Machine/Deep Learning Causal Pharmaco-Analytics for Preclinical System Pharmacology Modeling and Clinical Outcomes Analytics

By

Yankang Jing

BS, Huazhong University of Science and Technology, China 2011

MS, Northeastern University, MA, USA 2013

Submitted to the Graduate Faculty of

School of Pharmacy in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh, Pittsburgh, PA, USA

2021

UNIVERSITY OF PITTSBURGH

SCHOOL OF PHARMACY

This dissertation was presented

By

Yankang Jing

It was defended on

September 28, 2021

And approved by

Levent Kirisci, Professor, Department of Pharmaceutical Science

Xinhua Lu, Professor, Department of Biomedical Informatics

Ralph E. Tarter, Professor, Department of Pharmaceutical Sciences

Ying Xue, Assistant Professor, Department of Pharmacy and Therapeutics

Dissertation Advisor: Xiang-Qun Xie, Professor, Department of Pharmaceutical Sciences

Machine/Deep Learning Causal Pharmaco-Analytics for Preclinical System Pharmacology Modeling and Clinical Outcomes Analytics

Yankang Jing, PhD

University of Pittsburgh, 2020

The modern drug discovery and development pipeline are complex, lengthy, and costly. The amount and availability of biomedical data explosive increase in the past decade, accompanied by rapid developments in computational technology. Those provide new and exciting opportunities to better and systematically understand the biology and pharmacology of diseases with the help of data science, machine/deep learning (ML/DL), and artificial intelligence (AI) technologies. Pharmaco-Analytics rises from introducing data science-driven methods to the traditional modeling and simulation in pharmaceutical & clinical sciences. It encompasses topics that cover preclinical and clinical analyses, for example, computational drug discovery, bioanalytical methodology, Pharmacometrics and Systems Pharmacology, Pharmacoeconomics, and outcomes analytics. In the Pharmaco-Analytics field, there are emerging AI/ML technology development and growing numbers of applications published increasingly facilitate pharmaceutical sciences and health care research.

We present six studies in Chapters 2 to 4 introducing our innovation of developing ML/AI methods to inform preclinical modeling and clinical outcomes research. The first two studies describe two computational methods on target identification using Pharmaco-Analytics technology. The first study involves the development of an AI platform to investigate drug abuse Poly-pharmacology using computational chemistry and machine learning algorithms. The second study introduces a novel algorithm (DeepTargetHunter) to identify the target of small molecules based on a novel deep learning technique for drug repurposing. The subsequent two studies focus on developments for preclinical properties prediction. The first study introduces a novel graph-based method (DeepGhERG), to predict the hERG cardiotoxicity of small molecules and the second study describes DL methods to predict blood-brain barrier permeability.

Lastly, we examine two methods to predict the risk of substance use disorder (SUD) based on childhood psychopathological traits. The first study presents a novel ML method to predict SUD outcomes based on deriving 30 of the most important questionnaire items predicting SUD. Whereas the second study introduces a novel approach called CausalSUD to identify the causal relationship between psychopathological cluster patterns and risk of SUD from late childhood to adulthood. In aggregate, the results from this research demonstrate the heuristic utility of AI/ML methods for advancing the Pharmaco-Analytics research in preclinical modeling and causal machine learning on clinical outcomes analysis.

Keywords: Pharmaco-Analytics, artificial intelligence, machine learning, deep learning, graphic neural network, TargetHunter, hERG, BBB, GPCRs, substance use disorder, causal analysis

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

This dissertation is dedicated to my parents, Xiangfeng and Jing, and my fiancée Siyi. Thanks to them for never losing hope in me and encouraging me during the hard times. Without their unconditional love and insistent support, I may not be able to accomplish this journey.

I am lucky to be a member of the University of Pittsburgh School of Pharmacy; the professors and colleagues here are so lovely and outstanding. I want to thank my supervisor, Dr. Xiang-Qun Xie, for his excellent guidance and support during my Ph.D. training. He always supports me and gives me good suggestions in both my research and my life, and I learn a lot from him, which I believe would be the fortune of all my life.

At the same time, I'd like to owe my gratitude to all the committee members, Dr. Kirisci, Dr. Lu, Dr. Tarter, and Dr. Xue, for their helpful and valuable suggestions during my research and final dissertation preparation. I also like to thank Dr. Junmei Wang, Dr. Lirong Wang, Dr. McGuire, and other professors in our CCGS center; I really appreciate your help during my training. I am constantly proud of being a family member of our CCGS center, and I have met many excellent friends here; I benefitted a lot learning from all of you. Finally, give my sincere thanks to the faculty and staff at the University of Pittsburgh School of Pharmacy, especially Dr. Folan, who gave me great help when my life was miserable. To my other colleagues and friends in the School of Pharmacy, I would like to thank you for your help during those several years.

ABBREVIATIONS

AdaBoost	adaptive boosting
ADMET	absorption, distribution, metabolism, excretion, and toxicity
AI	artificial intelligence
ANN	Artificial Neural Network
AUC	The area under the curve
AUROC	The area under the receiver operating curve
BIC	Bayesian information criterion
BN	Bayesian network
BRANN	Bayesian regularized artificial neural network
CEDAR	Center for Education and Drug Abuse Research
CNN	convolutional neural network
CNS	central neural system
DAG	directed acyclic graph
DEAN	Deep Auto-Encoder Network
DL	deep learning
DNN	deep neural network
DTI	drug-target interaction
ECFP	extended connectivity fingerprints
FCFP	functional class fingerprints
FNN	feedforward neural network
GAFF	general AMBER force field

GAN	generative adversarial networks
GB	gradient boosting
GPU	graphics processing unit
HBA	hydrogen-bond acceptor
HBD	hydrogen-bond donor
kNN	k nearest neighbor
MCCS	marine corps community services fingerprints
MLP	multilayer perceptron
MW	molecular weight
NLP	natural language processing
OOB	out-of-bag
QSAR	quantitative structure-activity relationship
QSPR	quantitative structure property relationship
RBM	Restricted Boltzmann Machine
RF	random forest
RNN	recurrent neural network
ROC	receiver operating curve
SGD	stochastic gradient descent
SUD	substance use disorder
SVM	support vector machine
TPSA	topological polar surface area

CHAPTER 1. INTRODUCTION

1.1 An Overview of Artificial Intelligence and Pharmaco-Analytics

Modern drug discovery and development pipelines are complex, lengthy, and costly (Khanna 2012). To understand the disease development and identify plausible therapeutic hypotheses, various advanced experimental technologies, such as high-throughput techniques, were developed, followed by the generation of a vast amount of biomedical data (Hertzberg and Pope 2000). Such data explosion provides opportunities for scientists to study the biology and pharmacology of diseases better. Furthermore, the conventional target-based drug discovery, which is the basis of the ‘one-drug-one-target-one-disease’ assumption, has been examined as the less successful way for complex diseases.

Systems pharmacology has emerged as the new discipline to tackle such challenges in drug discovery (Zhou et al. 2016; Vicini and van der Graaf 2013). Integrating and analyzing those massive heterogeneous pharmaceutical science data collected from all stages of drug discovery and development and clinical trials becomes a critical problem. The recent growth in data science and artificial intelligence (AI) (**Figure 1.1.1**) technology offered a systematical methodology to analyze that information comprehensively('Data Science and its Relationship to Big Data and Data-Driven Decision Making' 2013).

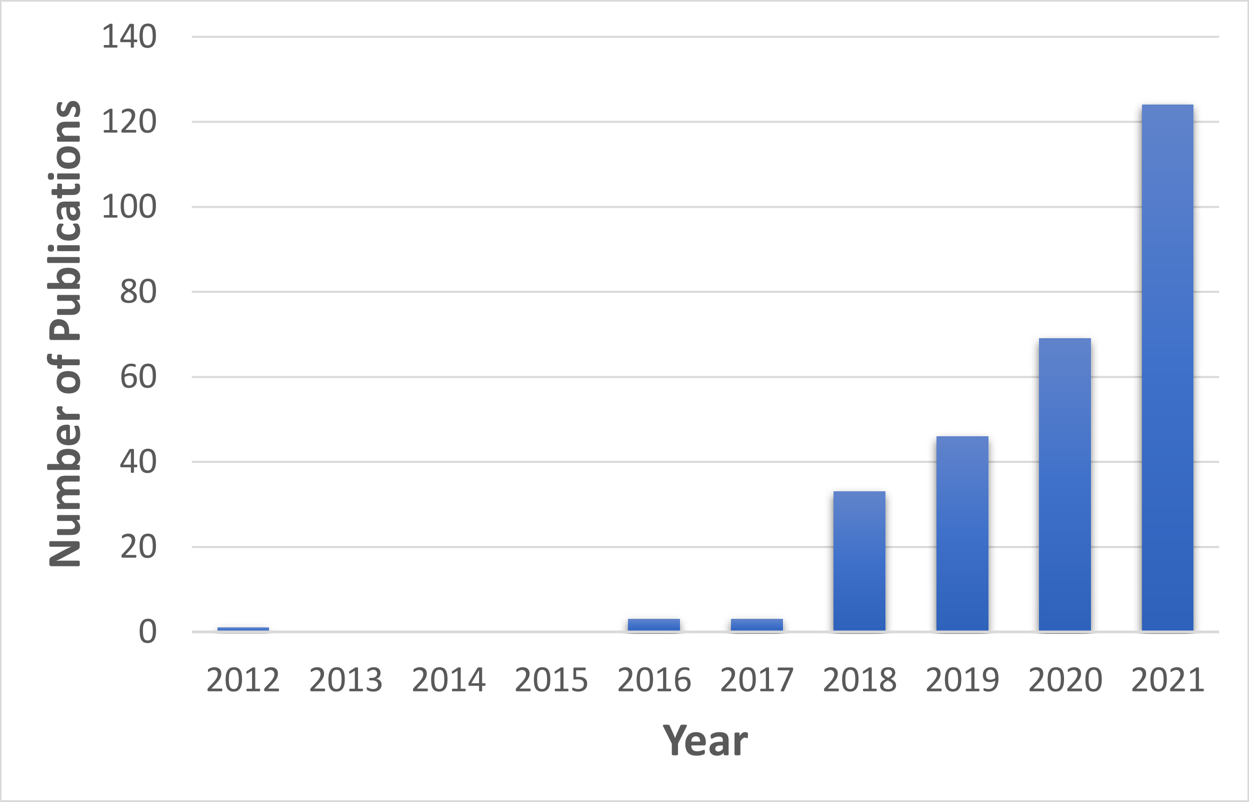


Figure 1.1.1 Number of published research using different AI for drug discovery in the recent decade. The data shown in this figure was collected by searching ‘Artificial intelligence’, ‘drug discovery’ in SciFinder (scifinder.cas.org)

The general philosophy of AI can be traced back to the 1950s, which aimed to develop a human-like machine system that interacts with the environment automatically (Lungarella et al. 2007). Thanks to the rapid development of modern computational technology, AI has moved from largely theoretical studies to real-world applications over the past decade. The success of AI applications can be observed in different areas such as computer vision, natural language processing, anomaly detection, and so on (Das et al. 2015). In the Pharmaco-Analytics field, more and more AI/ML method and applications are developed to facilitate drug development and health care research (Jing et al. 2018c; Jumper et al. 2021; Zhou, Wang, et al. 2020).

The AI system can 'learn' from a large volume of biomedical data, and then use the obtained insights to assist decision-making tasks. For example, the health care AI system may help physicians by providing real-time medical information to inform proper patient care or giving appropriate suggestions to reduce diagnostic and therapeutic errors common in the clinical practice (Yu, Beam, and Kohane 2018). In drug discovery and development, AI applications help scientists in various forms and achieve varying degrees of success in drug design (Figure 1.1.2). One of those examples is the AI robotic synthesis platform which can predict the synthetic route of an organic compound and automatically synthesis the compound (Coley et al. 2019). Another example of AI in this field is the development of Alpha-Fold (Jumper et al. 2021), an ML-based protein structure prediction method that accurately predicted the structures of 98.5% of human proteins in the entire human proteome (Tunyasuvunakool et al. 2021).

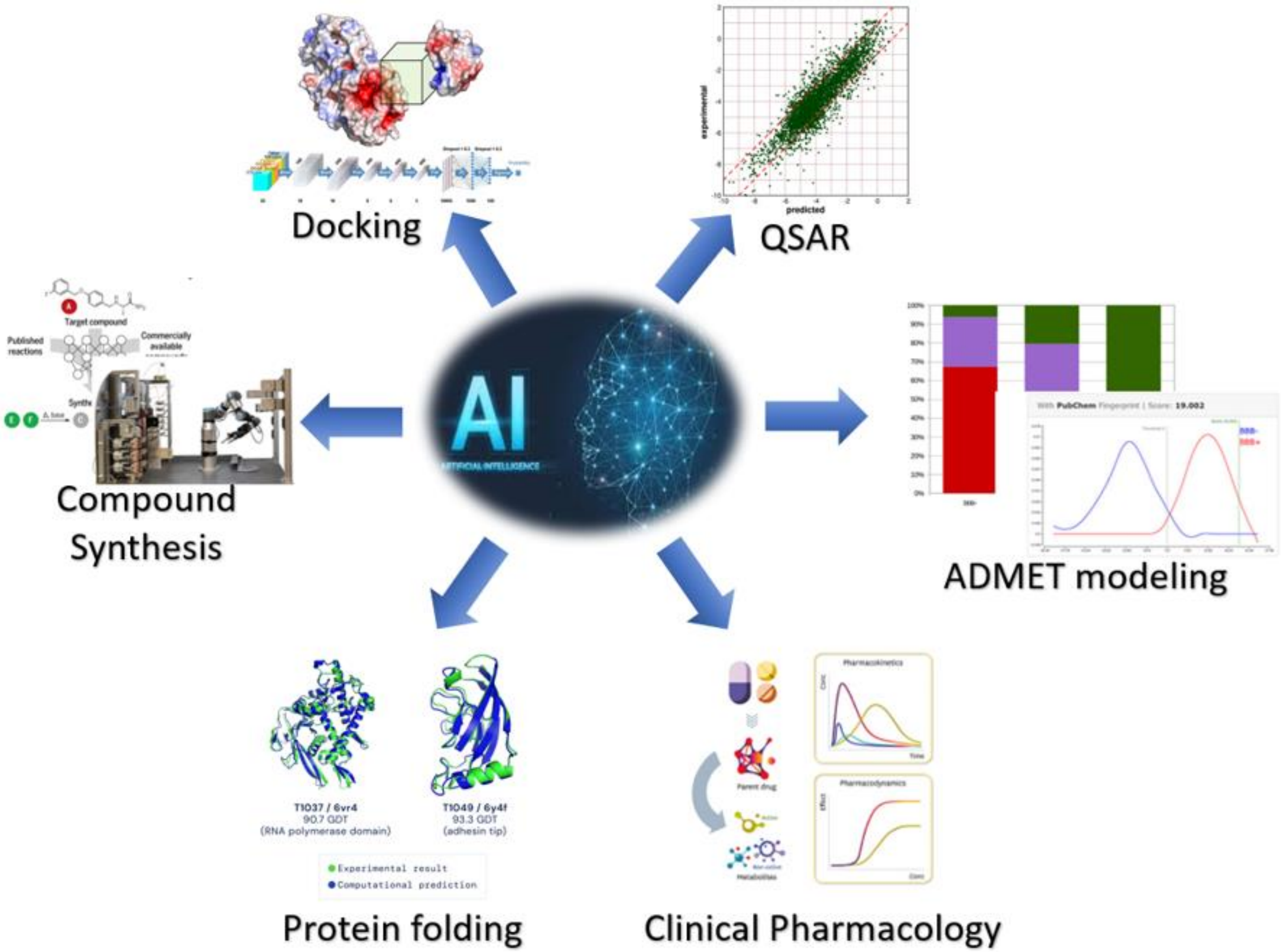
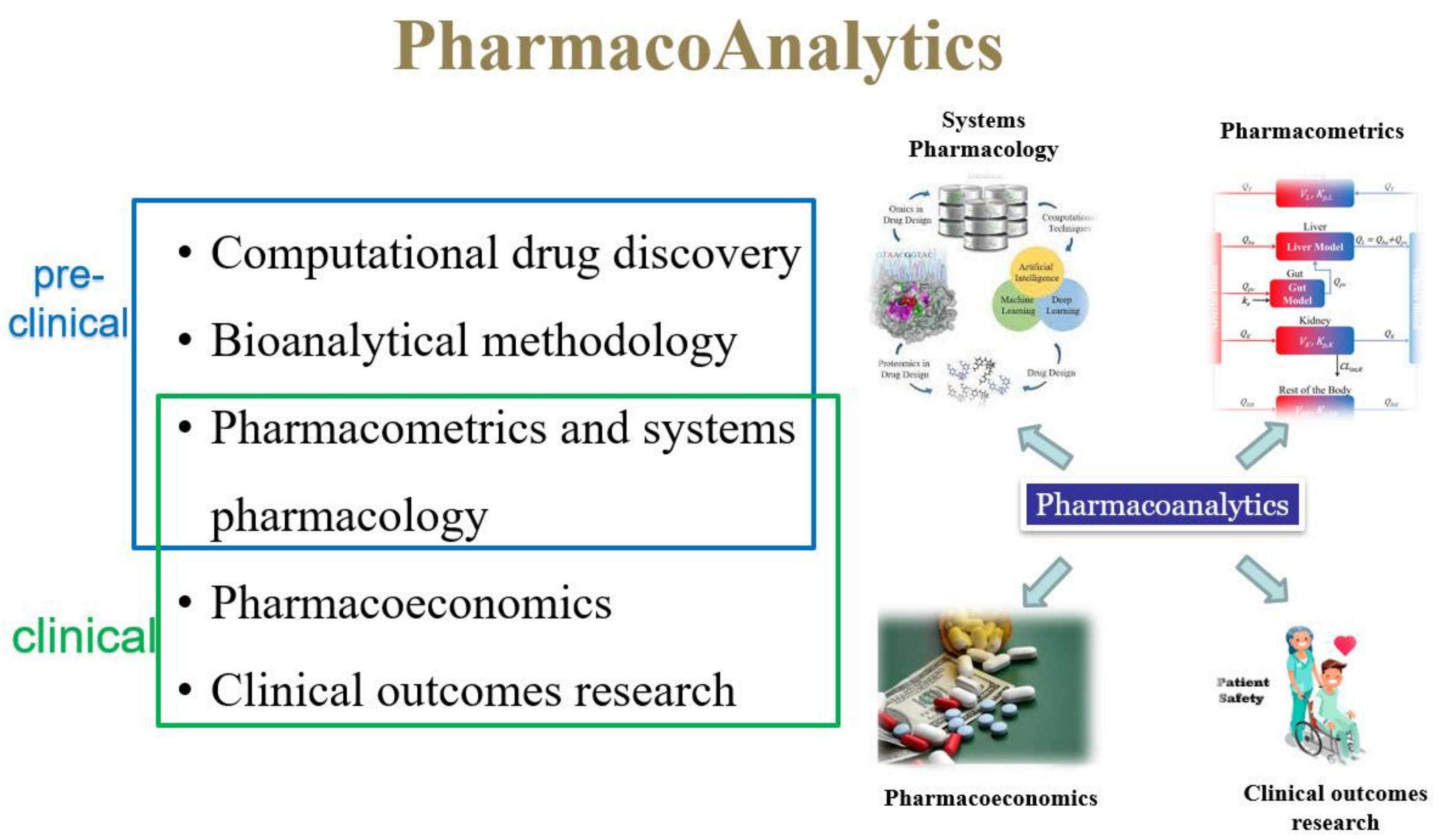


Figure 1.1.2 AI and PharmacoAnalytics in drug discovery

The concept of Pharmaco-Analytics comes from introducing data-driven methods to the traditional modeling and simulation in pharmaceutical science. It is a collection of medication-oriented computational data science technologies across the drug discovery life cycle. The aim of developing Pharmaco-Analytics is to develop and apply in-depth knowledge of drug action, and apply patient factors for drug efficacy and safety.

Pharmaco-Analytics comprises preclinical and clinical methods to advance data-driven improvements in multiple drug discovery and development processes, such as computational drug discovery, bioanalytical methodology, pharmacometrics, and systems pharmacology (PSP), Pharmacoeconomics and clinical outcomes, and patient safety (**Figure 1.1.3**). The power of AI and data science applications will enhance that data-driven Pharmaco-Analytics research when integrated with traditional AI computational analysis and multiscale modeling (Hart and Xie 2016; Jing et al. 2018b).

Overall, the goals of applying AI in Pharmaco-Analytics involve different challenges such as target selection, hit identification, lead optimization to preclinical studies and clinical trials, etc. While AI may not answer every challenge, it is a valuable tool with correct usage to help drive discoveries (Sellwood et al. 2018).



PharmacoAnalytics

pre-clinical

- Computational drug discovery
- Bioanalytical methodology
- Pharmacometrics and systems pharmacology

clinical

- Pharmacoeconomics
- Clinical outcomes research

Figure 1.1.3 General overview of Pharmaco-Analytics and its application

1.2 Machine Learning and Pharmaco-Analytics

1.2.1 Machine Learning and Pharmaco-Analytics

Machine learning (ML) is the technology for achieving AI. ML algorithms were developed to provide computer machine systems the ability to learn from experience without explicit programming (Brunette, Flemmer, and Flemmer 2009). In the learning process, the algorithms adaptively improve the model's predictive performance with increased quantity and quality of data. Therefore, ML is best applied to tasks with extensive training data.

Based on the type of problem to be solved, ML techniques can be divided into three main subclasses: supervised learning, unsupervised learning, and reinforcement learning (Shobha and Rangaswamy 2018). Supervised learning aims to learn the mapping function from the input variable (X) to the output variables (Y) so that it can be used to predict future values of data categories (classification) or continuous variables (regression) (Mahesh 2020). In unsupervised learning, there are no corresponding output variables (Y); therefore, it can be used for exploratory purposes to learn the underlying structure or distribution of the data, such as clustering (Hastie, Tibshirani, and Friedman 2009). Reinforcement Learning is the algorithm that guides the learning process in an interactive environment by rewarding desired behaviors and punishing undesired behaviors (Li 2017). This section will mainly provide an overview of supervised algorithms and introduce ML concepts and methods described later.

The fundamental concept in supervised learning is using algorithms to learn from large amounts of data and subsequently predict the future with new data sets (Mohri, Rostamizadeh, and Talwalkar 2018). The general workflow in supervised learning consists of several steps: 1) data preparation; 2) feature engineering; 3) model training; 4) model validation; and 5) model refinement (Praveena and Jaiganesh 2017). The prepared clean data are usually split into three subsets: a training set, a validation set, and a test set. The training set provides information for model fitting; the validation set assesses model performance during the training process and gives feedback applied for model refinement or selection; the test set provides an unbiased evaluation of model accuracy.

Feature selection. The concept of "feature" in ML refers to the explanatory variable used in statistical analysis. Feature selection is the technique used to recognize the variables needed to 1) predict the raw data and 2) reduce the imbalance between small sample size and large feature number which usually decreases the accuracy of the predictive model (Liu and Zhao 2012). To extract the most informative features and remove redundant information, feature selection reduces the dataset's dimensionality (number of features).

Generalized linear model. A generalized linear model (GLM) is a statistical modeling method that estimates the linear relationship between the dependent variable (Y) and the predictor variables ($X = [X_1, X_2, X_3, \ldots, X_n]$) (Nelder and Wedderburn 1972). Common examples of GLMs include linear regression and logistic regression (Cox 1958). For instance, in a linear regression model, Y is expressed as a linear combination of all the predictor variables X, and the random error distribution of X is considered normally distributed.

Support vector machine. Support vector machine (SVM) is a widely used non-probabilistic classification algorithm for handling high-dimensional data (Cortes and Vapnik 1995). Early SVM was designed to find the linear decision boundary (hyperplane) in the geometric feature space that separates the two classes as widely as possible. Optimized SVM methods were developed for nonlinear data sets by introducing kernel methods that use mathematical transformations to map the original data into another dimension where the two classes are linearly separable (Soman, Loganathan, and Ajay 2009).

Naïve Bayesian. The Naïve Bayes algorithm is the machine learning algorithm that takes advantage of the Bayes' theorem under the “naive” assumption that explanatory variables are conditional independents (Domingos and Pazzani 1997). Naive Bayes calculates the posterior probability of each class at the condition of given features, and the outcome class with the highest probability is the predictive outcome. Even with the ‘naïve’ assumption that is most unlikely in real data, in practice, Naïve Bayes methods usually perform surprisingly well on data where this assumption does not hold.

Decision tree. The decision tree is a graphical method of classifying cases following a hierarchy of “if-else” conditions based on the features. The class label distribution in the original population is usually represented at the starting node of the tree (root node). The population can then be split into sub-groups (branches) based on a feature's values that differentiates the class distribution. This process is repeated at each branch (recursive partitioning) until all cases with the same class label or no features can further determine the class distributions.

Tree-based ensemble methods. Random forest (RF) algorithm is a commonly used tree-based ensemble learning algorithm for various tasks such as classification and regression (Breiman 2001a; Ho 1998). It consists of many decision tree models trained using a randomly drawn subset of the original dataset from bagging (Quinlan 1996). This approach ensures the independence of each decision tree. In addition, the features for training each decision tree are randomly drawn from the original feature set to avoid having all trees focus on the same few strong predictors and ignore the others. The final prediction is based on majority voting from all decision tree models; therefore, although each decision tree may be sensitive to random noise in the training data, their collective decision is reasonably robust. Another commonly used tree-based ensemble method is adaptive boosting (AdaBoost), consisting of multiple weak decision tree models (Solomatine and Shrestha 2004; Ma, Wang, and Xie 2011b). Instead of using randomized bagging as a random forest, AdaBoost adopts a boosting algorithm to conjugate weak classifiers into an ensemble model by assigning them weights and automatically adjusting weights using different training samples to achieve better predictability. And the final prediction is the weighted sum up of the output from those weak classifiers.

k-Nearest Neighbor. k-nearest neighbor (kNN) is an example of an instance-based learning algorithm used for classification and regression (Patrick and Fischer III 1970). When kNN model predicts the class label of a sample, the kNN algorithm first calculates the similarity of this case to all cases in the training set using a distance measure. It takes the average value or majority voting of the k training cases that are nearest to the query sample as the predicted label. The choice of k depends on the data and can be optimized via a heuristic search. Alternatively, the distances can also be used to weight the training cases so that the label of the query sample can be predicted as a weighted sum of all label values in the training set. In addition, the kNN algorithm is also widely used for non-supervised learning (Ding and He 2004)and data imputation (Beretta and Santaniello 2016).

Overall, ML algorithms can do linear and nonlinear fitting between dependent and independent variables and thus usually provide better predictive performance than traditional methods. However, the conventional ML algorithms introduced in this section have difficulty processing naturalistic data of raw forms. Therefore, hand-engineered features must be extracted to represent the input data, which is crucial but often intractable and requires expertise in the specific input data area. Thanks to well-established feature selection and model fitting techniques, a machine-learning-based model can be quickly built using a large dataset with relatively good performance. However, there are still disadvantages to using those machine learning models. For example, most machine learning models lack interpretability due to the complex non-linear fitting, and those ‘black box’ models help a little with explaining the mechanism and further optimization of the target compound. Secondly, although machine learning algorithms can deal with large datasets, redundancy and overfitting problems often make the prediction and classification unreliable (Ghasemi et al. 2018).

Currently, ML drives the success of artificial intelligence in both academia and industry. Specifically, ML methods have been widely used in Pharmaco-Analytics study of every drug discovery and development (Das et al. 2015) (**Figure 1.2.1**). One successful example of ML in Pharmaco-Analytics is the applications of ML in a quantitative structure-activity relationship (QSAR) or quantitative structure-property relationship (QSPR) models (Myint et al. 2012; Svetnik et al. 2003). QSAR/QSPR analysis is an important computational method for small molecular drug discovery, which can predict either the biological effect against a specific protein or the physicochemical properties based on the chemical structure (Roy, Kar, and Das 2015). Generally, a mathematical or statistical relationship (equation) will be established between the compound descriptors and the target affinity or physiochemical property to provide reliable predictions on the biological activity of target molecules. By applying ML-based modeling techniques in learning this relationship or equation, a more accurate forecast can be achieved with the increased size of the training dataset. However, there are also disadvantages of ML-based QSAR/QSPR. Firstly, the experimental error and noise may heavily increase the false correlation in such regression models. Secondly, the descriptors used for generating QSAR/QSPR models may cross-correlate, making it harder to build a robust model. Thirdly, different variables are not suggested to be mixed when building regression models, limiting the selection of descriptors to represent the compounds.

