIV.

(F and G) The top 20 enrichment results of KEGG pathway and Gene Ontology Biological Process, which are obtained by key genes involved in the key DGE features.

CAMSAP2 gene has been proved to be a promising therapeutic target for the treatment of metastatic colorectal cancer patients.[72] PLOD3 has also been proved to be a potential biomarker for CRC diagnosis and prognosis prediction.[73] In MCF7 breast cancer cell and PC3 prostate carcinoma cell line, the top contributing genes, PGM1 and SPRED2, as well as SIRT3 and UFM1, are also proved to play a key role in breast and prostate cancers.[74–78]

DISCUSSION

In this study, we propose a new algorithm, ForSyn, to predict synergistic drug combinations in different cancer cell lines. Two novel forest types are designed to embed in ForSyn, including the RF-CSU unit dealing with data imbalance and the ETF-DR unit dealing with high-dimensional features. The ForSyn can effectively solve the problems of class imbalanced, and high feature dimension in the medium-scale datasets. Compared with 12 advanced algorithms on five metrics, ForSyn ranks first in four metrics, F1 score, AUPR, MCC and G-mean. Two statistical tests confirm that ForSyn perform significantly better than other algorithms in most cases. Next, the different configurations of ForSyn are analyzed. The results show that the under-sampling ratio for the majority class in RF-CSU, the feature dimension of the training sample in ETF-DR, the number of base classifiers, the types and numbers of units have influence on the performance of ForSyn. In addition, the novel synergistic drug combinations predicted by ForSyn are verified by cellular experiment, showing the predictive ability of ForSyn. Finally, a systematic interpretable analysis of the FIVs evaluated by ForSyn is performed. The layer-by-layer error correction and the difference of FIVs between different layers show the association between FIVs with prediction process. By summing the FIVs of each feature type, the DGE has been proved to play a critical role in the prediction process. Then the key genes involved in the key DGE features are explored by enrichment analysis. The key genes extracted by ForSyn may have potential effects on corresponding cancers.

Two forest types are designed in ForSyn, including RF-CSU and ETF-DR. The reason for choosing RF and ETF is that both models have their own advantages in dealing with high-dimensional and unbalanced data. The RF selects $\sqrt{d}$ (where d is the dimension of the training data) features for each decision tree. Thus, the high feature dimension will not have a great negative impact on the performance of the RF, and effectively solving the problem of data imbalance is the key factor to improve the performance of RF. In the ETF model, the tree will continuously grow until each leaf node contains samples of the same class. Thus, the ETF has some advantages when dealing with imbalanced data. For example, the pure leaf node that stores minority samples can effectively identify unknown minority samples. However, the high feature dimension and the behavior of randomly selecting the feature, which deepens the depth of the tree and easily causes over-fitting. The effective dimension reduction methods may reduce computational cost and avoid over-fitting of the ETF. Thus, to obtain an excellent model to deal with the imbalanced and high-dimensional data, we design the modules of imbalanced data process and dimensionality reduction on RF and ETF respectively.

For the input feature data, the DGE and CGE can be quickly obtained at low cost through L1000 method or published predicted models when there are new drugs and cell lines to be predicted. In this study, the DGE and CGE are obtained from the National Institutes of Health (NIH) Library of Integrated Network-Based Cellular Signatures (LINCS)[79] database. In LINCS database, the data are obtained using the L1000 method, which is a low-cost, high-throughput method and only needs 1,058 probes for 978 landmark transcripts and 80 control transcripts. The reagent cost of the L1000 assay is approximately $2. The 978 landmarks have been shown to be sufficient to recover ~80% of the information in the full transcriptome. In addition, DGE and CGE also can be generated or predicted by machine learning models.[80,81] For example, Zhu et al.[80] have proposed a deep learning-based model, DLEPS, using SMILES of molecules to predict the 978-dimentional DGE obtained from LINCS database. DLEPS has been validated in the use of screening potential drugs in obesity, hyperuricemia and nonalcoholic steatohepatitis.

ForSyn has shown an excellent predictive performance in drug combination prediction, which is validated by computational and biological experimental results. The novel units designed in ForSyn can largely solve the problems of imbalanced and high-dimensional data. Both are common problems in the datasets of drug-related biomedical studies. We hope that the propose of ForSyn can not only apply narrow down the candidates of drug combinations for experimental validations but also provide insights for other studies in drug discovery.

Limitations of the study

Although ForSyn shows excellent prediction performance and interpretability, this study is limited by the number of training samples when using DGE and CGE as features. The importance of DGE has been shown in this study. In future work, we expect that the scale of the training dataset will expand with the accumulation of DGE, and the performance and interpretability of EC-DFR would be further improved accordingly. In addition, the predictive model cannot generalize well on novel drugs or novel cell lines, which is an inherent problem in drug combination prediction and should be explored in future work. Finally, some potential drug combinations and key genes has been found on the basis of ForSyn. The key factors should be further investigated through more biological experiments.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - ○ Lead contact
  - ○ Materials availability
  - ○ Data and code availability
- EXPERIMENTAL MODEL AND SUBJECT DETAILS
- METHOD DETAILS
  - ○ Drug combination dataset
  - ○ Feature set
  - ○ Description of ForSyn
  - ○ Comparison methods
  - ○ Evaluation metrics
  - ○ Cross validations
  - ○ Evaluation of feature importance value
  - ○ Drug combination screening
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.crmeth.2023.100411.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (grants 62103436 and 61772023), the National Key Research and Development Program of China (grant 2019QY1803), the Natural Science Foundation of Fujian Province (grant 2022J01707), the High-Level Talents Research Start-Up Project of Fujian Medical University (grant XRCZX2021025), and the Fujian Science and Technology Plan Industry-University-Research Cooperation Project (grant 2021H6015).

AUTHOR CONTRIBUTIONS

L.W., J.G., K.L., S.H., and X.B. designed the study and wrote the manuscript. L.W. and Y.W. acquired the data. J.G. and K.L. designed and applied the ForSyn model. L.W., J.G., B.S., and Q.W. analyzed the results. Y.Z. performed the cellular experiments. X.B., S.H., and K.L. supervised the research.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: July 1, 2022
Revised: November 27, 2022
Accepted: January 27, 2023
Published: February 21, 2023

REFERENCES

1. Jaaks, P., Coker, E.A., Vis, D.J., Edwards, O., Carpenter, E.F., Leto, S.M., Dwane, L., Sassi, F., Lightfoot, H., Barthorpe, S., et al. (2022). Effective drug combinations in breast, colon and pancreatic cancer cells. Nature *603*, 166–173. https://doi.org/10.1038/s41586-022-04437-2.
2. Narayan, R.S., Molenaar, P., Teng, J., Cornelissen, F.M.G., Roelofs, I., Menezes, R., Dik, R., Lagerweij, T., Broersma, Y., Petersen, N., et al. (2020). A cancer drug atlas enables synergistic targeting of independent drug vulnerabilities. Nat. Commun. *11*, 2935. https://doi.org/10.1038/s41467-020-16735-2.
3. Housman, G., Byler, S., Heerboth, S., Lapinska, K., Longacre, M., Snyder, N., and Sarkar, S. (2014). Drug resistance in cancer: an overview. Cancers *6*, 1769–1792. https://doi.org/10.3390/cancers6031769.
4. Chou, T.-C. (2006). Theoretical basis, experimental design, and computerized simulation of synergism and antagonism in drug combination studies. Pharmacol. Rev. *58*, 621–681. https://doi.org/10.1124/pr.58.3.10.
5. Wu, L., Wen, Y., Leng, D., Zhang, Q., Dai, C., Wang, Z., Liu, Z., Yan, B., Zhang, Y., Wang, J., et al. (2022). Machine learning methods, databases and tools for drug combination prediction. Briefings Bioinf. *23*, bbab355. https://doi.org/10.1093/bib/bbab355.
6. Palmer, A.C., and Sorger, P.K. (2017). Combination cancer therapy can confer benefit via patient-to-patient variability without drug additivity or synergy. Cell *171*, 1678–1691.e13. https://doi.org/10.1016/j.cell.2017.11.009.
7. Ianevski, A., Giri, A.K., Gautam, P., Kononov, A., Potdar, S., Saarela, J., Wennerberg, K., and Aittokallio, T. (2019). Prediction of drug combination effects with a minimal set of experiments. Nat. Mach. Intell. *1*, 568–577. https://doi.org/10.1038/s42256-019-0122-4.
8. Sheng, Z., Sun, Y., Yin, Z., Tang, K., and Cao, Z. (2018). Advances in computational approaches in identifying synergistic drug combinations. Briefings Bioinf. *19*, 1172–1182. https://doi.org/10.1093/bib/bbx047.
9. Ramsay, R.R., Popovic-Nikolic, M.R., Nikolic, K., Uliassi, E., and Bolognesi, M.L. (2018). A perspective on multi-target drug discovery and design for complex diseases. Clin. Transl. Med. *7*, 3. https://doi.org/10.1186/s40169-017-0181-2.
10. Lehár, J., Krueger, A.S., Avery, W., Heilbut, A.M., Johansen, L.M., Price, E.R., Rickles, R.J., Short, G.F., III, Staunton, J.E., Jin, X., et al. (2009). Synergistic drug combinations tend to improve therapeutically relevant selectivity. Nat. Biotechnol. *27*, 659–666. https://doi.org/10.1038/nbt.1549.
11. Zhao, X.-M., Iskar, M., Zeller, G., Kuhn, M., van Noort, V., and Bork, P. (2011). Prediction of drug combinations by integrating molecular and pharmacological data. PLoS Comput. Biol. *7*, e1002323. https://doi.org/10.1371/journal.pcbi.1002323.
12. Bleicher, K.H., Böhm, H.J., Müller, K., and Alanine, A.I. (2003). Hit and lead generation: beyond high-throughput screening. Nat. Rev. Drug Discov. *2*, 369–378. https://doi.org/10.1038/nrd1086.
13. Morris, M.K., Clarke, D.C., Osimiri, L.C., and Lauffenburger, D.A. (2016). Systematic analysis of quantitative logic model ensembles predicts drug combination effects on cell signaling networks. CPT Pharmacometrics Syst. Pharmacol. *5*, 544–553. https://doi.org/10.1002/psp4.12104.
14. Feala, J.D., Cortes, J., Duxbury, P.M., Piermarocchi, C., McCulloch, A.D., and Paternostro, G. (2010). Systems approaches and algorithms for discovery of combinatorial therapies. Wiley Interdiscip. Rev. Syst. Biol. Med. *2*, 181–193. https://doi.org/10.1002/wsbm.51.
15. Cheng, F., Kovács, I.A., and Barabási, A.L. (2019). Network-based prediction of drug combinations. Nat. Commun. *10*, 1197. https://doi.org/10.1038/s41467-019-09186-x.
16. Tang, J., Karhinen, L., Xu, T., Szwajda, A., Yadav, B., Wennerberg, K., Aittokallio, T., and Aittokallio, T. (2013). Target inhibition networks: predicting selective combinations of druggable targets to block cancer survival pathways. PLoS Comput. Biol. *9*, e1003226. https://doi.org/10.1371/journal.pcbi.1003226.
17. Lee, J.-H., Kim, D.G., Bae, T.J., Rho, K., Kim, J.-T., Lee, J.-J., Jang, Y., Kim, B.C., Park, K.M., and Kim, S. (2012). CDA: combinatorial drug discovery using transcriptional response modules. PLoS One *7*, e42573. https://doi.org/10.1371/journal.pone.0042573.
18. Preuer, K., Lewis, R.P.I., Hochreiter, S., Bender, A., Bulusu, K.C., and Klambauer, G. (2018). DeepSynergy: predicting anti-cancer drug synergy with Deep Learning. Bioinformatics *34*, 1538–1546. https://doi.org/10.1093/bioinformatics/btx806.
19. Kuru, H.I., Tastan, O., and Cicek, A.E. (2022). MatchMaker: a deep learning framework for drug synergy prediction. IEEE ACM Trans. Comput. Biol. Bioinf *19*, 2334–2344. https://doi.org/10.1109/TCBB.2021.3086702.
20. Chen, G., Tsoi, A., Xu, H., and Zheng, W.J. (2018). Predict effective drug combination by deep belief network and ontology fingerprints. J. Biomed. Inf. *85*, 149–154. https://doi.org/10.1016/j.jbi.2018.07.024.
21. Zhang, T., Zhang, L., Payne, P.R.O., and Li, F. (2021). Synergistic drug combination prediction by integrating multiomics data in deep learning models. Methods Mol. Biol. *2194*, 223–238. https://doi.org/10.1007/978-1-0716-0849-4_12.
22. Liu, Q., and Xie, L. (2021). TranSynergy: mechanism-driven interpretable deep neural network for the synergistic prediction and pathway deconvolution of drug combinations. PLoS Comput. Biol. *17*, e1008653. https://doi.org/10.1371/journal.pcbi.1008653.
23. Wang, J., Liu, X., Shen, S., Deng, L., and Liu, H. (2022). DeepDDS: deep graph neural network with attention mechanism to predict synergistic drug combinations. Briefings Bioinf. *23*, bbab390. https://doi.org/10.1093/bib/bbab390.
24. Johnson, J.M., and Khoshgoftaar, T.M. (2019). Survey on deep learning with class imbalance. J. Big Data *6*, 27. https://doi.org/10.1186/s40537-019-0192-5.
25. Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., and Bing, G. (2017). Learning from class-imbalanced data: review of methods and applications. Expert Syst. Appl. *73*, 220–239. https://doi.org/10.1016/j.eswa.2016.12.035.
26. Anand, R., Mehrotra, K.G., Mohan, C.K., and Ranka, S. (1993). An improved algorithm for neural network classification of imbalanced training sets. IEEE Trans. Neural Network. *4*, 962–969. https://doi.org/10.1109/72.286891.
27. Bollenbach, T., and Kishony, R. (2011). Resolution of gene regulatory conflicts caused by combinations of antibiotics. Mol. Cell *42*, 413–425. https://doi.org/10.1016/j.molcel.2011.04.016.
28. López-Maury, L., Marguerat, S., and Bähler, J. (2008). Tuning gene expression to changing environments: from rapid responses to evolutionary adaptation. Nat. Rev. Genet. *9*, 583–593. https://doi.org/10.1038/nrg2398.

