Learning-to-rank based compound virtual screening by using pairwise kernel with multiple heterogeneous experimental data

22nd International Symposium on Artificial Life and Robotics (2017.1.19-21)

Tokyo Tech

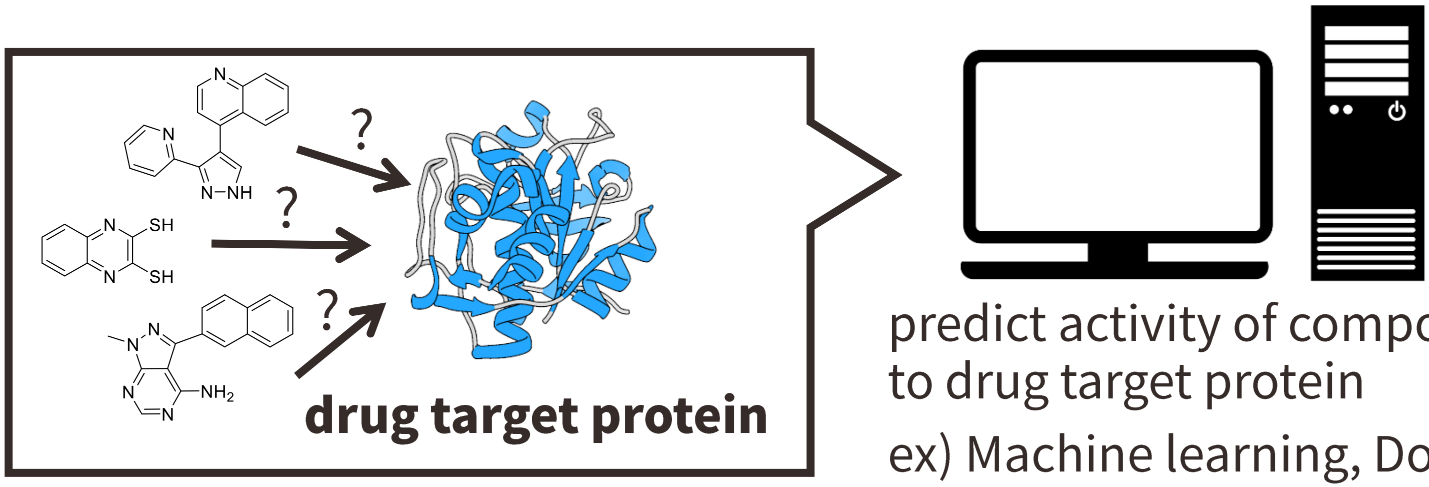
Academy of Computational Life Sciences · Education

Graduate School of Information Science and Technology
Tokyo Institute of Technology
Shogo D. Suzuki, Masahito Ohue, Yutaka Akiyama
鈴木 翔吾

