# Supplementary Document: Prediction of Synthetic Lethal Interactions in Human Cancers using Multi-view Graph Auto-Encoder

ZHIFENG HAO, DI WU, YUAN FANG, MIN WU, RUICHU CAI, XIAOLI LI

## 1. FEATURE EXTRACTION FOR FEATURE-BASED METHODS

For feature-based methods, we extracted 8 pair-wise features from different genres of biological data and 10 node-wise network features from PPI network. Specifically, we first downloaded the ontology and annotation files from http://geneontology.org/. Then we calculated three semantic similarity matrices for genes based on the sub-ontologies "biological process (BP)", "molecular function(MF)" and "cellular component (CC)", using the method proposed by Wang *et al*. [1]. We further downloaded the PPI data from BioGrid to construct a PPI network. Note that we removed all the SL pairs curated in this PPI network constructed from BioGrid [2]. Besides, we also constructed 4 features for each SL pair, derived from four sources: Pathway Co-membership, using the Canonical pathway database from Broad Institute's Molecular Signatures Database (MSigDB) [3]; Protein Complex Co-membership, using the CORUM protein complex database [4]; Protein interaction scores, using human protein-protein interaction database (Hippie) [5]; Protein top similarity, using human protein reference database (HPRD) [6]. Node-wise network features were calculated based on the PPI network constructed from BioGrid. They included degree, closeness, betweenness, eigenvector centrality and clustering.

**Table S1.** Names and descriptions of the features of genes.

| Name | Type | Description |
|---|---|---|
| BP | Pairwise | The number of biological process GO annotations shared between the source and target node. |
| CC | Pairwise | The number of molecular function GO annotations shared between the source and target node. |
| MF | Pairwise | The number of cellular component GO annotations shared between the source and target node. |
| Co-pathway | Pairwise | The number of protein pathways shared between the source and target node. |
| Co-complex | Pairwise | A value to measure how well associated a given node is with the other node. |
| Protein top similarity | pairswise | A value to measure the structure similarity between the source and target node. |
| PPI | pairwise | A binary matrix recording whether a give node is confirmed to be associated with the other node. |
| Degree | Node-wise | The number of edges coming in to or out of the node. |
| Closeness | Node-wise | The number of steps required to reach all other nodes from a given node. |
| Betweenness | Node-wise | The number of shortest paths in the entire graph that pass through the node. |
| Eigenvector | Node-wise | A measure of how well connected a given node is to other well-connected nodes. |
| Clustering | Node-wise | The clustering coefficient of the node. |

## 2. PARAMETER SETTING FOR THE EXPERIMENTS ON SYNLETHDB-BC

In the experiment on SynLethDB-BC, we also used a 2-layer GCN for all the GCN-based methods. In order to avoid overfitting, we reduced the training epochs and the size of hidden layers. The parameter settings of GCN-based methods are summarized in Table S2. In addition, we used the same hyper parameter setting in our SLMGAE, where the learning rate $\eta$, dropout rate $\gamma$, the parameters $\alpha$, $\beta$ and $C$ are set to 0.001, 0.3, 0.5, 2.0 and 1.0.

In CMFW, the dimension of latent representation $k$ is set to 50. In BLM-NII, we set the value of the linear combination weight as 0.79 and used the max function to generate the prediction

**Table S2.** Parameter settings for GCN based methods on SynLethDB-BC.

| Parameters | SLMGAE | GAE | DDGCN | MVGCN | A-MVGAE |
|---|---|---|---|---|---|
| Learning rate $\eta$ | 0.001 | 0.01 | 0.01 | 0.001 | 0.001 |
| Dropout rate $\gamma$ | 0.2 | 0.3 | 0.5 | 0.2 | 0.4 |
| # training epochs | 200 | 2,000 | 2,000 | 200 | 200 |
| early stop threshold | – | 1e-5 | 1e-5 | – | – |
| # GCN layers | 2 | 2 | 2 | 2 | 2 |
| # units in layer1 | 128 | 128 | 128 | 128 | 64 |
| # units in layer2 | 64 | 64 | 64 | 64 | 64 |

scores. In SL2MF, the parameter $c$ is set to 50. In GRSMF, the parameters $\lambda$ is set to $2^7$. The weight coefficients of each support view for the above two methods are summarized in the Table S5.

