

METHODOLOGY ARTICLE

Open Access



# Identifying potential association on gene-disease network via dual hypergraph regularized least squares

Hongpeng Yang<sup>1†</sup>, Yijie Ding<sup>2\*</sup>, Jijun Tang<sup>3†</sup> and Fei Guo<sup>4\*</sup> 

## Abstract

**Background:** Identifying potential associations between genes and diseases via biomedical experiments must be the time-consuming and expensive research works. The computational technologies based on machine learning models have been widely utilized to explore genetic information related to complex diseases. Importantly, the gene-disease association detection can be defined as the link prediction problem in bipartite network. However, many existing methods do not utilize multiple sources of biological information; Additionally, they do not extract higher-order relationships among genes and diseases.

**Results:** In this study, we propose a novel method called Dual Hypergraph Regularized Least Squares (DHRLS) with Centered Kernel Alignment-based Multiple Kernel Learning (CKA-MKL), in order to detect all potential gene-disease associations. First, we construct multiple kernels based on various biological data sources in gene and disease spaces respectively. After that, we use CKA-MKL to obtain the optimal kernels in the two spaces respectively. To specific, hypergraph can be employed to establish higher-order relationships. Finally, our DHRLS model is solved by the Alternating Least squares algorithm (ALSA), for predicting gene-disease associations.

**Conclusion:** Comparing with many outstanding prediction tools, DHRLS achieves best performance on gene-disease associations network under two types of cross validation. To verify robustness, our proposed approach has excellent prediction performance on six real-world networks. Our research work can effectively discover potential disease-associated genes and provide guidance for the follow-up verification methods of complex diseases.

**Keywords:** Gene-disease association network, Hypergraph learning, Dual Laplacian regularized least squares, Bipartite network, Multiple kernel learning

## Background

Identification of the association between disease and human gene has attracted more attention in the field of biomedicine, and has become an important research topic. A great deal of evidence shows that understanding genes related to diseases is of great help to prevent

and treat diseases. However, identifying the relationship between disease and gene by biological experiments has to spend a long time and cost. Many computational models have been proposed to solve some similar biologically related problems. For example, in the fields of biology [1–3], pharmacy [4], and medicine [5, 6], machine learning methods help solve many analytical tasks.

In order to explore the relationship between gene and disease, a variety of algorithms have been proposed for association prediction. The typical machine learning methods [7–10] is to extract relevant features of known genetic data of each disease and train the model to deter-

\*Correspondence: wuxi\_dyj@163.com; guofei@eileen.com

†Hongpeng Yang and Jijun Tang contributed equally to this work.

<sup>2</sup>Yangtze Delta Region Institute, University of Electronic Science and Technology of China, Quzhou, China

<sup>4</sup>School of Computer Science and Engineering, Central South University, Changsha, China

Full list of author information is available at the end of the article



© The Author(s). 2021 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

mine which disease is related to those genes, so these algorithms are usually single-task algorithms for each disease. This model needs to be trained separately. Therefore, for a new disease or an existing disease with few known genes, due to the lack of known association data or the relevant information between various diseases, it is difficult to train the learning model. As a machine learning method, the matrix completion methods [11–13] can solve the above problem by calculating the similarity information and predicting the association between disease and gene, but the matrix completion method usually takes a long time to converge the local optimal solution. The other type is network-based model [14–17]. Li et al. [17] predicted the association by systematically embedding a heterogeneous network of genes and diseases into Graph Convolutional Network. This model usually divides genes and diseases into two heterogeneous networks. The edges in network represent the similarity between nodes. The model is based on the assumption that genes with high similarity are easily related to similar diseases. However, they are biased by the network topology, and it is necessary to rely on effective similarity information. It is not easy for these methods to integrate related sources of multiple genes and diseases.

Multiple Kernel Learning (MKL) is an important machine learning method, which can effectively combine multi-source information to improve the model effect, and is applied to many biological problems. For instance, Yu et al. [8] implemented one-class of Support Vector Machine while optimizing the linear combination of the gene nucleus and the MKL method. Ding et al. [18–21] proposed multiple information fusion models to identify drug-target and drug-side effect associations. Wang et al. [22] proposed a novel Multiple Kernel Support Vector Machine (MKSVM) classifier based on Hilbert Schmidt Independence Criterion to identify membrane proteins. Shen [23] and Ding et al. [24] proposed a MKSVM model to identify multi-label protein subcellular localization. Ding et al also employ fuzzy-besd model to predict DNA-binding proteins [25] and protein crystallization [26]. Zhang et al. [27] developed an ensemble predictive model of classifier chain to identify anti-inflammatory peptides.

LapRLS framework [28] is often used in various fields based on machine learning model, such as the prediction of Human Microbe-Disease Association [29] and the detection of human microRNA-disease association [30]. At the same time, Hypergraph learning [31–33] is becoming popular. Hypergraphs can represent more complex relationships among various objects. Bai et al. [34] introduced two end-to-end trainable operators to the family of graph neural networks, i.e., hypergraph convolution and hypergraph attention. Whilst hypergraph convolution defines the basic formulation of performing convolution on a hypergraph, hypergraph attention further enhances

the capacity of representation learning by leveraging an attention module. Zhang et al. [35] developed a new self-attention based graph neural network called Hyper-SAGNN applicable to homogeneous and heterogeneous hypergraphs with variable hyperedge sizes. Ding et al. [36] predicted miRNAs-disease associations by a hypergraph regularized bipartite local model, which is based on hypergraph embedded Laplacian support vector machine.

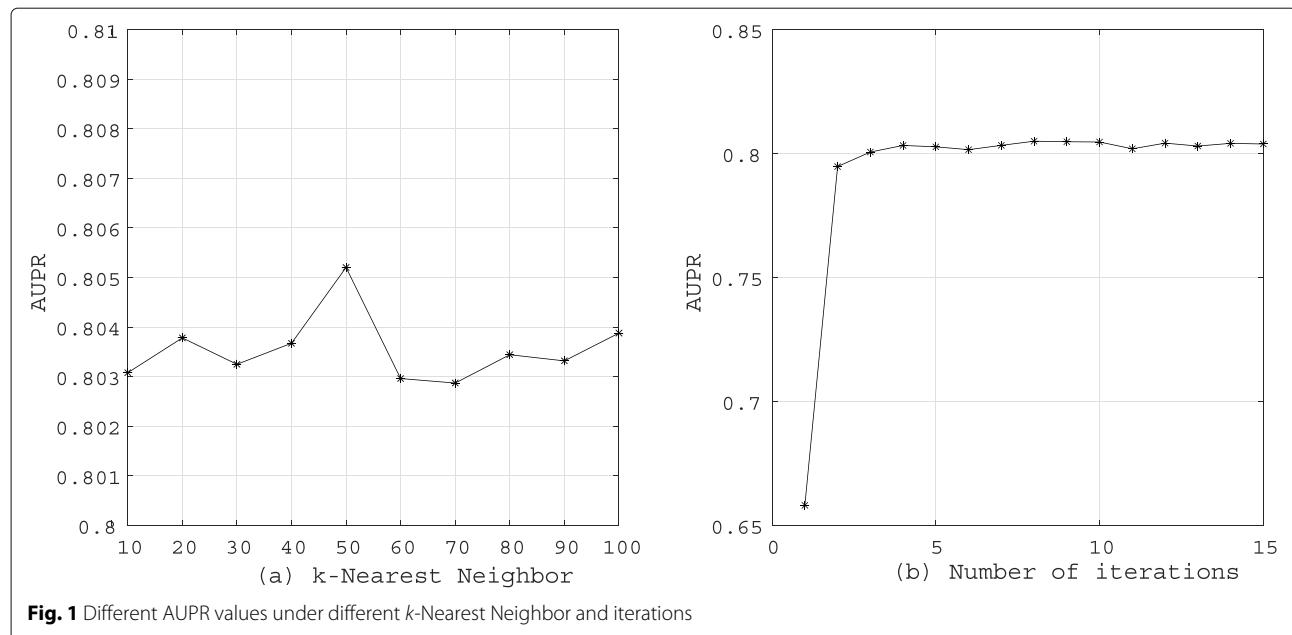
Inspired by what is mentioned above, we propose a novel prediction method named Dual Graph Hypergraph Least Squares model (DHRLS) to predict gene-disease associations. Some computational models based on graph learning can effectively solve various network problems. In this paper, the gene-disease association detection can be defined as the link prediction problem in bipartite network [37–39]. Furthermore, two feature spaces are described by similarity information of multiple genes and diseases. Multiple kernel learning is also used to combine multiple informations linearly. Here, we use the Centered Kernel Alignment-based Multiple Kernel Learning (CKA-MKL) [40] to obtain weights of multiple kernels and then combine these kernels via optimal weights in two spaces, respectively. In addition, we also embed hypergraphs in graph regular terms to preserve high-order information of genes and diseases, using more complex information to improve prediction performance. To prove the effectiveness of our proposed method, six types of real networks and one gene-disease associations network are employed to test our predictive model. On the gene-disease associations dataset, our method has been compared with some methods under two types of cross-validation (CV). Comparing DHRLS with other state-of-the-art methods on predicting gene-disease associations, including CMF, GRMF and Spa-LapRLS, our model achieves the highest AUC and AUPR in 10-fold cross validation under CV1, but our model achieves lower AUC under CV2 compared with Spa-LapRLS. At the same time, DHRLS has excellent prediction performance on six benchmark datasets.

## Results

In order to better test the performance of our method, our proposed approach is verified on real gene-disease associations dataset under two types of cross validation. We also test the capability of DHRLS in predicting novel disease after confirming the excellent performance of our method based on cross validation. Furthermore, we employ benchmark datasets to evaluate our approach and compare it with other existing methods.