Table of Contents

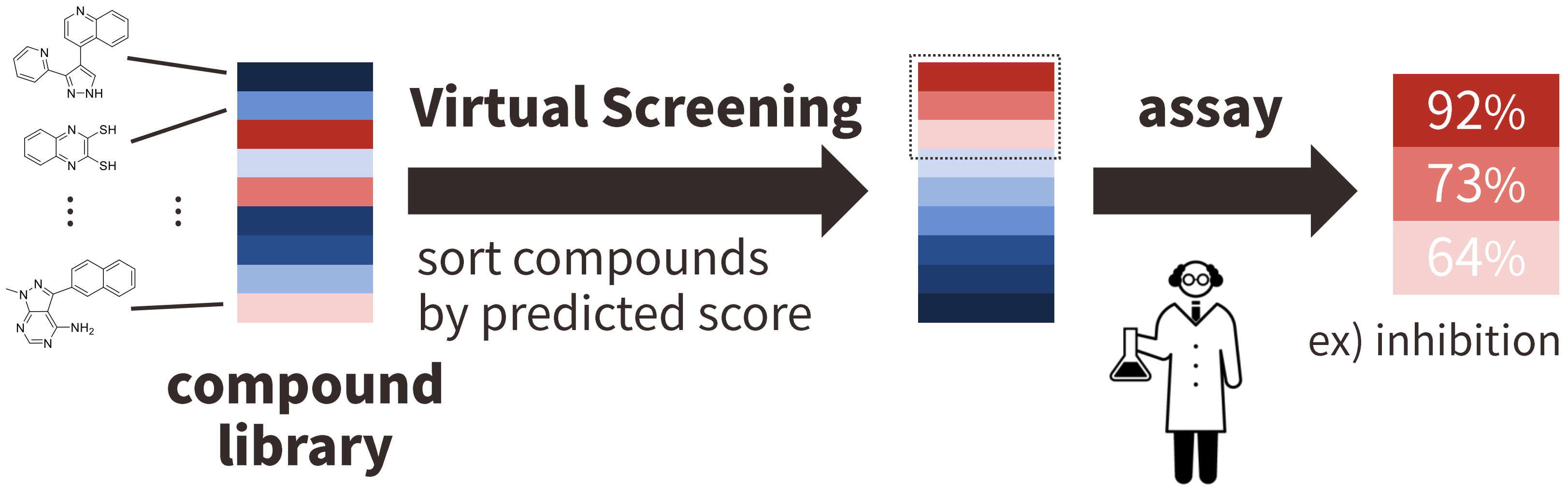
Introduction | Virtual Screening

Virtual Screening (VS): computational technique for drug discovery



predict activity of compounds
to drug target protein

ex) Machine learning, Docking simulation

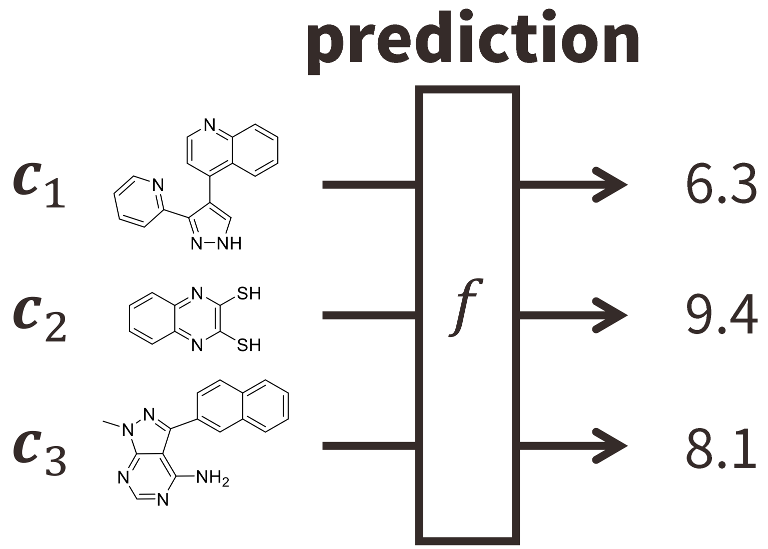


compound library

sort compounds
by predicted score

ex) inhibition rate, IC_{50}

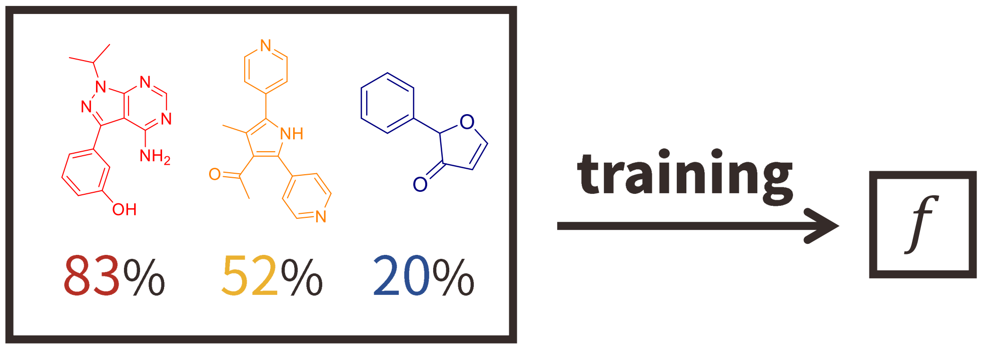
Introduction | VS with Machine Learning



predict the ranking of compounds

$$c_2 \succ c_3 \succ c_1$$

How to construct prediction model f ? ⟶ Machine Learning approach



train dataset
(known assay data)