**Table S3.** Weight parameters for each support view in SL2MF and GRSMF on SynLethDB-BC.

| Support view | SL2MF | GRSMF |
|---|---|---|
| Co-expression | $2^{-7}$ | $2^{-5}$ |
| Mutual Exclusivity | $2^{-6}$ | $2^{-7}$ |
| Pathway | $2^{-6}$ | $2^{-1}$ |
| Protein Complex | $2^{-6}$ | $2^{-5}$ |
| PPI | $2^{-5}$ | $2^{-7}$ |

Following the setting as Liany *et al.* [7], we didn't built KNN graph for the support views.

## 3. PARAMETER SETTING FOR THE EXPERIMENTS ON GIS DATA

In the experiment on GIs data, we also used a 2-layer GCN for all the GCN-based methods. The parameter settings of GCN-based methods are summarized in Table S4. The hyper parameter in our SLMGAE are set to the learning rate $\eta = 0.0006$, dropout rate $\gamma = 0.6$, the parameters $\alpha = 4.0$, $\beta = 0.125$ and $C = 4.0$.

**Table S4.** Parameter settings for GCN based methods on Gis data.

| Parameters | SLMGAE | GAE | DDGCN | MVGCN | A-MVGAE |
|---|---|---|---|---|---|
| Learning rate $\eta$ | 0.0006 | 0.001 | 0.006 | 0.001 | 0.0006 |
| Dropout rate $\gamma$ | 0.4 | 0.3 | 0.5 | 0.5 | 0.25 |
| # training epochs | 800 | 800 | 1,000 | 800 | 800 |
| early stop threshold | – | – | 1e-5 | – | – |
| # GCN layers | 2 | 2 | 2 | 2 | 2 |
| # units in layer1 | 128 | 128 | 128 | 128 | 128 |
| # units in layer2 | 64 | 64 | 64 | 64 | 64 |

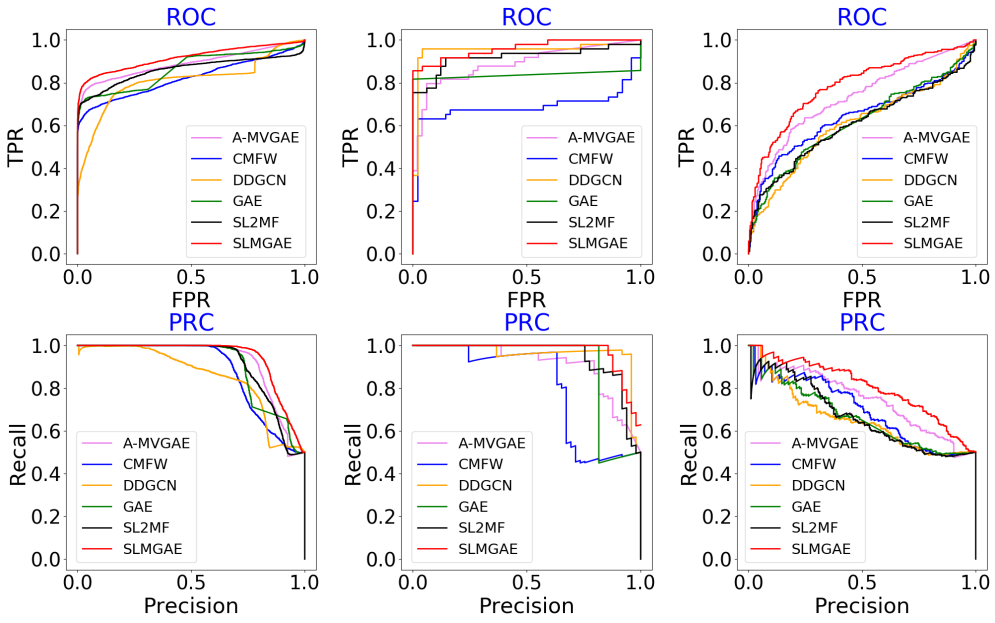
In CMFW, the dimension of latent representation $k$ is set to 30. In BLM-NII, we set the value of the linear combination weight as 0.82 and used the max function to generate the prediction scores. In SL2MF, the parameter $c$ is set to 50. In GRSMF, the parameters $\lambda$ is set to $2^{-1}$. The weight coefficients of each support view for the above two methods are summarized in the Table S5.