29. Geva-Zatorsky, N., Dekel, E., Cohen, A.A., Danon, T., Cohen, L., and Alon, U. (2010). Protein dynamics in drug combinations: a linear superposition of individual-drug responses. Cell *140*, 643–651. https://doi.org/10.1016/j.cell.2010.02.011.

30. Lukačišin, M., and Bollenbach, T. (2019). Emergent gene expression responses to drug combinations predict higher-order drug interactions. Cell Syst. *9*, 423–433.e3. https://doi.org/10.1016/j.cels.2019.10.004.

31. Zhou, Z.H., and Feng, J. (2017). Deep forest: towards an alternative to deep neural networks. IJCAI, 3553–3559.

32. Zhou, Z.-H., and Feng, J. (2019). Deep forest. Natl. Sci. Rev. *6*, 74–86. https://doi.org/10.1093/nsr/nwy108.

33. Lin, W., Wu, L., Zhang, Y., Wen, Y., Yan, B., Dai, C., Liu, K., He, S., and Bo, X. (2022). An enhanced cascade-based deep forest model for drug combination prediction. Briefings Bioinf. *23*, bbab562. https://doi.org/10.1093/bib/bbab562.

34. Zhou, M., Zeng, X., and Chen, A. (2019). Deep forest hashing for image retrieval. Pattern Recognit. DAGM. *95*, 114–127. https://doi.org/10.1016/j.patcog.2019.06.005.

35. Guo, Y., Liu, S., Li, Z., and Shang, X. (2017). Towards the classification of cancer subtypes by using cascade deep forest model in gene expression data. IEEE international conference on bioinformatics and biomedicine (BIBM), 1664–1669.

36. Zhang, Y.-L., Zhou, J., Zheng, W., Feng, J., Li, L., Liu, Z., Li, M., Zhang, Z., Chen, C., Li, X., et al. (2019). Distributed deep forest and its application to automatic detection of cash-out fraud. ACM Trans. Intell. Syst. Technol. *10*, 1–19. https://doi.org/10.1145/3342241.

37. Zhang, W., Xue, Z., Li, Z., and Yin, H. (2022). DCE-DForest: a deep forest model for the prediction of anticancer drug combination effects. Comput. Math. Methods Med. *2022*, 8693746. https://doi.org/10.1155/2022/8693746.

38. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: a robustly optimized bert pretraining approach. Preprint at arXiv. https://doi.org/10.48550/arXiv.1907.11692.

39. Subramanian, A., Narayan, R., Corsello, S.M., Peck, D.D., Natoli, T.E., Lu, X., Gould, J., Davis, J.F., Tubelli, A.A., Asiedu, J.K., et al. (2017). A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. Cell *171*, 1437–1452.e17. https://doi.org/10.1016/j.cell.2017.10.049.

40. Frey, B.J., and Dueck, D. (2007). Clustering by passing messages between data points. Science *315*, 972–976. https://doi.org/10.1126/science.1136800.

41. Zagidullin, B., Aldahdooh, J., Zheng, S., Wang, W., Wang, Y., Saad, J., Malyutina, A., Jafari, M., Tanoli, Z., Pessia, A., and Tang, J. (2019). DrugComb: an integrative cancer drug combination data portal. Nucleic Acids Res. *47*, W43–W51. https://doi.org/10.1093/nar/gkz337.

42. Liu, H., Zhang, W., Zou, B., Wang, J., Deng, Y., and Deng, L. (2020). DrugCombDB: a comprehensive database of drug combinations toward the discovery of combinatorial therapy. Nucleic Acids Res. *48*, D871–D881. https://doi.org/10.1093/nar/gkz1007.

43. Menden, M.P., Wang, D., Mason, M.J., Szalai, B., Bulusu, K.C., Guan, Y., Yu, T., Kang, J., Jeon, M., Wolfinger, R., et al. (2019). Community assessment to advance computational prediction of cancer drug combinations in a pharmacogenomic screen. Nat. Commun. *10*, 2674. https://doi.org/10.1038/s41467-019-09799-2.

44. Malyutina, A., Majumder, M.M., Wang, W., Pessia, A., Heckman, C.A., and Tang, J. (2019). Drug combination sensitivity scoring facilitates the discovery of synergistic and efficacious drug combinations in cancer. PLoS Comput. Biol. *15*, e1006752. https://doi.org/10.1371/journal.pcbi.1006752.

45. Zagidullin, B., Wang, Z., Guan, Y., Pitkänen, E., and Tang, J. (2021). Comparative analysis of molecular fingerprints in prediction of drug combination effects. Briefings Bioinf. *22*, bbab291. https://doi.org/10.1093/bib/bbab291.

46. Huang, Y.-A., You, Z.-H., and Chen, X. (2018). A systematic prediction of drug-target interactions using molecular fingerprints and protein sequences. Curr. Protein Pept. Sci. *19*, 468–478. https://doi.org/10.2174/1389203718666161122103057.

47. Hessler, G., and Baringhaus, K.-H. (2018). Artificial intelligence in drug design. Molecules *23*, 2520. https://doi.org/10.3390/molecules23102520.

48. Rifaioglu, A.S., Cetin Atalay, R., Cansen Kahraman, D., Doğan, T., Martin, M., and Atalay, V. (2021). MDeePred: novel multi-channel protein featurization for deep learning-based binding affinity prediction in drug discovery. Bioinformatics *37*, 693–704. https://doi.org/10.1093/bioinformatics/btaa858.

49. Xing, J., Shankar, R., Drelich, A., Paithankar, S., Chekalin, E., Dexheimer, T., Rajasekaran, S., Tseng, C.-T.K., and Chen, B. (2020). Reversal of infected host gene expression identifies repurposed drug candidates for COVID-19. Preprint at bioRxiv. https://doi.org/10.1101/2020.04.07.030734.

50. Cano, J.-R. (2013). Analysis of data complexity measures for classification. Expert Syst. Appl. *40*, 4820–4831. https://doi.org/10.1016/j.eswa.2013.02.025.

51. Sun, M., Liu, K., Wu, Q., Hong, Q., Wang, B., and Zhang, H. (2019). A novel ECOC algorithm for multiclass microarray data classification based on data complexity analysis. Pattern Recognit. DAGM. *90*, 346–362. https://doi.org/10.1016/j.patcog.2019.01.047.

52. Tang, Y.C., and Gottlieb, A. (2022). SynPathy: predicting drug synergy through drug-associated pathways using deep learning. Mol. Cancer Res. *20*, 762–769. https://doi.org/10.1158/1541-7786.MCR-21-0735.

53. Chen, T.Q., and Guestrin, C. (2016). XGBoost: a scalable tree boosting system. In Kdd'16: Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, pp. 785–794. https://doi.org/10.1145/2939672.2939785.

54. Seiffert, C., Khoshgoftaar, T.M., Van Hulse, J., and Napolitano, A. (2010). RUSBoost: a hybrid approach to alleviating class imbalance. IEEE Trans. Syst. Man Cybern. A. *40*, 185–197. https://doi.org/10.1109/TSMCA.2009.2029559.

55. Hido, S., Kashima, H., and Takahashi, Y. (2009). Roughly balanced bagging for imbalanced data. Stat. Anal. Data Min. *2*, 412–426. https://doi.org/10.1002/sam.10061.

56. Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. 7, 1–30.