Learning to Rank

training the order of items
ex) web page ranking

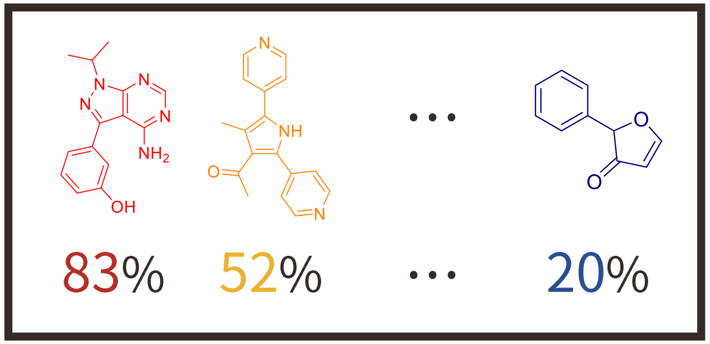
f(●) > f(●) > f(●)

[Agarwal+ 2010, Rathke+ 2011, Zhang+ 2015]

Introduction | difficulty of VS

For some drug target proteins, there are **few or no** known assayed compounds.

drug target protein

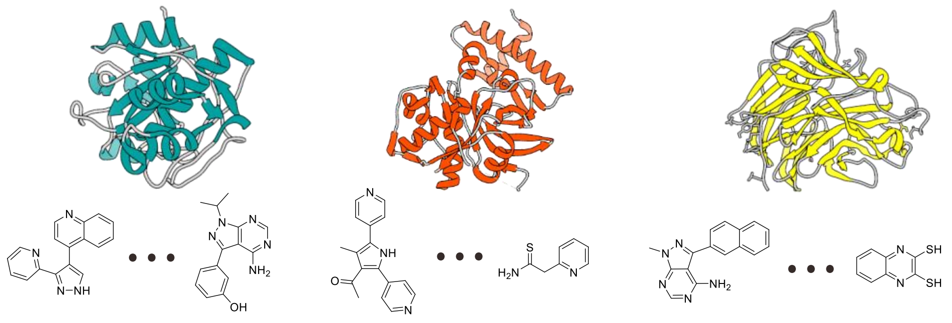


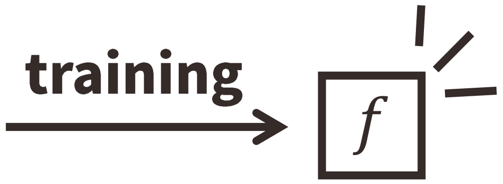
few known assayed compounds (~100)

Problem: It's hard to make good prediction model f

Solution: Use assay data whose target protein is related to drug target.

assay data for related proteins



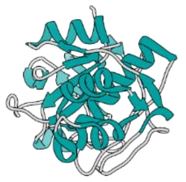


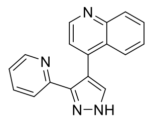
Introduction | Zhang+ approach

Zhang+ (2015) approach: Learning to Rank + using multiple data

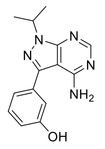
tensor product of feature vectors of protein and compound

protein: 1





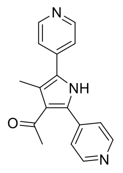
$\succ$



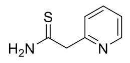
$\succ \cdots$

$\mathbf{p}_1 \otimes \mathbf{c}_{11}$ $\mathbf{p}_1 \otimes \mathbf{c}_{12}$

protein: 2



$\succ$



$\succ \cdots$

$\mathbf{p}_2 \otimes \mathbf{c}_{21}$ $\mathbf{p}_2 \otimes \mathbf{c}_{22}$

$\vdots$ $\vdots$

$\underbrace{\hspace{10em}}_{\text{training}}$

RankSVM (Learning to Rank method)

$\mathbf{p}_i$: protein feature vector
$\mathbf{c}_{ij}$: compound feature vector

$$(2,3) \otimes (2,4,5)$$
$$= (2 \times 2, 2 \times 4, 2 \times 5, 3 \times 2, 3 \times 4, 3 \times 5)$$
$$= (4, 8, 10, 6, 12, 15)$$