### Dataset

We download the dataset of gene-disease associations from [41] (<http://cssb2.biology.gatech.edu/knowgene>). Since the number of genes is too large and the information



**Fig. 1** Different AUPR values under different  $k$ -Nearest Neighbor and iterations

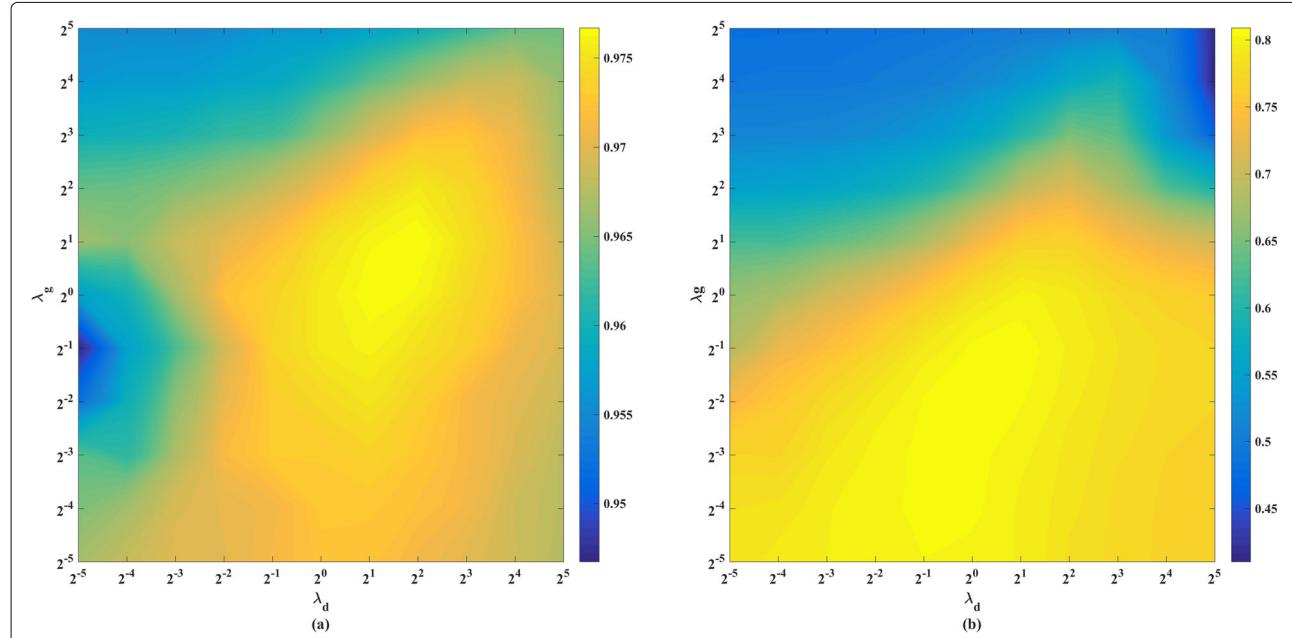
is insufficient, we remove some redundant gene data. Finally, our dataset contains 31,519 associations of 960 diseases and 3,118 genes, where 279 genes are associated with only one disease.

#### Evaluation measurements

The 10-fold Cross Validation (CV) is usually used to verify the bipartite network detection. In order to compare the prediction performance with other methods under

the same evaluation measurement, we will also use 10-fold CV for verification. At the same time, Area under the receiver operating characteristic curve (AUC) and Area Under the Precision-Recall curve (AUPR) as the major evaluation indicator, will also be applied to evaluate methods. There are two CV settings as follows:

**CV1:** Pair prediction. All gene-disease associations are randomly divided into test set and training set, and the associations in the test set are removed.



**Fig. 2** The AUC (a) and AUPR (b) of models with different  $\lambda_d$  and  $\lambda_g$  under CV1.  $\lambda_d$  (horizontal axis) and  $\lambda_g$  (vertical axis) are set from  $2^{-5}$  to  $2^5$  with step  $2^1$ . The yellow color is the higher value, and blue color is the lower value

**Table 1** The performance of different models under CV1

Model	AUC	AUPR
CKA-MKL + DHRLS	<b>0.9742</b>	<b>0.8092</b>
mean weighted + DHRLS	0.9703	0.8006
$\mathbf{K}_{GIP}^d \& \mathbf{K}_{GIP}^g + DHRLS$	0.9554	0.7377
$\mathbf{K}_{SEM}^d \& \mathbf{K}_{GO}^g + DHRLS$	0.9154	0.2827

**CV2:** Disease prediction. All diseases are randomly divided into test set and training set, and all associations of diseases in the test set are removed.

#### Parameter settings

In our study, DHRLS has some parameters  $\lambda_d$ ,  $\lambda_g$ ,  $\beta$ ,  $k$  and number of iterations. In the parameter selection, we consider all combinations of following values: number of  $k$ -Nearest Neighbor is from 10 to 100 (with step 10); number of iterations is {1,2,...,15};  $\{\lambda_d^d, \lambda_g^g\}$  for  $\lambda_d$  and  $\lambda_g$ ;  $\beta = 1$ .

Figure 1 shows the results of our model obtained under different iteration times and  $k$  values. For the number of  $k$ -Nearest Neighbor, we select the optimal  $k$  under the highest AUPR value and can clearly find that AUPR reaches its peak when  $k = 50$ . For the number of iterations, it basically converges at the four times. In order to train the model more fully, we finally choose the number of iterations to be 10.

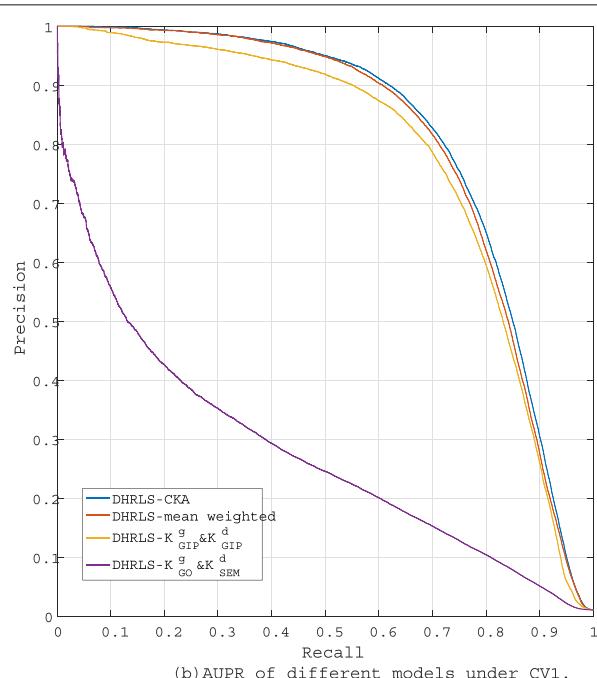
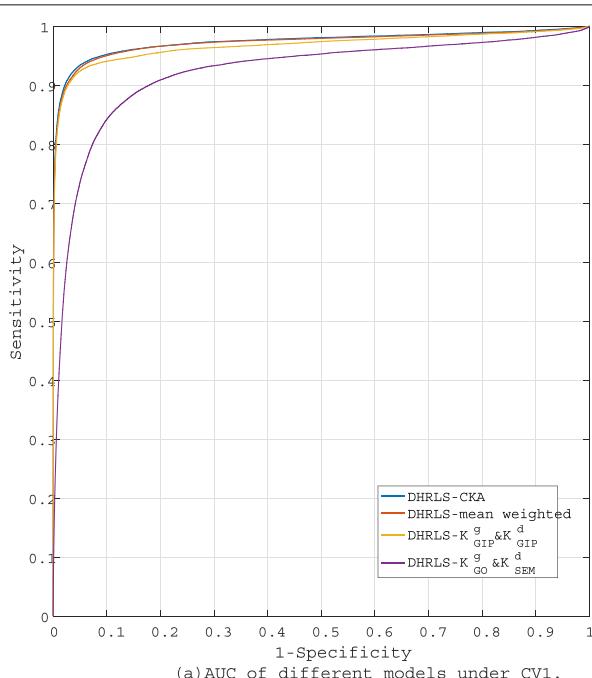
Figure 2 shows the results of AUC and AUPR in grid search for parameters  $\lambda_d$  and  $\lambda_g$ . The optimal  $\lambda_d^d$  and  $\lambda_g^g$  are also selected under highest AUPR value. In this study, the optimal parameters of Hypergraph Laplace regular terms are obtained on  $\lambda_d = 1$  and  $\lambda_g = 0.25$ . Under this parameter selection, the AUC value is relatively high.

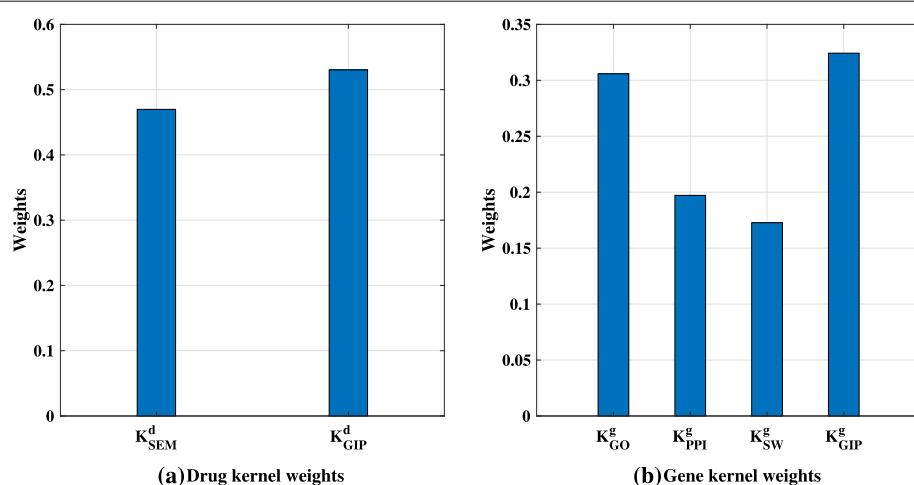
#### Evaluation on gene-disease association data

##### Performance analysis

We evaluate the different performance of CKA-MKL, mean weighted-based MKL and single kernel ( $\mathbf{K}_{SEM}^d \& \mathbf{K}_{GO}^g$  and  $\mathbf{K}_{GIP}^d \& \mathbf{K}_{GIP}^g$ ). The testing results are shown in Table 1 and Fig. 3.

Obviously, the model of CKA-MKL on DHRLS obtains the best performance with AUC of 0.9742 and AUPR of 0.8092. Comparing with mean weighted on DHRLS, AUPR and AUC are increased by 0.0086 and 0.0039. This means that CKA combines multi-kernel information more effectively than simple average combination.

**Fig. 3** The performance of different models under CV1

**Fig. 4** The kernel weights of drug and gene space respectively

What's more, DHRLS with single kernel ( $K_{SEM}^d$  &  $K_{GO}^g$ ) obtains lower performance than the model with GIP kernel. Therefore, GIP is an effective method to calculate the kernel matrix. By comparing the results of single kernel and multi-kernel models, combining multiple information is an effective method to improve the prediction effect of the model.

Furthermore, Fig. 4 shows the weights of each kernel matrix in the gene space and disease space. The weight of the kernel indicates the degree of contribution of the corresponding kernel matrix. Comparing the weights in the gene and disease spaces, the GIP kernel has a higher weight in both spaces, which is consistent with the results in Table 1. In gene space, except for GIP kernel, the kernel weight of  $K_{GO}^g$  is higher than  $K_{PPI}^g$  and  $K_{SW}^g$ . This means that  $K_{GO}^g$ 's contribution to the overall is better than the other two kernel matrices.