**Table S5.** Weight parameters for each support view in SL2MF and GRSMF on GIs data.

| Support view | SL2MF | GRSMF |
|---|---|---|
| GO BP | $2^{-1}$ | $2^{2}$ |
| GO CC | $2^{-2}$ | $2^{1}$ |
| PPI | $2^{-1}$ | $2^{2}$ |

## 4. ROC AND PRC OF VARIOUS METHODS

In this section, we summarized the ROC and PRC of various methods in figure S1. We can see that the results in Figure S1 verify with the results in Tables IV, V and VI of the paper that comparable results can be obtained for our SLMGAE. From Figure S1 we can visually observe that the ROC curve and PR curve of our model are mostly above the other methods, which means that our method can achieve a larger area under the ROC curve and area under the PR curve, indicating a better performance of our method.

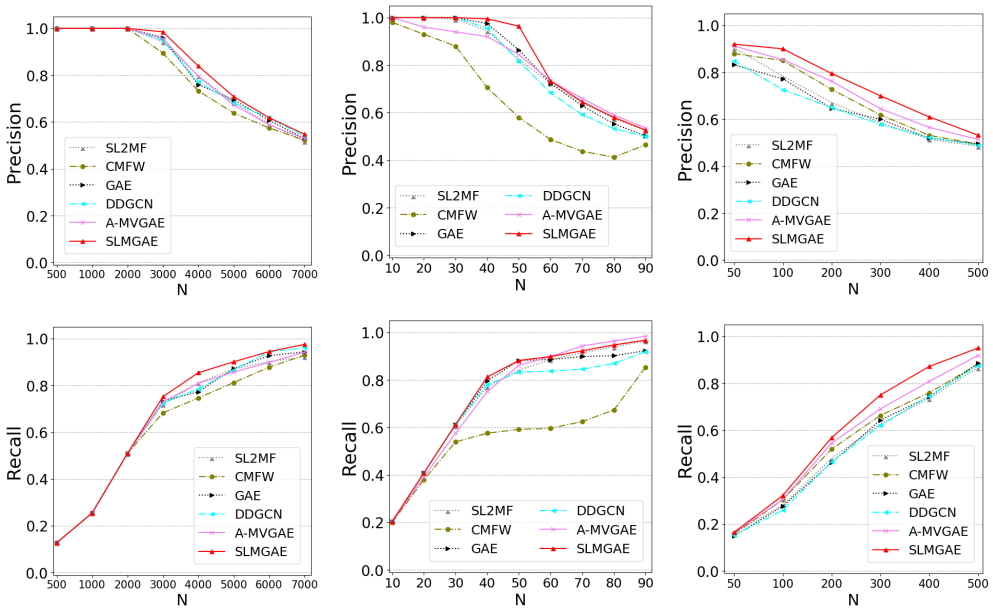



(a) Results on SynLethDB (b) Results on SynLethDB-BC (c) Results on GIs data

**Fig. S1.** ROC and PRC of various methods

## 5. COMPARISON OF MODEL PERFORMANCE ON THE PRECISION@N AND RECALL@N

For better comparison, we have also used both Precision@N and Recall@N metrics to evaluate our model and baseline approach. Figure S2 show the performance of each method on the different datasets. The results in Figure S2 can also validate the results in our paper. Specifically,in the experiments on the SynLethDB dataset, we can observe that all methods have almost the same precision and recall when $N \leq 2000$, and that our method outperforms the other methods when $N \geq 2000$. In summary, although our method does not outperform the other methods in all the value of N, our method also achieves comparable results in both Precision@N and Recall@N. We have included the evaluation results for various methods in terms of Precision@n and Recall@n in our supplementary materials.