Introduction | Purpose of this study

Purpose

obtain <u>more accurate</u> prediction model than tensor product method

Approach

PKRank: <u>P</u>airwise <u>K</u>ernel + Kernel <u>Rank</u>SVM

generalize tensor product method with pairwise kernel

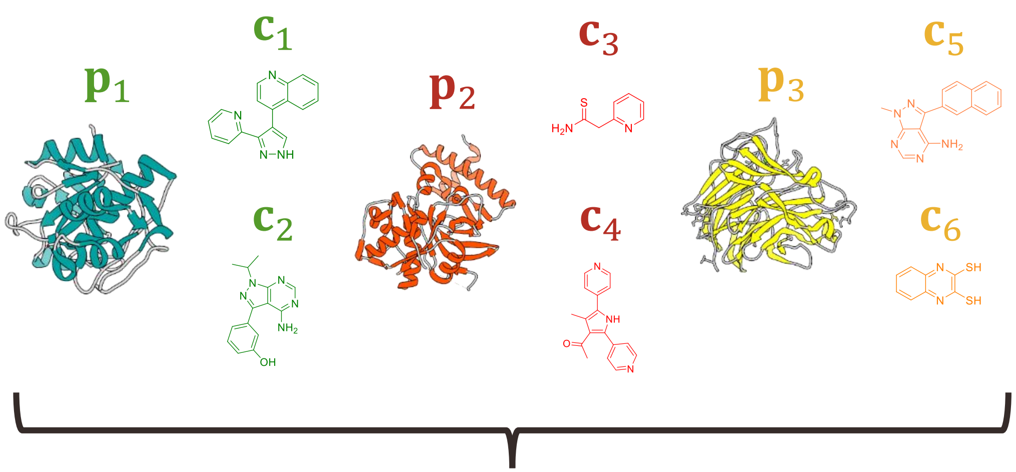
▼

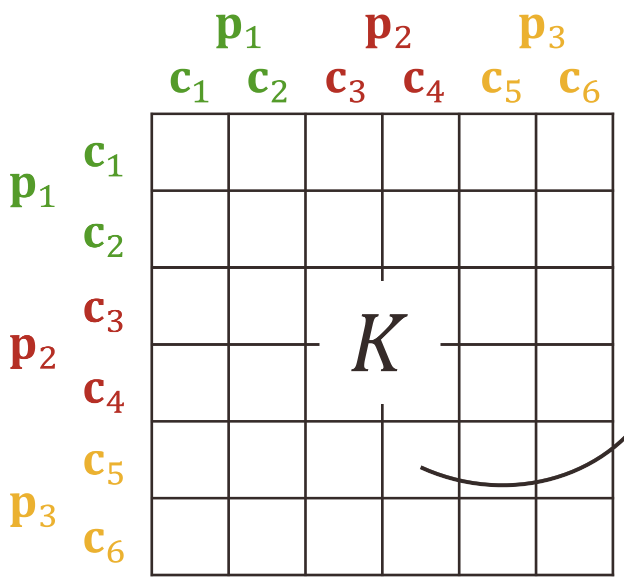
construct more flexible prediction model

Table of Contents

Method | overview of PKRank

1. Generate Gram matrix of pairwise kernel




$\mathbf{p}_1$ $\mathbf{p}_2$ $\mathbf{p}_3$
$\mathbf{c}_1$ $\mathbf{c}_2$ $\mathbf{c}_3$ $\mathbf{c}_4$ $\mathbf{c}_5$ $\mathbf{c}_6$
K

pairwise kernel

$$k\big((\mathbf{c},\mathbf{p}),(\mathbf{c}',\mathbf{p}')\big) = k_{\mathrm{com}}(\mathbf{c},\mathbf{c}') \times k_{\mathrm{pro}}(\mathbf{p},\mathbf{p}')$$

2. training (kernel RankSVM)

[Kuo+ 2014]