#### Comparison to existing predictors

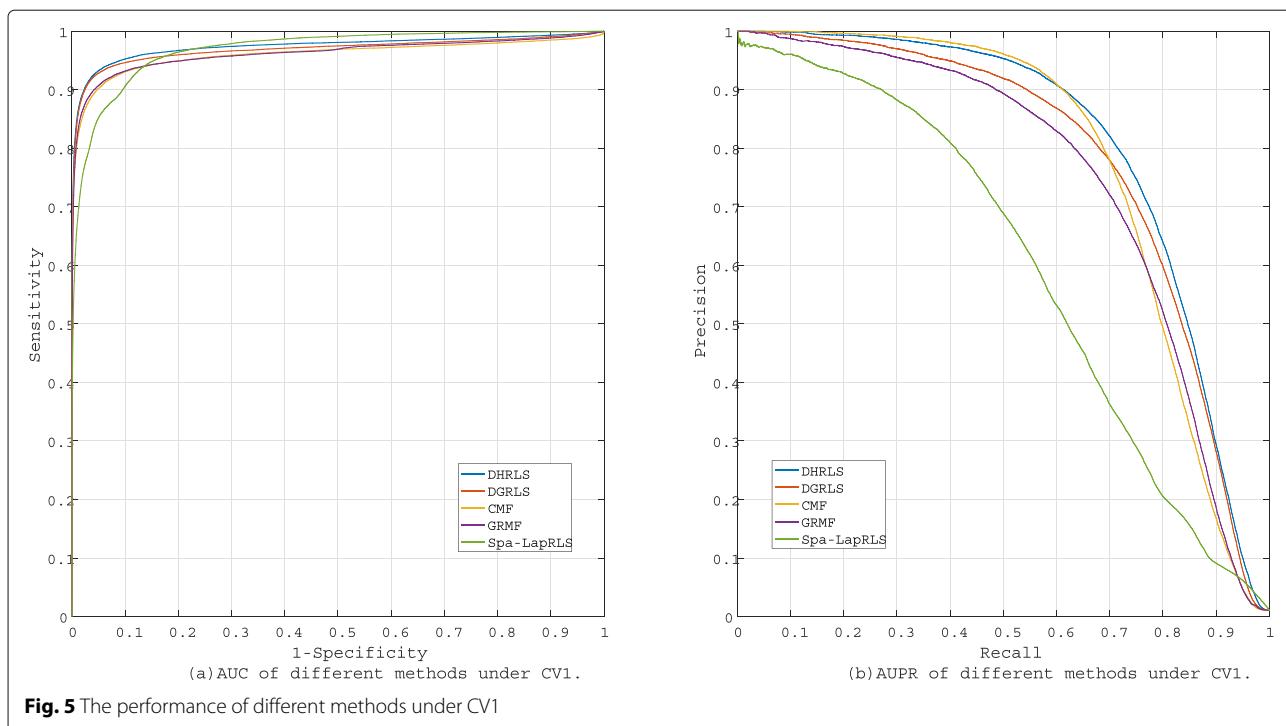
Many excellent methods have been proposed to predict the bipartite network link, including Spa-LapRLS [30], GRMF [42] and CMF [43]. Our method is compared to the existing methods and DGRLS under CV1 and CV2, respectively. Under CV1, the results are shown in

Table 2 and Fig. 5. Our method achieves the best AUC (0.9742) and AUPR (0.8092). For AUC, DHRLS is not much different from DGRLS and Spa-LapRLS, which is about 0.01 higher than GRMF and CMF. As for AUPR, DHRLS achieves better performance than other methods. Comparing the results of DHRLS and DGRLS, it can be seen that the hypergraph-based model is better than the normal graph model, which shows that the high-level graph information constructed by the hypergraph is helpful for the predict performance. This is related to the ability of hypergraph to effectively find similar information between nodes. At the same time, the methods based on LapRLS (DHRLS, DGRLS and Spa-LapRLS) are higher than those based on matrix factorization (GRMF and CMF), indicating that the model framework of LapRLS has more advantages in the prediction of gene-disease associations.

In order to test the performance of our method detecting new diseases, the associations for new diseases (CV2) are not observed in the training set. Table 3 and Fig. 6 show the results of CV2. Under CV2, our method obtains best AUPR (0.1413). However, the performance of our model on AUC (0.8987) is secondary best, which is about 0.02 lower than that of Spa-LapRLS.

**Table 2** The performance of different methods under CV1

Method	AUC	AUPR
DHRLS	<b>0.9742</b>	<b>0.8092</b>
DGRLS	0.9700	0.7842
Spa-LapRLS	0.9704	0.6222
GRMF	0.9609	0.7521
CMF	0.9594	0.7823



**Fig. 5** The performance of different methods under CV1

Comparing the results of DGRLS and DHRLS under CV1 and CV2, we clearly find that utilizing hypergraph to establish higher-order relationships greatly improves the predictive ability of the model.

#### Case study

Our model can predict genes associated with new diseases. Here, we use DHRLS to rank the predicted values of genes related to new diseases in descending order. The higher the ranking, the more likely it is to interact. We set the value of a disease in the correlation matrix to 0 as a new disease. One example is Lung Diseases. We intercepted the top 50 predicted genes and 40 (80%) known related genes in the predicted results. All predicted ranking results are shown in Table 4.

#### Evaluation on six benchmark datasets

To test the performance of our proposed method, we consider six real-world networks: (i) G-protein coupled receptors (GPC Receptors): the biological network of drugs binding GPC receptors; (ii) Ion channels: the

biological network of drugs binding ion channel proteins; (iii) Enzymes: the biological network of drugs binding enzyme proteins; (iv) Southern Women (referred here as "SW"): the social relations network of women and events; (v) Drug-target: the chemical network of drug-target interaction; (vi) Country-organization (referred here as "CO"): the network of organization most related to the country. Detailed information about six datasets is described in Table 5.

Since there is only the data of interaction matrix of binary network, in order not to introduce additional data, we directly use the GIP kernel extracted from the interaction matrix as the kernel matrix for each real-world network. The kernel is defined as follows:

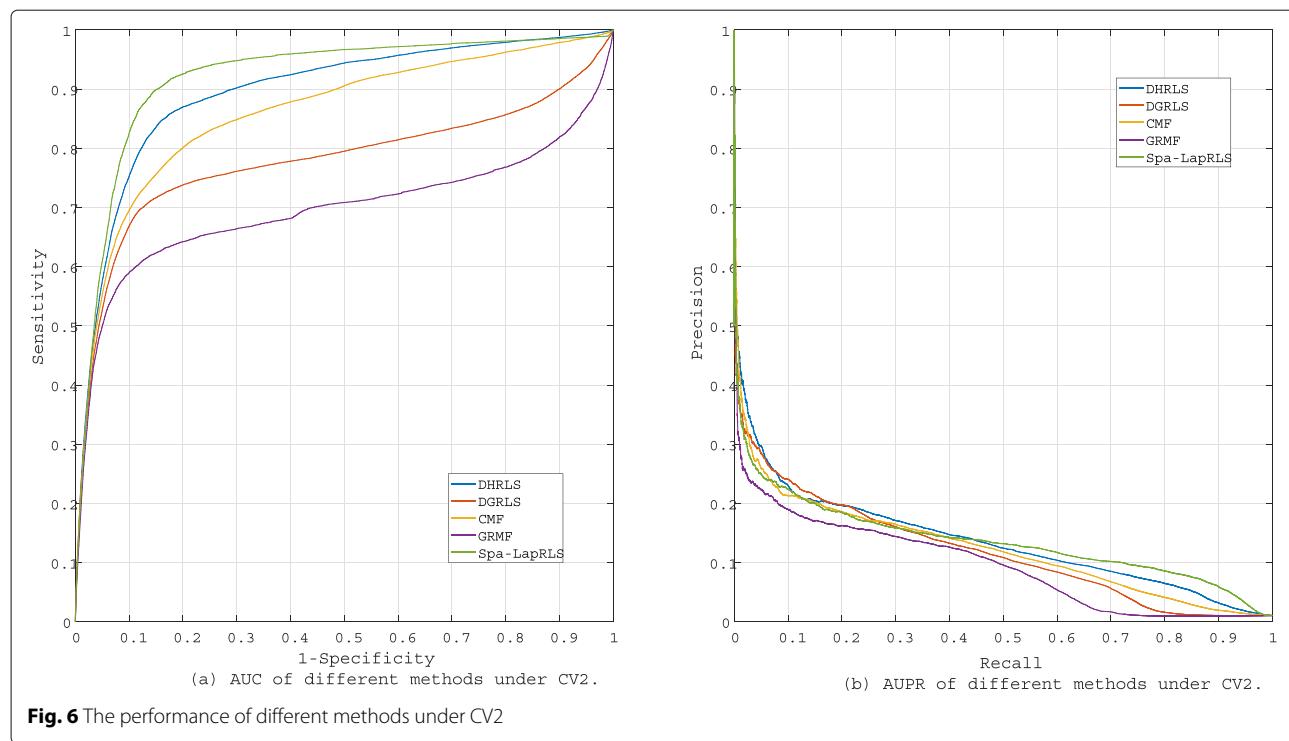
$$\mathbf{K}_1^*(i, j) = \exp(-\gamma_1 \|\mathbf{Y}_i - \mathbf{Y}_j\|^2) \quad (1a)$$

$$\mathbf{K}_2^*(m, n) = \exp(-\gamma_2 \|\mathbf{Y}_m^T - \mathbf{Y}_n^T\|^2) \quad (1b)$$

where  $\mathbf{Y}$  is the train set of binary network, and  $\mathbf{Y}_i$  is the vector of associations.

**Table 3** The performance of different methods under CV2

Method	AUC	AUPR
DHRLS	0.8987	<b>0.1413</b>
DGRLS	0.7887	0.1233
Spa-LapRLS	<b>0.9199</b>	0.1402
GRMF	0.7240	0.1013
CMF	0.8640	0.1256



**Fig. 6** The performance of different methods under CV2

We test our method on above six datasets and compare results with other methods [44]. Wang et al. [44] proposed a framework, called Similarity Regularized Non-negative Matrix Factorization (SRNMF), for link prediction in bipartite networks by combining the similarity based structure and the latent feature model from a new perspective. Tables 6 and 7 show the comparison of precision and AUC for six real-world networks. DHRLS performs better than other methods on Enzymes and Ionchannel networks, and values of our precision and AUC are higher than others. For GPC and Drug-target networks, the precision is same, but AUC is slightly higher. This directly indicates the clear performance advantage of our approach in real-world binary networks.

## Discussion

We developed the model DHRLS for the gene-disease association prediction. In order to evaluate our model, we test not only on real gene-disease associations dataset, but also on some benchmark datasets. By comparing the results of single-kernel model and multi-kernel model, MKL can effectively combine multi-kernel information to improve the predictive ability of the model. By adjusting different kernel weights, different kernel matrices can express different levels of information. However, MKL needs to be applied to samples with multiple feature information, and the application effect is not obvious for problems with fewer features. The comparison of DHRLS

and DGRLS can illustrate the effectiveness of hypergraph. After adding the hypergraph, the result of the model is obviously improved, which is caused by the characteristics of the hypergraph. Hypergraph uses high-order information between nodes, that is, a hyperedge can connect more than two nodes, which can better indicate the degree of similarity between nodes. Comparing DHRLS with other state-of-the-art methods on predicting gene-disease associations, including CMF, GRMF and Spa-LapRLS, our model achieves the highest AUC and AUPR in 10-fold cross validation under CV1, but our model achieves lower AUC under CV2 compared with Spa-LapRLS. At the same time, DHRLS has excellent prediction performance on six benchmark datasets.

Nevertheless, our model still has some flaws. First of all, the model contains a large number of matrix operations and optimization problems, and lacks a certain degree of simplicity. Secondly, we need to calculate the multi-kernel information of the sample. Therefore, we cannot achieve predictions for samples without features. At present, most of the computational methods are developed to predict the associations of gene-disease, and there is still a great possibility to improve the prediction performance. For example, hypergraph can be considered in the graph based method. In the future, for optimizing the model and improving the prediction performance, we can add some data preprocessing and calculate simplification on the basis of DHRLS, as well as better method to build hypergraph.