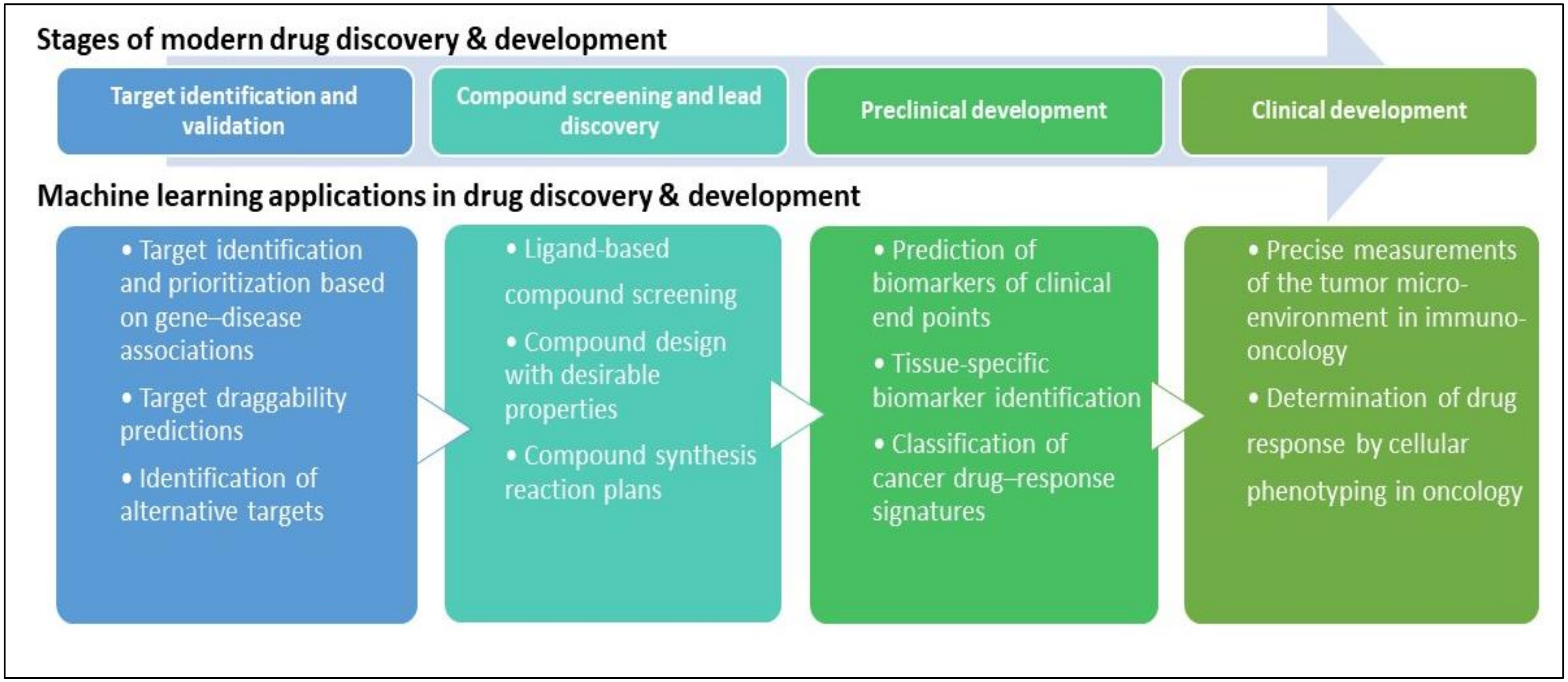

Stages of modern drug discovery & development
Target identification and validation
Compound screening and lead discovery
Preclinical development
Clinical development
Machine learning applications in drug discovery & development
• Target identification and prioritization based on gene–disease associations
• Target draggability predictions
• Identification of alternative targets
• Ligand-based compound screening
• Compound design with desirable properties
• Compound synthesis reaction plans
• Prediction of biomarkers of clinical end points
• Tissue-specific biomarker identification
• Classification of cancer drug–response signatures
• Precise measurements of the tumor micro-environment in immuno-oncology
• Determination of drug response by cellular phenotyping in oncology

Figure 1.2.1 Illustration of machine learning applications in systems pharmacology drug discovery and development research

1.2.2 Deep Learning Technology for AI in Pharmaco-Analytics

1.2.2.1 Background of deep learning

In March 2016, *AlphaGo* knocked out Lee Sedol, one of the best Go players in the world, bringing AI back into public attention overnight, spurring extensive interest ('Artificial intelligence: Google's AlphaGo beats Go master Lee Se-dol' 2016). Compared to *Deep Blue*, the chess-playing computer developed by IBM that beat the world champion for the first time back in the 1990s, *AlphaGo* integrated an advanced and innovative architecture called the convolutional neural network (CNN), which is one of the most successful implementations of the deep learning (DL) algorithms in neural networks (Silver et al. 2016). Benefiting from the rise of big data analysis and the advance of large-scale computing capabilities, especially the development of graphics processing unit (GPU) computing (Ma, Wang, and Xie 2011a), using DL architectures has emerged as the first attempted technologies to address AI challenges (Baskin, Winkler, and Tetko 2016).

DL is a rebranding of a traditional machine learning algorithm called artificial neural network (ANN), which consists of connected artificial neurons to mimic the human brain (McCulloch and Pitts 1943). The origin of a neural network can be traced back to the neural network algorithm proposed by Warren McCulloch and Walter Pitts in the 1940s and the perceptron algorithm invented by Frank Rosenblatt in the 1950s (Rosenblatt 1957). Both algorithms were designed to mimic the excitation of neurons in the human brain by analogizing the activation of a binary logic gate in the neural network. The main idea of this early ANN was to define an algorithm to learn the weight vector *w*, which was used as the coefficient of an eigenvalue. Then an activation function inside the neuron, such as *Heaviside Step Function* or S*igmoid Function*, was used to determine whether the neuron was activated or not(McCulloch and Pitts 1943; Rosenblatt 1957). Later on, the development of the BP algorithm (Kelley 1960) for ANN modeling brought the boom of those statistics-based ML methods for supervised learning. Such ANN was not ‘deep’ but ‘shallow’; it was composed of one input layer, one output layer, and one hidden layer in between (**Figure 1.2.2**). It receives signals from the hidden layer and then uses an activation function to produce an outcome. With a data stream following the process, those neural networks could be considered feedforward neural networks (FNN) (McCulloch and Pitts 1943). Optimization of these ‘shallow’ networks systems is achieved by first calculating the error between the output result and the actual value using the backpropagation algorithm (Rumelhart, Hinton, and Williams 1986). It then modifies the internal adjustable parameters (weights) to minimize the error through gradient descent (LeCun, Bengio, and Hinton 2015). The Universal Approximation Theorem states that ‘shallow’ networks, with only one hidden layer containing a finite number of nodes, could approximate any continuous function (Gao and Xu 1993). However, models with such architectures may be susceptible to overfitting when the number of adjustable parameters increases (such as several nodes with adjustable weight connections in the hidden layer). The overfitting problem can be minimized by carefully training of shallow networks, especially when regularization is applied(Lawrence and Giles 2000).

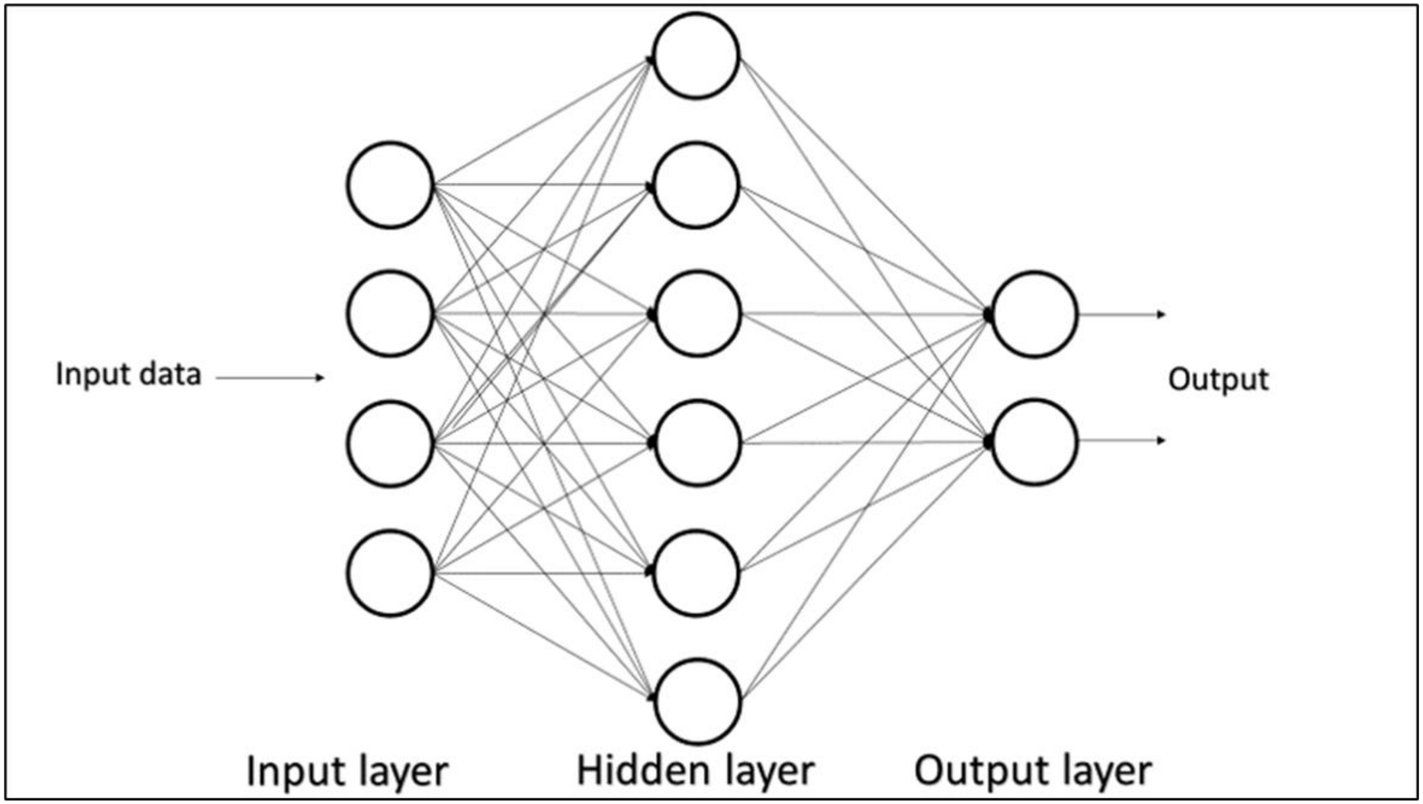


Figure 1.2.2 Architecture of artificial neural networks. The input layer receives input data directly by putting a feature into each node. Then each node in the hidden layer receives input of a weighted linear combination from all the units in the input layer and then uses an activation function to perform a nonlinear transformation. The output layer does similar work to the hidden layer.

Nevertheless, more hidden layers could be designed to recognize more abstract patterns from input data, with lower layers learning basic patterns and upper layers learning higher-level patterns. Adding more hidden layers and nodes could significantly increase the computation task. Those multilayer neural networks with many hidden layers may suffer from gradient vanishing problems (Hochreiter 1998), resulting in the difficulty of changing weights to optimize the model training. To overcome these situations, the network architectures were modified to optimize the initialization and the updating of the weights. Different transfer functions and regularization techniques are adopted to minimize overfitting (Winkler and Le 2017). Examples of those architectures include the deep belief network (Hinton, Osindero, and Teh 2006), CNN, and recurrent neural network (RNN) (Olurotimi 1994). Meanwhile, the development of GPU acceleration technology tremendously improved the computing power and helped the development of the DL method (Ma, Wang, and Xie 2011a).

The early practical framework of DL was proposed by Geoffrey Hinton and other scientists in 2006, opening the revolutionary waves of DL and new AI in academia and the industry (LeCun, Bengio, and Hinton 2015). They developed a novel architecture for multilayer NNs to introduce feature learning into DL for abstracting the essentials of the data. DL methods could automatically extract features from input data with the raw format through feature learning, then transform and distribute them into more abstract levels (LeCun, Bengio, and Hinton 2015). Meanwhile, the rapid development in parallel computing techniques and computing hardware, especially the emerging application-specific integrated circuit designed for DL study such as the Tensor processing unit technique ('Google supercharges machine learning tasks with TPU custom chip' 2016), ensured that the tremendous computing workload might no longer be an inaccessible domain (Schmidhuber 2015).

1.2.2.2 Common deep learning architectures and concepts

There are various DL architectures, each of which can recognize patterns and extract high-level features in distinct ways based on the training data structure. In this section we mainly discussed the basic DL architectures, including CNN, RNN, and the generative networks (LeCun, Bengio, and Hinton 2015).

Convolutional Neural Network. CNN is one of the most representative architectures in DL and is

widely adopted in many fields such as image and voice recognition, and natural language processing (NLP). The modern CNN came from the development of the recognition by Fukushima in the 1980s, which was inspired by the research into the cat's visual cortex receptive field by Hubel and Wiesel(Hubel and Wiesel 1962, 1959). When processing visual signals, local neuron patterns take responsibility for perceiving particular regions in the sensory space and CNN mimics its traits by developing two main characters in the convolutional layers: sparse connectivity and shared weights (Figure 1.2.2A) (Zeiler and Fergus 2014). Furthermore, the increase of robustness achieved by pooling layers and the integration of the dropout technique for regularization makes the CNN even more sophisticated (LeCun, Bengio, and Hinton 2015). For those complicated signaling processes, in which the input data have a massive number of input features and extremely abstract connections, the adoption of CNN could circumvent the headache of feature selection by directly importing the input data into the model (Pastur-Romay et al. 2016). There are three types of layers commonly used in CNN: the convolutional layer, the pooling layer, and the fully connection layer (Figure 1.2.2B). Those layers were carefully selected and arranged to form the multilayer network(LeCun et al. 1995; LeCun et al. 1998). Depending on the input data modality, different forms of layers can be considered. For example, for sequence signals such as language, layers can be formed with 1D arrays; for images or audios, layers can be formed with 2D arrays; and for videos, layers formed with 3D arrays can be applied(LeCun, Bengio, and Hinton 2015).

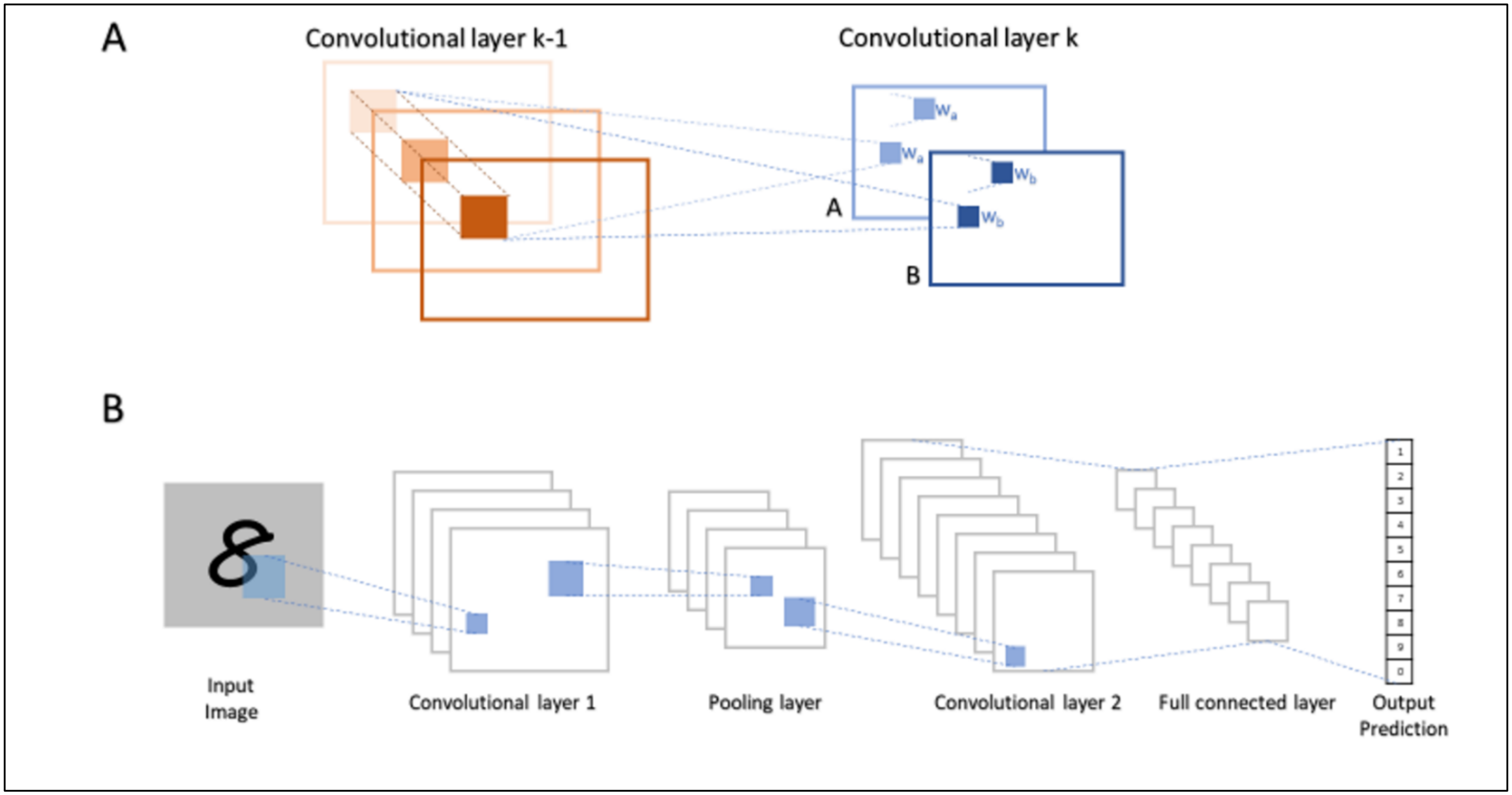


Figure 1.2.3 The architecture of convolutional neural network (CNN). In the convolutional layer k, there are two feature maps (A and B), either of which shares the same weight (we or wb). Every pixel in each feature map of the hidden layer k comes from the convolution of the weight matrix and the local pixel cluster of layer k-1.

Recurrent Neural Network. RNN is another representative type of architecture in DL. Especially aiming to handle sequence data, RNN has been widely used and succeeded in NLP. RNN is different from regular FNNs, which follow the feedforward architecture. In regular FNNs, there is no connection between hidden nodes in the same layer but only between nodes in adjacent layers (Figure 1.2.4). One of the significant shortages of FNNs is that they cannot handle sequence problems because the output is related to the current input information and prior information, for example, machine translation. However, RNN can process sequential information by 1) introducing directed cycles into its network; 2) affiliating the adjacent hidden nodes with each other; 3) capturing the calculated information from preceding time slices; and 4) storing it for the subsequent procedure(LeCun, Bengio, and Hinton 2015; Olurotimi 1994). In the RNN, each hidden layer with directed cycles could be unfolded and processed as a traditional NN sharing the same weight matrices U, V, W in every same layer. There are plenty of variations of RNNs. The most common ones are Gated Recurrent Unit Recurrent Neural Network (GRURNN)(Cho et al. 2014), Long short-term memory (LSTM) network(Hochreiter and Schmidhuber 1997), and Clockwork RNN (CW-RNN)(Si, Hsieh, and Dhillon 2014). Among these RNN architectures, LSTM is currently the most popular and widely used in NLP. In NLP, LSTM is often combined with a distributed word embedding representation, which is achieved by checking the statements and Part-of-Speech tagging(LeCun, Bengio, and Hinton 2015; Schmidhuber 2015). Using a specialized function to compute the transition state in the hidden layer, the LSTM network is robust when capturing long-term dependencies compared to regular RNNs. In addition, LSTM is also as popular and successful as CNN in the image retrieval domain and is usually combined with CNNs for the automatic generation of image description in AI (LeCun, Bengio, and Hinton 2015).

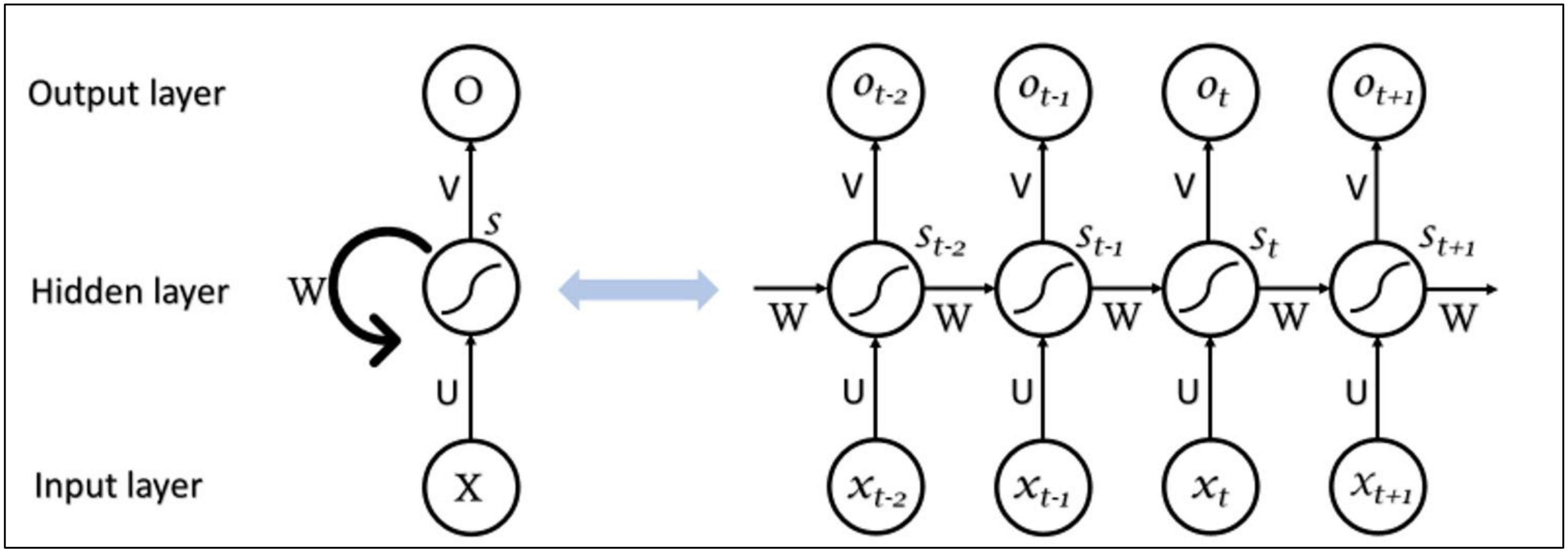


Figure 1.2.4 The architecture of recurrent neural network (RNN). RNN consists of input units (x, the vector representing the matrix of input data) and hidden units (s, the vector representing the matrix in the hidden layer), and output units (o, the vector representing the matrix of output data). U, V, W are the weight matrixes for the transition from x to s, s to s, and s to o, respectively. The one-way data flow streams from input units to output units, going through each sequential hidden unit. S_t represents the transition states of step t, which stands for the memorial units in the network containing all the extracted information from the prior data in the sequence. The output from the output units in that step (*t*) is only correlated with the transition state at that moment (S_t).

Generative Deep Neural Network. DNNs are not only for processing labeled data in supervised learning but also for analyzing non-labeled data in unsupervised learning. One such example is Deep Auto-Encoder Network (DEAN), a generative network for unsupervised learning (Bengio 2009; Pastur-Romay et al. 2016). DEAN consists of an encoder and a decoder, two symmetric DBNs, a deep neural network (DNN) proposed by Hilton et al. in 2006 (Hinton, Osindero, and Teh 2006). Those two DBNs are usually composed of several Restricted Boltzmann Machines (RBMs) (Hinton and Salakhutdinov 2006), a bipartite network containing one visible layer and one invisible layer. In RBM, there are symmetric connections between every two nodes from different layers and no connection between nodes from the same layer. The function of a simple Auto-Encoder can be regarded as the compression of data which can then be decompressed and recovered based on a BP algorithm with a minimal loss of information (Kelley 1960). Thus, DAEN is also considered an optimal method for dimensionality reduction because of its capacity to reduce redundancy. In this case, DAEN can be explicitly used for feature extraction for the reduced features to train a classification model using supervised learning algorithms (Chen et al. 2014). This paradigm may be valuable in the future development of DL applications. More recently, Generative Adversarial Networks (GANs), another type of DL algorithm for unsupervised learning, have been developed and widely used in image synthesis, image-to-image translation, and super-resolution (Goodfellow et al. 2014). Its development is motivated by observational data’s underlying probability density or probability mass function. Generator (G) is responsible for making non-realistic images from random vectors to confuse the other network known as Discriminator (D). When D receives both forgeries and real (authentic) images, it will tell them apart. In that module, G and D compete with each other and are trained simultaneously until they find the optimal parameters. Under those parameters, the G maximizes its classification accuracy, and the D maximizes its discrimination accuracy. Multilayer networks can be implemented by fully connected GANs, convolutional GANs, conditional GANs, GANs with inference models, and adversarial auto-encoder (AAE).

Regularization and Drop Out. Since over-fitting is a severe problem in multilayer DNNs, a broad range of techniques for regularizing has been developed to minimize the over-fitting problem. Dropout is one of the common ways to regularize NNs by dropping out units (hidden and visible) in NNs [47]. The critical idea of dropout is to add noise to its hidden units randomly; therefore, preventing over-fitting and improving test performance. Those DNNs which adopt dropout techniques can be trained through stochastic gradient descent (SGD) like regular DNNs. Similarly, each hidden unit in an NN adopted dropout must learn to work with a randomly chosen sample of other units, making them more robust than relying on other hidden units to correct its mistakes. Bayesian regularized artificial neural network (BRANN) is another development that introduced regularization into NN architecture. By using ridge regression in the mathematical process of model training, nonlinear regression can be converted into a “well-posed” statistical problem in the BRANN [48]. The cross-validation step for assessing the model in BRANN, which is usually tedious and time-consuming in DL modeling, may also be omitted. Automatic relevance determination (ARD) of the input features can be applied in BRANN to help calculate several effective network parameters or weights, which will cause the removal of parameters with smaller weights. In such a way, those irrelevant or highly correlated indices are neglected, and the essential variables for modeling are highlighted. Those two characteristics are beneficial for cheminformatics and QSAR/QSPR research because there are usually too many features to describe one molecule.

1.2.2.3 Deep learning algorithms for Pharmaco-Analytics

In recent years, the DL techniques have been adopted in Pharmaco-Analytics, opening a new door to computational decision-making in the pharmaceutical industry(Jing et al. 2018c). The success of DL techniques in Pharmaco-Analytics benefits from multiple aspects, including the innovative development of the DL algorithms, the progress in high-performance computing techniques, and the explosion of chemical information in chemical databases (Gray et al. 2015). Specifically, with the rapid development of big data and data science, the benchmark dataset is essential for constructing a model. In the drug discovery field, the *Merck Kaggle challenge* using a Merck-activity dataset, as well as the *Tox21 challenge* using its benchmark datasets significantly speeded up the application of machine learning methods in the QSAR/QSPR studies (Unterthiner et al. 2014; Casey 2013). Compared to traditional ML methods, DL methods have the capacity of processing 'big data'. Therefore, the need for large, standardized datasets for DL modeling is dire.

In 2017, Wu *et al.* introduced their large-scale benchmark package (*MoleculeNet)* for molecular modeling study (Wu et al. 2017). This dataset integrated multiple public molecular datasets, covering quantum mechanics, physical chemistry, biophysics, and physiology. In addition, all the datasets established metrics for model evaluation and implementations for calculated molecular features were packaged together with the modeling toolkits in their python library called *DeepChem*. In addition, Lenselink *et al*. published their benchmark dataset generated from the ChEMBL database, another standardized dataset for developing DL models (Gaulton et al. 2018).