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\hat{Q}\boldsymbol{\alpha} - \mathbf{e}^{\mathrm{T}}\boldsymbol{\alpha}$$

subject to $0 \le \alpha_{i,j} \le C$

$$\hat{Q}_{(i,j),(u,v)} = K(x_i,x_u) + K(x_j,x_v) - K(x_i,x_v) - K(x_j,x_u)$$

Method | pairwise kernel

Pairwise Kernel: kernel function between two pairs of compounds and proteins

Pairwise kernel is obtained from compound kernel and protein kernel

$$\underbrace{k\big((\mathbf{c},\mathbf{p}),(\mathbf{c}',\mathbf{p}')\big)}_{\text{pairwise kernel}} = \underbrace{k_{\mathrm{com}}(\mathbf{c},\mathbf{c}')}_{\text{compound kernel}} \times \underbrace{k_{\mathrm{pro}}(\mathbf{p},\mathbf{p}')}_{\text{protein kernel}}$$

$\mathbf{c},\mathbf{c}'$: compound feature
$\mathbf{p},\mathbf{p}'$: protein feature

If both k_{com} and k_{pro} are represented as a linear kernel,
PKRank is equivalent to the tensor product method.

※the detail in my proceeding

Method | advantages of PKRank

1. PKRank can treat high dimensional feature vector

tensor product method: $d(\mathbf{c}) \times d(\mathbf{p})$

If $d(\mathbf{c})$ or $d(\mathbf{p})$ is large, tensor product feature is too large.

PKRank can avoid $d(\mathbf{c}) \times d(\mathbf{p})$ feature with kernel method.

$d(\cdot)$: dimension

2. PKRank can treat various kernels

tensor product method: equivalent to PKRank with linear kernel.

Other kernels can be used for compound kernel and protein kernel.

3. PKRank can treat similarity measurement for training

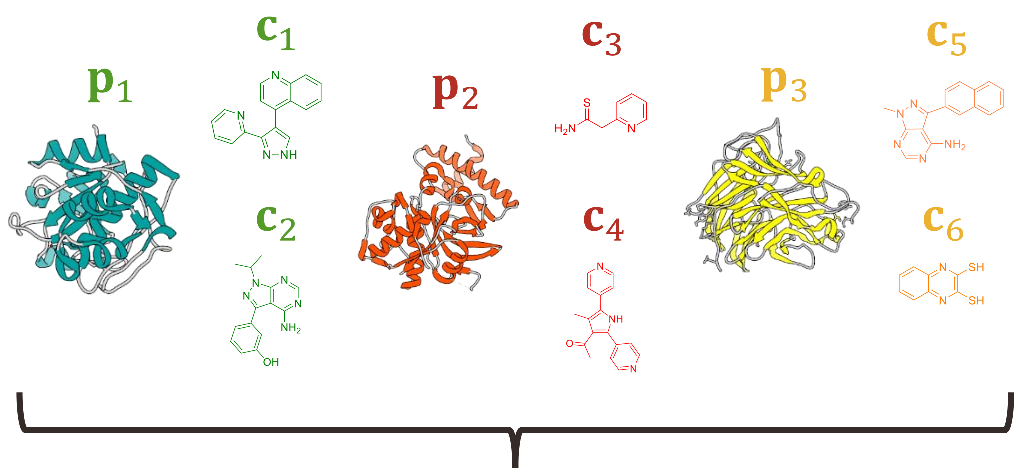
kernel function can be regarded as similarity measure.

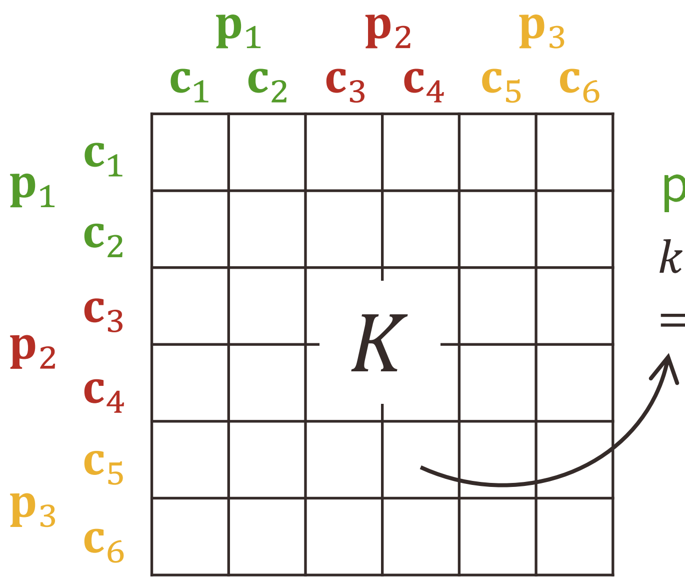
tensor product method cannot treat similarity measurement.