57. Meyerhardt, J.A., Zhu, A.X., Enzinger, P.C., Ryan, D.P., Clark, J.W., Kulke, M.H., Earle, C.C., Vincitore, M., Michelini, A., Sheehan, S., and Fuchs, C.S. (2006). Phase II study of capecitabine, oxaliplatin, and erlotinib in previously treated patients with metastatic colorectal cancer. J. Clin. Oncol. *24*, 1892–1897. https://doi.org/10.1200/JCO.2005.05.3728.

58. Mendelsohn, J., and Baselga, J. (2000). The EGF receptor family as targets for cancer therapy. Oncogene *19*, 6550–6565. https://doi.org/10.1038/sj.onc.1204082.

59. Mayer, A., Takimoto, M., Fritz, E., Schellander, G., Kofler, K., and Ludwig, H. (1993). The prognostic significance of proliferating cell nuclear antigen, epidermal growth factor receptor, and mdr gene expression in colorectal cancer. Cancer *71*, 2454–2460. https://doi.org/10.1002/1097-0142(19930415)71:8<2454::AID-CNCR2820710805>3.0.CO;2-2.

60. Salomon, D.S., Brandt, R., Ciardiello, F., and Normanno, N. (1995). Epidermal growth factor-related peptides and their receptors in human malignancies. Crit. Rev. Oncol. Hematol. *19*, 183–232. https://doi.org/10.1016/1040-8428(94)00144-I.

61. Van Cutsem, E., Verslype, C., Beale, P., Clarke, S., Bugat, R., Rakhit, A., Fettner, S.H., Brennscheidt, U., Feyereislova, A., and Delord, J.P. (2008). A phase Ib dose-escalation study of erlotinib, capecitabine and oxaliplatin in metastatic colorectal cancer patients. Ann. Oncol. *19*, 332–339. https://doi.org/10.1093/annonc/mdm452.

62. Pérez-Soler, R., Chachoua, A., Hammond, L.A., Rowinsky, E.K., Huberman, M., Karp, D., Rigas, J., Clark, G.M., Santabárbara, P., and Bonomi,

P. (2004). Determinants of tumor response and survival with erlotinib in patients with non—small-cell lung cancer. J. Clin. Oncol. *22*, 3238–3247. https://doi.org/10.1200/JCO.2004.11.057.

63. Tang, P.A., Tsao, M.-S., and Moore, M.J. (2006). A review of erlotinib and its clinical use. Expet Opin. Pharmacother. *7*, 177–193. https://doi.org/10.1517/14656566.7.2.177.

64. Zhang, Y., Zhou, F., Zhang, J., Zou, Q., Fan, Q., and Zhang, F. (2020). Erlotinib enhanced chemoradiotherapy sensitivity via inhibiting DNA damage repair in nasopharyngeal carcinoma CNE2 cells. Ann. Palliat. Med. *9*, 2559–2567. https://doi.org/10.21037/apm-19-466.

65. Watanabe, N., Broome, M., and Hunter, T. (1995). Regulation of the human WEE1Hu CDK tyrosine 15-kinase during the cell cycle. EMBO J. *14*, 1878–1891. https://doi.org/10.1002/j.1460-2075.1995.tb07180.x.

66. Niu, H., Manfredi, M., and Ecsedy, J.A. (2015). Scientific rationale supporting the clinical development strategy for the investigational Aurora A kinase inhibitor alisertib in cancer. Front. Oncol. *5*, 189. https://doi.org/10.3389/fonc.2015.00189.

67. McLaughlin, R.P., He, J., van der Noord, V.E., Redel, J., Foekens, J.A., Martens, J.W.M., Smid, M., Zhang, Y., and van de Water, B. (2019). A kinase inhibitor screen identifies a dual cdc7/CDK9 inhibitor to sensitise triple-negative breast cancer to EGFR-targeted therapy. Breast Cancer Res. *21*, 77. https://doi.org/10.1186/s13058-019-1161-9.

68. Malyutina, A., Majumder, M.M., Wang, W., Pessia, A., Heckman, C.A., and Tang, J. (2019). Drug combination sensitivity scoring facilitates the discovery of synergistic and efficacious drug combinations in cancer. PLoS Comput. Biol. *15*, e1006752. https://doi.org/10.1371/journal.pcbi.1006752.

69. Taheri, S.M., and Hesamian, G. (2013). A generalization of the Wilcoxon signed-rank test and its applications. Stat. Papers *54*, 457–470. https://doi.org/10.1007/s00362-012-0443-4.

70. Song, Z., Zhang, Y., Chen, Z., and Zhang, B. (2021). Identification of key genes in lung adenocarcinoma based on a competing endogenous RNA network. Oncol. Lett. *21*, 60. https://doi.org/10.3892/ol.2020.12322.

71. Jovanović, M., Stanković, T., Stojković Burić, S., Banković, J., Dinić, J., Ljujić, M., Pešić, M., and Dragoj, M. (2022). Decreased TSPAN14 expression contributes to NSCLC progression. Life *12*, 1291. https://doi.org/10.3390/life12091291.

72. Wang, X., Liu, Y., Ding, Y., and Feng, G. (2022). CAMSAP2 promotes colorectal cancer cell migration and invasion through activation of JNK/c-Jun/MMP-1 signaling pathway. Sci. Rep. *12*, 16899. https://doi.org/10.1038/s41598-022-21345-7.

73. Shi, J., Bao, M., Wang, W., Wu, X., Li, Y., Zhao, C., and Liu, W. (2021). Integrated profiling identifies PLOD3 as a potential prognostic and immunotherapy relevant biomarker in colorectal cancer. Front. Immunol. *12*, 722807. https://doi.org/10.3389/fimmu.2021.722807.

74. Zheng, Z., Zhang, X., Bai, J., Long, L., Liu, D., and Zhou, Y. (2022). PGM1 suppresses colorectal cancer cell migration and invasion by regulating the PI3K/AKT pathway. Cancer Cell Int. *22*, 201. https://doi.org/10.1186/s12935-022-02545-7.

75. Vafeiadou, V., Hany, D., and Picard, D. (2022). Hyperactivation of MAPK induces tamoxifen resistance in SPRED2-deficient ERalpha-positive breast cancer. Cancers *14*, 954. https://doi.org/10.3390/cancers14040954.

76. Li, R., Quan, Y., and Xia, W. (2018). SIRT3 inhibits prostate cancer metastasis through regulation of FOXO3A by suppressing Wnt/beta-catenin pathway. Exp. Cell Res. *364*, 143–151. https://doi.org/10.1016/j.yexcr.2018.01.036.

77. Sawant Dessai, A., Dominguez, M.P., Chen, U.I., Hasper, J., Prechtl, C., Yu, C., Katsuta, E., Dai, T., Zhu, B., Jung, S.Y., et al. (2021). Transcriptional repression of SIRT3 potentiates mitochondrial aconitase activation to drive aggressive prostate cancer to the bone. Cancer Res. *81*, 50–63. https://doi.org/10.1158/0008-5472.CAN-20-1708.

78. Wei, Y., and Xu, X. (2016). UFMylation: a unique & fashionable modification for life. Dev. Reprod. Biol. *14*, 140–146. https://doi.org/10.1016/j.gpb.2016.04.001.

79. Pham, T.H., Qiu, Y., Zeng, J., Xie, L., and Zhang, P. (2021). A deep learning framework for high-throughput mechanism-driven phenotype compound screening and its application to COVID-19 drug repurposing. Nat. Mach. Intell. *3*, 247–257. https://doi.org/10.1038/s42256-020-00285-9.

80. Zhu, J., Wang, J., Wang, X., Gao, M., Guo, B., Gao, M., Liu, J., Yu, Y., Wang, L., Kong, W., et al. (2021). Prediction of drug efficacy from transcriptional profiles with deep learning. Nat. Biotechnol. *39*, 1444–1452. https://doi.org/10.1038/s41587-021-00946-z.

81. Stathias, V., Turner, J., Koleti, A., Vidovic, D., Cooper, D., Fazel-Najafabadi, M., Pilarczyk, M., Terryn, R., Chung, C., Umeano, A., et al. (2020). LINCS Data Portal 2.0: next generation access point for perturbation-response signatures. Nucleic Acids Res. *48*, D431–D439. https://doi.org/10.1093/nar/gkz1023.

82. Subramanian, A., Narayan, R., Corsello, S.M., Peck, D.D., Natoli, T.E., Lu, X., Gould, J., Davis, J.F., Tubelli, A.A., Asiedu, J.K., et al. (2017). A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. Cell *171*, 1437–1452.e17. https://doi.org/10.1016/j.cell.2017.10.049.

83. Wang, Y., Xiao, J., Suzek, T.O., Zhang, J., Wang, J., and Bryant, S.H. (2009). PubChem: a public information system for analyzing bioactivities of small molecules. Nucleic Acids Res. *37*, W623–W633. https://doi.org/10.1093/nar/gkp456.

84. Cao, Y., Charisi, A., Cheng, L.C., Jiang, T., and Girke, T. (2008). ChemmineR: a compound mining framework for R. Bioinformatics *24*, 1733–1734. https://doi.org/10.1093/bioinformatics/btn307.

85. O'Boyle, N.M., Morley, C., and Hutchison, G.R. (2008). Pybel: a Python wrapper for the OpenBabel cheminformatics toolkit. Chem. Cent. J. *2*, 5. https://doi.org/10.1186/1752-153x-2-5.

86. Branco, P., Torgo, L., and Ribeiro, R.P. (2016). A survey of predictive modeling on imbalanced domains. ACM Comput. Surv. *49*, 1–50. https://doi.org/10.1145/2907070.

87. Davis, J., and Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd international conference on Machine learning, pp. 233–240. https://doi.org/10.1145/1143844.1143874.

88. Al Iqbal, M.R., Rahman, S., Nabil, S.I., and Chowdhury, I.U.A. (2012). Knowledge based decision tree construction with feature importance domain knowledge. In 2012 7th International Conference on Electrical and Computer Engineering, pp. 659–662. https://doi.org/10.1109/ICECE.2012.6471636.