(a) Results on SynLethDB (b) Results on SynLethDB-BC (c) Results on GIs data

**Fig. S2.** Precision@N and Recall@N of various SL prediction methods

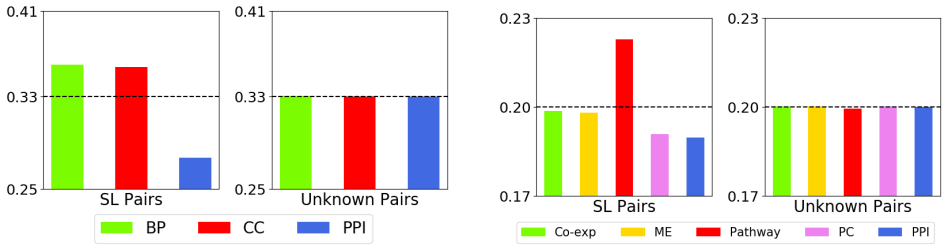
## 6. HOW THE ATTENTION LAYER WORKS

We averaged the attention scores of all positive and negative edges in each view to get the average attention scores $a_1, a_2, ..., a_n$ of each support view, then we use $softmax$ function normalize them to get the normalized attention scores $\boldsymbol{a} = softmax(a_1, a_2, ..., a_n)$, $softmax$ defined as follows:

$$S_i = \frac{e^{v_i}}{\sum_j e^{v_j}} \tag{S1}$$

Where $v_i$ represents the i-th element in vector $\boldsymbol{V}$, then the softmax value of this element is $S_i$.

The normalized attention scores of our model are summarized in Fig.S4. As shown in the figure, in our model, the distinction between positive and negative edges is very obvious. In the attention score of the positive edges, there will be some views that contribute particularly prominently (i.e. BP and CC in SynLethDB and Pathway in SynLethDB-BC). In the attention score of the negative side, the contribution of each perspective is similar, and there is no obvious difference. It is this difference that distinguishes the prediction of the positive and negative edges in our model and improves the performance of the model.




(a) Attention score in SynLethDB (b) Attention score in SynLethDB-BC

**Fig. S3.** Attention score

## 7. FEATURE ANALYSIS

Following the experiment setting as Cai *et al.* [8], we also consider using identity matrix $I$, adjacency matrix $A$ and adjacency matrix with self-loop $A + I$ as input feature matrix for our SLMGAE model. The results summarized in Table. S6.

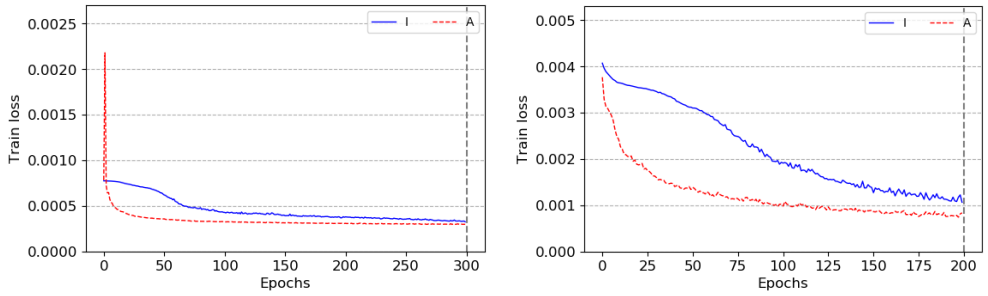
**Table S6.** SLMGAE with different feature matrix