ex) similarity between two proteins -> alignment score

Method | overview of PKRank (written again)

1. Generate Gram matrix of pairwise kernel




$\mathbf{p}_1$ $\mathbf{p}_2$ $\mathbf{p}_3$
$\mathbf{c}_1$ $\mathbf{c}_2$ $\mathbf{c}_3$ $\mathbf{c}_4$ $\mathbf{c}_5$ $\mathbf{c}_6$
K

pairwise kernel

$$k\big((\mathbf{c},\mathbf{p}),(\mathbf{c}',\mathbf{p}')\big) = k_{\mathrm{com}}(\mathbf{c},\mathbf{c}') \times k_{\mathrm{pro}}(\mathbf{p},\mathbf{p}')$$

2. training (kernel RankSVM)

[Kuo+ 2014]

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\hat{Q}\boldsymbol{\alpha} - \mathbf{e}^{\mathrm{T}}\boldsymbol{\alpha}$$

subject to $0 \le \alpha_{i,j} \le C$

$$\hat{Q}_{(i,j),(u,v)} = K(x_i, x_u) + K(x_j, x_v) - K(x_i, x_v) - K(x_j, x_u)$$

Table of Contents

Experiment | conditions

Dataset test data: PDE5, CTSK, ADORA3

※number shows #compounds

PDE family (15 subfamilies)				
PDE1a (12)	PDE1b (132)	PDE1c (141)	PDE2a (324)	PDE3a (177)
PDE3b (22)	PDE4a (356)	PDE4b (514)	PDE4c (83)	**PDE5 (835)**
PDE6a (32)	PDE6c (13)	PDE9a (72)	PDE10 (1307)	PDE11a (76)
CTS family (10 subfamilies)				
CTSB (440)	CTSD (686)	CTSE (20)	CTSF (20)	CTSG (186)
CTSH (15)	**CTSK (735)**	CTSL (566)	CTSS (771)	CTSZ (6)
ADOR family (4 subfamilies)				
ADORA1 (390)	ADORA2a (141)	ADORA2b (199)	**ADORA3 (201)**	

Evaluation

Normalized Discounted Cumulative Gain (NDCG)

NDCG1@100・NDCG1@10・NDCG2@10

※the detail in my proceeding

Experiment | result

The result of PDE family dataset. The other results are in my proceeding.

compound feature	compound kernel	protein feature	protein kernel	NDCG1@100	NDCG1@10	NDCG2@10
GD	linear	CTD	linear	0.821	0.729	0.258
GD	RBF	CTD	RBF	*0.834	*0.830	*0.336
ECFP4	linear	CTD	linear	0.776	0.715	0.275
ECFP4	Tanimoto	CTD	RBF	0.827	0.740	0.313
ECFP4	RBF	CTD	RBF	*0.838	*0.811	*0.390
GD	RBF	sequence	nSW	***0.855**	***0.847**	*0.371
ECFP4	Tanimoto	sequence	nSW	0.827	0.745	*0.329
ECFP4	RBF	sequence	nSW	*0.849	*0.835	***0.399**

gray line	correspond to tensor product method of Zhang+ (2015)
bold	best score for each evaluation score
(*)	significantly improvement (paired t-test P < 0.05)

PKRank outperforms tensor product method.

Table of Contents

Conclusion

Purpose

obtain more accurate prediction model than tensor product method

Approach

PKRank: Pairwise Kernel + Kernel RankSVM

$$k\big((\mathbf{c},\mathbf{p}),(\mathbf{c}',\mathbf{p}')\big) = k_{\mathrm{com}}(\mathbf{c},\mathbf{c}') \times k_{\mathrm{pro}}(\mathbf{p},\mathbf{p}')$$

Result

PKRank outperforms tensor product method

Future study

Will more assay data improve prediction accuracy ?
Which combination of kernels works well ?