89. Yuan, Y., Wu, L., and Zhang, X. (2021). Gini-Impurity index analysis. IEEE Trans. Inf. Forensics Secur. *16*, 3154–3169. https://doi.org/10.1109/TIFS.2021.3076932.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Chemicals, peptides, and recombinant proteins		
Erlotinib Hydrochloride EGFR inhibitor	MedChemExpress(MCE)	HY-12008; CAS: 183319-69-9
Gefitinib (ZD1839) EGFR inhibitor	MedChemExpress(MCE)	HY-50895; CAS: 184475-35-2
AZD1775 Wee1 inhibitor	MedChemExpress(MCE)	HY-10993; CAS: 955,365-80-7
MK-5180 Aurora A inhibitor	MedChemExpress(MCE)	HY-13252; CAS: 1010085-13-8
Etoposide Topoisomerase-II inhibitor	MedChemExpress(MCE)	HY-13629; CAS: 33419-42-0
Dinaciclib CDK inhibitor	MedChemExpress(MCE)	HY-10492; CAS: 779353-01-4
Critical commercial assays		
Counting Kit-8 (CCK8) assay	Sigma-Aldrich	M5655
Deposited data		
DrugComb	Zagidullin et al.[41]	https://drugcomb.fimm.fi/
DrugCombDB	Liu et al.[42]	http://drugcombdb.denglab.org/main
AstraZeneca-Sanger Drug Combination Prediction	Menden et al.[43]	https://www.synapse.org/#!Synapse:syn4231880/wiki/235645
LINCS L1000	Stathias et al. [81]	https://lincsproject.org/LINCS/tools/workflows/find-the-best-place-to-obtain-the-lincs-l1000-data
Experimental models: Cell lines		
Human:HT29	ATCC	ATCC HTB-38
Software and algorithms		
ForSyn	This paper	https://doi.org/10.5281/zenodo.7562405
Original Deep Forest	Zhou et al.[31]	https://github.com/kingfengji/gcForest
DCE-DForest	Zhang et al.[37]	N/A
DeepSynergy	Preuer et al.[18]	www.bioinf.jku.at/software/DeepSynergy
MatchMaker	Kuru et al.[19]	https://github.com/tastanlab/matchmaker
TranSynergy	Liu et al.[22]	https://github.com/qiaoliuhub/drug_combination
SynPathy	Tang et al.[52]	https://github.com/TangYiChing/SynPathy
DeepDDS	Wang et al.[23]	https://github.com/Sinwang404/DeepDDS/tree/master
ChemmineR	Cao et al.[82]	https://www.bioconductor.org/packages/release/bioc/html/ChemmineR.html
OpenBabel	O'Boyle et al.[83]	https://openbabel.org/wiki/Category:Installation

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Xiaochen Bo (boxc@bmi.ac.cn, boxiaoc@163.com).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- The data reported in this paper is publicly available on GitHub (https://github.com/Lianlian-Wu/ForSyn) and Zenodo (https://doi.org/10.5281/zenodo.7562405).
- The original code is publicly available on GitHub (https://github.com/Lianlian-Wu/ForSyn) and Zenodo (https://doi.org/10.5281/zenodo.7562405).
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

EXPERIMENTAL MODEL AND SUBJECT DETAILS

The HT-29 cell line purchased from the American Type Culture Collection (ATCC, Manassas, VA, USA) are used for drug combination screening. The cell line is verified by Short Tandem Repeat (STR) profiling performed by Cell Line Authentication Service at ATCC. HT29 cells are cultured in McCoy's 5A (Hyclone, Logan, UT, USA) containing 10% fetal bovine serum (FBS) (Hyclone, USA), 1% penicillin (Invitrogen, USA), and 1% streptomycin (Invitrogen, USA). The cell line is maintained at 37°C and 5% CO2 in a humidified incubator.

METHOD DETAILS

Drug combination dataset

In the dataset, a sample represent a drug combination on a particular cancer cell line, i.e., a drug combination-cell line pair. The samples are collected from DrugComb, DrugCombDB and AstraZeneca-Sanger Drug Combination Prediction databases. To ensure that each sample contains all the four feature types, the samples with missing features are removed. For the samples collected from DrugComb and DrugCombDB, four kinds of synergy scores (Loewe, Bliss, HSA and ZIP) are used to quantify the effects of samples. The synergy scores in replicate experiments are averaged to get a unique value for each sample. Finally, 3,192 samples are obtained, covering 77 drugs and 15 cancer cell lines.

The samples are classified according to the scheme proposed by Malyutina et al.[44] due to the noise data exists in the datasets. The 200 samples with above four synergy scores greater than five are regarded as the synergism class (minority class), the remaining 2,992 samples are classified as non-synergism class (majority class). The imbalance rate is close to 15, which is defined as the ratio between the size of the majority class and that of the minority class.

Feature set

To construct the input dataset, the representation of each sample is the concatenation of pairwise drug feature and the cell line feature. The feature vector of a single drug is composed of the 978-dimensional DGE, 881-dimensional DMF, and 55-dimensional DPP. The 978-dimensional CGE is used to represent cancer cell lines. It should be noted that the DGE and CGE features are cell line-specific, each dataset should include DGE or CGE to distinguish samples on different cell lines. To further investigate the influence of different representations in the classification process, eight different datasets including different feature combinations are generated (Table 1).

Drug-induced gene expression profiles

The 978-dimensional DGE of 15 cell lines are collected from LINCS L1000 database (Level 5).[79,82] The expression profiles of the same drug at different doses and different time points are fused by weighted average method.[82] The 978 landmark genes and their *Z* values in the expression profiles are selected as the gene expression values under drug perturbation. Each gene expression value represents the relative value of gene expression level. A positive value indicates that gene expression is up-regulated and a negative value indicates that expression is down-regulated.

Chemical structure of drugs

The 881-dimensional PubChem[83] DMF calculated by R-package ChemmineR are used to represent the chemical structure of drugs.[84] In the binary vectors, each bit represents whether the specific substructure exists.

Physicochemical properties of drugs

The 55-dimensional DPP is calculated by ChemmineR, OpenBabel[85] and JoeLib packages. Physicochemical properties include molecular weight, solubility and hydrophobic parameters.

Gene expression profiles of untreated cell lines

The 978-dimensional CGE are also collected from the LINCS L1000 database. The transcriptional expression levels of 15 cell lines treated with dimethyl sulfoxide as solvent control are used as the feature of cell lines.

Description of ForSyn

Random Forest dealing with imbalanced data

RF performs bootstrap sampling and random feature selection in the induction process of the base classifier. However, the RF cannot effectively handle imbalanced data. To deal with the problem of imbalanced data, a stratified under-sampling based on AP clustering is designed to rebalance the training set and minimize the information loss caused by random sampling. The proposed under-sampling method is combined with the standard RF framework to rebalance the training set of each decision tree. Data S1 and Figure S2A show the training process of the proposed RF.

Figure S2B shows a diagram of stratified under-sampling based on clustering. The purpose of clustering is to make the samples with high similarity as concentrated as possible. The stratified sampling is to obtain samples from each cluster in a balanced manner. This strategy can preserve the useful information of the majority class as much as possible. Traditional clustering methods, such as K-means, need to preset the number of clusters, and randomly select the initial cluster centers in the training data. The performance of the K-means algorithm is sensitive to the initial cluster centers. It often takes multiple runs to find relatively good initial cluster centers. While AP clustering does not need to preset the number of clusters. Each sample is regarded as a node in a network. The pairwise nodes perform information exchange recursively along the edge until a set of good cluster centers and corresponding clusters appear. Compared with traditional clustering methods, AP clustering always perform better and takes less time. Next, the AP clustering results are stratified sampled to obtain more representative samples of majority class. After resampling, the number of samples of the majority class is equal to that of the minority class.

Extreme tree forest dealing with high feature dimension

Different from RF, the ETF uses all the features as candidates, and then randomly selects a feature as the split node of the tree. According to the properties, the ETF is less affected by imbalanced data compared with RF. However, the high feature dimension and random selection of features would deepen the depth of the tree and lead to over-fitting. To overcome the problem of ETF, we propose a greedy dimension reduction method and combines a data complexity metric. The data complexity metric is defined as the tail overlap of the conditional distribution between two classes.[50] For each feature in the dataset, find the maximum and minimum values of the feature in different classes, and calculate the overlap area between them (Equation 1):

$$F_i = \frac{MIN\ (max(f_i, c_1), max(f_i, c_2)) - MAX\ (min(f_i, c_1), min(f_i, c_2))}{MAX\ (max(f_i, c_1), max(f_i, c_2)) - MIN\ (min(f_i, c_1), min(f_i, c_2))} \qquad \text{(Equation 1)}$$

where $max(f_i, c_j)$ and $min(f_i, c_j)$ refer to the maximum and minimum values of the i-th feature in the j-th class, $i = 1 \ldots d$, d corresponds to the feature dimension. $j \in [1,2]$ corresponds to two classes. The value range of the overlap area is [0,1]. The smaller the value indicates the smaller the overlap area of the feature between different classes. The smaller the overlapping area, the greater the contribution of the feature to the classification result.

Data S2 performs dimension reduction on the input data. First, the data complexity metric is used to evaluate the classification contribution (overlap area) of each feature. Then combines the greedy algorithm to approximate the local optimal feature subspace. Finally, the selected features are used to train the ETF. Suppose the dataset D is divided into training set T and validation set V ($D = T \cup V$). T is used to train the ETF, and V is used to evaluate the performance of the model. The algorithm process of Data S2 can be described as follows.

Step 1. Calculate F_i of each feature, and then sort the feature space by ascending order of F_i (Algorithm process 1–2 in Data S2).
Step 2. The dataset D after the feature reordering is horizontally divided into R subsets (Algorithm process 3–4 in Data S2).
Step 3. R-1 subsets are used as the training set and one subset as the verification set, to train and evaluate the performance of the ETF (Algorithm process 5–8 in Data S2).
Step 4. The feature dimension is iteratively reduced, and the performance of trained ETF is checked. If the performance drops, stop iterative training and return to the dimension of the current feature subset (Algorithm process 9–18 in Data S2).
Step 5. Repeat step 3 to step 4, until R-fold cross-verification training is implemented. The average dimension of R retained feature subspaces is obtained, and then the local optimal feature subspace is returned to train an ETF (Algorithm process 19–23 in Data S2).

Framework of the ForSyn

As shown in Figure 1, RF-CUS and ETF-DR are embedded as the unit in the cascade structure of the ForSyn. In the ForSyn, l represents the index of each layer, $l = 1 \ldots L$. Each layer has T units, t represents the index of the unit in each cascade layer, $t = 1 \ldots T$. The $p_{i,j}^{t,l}$ refers to the probability that the forest in t-th unit of the l-th layer predicts sample x_i belongs to the j-th class. In the binary classification of imbalanced data, the majority class and the minority class are represented by -1 and 1, respectively, $j \in [-1,1]$, then:

$$p_{i,-1}^{t,l} + p_{i,1}^{t,l} = 1 \qquad \text{(Equation 2)}$$

When predicting an unknown sample, each unit generates a class probability vector $[p_{i,-1}^{t,l}, p_{i,1}^{t,l}]$. In one layer, the output class probability vectors of the units are concatenated with the raw feature vector as the input of the next cascade layer. Thus, the classification result of the previous layer can guide the classification process of the next layer. Assuming C_l represents the l-th layer, F_l is the cascade of the first l layers, the relationship between the layer and the cascade can be expressed as Equation 3:

$$F_l(x_i) = \begin{cases} C_l(x_i) & l = 1 \\ C_l\left(\left[x_i, p_{i,j}^{t,l-1}, t = 1 \ldots T\right]\right) & l > 1 \end{cases} \qquad \text{(Eqaution 3)}$$

where $[x_i, p_{i,j}^{t,l-1}, t = 1 \ldots T]$ represents the concatenation of the raw feature vector and the output of the previous layer. Through the cascade structure, the layer-by-layer processing and in-model feature transformation are implemented for the raw data. For the unknown sample x_i, the output of the last cascade layer (C_L) is formulated as Equation 4:

$$C_L(x_i) = argmax_{j \in [-1,1]} \frac{1}{T} \sum_{t=1}^{T} p_{i,j}^{t,L} \qquad \text{(Equation 4)}$$

In addition, the growth of the cascade layer adopts a greedy convergence mode during the training process. The number of cascade layers is adaptively determined under sufficient training.