| Features | SynLethDB | | | SynLethDB-BC | | |
|---|---|---|---|---|---|---|
| | AUROC | AUPR | F1 | AUROC | AUPR | F1 |
| I | 0.8445 ± 0.0096 | 0.8907 ± 0.0048 | 0.8574 ± 0.0068 | 0.9286 ± 0.0405 | 0.9379 ± 0.0242 | 0.9375 ± 0.0381 |
| A | **0.9174** ± 0.0045 | **0.9428** ± 0.0033 | 0.8717 ± 0.0074 | 0.9192 ± 0.0600 | 0.9279 ± 0.0377 | 0.9231 ± 0.0490 |
| A+I | 0.9145 ± 0.0039 | 0.9418 ± 0.0024 | **0.8726** ± 0.0054 | **0.9342** ± 0.0453 | **0.9414** ± 0.0259 | **0.9444** ± 0.0348 |

From the experimental results of SynLethDB, when we use the identity matrix $I$ as the initial feature, the performance of the model will be poor. As mentioned by kpif *et al*, the identity matrix is a very weak initial feature [9]. Our model can learn more information from adjacency matrix to improve the accuracy of prediction.

But judging from the experimental results of SynLethDB-BC, it is another situation.The performance of using the identity matrix as the initial feature is the best. This is because too few samples in SynLethDB-BC, coupled with the rich information provided by the adjacency matrix, caused the model to over-fit. From Figure.S5(b), we can see that when our model use $A$ as the initial feature, the final training loss is smaller than when $I$ is used, but the performance of the model is worse, which also shows that the model is over-fitting. From Figure.S5(a), in the experiments on SynLethDB, we can see that the loss of the model converges to a similar position, indicating that in the BC dataset with more samples, the model does not appear to over-fit.

**Fig. S4.** Training loss of SLMGAE with different feature matrix




(a) Training loss in SynLethDB

(b) Training loss in SynLethDB-BC

## 8. EVALUATION BY MODIFIED F-SCORE

In the Positive and Unknown (PU) learning problem, using standard metrics such as AUROC and AUPR to evaluate models is somewhat biased. This is because these negative examples are likely to be sampled very far away from the true decision boundary which we are trying to learn. To further validate the effectiveness of our model, we followed these two references [10, 11], we used the modified F-score to evaluate our model against the baseline algorithms. Specifically, we followed the setting from the experiments on the SynlethDB dataset and searched for the best F-score after sorting the predicted values. The modified F-score is defined by the following equation.

$$F - score = \frac{r \times r}{\Pr[f(X) = 1]}, \tag{S2}$$

where $r$ denotes Recall, and $\Pr[f(X) = 1]$ denotes the probability that a sample in the test set is predicted to be positive.

**Table S7.** Performance comparison of various SL prediction methods under 5-fold cross-validation.

| Methods | AUROC | AUPR | F1 | Modified F-score |
|---|---|---|---|---|
| BLM-NII | 0.6116 ± 0.0157 | 0.6507 ± 0.0281 | 0.6319 ± 0.0469 | 0.9998 ± 0.0003 |
| SL2MF | 0.8631 ± 0.0053 | 0.9106 ± 0.0026 | 0.8176 ± 0.0053 | 1.3760 ± 0.0058 |
| CMFW | 0.8209 ± 0.0030 | 0.8798 ± 0.0017 | 0.7795 ± 0.0021 | 1.2503 ± 0.0062 |
| g-CMF | 0.5536 ± 0.0057 | 0.5646 ± 0.0057 | 0.6469 ± 0.0049 | 0.7527 ± 0.0086 |
| GRSMF | 0.8642 ± 0.0046 | 0.8989 ± 0.0039 | 0.8220 ± 0.0032 | 1.3629 ± 0.0290 |
| GAE | 0.8664 ± 0.0043 | 0.9028 ± 0.0045 | 0.8307 ± 0.0065 | 1.4371 ± 0.0329 |
| DDGCN | 0.8783 ± 0.0040 | 0.9152 ± 0.0022 | 0.8204 ± 0.0048 | 1.3969 ± 0.0133 |
| MVGCN | 0.8556 ± 0.0049 | 0.9036 ± 0.0046 | 0.8326 ± 0.0039 | 1.4456 ± 0.0139 |
| A-MVGAE | 0.8796 ± 0.0036 | 0.9128 ± 0.0029 | 0.8489 ± 0.0048 | 1.4469 ± 0.0144 |
| SLMGAE | **0.9174 ± 0.0045** | **0.9428 ± 0.0032** | **0.8717 ± 0.0074** | **1.5363 ± 0.0101** |