DL models have been reported in three major areas in computational chemistry-- predicting the drug-target interactions (DTI), generating novel molecules, and predicting absorption, distribution, metabolism, excretion, and toxicity (ADMET) properties for translational research (Rubio et al. 2010). Like other ML algorithms, DL undergoes more and more successful applications in building QSAR/QSPR models. As early as 2012, Hilton's group won the *Merck Kaggle challenge* (https://www.kaggle.com/c/MerckActivity) using their DL models, opening a new chapter of applications using DL methods for predicting chemical compound activity and properties. Similarly, Wang and Zeng published their DTI-discriminative model using RBM, the commonly recognized first generation of DNNs (Wang and Zeng 2013). In the following year, Dahl *et al.* from Hilton's group and Google Inc. published several papers on DL-based QSAR modeling. They tried multiple tasks and different features using DNNs with various hyper-parameters and started using GPUs for a benchmark test(Ma et al. 2015; Dahl, Jaitly, and Salakhutdinov 2014; Ramsundar et al. 2015). In 2014, Wang *et al.* reported their DIT-predictive model using pairwise-input NNs, offering a new reasonable idea of adding target information into the model (Wang et al. 2014). To mimic the interactions between compounds and proteins, separated groups of weights were assigned to the compound features and protein features and then fed into the first hidden layer, respectively. In 2015, Wallach *et al.* introduced their DL model, AtomNet, to predict binding affinity for selecting active compounds for drug discovery(Wallach, Dzamba, and Heifets 2015). AtomNet was the first DL model to adopt CNN for small molecular binding affinity prediction. In AtomNet, a novel approach to combine both ligand and target structure information was used. However, AtomNet required the 3D structures for both ligand and target protein containing the location of each atom involved in the interaction at the binding site of the target. Recently, Wan and Zeng published their new model for compound-protein interaction prediction using DL methods, in which they adopted a widely used technique in NLP studies called feature embedding (Wan and Zeng 2016). In their model, both the ligand information (molecular fingerprints) (Rogers and Hahn 2010) and protein sequence were embedded into multi-dimensional vectors. Following the embedding process, a sequence of fully-connected layers consisting of rectified linear units (ReLUs) was constructed(Nair and Hinton 2010).

Besides predicting target selectivity and DTIs, DL methods have been adopted to predict ADMET properties. In 2013, Lusci *et al.* reported their model for predicting aqueous solubility using DL architecture (Lusci, Pollastri, and Baldi 2013). They segmented small molecules into atoms and bonds to build a digraph by sequencing those atoms and linking them using their corresponding bonds, and then put the contracted graph into the RNN model. In 2015, Shin *et al.* published their model developed using the DL method to predict the absorption potential of small molecules(Shin et al. 2016). In-vitro permeability data of 663 small molecules from the human colorectal carcinoma cell line (Caco-2) were used as training data, and 209 molecular descriptors were calculated using CDK toolkits based on their 2D structures (http://www.rguha.net/code/java/cdkdesc.html). Without using any specialized architecture, four layers of fully connected neural networks were generated to extract and transform the input information and finally classify the absorption potential of the input compound. DL methods were also helpful in predicting the toxicity of small molecules in the *Tox21 Data Challenge* launched by the NIH, EPA, and FDA. Unterthiner and colleagues reported their DL-based models for toxicity prediction in 2015 (Unterthiner et al. 2015). Multiple types of molecular features, such as different fingerprints and chemical properties, were tested and compared in their study. Forty thousand input features and a considerable number of hidden layers were adopted in their models. The average performance of their DL-based models was good in multi-task testing, showing that the DL algorithm was quite robust regarding training data, parameters, and tasks. Recently, Pereira *et al.* proposed their DL-based protocol for docking-based virtual screening (Pereira, Caffarena, and Dos Santos 2016). Their model used both ligand information and the interactive amino acids from docking to optimize the docking results. The input data were the distributed representation (Hinton 1984) of the compound-protein complexes generated using the embedding technique, followed by a three-layer convolutional neural network.

A lot of the earlier DL attempts in the drug discovery field had been using human-engineered features like molecular descriptors and fingerprints. In such cases, the characteristic of DL as representation learning, which allows DL to engineer molecular features directly from data automatically, is largely missing. Yet that is possibly the most crucial aspect that distinguishes DNNs from traditional ML algorithms. It is nice to see that more recent publications have demonstrated that learning directly on 'unprocessed' chemical data may also be a viable strategy. A work using 'unprocessed' chemical data on convolutional neural networks was published by Yao and Parkhill(Yao and Parkhill 2016). Notably, they used electron density from the 3D small molecules, rather than 2D molecular fingerprints or physical-chemical properties, as the input data and developed a 3D convolutional neural network model to predict the Kohn-Sham kinetic energy of hydrocarbons. Bjerrum reported a DL model using LSTM cell-based NN (Bjerrum 2017). The innovative part of his research was that he used SMILES (Weininger 1988) enumeration, a single line text uniquely representing one molecule, as the raw input data in the model. Another research from Goh et al. tried to use 2D molecule drawing images of molecules as the input data of a CNN model to predict chemical properties (Goh et al. 2017a, 2017b). They also compared their method to a CNN model using conventional molecular features as the input features. The model constructed using their image-based input features slightly outperformed traditional molecular features.

More recently, with the development of unsupervised learning and generative NNs, the application of those generative models using DL algorithms has seen progress. Kadurin *et al.* developed a seven-layer generative AAE model for screening compounds(Kadurin et al. 2017). Unlike a standard screening method using the QSAR model, their model extracted features from the input molecular fingerprints of 6,252 training molecules and generated new fingerprint vectors for potential selective compounds using a non-supervised generative model. Then they screened those selected outputs vectors against an extensive library of 72 million compounds from PubChem (Kim et al. 2016) and predicted 320 compounds as potential compounds, in which 69 were identified as actual hits experimentally. Besides selecting novel compounds using auto-encoders, there were several attempts to generate novel compounds using other deep generative networks. Segler *et al*. introduced their generative models for designing novel focused libraries using RNNs, achieving a satisfactory performance to complete the *de novo* drug design cycle (Segler et al. 2017).

Similar methods were developed for the de-novo library design by Olivecrona *et al.*, with the novelty of adding RL (Schmidhuber 2015) into the method (Olivecrona et al. 2017). Guimaraes *et al.* adopted GANs, and RL to construct a generative model for generating different types of molecules using their SMILES data, giving a novel idea of designing novel compounds using state-of-the-art unsupervised DL methods (Guimaraes et al. 2017).

1.2.3 Deep Learning versus Traditional Machine Learning

In the era of big data, DL has a significant advantage compared to other traditional algorithms, which are also considered 'shallow' in their learning capability compared to DL algorithms (Goh, Hodas, and Vishnu 2017). DL algorithms belong to the representation learning class, which can handle raw data and automatically extract features as the representations needed for further detection or classification (LeCun, Bengio, and Hinton 2015). As state-of-the-art machine learning algorithms, DL algorithms have been challenged by comparing them to other 'shallow' machine learning algorithms (Goh, Hodas, and Vishnu 2017). Winkler *et al*. recently reported comparing their Bayesian regularized neural network (BNN) models and the DL models generated by Ma *et al*. using the same KAGGLE data set from Merck (Ma et al. 2015). They showed that 'shallow' neural networks with one single hidden layer could perform as well as DNNs with more hidden layers, given sufficient training data in QSAR or QSPR modeling (Winkler and Le 2017). A similar conclusion was generated by Capuzzi *et al*. from the comparison with Tox21 data (Capuzzi et al. 2016). It appears that those results were consistent with the universal approximation theorem (Gao and Xu 1993), inferring that DL algorithms may not have superiority over regular 'shallow' networks. Those results may overturn our preconception that novel DL should be better than traditional 'shallow' machine learning methods. In fact, both DL and Shallow Learning have their places for supervised learning with the final purpose of classification or regression (Winkler and Le 2017; Ma et al. 2013).

Schmidhuber. et al. suggested that the primary deficiency of most traditional machine learning methods is that they have a limited ability to simulate a complicated approximation function and generalize to an unseen instance (Schmidhuber 2015). NNs have advances in QSAR/QSPR modeling (Baskin, Winkler, and Tetko 2016), and the universal approximation theorem proves its advanced capacity on approximation. Shallow NNs can generalize to new data very well in most cases, given sufficient diverse data. Given the same descriptors and training data, both types of neural networks generate similar quality models. However, DNNs can generate complex abstractions of the descriptors. As mentioned, the essential features of DL methods that distinguish them from shallow neural networks are not only the emphasis on the depth of the network but also the emphasis on feature learning. Compared to the shallow neural networks that need to 'manually' select the features, DL methods can learn features from data by constructing nonlinear network models to extract latent information of the 'big data.' In the early QSAR/QSPR studies, descriptors were designed manually, not capturing all the features impacting the QSAR/QSPR response surface (Winkler and Le 2017). As a result, one tiny change in the values of those descriptors could lead to a significant difference in the activity. Such phenomena are called activity cliffs (Maggiora 2006), which is a common concern in QSAR modeling. The presence of activity cliffs is also highly correlated with the distribution of the activity responding surface used for training the model, which refers to small molecular feature learning and protein target feature extraction. Research has shown that adding protein features makes the DL model perform better(Lenselink et al. 2017; van Westen et al. 2011). From the aspect of DL modeling, both the choice of different DL architectures and the configuration of hyper-parameters are very vital for achieving good performance.

Moreover, other differences between DL methods and traditional shallow machine learning methods were explored by other researchers. Lenselink *et al*. found that DL methods and traditional shallow machine learning methods performed similarly on randomly split data; however, they had significant differences when the data were divided by congeneric chemical series (such as by the nature of publishing) (Lenselink et al. 2017). They thought that compounds published together were usually very similar in chemical structure and splitting in such a way could make the validation more in line with the experiments performed. Because of the advance of feature learning, DL can reach a high identification accuracy under the premise that the training set should contain a tremendous amount of data. With limited data, the DL techniques cannot achieve an unbiased estimate of the generalization, so that they may not be as practical as some traditional shallow machine learning methods(Winkler and Le 2017; Schmidhuber 2015). Also, with the rapid increase of time complexity because of the complication of the network architecture, stronger hardware facilities and advanced programming skills are required to grant the feasibility and effectiveness of DL methods. In addition, although DL methods usually have outstanding performance in practice, the tuning of the hyper-parameters in DL modeling is often tricky. Also, it is hard to know how many hidden layers and nodes could be enough to establish the best simulation without redundancy for a specific DL modeling. Finally, the strategy for unsupervised learning in DL is inspiring but still falling far behind (Schmidhuber 2015). In real-world applications, especially in drug discovery, most of the data are non-labeled data, with plenty of information contained. Exploring and developing novel unsupervised learning methods using DL methods and mining useful information from those data is still difficult.

Overall, modern pharmaceutical science and drug discovery will become more and more complex. Designed for intricate simulation, DL should have the capability to handle that complexity. Also, with DL methods, we should not restrict ourselves to the traditional predictions on biological activities, ADMET properties, or pharmacokinetics simulations; but it may also be possible to systematically integrate all the data and information and achieve a new level of Pharmaco-Analytics AI technology in drug discovery.

1.3 Causal Inference and Bayesian Networks

1.3.1 Introduction to Causal Inference

Machine learning and deep learning are powerful for predictive tasks; however, they also have drawbacks. The major drawback of machine/deep learning algorithms is that they treat all variables as equals and do not necessarily distinguish between causes, outcomes, and confounders. This problem is especially pronounced when predicting the impact of treatment interventions using nonrandomized, observational data in health care research (Prosperi et al. 2020). To address this problem, the causal discovery was developed to provide a concise way of representing and quantifying causal relationships among variables (Glymour, Zhang, and Spirtes 2019b). Now, the causal discovery has already been applied in different areas such as genomics, ecology, epidemiology, space physics, clinical medicine, neuroscience, and other domains (Sachs et al. 2005; Schadt et al. 2005; Sinoquet 2014; Vasimuddin and Aluru 2017).

The traditional way to discover causal relations is to use interventions or randomized experiments, which are expensive, time-consuming, or sometimes impossible. Causal discovery offers a different pathway for revealing causal information by analyzing complex observational data using Bayesian networks (BN) (Spirtes, Glymour, Scheines, Kauffman, et al. 2000). BN is a specific type of probabilistic graphical model (PGM) that represents the conditional dependencies among variables by a directed acyclic graph (DAG) and a set of parameters. In the PGM, the dependence structure among variables is described in a graph of nodes and edges, in which each node represents the variable, and the edge represents the dependency between two variables. Specifically, in the causal BN, the directed edge with an arrow pointing from one node to another represents the causal relationship between those two variables. In other words, if A →

B is in the directed graph of BN, then interventions on A will directly change the value or distribution of B.

1.3.2 Searching Causal Structures

When training BN, the pairing of a DAG and a joint probability distribution on values of its variables is subject to the constraint that a graphical condition called d-separation must imply conditional independence in the probability distribution (Geiger, Verma, and Pearl 1990). For example, in three disjoint sets of variables (X, Y, Z), if all paths between X and Y are blocked by Z in the DAG, then variable sets X and Y are d-separated conditional on Z. The d-separation and its connection with conditional independence has an equivalent in the local Markov Condition, which is that any variable X in a DAG is independent of its non-descendants given its parents (Glymour, Zhang, and Spirtes 2019a). This is an essential precondition for causal inference called causal Markov assumption (Hausman and Woodward 1999).

The causal dependence relations among variables can be identified by either acquiring from domain knowledge, learning from experimental and observational data, or combining of both. The domain knowledge is commonly used at the beginning of model construction, providing a valuable clue and starting point for the causal learning procedure. Then the following learning BN structures from data serve as an excellent approach for generating causal hypotheses that explain the underlying distribution of the data. The essence of causal structure learning is the statistical estimation of parameters describing the graphical causal structure. This computational estimation usually requires iterative or Monte Carlo procedures, also called causal structure searching. There are mainly two types of search algorithms: constraint-based algorithms and score-based algorithms (Triantafillou and Tsamardinos 2016). The constraint-based algorithms use hypothesis testing to estimate the dependence or conditional independence relationship of each variable implied by the data and then use these relations to construct a directed graphical model. Examples of constraint-based algorithms are the PC algorithm (Harris and Drton 2013;

Casini and Baumgartner 2020) and the Fast Causal Inference (FCI) algorithm (Spirtes 2001). The score-based search algorithms are mainly used in situations without confounders. Those algorithms aim to use a pre-defined performance score function (e.g., BIC score) to find the Markov equivalence class of graphs that most closely entails the set of conditional independence relations in the data. However, the causal structure search is a data-driven process, and sometimes it can be tricky. Therefore, it is often necessary to refine the identified causal relationship based on the domain knowledge.

Conventional statistical analysis and machine learning methods can query the posterior probability of a target variable Y based on the probability distribution of the observational variables X, P(Y|X=x). However, such posterior probability is based on associations between X and Y; therefore, it can be not accurate because of confounding variables or biases that cause both the X and Y. In addition, in the data collection and analysis procedure, the selection of variables and covariates are usually arbitrary, which may lead to redundancy, incomplete blockage of confounding, as well as additional sources of bias (Pearl 1998). In comparison, Causal BN can identify causations rather than associations by making inferences on intervention effects. Instead of evaluating P(Y | X=x), causal BN adopts the *do-calculus* method to estimate the P(Y | do(X=x)), which means the condition that the value of X is artificially set to x regardless of its causal parents' values (Tucci 2013; Pearl 1995). Causal inferences from observational data can be achieved by controlling for a set of confounding variables. In the situation with unobserved confounders, P(Y | do(X=x)) can be estimated using the 'front-door' criterion by introducing a mediator. At the same time, the latent confounding and bias between two variables (X, Y) can also be eliminated by controlling for a set of variables that block all ‘back-door’ paths (Pearl 1995). In this way, causal analysis using BN provides a systematic and rational approach for identifying confounders.

1.3.3 Perspective of Causal Inference Analysis

Data-driven methods have been highly successful in Pharmaco-Analytics research, especially for diagnostic and predictive tasks (Jing et al. 2018a). However, machine learning methods are speculative interventional and decision-supporting tasks, which are more common in clinical research and precision medicine. In addition, observational data collected retrospectively are usually collected with various biases; therefore, the estimation of causal effects from observational data requires thoughtful handling of potential biases and confounding. The causal analysis attempts to address those problems by estimating the interventional effects under the counterfactual conditional probabilities. By using conditional independence tests and causal structure search algorithms, causal analysis can reproduce the potential causal mechanisms from observational data even in the presence of bias and confounding (Prosperi et al. 2020). However, the computation power required for causal graph searching is super-exponential in the number of observed variables and hidden variables, making the exhaustive causality search computationally unfeasible. The exploration of combining deep learning with causal methods provides fascinating new insights to address this issue (van Amsterdam et al. 2019).

Moreover, the interpretability of the causal model helps explain of variables and mechanisms, making up the ‘black-box’ model property of machine learning (Rudin 2019). In the future, intervention models could be more successful in Pharmaco-Analytics research and applications, in line with the current standards for diagnostic and discriminative models. And intervention models are expected to be integrated into clinical guidelines to facilitate future AI development in drug discovery and patient healthcare.

CHAPTER 2. AI ML/DL PHARMACO-ANALYTICS FOR TARGET IDENTIFICATION

2.1 DAKB-GPCRs Platform for GPCRs-related Drug Abuse Research

2.1.1 Significance and Background

Drug abuse (DA) and addiction are complicated neurological disorders. The main symptom is compulsive behavior pertaining to drug craving, seeking, and persistent use despite serious adverse consequences. According to Drug Facts (NIDA 2020), the total overall costs of substance abuse in the U.S. exceed $740 billion annually, including approximately $193 billion for illicit drugs (National Drug Intelligence Center (NDIC) 2017; Birnbaum et al. 2011), $300 billion for tobacco (Services ; Xu et al. 2015), and $249 billion for alcohol (CDC 2016) (**Figure 2.1.1A, B**). Thus, research into DA prevention and treatment is a high priority.

To accelerate and facilitate DA-related research, we constructed a self-serve online library named GPCRs-specific chemogenomics knowledgebase for DA research (DAKB-GPCRs) containing information about drug abuse-related protein targets and small molecules as well as tools and algorithms for computational data analyses and visualization of those data. In our established drug abuse chemogenomics knowledgebase (DA-KB) (Xie et al. 2014; Xie et al. 2016) regarding protein targets of abused drugs, we found that 86 out of 258 proteins (33.3%) are G protein-coupled receptors (GPCRs) (Venter 2001), which GPCRs are targeted by approximate 40% of marketed drugs worldwide. However, only 39 GPCRs have been published with crystal structures in the last two decades (Xie et al. 2016). Moreover, among 86 GPCRs related to DA, only 18 GPCRs' (21%) experimental structures are available (**Figure 2.1.1C, D**). Recently, homology modeling with a sequence identity of 30% or greater (Chothia and Lesk 1986) and/or multiple conformations-based docking is becoming a powerful tool and strategy for structural study and drug discovery (Kitchen et al. 2004). Therefore, we constructed this DAKB-

GPCR platform for drug abuse researchers/scientists who have no experience in programming, statistics, or in silico drug design to facilitate their DA research.

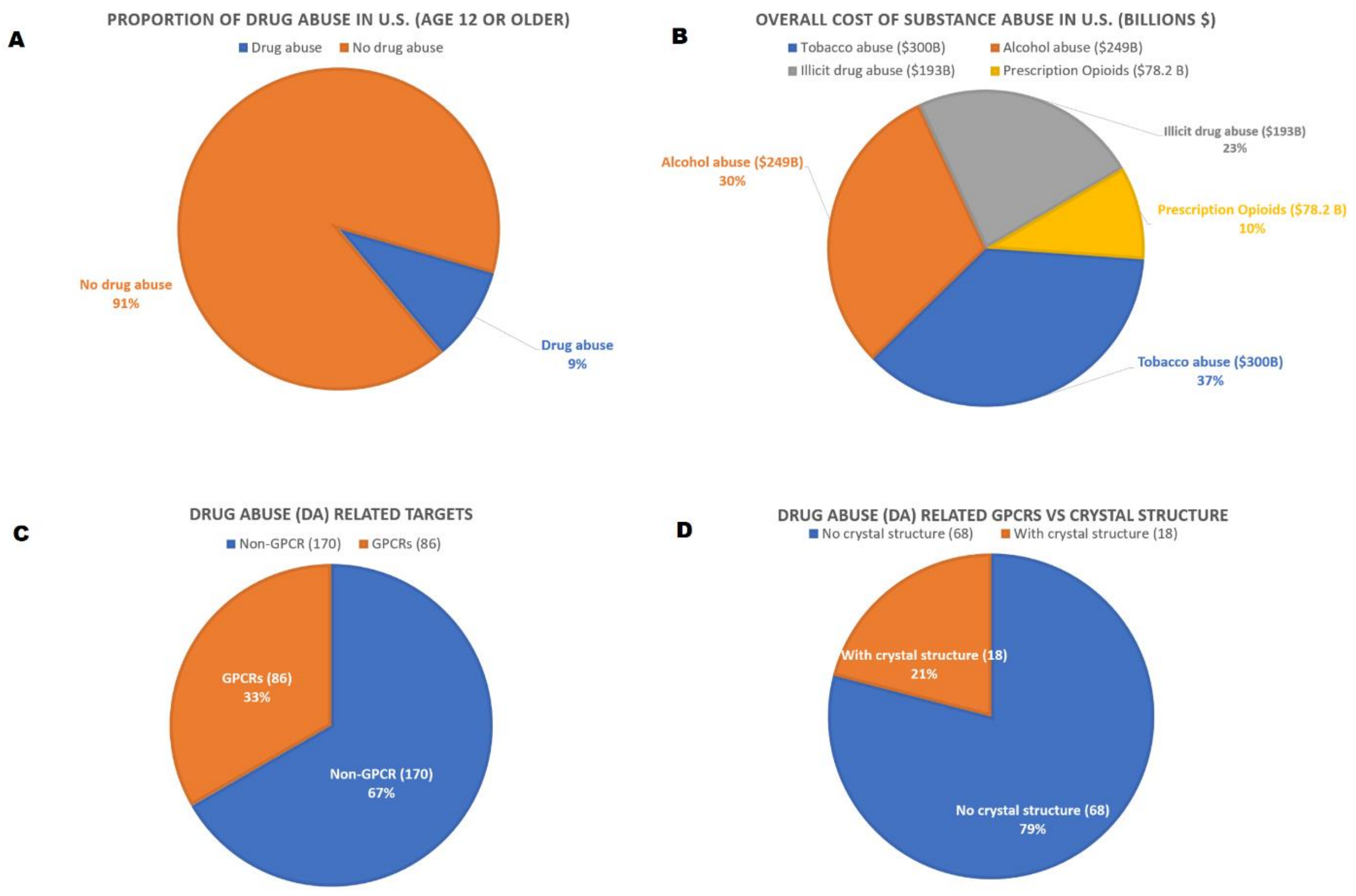


Figure 2.1.1 The influence and current research status of drug abuse in the U.S.

2.1.2 Methods

2.1.2.1 Data collection and content

*Genes/*Proteins. Genes/proteins related to drug abuse were collected from public databases such as Ensembl (Zerbino et al. 2017), UniProt (The UniProt 2017), KEGG (Kanehisa and Goto 2000), GPCRdb (Isberg et al. 2017), and NCBI Protein Database (Coordinators 2017). Available crystal structures of GPCRs were retrieved from the Protein Data Bank (PDB) (http://www.pdb.org/pdb/).

Drugs and Chemicals. ChEMBL database (version 23) was used in our work (Gaulton et al. 2018). The experimental data for each small molecule against its respective target proteins was collected using the text mining technique and cleaned by manual inspection. Bioactivity data from different resources were normalized using the same standard. Especially, small molecules with IC50 lower than 1 µM towards a GPCR target were regarded as the active compounds, while those larger than 10 µM were considered inactive compounds. A training dataset that consists of both active and inactive compounds of each GPCR will be used for prescreening and similarity search using TargetHunter (Wang et al. 2013).

Homology Models. The sequences of the human GPCRs were collected from the UniProt (http://www.uniprot.org/uniprot/) website. Modeler 9.18 (Webb and Sali 2016) was used to construct the homology models. After generating the 3D models of protein targets, SYBYL-X 1.3 ("SYBYL-X" 2010) was adopted to carry out the energy minimization. Then the model after energy minimization was selected for further molecular dynamics simulation.

Molecular Dynamics Simulation. To sample the conformations of each homology model or crystal structure, 10ns molecular dynamics simulations were performed. The NAMD package (version 2.9b1) (Kalé et al. 1999) using the CHARMM27 (Brooks et al. 2009) force field was applied to the MD simulation. All the systems were equilibrant after 5ns, so we analyzed the data based on the trajectory file from 5ns to 10ns. Ten conformations with the lowest energy during the last 5ns MD simulation (from 5ns to10ns) were selected for prescreening.

Prescreening. Ten models of each GPCR obtained after MD simulation was utilized to perform the prescreening against its training dataset (see Drugs and Chemicals). Three conformations of each GPCR with the best ROC curve were selected and integrated into our platform. Taking adrenoceptor alpha 1d (ADRA1D) as an example, Supplementary Figure S2.1 shows the curves of its best three conformations. It shows the statistical results of Model 1: the docking score of 6.1539 was chosen as the best threshold because the docking scores of 81.111% (1-0.28889) inactive compounds were lower than 6.1539, while the docking scores of 78.498% (0.78498) active compounds were higher than 6.1539. The thresholds of docking scores of other models were 6.5873 (**Figure S2.1B**) and 6.709 (**Figure S2.1C**), respectively.

2.1.2.2 Database Implementation and cheminformatics tools development

Database Infrastructure. A query compound can be submitted using JSME Molecular Editor (Ertl 2010). DAKB-GPCRs were implemented based on our established molecular database prototype CBID (http://www.cbligand.org/cbid/) using SQLite database management system (https://sqlite.org/) and Kestrel HTTP server (https://github.com/aspnet/KestrelHttpServer) with Apache HTTP server (https://httpd.apache.org/) as its reverse proxy server. The overview of our design for DAKB-GPCRs is depicted in Figure 2.1.2.

TargetHunter. DAKB-GPCRs integrates our online target-identification service TargetHunter (Wang et al. 2013) for predicting the potential off-targets for submitted compounds. TargetHunter exploits a common principle of medicinal chemistry: compounds with structural similarities often have similar physicochemical properties and biological profiles. For each query compound, TargetHunter calculates the similarity (from 0.0-1.0, totally different to 100% similar) with its known active compound's dataset that was collected from Drugs and Chemicals.

HTDocking. DAKB-GPCRs adopts our online high-throughput molecular docking technique-HTDocking (Liu et al. 2014; Xu et al. 2016; Zhang, Wang, et al. 2016), for identifying possible interactions between protein targets and small molecules. Three different conformations for each GPCRs were selected from MD sampling and validated by prescreening. For each query compound, HTDocking will automatically dock it into three different conformations and generate docking scores. A higher docking score indicates that the protein is more likely to be the candidate target of the queried small molecule.