Comparison methods

We contrast the ForSyn with 12 advanced algorithms, including eight deep learning-based state-of-the-art methods and four advanced machine learning methods. Among the deep learning-based methods, DeepSynergy, MatchMaker, TranSynergy, SynPathy are DNN-based methods for drug combination prediction. DeepSynergy is a feedforward neural network with two hidden layers, which is proposed by Preuer et al. The author claims that it is the first application of deep learning technology in drug combination prediction. MatchMaker applies three subnetworks. Two subnetworks are trained in parallel to learn the representation of drugs on a particular cell line, and then the joint representation for the combination is fed into the third network to predict the synergy score. TranSynergy is a transformer-based deep learning model with dimension reduction component. A transformer is applied to learn the representation of samples and a fully connected layer is used to predict. SynPathy is a DNN-based model with three fully connected layers. The DCE-DForest and original DF are DF-based methods. DCE-DForest is a DF-based model with four original RFs in each layer. The original DF put two RF and two ETF units in each layer. DeepDDS-GCN and DeepDDS-GAT are GNN-based methods proposed by Wang et al. It should be noted that the datasets applied by the two DeepDDS methods are different from other comparison methods. Other methods support tabular input formats, while only graph-structured data can be accepted as input of DeepDDS. To construct the dataset of graph format, the SMILES of each drug are obtained and converted to a graph using RDKit followed by the article of DeepDDS. In the graph of drug, the vertices are atoms and the edges are chemical bonds. The CGE data is used to represent the cell lines. DeepDDS applies two types of GNN, GCN and GAT, to extract the representations of drugs. Then the representations of drugs and cell lines are concatenated and input into the fully connected layers to achieve predict task.

Among the machine learning algorithms, XGBoost and RF are two advanced ensemble learning algorithms, which are representative of sequential ensemble and parallel ensemble respectively. RUSBoost and Balanced Bagging are the classical methods in imbalanced learning. Both are the combination of ensemble learning and data resampling technology. RUSBoost balances the training set by randomly under-sampling majority class, and sequentially trains decision trees. Balanced Bagging uses similar under-sampling techniques to train all decision trees in parallel. All the algorithms are implemented according to the corresponding literature or the default configuration.

Evaluation metrics

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{TN} + \text{FP}} \quad \text{(Equation 5)}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{(Equation 6)}$$

$$\text{Recall} = \text{TP rate} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad \text{(Equation 7)}$$

$$\text{F} - \text{value} = \frac{(1 + \beta^2) * \text{Recall} * \text{Precision}}{\beta^2 * (\text{Recall} + \text{Precision})} \quad \text{(Equation 8)}$$

$$\text{TN rate} = \frac{\text{TN}}{\text{TN} + \text{FP}} \quad \text{(Equation 9)}$$

$$G - Mean = \sqrt{TP\ rate * TN\ rate} \quad \text{(Equation 10)}$$

$$MCC = \frac{TP * TN - FP * FN}{(\text{TP} + \text{FN})(\text{TP} + \text{FP})(\text{TN} + \text{FP})(\text{TN} + \text{FN})} \quad \text{(Equation 11)}$$

There are more than 40 evaluation metrics used to deal with the problem of imbalanced data, which are introduced in a comprehensive review proposed by Branco et al.[86] In this study, five metrics are used to evaluate the performance of algorithms. The metrics including F1-score, AUPR, Recall, MCC and G-Mean. The formulas are shown in Equations 5, 6, 7, 8, 9, 10, and 11.

Accuracy (Equation 5), as a global metric, is often used to evaluate the classification performance of models. But it is not suitable for imbalanced learning. It will induce the classification result biased toward the majority class. Suppose that the proportions of the majority class and the minority class in a dataset are 95% and 5%, respectively. In this case, a classifier can obtain high Accuracy by marking all test samples as the majority class. However, the purpose of imbalanced learning is to identify unknown samples in minority class. The Accuracy cannot effectively reflect the predictive performance of the classifier on minority class samples.

Precision takes FP (False Positives) into account and is more sensitive to imbalanced data (Equation 6). But using Precision alone cannot accurately express the classification status because it ignores FN (False Negatives). The Recall is not affected by imbalanced data, because it only relies on the minority classes, but it ignores FP (Equation 7). Precision and Recall often conflict with each other. To solve this problem, F-value is defined as a trade-off between them, where β is used to adjust the relative importance between Precision and Recall (Equation 8). In the current learning, β is set to 1.

G-Mean calculates the square root of the product of TP (True Positives) rate and TN (True Negatives) rate, and pays attention to the classification result of the majority class and the minority class simultaneously (Equation 10). MCC can be regarded as a balanced metric (Equation 11). It is a correlation coefficient value between -1 and 1. 1 represents the perfect performance of the classifier, and 0 represents the performance of average random prediction, -1 means the opposite performance.

In previous studies, the authors also used the ROC curve, which plots the relationship between the true positive rate and the false positive rate. It creates a visual model that describes the trade-off between correctly classified positive samples and incorrectly classified negative samples. However, when comparing the ROC curves of multiple algorithms on the plane, it is difficult to intuitively select the best algorithm because of the overlap between the curves, unless one curve can completely outperform all other curves in the overall space. The area under the ROC curve provides a single value to measure the average performance of each algorithm. According to Davis's research, the ROC curve expresses overly optimistic results on a highly imbalanced dataset, and the PR curve should be used instead.[87] Similar to AUC, AUPR represents the area under the PR curve.

Although Recall ignores FP, it intuitively reflects the performance of the classifier on the minority class. The remaining four metrics pay attention to the classification results of the minority class and the majority class synchronously.

Cross validations

In addition to the random 5-folds CV, three CV strategies are also performed to evaluate the generalization performance on novel unseen cell lines, drugs, and drug combinations. The training and test sets are shuffled by cell lines, drugs, or drug combinations, which are described as leave-cell-line-out CV, leave-drug-out CV and leave-drug-combination-out CV, as shown in Figure S2C. There are 15 cell lines, 77 drugs and 884 unique drug combinations (without considering cell lines) in this dataset. For leave-cell-line-out CV, the drug combinations in each cell line is regarded as the test set in turn and in the other 14 cell lines are the training set. Then, a 15-fold CV is performed. For leave-drug- out CV, we divide the 77 drugs into n = 7 groups. The drug combination samples involved in a drug group are combined as the test set, and those in the other n-1 groups are the training set. Consequently, all drug groups are regarded as test sets in turn. For the leave-drug-combination-out CV, 884 unique drug combinations (without considering cell lines) are extracted to split the training and testing data. The 884 unique drug combinations are divided into n = 8 groups. The samples (considering cell lines) involved in one drug combination group is regarded as the test set and in the other n-1 groups are the training set. Then, an 8-fold CV is performed.

Evaluation of feature importance value

ForSyn is a decision tree-based deep learning method. It contains multiple cascade layers, each of which is composed of multiple decision tree-based forests. Thus, each layer is the integration of decision trees. The decision tree is an excellent interpretable model, which quantify the global relationship between each feature and the output by evaluating the FIV. That is, the FIV can be used to represent the amount of influence a feature has over the classification process.[88]

Then, ForSyn can evaluate the FIV of each feature by extracting and aggregating the FIVs of all decision trees. In each decision tree, the FIV can be evaluated by some classic data-dependent metrics, such as information gain or gini impurity.[89] Suppose a trained ForSyn has L layers, each layer contains J forests, and each forest contains T trees. Then the FIV of the d-th feature in the t-th tree of the j-th forest in the l-th layer is represented by $I_{j,t}^{d,l}$, where $d = 1 \ldots D$, $l = 1 \ldots L$, $j = 1 \ldots J$, and $t = 1 \ldots T$. C_l^d represents the FIV of the d-th feature on the l-th layer, and is calculated by Equation 12. C_d represents the FIV of the d-th feature in all layers (Equation 13). It can be regarded as the contribution of the d-th feature in the overall model.

$$C_l^d = \sum\nolimits_{j=1}^{J} \sum\nolimits_{t=1}^{T} I_{j,t}^{d,l} \quad \text{(Equation 12)}$$

$$C_d = \frac{\sum\nolimits_{l=1}^{L} C_l^d}{\sum\nolimits_{d=1}^{D} \sum\nolimits_{l=1}^{L} C_l^d} \quad \text{(Equation 13)}$$

Drug combination screening

The Cell Counting Kit-8 (CCK8) assay (Sigma, M5655) is performed to evaluate the synergy potential in HT29 cell line. The experimental design is adopted from Chou et al.,[4] where 10 concentration values of drugs are set. Cells are seeded in 96-well plates

(Nest, 701,001) at 4000 cells/well densities and cultured at the indicated concentrations with the solvent DMSO or drug combinations for 72 h. After 72 h of incubation, 10μL per well of CCK8 reagent is added to the plates, followed by 4 h of incubation. The luminesce of optical density value (OD) is measured using Envision plate reader (PerkinElmer) at 450 nm wavelength. The cellular proliferation inhibition rate is defined as Equation 14. Then the CI value of each drug combination is calculated by CompuSyn software[4] to quantify the synergistic potential.

$$\mathit{inhibition}\ (\%) = (\mathit{ODsample} - \mathit{ODNegative})/(\mathit{ODPositive} - \mathit{ODNegative}) \times 100\% \tag{Equation 14}$$

QUANTIFICATION AND STATISTICAL ANALYSIS

The Friedman test and Nemenyi test are used to analyze the performance difference among different algorithms. The Friedman test compares the performance differences of multiple algorithms on multiple datasets. Under the null hypothesis, it believes that the performances of all algorithms are equal. Equation 15 and Equation 16 show the original Friedman statistics and an improved version respectively.

$$\mathrm{X}_{\mathrm{F}}^2 = \frac{12\mathrm{N}}{\mathrm{K}(\mathrm{K}+1)}\left[\sum_{\mathrm{j}}\mathrm{R}_{\mathrm{j}}^2 - \frac{\mathrm{K}(\mathrm{K}+1)^2}{4}\right] \tag{Equation 15}$$

$$F_F = \frac{(N-1)X_F^2}{N(K-1) - X_F^2} \tag{Equation 16}$$

Then, the Nemenyi test is performed to verify the performance difference between pairwise algorithms. The performances of two classifiers are significantly different if the corresponding average ranks differ by at least the critical difference (CD):

$$\mathrm{CD} = q_\alpha\sqrt{\frac{K(K+1)}{6N}} \tag{Equation 17}$$

The statistical details and exact values of experiments can be found in RESULTS-Performance evaluation section.