Table S7 summarized the F-scores for each algorithm. As we can observe from the Table R1, SLMGAE achieves the best performance in terms of the modified F-score. Overall, the results for various algorithms in terms of the modified F-score are similar to those of the other metrics. For example, MVGCN and A-MVGAE have an modified F-score of approximately 1.44, and they also have comparable AUROC, AUPR and F1 scores. We calculated the spearman correlation between the modified F score and other metrics and the results are summarised in Table S8. As table S8 shows, the spearman correlation between the modified F score and the AUROC and AUPR scores is above 0.85, and the correlation with the F1 score is even higher, at 0.95. Although these metrics (i.e. AUROC, AUPR and F1) are biased, the consistency between several metrics shows that they are still informative.

**Table S8.** Spearman correlation between modified F-score and other metrics

| | AUROC | AUPR | F1 |
|---|---|---|---|
| Modified F-score | 0.8545 | 0.8667 | 0.9515 |

## 9. CASE STUDY

We summarized 30 SL pairs supported by existing publications in Table.S9. In addition, we also summarized top 5000 prediction of our SLMGAE model, a total of 123 SL pairs can be supported by existing literature. You can find these data on https://github.com/DiNg1011/SLMGAE.

**Table S9.** Top predicted SL pairs with literature support

| # | Rank | Gene1 | Gene2 | PubMed ID | Evidence | Prediction score |
|---|---|---|---|---|---|---|
| 1 | 21 | MAPK1 | TP53 | 23728082 | in-silico prediction | 0.5742 |
| 2 | 25 | PIK3CA | TP53 | 26427375 | in-silico prediction | 0.5323 |
| 3 | 50 | JUN | TP53 | 23728082 | in-silico prediction | 0.4120 |
| 4 | 90 | RAD51 | TP53 | 23728082 | in-silico prediction | 0.3685 |
| 5 | 137 | AR | TP53 | 23728082 | in-silico prediction | 0.3357 |
| 6 | 143 | BRCA1 | KRAS | 24104479 | shRNA screening | 0.3322 |
| 7 | 174 | CCT5 | KRAS | 28700943 | CRISPR and shRNA screens | 0.3147 |
| 8 | 176 | BRCA1 | PIK3CA | 26427375 | in-silico prediction | 0.3146 |
| 9 | 182 | BIRC5 | KRAS | 28700943 | CRISPR and shRNA screens | 0.3100 |
| 10 | 198 | HDAC9 | MYC | 29764852 | HDAC inhibition | 0.3055 |
| 11 | 234 | MCL1 | PARP1 | 31300006 | in-silico prediction | 0.2952 |
| 12 | 289 | ADK | KRAS | 27655641 | in-silico prediction | 0.2829 |
| 13 | 296 | SRC | TP53 | 23728082 | in-silico prediction | 0.2813 |
| 14 | 335 | KRAS | UNC13B | 24104479 | shRNA screening | 0.2741 |
| 15 | 352 | BRAF | TP53 | 24025726 | in-silico prediction | 0.2712 |
| 16 | 402 | KRAS | TRIP11 | 25407795 | Combinatorial RNAi | 0.2643 |
| 17 | 428 | BCL2L1 | PIK3CA | 31300006 | in-silico prediction | 0.2594 |
| 18 | 438 | CALM1 | KRAS | 27655641 | in-silico prediction | 0.2585 |
| 19 | 506 | KRAS | LATS1 | 27655641 | in-silico prediction | 0.2501 |
| 20 | 507 | BCL2 | BCL2L1 | 29251726 | CRISPR screening | 0.2500 |
| 21 | 522 | KRAS | WBP11 | 24104479 | shRNA screening | 0.2481 |
| 22 | 566 | BRCA1 | BRCA2 | 31300006 | in-silico prediction | 0.2440 |
| 23 | 641 | KRAS | PCNT | 28700943 | CRISPR and shRNA screens | 0.2376 |
| 24 | 666 | SMAD3 | TP53 | 23728082 | in-silico prediction | 0.2351 |
| 25 | 757 | EGLN3 | KRAS | 27655641 | in-silico prediction | 0.2276 |
| 26 | 758 | KRAS | NOP2 | 28700943 | CRISPR and shRNA screens | 0.2275 |
| 27 | 840 | KRAS | RPS16 | 24104479 | shRNA screening | 0.2216 |
| 28 | 841 | AKT1 | CHEK1 | 28319113 | CRISPR-Cas9 | 0.2214 |
| 29 | 927 | BCL2L1 | CDKN1A | 31300006 | in-silico prediction | 0.2160 |
| 30 | 998 | JAK2 | KRAS | 25407795 | Combinatorial RNAi | 0.2122 |