**Table 4** Predicted top 50 genes for Lung Diseases by our method

Rank	Gene	Confirm	Rank	Gene	Confirm
1	DQB1_HUMAN	Y	26	PBX2_HUMAN	Y
2	DQA1_HUMAN	Y	27	BRD2_HUMAN	Y
3	IFNG_HUMAN	N	28	TGFB1_HUMAN	N
4	DQA2_HUMAN	Y	29	CFTR_HUMAN	Y
5	ACE_HUMAN	Y	30	PAFA_HUMAN	Y
6	TNFA_HUMAN	Y	31	TAP2_HUMAN	N
7	DPB1_HUMAN	Y	32	UBP38_HUMAN	Y
8	CTLA4_HUMAN	N	33	MUC7_HUMAN	Y
9	ADA33_HUMAN	Y	34	DPP10_HUMAN	Y
10	NOTC4_HUMAN	Y	35	CH3L1_HUMAN	Y
11	PDE4D_HUMAN	Y	36	IL6RA_HUMAN	Y
12	IL13_HUMAN	N	37	RNBP6_HUMAN	Y
13	DRA_HUMAN	Y	38	CHIT1_HUMAN	Y
14	SMAD3_HUMAN	Y	39	ELF3_HUMAN	Y
15	IL4_HUMAN	N	40	ORML3_HUMAN	Y
16	DPA1_HUMAN	Y	41	S2546_HUMAN	Y
17	SUOX_HUMAN	Y	42	CDK2_HUMAN	Y
18	IKZF4_HUMAN	Y	43	IL2RB_HUMAN	Y
19	EMSY_HUMAN	Y	44	IL33_HUMAN	Y
20	IL18R_HUMAN	Y	45	DOA_HUMAN	Y
21	ILRL1_HUMAN	Y	46	X5CF87_HUMAN	Y
22	ZNT8_HUMAN	Y	47	TBX21_HUMAN	N
23	IL10_HUMAN	N	48	IL12A_HUMAN	N
24	CRUM1_HUMAN	Y	49	2B1G_HUMAN	N
25	TSLP_HUMAN	Y	50	PSPB_HUMAN	Y

## Conclusion

In summary, we propose a Dual Hypergraph Regularized Least Squares (DHRLS) based on CKA-MKL algorithm, for the gene-disease association prediction. We use multiple kernels to describe gene and disease spaces. The weights of these kernels are obtained by CKA-MKL and used to combine kernels. We use hypergraph to describe more complex information to improve our prediction. Our purpose is to establish an accurate and effective

prediction model of gene-disease association based on the existing data of gene-disease associations, and provide guidance for the follow-up verification methods of complex diseases.

## Methods

In this study, we first use two disease kernels and four gene kernels to reveal potential associations of genes and diseases. Then, the MKL method CKA is used to combine

**Table 5** Statistics of six real-world networks

Network	V	W	E	LD	AD	LAD	RAD
GPC	95	223	635	0.0300	2.00	6.68	2.85
Enzymes	664	445	2926	0.0099	2.64	4.41	6.58
Ionchannel	210	204	1476	0.0345	3.57	7.03	7.24
Drug-target	200	150	454	0.0151	1.30	2.27	3.03
SW	18	14	89	0.3532	2.78	4.94	6.36
CO	144	151	12170	0.5597	41.25	84.51	80.60

|V|, |W| denote the number of two types of nodes respectively; |E| is the number of edges; LD, AD, LAD, and RAD are the link density, the average degree, the left average degree, the right average degree.

**Table 6** Precision by different methods on six real networks

Method	GPC	Enzymes	Ionchannel	Drug-target	SW	CO
DHRLS	0.43	0.73	0.74	0.75	0.26	0.94
SRNMF-CN	0.41	0.69	0.69	0.74	0.20	0.94
SRNMF-AA	0.43	0.69	0.69	0.74	0.22	0.92
SRNMF-JC	0.43	0.69	0.69	0.74	0.23	0.93
SRNMF-CAA	0.42	0.69	0.69	0.74	0.20	0.93
SRNMF-CJC	0.42	0.69	0.69	0.73	0.22	0.94

Comparison refers to the reference [44].

above kernels into one disease kernel and one gene kernel. Finally, we use Dual Hypergraph Regularized Least Squares to identify gene-disease associations. Figure 7 show the flowchart of our method DHRLS.

#### Problem definition

The prediction of gene-disease associations can be regarded as a recommendation system. Given  $n$  diseases  $D = \{d_1, d_2, \dots, d_n\}$ ,  $m$  genes  $S = \{g_1, g_2, \dots, g_m\}$  and gene-disease associations. The association between gene and disease items can be expressed as an adjacent matrix  $\mathbf{Y} \in \mathbb{R}^{n \times m}$ . The element of adjacent matrix  $\mathbf{Y}$  is the relationship between genes and diseases. If disease  $d_j (1 \leq j \leq m)$  is associated with gene  $g_i (1 \leq i \leq n)$ , the value of  $\mathbf{Y}_{i,j}$  is set as 1, otherwise it is 0. Genes, diseases, and their associations are formulated as a bipartite network.

#### Related work

LapRLS framework [28] is often used in various fields based on machine learning model, such as the prediction of Human Microbe-Disease Association [29] and the detection of human microRNA-disease association [30]. At the same time, Hypergraph learning [31–33] is becoming popular. Hypergraphs can represent more complex relationships among various objects.

The Laplacian Regularized Least Squares (LapRLS) model [45] based on graph regularization is employed to predict potential associations in a bipartite network. The

functions of model can be defined as follows:

$$\begin{aligned}\mathbf{F}_a^* &= \arg \min J(\mathbf{F}_a) = \|\mathbf{Y}_{train} - \mathbf{F}_a\|_F^2 + \lambda_a tr(\mathbf{F}_a^T \mathbf{L}_a \mathbf{F}_a) \\ \mathbf{F}_b^* &= \arg \min J(\mathbf{F}_b) = \|\mathbf{Y}_{train} - \mathbf{F}_b\|_F^2 + \lambda_b tr(\mathbf{F}_b^T \mathbf{L}_b \mathbf{F}_b)\end{aligned}\quad (2)$$

where  $\mathbf{F}_a^* = \mathbf{K}_a \boldsymbol{\alpha}_a^*$ ,  $\mathbf{F}_b^* = \mathbf{K}_b \boldsymbol{\alpha}_b^*$ , and  $\mathbf{F}_a$ ,  $\boldsymbol{\alpha}_a^*$ ,  $\mathbf{F}_b^T$ ,  $\boldsymbol{\alpha}_b^{*T}$ ,  $\mathbf{Y}_{train} \in \mathbb{R}^{n \times m}$ .  $\mathbf{K}_a \in \mathbb{R}^{n \times n}$  and  $\mathbf{K}_b \in \mathbb{R}^{m \times m}$  are kernels in two feature space, separately.

$\mathbf{L}_a \in \mathbb{R}^{n \times n}$  and  $\mathbf{L}_b \in \mathbb{R}^{m \times m}$  are the normalized Laplacian matrices as follows:

$$\begin{aligned}\mathbf{L}_a &= \mathbf{D}_a^{-1/2} \Delta_a \mathbf{D}_a^{1/2}, \Delta_a = \mathbf{D}_a - \mathbf{K}_a \\ \mathbf{L}_b &= \mathbf{D}_b^{-1/2} \Delta_b \mathbf{D}_b^{1/2}, \Delta_b = \mathbf{D}_b - \mathbf{K}_b\end{aligned}\quad (3)$$

where  $\mathbf{D}_a$  and  $\mathbf{D}_b$  are diagonal matrices,  $\mathbf{D}_a(k, k) = \sum_{l=1}^n \mathbf{K}_a(k, l)$ ,  $\mathbf{D}_b(k, k) = \sum_{l=1}^m \mathbf{K}_b(k, l)$

The variables  $\boldsymbol{\alpha}_a$  and  $\boldsymbol{\alpha}_b^*$  of LapRLS can be solved as follows:

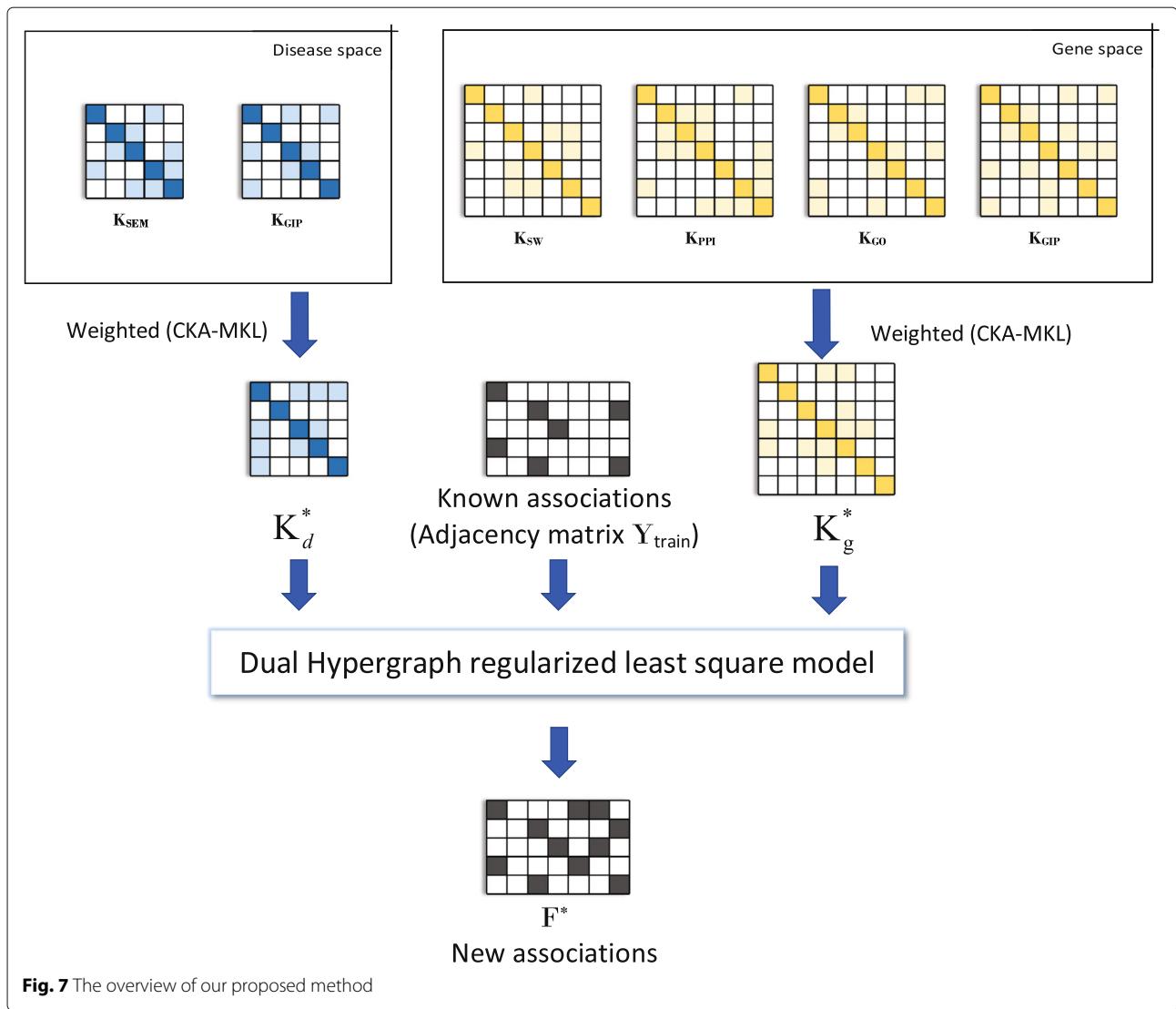
$$\begin{aligned}\boldsymbol{\alpha}_a^* &= (\mathbf{K}_a + \lambda_a \mathbf{L}_a \mathbf{K}_a)^{-1} \mathbf{Y}_{train} \\ \boldsymbol{\alpha}_b^* &= (\mathbf{K}_b + \lambda_b \mathbf{L}_b \mathbf{K}_b)^{-1} (\mathbf{Y}_{train})^T\end{aligned}\quad (4)$$