Cell Reports Methods, Volume *3*

Supplemental information

A hybrid deep forest-based method

for predicting synergistic drug combinations

Lianlian Wu, Jie Gao, Yixin Zhang, Binsheng Sui, Yuqi Wen, Qingqiang Wu, Kunhong Liu, Song He, and Xiaochen Bo

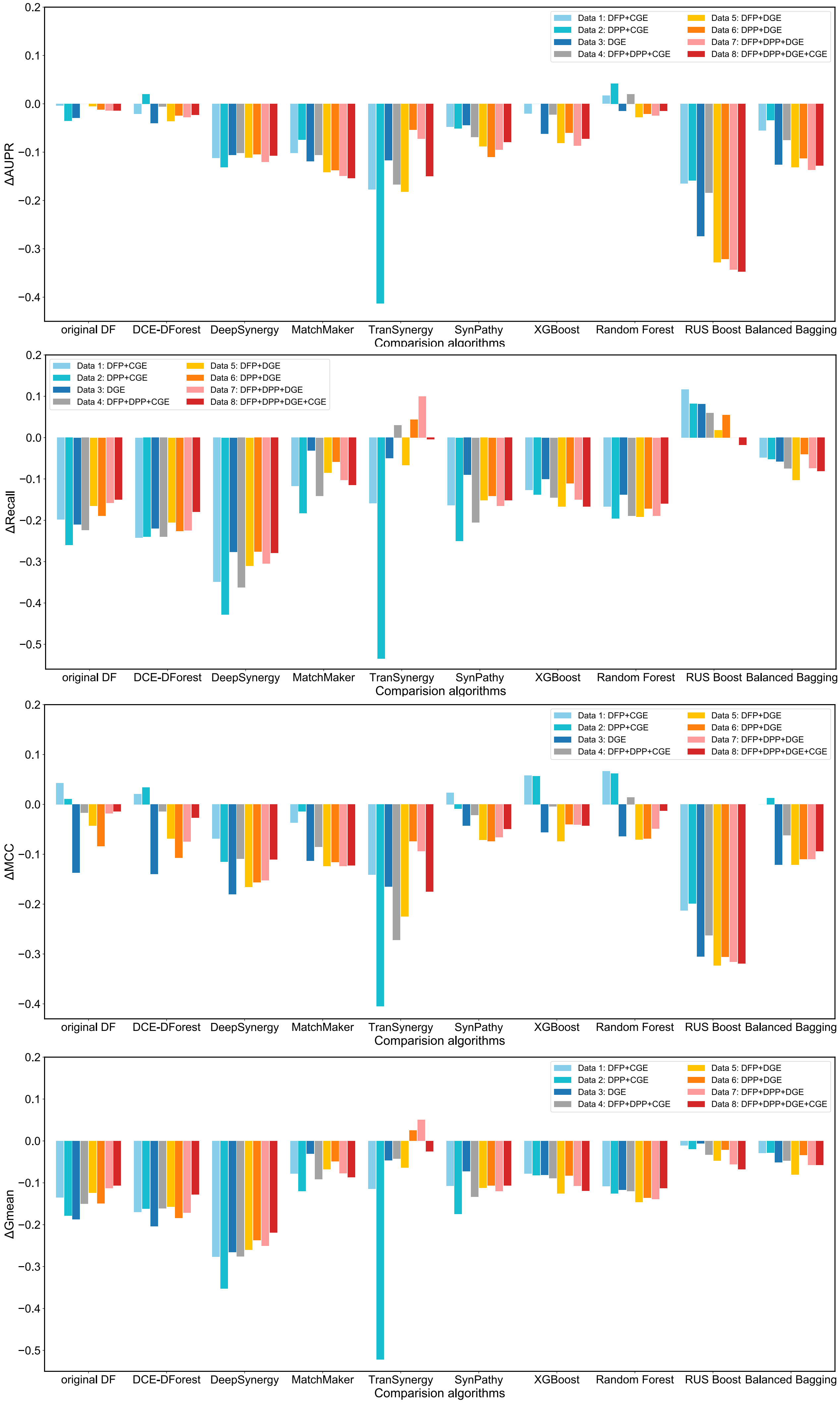


Figure S1. The performance of all algorithms under different datasets, related to Figure 2.

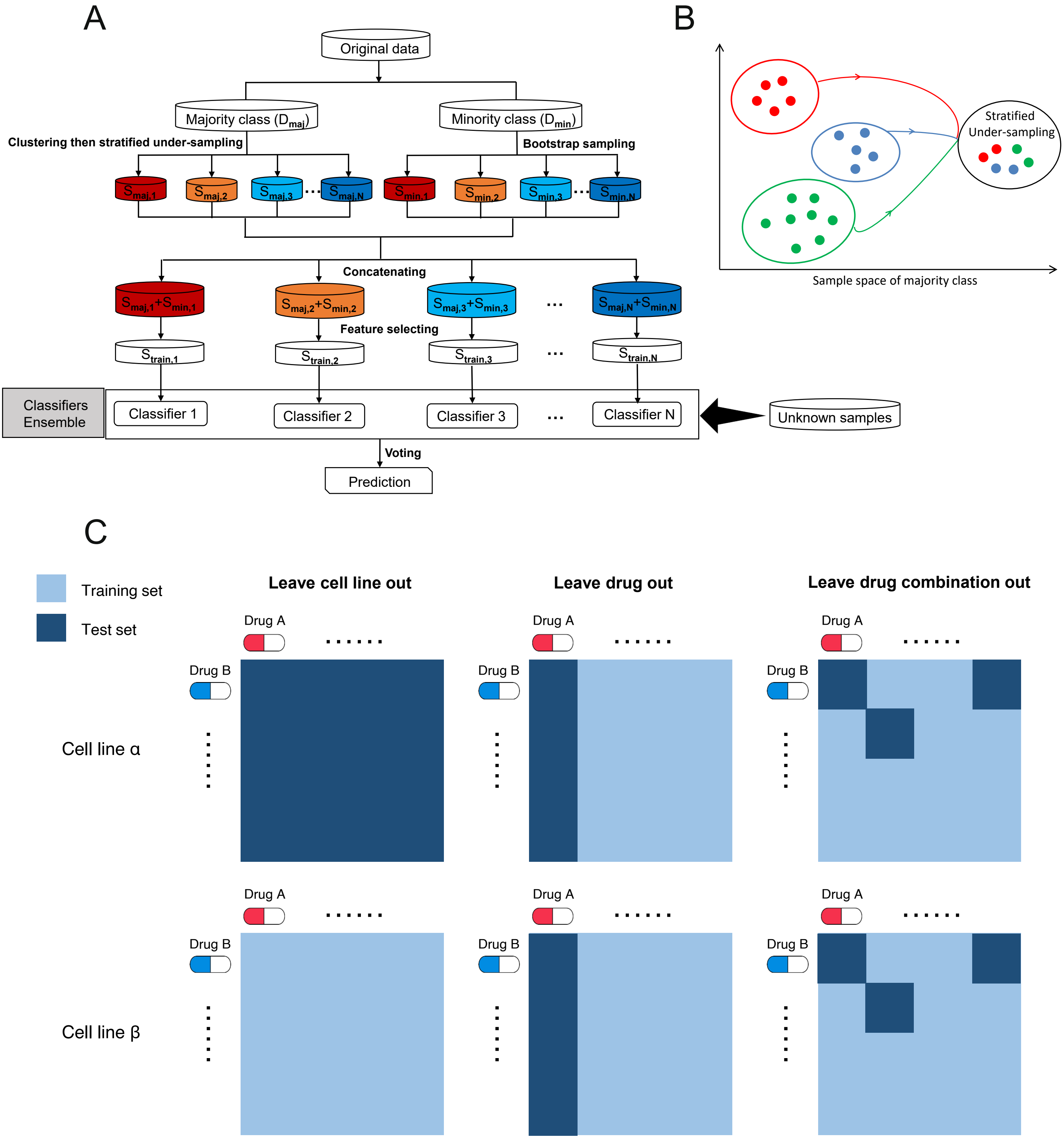


Figure S2. The diagram of ForSyn's random forest unit and three cross-validation strategies, related to STAR Methods.

(A) The framework of random forest unit based on clustering and stratified under-sampling.

(B) The diagram of stratified under-sampling based on clustering.

(C) The diagram of three cross-validation strategies, i.e., leave-cell-line-out, leave-drug-out and leave-drug-combination-out cross-validation.

Table S1. Performance comparison of all algorithms based on F1-score, related to Figure 2.

	ForSyn	Original DF	DCE-DForest	Deep Synergy	Match Maker	Tran Synergy	SynPathy	XGBoost	Random Forest	RUS Boost	Balanced Bagging	DeepDDS-GCN	DeepDDS-GAT
Data 1	0.499(3.5)	0.487 (6.0)	0.451 (8.0)	0.330 (10.0)	0.459 (7.0)	0.367 (9.0)	0.491 (5.0)	0.528 (1.0)	0.518 (2.0)	0.269 (11.0)	0.499 (3.5)	/	/
Data 2	0.496 (4.0)	0.451 (7.0)	0.473 (5.0)	0.262 (10.0)	0.466 (6.0)	0.076 (11.0)	0.445 (8.0)	0.533 (1.0)	0.514 (2.0)	0.282 (9.0)	0.510 (3.0)	/	/
Data 3	0.519 (1.0)	0.328 (8.0)	0.315 (9.0)	0.248 (10.0)	0.427 (4.0)	0.384 (7.0)	0.451 (2.0)	0.442 (3.0)	0.411 (6.0)	0.247 (11.0)	0.415 (5.0)	/	/
Data 4	0.529 (2.5)	0.493 (6.0)	0.486 (7.0)	0.357 (9.0)	0.470 (8.0)	0.278 (11.0)	0.500 (4.0)	0.534 (1.0)	0.529 (2.5)	0.283 (10.0)	0.497 (5.0)	/	/
Data 5	0.568 (1.0)	0.483 (2.0)	0.445 (8.0)	0.324 (10.0)	0.455 (5.5)	0.367 (9.0)	0.475 (3.0)	0.466 (4.0)	0.452 (7.0)	0.263 (11.0)	0.455 (5.5)	/	/
Data 6	0.551 (1.0)	0.419 (8.0)	0.381 (9.0)	0.320 (10.0)	0.447 (6.0)	0.494 (2.0)	0.452 (5.0)	0.485 (3.0)	0.436 (7.0)	0.264 (11.0)	0.456 (4.0)	/	/
Data 7	0.564 (1.0)	0.502 (2.0)	0.431 (9.0)	0.338 (10.0)	0.447 (8.0)	0.476 (4.0)	0.470 (5.0)	0.493 (3.0)	0.467 (6.0)	0.264 (11.0)	0.463 (7.0)	/	/
Data 8	0.572 (1.0)	0.509 (2.0)	0.488 (5.0)	0.383 (10.0)	0.444 (8.0)	0.395 (9.0)	0.490 (4.0)	0.486 (6.0)	0.506 (3.0)	0.257 (11.0)	0.472 (7.0)	/	/
Graph Data	/	/	/	/	/	/	/	/	/	/	/	0.474	0.426
Average Rank	(1.9)	(5.1)	(7.5)	(9.9)	(6.6)	(7.8)	(4.5)	(2.8)	(4.4)	(10.6)	(5.0)	/	/

Note: The performance of the first 11 algorithms is evaluated on Data 1-8 in tabular formats. Deep-GCN and Deep-GAT are the graph neural network-based algorithms and are tested on an additional dataset, Graph Data. The value in parentheses represents the ranking value of the corresponding performance. Taking Data 8 as an example, the ForSyn on this dataset has the best performance (0.572) and is assigned a ranking value of 1.0; the performance of Random Forest is the second (0.521) and is assigned a ranking value of 2.0; the performance of cascade forest and XGBoost are the same (0.515), they occupy the third and fourth positions respectively, so their ranking values are uniformly assigned 3.5 ((3.0+4.0)/2). The average rank of each algorithm is defined as the average of its ranks on all datasets.