## REFERENCES


1. J. Z. Wang, Z. Du, R. Payattakool, P. S. Yu, and C.-F. Chen, "A new method to measure the semantic similarity of go terms," Bioinformatics **23**, 1274–1281 (2007).
2. R. Oughtred, C. Stark, B.-J. Breitkreutz, J. Rust, L. Boucher, C. Chang, N. Kolas, L. O'Donnell, G. Leung, R. McAdam *et al.*, "The biogrid interaction database: 2019 update," Nucleic acids research **47**, D529–D541 (2019).
3. A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander *et al.*, "Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles," Proc. Natl. Acad. Sci. **102**, 15545–15550 (2005).
4. M. Giurgiu, J. Reinhard, B. Brauner, I. Dunger-Kaltenbach, G. Fobo, G. Frishman, C. Montrone, and A. Ruepp, "Corum: the comprehensive resource of mammalian protein complexes—2019," Nucleic acids research **47**, D559–D563 (2019).
5. G. Alanis-Lobato, M. A. Andrade-Navarro, and M. H. Schaefer, "Hippie v2. 0: enhancing meaningfulness and reliability of protein–protein interaction networks," Nucleic acids research p. gkw985 (2016).
6. T. t. Keshava Prasad, R. Goel, K. Kandasamy, S. Keerthikumar, S. Kumar, S. Mathivanan, D. Telikicherla, R. Raju, B. Shafreen, A. Venugopal *et al.*, "Human protein reference database—2009 update," Nucleic acids research **37**, D767–D772 (2009).
7. H. Liany, A. D. Jeyasekharan, and V. Rajan, "Predicting synthetic lethal interactions using heterogeneous data sources," Bioinformatics (2019).
8. R. Cai, X. Chen, Y. Fang, M. Wu, and Y. Hao, "Dual-dropout graph convolutional network for predicting synthetic lethality in human cancers," Bioinformatics (2020).
9. T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," (2017).
10. W. S. Lee and B. Liu, "Learning with positive and unlabeled examples using weighted logistic regression," in *ICML*, vol. 3 (2003), pp. 448–455.
11. J. Bekker and J. Davis, "Learning from positive and unlabeled data: A survey," Mach. Learn. **109**, 719–760 (2020).