And  $\mathbf{F}_a^*$  and  $\mathbf{F}_b^*$  can be calculated as follows:

**Table 7** AUC by different methods on six real networks

Method	GPC	Enzymes	Ionchannel	Drug-target	SW	CO
DHRLS	0.89	0.96	0.97	0.98	0.85	1.00
SRNMF-CN	0.84	0.88	0.94	0.93	0.83	1.00
SRNMF-AA	0.83	0.88	0.94	0.93	0.82	1.00
SRNMF-JC	0.83	0.88	0.93	0.92	0.85	1.00
SRNMF-CAA	0.83	0.87	0.94	0.92	0.82	1.00
SRNMF-CJC	0.83	0.87	0.95	0.93	0.80	1.00

Comparison refers to the reference [44].

**Fig. 7** The overview of our proposed method

$$\begin{aligned} \mathbf{F}_a^* &= \mathbf{K}_a(\mathbf{K}_a + \lambda_a \mathbf{L}_a \mathbf{K}_a)^{-1} \mathbf{Y}_{train} \\ \mathbf{F}_b^* &= \mathbf{K}_b(\mathbf{K}_b + \lambda_b \mathbf{L}_b \mathbf{K}_b)^{-1} (\mathbf{Y}_{train})^T \end{aligned} \quad (5)$$

The predictions from two feature spaces are combined into:

$$\mathbf{F}^* = \frac{\mathbf{F}_a^* + (\mathbf{F}_b^*)^T}{2} \quad (6)$$

#### Feature extraction

To improve effectiveness of detecting gene-disease associations, We use two and four types of similarity for disease and gene separately. In our work, we constructed the multiple kernels of diseases and genes to represent the feature sets. Table 8 summarizes whole kernels, including two feature spaces.

#### Disease space

We calculate two classes of disease kernels, including semantic similarity kernel and Gaussian Interaction Profile (GIP) kernel (for disease).

**a) Semantic similarity** The disease semantic similarity kernel is calculated by the relative positions in the MeSH [46] disease. Directed Acyclic Graph (DAG) [47] can describe disease  $d_i$  as a node. A disease  $d_i$  can be described as a node in DAG and denoted as  $DAG_{d_i} = (d_i, T_{d_i}, E_{d_i})$ , where  $T_{d_i}$  is the set of all ancestor nodes of  $d_i$  including node  $d_i$  itself and  $E_{d_i}$  is the set of corresponding links. A semantic score of each disease  $t \in T_{d_i}$  can be calculated as follows:

$$D_{d_i}(t) = \begin{cases} 1 & \text{if } t = d_i \\ \max\{\Delta * D_{d_i}(t') | t' \in \text{children of } t\} & \text{if } t \neq d_i \end{cases} \quad (7)$$

**Table 8** Summary of kernels in two feature spaces

Space	Kernel	Description
Disease	$K_{SEM}^d$	Semantic similarity for disease
	$K_{GIP}^d$	Gaussian interaction profile for disease
	$K_{GO}^g$	Functional information of gene
Gene	$K_{PPI}^g$	Protein-protein interactions(PPIs) network of gene
	$K_{SW}^g$	Sequence information of gene
	$K_{GIP}^g$	Gaussian interaction profile for gene

where  $\Delta$  is the semantic contribution factor, which is set to 0.5 in this paper.

Then, the semantic score of disease  $d_i$  can be calculated as follows:

$$DV(d_i) = \sum_{t \in T_{d_i}} D_{d_i}(t) \quad (8)$$

So, the disease semantic similarity kernel  $K_{SEM}^d \in \mathbf{R}^{n \times n}$  is calculated as follows:

$$K_{SEM}^d(d_i, d_j) = \frac{\sum_{t \in T_{d_i} \cap T_{d_j}} (D_{d_i}(t) + D_{d_j}(t))}{DV(d_i) + DV(d_j)} \quad (9)$$

**b) GIP kernel similarity** The similarity between diseases can also be calculated by GIP. Given two diseases  $d_i$  and  $d_j$  ( $i, j = 1, 2, \dots, n$ ), the GIP kernel can be calculated as follows:

$$K_{GIP}^d(d_i, d_j) = \exp(-\gamma_d \|\mathbf{Y}_{d_i} - \mathbf{Y}_{d_j}\|^2) \quad (10)$$

where  $\mathbf{Y}_{d_i}$  and  $\mathbf{Y}_{d_j}$  are the information of associations for vector disease  $d_i$  and  $d_j$ .  $\gamma_d$  (set as 0.5) is the bandwidth of GIP kernel.

#### Gene space

Four types of gene kernels, including Gene Ontology (GO) [48] similarity, Protein-protein interactions (PPIs) network similarity, sequence similarity kernel and GIP kernel (for gene) are utilized to represent the relationship between genes.

**a) GO similarity** The information of GO is obtained through DAVID [49]. GO similarity ( $K_{GO}^g \in \mathbf{R}^{m \times m}$ ) is the overlap of GO annotations on two genes, and we simply use GOSemSim [50] to get it. We consider one option of GO: cellular component (CC) to represent gene functional annotation.

**b) PPIs similarity** We download the protein-protein interactions network from previous research [41] and select the sub-networks related to our genes. Give the topological feature vectors  $p_i$  and  $p_j$  of two genes in the

PPIs network. The Cosine similarity of PPIs network can be calculated as follows:

$$K_{PPI}^g(p_i, p_j) = \frac{p_i \cdot p_j}{\|p_i\| \|p_j\|} \quad (11)$$

**c) Sequence similarity** We use the normalized Smith Waterman (SW) score [51] to measure the sequence similarity between the two gene sequences, which is calculated as follows:

$$K_{SW}^g(g_i, g_j) = \frac{SW(S_{g_i}, S_{g_j})}{\sqrt{SW(S_{g_i}, S_{g_i})} \sqrt{SW(S_{g_j}, S_{g_j})}} \quad (12)$$

where  $SW(., .)$  is Smith Waterman score.  $S_{g_i}$  is the information of sequence for gene  $g_i$ .

**d) GIP kernel similarity** GIP is also employed to build gene GIP kernel ( $K_{GIP}^g$ ). Given two genes  $g_i$  and  $g_j$  ( $i, j = 1, 2, \dots, m$ ), the GIP kernel can be calculated as follows:

$$K_{GIP}^g(g_i, g_j) = \exp(-\gamma_g \|\mathbf{Y}_{g_i} - \mathbf{Y}_{g_j}\|^2) \quad (13)$$

where  $\mathbf{Y}_{g_i}$  and  $\mathbf{Y}_{g_j}$  are the information of associations for vector gene  $g_i$  and  $g_j$ .  $\gamma_g$  (set as 0.5) is the bandwidth of GIP kernel.

#### Multiple kernel learning

In our work, two kernels in the disease space including  $K_{SEM}^d$  and  $K_{GIP}^d$ , and four kernels of gene space including  $K_{SW}^g$ ,  $K_{GO}^g$ ,  $K_{PPI}^g$  and  $K_{GIP}^g$ . We then need to combine these kernels by means of linear combination in order to achieve the optimal ones.

$$\begin{aligned} \mathbf{K}^* &= \sum_{i=1}^k \omega_i \mathbf{K}_i \\ \mathbf{K}_i &\in \mathbf{R}^{N \times N} \\ \sum_{i=1}^k \omega_i &= 1 \end{aligned} \quad (14)$$

where  $k$  is the number of kernels and  $\omega_i$  is the weight of the kernel  $\mathbf{K}_i$ .  $N$  is the number of samples in kernel  $\mathbf{K}_i$ .

The method CKA-MKL is utilized to combine gene kernels and disease kernels, respectively. The cosine similarity between  $\mathbf{K}_1$  and  $\mathbf{K}_2$  is defined as follows:

$$CA(\mathbf{K}_1, \mathbf{K}_2) = \frac{\langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F}{\|\mathbf{K}_1\|_F \|\mathbf{K}_2\|_F} \quad (15)$$

where  $\mathbf{K}_1, \mathbf{K}_2 \in \mathbb{R}^{n \times n}$ ,  $\langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F = \text{Trace}(\mathbf{K}_1^T \mathbf{K}_2)$  is the Frobenius inner product and  $\|\mathbf{K}_1\|_F = \sqrt{\langle \mathbf{K}_1, \mathbf{K}_1 \rangle_F}$  is Frobenius norm.

The higher the cosine value, the greater the similarity between the kernels. CKA is based on the assumption that the combined kernel (feature space) should be similar to the ideal kernel (label space). Therefore, the alignment score between the combined kernel and the ideal kernel should be maximized. The objective function of centered kernel alignment is as follows:

$$\begin{aligned} & \max_{\omega \geq 0} CA(\mathbf{K}_\omega, \mathbf{K}_{ideal}) \\ &= \max_{\omega \geq 0} \frac{\langle \mathbf{K}_\omega^c, \mathbf{K}_{ideal} \rangle_F}{\|\mathbf{K}_\omega^c\|_F \|\mathbf{K}_{ideal}\|_F} \\ & \text{subject to } \mathbf{K}_\omega = \sum_{i=1}^k \omega_i \mathbf{K}_i \\ & \mathbf{K}_\omega^c = \mathbf{U}_N \mathbf{K}_\omega \mathbf{U}_N \end{aligned} \quad (16)$$

where  $\sum_{i=1}^k \omega_i = 1$ ,  $\omega_i \geq 0$ ,  $i = 1, 2, \dots, k$ .  $\mathbf{U}_N = \mathbf{I}_N - (1/N)\mathbf{I}_N\mathbf{I}_N^T$  denotes a centering matrix, and  $\mathbf{I}_N \in \mathbb{R}^{N \times N}$  is the N-order identity matrix,  $\mathbf{I}_N$  is the N-order vector with all entries equal to one.  $\mathbf{K}_\omega^c$  is the centered kernel matrix associated with  $\mathbf{K}_\omega$ . Equation 16 can be written as follow:

$$\min_{\omega \geq 0} \omega^T \mathbf{M} \omega - 2\omega^T \mathbf{a} \quad (17)$$

where  $\mathbf{a} = (\langle \mathbf{K}_1^c, \mathbf{K}_{ideal} \rangle_F, \langle \mathbf{K}_2^c, \mathbf{K}_{ideal} \rangle_F, \dots, \langle \mathbf{K}_k^c, \mathbf{K}_{ideal} \rangle_F)^T \in \mathbb{R}^{k \times 1}$  and  $\mathbf{M}$  denotes the matrix defined by  $M_{ij} = \langle \mathbf{K}_i^c, \mathbf{K}_j^c \rangle_F$ , for  $i, j = 1, \dots, k$ . We can obtain the weight ( $\omega$ ) by solving this simple quadratic programming problem.