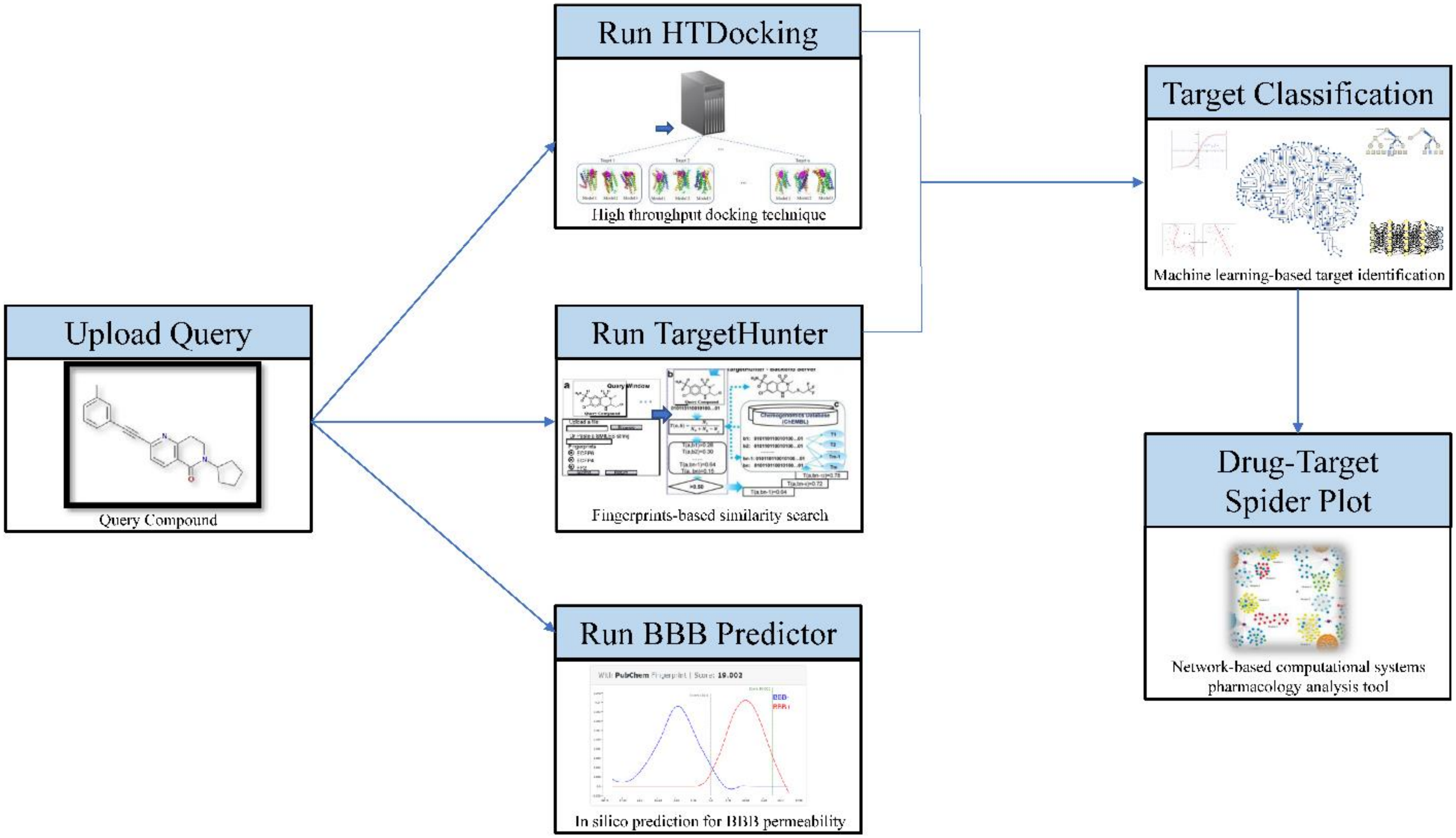

Upload Query
Query Compound
Run HTDocking
High throughput docking technique
Run TargetHunter
Fingerprints-based similarity search
Run BBB Predictor
In silico prediction for BBB permeability
Target Classification
Machine learning-based target identification
Drug-Target Spider Plot
Network-based computational systems pharmacology analysis tool

Figure 2.1.2 The overview design for DAKB-GPCRs. When submitting a query compound, our platform will automatically generate three docking scores (each protein has three confirmations) via HTDocking (high-throughput molecular docking) and one similarity score via TargetHunter (predicting the potential targets/off-targets of submitted compounds based on molecular similarity) for each GPCR. Then target classification for each GPCR will be generated by our deep/machine learning algorithms that combined both docking scores and similarity scores. Spider-Plot, our new mapping tool that is similar to Cytoscape was used to map out the molecule-proteins networks. Last but not least, the blood-brain barrier (BBB) predictor will visualize the results of the query compound.

Machine Learning-based Target Classification. For each GPCR, a dataset consisting of three docking scores and one similarity score for each known compound was trained to build the target classification models using established machine learning (ML) algorithms. The compound dataset collected for each target (as discussed in section 1.2) was used for training and testing the classification models. Molecular docking scores and molecular fingerprint similarity scores were computed using the protocols discussed in sections 2.2 and 2.3, respectively. Four ML algorithms were adopted by our classification models: logistic regression (Kleinbaum et al. 2002), support vector machine (SVM) (Steinwart and Christmann 2008), random forest (RF) (Breiman 2001b), and artificial neural networks (ANN) (Gardner and Dorling 1998). 10-fold cross-validation was used to assess the predictive capability of all the models. The performance and evaluation for those classifications are listed in **Figure 2.1.3**. The performance of the ANN models was not robust in this case; thus, those models were eliminated. Therefore, the final prediction for each protein target was determined by the classifications from the three selected models and their confidence levels.

Spider Plot. Based on the target classification, our online tool Spider Plot visualizes the molecule-protein interaction network (Chen et al. 2019). The average docking scores are displayed as connection labels, and the entire network graph can be exported as an image file.

BBB Predictor. The blood-brain barrier predictor was integrated into DAKB-GPCRs. It predicts whether or not a query compound can move across the BBB to the central nervous system (CNS). The BBB predictor is available for access from http://www.cbligand.org/BBB.

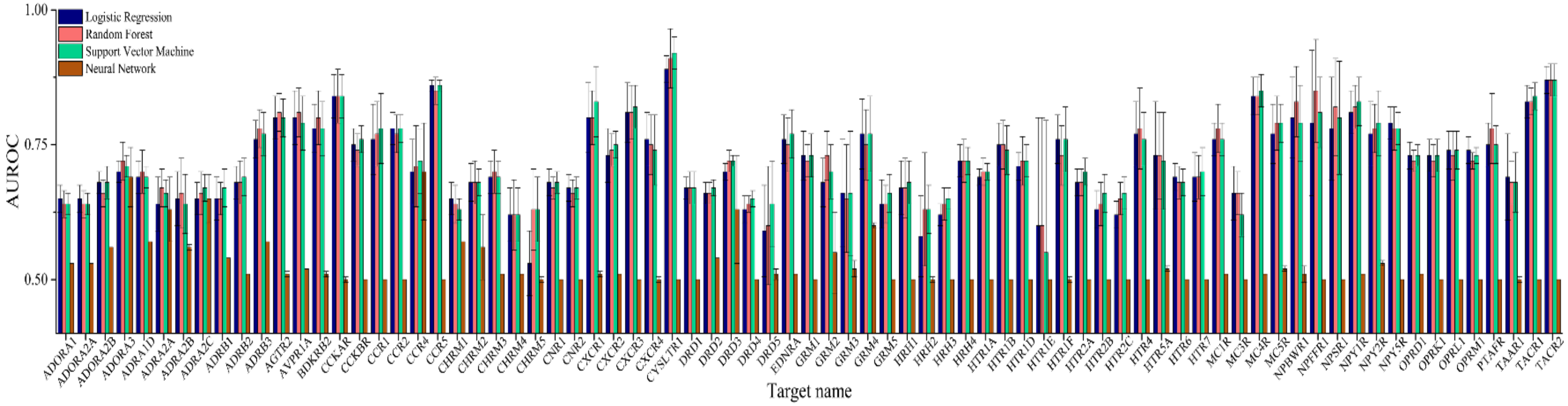


Figure 2.1.3 The evaluation of the machine learning classification models

2.1.3 Results and Discussion

To elaborate how the DAKB-GPCRs website can be used to facilitate research, we use a published compound (CHEMBL1779871, a positive allosteric modulator of human mGluR5), to showcase the functionalities step by step (**Figure 2.1.4**).

Home Page. The DAKB-GPCRs can be accessed from https://www.cbligand.org/g/dakb-gpcrs. On the top of the home page (**Figure 2.1.4A**) is the navigation bar which contains the HOME button and the 'ALL TASKS' button. Clicking on the HOME button brings the user back to the home page, while the 'ALL TASKS' button directs the user to the task list page.

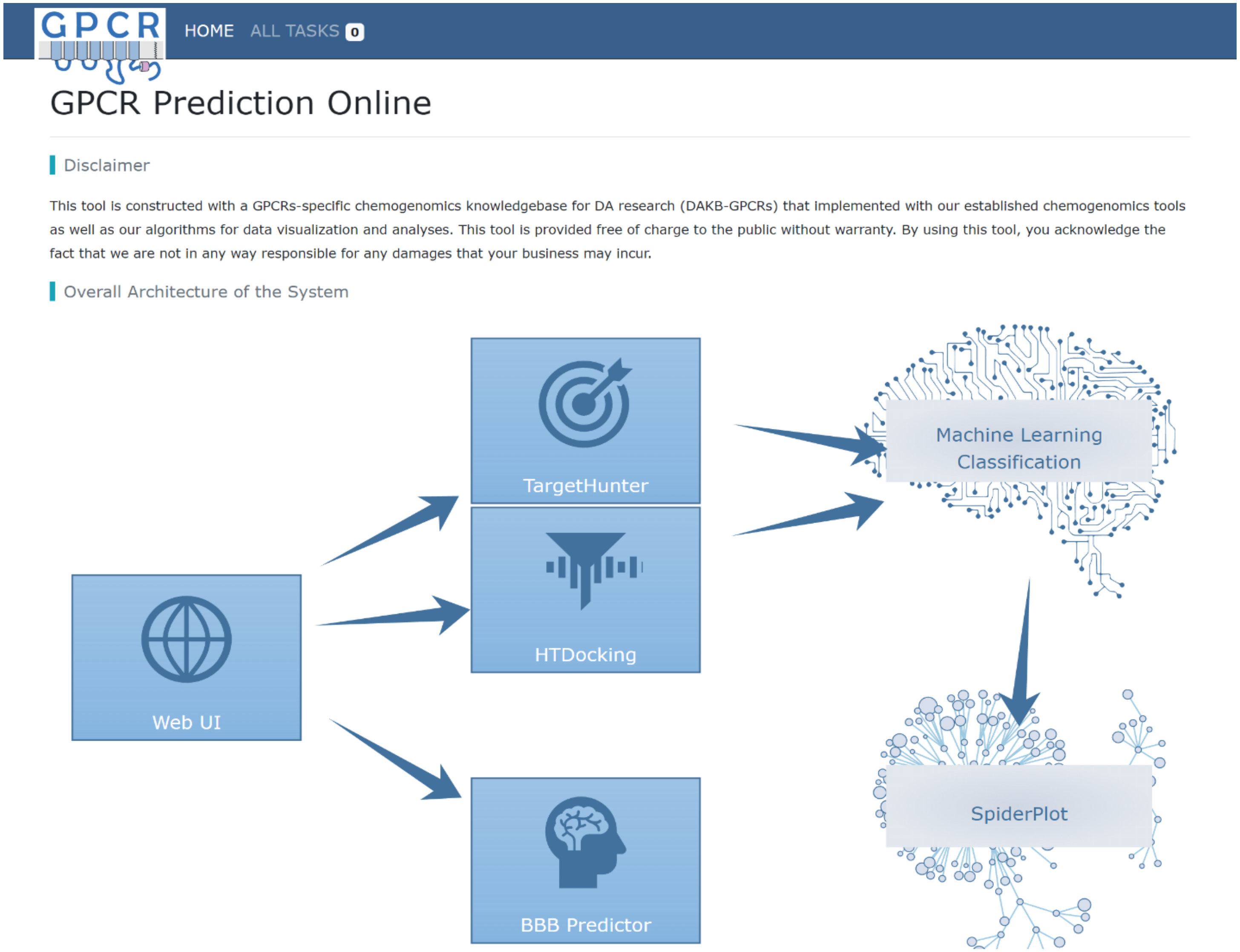
Task List Page. The task list page displays all the tasks submitted in a table layout. (**Figure 2.1.4B**) Completed tasks are shown as *Finished* in the status column, and ongoing tasks are shown as *Running*. Users can click on the task name to access the detailed information of the input compound and the output of the computation task. Users can initiate a new job by clicking on the *Create a new task* button in the upper right corner of the page.

*Start a New Task using Structure Query***.** Users can submit a query compound by drawing its 2D structure with JSME Molecular Editor or uploading a chemical file with the button (**Figure 2.1.4C**). A meaningful name for the task is also required. Clicking on the *Create Task* button will initiate a new job. Afterward, a background job worker who periodically monitors the task queue will allocate computation resources and dispatch the computation task. In **Figure 2.1.4C**, the structure of CHEMBL1779871 in SMILES format was uploaded.

*Detailed Task Page***.** After submitting the request, the website will automatically direct the user to the detailed task page (**Figure 2.1.4D**). This page shows the task information and the progress bar on the top showing the real-time progress of computation for target prediction. Following the task, information is the section of ligand information which includes 2D/3D structures, multiple structure files of the query compound, and various computed molecular fingerprints. Moreover, the results from the BBB predictor can be found on this page (**Figure 2.1.4E**).

*Output Page for Target Prediction***.** The output page can be accessed by clicking on the *See Detailed Output* button from the detailed task page. On the output page, each block presents three docking scores by HTDocking computed against three protein target models and a similarity score by TargetHunter (**Figure 2.1.4F**). The classification result is presented, shown whether the query compound is an active or inactive compound for this protein target, and the confidence level of the prediction is also provided. When clicking on the download button, the user can download the docking results of the submitted compound. The most similar compound within the compound library of a protein target can be seen from the *Best Match* field. After clicking the compound ID ↗, its comparison with the query compound is shown in a popup window (**Figure 2.1.4G**). When clicking on the target name, a window consisting of additional resources for the protein target (**Figure 2.1.4H**) will be shown, including a 3D interactive visualization, multiple links to other database websites, and models for direct downloading.

The *SPIDER PLOT* button on the top of the page leads the users to the drug-target network plotting tool for data visualization and analysis (**Figure 2.1.4I**). The validated and most promising predictions for the query compound are shown as green nodes connected with a solid line. In contrast, unvalidated predictions are shown as purple nodes connected with a dashed line. The name and structure of the query compound are placed in the center of the plot. As illustrated in Supplementary Figure S5I, GRM5 was predicted as the most promising target for CHEMBL1779871, which was consistent with the bioactivity data where CHEMBL1779871 is an active compound for GRM5 (EC_{50} = 7.5nM).



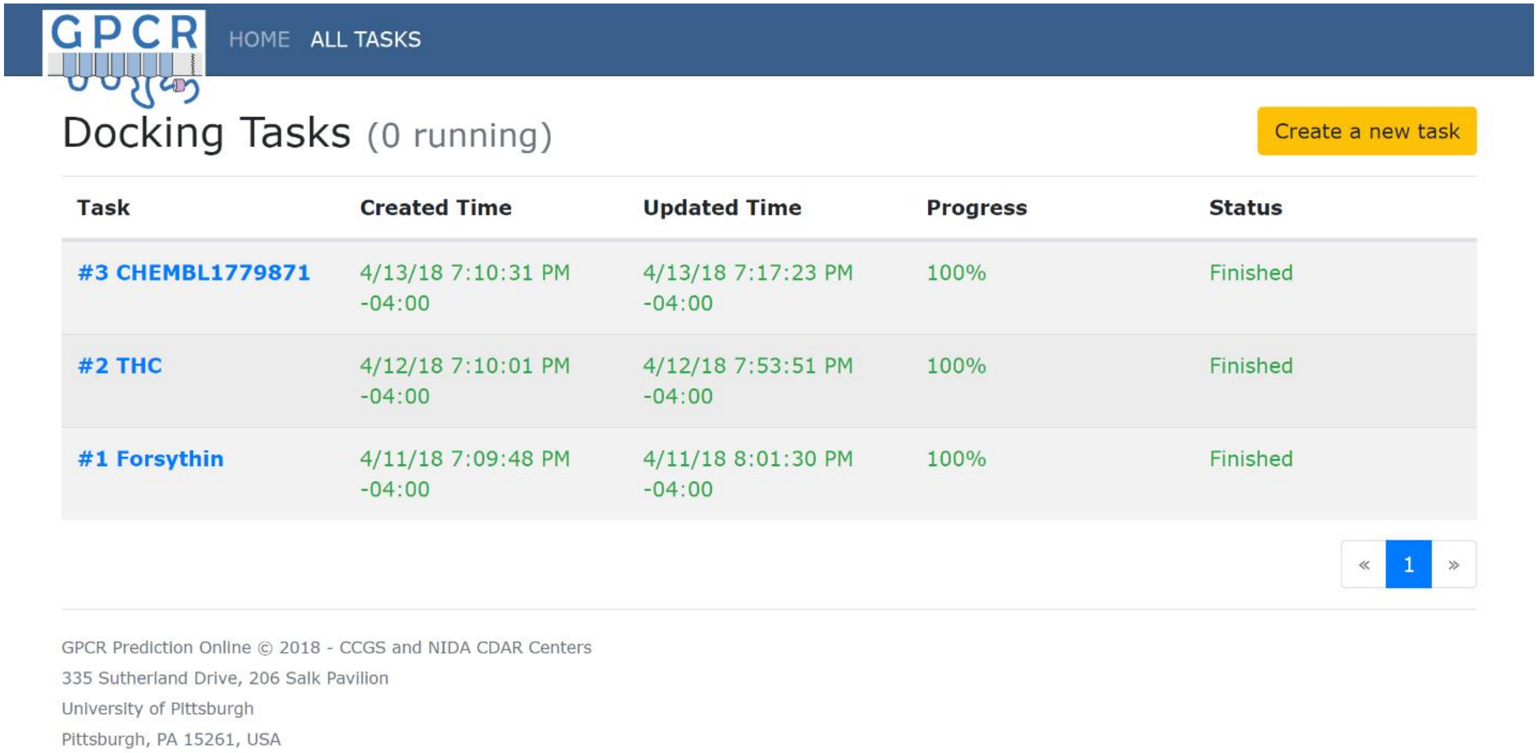
GPCR Prediction Online

Disclaimer

This tool is constructed with a GPCRs-specific chemogenomics knowledgebase for DA research (DAKB-GPCRs) that implemented with our established chemogenomics tools as well as our algorithms for data visualization and analyses. This tool is provided free of charge to the public without warranty. By using this tool, you acknowledge the fact that we are not in any way responsible for any damages that your business may incur.

Overall Architecture of the System

Figure 2.1.4A The home page of DAKB-GPCRs.

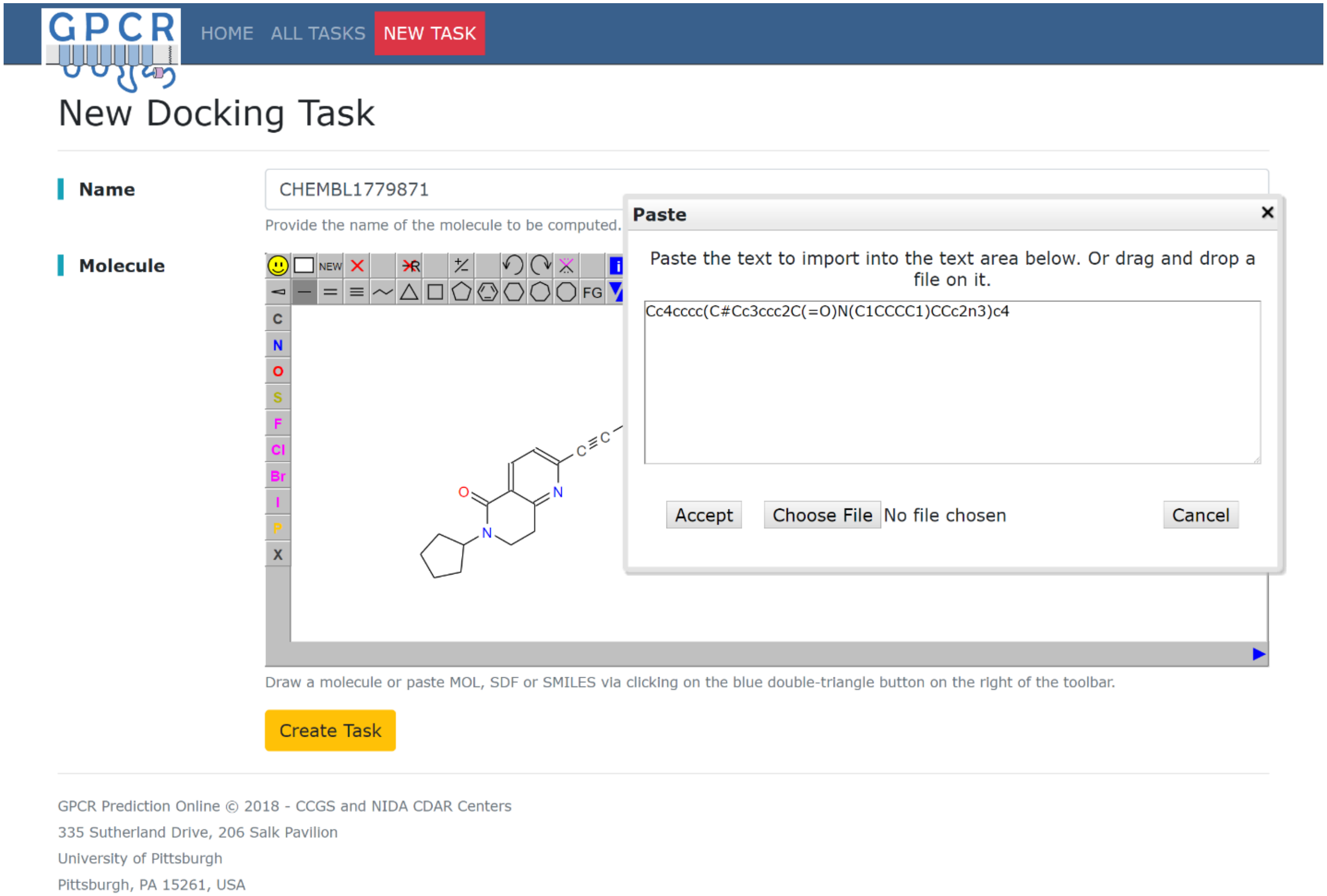


Docking Tasks (0 running)

Task	Created Time	Updated Time	Progress	Status
#3 CHEMBL1779871	4/13/18 7:10:31 PM -04:00	4/13/18 7:17:23 PM -04:00	100%	Finished
#2 THC	4/12/18 7:10:01 PM -04:00	4/12/18 7:53:51 PM -04:00	100%	Finished
#1 Forsythin	4/11/18 7:09:48 PM -04:00	4/11/18 8:01:30 PM -04:00	100%	Finished

GPCR Prediction Online © 2018 - CCGS and NIDA CDAR Centers
335 Sutherland Drive, 206 Salk Pavilion
University of Pittsburgh
Pittsburgh, PA 15261, USA

Figure 2.1.4B Tasklist page



New Docking Task

Name: CHEMBL1779871

Provide the name of the molecule to be computed.

Molecule

Paste

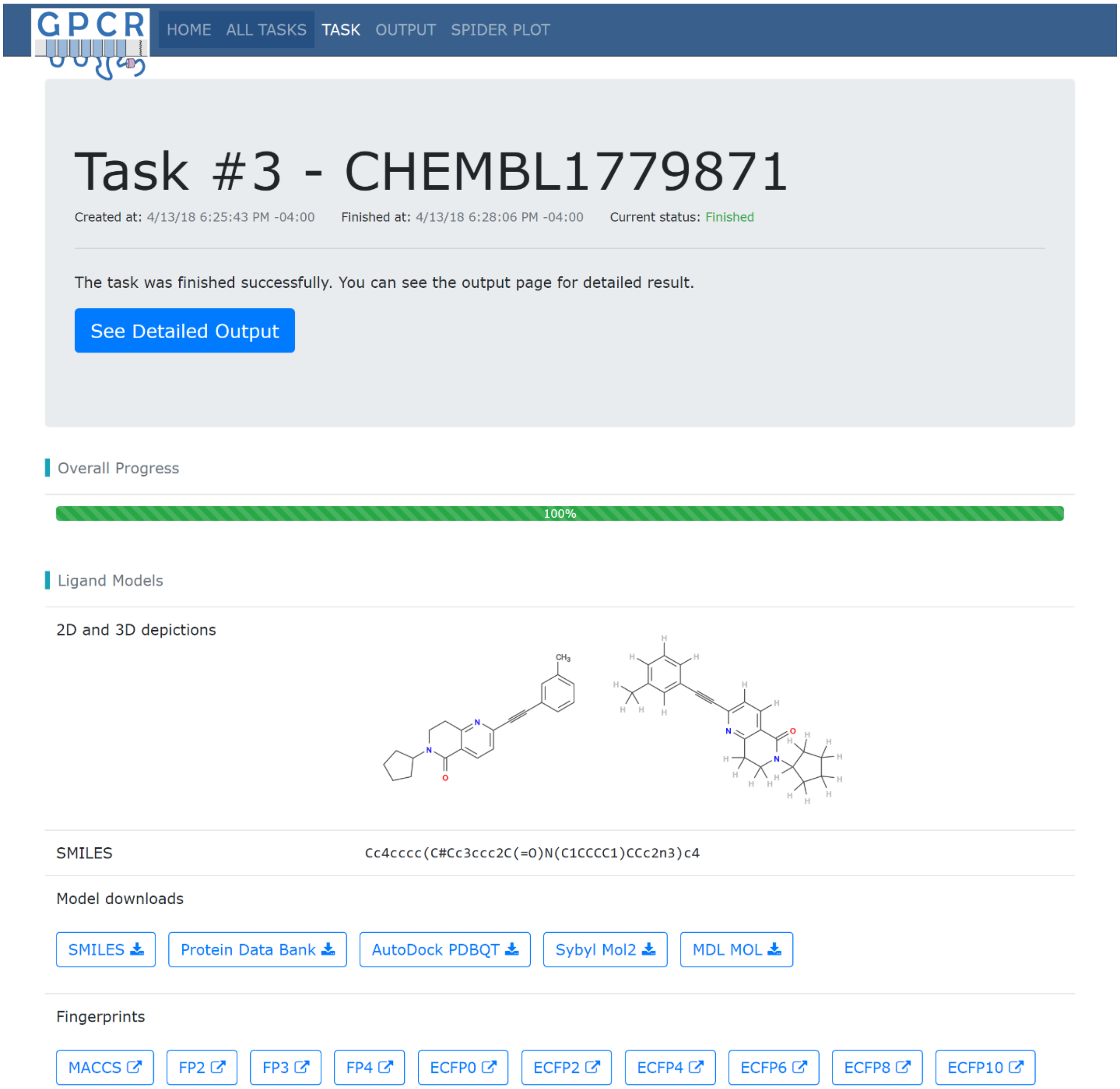
Paste the text to import into the text area below. Or drag and drop a file on it.

```
Cc4cccc(C#Cc3ccc2C(=O)N(C1CCCC1)CCc2n3)c4
```

Draw a molecule or paste MOL, SDF or SMILES via clicking on the blue double-triangle button on the right of the toolbar.

GPCR Prediction Online © 2018 - CCGS and NIDA CDAR Centers
335 Sutherland Drive, 206 Salk Pavilion
University of Pittsburgh
Pittsburgh, PA 15261, USA

Figure 2.1.4C Start a new task by uploading the structure information of the query compound using mol/sdf file or SMILES string.



Task #3 - CHEMBL1779871

Created at: 4/13/18 6:25:43 PM -04:00 Finished at: 4/13/18 6:28:06 PM -04:00 Current status: Finished

The task was finished successfully. You can see the output page for detailed result.