Table S2. Performance comparison of all algorithms based on AUPR, related to Figure 2.

	ForSyn	Original DF	DCE-DForest	Deep Synergy	MatchMaker	TranSynergy	SynPathy	XGBoost	Random Forest	RUS Boost	Balanced Bagging	DeepDDS-GCN	DeepDDS-GAT
Data 1	0.566 (2.0)	0.562 (3.0)	0.545 (5.0)	0.454 (9.0)	0.465 (8.0)	0.389 (11.0)	0.518 (6.0)	0.546 (4.0)	0.583 (1.0)	0.401 (10.0)	0.511 (7.0)	/	/
Data 2	0.551 (3.5)	0.516 (6.0)	0.571 (2.0)	0.420 (9.0)	0.477 (8.0)	0.138 (11.0)	0.500 (7.0)	0.551 (3.5)	0.593 (1.0)	0.392 (10.0)	0.517 (5.0)	/	/
Data 3	0.521 (1.0)	0.492 (3.0)	0.481 (4.0)	0.415 (7.0)	0.402 (9.0)	0.404 (8.0)	0.477 (5.0)	0.459 (6.0)	0.506 (2.0)	0.247 (11.0)	0.395 (10.0)	/	/
Data 4	0.581 (2.5)	0.581 (2.5)	0.575 (4.0)	0.479 (8.0)	0.475 (9.0)	0.414 (10.0)	0.512 (6.0)	0.559 (5.0)	0.601 (1.0)	0.397 (11.0)	0.506 (7.0)	/	/
Data 5	0.579 (1.0)	0.574 (2.0)	0.543 (4.0)	0.468 (7.0)	0.438 (9.0)	0.397 (10.0)	0.491 (6.0)	0.498 (5.0)	0.551 (3.0)	0.251 (11.0)	0.448 (8.0)	/	/
Data 6	0.565 (1.0)	0.553 (2.0)	0.541 (4.0)	0.461 (7.0)	0.428 (10.0)	0.511 (5.0)	0.455 (8.0)	0.505 (6.0)	0.544 (3.0)	0.244 (11.0)	0.452 (9.0)	/	/
Data 7	0.589 (1.0)	0.575 (2.0)	0.561 (4.0)	0.469 (8.0)	0.440 (10.0)	0.517 (5.0)	0.494 (7.0)	0.502 (6.0)	0.565 (3.0)	0.246 (11.0)	0.452 (9.0)	/	/
Data 8	0.591 (1.0)	0.577 (2.0)	0.568 (4.0)	0.483 (7.0)	0.437 (10.0)	0.441 (9.0)	0.512 (6.0)	0.519 (5.0)	0.576 (3.0)	0.244 (11.0)	0.463 (8.0)	/	/
Graph Data	/	/	/	/	/	/	/	/	/	/	/	0.508	0.464
Average Rank	(1.6)	(2.8)	(3.9)	(7.8)	(9.1)	(8.6)	(6.4)	(5.1)	(2.1)	(10.8)	(7.9)	/	/

Table S3. Performance comparison of all algorithms based on Recall, related to Figure 2.

	ForSyn	Original DF	DCE-DForest	Deep Synergy	MatchMaker	TranSynergy	SynPathy	XGBoost	Random Forest	RUS Boost	Balanced Bagging	DeepDDS-GCN	DeepDDS-GAT
Data 1	0.559 (2.0)	0.361 (9.0)	0.317 (10.0)	0.210 (11.0)	0.442 (4.0)	0.400 (6.0)	0.395 (7.0)	0.432 (5.0)	0.392 (8.0)	0.676 (1.0)	0.511 (3.0)	/	/
Data 2	0.585 (2.0)	0.325 (9.0)	0.345 (7.0)	0.157 (10.0)	0.402 (5.0)	0.050 (11.0)	0.335 (8.0)	0.447 (4.0)	0.389 (6.0)	0.667 (1.0)	0.533 (3.0)	/	/
Data 3	0.425 (2.0)	0.215 (9.0)	0.205 (10.0)	0.149 (11.0)	0.394 (3.0)	0.375 (4.0)	0.335 (6.0)	0.325 (7.0)	0.287 (8.0)	0.506 (1.0)	0.367 (5.0)	/	/
Data 4	0.595 (3.0)	0.371 (9.0)	0.355 (10.0)	0.233 (11.0)	0.454 (5.0)	0.625 (2.0)	0.390 (8.0)	0.450 (6.0)	0.406 (7.0)	0.655 (1.0)	0.520 (4.0)	/	/
Data 5	0.517 (2.0)	0.352 (7.0)	0.312 (10.0)	0.207 (11.0)	0.432 (4.0)	0.450 (3.0)	0.365 (6.0)	0.350 (8.0)	0.325 (9.0)	0.535 (1.0)	0.414 (5.0)	/	/
Data 6	0.481 (3.0)	0.292 (9.0)	0.255 (10.0)	0.205 (11.0)	0.422 (5.0)	0.525 (2.0)	0.340 (7.0)	0.370 (6.0)	0.309 (8.0)	0.536 (1.0)	0.441 (4.0)	/	/
Data 7	0.525 (2.5)	0.367 (7.0)	0.300 (10.0)	0.220 (11.0)	0.422 (5.0)	0.625 (1.0)	0.360 (8.0)	0.375 (6.0)	0.336 (9.0)	0.525 (2.5)	0.451 (4.0)	/	/
Data 8	0.537 (1.0)	0.387 (6.0)	0.357 (10.0)	0.258 (11.0)	0.422 (5.0)	0.533 (2.0)	0.385 (7.0)	0.370 (9.0)	0.377 (8.0)	0.519 (3.0)	0.456 (4.0)	/	/
Graph Data	/	/	/	/	/	/	/	/	/	/	/	0.442	0.364
Average Rank	(2.2)	(8.1)	(9.6)	(10.9)	(4.5)	(3.9)	(7.1)	(6.4)	(7.9)	(1.4)	(4.0)	/	/

Table S4. Performance comparison of all algorithms based on MCC, related to Figure 2.

	ForSyn	Original DF	DCE-DForest	Deep Synergy	MatchMaker	TranSynergy	SynPathy	XGBoost	Random Forest	RUS Boost	Balanced Bagging	DeepDDS-GCN	DeepDDS-GAT
Data 1	0.464 (6.5)	0.507 (3.0)	0.485 (5.0)	0.395 (9.0)	0.427 (8.0)	0.323 (10.0)	0.488 (4.0)	0.522 (2.0)	0.531 (1.0)	0.251 (11.0)	0.464 (6.5)	/	/
Data 2	0.464 (6.0)	0.475 (5.0)	0.498 (3.0)	0.349 (9.0)	0.449 (8.0)	0.059 (11.0)	0.455 (7.0)	0.521 (2.0)	0.526 (1.0)	0.265 (10.0)	0.477 (4.0)	/	/
Data 3	0.509 (1.0)	0.372 (7.0)	0.369 (8.0)	0.329 (10.0)	0.396 (5.0)	0.344 (9.0)	0.466 (2.0)	0.453 (3.0)	0.445 (4.0)	0.204 (11.0)	0.388 (6.0)	/	/
Data 4	0.525 (2.0)	0.508 (5.0)	0.511 (4.0)	0.415 (9.0)	0.440 (8.0)	0.253 (11.0)	0.503 (6.0)	0.521 (3.0)	0.539 (1.0)	0.262 (10.0)	0.463 (7.0)	/	/
Data 5	0.548 (1.0)	0.505 (2.0)	0.479 (3.0)	0.382 (9.0)	0.424 (8.0)	0.323 (10.0)	0.477 (4.5)	0.474 (6.0)	0.477 (4.5)	0.225 (11.0)	0.427 (7.0)	/	/
Data 6	0.533 (1.0)	0.449 (6.0)	0.426 (7.0)	0.377 (10.0)	0.417 (9.0)	0.459 (4.5)	0.459 (4.5)	0.493 (2.0)	0.464 (3.0)	0.227 (11.0)	0.423 (8.0)	/	/
Data 7	0.541 (1.0)	0.523 (2.0)	0.466 (6.0)	0.389 (10.0)	0.417 (9.0)	0.447 (7.0)	0.475 (5.0)	0.500 (3.0)	0.492 (4.0)	0.225 (11.0)	0.431 (8.0)	/	/
Data 8	0.535 (1.0)	0.521 (3.0)	0.508 (4.0)	0.425 (8.0)	0.413 (9.0)	0.360 (10.0)	0.486 (6.0)	0.492 (5.0)	0.522 (2.0)	0.216 (11.0)	0.441 (7.0)	/	/
Graph Data	/	/	/	/	/	/	/	/	/	/	/	0.450	0.414
Average Rank	(2.4)	(4.1)	(5.0)	(9.3)	(8.0)	(9.1)	(4.9)	(3.3)	(2.6)	(10.8)	(6.7)	/	/

Table S5. Performance comparison of all algorithms based on Gmean, related to Figure 2.