CKA-MKL estimates the weights of  $w_d \in \mathbb{R}^{k_d \times 1}$ ,  $w_g \in \mathbb{R}^{k_g \times 1}$ , to combine disease  $(\mathbf{K}_{SEM}^d, \mathbf{K}_{GIP}^d \in \mathbb{R}^{n \times n})$  and gene  $(\mathbf{K}_{SW}^g, \mathbf{K}_{GO}^g, \mathbf{K}_{PPI}^g, \mathbf{K}_{GIP}^g \in \mathbb{R}^{m \times m})$  kernels, separately.  $k_d$  and  $k_g$  are the number of kernels in disease space and gene space. In order to obtain the optimal kernel matrix  $\mathbf{K}_d^*$  and  $\mathbf{K}_g^*$  in the two spaces, first calculate the weights of kernel matrices in each space by Eq. 17, and then combine them by Eq. 14. Here,  $\mathbf{K}_{ideal}^d = \mathbf{Y}_{train} \mathbf{Y}_{train}^T \in \mathbb{R}^{n \times n}$  in the disease space; and  $\mathbf{K}_{ideal}^g = \mathbf{Y}_{train}^T \mathbf{Y}_{train} \in \mathbb{R}^{m \times m}$  in the gene space.

### Hypergraph learning

In graph theory, a graph represents the pairwise relationship between a group of objects. In traditional graph structures, vertices represent objects, and edges represent

relationships between two objects. However, traditional graph structures cannot express more complex relationships. For example, they cannot express more than three relationships in pairs. Hypergraph [31] solves this problem well. In hypergraph theory, this kind of multi-object relationship is represented by using a subset of vertex sets as super edges. In this study, we use hypergraph to establish this higher-order relationship. In Fig. 8 (left),  $\{v_1, v_2, \dots, v_7\}$  represents the vertex set, and  $\{v_2, v_4, v_6\}$  are contained in hyperedge  $e_1$ . Each hyperedge may comprise two or more vertices. The hyperedge will degenerate into a normal edge, when there are only two vertices in the hyperedge.

The construction of hypergraph is similar to that of ordinary graph. Hypergraph also needs a vertex set  $\mathbf{V}$ , an hyperedge set  $\mathbf{E}$  and the weight of hyperedge  $\mathbf{w} \in \mathbb{R}^{N_e \times 1}$ . Here, each hyperedge  $e_i (i = 1, 2, \dots, N_e)$  is given a weight  $w(e_i)$ . The difference is that the hyperedge set of a hypergraph is actually a set of vertices. Therefore, a hypergraph can be represented by  $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathbf{w})$ .

For the hypergraph  $\mathbf{G}$ , the incidence matrix  $\mathbf{H}$  conveys the affinity between vertices and hyperedges. And, each element of  $\mathbf{H}$  can be given by the following formula:

$$H(v, e) = \begin{cases} 1 & \text{if } v \in e \\ 0 & \text{if } v \notin e \end{cases} \quad (18)$$

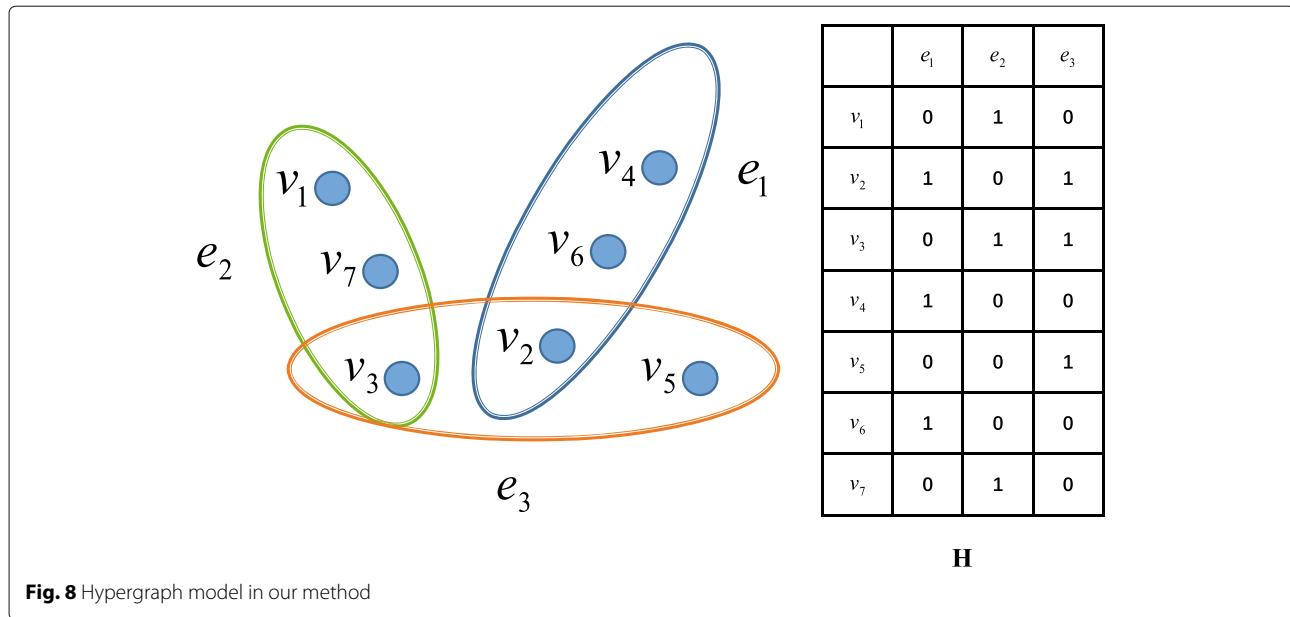
The matrix  $\mathbf{H}$  describes the relationship between vertices and is shown in Fig. 2 (right). Specifically,  $H_{i,j} = 1$  means the vertex  $v_i$  is included in the hyperedge  $e_j$ . On the contrary,  $H_{i,j} = 0$  means that the vertex  $v_i$  is not in the hyperedge  $e_j$

In a hypergraph  $\mathbf{G}$ . The degree of each vertex and hyperedge and the weight of hpyperdege are expressed as follows:

$$\begin{aligned} d(v) &= \sum_{e \in \mathbf{E}} H(v, e) \\ \delta(e) &= \sum_{v \in \mathbf{V}} H(v, e) \\ w(e_j) &= \sum_{i=1}^k K^*(v_i, v_j) \end{aligned} \quad (19)$$

where  $K^*$  is the combined kernel.

The hypergraph is constructed using the  $k$  Nearest Neighbor (kNN) algorithm. Specifically, each vertex as the center point, and find the  $k$  vertices with the largest similarity according to the kernel matrix to form a hyperedge. Assuming that there are  $N$  samples, we can construct  $N$  hyperedges. In this study, we define the weight of each hyperedge is the sum of kernel values of the  $k$  vertices closest to center point, and finally the weight is normalized.



**Fig. 8** Hypergraph model in our method

Then, we compute three matrices  $\mathbf{D}_v$ ,  $\mathbf{D}_e$  and  $\mathbf{D}_w$ , where  $\mathbf{D}_v$  and  $\mathbf{D}_e$  are the diagonal matrices of  $d(v)$  and  $d(e)$ .  $\mathbf{D}_w$  is the matrix of hyperedge weights.

$$\begin{aligned}\mathbf{D}_v &= \text{diag}(d) \\ \mathbf{D}_e &= \text{diag}(\delta) \\ \mathbf{D}_w &= \text{diag}(w)\end{aligned}\quad (20)$$

The hypergraph Laplacian matrix  $\mathbf{L}^h$  [31] is defined as follows:

$$\begin{aligned}\mathbf{L}^h &= \mathbf{I} - \Theta \\ \Theta &= \mathbf{D}_v^{-1/2} \mathbf{H} \mathbf{D}_w \mathbf{D}_e^{-1} \mathbf{H}^T \mathbf{D}_v^{-1/2}\end{aligned}\quad (21)$$

where  $\mathbf{I}$  is the identity matrix.

Consequently, we can obtain the hypergraph Laplacian matrix  $\mathbf{L}_d^h$  and  $\mathbf{L}_g^h$  about the disease and gene spaces, respectively.

#### Dual hypergraph regularized least squares

Based on LapRLS method, we propose a novel model to predict the associations of genes and diseases, named Dual Hypergraph Regularized Least Squares (DHRLS), through incorporation of the multiple informations of gene and disease feature spaces into the dual hypergraph regularized least squares framework. The objective function can be written as follow:

$$\min E(\mathbf{F}^*) = \|\mathbf{K}_d^* \boldsymbol{\alpha}_d + (\mathbf{K}_g^* \boldsymbol{\alpha}_g)^T - 2\mathbf{Y}_{train}\|_F^2 \quad (22)$$

where  $\mathbf{F}_d^* = \mathbf{K}_d^* \boldsymbol{\alpha}_d$  and  $\mathbf{F}_g^* = \mathbf{K}_g^* \boldsymbol{\alpha}_g$ . The  $\mathbf{F}^*$  could be calculated by  $\mathbf{F}^* = (\mathbf{F}_d^* + (\mathbf{F}_g^*)^T)/2$ .  $\mathbf{F}^*$  is an average combination of gene and disease space evaluation as the final prediction result.

Then to avoid overfitting of  $\boldsymbol{\alpha}_d$  and  $\boldsymbol{\alpha}_g$  to training data, we apply L2 (Tikhonov) regularization to Eq. 22 by adding two terms regarding  $\boldsymbol{\alpha}_d$  and  $\boldsymbol{\alpha}_g$ .

$$\begin{aligned}\min E(\mathbf{F}^*) &= \|\mathbf{K}_d^* \boldsymbol{\alpha}_d + (\mathbf{K}_g^* \boldsymbol{\alpha}_g)^T - 2\mathbf{Y}_{train}\|_F^2 \\ &\quad + \beta (\|\boldsymbol{\alpha}_d\|_F^2 + \|\boldsymbol{\alpha}_g\|_F^2)\end{aligned}\quad (23)$$

where  $\beta$  is a regularization coefficient.