Overall Progress

Ligand Models

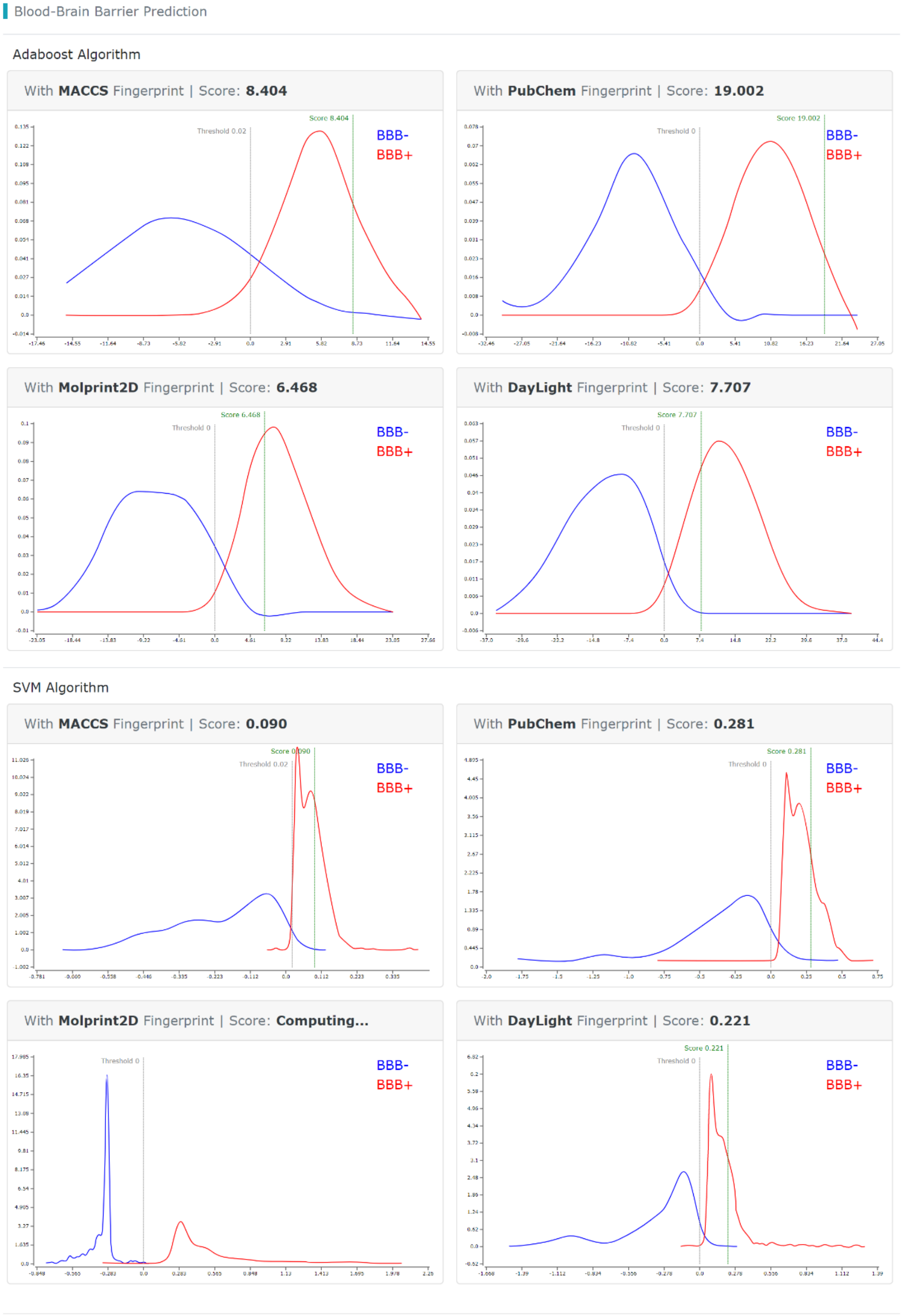
2D and 3D depictions

SMILES Cc4cccc(C#Cc3ccc2C(=O)N(C1CCCC1)CCc2n3)c4

Model downloads

Fingerprints

Figure 2.1.4D Detailed task page for a given compound



Blood-Brain Barrier Prediction

Adaboost Algorithm

With **MACCS** Fingerprint | Score: **8.404**

With **PubChem** Fingerprint | Score: **19.002**

With **Molprint2D** Fingerprint | Score: **6.468**

With **DayLight** Fingerprint | Score: **7.707**

SVM Algorithm

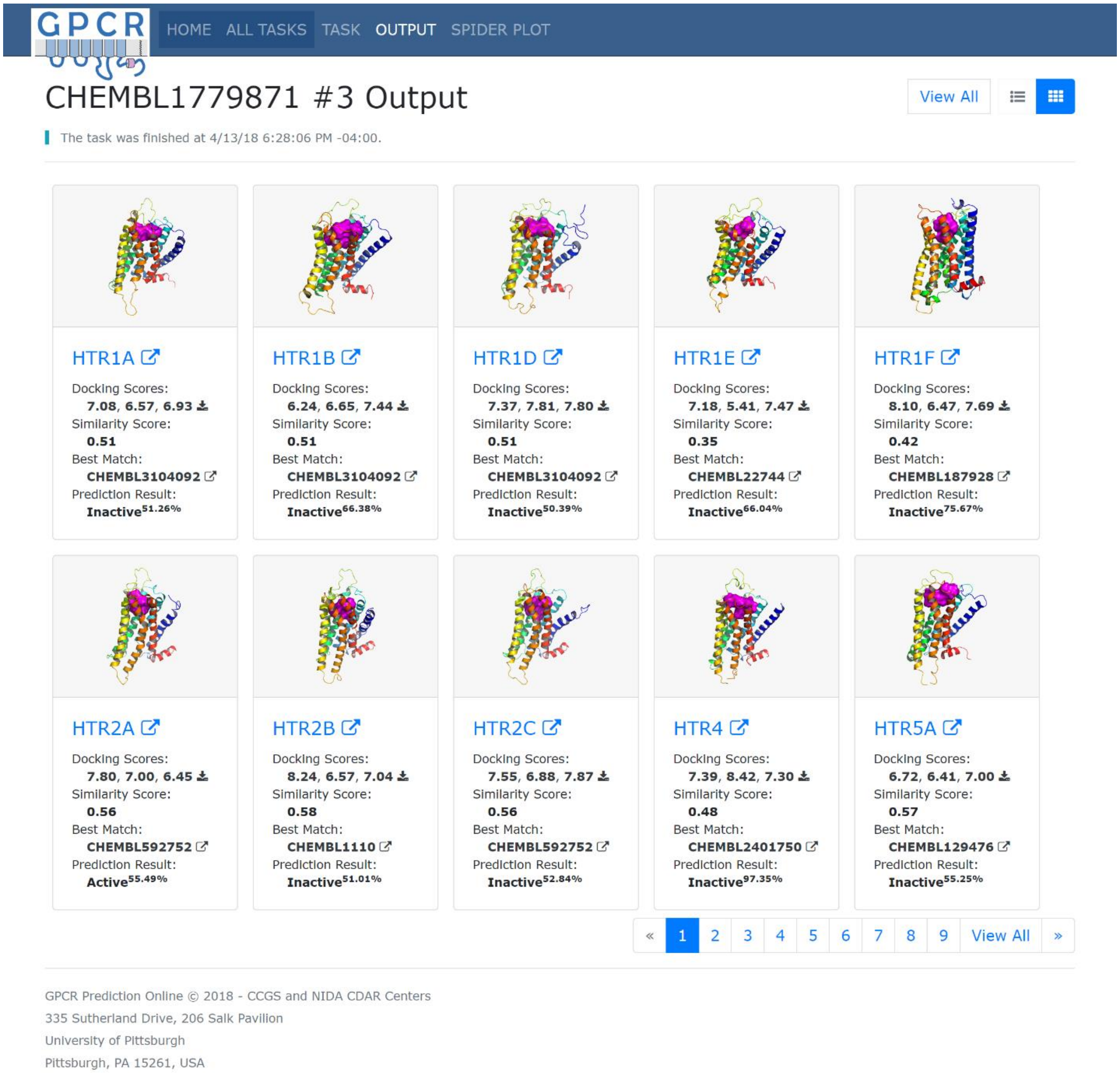
With **MACCS** Fingerprint | Score: **0.090**

With **PubChem** Fingerprint | Score: **0.281**

With **Molprint2D** Fingerprint | Score: **Computing...**

With **DayLight** Fingerprint | Score: **0.221**

Figure 2.1.4E Result from the BBB predictor for CHEMBL1779871



CHEMBL1779871 #3 Output

The task was finished at 4/13/18 6:28:06 PM -04:00.

HTR1A
Docking Scores:
7.08, 6.57, 6.93
Similarity Score:
0.51
Best Match:
CHEMBL3104092
Prediction Result:
Inactive$^{51.26\%}$

HTR1B
Docking Scores:
6.24, 6.65, 7.44
Similarity Score:
0.51
Best Match:
CHEMBL3104092
Prediction Result:
Inactive$^{66.38\%}$

HTR1D
Docking Scores:
7.37, 7.81, 7.80
Similarity Score:
0.51
Best Match:
CHEMBL3104092
Prediction Result:
Inactive$^{50.39\%}$

HTR1E
Docking Scores:
7.18, 5.41, 7.47
Similarity Score:
0.35
Best Match:
CHEMBL22744
Prediction Result:
Inactive$^{66.04\%}$

HTR1F
Docking Scores:
8.10, 6.47, 7.69
Similarity Score:
0.42
Best Match:
CHEMBL187928
Prediction Result:
Inactive$^{75.67\%}$

HTR2A
Docking Scores:
7.80, 7.00, 6.45
Similarity Score:
0.56
Best Match:
CHEMBL592752
Prediction Result:
Active$^{55.49\%}$

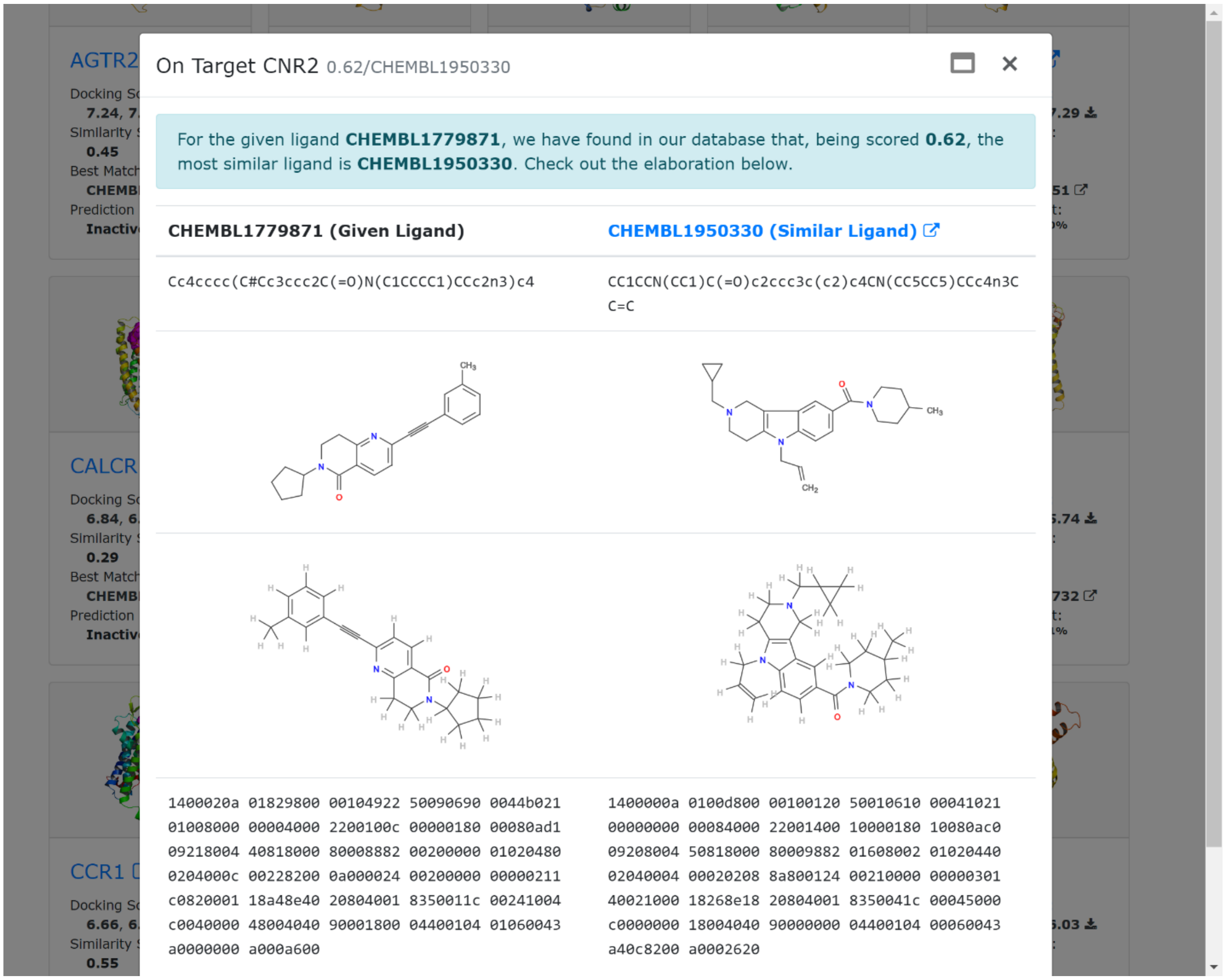
HTR2B
Docking Scores:
8.24, 6.57, 7.04
Similarity Score:
0.58
Best Match:
CHEMBL1110
Prediction Result:
Inactive$^{51.01\%}$

HTR2C
Docking Scores:
7.55, 6.88, 7.87
Similarity Score:
0.56
Best Match:
CHEMBL592752
Prediction Result:
Inactive$^{52.84\%}$

HTR4
Docking Scores:
7.39, 8.42, 7.30
Similarity Score:
0.48
Best Match:
CHEMBL2401750
Prediction Result:
Inactive$^{97.35\%}$

HTR5A
Docking Scores:
6.72, 6.41, 7.00
Similarity Score:
0.57
Best Match:
CHEMBL129476
Prediction Result:
Inactive$^{55.25\%}$

GPCR Prediction Online © 2018 - CCGS and NIDA CDAR Centers
335 Sutherland Drive, 206 Salk Pavilion
University of Pittsburgh
Pittsburgh, PA 15261, USA

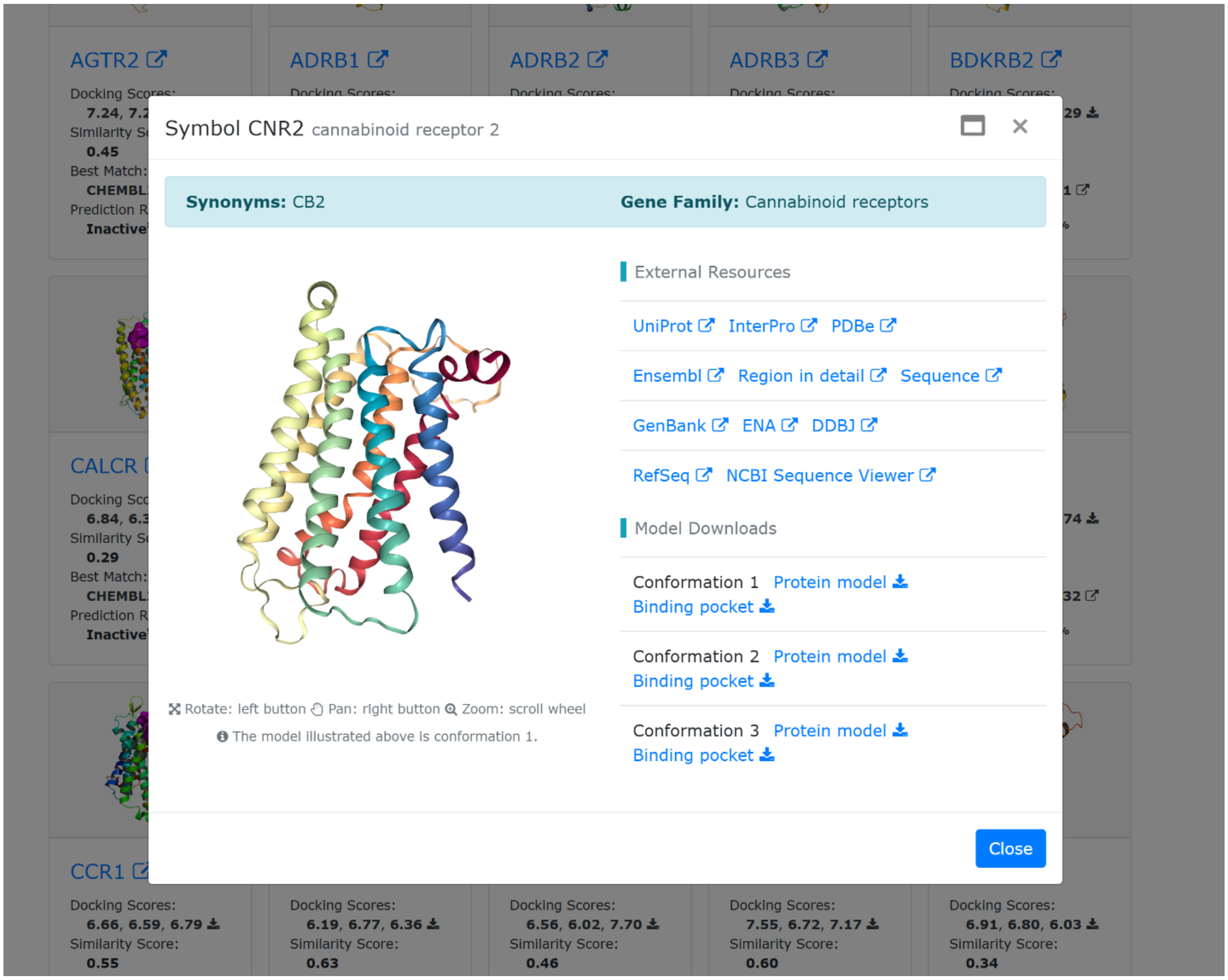


On Target CNR2 0.62/CHEMBL1950330

For the given ligand **CHEMBL1779871**, we have found in our database that, being scored **0.62**, the most similar ligand is **CHEMBL1950330**. Check out the elaboration below.

CHEMBL1779871 (Given Ligand)	CHEMBL1950330 (Similar Ligand)
Cc4cccc(C#Cc3ccc2C(=O)N(C1CCCC1)CCc2n3)c4	CC1CCN(CC1)C(=O)c2ccc3c(c2)c4CN(CC5CC5)CCc4n3CC=C
1400020a 01829800 00104922 50090690 0044b021 01008000 00004000 2200100c 00000180 00080ad1 09218004 40818000 80008882 00200000 01020480 0204000c 00228200 0a000024 00200000 00000211 c0820001 18a48e40 20804001 8350011c 00241004 c0040000 48004040 90001800 04400104 01060043 a0000000 a000a600	1400000a 0100d800 00100120 50010610 00041021 00000000 00084000 22001400 10000180 10080ac0 09208004 50818000 80009882 01608002 01020440 02040004 00020208 8a800124 00210000 00000301 40021000 18268e18 20804001 8350041c 00045000 c0000000 18004040 90000000 04400104 00060043 a40c8200 a0002620

Figure 2.1.4G Popup window showing the comparison between query compound and the most similar compound in the database



Symbol CNR2 cannabinoid receptor 2

Synonyms: CB2 **Gene Family:** Cannabinoid receptors

External Resources

UniProt InterPro PDBe

Ensembl Region in detail Sequence

GenBank ENA DDBJ

RefSeq NCBI Sequence Viewer

Model Downloads

Conformation 1 Protein model Binding pocket

Conformation 2 Protein model Binding pocket

Conformation 3 Protein model Binding pocket

Rotate: left button Pan: right button Zoom: scroll wheel

The model illustrated above is conformation 1.

Close

Docking Scores: 6.66, 6.59, 6.79 Similarity Score: 0.55

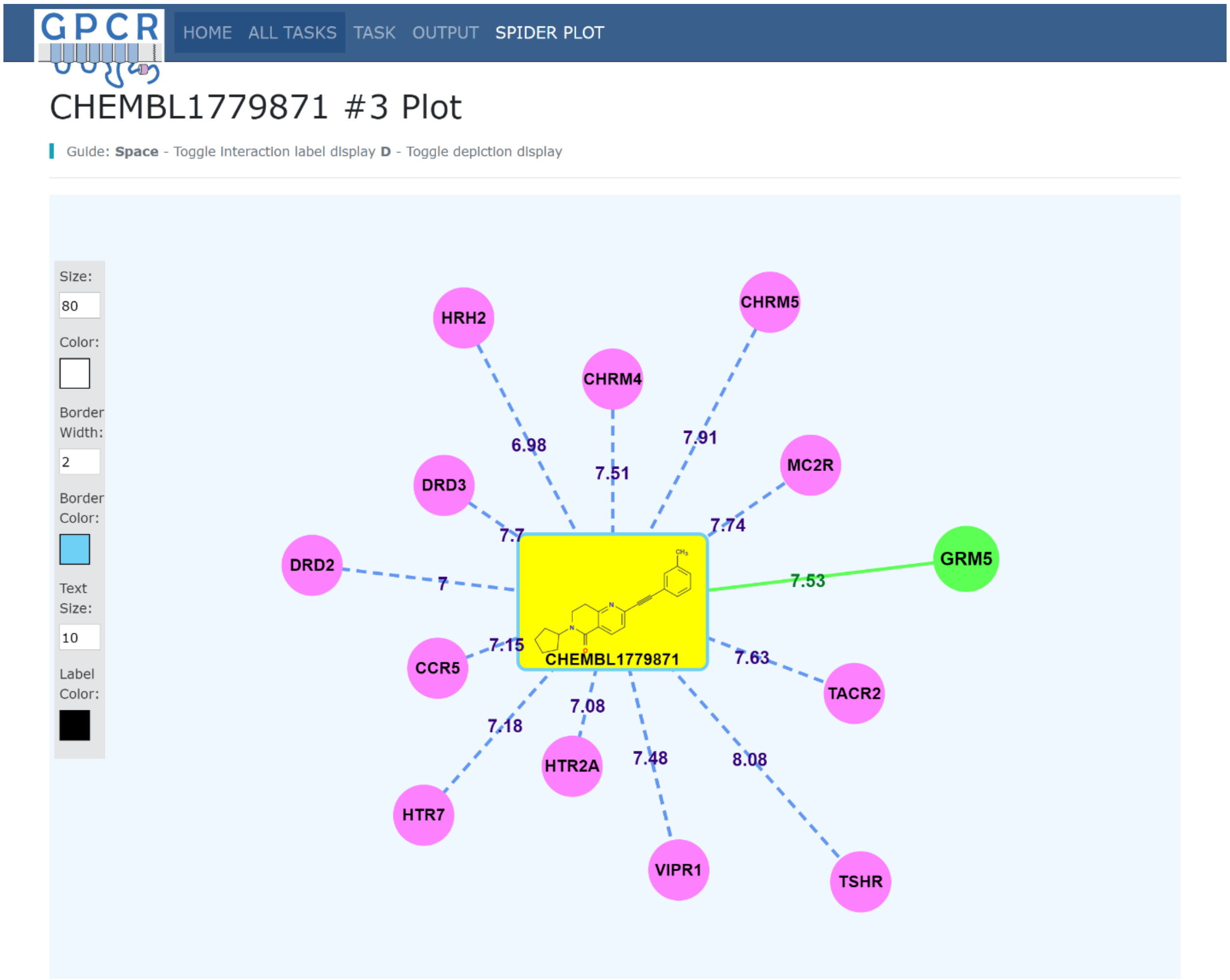
Docking Scores: 6.19, 6.77, 6.36 Similarity Score: 0.63

Docking Scores: 6.56, 6.02, 7.70 Similarity Score: 0.46

Docking Scores: 7.55, 6.72, 7.17 Similarity Score: 0.60

Docking Scores: 6.91, 6.80, 6.03 Similarity Score: 0.34

Figure 2.1.4H Popup window presents additional resources for a protein target



CHEMBL1779871 #3 Plot

Guide: **Space** - Toggle interaction label display **D** - Toggle depiction display

Size: 80

Color: □

Border Width: 2

Border Color: 🟦

Text Size: 10

Label Color: ■

Figure 2.1.4I Spider Plot for data virtualization and analysis

2.1.4 Conclusion

In this study, we have introduced a DA-domain-specific chemogenomics knowledgebase platform for GPCR-related biomedical and pharmaceutical science research. This platform includes both the biomedical records database and implemented cheminformatics and bioinformatics tools. With the illustrated systems pharmacology analysis on GPCRs targets and therapeutic agents, we demonstrate that DAKB-GPCR can be an efficient and powerful tool to analyze data in DA-related research. The analysis result from this platform will help to 1) identify the patterns of DA from the systems pharmacology perspective by exploring the interactions between small molecules and GPCR proteins; and 2) provide a better understanding of how genes/proteins and small molecules influence the various risks for DA. Considering the current knowledge of GPCR distribution in the brain, the platform also provides BBB penetration evaluation to help with understanding the brain circuitry that underlies DA.

The systematic effects and pharmacological mechanisms between abused drugs and DA-related GPCRs are complex. The state-of-the-art computational technologies, together with the Pharmaco-Analytics methods, enable us to understand them from a systems pharmacology aspect. Our TargetHunter, HTDocking, and AI/ML-based analysis boost the possibility of finding and optimizing lead compounds that combine multiple desirable mechanisms of action on these GPCRs in a single new chemical entity for DA intervention. In addition, our GPCR-DAKB platform focus on analyzing interactions between DA-related GPCRs targets and their ligands, which be generalized to other DA-related target protein families. Therefore, we believe that our GPCR-DAKB platform will facilitate biologists to quickly discover the mechanisms of action for active ligands for DA treatment. Moreover, our platform will also facilitate pharmaceutical science researchers on different tasks such as off-target adverse effect analysis as well as drug repurposing. The algorithms and methods developed and integrated into GPCR-DAKB can help predicting new targets of existing drugs/compounds, which will further help discovery of novel therapies.

To our knowledge, no such domain-specific system is available for the proposed computational applications. Our platform is the most comprehensive web-based service that integrates DA-related genes, proteins, and drugs for DA research. In the development of this platform, state-of-the-art computational chemistry/cheminformatics and machine learning algorithms established in our lab have been implemented. And by building cloud sourcing and cloud computing web service that can be accessed worldwide, we believe our DAKB-GPCR platform will boost the information exchange and data-sharing of knowledge new among DA researchers and healthcare providers, as well as relevant scientific communities.

2.2 DeepTargetHunter: A Novel Method for Target Identification using Deep Learning

2.2.1 Significance and Background

Modern drug discovery is based on the concept that the interaction between drugs and their protein targets can modulate the biological function of human protein, which drives the pharmaceutical action for disease treatment (Shuker et al. 1996). Exploring drug-target interactions (DTI) is the initial and vitally important step in drug discovery and development, which can be used not only for improving efficacy but also for drug-repurposing and off-target safety profiling (Hopkins 2008; Iorio et al. 2010; Iskar et al. 2013). At the same time, conventional drug discovery based on the 'one-drug-one-target-one-disease' assumption has been concluded to be the less successful tactic for complex diseases (Horrobin 2001). In recent decades, the concept of systems pharmacology has emerged as the new discipline to tackle such challenges in drug discovery (Zhou et al. 2016; Vicini and van der Graaf 2013). Single drugs can interact with multiple targets, and the treatment for one disease may require synergistic regulation between multiple targets. Therefore, there is a strong incentive to promote DTIs study to find a new therapeutic strategy. However, the identification of DTIs using large-scale chemical screening or biological experiments is usually time-consuming with high associated costs (Iorio et al. 2010; Iskar et al. 2013). To reduce the cost and save time to meet the needs of the pharmaceutical industry and pharmaceutical science academy, computational methods have been introduced into DTI prediction (Xu, Ru, and Song 2021).

Traditional computational methods for large-scale DTIs identification have limitations. Similarity-based methods (Wang et al. 2013) were developed based on the simple medicinal chemistry assumption that structurally similar compounds usually share similar physiochemical properties and similar biological targets. These methods are fast and straightforward; however, they only consider ligand information and rarely exploit the features from the protein data. High-throughput molecular docking (Liu et al. 2014) methods are relatively accurate and robust; however, they require the precise 3D structure of the protein target and are usually computationally expensive and time-consuming. Nowadays, an increasing number of researchers are changing their strategies from chemistry-centric modeling to combined methods, which not only consider small molecules features but also include target protein information (You, McLeod, and Hu 2019; Wang, You, et al. 2020; Huang et al. 2020; van Laarhoven, Nabuurs, and Marchiori 2011).

Enabled by the explosive increase in biomedical data in conjunction with machine learning, these techniques have become popular methods in drug-target interaction prediction research. Quickly and with no cost, machine learning methods offer an avenue for identifying the potential biological targets for small molecules accurately. Among machine learning applications, recommendation systems used for providing recommendations have become part of our everyday lives. For example, e-commerce sites offer suggestions based on our personal information and purchasing history, while online streaming sites recommend music and videos that we may enjoy. This concept of “user-item relationship” prediction can be extended to suggesting routes for exploring the drug-target relationship. Following this design, a machine learning-based recommendation framework for DTI prediction can be developed if considering the biological targets as users and the small molecular ligand as items. By projecting the ligand and target features into a high-dimensional space, the mathematical mapping between them can be learned as the abstraction of interaction.

To achieve this, state-of-the-art deep learning (DL) algorithms will be applied. DL is a new class of machine learning algorithms specifically designed for Artificial Intelligence (AI) (LeCun, Bengio, and Hinton 2015). The idea of DL came from the exploration of Artificial Neural Networks (ANN), which were initially developed to mimic the activity of neurons in the human brain (Gardner and Dorling 1998). The advantage of DL is that it consists of multiple layers, which can intensify the capacity of signaling transformation and feature extraction.(Schmidhuber 2015). With such an architecture of multiple layers, deep learning will have the capability of extracting contributed features for identifying DTIs from the chemical/biological structure of small molecules and protein targets. Using those extracted features, classification to determine whether there is an interaction between small molecules and protein targets will be addressed. Due to the ability of DL to use more complex nonlinear transformations, it is believed that the prediction of unknown targets of compounds will be more efficient than current methods. This work proposes a novel DL-based method for large-scale DTIs prediction to support drug discovery research.

2.2.2 Methods

2.2.2.1 Overall workflow

Our new DL-based design adopted many state-of-the-art technologies such as embedding techniques in natural language process (NLP), deep convolutional neural network (CNN), deep recurrent neural network (RNN), personalized recommendation ranking system, and so on. The overall framework is depicted in Figure 2.2.1, consisting of one non-supervised learning component for feature embedding and one supervised learning component for classification. In the non-supervised components, both small molecular information and protein target information are learned through embedding methods. Then the embedded representations of ligand and target are further abstracted through a deep attention architecture for abstracting high-level features. At last, the abstracted information is fitted into an MLP classifier for final prediction.