	ForSyn	Original DF	DCE-DForest	Deep Synergy	MatchMaker	TranSynergy	SynPathy	XGBoost	Random Forest	RUS Boost	Balanced Bagging	DeepDDS-GCN	DeepDDS-GAT
Data 1	0.729 (1.0)	0.594 (9.0)	0.559 (10.0)	0.453 (11.0)	0.651 (4.5)	0.615 (8.0)	0.622 (6.0)	0.651 (4.5)	0.621 (7.0)	0.718 (2.0)	0.700 (3.0)	/	/
Data 2	0.743 (1.0)	0.565 (9.0)	0.581 (7.0)	0.391 (10.0)	0.623 (5.0)	0.221 (11.0)	0.568 (8.0)	0.661 (4.0)	0.618 (6.0)	0.724 (2.0)	0.715 (3.0)	/	/
Data 3	0.646 (1.0)	0.459 (9.0)	0.442 (10.0)	0.381 (11.0)	0.616 (3.0)	0.600 (4.0)	0.573 (6.0)	0.565 (7.0)	0.529 (8.0)	0.64 (2.0)	0.595 (5.0)	/	/
Data 4	0.752 (1.0)	0.602 (9.0)	0.591 (10.0)	0.477 (11.0)	0.660 (6.0)	0.710 (3.0)	0.619 (8.0)	0.663 (5.0)	0.632 (7.0)	0.719 (2.0)	0.705 (4.0)	/	/
Data 5	0.711 (1.0)	0.587 (7.0)	0.554 (10.0)	0.451 (11.0)	0.643 (4.0)	0.647 (3.0)	0.599 (6.0)	0.586 (8.0)	0.565 (9.0)	0.664 (2.0)	0.631 (5.0)	/	/
Data 6	0.685 (2.0)	0.536 (9.0)	0.501 (10.0)	0.448 (11.0)	0.636 (5.0)	0.710 (1.0)	0.578 (7.0)	0.602 (6.0)	0.549 (8.0)	0.664 (3.0)	0.651 (4.0)	/	/
Data 7	0.714 (2.0)	0.601 (7.0)	0.543 (10.0)	0.464 (11.0)	0.637 (5.0)	0.764 (1.0)	0.594 (8.0)	0.607 (6.0)	0.575 (9.0)	0.658 (3.0)	0.657 (4.0)	/	/
Data 8	0.722 (1.0)	0.616 (6.0)	0.594 (10.0)	0.503 (11.0)	0.636 (5.0)	0.697 (2.0)	0.615 (7.0)	0.603 (9.0)	0.609 (8.0)	0.654 (4.0)	0.665 (3.0)	/	/
Graph Data	/	/	/	/	/	/	/	/	/	/	/	0.649	0.586
Average Rank	(1.3)	(8.1)	(9.6)	(10.9)	(4.7)	(4.1)	(7.0)	(6.2)	(7.8)	(2.5)	(3.9)	/	/

Table S6. Critical values for the two-tailed Nemenyi test in this study, related to STAR Methods.

#Num of algorithms	2	3	4	5	6	7	8	9	10	11
$q_{0.05}$	1.960	2.343	2.569	2.728	2.850	2.949	3.031	3.102	3.164	3.219
$q_{0.10}$	1.645	2.052	2.291	2.459	2.589	2.693	2.780	2.855	2.920	2.978

Table S7. The performance of all models on the leave-cell-line-out, leave-drug-out, and leave-drug-combination-out cross validation, related to STAR Methods.

	F1-score	AUPR	Recall	MCC	Gmean
Leave-cell-line-out					
ForSyn	**0.361**	**0.454**	0.328	**0.311**	0.387
Original DF	0.201	0.398	0.181	0.165	0.243
DCE-DForest	0.198	0.435	0.134	0.165	0.254
DeepSynergy	0.107	Nan	0.091	0.072	0.160
Matchmaker	0.256	Nan	0.263	0.170	0.349
TranSynergy	0.334	0.447	0.310	0.181	0.352
SynPathy	0.141	Nan	0.092	0.135	0.211
XGBoost	0.135	0.354	0.091	0.139	0.177
Random Forest	0.163	0.385	0.114	0.119	0.215
RUS Boost	0.201	0.238	**0.355**	0.062	0.294
Balanced Bagging	0.265	0.263	0.265	0.135	0.366
DeepDDS-GCN	0.237	Nan	0.221	0.149	**0.394**
DeepDDS-GAT	0.128	Nan	0.137	0.082	0.295
Leave-drug-out					
ForSyn	**0.306**	**0.356**	0.247	**0.339**	0.467
Original DF	0.299	0.319	0.208	0.263	0.429
DCE-DForest	0.297	0.344	0.203	0.329	0.422
DeepSynergy	0.233	0.328	0.148	0.284	0.362
Matchmaker	0.301	0.315	0.246	0.264	0.471
TranSynergy	0.303	0.352	0.312	0.275	**0.513**
SynPathy	0.274	0.332	0.189	0.281	0.410
XGBoost	0.183	0.296	0.114	0.221	0.315
Random Forest	0.284	0.333	0.203	0.292	0.417
RUS Boost	0.196	0.151	**0.333**	0.092	0.491
Balanced Bagging	0.255	0.254	0.257	0.193	0.455
DeepDDS-GCN	0.252	0.340	0.230	0.215	0.447
DeepDDS-GAT	0.187	0.248	0.147	0.186	0.301
Leave-drug-combination-out					
ForSyn	**0.381**	**0.438**	0.332	**0.428**	0.559
Original DF	0.352	0.425	0.236	0.400	0.470
DCE-DForest	0.372	0.406	0.250	0.356	0.487
DeepSynergy	0.303	0.369	0.191	0.379	0.423
sMatchmaker	0.356	0.381	0.296	0.343	0.519
TranSynergy	**0.382**	0.432	0.337	0.390	**0.599**
SynPathy	0.367	0.409	0.253	0.411	0.494

XGBoost	0.308	0.374	0.206	0.341	0.448
Random Forest	0.362	0.419	0.238	0.425	0.477
RUS Boost	0.209	0.187	**0.431**	0.153	0.581
Balanced Bagging	0.332	0.346	0.328	0.299	0.552
DeepDDS-GCN	0.317	0.406	0.278	0.305	0.502
DeepDDS-GAT	0.231	0.319	0.190	0.250	0.358

Table S8. Ablation experiment of deep forest based on Data 8 and F1-score metric, related to STAR Methods.

Configuration	Description	F1-score
$DF^{(ADA*1+BAG*1+GBC *1+RF-CUS*1+ETF-DR*1)}$	Five different units are placed on each cascade layer.	0.562
$DF^{(ADA*1+BAG*1+GBC *1+RF-CUS*1)}$	Remove ETF-DR unit.	0.532
$DF^{(ADA*1+BAG*1+GBC *1+ ETF-DR*1)}$	Remove RF-CUS unit.	0.537
$DF^{(ADA*1+BAG*1+RF-CUS*1+ETF-DR*1)}$	Remove GBC unit.	0.553
$DF^{(ADA*1+GBC *1+RF-CUS*1+ETF-DR*1)}$	Remove BAG unit.	0.558
$DF^{(BAG*1+GBC *1+RF-CUS*1+ETF-DR*1)}$	Remove ADA unit.	0.554

Table S9. The top eight synergistic drug combinations predicted by ForSyn, related to Figure 3.

Drug A	Drug B	Cell Line	Verified
Erlotinib Hydrochloride	Azd1775	HT29	√
Erlotinib Hydrochloride	Mk-5108	HT29	√
Erlotinib Hydrochloride	Mk-2206	HT29	
Etoposide	Gefitinib	HT29	√
Erlotinib Hydrochloride	Dinaciclib	HT29	√
Erlotinib Hydrochloride	Pd325901	HT29	
Erlotinib Hydrochloride	Bez-235	HT29	
Azd1775	Vemurafenib	SW620	

Table S10. The genes involved in the top contributing DGE features in four cell lines, related to Figure 4.

Index	A549 cell line	HT29 cell line	MCF7 cell line	PC3 cell line
1	CCND3	PLOD3	PGM1	UFM1
2	TSPAN4	CAMSAP2	SPRED2	SIRT3
3	CLPX	INSIG1	EIF4G1	RPA1
4	PRUNE1	GNA15	CEMIP2	NSDHL
5	ELAVL1	SKIV2L	RALA	AKAP8L
6	LPAR2	MINDY1	HIF1A	GRWD1
7	HK1	PCBD1	TPM1	NCK1
8	GNA11	FOXO3	BNIP3L	COG4
9	NT5DC2	EPHA3	FKBP14	ABHD4
10	PYGL	MTF2	GOLT1B	ATF5

Data S1. The algorithm of random forest by clustering and stratified under-sampling, related to STAR Methods.

```
Input:
D: a binary training set, which is composed of the majority class ($D_{maj}$) and the minority class ($D_{min}$), $D= D_{maj}$ U $D_{min}$
d: the feature dimension
T: the number of decision trees in the Random Forest
Output:
The ensemble of decision trees.
Algorithm process:
For t=1,2…T Do
    Step 1 Perform AP clustering on the majority class ($D_{maj}$).
    Step 2 Perform stratified sampling on the clustering result of Step 1, to obtain the dataset $S_{maj}$, where $|S_{maj}|=|D_{min}|$.
    Step 3 Perform bootstrap sampling on the minority class ($D_{min}$) to obtain the dataset $S_{min}$, where $|S_{min}|=|D_{min}|$.
    Step 4 $D_{train} = S_{maj}$ U $S_{min}$.
    Step 5 Randomly select $\sqrt{d}$ features in $D_{train}$, to get the dataset $S_{train}$.
    Step 6 Use $S_{train}$ to train a decision tree.
End For
```

Data S2. The algorithm of extreme tree forest based on data complexity dimension reduction, related to STAR Methods.

Input:

D: the overall dataset
d: the feature dimension
T: the training set
V: the validation set //D=T∪V
R: the number of iteration //R = 5 in the experiment
s: the step size of the greedy algorithm
size: the feature size after dimension reduction
ETF: an extreme tree forest

Output:

Feature subset after dimension reduction, and a trained ETF.

Algorithm process:

```
Use data complexity metric to calculate the overlap area ($F_i$) of each feature in D. // Refer to Eq. (1)
Sort all the features in D by ascending order according to $F_i$.
Divide D into R subsets evenly, $D_r \subset D$, $r \in [1,R]$.
Initialize Size=0.
For r in Range R
    $T=D \setminus D_r$          //(R-1) subsets as the training set
    $V=D_r$                      //One subset as the validation set
    ACC=Accuracy(ETF, T, V)
// Use T to train the extreme tree forest, then use V to verify its performance
    Initialize BestSubset=∅,   $ACC_{max}$= ACC
    While T≠∅ Do
        T = T[: , : (d-s)]       // Training set removes the last s dimension
        V= V[: , : (d-s)]        // Validation set removes the last s dimension
        $ACC_{temp}$= Accuracy(ETF, T, V)
        If $ACC_{temp} > ACC_{max}$ Then
            $ACC_{max}$= $ACC_{temp}$
            BestSubset=T
        Else Break
    End While
    size=size+|BestSubset|       // |*| represents the dimension of the set
End For
size= size/R
Return D[: , : size]             // Return the feature subset after dimension reduction
Using D[: , : size] to train a ETF as the unit of the deep forest.
```