Since previous studies [52] have shown that graph regularization terms are beneficial to improve the prediction effect of the model, graph regularization terms related to genes and diseases are added to the model. According to the local invariance assumption [53], if two data points are close in the intrinsic geometry of the data distribution, then the representations of these two points with respect to the new basis, are also close to each other. This assumption plays an essential role in the development of various kinds of algorithms. In our model, we minimize the distance between the potential feature vectors of two adjacent diseases and genes respectively

$$\begin{aligned}&\min_{\boldsymbol{\alpha}_d} \sum_{i,r}^n \mathbf{K}_d^*(i,r) \|\mathbf{F}_d^{*i} - \mathbf{F}_d^{*r}\|^2 \\ &\quad = \text{tr} (\boldsymbol{\alpha}_d^T \mathbf{K}_d^* \mathbf{L}_d \mathbf{K}_d^* \boldsymbol{\alpha}_d) \\ &\min_{\boldsymbol{\alpha}_g} \sum_{j,q}^m \mathbf{K}_g^*(j,q) \|\mathbf{F}_g^{*j} - \mathbf{F}_g^{*q}\|^2 \\ &\quad = \text{tr} (\boldsymbol{\alpha}_g^T \mathbf{K}_g^* \mathbf{L}_g \mathbf{K}_g^* \boldsymbol{\alpha}_g)\end{aligned}\quad (24)$$

where  $\mathbf{F}_d^{*i}$  is the  $i$ -th row vector of  $\mathbf{F}_d^* = \mathbf{K}_d^* \boldsymbol{\alpha}_d \in \mathbf{R}^{n \times m}$ ,  $i, r = 1, 2, \dots, n$ . Similarly,  $\mathbf{F}_g^{*j}$  is the  $j$ -th row vector of

**Table 9** The algorithm of our proposed method

Algorithm : The algorithm of our proposed method

**Input:** Known associations  $\mathbf{Y}_{train} \in \mathbb{R}^{n \times m}$ , disease space kernels ( $\mathbf{K}_{SEM}^d, \mathbf{K}_{GIP}^d \in \mathbb{R}^{n \times n}$ ) and gene space kernels ( $\mathbf{K}_{GO}^g, \mathbf{K}_{PPI}^g, \mathbf{K}_{SW}^g, \mathbf{K}_{GIP}^g \in \mathbb{R}^{m \times m}$ ), parameters  $\lambda_d, \lambda_g, \beta$  and k-Nearest Neighbor for DHRLS;

**Output:** Predicted associations  $\mathbf{F}^* \in \mathbb{R}^{n \times m}$ ;

- 1.Calculating disease and gene kernels, listed in Table 8;
- 2.Calculating disease kernel weights  $w_d$  and gene kernel weights  $w_g$  by Eq. 17 (CKA-MKL), respectively;
- 3.Calculating  $\mathbf{K}_d^*$  and  $\mathbf{K}_g^*$  by Eq. 14, respectively;
- 4.Calculating  $\mathbf{L}_d^h$  and  $\mathbf{L}_g^h$  by Eq. 21, respectively;
- 5.Solving Eqs. 27 and 28 (ALSA), and estimating  $\mathbf{F}^*$  by Eq. 29;

$\mathbf{F}_g^* = \mathbf{K}_g^* \boldsymbol{\alpha}_g \in \mathbb{R}^{m \times n}, j, q = 1, 2, \dots, m$ .  $\mathbf{F}_d^{*i}$  and  $\mathbf{F}_g^{*j}$  mean the representations of the new base.  $\mathbf{K}_d^*(i, r)$  and  $\mathbf{K}_g^*(j, q)$  are the weights of two points in two spaces respectively. After adding the graph regular term, the objective function is redefined as follows:

$$\begin{aligned} \min E(\mathbf{F}^*) = & ||\mathbf{K}_d^* \boldsymbol{\alpha}_d + (\mathbf{K}_g^* \boldsymbol{\alpha}_g)^T - 2\mathbf{Y}_{train}||_F^2 \\ & + \lambda_d tr(\boldsymbol{\alpha}_d^T \mathbf{K}_d^* \mathbf{L}_d \mathbf{K}_d^* \boldsymbol{\alpha}_d) \\ & + \lambda_g tr(\boldsymbol{\alpha}_g^T \mathbf{K}_g^* \mathbf{L}_g \mathbf{K}_g^* \boldsymbol{\alpha}_g) \\ & + \beta (||\boldsymbol{\alpha}_d||_F^2 + ||\boldsymbol{\alpha}_g||_F^2) \end{aligned} \quad (25)$$

where  $\lambda_d$  and  $\lambda_g$  are the coefficients of graph regular terms.

We take formula 25 as a model, called Dual Graph Regularized Least Squares (DGRLS). In order to be able to express the high-order relationship between nodes, while improving the prediction effect, Hypergraph Laplacian matrix is applied to our final model DHRLS. Thus, the final objective function can be described as follows:

$$\begin{aligned} \min E(\mathbf{F}^*) = & ||\mathbf{K}_d^* \boldsymbol{\alpha}_d + (\mathbf{K}_g^* \boldsymbol{\alpha}_g)^T - 2\mathbf{Y}_{train}||_F^2 \\ & + \lambda_d tr(\boldsymbol{\alpha}_d^T \mathbf{K}_d^* \mathbf{L}_d^h \mathbf{K}_d^* \boldsymbol{\alpha}_d) \\ & + \lambda_g tr(\boldsymbol{\alpha}_g^T \mathbf{K}_g^* \mathbf{L}_g^h \mathbf{K}_g^* \boldsymbol{\alpha}_g) \\ & + \beta (||\boldsymbol{\alpha}_d||_F^2 + ||\boldsymbol{\alpha}_g||_F^2) \end{aligned} \quad (26)$$

where  $\mathbf{L}^h$  is the hypergraph laplacian matrix, it can be calculated by Eq. 21.

### Objective function optimization for DHRLS

We select alternating least squares to estimate  $\boldsymbol{\alpha}_d$  and  $\boldsymbol{\alpha}_g$ , and then run alternately until convergence.

**Optimizing  $\boldsymbol{\alpha}_d$**  Suppose  $\boldsymbol{\alpha}_g$  are known, to obtain the optimal  $\boldsymbol{\alpha}_d$  by setting  $\partial E(\mathbf{F}^*)/\partial \boldsymbol{\alpha}_d = 0$ :

$$\begin{aligned} \frac{\partial E(\mathbf{F}^*)}{\boldsymbol{\alpha}_d} = 0 \\ \mathbf{K}_d^* \left( \mathbf{K}_d^* \boldsymbol{\alpha}_d + \boldsymbol{\alpha}_g^T (\mathbf{K}_g^*)^T - 2\mathbf{Y}_{train} \right) + \beta \boldsymbol{\alpha}_d + \lambda_d \mathbf{K}_d^* \mathbf{L}_d^h \mathbf{K}_d^* \boldsymbol{\alpha}_d = 0 \\ \left( \mathbf{K}_d^* \mathbf{K}_d^* + \beta \mathbf{I} + \lambda_d \mathbf{K}_d^* \mathbf{L}_d^h \mathbf{K}_d^* \right) \boldsymbol{\alpha}_d = 2\mathbf{K}_d^* \mathbf{Y}_{train} - \mathbf{K}_d^* \boldsymbol{\alpha}_g^T \mathbf{K}_g^* \\ \boldsymbol{\alpha}_d = \left( \mathbf{K}_d^* \mathbf{K}_d^* + \beta \mathbf{I} + \lambda_d \mathbf{K}_d^* \mathbf{L}_d^h \mathbf{K}_d^* \right)^{-1} \left( 2\mathbf{K}_d^* \mathbf{Y}_{train} - \mathbf{K}_d^* \boldsymbol{\alpha}_g^T \mathbf{K}_g^* \right) \end{aligned} \quad (27)$$

**Optimizing  $\boldsymbol{\alpha}_g$**  Similarly, suppose  $\boldsymbol{\alpha}_d$  are known, to obtain the optimal  $\boldsymbol{\alpha}_g$  by setting  $\partial E(\mathbf{F}^*)/\partial \boldsymbol{\alpha}_g = 0$ :

$$\begin{aligned} \frac{\partial E(\mathbf{F}^*)}{\boldsymbol{\alpha}_g} = 0 \\ \mathbf{K}_g^* \left( \mathbf{K}_g^* \boldsymbol{\alpha}_g + \boldsymbol{\alpha}_d^T (\mathbf{K}_d^*)^T - 2\mathbf{Y}_{train}^T \right) + \beta \boldsymbol{\alpha}_g + \lambda_g \mathbf{K}_g^* \mathbf{L}_g^h \mathbf{K}_g^* \boldsymbol{\alpha}_g = 0 \\ \left( \mathbf{K}_g^* \mathbf{K}_g^* + \beta \mathbf{I} + \lambda_g \mathbf{K}_g^* \mathbf{L}_g^h \mathbf{K}_g^* \right) \boldsymbol{\alpha}_g = 2\mathbf{K}_g^* \mathbf{Y}_{train}^T - \mathbf{K}_g^* \boldsymbol{\alpha}_d^T \mathbf{K}_d^* \\ \boldsymbol{\alpha}_g = \left( \mathbf{K}_g^* \mathbf{K}_g^* + \beta \mathbf{I} + \lambda_g \mathbf{K}_g^* \mathbf{L}_g^h \mathbf{K}_g^* \right)^{-1} \left( 2\mathbf{K}_g^* \mathbf{Y}_{train}^T - \mathbf{K}_g^* \boldsymbol{\alpha}_d^T \mathbf{K}_d^* \right) \end{aligned} \quad (28)$$

The final prediction result is by combining the matrices in the two spaces:

$$\mathbf{F}^* = \frac{\mathbf{K}_d^* \boldsymbol{\alpha}_d + (\mathbf{K}_g^* \boldsymbol{\alpha}_g)^T}{2} \quad (29)$$

The overview of our method is shown in Table 9.

### Abbreviations

DHRLS:Dual Hypergraph Regularized Least Squares; DGRLS:Dual Graph Regularized Least Squares; CKA:Centered Kernel Alignment; MKL:Multiple Kernel Learning; ALSA:Alternating Least squares algorithm; MKSVM:Multiple Kernel Support Vector Machine; CV:cross-validation; AUPR: Area Under the Precision-Recall curve; AUC:Area under the receiver operating characteristic curve; GIP:Gaussian Interaction Profile; kNN:k Nearest Neighbor; LapRLS:Laplacian Regularized Least Squares; PPIs:Protein-protein interactions; GO:Gene Ontology; CC:cellular component; SW:Smith Waterman;

### Acknowledgements

No application.

## Authors' contributions

HY conceived and designed the experiments; YD performed the experiments and analyzed the data; FG wrote the paper; JT reviewed the manuscript. All authors have read and approved the whole manuscript.

## Funding

This work is supported by a grant from the National Natural Science Foundation of China (NSFC 61772362, 61902271 and 61972280), and National Key R&D Program of China (2020YFA0908400), and the Natural Science Research of Jiangsu Higher Education Institutions of China (19KJB520014).

## Availability of data and materials

The datasets, codes and corresponding results of our model are available at <https://github.com/guofei-tju/DHRLS>. Associations for all genes associated with the 960 diseases are available at <http://cssb2.biology.gatech.edu/knowgene>. Six types of real networks are published at [44].