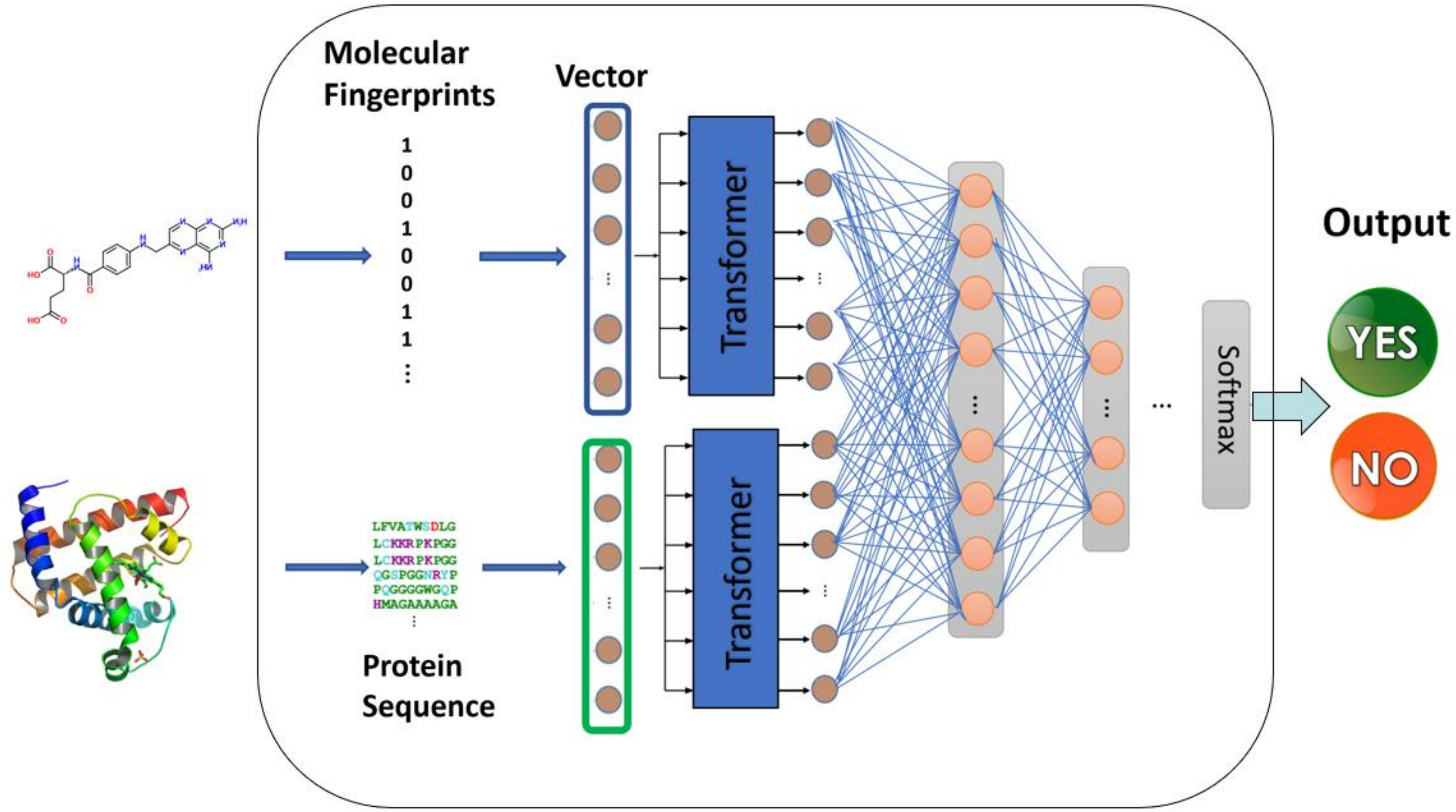


Figure 2.2.1 Overall design of the DeepTargetHunter. The input of the DeepTargetHunter involves both the chemical structure of the small molecule (2D) and the amino acid sequence of the protein target (1D). The mathematical representations of the compound and the target protein were learned from their structures using embedding models. Then the two vectors will be used as the input of the deep learning-based classification model to predict whether those two will interact with each other.

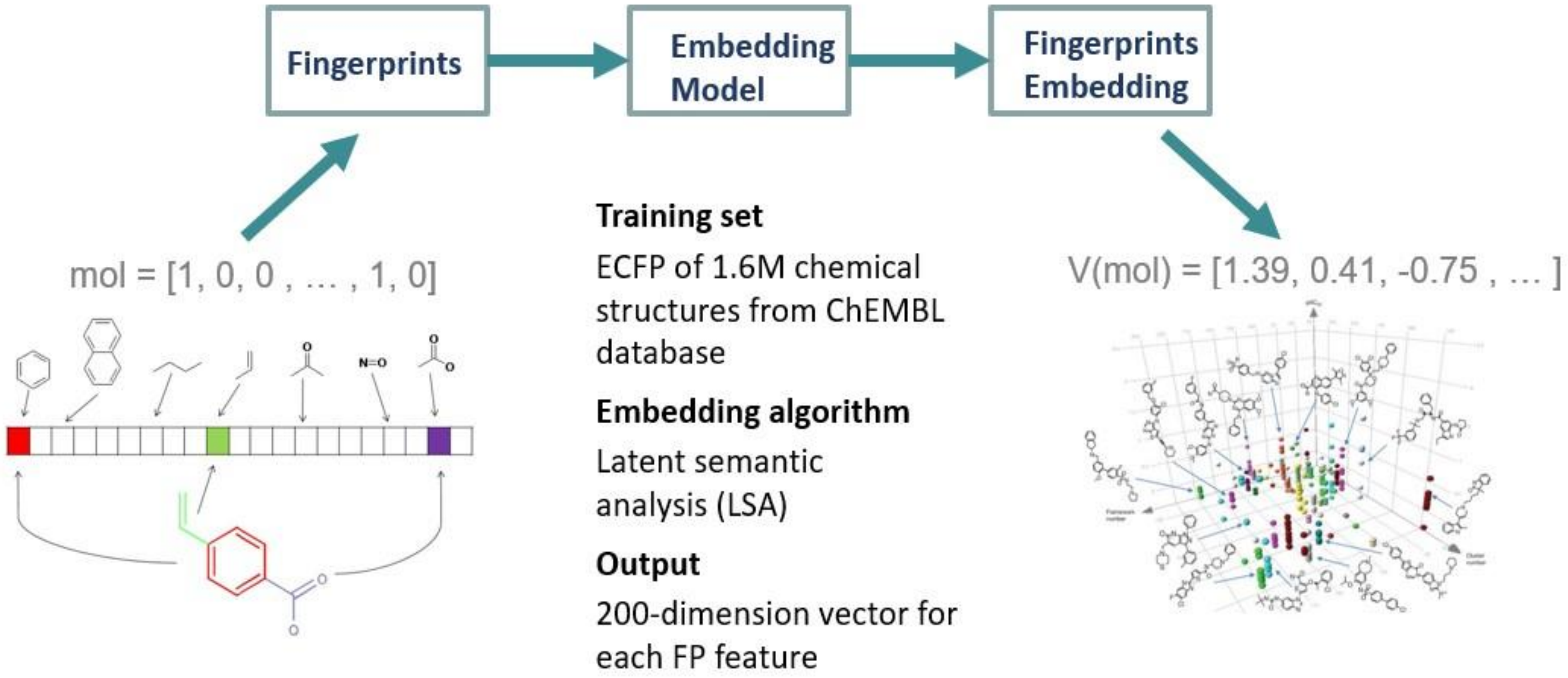
2.2.2.2 Data preparation

The drugs and targets data for training DTIs models were collected from the DrugBank database (Wishart et al. 2018), which is a unique bioinformatics and cheminformatics resource that combines detailed drug data with comprehensive drug-target information (http://www.drugbank.ca/). The entire dataset contains 8563 small molecules and 4345 targets, with 18586 known interactions. Among those interactions, some pairs are positive DTIs, and some pairs are not specified. The negative samples used in this study were randomly selected from the unknown DTIs. In the present study, we randomly selected 18586 samples from the non-specified DTIs as a negative data set. Thus, the whole data set has 38336 samples. A random split to the training set and test set was performed, with the training set assigned 30000 samples and the test set assigned 8336 samples.

For the feature embedding models training, additional data sets were collected. For small molecules, a total of 123182 drug-like compounds were collected from the ZINC database (Sterling and Irwin 2015) and then diversely selected based on the molecular fingerprint similarity. In total 6125 proteins targets with the primary structure were collected from RSCB PDB (Berman et al. 2000). In addition, an extra test set was constructed using the DUD data set (Mysinger et al. 2012), which contained 40 biological targets and 98,266 ligands (2950 active ligands and 95316 inactive ligands).

2.2.2.3 Small molecular feature calculation

The conventional method to describe the chemical feature of a small molecule is to use molecular fingerprints, which is a substructure search using hand-engineered structural keys. However, in this type of method, each feature (binary bit) is equidistant from every other and the correlation information between features is missing. Using the embedding method to learn a distributed representation of molecular fingerprints can address this problem. The latent semantic analysis (LSA) technique was adopted here to learn the distributed representation. LSA is commonly used in Natural language processing (NLP) for document similarity analysis (Nadkarni, Ohno-Machado, and Chapman 2011; Landauer, Foltz, and Laham 1998). In LSA, each document is represented by a vector storing the term frequency and inverse document frequency (TF-IDF) information, which is a numerical statistic to describe the importance of a word in a document. And the entire collection of documents can be represented by a matrix in which the matrix column contains the occurrence information of terms in the document. At last, a low-dimensional distributed representation of terms (features) can be retrieved using singular value decomposition (SVD) (Furnas et al. 2017). In this study, ECFP6 was calculated as the initial molecular fingerprints representation to fit the embedding model (Figure 2.2.2). Considering the ECFP6 molecular fingerprint as a document, each binary bit in the ECFP6 can be considered as a term (or word), and finally the distributed representation of each fingerprint bit can be learned through this method.



Training set

ECFP of 1.6M chemical structures from ChEMBL database

Embedding algorithm

Latent semantic analysis (LSA)

Output

200-dimension vector for each FP feature

Figure 2.2.2 Small molecular feature calculation using embedding technology. To calculate the mathematical representation of the small molecules, the molecular fingerprints (binary bits) will be calculated first, and then the binary bits will be translated into a multi-dimensional vector using an embedding model trained with a larger small-molecule training set

2.2.2.4 Protein feature calculation

There have been many successful applications in order to learn the accurate structure information of a protein from its amino acid sequence information, such as Alpha-Fold (Jumper et al. 2021). However, it may not be necessary to build a highly complex model to reproduce the protein structure. The primary purpose of this step is to learn the informative representation of a protein from its primary structure sequence. This representation may contain some structural features that can help to understand the mapping between ligand and protein. To convert protein sequences into sequential representation, we first split a protein sequence into a triple-gram amino acid sequence and then translated them to mathematical embeddings.

In this step, the word2vec technique was applied to learn the vector representation of a protein target (Goldberg and Levy 2014). It is an unsupervised technique to learn high-quality distributed vector representations that describe sophisticated syntactic and semantic word relationships. Specifically, the Continue Bag-of-Words (CBOW) method was adopted to train the embedding model (Kenter, Borisov, and De Rijke 2016). In this model, the amino acid sequence of each protein was considered as a ‘sentence,’ and any combination of three continuous amino acid was considered as a ‘word.’ The goal of the model is to predict the center ‘word’ given the surrounding context. As the result, this model finally maps the ‘word’ to low-dimensional real-valued vectors, where the words that have similar. For collecting the surrounding information, a context window was set to 10, and the embedding dimension was set to 128.

2.2.2.5 The deep learning model for identifying drug-target-interaction training

In this step, a DL-based classification model was developed using the learned features of both ligands and protein targets from the embedding models as input for assigning them the DTIs associated information. As shown in Figure 2.2.1, the transformer encoder architecture was adopted for both small molecule input and protein target input to encode the high-level features. The transformer is a new strategy for DL-based sequence transduction that replaced the old RNN, and recently it has become very successful in both NLP and computer vision fields. The general idea of the transformer is to use the ‘attention’ mechanism, commonly the self-attention model, to keep ‘long term memory’ when dealing with sequence (Vaswani et al. 2017)s. A typical transformer encoder is a stack of multiple identical layers. Each layer has two sub-layer:, one multi-head self-attention pooling layer and one position-wise feed-forward network. Specifically, in the encoder self-attention, queries, keys, and values are all from the outputs of the previous encoder layer. As a result, the transformer encoder outputs a multi-dimensional vector representation for each position of the input sequence.

After the transduction of features, the output from the transformers was fitted into an MLP classifier to predict whether the input compound and protein target interact with each other or not. For hyper-parameter tuning, different numbers of layers and hidden neurons in MLP, as well as the optimization algorithms, were evaluated. In the batch training process, 1/10 of the batch samples was used as the validation set for monitoring model performance.

For evaluating model performance, the ROC analysis was applied, and the area under the curve (AUC) was calculated as the representation of model accuracy of binning the sample into the correct category. The model construction and evaluation in this step were performed using *Tensorflow 2.1*.

In addition, comparative models were generated using traditional machine learning algorithms. The descriptors adopted for training these models were molecular fingerprints and molecular properties. All models were developed using the scikit-learn package in Python.

2.2.2.6 Hyperparameter optimization and model evaluation

The optimization of model architectures setup (hyperparameters) was processed based on the Bayesian optimization algorithm. Specifically, for deep-learning-based models including the deep graph neural network models, the optimization was completed using the *HParams* module in TensorFlow. The following hyperparameters were considered for optimization: dropout rate, type of optimizer, learning rate, batch size, number of fully connected hidden layers, and number of graph neural network layers. In the hyperparameter tuning, we selected the most optimized hyperparameter setting for each model showing the highest accuracy on the 10-fold-cross-validation. In addition, the hyperparameter tuning for traditional machine learning models adopted the *skopt* python package. For the model evaluation in the training and validation process, ROC analysis was applied to evaluate model performance, with the ROC curve plotted by a false-negative rate (FNR, 1-specificity) against a true positive rate (TPR, sensitivity, recall) at all classification thresholds. The area under the curve (AUC) represents the degree or measure of separability. This ROC AUC (also called AUROC), ranging from 0.5 to 1, specifies the accuracy of the classification model predicting the samples into its correct category.

2.2.3 Results and Discussion

2.2.3.1 Model optimization

Five different optimization methods, including Stochastic Gradient Descent (SGD), Adaptive Gradient (Adagrad), Adaptive Delta (Adadelta), Adaptive moment (Adam), and Root Mean Square Propagation (Rmsprop), were chosen to find the most efficient model in this case. Different dropout ratios (0 or 0.5) combined with different batch sizes (10, 100, 1000) were also added to the comparison. The results (**Figure 2.2.3**) showed that the model trained using Adadelta methods performed relatively better in this case, especially using a small batch size 10 without adopting the dropout technique for further regularization, achieve the best performance with the validation accuracy reaching 0.89 and test accuracy reaching 0.82. Although the model trained using the Adagrad method performed better, it seems that a significant over-fitting problem may occur in this case. In addition, different numbers of layers (1, 3, 5, 7, 9) were also tested and compared. The results showed that 7-layers in descending number of nodes achieved the optimized performance.

In addition, overfitting is a common problem discussed in ML-based modeling. To minimize the effect of overfitting, we also compared the training performance and the test performance when selecting the best hyper-parameter settings. For example, in **Figure 2.2.3**, we compared the changes of model performance (ROC AUC) in both training (upper row) and validation (bottom row) procedures. If we see convergence in the validation performance, but the training performance is still improving, then the model is starting to be over-fitted. Therefore, between Adadelta and Adagrade optimization methods, we finally choose the Adadelta method to minimize the overfitting problem.

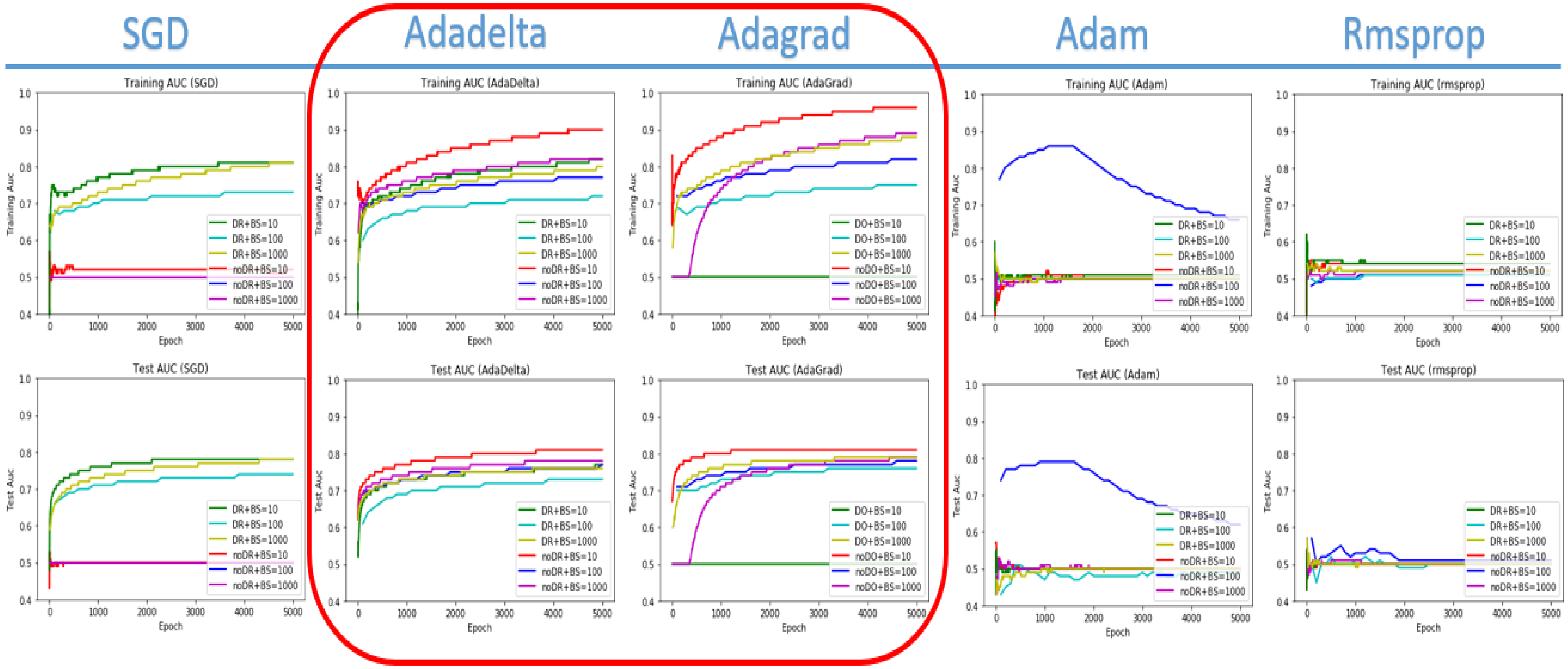


Figure 2.2.3 Hyperparameter tuning of DL-based DTIs models on optimizer, dropout ratio, and batch size. The entire figure shows the comparison of the model validation performance between different hyperparameter settings. Each small-block shows the predictive accuracy changes with the training of the model. The X-axis is epoch, and the Y-axis is the predictive accuracy in the validation

2.2.3.2 Model evaluations

Traditional ML methods were adopted to build comparative models. In this step, four types of ML algorithms were applied, including logistic regression (LR), support vector machine (SVM), random forest (RF), and Naïve Bayes. **Table 2.1** shows the comparison between the DL-based DTI prediction method and other ML methods. The DL-based model outperformed other machine learning methods in the evaluation on both validation set and test set. In contrast, the RF model achieved comparative performance in the test set evaluation. Additional evaluation on our new DL-based methods was performed using the extra test set. The novel DL-based model achieved an accuracy of 0.61, with sensitivity achieving 0.88 and specificity achieving 0.61. This performance is much worse compared to the primary evaluation. One reason for this may be that the extra test set has an unbalanced but relatively actual distribution of active interaction. Our model mostly predicts the non-active interaction into the active interaction class. However, with a high recall, this performance is acceptable as the model will be used in the first round to identify as much active interaction as possible to reduce the risk of off-target side effects or increase the chance of identifying new drugs.

Table 2.1 Comparison between deep learning and other machine learning algorithms

Algorithms	**Training Accuracy**	**Training AUC**	**Test Accuracy**	**Test AUC**
DL	0.84	0.89	0.85	0.82
LR	0.65	0.67	0.64	0.64
SVM	0.78	0.78	0.74	0.73
RF	0.83	0.85	0.83	0.82
Naïve Bayes	0.55	0.55	0.48	0.53

2.2.4 Conclusion

In this research, we established a novel DeepTargetHunter platform for target identification tasks using the state-of-art deep learning architecture with a self-attention mechanism for sequence feature extraction. Our DeepTargetHunter shows reliable performance on benchmark test sets. Moreover, we compared it with traditional machine learning-based models. The results show that our method achieved better-improved performance on the prediction tasks, suggesting it can learn desired interaction features and decrease the risk of hidden ligand bias. Finally, model interpretation capability was studied by mapping attention weights to protein sequences and compound atoms, which can explore whether a prediction is reliable and has physical significance.

It is also worth noting that we adopted 1D sequential data to represent the protein data in this work. It is because, firstly, the number of experimental 3D structured data is smaller than that of 1D sequential data, However, deep learning requires a relatively large number of training data samples. Although Alpha-Fold has successfully predicted the 3D structure of protein targets using amino acid sequence, it is still not promising to apply those predicted 3D data directly. In addition, the lack of pocket information and the time-consuming structure-based screening limit its application in target identification and drug discovery. However, the success of Alpha-Fold provides a novel perspective for processing the 1D protein sequence data and can be optimized in the future study of protein feature extraction. Overall, our DeepTargetHunter delivers a much faster and relatively accurate prediction of the potential targets of a small molecule and can be used widely for off-target identification and drug repurposing.

CHAPTER 3. AI/ML PHARMACO-ANALYTICS FOR PRECLINICAL MODELING

3.1 DeepGhERG: A Pharmaco-Analytics Prediction of Cardiotoxicity using Graph-based Deep Learning and Artificial Intelligence

3.1.1 Research Background of Cardiotoxicity

3.1.1.1 Proarrhythmic cardiotoxicity and hERG

Proarrhythmic cardiotoxicity is one of the most severe side effects in drug research and development (R&D) caused by the off-target interactions of drugs with cardiac ion channels that control the normal heart rhythm (Gintant, Sager, and Stockbridge 2016; Jing et al. 2015). The human Ether-a-go-go Related-Gene (hERG) encodes a voltage-gated potassium channel, which takes charge of the action potential repolarization of cardiomyocytes. This channel carries delayed rectifying potassium current (IKr), which underlies the cardiac action potential repolarization. It is the principal ion channel when evaluating the cardiotoxicity of a drug candidate. Pharmacological blockade of the hERG channel by non-cardiac drugs inhibits the delayed IKr, and then delays the cardiac action potential repolarization (Cavalli et al. 2002). This results in the Long QT syndrome, shown as the extended QT interval in the electrocardiograph (ECG). Such QT interval can significantly increase the risk of the proarrhythmic cardiotoxicity termed Torsade de Points (TdP) and lead to life-threatening adverse side effects such as sudden cardiac death. Several non-antiarrhythmic drugs have been withdrawn from the market because of inducing TdP, such as Cisapride, Terfenadine, and Terodiline (Aronov 2005; Stockbridge et al. 2013). Therefore, it is necessary to develop efficient methods to evaluate proarrhythmic liabilities of drug candidates at the early stage of the drug discovery process to avoid investing in risky lead series.

3.1.1.2 Experimental methods for evaluating drug-induced cardiotoxicity

The standard cardiac safety paradigm was documented by the International Committee on Harmonization (ICH) S7B preclinical guidelines and E14 clinical guidelines (Pugsley and Curtis 2006). Instead of assessing the risk of TdP directly, those guidelines mainly focus on the detection of surrogate biomarkers, namely preclinical evaluation focuses on testing the blockade of the repolarizing I_{Kr} current that flows through the hERG ion channel. In contrast, the clinical study focuses on QT/QTc interval prolongation on electrocardiography (ECG) at the therapeutic and supratherapeutic exposure levels of the pharmaceuticals. In the preclinical phase of the drug discovery process, in vitro and in vivo experiments were established following the S7B guidelines (Group 2005). In vitro methods assess the potency of compounds to block the hERG channel. The standard in vitro method utilizes whole-cell patch clamping from recombinant cell lines that stably express the hERG channel (Houtmann et al. 2017). These electrophysiology assays can present a better estimation of hERG potency when compared with non-electrophysiological methods. Other non-electrophysiology assays, including rubidium efflux assay (Chaudhary et al. 2006), radioligand binding assay (Yu et al. 2014), and fluorescence-based assay (Piper et al. 2008), were developed for cases in which higher throughput and lower cost are desirable, even though those assays may result in under or overestimation of hERG potency. In vivo tests propose to detect the changes in the ECG in animal models and use them to predict the QT prolongation in humans; however, the significant species differences in ventricular repolarization and response to drugs limit the use of animal models in preclinical research (Raghib, Stebbing, and Majewski 2006).

3.1.1.3 Computational methods for evaluating drug-induced cardiotoxicity

Computational methods have been widely used in the recent two decades to predict either hERG inhibition or the binding affinity of compounds. Compared to in vitro and in vivo assays, in silico models require less cost and time and are very common in the early phases of drug discovery. Generally, in silico methods for small molecular drug discovery can be mainly divided into two main classes: structure-based methods and ligand-based methods. Structure-based methods such as docking and molecular dynamics are based on the availability of the 3D atomic structures of the targets, in which a simulation of molecular binding will run between the 3D structures of small molecule and target protein to mimic the actual molecular interaction (Jing et al. 2015; Kalyaanamoorthy and Barakat 2018). Until the recent determination of the cryo-EM structure of the hERG channel (Wang and MacKinnon 2017), several well-established homology models were constructed based on the solved crystal structures of other similar homologous proteins (Xiao et al. 2017). However, due to the uncertainty of the binding mode between compounds and the hERG channel, it is still hard to accurately predict the interaction between chemical compounds and the hERG channel. Thanks to the exploration of computational algorithms in modern technology of data mining and artificial intelligence (AI), as well as the development of advanced experimental technologies that efficiently generate large-scale biochemical data, ligand-based approaches are continually explored to predict the inhibition between the ligand and hERG channel (Jing et al. 2018a). Generally, ligand-based methods are developed based on the medicinal chemistry assumption that structurally similar compounds should share similar physiochemical properties and biological targets. In the recent two decades, ligand-based models have been commonly adopted to explore the structure-affinity relationship of hERG blockers using various algorithms. Those methods include pharmacophore methods, CoMFA-based methods, three-dimensional (3D) quantitative-structure activity relationship (QSAR) methods, and machine learning-based models such as support vector machine (SVM), random forest (RF), Naïve Bayes (NB), and so on (Jing et al. 2015; Ermondi, Visentin, and Caron 2009; Cavalli et al. 2002; Lu et al. 2018; Konda, Keerthi Praba, and Kristam 2019). Those methods mainly aim to retrieve and summarize the common patterns that contribute to the interaction between compound and protein. In most of those methods, molecular properties (such as molecular weight and log P) and/or molecular fingerprints were used as the feature to represent either the chemical or the physical-chemical characteristics of the small molecules(Liu et al. 2020). And a manual or automatic feature selection process will be conducted to select the relevant feature subset for model training (Cano et al. 2017).

3.1.1.4 Deep learning for drug-induced cardiotoxicity

In recent years, with the development of state-of-the-art deep learning algorithms for AI technology, deep neural networks (DNN) architectures were utilized to develop classification models to discriminate hERG blockers, showing their capability to process large-scale datasets and usually provide better predictive performance (Cai et al. 2019; Sharifi et al. 2017; Zhang et al. 2019; Wang, Huang, et al. 2020). However, optimized deep learning architectures such as CNN and RNN are specifically designed for Euclidean data with clear spatial order (e.g., image data and/or text data); therefore, it is inappropriate to adopt them on non-Euclidean data such as molecular data.

More recently, graph neural networks (GNNs) became more and more popular due to their capability in dealing with non-Euclidean data by processing those data in a graph-like format (Wu et al. 2020). The motivation of GNNs came from CNNs and graph embedding theory (Wu et al. 2020; Zhou, Cui, et al. 2020). In GNNs, the graph nodes representations are learned through graph embedding. Then the graph structure information can be aggregated collectively, learning from the concept of local connection, shared weights in CNN. Therefore, GNNs can model input and output consisting of elements and their dependency.

Furthermore, incorporating the gate theory in RNNs, GNNs can simultaneously model the sequential diffusion process on the graph. Using different embedding and diffusion methods, different types of GNNs were developed, such as graph convolutional networks, graph attention networks, graph auto-encoders, graph generative networks, and others. Several published studies have tried to use GNNs in drug discovery (Sakai et al. 2021; Liu et al. 2019; Ryu et al. 2020), and those works proved that GNN-based methods have advantages over those traditional methods in drug discovery. However, those methods mainly focused on constructing the model architecture using GNNs but did less research on learning representation of the atom node in the molecular graph. This continuous vector representations of the atom nodes were learnt to manually define the atom characteristics and represent the atom node, for example, atom type, atom charge, atom aromaticity, atom chirality, etc. Since there is no strict standard to select the atom characteristics, this step can be subjective and arbitrary. The alternative option is to learn atom representations directly from raw data chemical structure. Word embedding methods in natural language programming have proved efficient in learning dense vector representations (Goldberg and Levy 2014). It is valuable to explore the feasibility and efficiency of learning atom representation automatically using similar embedding methods.

In the present study, we develop a novel in-silico method to predict the hERG-related toxicity of small molecules using GNNs. For discriminating hERG inhibitor and non-inhibitor, 1) 2D chemical structure is used as the input; 2) the molecular representations are generated by specified atom-type-based embedding models; 3) the potential hERG inhibition of the small molecular are predicted. Our results show that by integrating the atom feature learning, our GNN-based model can better predict the hERG inhibition of small molecules.

3.1.2 Methods

3.1.2.1 Data preparation

The small molecules with experimental bioactivities against the hERG channels were originally collected from the ChEMBL database (Gaulton et al. 2018). The chemical structure of the compounds was extracted from the SMILES string using the Open Babel toolkit (O'Boyle et al. 2011), with solvents and salts removed using the RDKit toolkit (Landrum 2013). For the general consistency of the experimental data, only hERG IC50 data were assembled, including those from patch-clamp assay and binding assay. Data were then primarily cleaned by using a few criteria: 1) compounds without well-defined experimental bioactivities were eliminated; 2) incompatible data with the original published source were eliminated; 3) duplicated records identified by SMILES were removed; 4) for compounds with the same structure but inconsistent inhibitory activity values, the average binding affinity were kept. Finally, there were a total of 7909 small molecules in the entire dataset. The entire dataset was randomly divided into two subsets, a training validation set (90%) and a test set (10%). For evaluating the chemical diversity of the data set, the chemical similarity between compounds in the datasets was calculated using the Tanimoto coefficient and molecular fingerprint. In addition to examining whether the distribution of the training set and test set is similar or not, compounds similarity was also calculated between the training set and test set. Compounds were binned to hERG blockers and non-blockers to develop classification models based on their hERG bioactivity data. Compounds with IC50 values less than 10 micromolar (µM) or other equal measurements were considered blockers, while the rest were non-blockers. The distribution of hERG blockers and non-blockers is depicted in Figure 3.1.1. For comparative analysis, an external test set of 44 small molecules from the published paper were used as an external test set (Ryu et al. 2020).

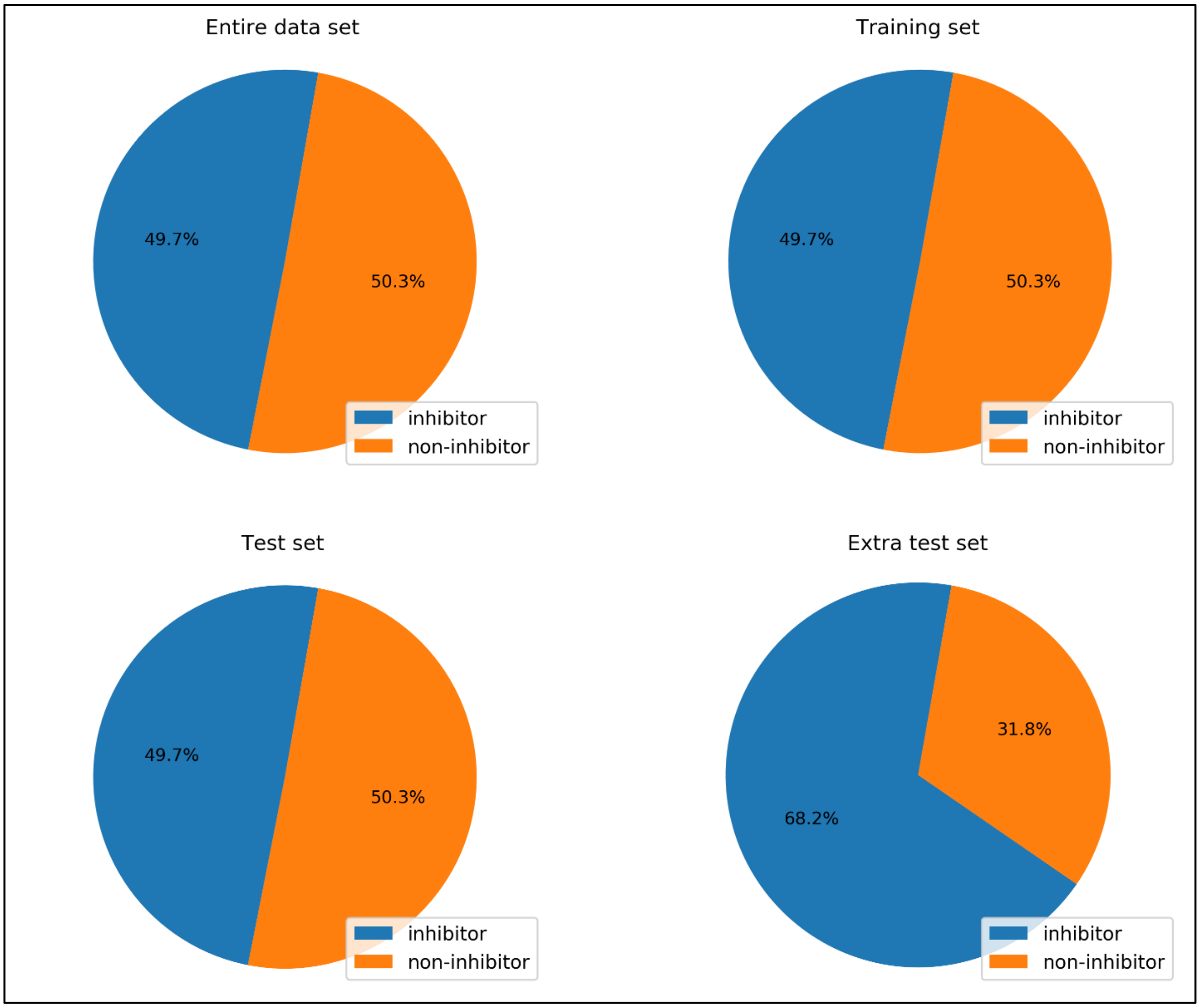


Figure 3.1.1. Distribution of the collected hERG datasets. This figure shows the distribution between hERG inhibitor (active) and non-inhibitor (inactive) in different data sets including the entire dataset, training set, test set, and extra test set

3.1.2.2 Calculation of different molecular feature datasets

In this study, three types of the chemical structure representations were generated as the feature to build models: 1) a graph-based molecular representation, 2) molecular fingerprints, and 3) molecular physicochemical properties. The graph-based molecular representation was explicitly generated for generating graph neural network models. Additionally, both molecular fingerprints and molecular properties were used to construct classifiers using traditional machine learning for comparison.

For the graph-based molecular featurization in this study, the representation of each node in the molecular graph (atom types) was learned using embedding technology. Similar to the embedding methods in the natural language process (Kenter, Borisov, and De Rijke 2016), the input of the embedding model was the collection of one-hot-vectors for each atom type surrounding a specific atom type in a molecule, and the output was the one-hot-vectors of the center atom. The weights between the input layer and the first hidden layer were extracted from the initiated embedding vector for each atom type. Those embedding vectors were optimized by maximizing the conditional probability of accurately predicting the central atom type, given the information of surrounding atom types. All 53 atom types were originally defined in the general AMBER force field (GAFF), which was designed for rational small molecular drug discovery (Wang et al. 2006). The entire atom embedding section was processed using Python code (Python 3.7.6), and the DNN models were developed using the TensorFlow package (TensorFlow 2.1.0) (Abadi et al. 2016).

The molecular fingerprint is a class of descriptors that records the chemical substructures in binary/numerical data format. Multiple types of molecular fingerprints were calculated as the molecular representation when building traditional ML models. Those fingerprints included extended connectivity fingerprints (ECFP), functional class fingerprints (FCFP), marine corps community services fingerprints (MCCS), and atom-pair fingerprints(Rogers and Hahn 2010; Cheng et al. 2017; Polton 1982). The calculation of molecular fingerprints for each compound was proceeded using the RDKit toolkit (Landrum 2013). A total of 73 physicochemical properties of small molecules, such as molecular weight (MW), topological polar surface area (TPSA), and hydrogen-bond donor/acceptor (HBD/HBA), were also calculated using RDKit toolkit (Landrum 2013).

3.1.2.3 Graph neural networks development for hERG inhibition classification

GNN is a type of deep learning approach to handling non-Euclidean, such as graph data(Zhou, Cui, et al. 2020). In chemistry, the chemical structure of a small molecule is commonly depicted as a graph, in which atoms are graph nodes, and molecular bonds are edges. Therefore, this graph format of molecular representation can be used as the input of GNNs to build a graph-based learning system, which can exploit both the potential interactions between atoms and the property of the entire molecule, such as bioactivity for drug discovery. In this study, we developed a novel framework to specify hERG inhibitors from non-inhibitors using deep learning methods with GNNs architecture.

The general procedure to develop a GNNs classification model involves two main sections: node-level graph embedding and graph-level classification (Wu et al. 2020). The first node-level graph embedding section aims to learn the representation of the network as a set of multi-dimensional vectors (Cai, Zheng, and Chang 2018). Those vectors represent both the node information and the network topological information. All the node features will then be linked together using different aggregation methods based on the adjacency information. For example, in a convolutional graph neural network, the adjacency matrix can be used as the basic convolutional kernel or can be used to construct more complex kernels such as the Laplacian matrix. Noteworthily, all the atoms in one molecule are sorted in ascending order according to their degrees. This order will be retained throughout the operations in the model. The implicit integration of the features of non-immediate neighboring atoms is implemented in the GNN operations. Since the output of the first GNN layer for each atom includes the atomic features of immediate adjacent atoms, in the next GNN layer, the operation considers the atoms that are next to the direct neighboring atoms, which is analogous to the situation of radius equals to 2 in ECFP. It is plausible that when we increase the number of GNN layers, the radius of the local substructure will increase. Therefore, the aggregation step can also be considered the extraction of high-level representations. Given the potential effect of the size of the substructure in classification, the number of GNN was the model parameter of primary concern in this study. Next, a graph-level classification section such as a deep neural network can be added, and then an entire graph-based deep learning classification model is built. The entire GNN process was implemented using the TensorFlow package (Abadi et al. 2016) on Python.

3.1.2.4 Comparative models and hyperparameter optimization

Comparative models were generated using traditional machine learning algorithms. Those algorithms involved random forests (RF), gradient boosting (GB), support vector machine (SVM), naïve Bayes, K nearest neighbor (kNN), and multilayer perceptron (MLP) (Ma, Wang, and Xie 2011b; Breiman 2001b; Gardner and Dorling 1998; Myint et al. 2012; Wang 2005; Solomatine and Shrestha 2004; Patrick and Fischer III 1970; Rish 2001). The descriptors adopted for training those models were molecular fingerprints and molecular properties, which were introduced in the previous section 2.2.2. All models were developed using the *scikit-learn* package in Python (Pedregosa et al. 2011a)

The optimization of model architectures was processed based on the Bayesian optimization algorithm (Snoek, Larochelle, and Adams 2012). Specifically, for deeplearning-based models, including the deep graph neural network models, the optimization was completed using the *HParams* module in *TensorFlow* (Abadi et al. 2016). The following hyperparameters were considered for optimization: dropout rate, type of optimizer, learning rate, batch size, number of fully connected hidden layers, and number of graph neural network layers. In the hyperparameter tuning, we selected the most optimized hyperparameter setting for each model showing the highest accuracy on the 10-fold-cross-validation (Shao 1993). In addition, the hyperparameter tuning for traditional machine learning models adopted the *skopt* python package (Pedregosa et al. 2011a).

3.1.2.5 Model evaluation metrics

For the model evaluation in the training and validation process, receiver operating curve (ROC) analysis was applied to evaluate model performance (Metz 1978), with the ROC curve plotted by false-negative rate (FNR, 1-specificity) against true positive rate (TPR, sensitivity, recall) at all classification thresholds. The area under the curve (AUC) represents the degree or measure of separability. This area under the receiver operating curve (ROC AUC, also called AUROC), ranging from 0.5 to 1, specifies the accuracy of the classification model binning the sample into its correct category. For the evaluation of predictive performance on the test set, five metrics, including 1) recall, 2) precision, 3) accuracy, 4) F1 score, and 5) Matthew’s correlation coefficient (MCC) score, were considered comprehensively through the radar chart. The detailed formulas are shown below:

$$\text{FNR} = \frac{\text{FN}}{\text{FP} + \text{TN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{sensitivity (recall)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4}$$

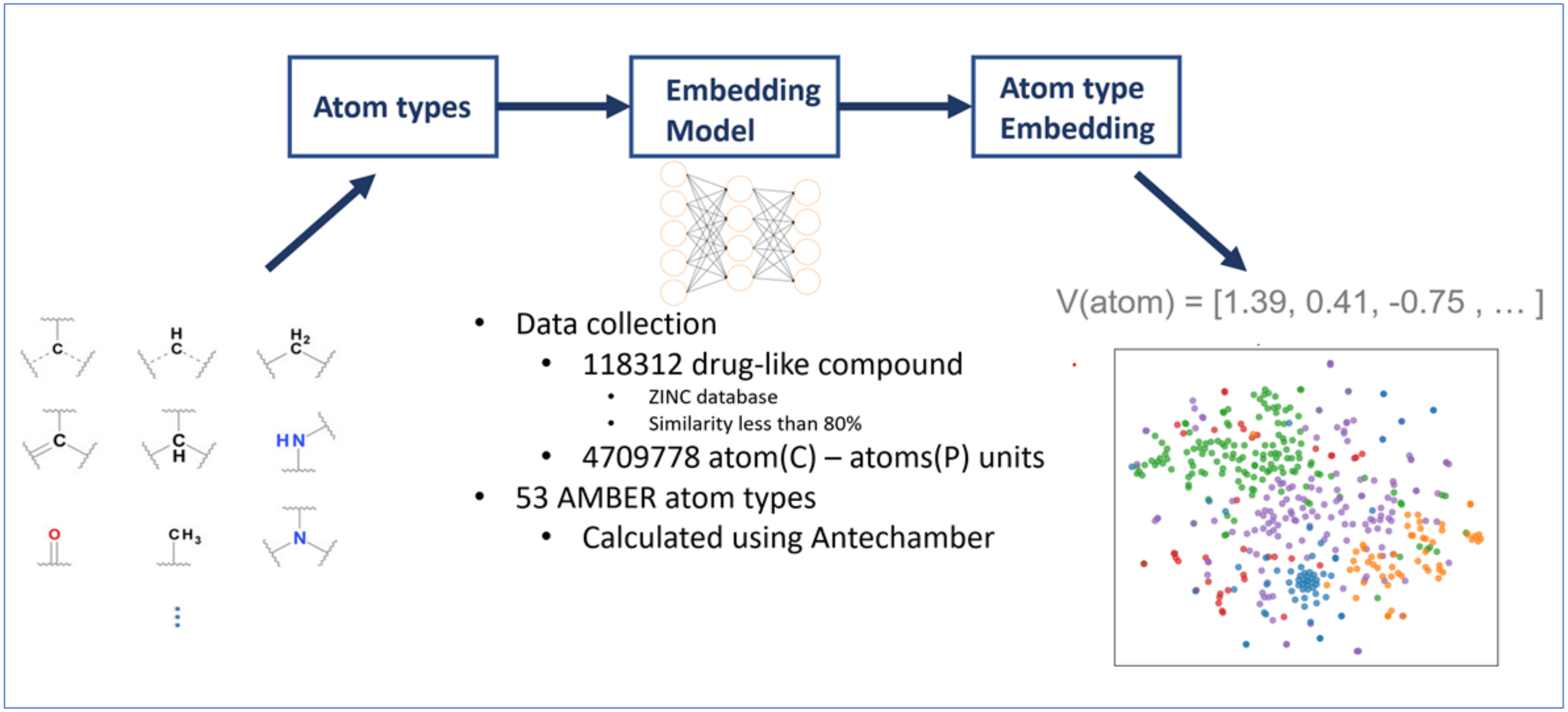
$$\text{F1 score} = \frac{2 * \text{TP}}{2 * \text{TP} + \text{FP} + \text{FN}} \tag{5}$$

$$\text{MCC score} = \frac{\text{TP} * \text{TN} - \text{FP} * \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}} \tag{6}$$

3.1.3 Results and Discussion

3.1.3.1 Atom type embedding using neural network models

Deep neural network (DNN) embedding models were built to learn the mathematical representation of a total of 57 predefined atom types. **Figure 3.1.2** illustrates how the input Layer in our DeepGhERG learns the graph-based molecular features. The training data set contained 123182 drug-like compounds initially collected from the ZINC database (Sterling and Irwin 2015) and then diversely selected based on the molecular fingerprint similarity. A different number of hidden layers (1, 2, 3, 4, 5, 6) in the DNN model were tested, and 4 hidden layers were selected, achieving an accuracy of 93% in the 10-fold cross-validation.

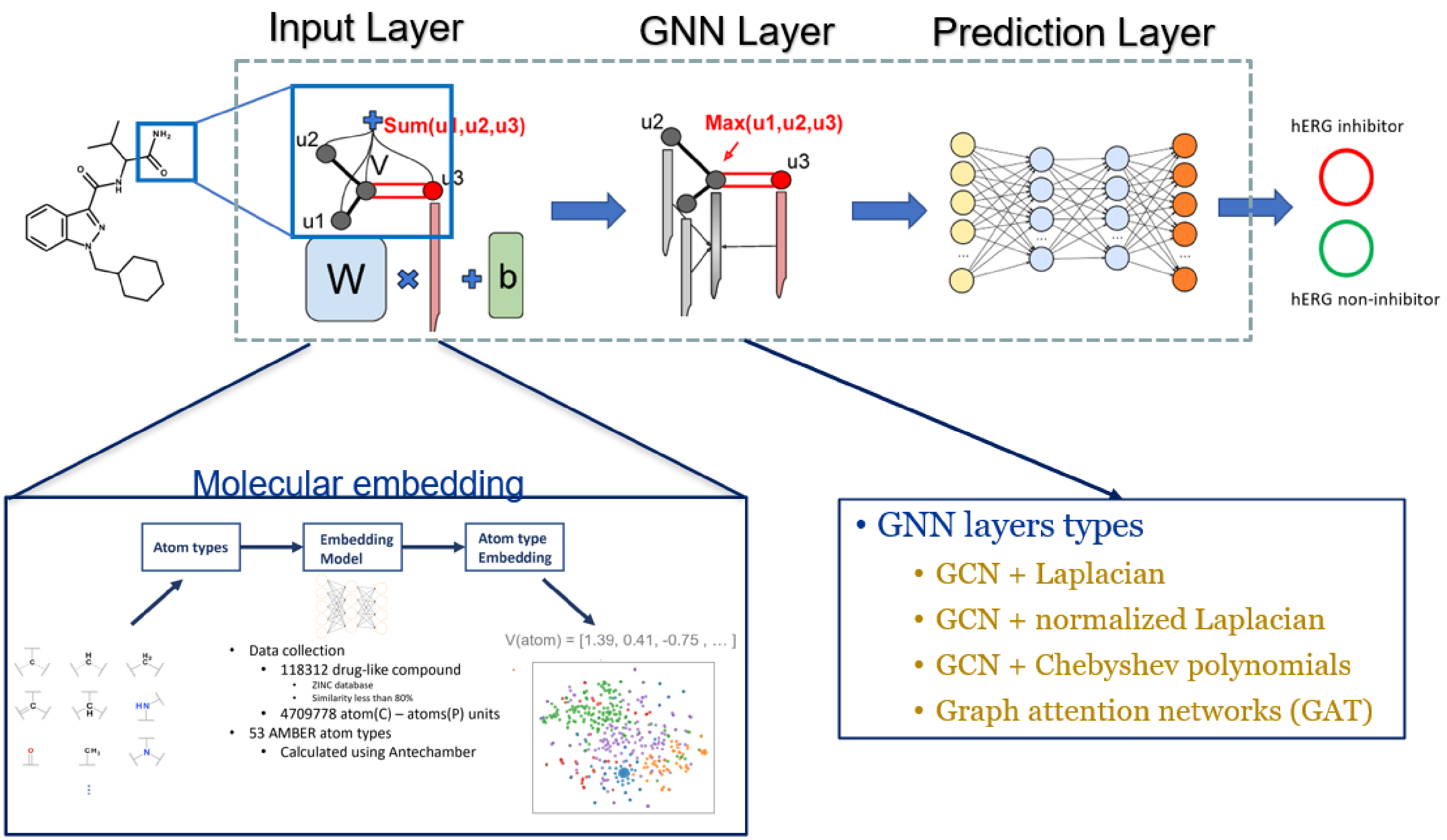


- Data collection
  - 118312 drug-like compound
    - ZINC database
    - Similarity less than 80%
  - 4709778 atom(C) – atoms(P) units
- 53 AMBER atom types
  - Calculated using Antechamber

Figure 3.1.2 Atom type (Amber) embedding using deep neural networks. In this molecular graph-based feature calculation, we used amber atom type to represent both node and edge features. To train this embedding model, a drug-like compounds dataset was used. The key results of this molecular embedding model are the mathematical representation of all the AMBER atom types

3.1.3.2 GNN model construction and optimization

To predict reliable hERG inhibition of small molecules, the GNN-based classification model was designed, as shown in **Figure 3.1.3**. Firstly, the graph-based molecular feature was applied as the input data using the embedded vector for each atom type. The topology & connectivity of all the atom types for a molecular was extracted using the adjacency matrix. By using the adjacency matrix, for each atom in the molecule, its neighboring atoms can be identified. Those atomic features of the corresponding atom and its neighboring atoms will be added up and projected to another space. Secondly, four types of GNN architectures were adopted to construct the models, which included: 1) convolutional graph neural networks using the adjacency matrix as the filter; 2) convolutional graph neural networks using normalized Laplacian matrix as the filter; 3) convolutional graph neural networks using Chebyshev polynomials of the Laplacian matrix as the kernel; 4) graph attention networks. In addition, both the dropout layer and pooling layer were added after the GNN layers as the optimization of the model. Thirdly, after the GNN and pool layers, the resultant features vectors of each atom will be flattened into a 1D vector using fully connected (FC) hidden layers. Then a Softmax function component was added to the architecture for the classification. The complexity of the GNNs was tested by adding more GNN layers (1 layer to 6 layers) and FC layers (1 layer to 5 layers).



- Data collection
  - 118312 drug-like compound
    - ZINC database
    - Similarity less than 80%
  - 4709778 atom(C) – atoms(P) units
- 53 AMBER atom types
  - Calculated using Antechamber

- GNN layers types
  - GCN + Laplacian
  - GCN + normalized Laplacian
  - GCN + Chebyshev polynomials
  - Graph attention networks (GAT)

Figure 3.1.3 Overall GNN architecture of DeepGhERG for predicting hERG inhibition. This figure shows how DeepGhERG predicts hERG inhibition. First, the query compound is transferred to a graph-based feature using the established embedded model. Then the graph-based feature is processed by deep GNNs for pattern abstraction and recognition. As mentioned, four types of GNN architectures mentioned are adopted. At last, the extracted information is passed into a fully connected neural network for classification purposes.

3.1.3.3 Performance comparison with each comparative model

The comparison of cross-validation performance between different GNN models and traditional machine learning models is presented in **Figure 3.1.4**. The validation results show that GNNs models performed better than comparative models developed using traditional machine learning models, regardless of which type of molecular representations were used. All types of GNNs architecture benefited the model performance, with the GNN model using Chebyshev kernel (GCN_Chebyshev Attention in Figure 3.1.4) achieving the best predictive accuracy 0.8213. Among the traditional machine-learning-based comparative models, the SVM model with ECFP and RF models with atom-pair fingerprints performed better. It is noticed that the Naïve Bayes model did not perform well, compared to another ML method. This is mainly because of the assumption of independence among predictors in Bayes' Theorem; however, in the cheminformatics area, most of the predictors for small molecules are correlated with each other. One way to minimize this issue may be to add a feature learning process to remove correlated predictors or use principal component analysis to perform a dimension reduction and generate independent new components.

Next, we evaluated the predictive performance of GNNs-based models and comparative models developed using traditional machine learning algorithms using an independent test set. Five different evaluation metrics were adopted to systematically assess the model predictability (**Table 3.1**), and the evaluation results were also visualized using the radar chart (**Figure 3.1.5**). Overall, the GNN-based models could achieve reliable predictive performance, and the graph attention model (GATConv) did slightly better when predicting the test set compounds.

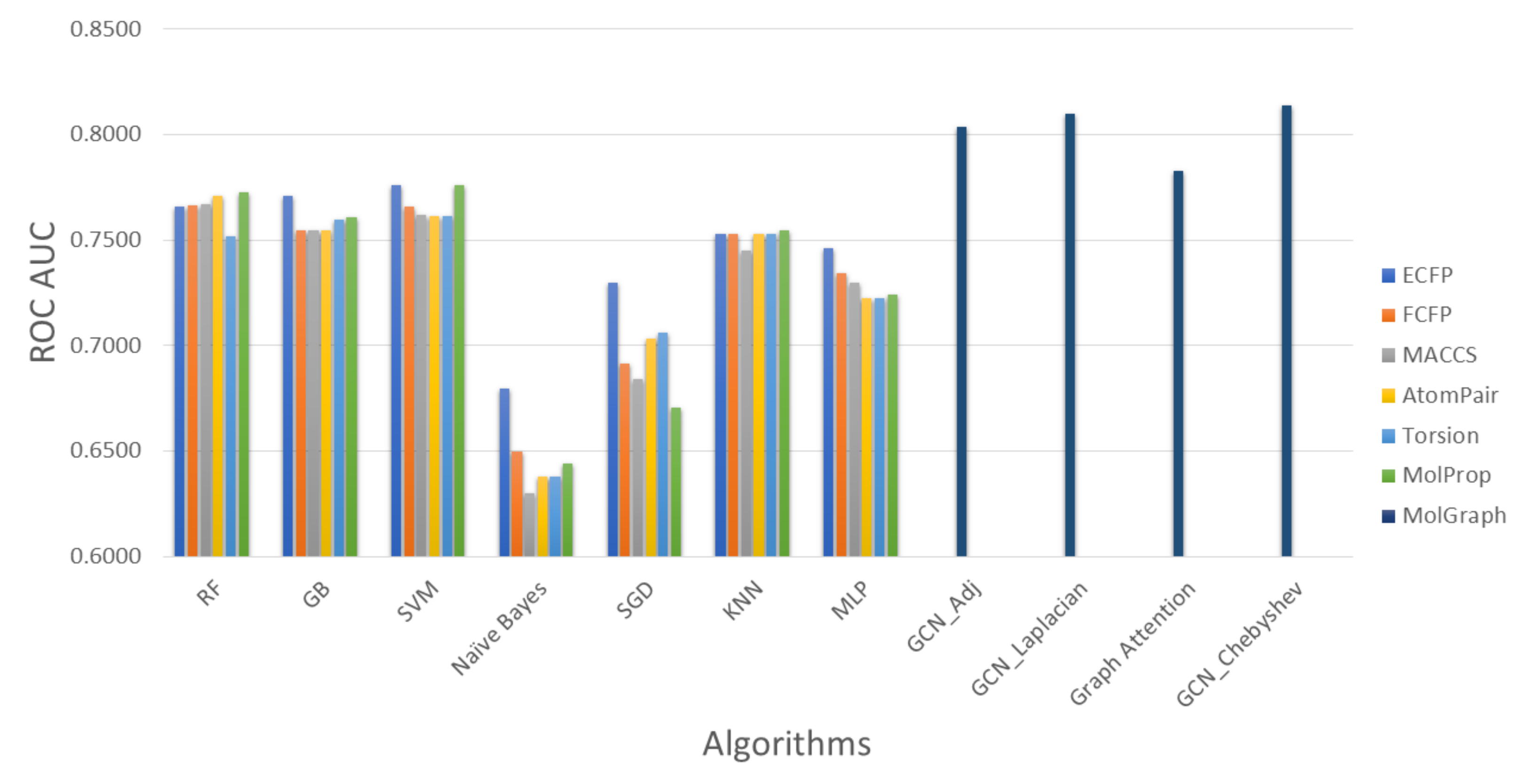


Figure 3.1.4 Performance of the test accuracy over all the models. RF: random forest; SVM: support vector machine; GCN: graph convolutional network; GAT: graph attention network; ChebConv: graph convolutional network using Chebyshev polynomials kernel; GB: gradient boosting; KNN: k nearest neighbor; MLP: multi-layer perceptron; SGD: Stochastic gradient descent.