## Declarations

### Ethics approval and consent to participate

No application.

### Consent for publication

No application.

### Competing interests

The authors declare that they have no competing interests.

### Author details

<sup>1</sup>School of Computer Science and Technology, College of Intelligence and Computing, Tianjin University, Tianjin, China. <sup>2</sup>Yangtze Delta Region Institute, University of Electronic Science and Technology of China, Quzhou, China. <sup>3</sup>Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. <sup>4</sup>School of Computer Science and Engineering, Central South University, Changsha, China.

Received: 9 April 2021 Accepted: 29 June 2021

Published online: 09 August 2021

## References

- Wei L, Liao M, Gao Y, Ji R, He Z, Zou Q. Improved and promising identification of human micrornas by incorporating a high-quality negative set. *IEEE/ACM Trans Compu Biol Bioinforma*. 2013;11(1):192–201.
- Wei L, Su R, Wang B, Li X, Zou Q, Gao X. Integration of deep feature representations and handcrafted features to improve the prediction of n6-methyladenosine sites. *Neurocomputing*. 2019;324:3–9.
- Liu H, Ren G, Chen H, Liu Q, Yang Y, Zhao Q. Predicting lncrna–miRNA interactions based on logistic matrix factorization with neighborhood regularized. *Knowl-Based Syst*. 2020;191:105261.
- Wang J, Wang H, Wang X, Chang H. Predicting drug-target interactions via fm-dhn learning. *Curr Bioinforma*. 2020;15(1):68–76.
- Huang Y, Yuan K, Tang M, Yue J, Bao L, Wu S, Zhang Y, Li Y, Wang Y, Ou X, et al. Melatonin inhibiting the survival of human gastric cancer cells under er stress involving autophagy and ras-raf-mapk signalling. *J Cell Mol Med*. 2021;25(3):1480–92.
- Xiao Y, Wu J, Lin Z, Zhao X. A deep learning-based multi-model ensemble method for cancer prediction. *Comput Methods Prog Biomed*. 2018;153:1–9.
- Mordelet F, Vert J-P. Prodigy: Prioritization of disease genes with multitask machine learning from positive and unlabeled examples. *BMC Bioinforma*. 2011;12(1):389.
- Yu S, Falck T, Daemen A, Tranchevent L-C, Suykens J, De Moor B, Moreau Y. L 2-norm multiple kernel learning and its application to biomedical data fusion. *BMC Bioinforma*. 2010;11(1):309.
- Deo R, Musso G, Tasan M, Tang P, Poon A, Yuan C, Felix J, Vasan R, Beroukhim R, De Marco T, et al. Prioritizing causal disease genes using unbiased genomic features. *Genome Biol*. 2014;15(12):534.
- Yang P, Li X-L, Mei J-P, Kwok C-K, Ng S-K. Positive-unlabeled learning for disease gene identification. *Bioinforma*. 2012;28(20):2640–7.
- Natarajan N, Dhillon I. Inductive matrix completion for predicting gene–disease associations. *Bioinforma*. 2014;30(12):60–8.
- Zakeri P, Simm J, Arany A, ElShal S, Moreau Y. Gene prioritization using bayesian matrix factorization with genomic and phenotypic side information. *Bioinforma*. 2018;34(13):447–56.
- Zeng X, Ding N, Rodríguez-Patón A, Zou Q. Probability-based collaborative filtering model for predicting gene–disease associations. *BMC Med Genet*. 2017;10(5):76.
- Singh-Blom U, Natarajan N, Tewari A, Woods J, Dhillon I, Marcotte E. Prediction and validation of gene-disease associations using methods inspired by social network analyses. *PLoS One*. 2013;8(5):e58977.
- Luo P, Ding Y, Lei X, Wu F-X. deepdriver: predicting cancer driver genes based on somatic mutations using deep convolutional neural networks. *Front Genet*. 2019;10:13.
- Rao A, Saipradeep V, Joseph T, Kotte S, Sivadasan N, Srinivasan R. Phenotype-driven gene prioritization for rare diseases using graph convolution on heterogeneous networks. *BMC Med Genet*. 2018;11(1):57.
- Li Y, Kuwahara H, Yang P, Song L, Gao X. Pgcn: Disease gene prioritization by disease and gene embedding through graph convolutional neural networks. *bioRxiv*. 2019;532226.
- Ding Y, Tang J, Guo F. Identification of drug–target interactions via fuzzy bipartite local model. *Neural Comput Appl*. 2019;1–17.
- Ding Y, Tang J, Guo F. Identification of drug–target interactions via dual laplacian regularized least squares with multiple kernel fusion. *Knowl-Based Syst*. 2020;204:106254.
- Ding Y, Tang J, Guo F. Identification of drug-side effect association via semisupervised model and multiple kernel learning. *IEEE J Biomed Health Inform*. 2018;23(6):2619–32.
- Ding Y, Tang J, Guo F. Identification of drug-side effect association via multiple information integration with centered kernel alignment. *Neurocomputing*. 2019;325:211–24.
- Wang H, Ding Y, Tang J, Guo F. Identification of membrane protein types via multivariate information fusion with hilbert–schmidt independence criterion. *Neurocomputing*. 2020;383:257–69.
- Shen Y, Tang J, Guo F. Identification of protein subcellular localization via integrating evolutionary and physicochemical information into chou's general pseaa. *J Theor Biol*. 2019;462:230–9.
- Ding Y, Tang J, Guo F. Human protein subcellular localization identification via fuzzy model on kernelized neighborhood representation. *Appl Soft Comput*. 2020;106596.
- Yi Zou, XGLPYDJT HongjieWu, Guo F. Mk-fsvm-svdd: A multiple kernel-based fuzzy svm model for predicting dna-binding proteins via support vector data description. *Curr Bioinforma*. 2020;16:274–83.
- Ding Y, Tang J, Guo F. Protein crystallization identification via fuzzy model on linear neighborhood representation. *IEEE/ACM Trans Comput Biol Bioinforma*. 2019;1.
- Zhang J, Zhang Z, Pu L, Tang J, Guo F. Aiepred: an ensemble predictive model of classifier chain to identify anti-inflammatory peptides. *IEEE/ACM Trans Comput Biol Bioinforma*. 2020;1.
- Belkin M, Niyogi P, Sindhwani V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *J Mach Learn Res*. 2006;7:2399–434.
- Wang F, Huang Z-A, Chen X, Zhu Z, Wen Z, Zhao J, Yan G-Y. Lrlshmda: Laplacian regularized least squares for human microbe–disease association prediction. *Sci Rep*. 2017;7(1):1–11.
- Jiang L, Xiao Y, Ding Y, Tang J, Guo F. Fkl-spa-laprls: an accurate method for identifying human microrna-disease association. *BMC Genomics*. 2018;19(10):11–25.
- Zhou D, Huang J, Schölkopf B. Learning with hypergraphs: Clustering, classification, and embedding. In: *Advances in Neural Information Processing Systems*. Cambridge: MIT Press; 2007. p. 1601–1608.
- Wu W, Kwong S, Zhou Y, Jia Y, Gao W. Nonnegative matrix factorization with mixed hypergraph regularization for community detection. *Inf Sci*. 2018;435:263–81.
- Xu X-X, Dai L-Y, Kong X-Z, Liu J-X. A low-rank representation method regularized by dual-hypergraph laplacian for selecting differentially expressed genes. *Hum Hered*. 2019;84(1):1–13.

34. Bai S, Zhang F, Torr P. Hypergraph convolution and hypergraph attention. *Pattern Recog.* 2021;110:107637.
35. Zhang R, Zou Y, Ma J. Hyper-sagnn: a self-attention based graph neural network for hypergraphs. arXiv preprint arXiv:1911.02613. 2019.
36. Ding Y, Jiang L, Tang J, Guo F. Identification of human microRNA-disease association via hypergraph embedded bipartite local model. *Comput Biol Chem.* 2020;89:107369.
37. Lü L, Zhou T. Link prediction in complex networks: A survey. *Physica A Stat Mech its Appl.* 2011;390(6):1150–70.
38. Holme P, Liljeros F, Edling C, Kim B. Network bipartivity. *Phys Rev E.* 2003;68(5):056107.
39. Kunegis J, De Luca E, Albayrak S. The link prediction problem in bipartite networks. In: International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems. Berlin Heidelberg: Springer; 2010. p. 380–9.
40. Lu Y, Wang L, Lu J, Yang J, Shen C. Multiple kernel clustering based on centered kernel alignment. *Pattern Recog.* 2014;47(11):3656–64.
41. Zhou H, Skolnick J. A knowledge-based approach for predicting gene–disease associations. *Bioinforma.* 2016;32(18):2831–8.
42. Ezzat A, Zhao P, Wu M, Li X-L, Kwoh C-K. Drug-target interaction prediction with graph regularized matrix factorization. *IEEE/ACM Trans Comput Biol Bioinforma.* 2016;14(3):646–56.
43. Zheng X, Ding H, Mamitsuka H, Zhu S. Collaborative matrix factorization with multiple similarities for predicting drug-target interactions. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM; 2013. p. 1025–33.
44. Wang W, Chen X, Jiao P, Jin D. Similarity-based regularized latent feature model for link prediction in bipartite networks. *Sci Rep.* 2017;7(1):1–12.
45. Xia Z, Wu L-Y, Zhou X, Wong S. Semi-supervised drug-protein interaction prediction from heterogeneous biological spaces. In: BMC Systems Biology. BioMed Central; 2010. p. 6.
46. Lowe H, Barnett G. Understanding and using the medical subject headings (mesh) vocabulary to perform literature searches. *Jama.* 1994;271(14):1103–8.
47. Wang D, Wang J, Lu M, Song F, Cui Q. Inferring the human microRNA functional similarity and functional network based on microRNA-associated diseases. *Bioinforma.* 2010;26(13):1644–50.
48. Ashburner M, Ball C, Blake J, Botstein D, Butler H, Cherry J, Davis A, Dolinski K, Dwight S, Eppig J, et al. Gene ontology: tool for the unification of biology. *Nat Genet.* 2000;25(1):25–9.
49. Sherman B, Lempicki R, et al. Systematic and integrative analysis of large gene lists using david bioinformatics resources. *Nat Protocol.* 2009;4(1):44.
50. Yu G, Li F, Qin Y, Bo X, Wu Y, Wang S. Gosemism: an r package for measuring semantic similarity among go terms and gene products. *Bioinforma.* 2010;26(7):976–8.
51. Yamanishi Y, Araki M, Gutteridge A, Honda W, Kanehisa M. Prediction of drug–target interaction networks from the integration of chemical and genomic spaces. *Bioinforma.* 2008;24(13):232–40.
52. Gu Q, Zhou J, Ding C. Collaborative filtering: Weighted nonnegative matrix factorization incorporating user and item graphs. In: Proceedings of the 2010 SIAM International Conference on Data Mining. Columbus: SIAM; 2010. p. 199–210.
53. Cai D, He X, Han J, Huang T. Graph regularized nonnegative matrix factorization for data representation. *IEEE Trans Pattern Anal Mach Intell.* 2010;33(8):1548–60.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more [biomedcentral.com/submissions](http://biomedcentral.com/submissions)