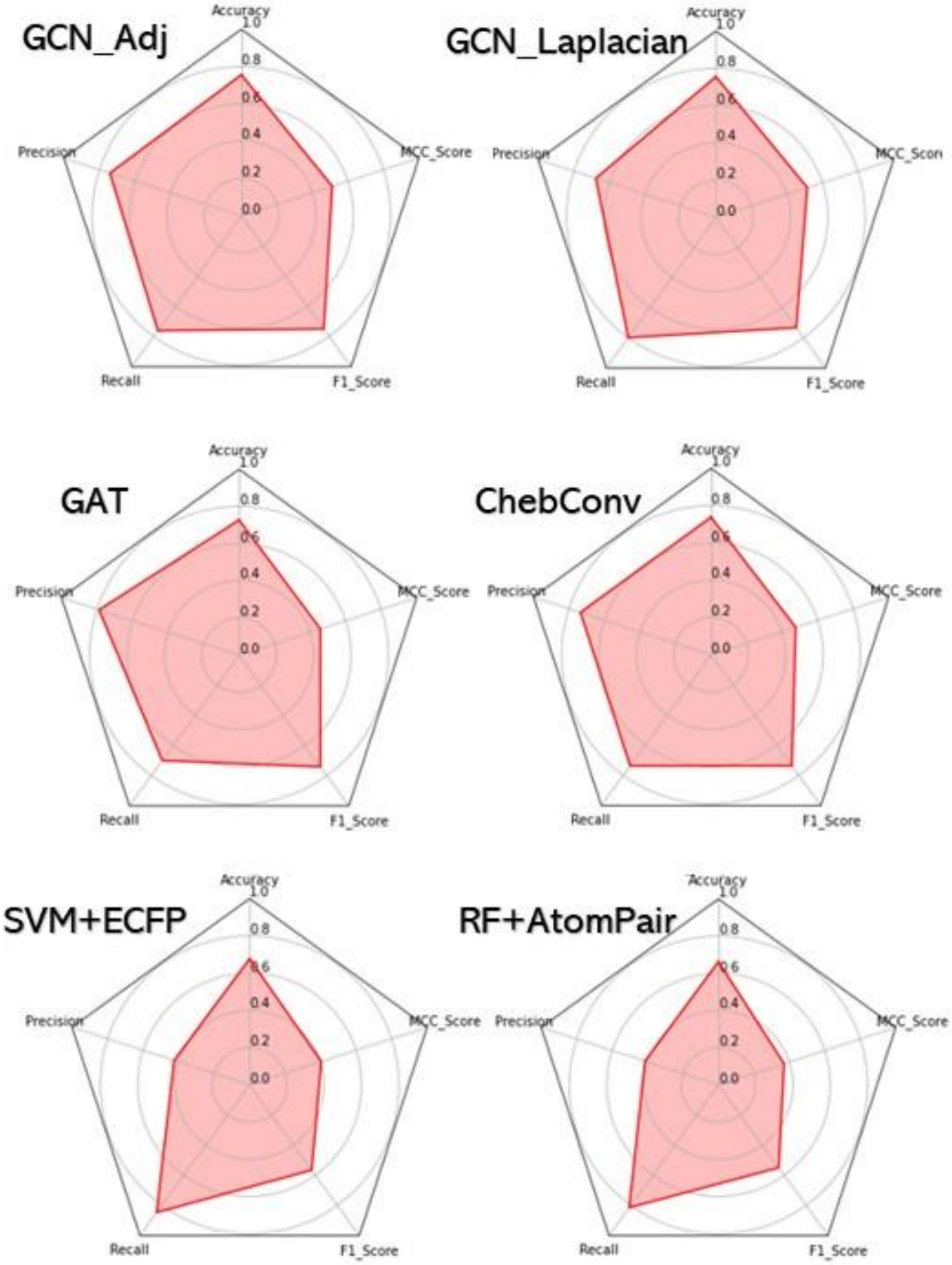


Figure 3.1.5 Performance of the extra test accuracy over the deep graph-based models. MCC_Score: Matthew's correlation coefficient; F1_score: F1 means, or the harmonic mean of precision and recall; Precision: true positive rate; Recall (sensitivity): the number of true positives divided by the number of true positives plus the number of false negatives

Table 3.1 Evaluation of different methods on the test set

	Accuracy	Precision	Recall	F1_Score	MCC_Score
GCN_Adj	0.6764	0.4249	0.8477	0.5661	0.4041
GCN_Laplacian	0.6625	0.4148	0.8150	0.5497	0.3702
GATConv	0.7383	0.7354	0.7372	0.7363	0.4766
ChebConv	0.7547	0.6768	0.7988	0.7328	0.5150
SVM+ECFP	0.7269	0.7888	0.6998	0.7416	0.4579
RF+AtomPair	0.7360	0.7405	0.7318	0.7210	0.5121

RF: random forest; SVM: support vector machine; GCN: graph convolutional network; GAT: graph attention network; ChebConv: graph convolutional network using Chebyshev polynomials kernel; MCC_Score: Matthew's correlation coefficient; F1_score: F1 means, or the harmonic mean of precision and recall; Precision: true positive rate; Recall (sensitivity): the number of true positives divided by the number of true positives plus the number of false negatives.

3.1.3.4 Comprehensive performance comparison with an external test set

Finally, we compared the prediction performance of our GNNs models to published models using an external test dataset containing 30 hERG blockers and 14 hERG non-blockers. External data and all prediction performance of published models were retrieved from the publication of the *DeepHIT* tool (Ryu et al. 2020). Our best GNNs model (ChebConv) showed an accuracy of 0.84, precision of 0.83, recall of 0.93, F1 score of 0.88, and MCC score of 0.66 (**Table 3.2**). According to the results of the comparative analysis, our GNNs model overall performed best, demonstrating that our GNNs model can predict hERG non-blockers more reliably than other prediction tools.

Table 3.2 Evaluation of different methods on the external test set

	Accuracy	Precision	Recall	F1_Score	MCC_Score
GCN_Adj	0.6136	0.5667	0.8095	0.6667	0.2620
GCN_Laplacian	0.7955	0.8000	0.8889	0.8421	0.5603
GATConv	0.7045	0.7000	0.8400	0.7636	0.3896
ChebConv	0.8409	0.8333	0.9259	0.8772	0.6605
DeepHIT	0.7727	0.8333	0.8333	0.8333	0.4762
CardPred	0.3636	0.0667	1.0000	0.1250	0.1491
OCHEM_ConsensusI	0.4318	0.2000	0.8571	0.3243	0.1637
OCHEM_ConsensusII	0.7045	0.8000	0.7742	0.7869	0.3063
Pred_hERG_V42	0.6136	0.6333	0.7600	0.6909	0.1925

3.1.4 Conclusion

In this section, a novel DL-based method in predicting the drug cardiotoxicity was introduced. Significantly, the predictive performance of our novel GNNs-based methods in distinguishing hERG blockers and non-blockers was examined and compared to existing methods. The best prediction accuracy over the test set already achieved 0.75, with the sensitivity/recall achieving 0.80 and the F1 score achieving 0.73. It performed better than the best machine learning methods in the validation (SVM and RF), indicating that when dealing with complex tasks such as identifying hERG ion channel blockers. The inhibition of the hERG channel is the primary cause of such cardiotoxicity; however, other ion channels also contribute to this in a synergetic form. In the future, we plan to focus on those types of cardiac ion channels using our established method and provide a more comprehensive evaluation of small molecular-induced cardiotoxicity.

3.2 Blood-Brain Barrier (BBB) Permeability Prediction using ML/DL Methods

3.2.1 Research Background to BBB Permeability

3.2.1.1 Basic principle of BBB permeation

The BBB is a physiological and biochemical barrier between the CNS and the peripheral tissues. BBB serves as the primary active interface between the changeable blood environment of CNS and the extracellular fluid (Redzic 2011) and prevents the neurotoxic plasma components, blood cells, and pathogens from entering the brain (Sweeney et al. 2019). By regulating the transport of small molecules or macromolecules into and out the brain, BBB only permits the movement of selective molecules essential for keeping the brain function to maintain the homeostasis of the CNS (Małkiewicz et al. 2019).

Generally, BBB is a capillary wall composed of brain endothelial cells, basement membrane, pericytes, vascular smooth muscle cells, astrocytes, and others (Sharif et al. 2018). And the physiological structures responsible for BBB transport mainly include the tight cell-cell junctions, endothelial and pericyte transporters, and perivascular transport (Sweeney et al. 2019). There are several transport routes for molecules to cross the BBB: paracellular and transcellular Diffusion, Carrier-Mediated Transport, and transcytosis (Dong 2018).

The transport of most of the molecules between the brain and vascular system is mainly through transcellular transport (**Figure 3.2.1**). Therefore a few small molecules can enter the brain by paracellular and transcellular diffusion due to the tight junctions between capillary endothelial cells (Wong et al. 2019). However, the brain endothelial cells also provide alternate transport pathways, such as through active efflux transport proteins, including P-glycoprotein (P-gp) and breast cancer resistance protein (BCRP); and other substances including large molecules may access the brain through transcytosis (Bors and Erdő 2019). In addition, BBB also helps the clearance of metabolites and toxins in the brain and regulates the composition and volume of the cerebrospinal fluid (Małkiewicz et al. 2019).

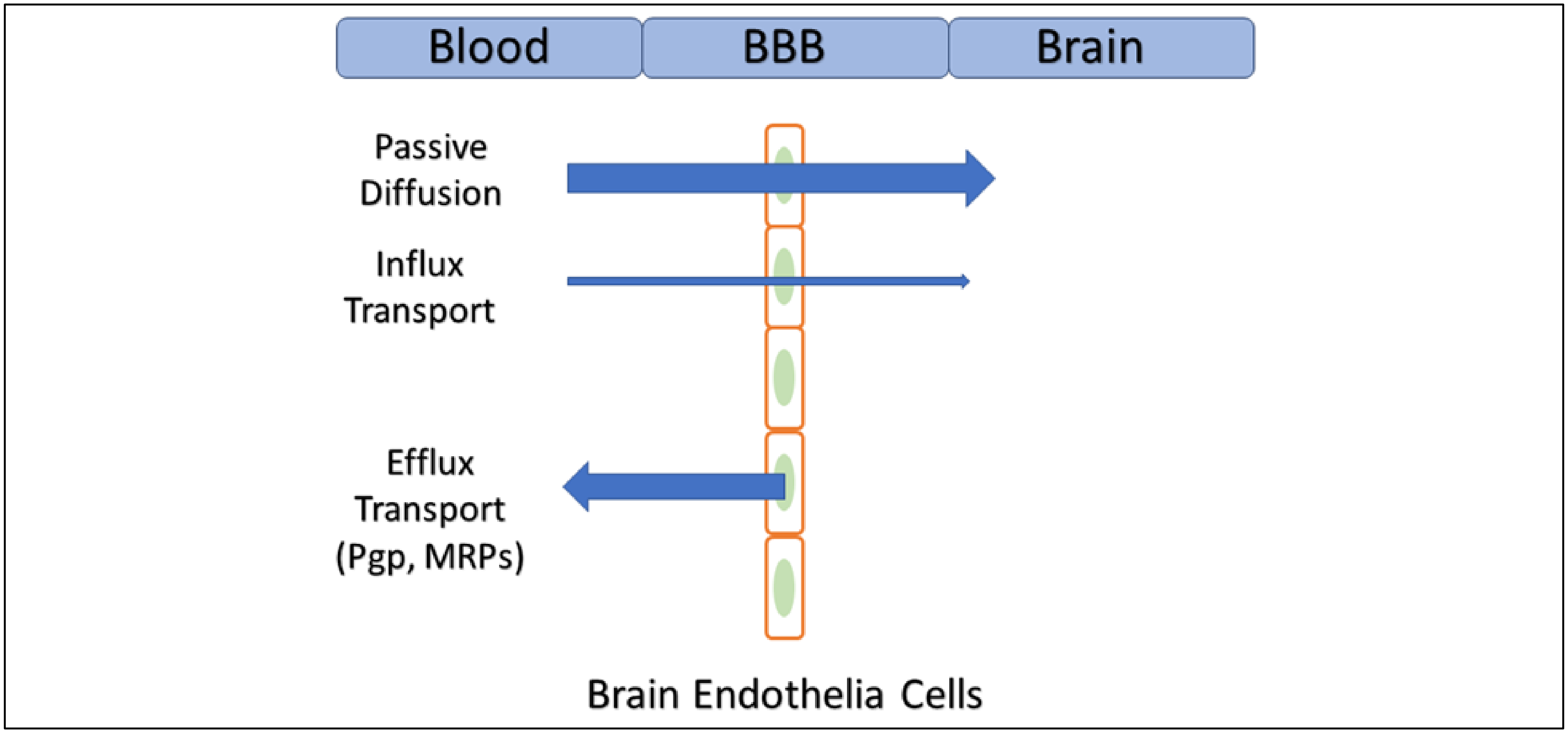


Figure 3.2.1 General pathway across the BBB

3.2.1.2 Role of the Blood-Brain Barrier in drug delivery

Disorders of the central nervous system (CNS) such as Alzheimer's disease (AD) are becoming one of the major burdensome disease areas. Up to the year 2017, hundreds of millions of people are suffering from at least one type of those CNS disease, including migraine (68.5 million people) and AD and other dementias (2.9 million people) (Collaborators et al. 2021). Also, the deaths from major CNS disorders were ranked third (10.8%) in the US among all causes of death and fifth (5.5%) globally ((IHME) 2019). As the aging of the population, the number of people affected by CNS diseases has substantially increased in the past several decades, and there is no doubt that the number will continue rising in the future. On the other hand, the global sales of CNS disease drugs and related therapeutical products totaled more than $80 billion in the year 2019, and were forecasted to be more than $100 billion in the year 2022 (Dealmakers 2020).

However, the failure rate for the effective drug targeting CNS diseases is very high compared to most other non-CNS areas of drug discovery. There is an urgent need for medications against many CNS diseases that lack effective treatment, such as AD (Gribkoff and Kaczmarek 2017). For example, the current FDA-approved drugs (e.g., donepezil, memantine) for AD can only relieve symptoms rather than treat the disease. The blood-brain barrier (BBB) is the major hurdle for CNS drug delivery; therefore, BBB must be considered to develop a successful treatment for those CNS diseases (Wong et al. 2019). However, the protective characteristics of BBB make it one of the most complicated microenvironments in drug discovery and limit the development of novel drugs targeting CNS diseases. For example, more than 98% of small molecule drug candidates have limited delivery to the brain (Pardridge 2005), with water-soluble molecule drugs in the blood being prevented from entering the CNS and lipophilic molecule drugs being excluded by efflux transporters (Banks 2009). Thus, because it is vitally important to optimize the BBB impermeability of the drug for the effective treatment of CNS diseases, increasingly more research efforts are devoted to this topic.

3.2.1.3 Experimental methods for assessing BBB permeability

Consequently, scientific interest in BBB physiology and pathology led to numerous experimental models. These models emanated from research aimed at accelerating the development of effective drugs to treat CNS diseases. BBB models have been developed using both in-vitro and in-vivo methods (Bagchi et al. 2019; Vastag and Keseru 2009). In-vitro methods for evaluating BBB permeability of small molecules involve two types. One is to test the physiochemical properties of the chemical, such as Log P and Log D (Leo, Hansch, and Elkins 1971). One example is the parallel artificial membrane permeability assay (PAMPA) method that evaluates passive BBB permeability using porcine brain lipid in dodecane as the artificial permeability membrane (Ottaviani, Martel, and Carrupt 2006). Other in-vitro methods to assess BBB permeability are cell-based assays, such as the Caco-2 method and MDR1 methods. Those assays have inherent appeal as in vitro models of BBB permeation because they use living cells and are more similar to the BBB than the physiochemical-based methods; however, they may not closely resemble the complex conditions at the BBB.

The in vivo B/P experiment, which provides the brain distribution data of chemicals, is one of the standard approaches to evaluate BBB permeability, and it is widely used in the industry (Kerns and Di 2008). Overall, there are various methods available for assessing brain penetration, which is relatively expensive and time-consuming. The computational approach can also be applied to supplement and is an alternative to costly and labor-intensive experiments carried out in laboratories in drug discovery. These methods thus increase the survival of drug candidates and move the medicinal chemistry design to a higher probability space for success.

Therefore, to help to reduce the cost and time for CNS drug discovery, it is essential to explore computational methods to access the BBB permeability.

3.2.1.4 Current computational method to predict BBB permeability

Many review articles have already summarized different computational methods for BBB permeability prediction (Pardridge 1998; Pajouhesh and Lenz 2005; Clark 2003; Gupta 1989; Bradbury 1993; Saxena et al. 2019). Simple rule-of-thumb methods were applied to select compounds with potential good permeability; for example, a rule-based scoring system called central nervous system multiparameter optimization (CNS MPO) was introduced by Wager et al. from Pfizer to select optimal CNS molecules (Wager et al. 2010). Traditional QSAR/QSPR analyses were widely applied for predicting those experimental evaluations of BBB permeability, such as B/P ratio or logBB (Österberg and Norinder 2000; van de Waterbeemd et al. 1998). For example, Kelder et al. developed a QSAR model to predict BBB permeability using a relatively large dataset containing 2366 compounds (776 CNS drugs, 1590 non-CNS drugs), concluding that general CNS drugs had a much less PSA compared to non-CNS drugs (Kelder et al. 1999). Those traditional QSAR/QSPR methods to predict BBB permeability are widely used in modern drug discovery since QSAR/QSPR helps with the understanding of the effect of structure on activity/property, leading to the synthesis of novel analogs.

Those traditional QSAR/QSPR studies generally applied linear methods to analyze the BBB penetration of small molecules (Plisson and Piggott 2019). However, the correlation between BBB permeability and compound descriptors can be more complex. With the help of machine learning, more and more complex but accurate models were developed using larger training data set in the last decade. For example, Tropsha’s group published their BBB predictive regression models using combined machine learning methods (Zhang et al. 2008). In their methods, the k-nearest neighbors (k-NN) method was used for selecting important features from a total of 346 Dragon, MOE, and MolConnZ descriptors, and the support vector machine (SVM) regression algorithms were used for model construction. Besides using machine learning in regression tasks for predicting the exact value of BBB permeability parameters such as logBB, many computational models were developed attempting to discriminate BBB permeable compounds from non-permeable compounds (Singh et al. 2020; Lingineni et al. 2017).

Generally, in those classification models, compounds were labeled as CNS+ (or CNS-) based on the in vivo or in vitro experimental results, such as B/P ratio or logBB, indicating that a compound can penetrate the brain (or not). One of the advantages of generating classification models is that labeling compounds into binary classes (CNS+/CNS-) may be considered a relatively reasonable way to merge several small datasets from multiple resources into a large one, as their experimental standards for validating BBB permeability were various (Kunwittaya et al. 2013). For example, Kortagere and colleagues developed classification models using the SVM classification technique with a combined BBB dataset from multiple resources (Kortagere et al. 2008). Those individual datasets were relatively small and used different experimental standards to get the logBB value. By using different thresholds of logBB value suggested by the authors to specify BBB+ and BBB-compounds, it would be hard to merge those small datasets into a large dataset. Recent studies also tried to use different types of data in their BBB classification modeling works, such as solvation energy descriptors and binary molecular fingerprints descriptors (Roy, Hinge, and Kovalenko 2019; Wang et al. 2018; Yuan, Zheng, and Zhan 2018; Zhang, Liu, et al. 2016). Overall, machine learning methods significantly promoted the development of computational prediction of BBB permeability in early drug discovery and preclinical research.

In recent years, deep learning methods have been used in drug discovery (Jing et al. 2018b; Miao et al. 2019). Specifically, there have been several studies using deep learning for BBB permeability prediction. Recently, a deep learning model using recurrent neural network (RNN) was reported by Alsenan and colleagues (Alsenan, Al-Turaiki, and Hafez 2020). RNN is a deep learning architecture for dealing with sequential data, and it is widely used in natural language processing and signaling processing. In the study, they used molecular fingerprints as the molecular sequence to be fitted into the RNN model and attempted to classify molecular into high penetration group (BBB+) and low penetration group (BBB-). Although deep learning is a powerful new technique for developing predictive models, only a few BBB permeability studies were reported using deep learning to address this gap in research. We explored the capability of several DL-based methods on discriminating BBB+ and BBB- compounds and proposed a novel in silico framework for evaluating the BBB permeability of small molecules.

3.2.2 Methods

3.2.2.1 Data preparation

Chemical information from diverse published resources was combined to generate the dataset used in this study, which contained 1924, including 1465 BBB+ and 353 BBB− compounds). The entire dataset was collected from multiple publicly available datasets (Adenot and Lahana 2004; Li et al. 2005; Zhao et al. 2007; Yuan, Zheng, and Zhan 2018). BBB permeability is modeled as a classification task, in which a molecule is categorized into either a high penetration rate (BBB+) class or low penetration rate (BBB-) class. The classification of BBB+ and BBB− was based on the BB ratio, representing the ratio of total steady-state concentration in the brain to blood. Compounds assigned BBB+ should have a sufficiently high penetration rate (with the BB ratio ≥ 0.1) or has been known as CNS drugs or drug candidates under clinical development; while compounds assigned as BBB− should have a relatively low penetration rate (with the BB ratio < 0.1) or have been known not to cross the BBB. The chemical structures were retrieved from the SMILES string and then converted to the SDF format using the RDKit toolkit (Landrum 2013). The 3D conformations of the compounds were generated and optimized using the LigPrep package of the Schrodinger software (Release 2017), with the protonation or deprotonation states of compounds calculated in pH 7.4. To reduce biases from the training set selection, the dataset was split into the training and the test sets using a diverse selection. Firstly, the entire dataset was clustered based on the chemical diversity represented by molecular fingerprints. Next, compounds in each cluster were selected as the training compounds until reaching the required selection number for the training set, and the remaining compounds were collected into the test set. In this way, the selected training set was expected to uniformly cover the chemical space distribution of the entire dataset, which may be considered an ideal representative subset of the original dataset.

3.2.2.2 Descriptor calculation and selection

Four types of molecular descriptors were adopted to represent the molecular structure of all compounds, including physicochemical molecular descriptors, molecular fingerprints, atom type count descriptors, and graph-based molecular representation. For physicochemical molecular descriptors calculation, 119 molecular descriptors were calculated using the RDKit toolkit (Landrum 2013), included MW, SlogP, TPSA, NumHBD, NumHBA, etc. For molecular fingerprints calculation, five types of different molecular fingerprints were collected using cheminformatics tools (Zhao et al. 2007). The MACCS fingerprint contains 166 binary fingerprints as substructure keys, each of which indicates the presence of one of the 166 MACCS substructure keys calculated from the molecular graph. The ECFP is circular topological fingerprints with 1024 descriptors, which represent the presence of particular substructures using circular atom neighborhoods. The FCFP is a variation of ECFP, which is further abstracted in that instead of indexing a specific atom in the environment, the index that atom's role. The PubChem fingerprint covers a collection of 881 diverse substructure keys designed and used by PubChem. The atom pairs fingerprint contains 1024 bits of binary data collected based on the atomic environments and shortest path separations of every atom pair in the molecule. All those molecular descriptors were used for constructing classifiers using traditional machine learning. The atom type count descriptor was calculated by the amber force field software toolkit, which defines the specific atom types and counts the total number of atoms belonging to each atom type (Wang et al. 2006). The graph-based molecular representation introduced in section 3.1 was also generated specifically for generating graph neural network models.

For data pre-processing and cleaning, all the descriptors with low variance were removed, and the threshold of the variance was set to be 0.05. To avoid over-fitting issues, a feature eliminating process was performed using L1 regularization (Lasso) for eliminating descriptors with low correlation (Zhao and Yu 2006).

3.2.2.3 Model construction using machine learning

A prediction pipeline was developed for supervised classification with various machine learning algorithms, including random forest (RF) (Breiman 2001b), support vector machine (SVM) (Soman, Loganathan, and Ajay 2009), AdaBoost decision tree (AdaBoost) (Solomatine and Shrestha 2004), naïve Bayes (NB) (Rish 2001), logistic regression (LR) (Kleinbaum et al. 2002), and neural network/multilayer perceptron (MLP) (Gardner and Dorling 1998). The open-source Python module Scikit-learn was used for model training, data prediction, and interpretation of results (Pedregosa et al. 2011a).

For RF, the *RandomForestClassifier* function from Scikit-learn was applied. The model was saved after the optimization on parameters *n_estimators* (50, 100, 500) and *max_depth* (2, 3, 4, 5). For SVM, the *svm.SVC* method was applied with three kernel functions (linear, RBF, poly). The SVM model with the best performance was saved after optimizing penalty parameter C and parameter γ for RBF and poly kernels. For AdaBoost, the *AdaBoostClassifier* function was applied with the optimization on parameters *n_estimators* (10, 100, 1000) and *learning_rate* (0.01, 0.1, 1). The weaker classifier used in AdaBoost was set to *DecisionTreeClassifier*. When training NB models, the *BernoulliNB* method was applied for datasets with fingerprints as features. Given that Bernoulli naive Bayes requires binary-valued feature vectors for samples, the prior probabilities of the classes were set to none. For MLP, the *MLPClassifier* method was applied with the setting of different hidden layers (1-5). The following parameters were also optimized during the model training: activation function (identity, logistic, tanh, relu), and learning rate (0.1, 0.01, 0.001, 0.0001). The *LogisticRegression* was applied to implement the logistic regression model with an L1 penalty (lasso). The parameter solver was set to sag to handle the multinomial loss in large datasets. The hyper-parameters tunning of each estimator was performed using the *GridSearchCV* tools, and the best-performance models in the cross-validation were selected. Moreover, graph-based deep neural network models were constructed and compared in this study. As described in section 3.1, four types of graph neural network (GNN) architecture were applied in this study. Those include convolutional graph neural networks using the adjacency matrix as the filter (GCN_1), convolutional graph neural networks using normalized Laplacian matrix as the filter (GCN_2), convolutional graph neural networks using Chebyshev polynomials of the Laplacian matrix as the kernel (GCN_3), and graph attention networks (GAT). For fitting the GNN models, the molecular graph feature was calculated using the same atom-type embedding model trained and shown in section 3.1.

3.2.2.4 Model evaluation

Ten-fold cross-validation was performed for model generation and evaluation. The model was trained using any nine folds as training data, and the resulting model is validated on the remaining fold of data. Different statistical metrics were calculated to evaluate the performance of machine learning models from diverse aspects. The area under the receiver operating characteristic curve (AUROC) was calculated after acquiring the true-positive rate and false-positive rate . AUROC can be referred to indicate the performance of the model on separating classes, while Balanced F-score was calculated as the weighted average of the precision and the recall. The accuracy classification score (ACC) was calculated to compute subset accuracy that whether the label predicted for one sample matches with the corresponding true value. In addition, Matthew’s correlation coefficient (MCC) was calculated to measure the quality of binary and multiclass classifications. MCC score is a balanced measure that both the true and false positives and negatives are considered.

3.2.3 Results and Discussion

3.2.3.1 Overall workflow

The schematic illustration of the workflow of this study is shown in Figure 3.2.2. Compounds with the experimental label of BBB+ or BBB- were extracted and cleaned from the multiple public resources. Four types of features, molecular descriptors, molecular fingerprints, atom count descriptors, and atom-type-based graphic features, were calculated for the entire compound sets. The training sets and test sets were divided at a 3:1 ratio. Six supervised machine learning algorithms were applied to build classifiers for each of the prepared training sets. Different types of features can evaluate the properties of compounds from diverse aspects, and various machine learning algorithms may favor distinctive data structures. In the end, all the models were evaluated and compared to achieve the best model.

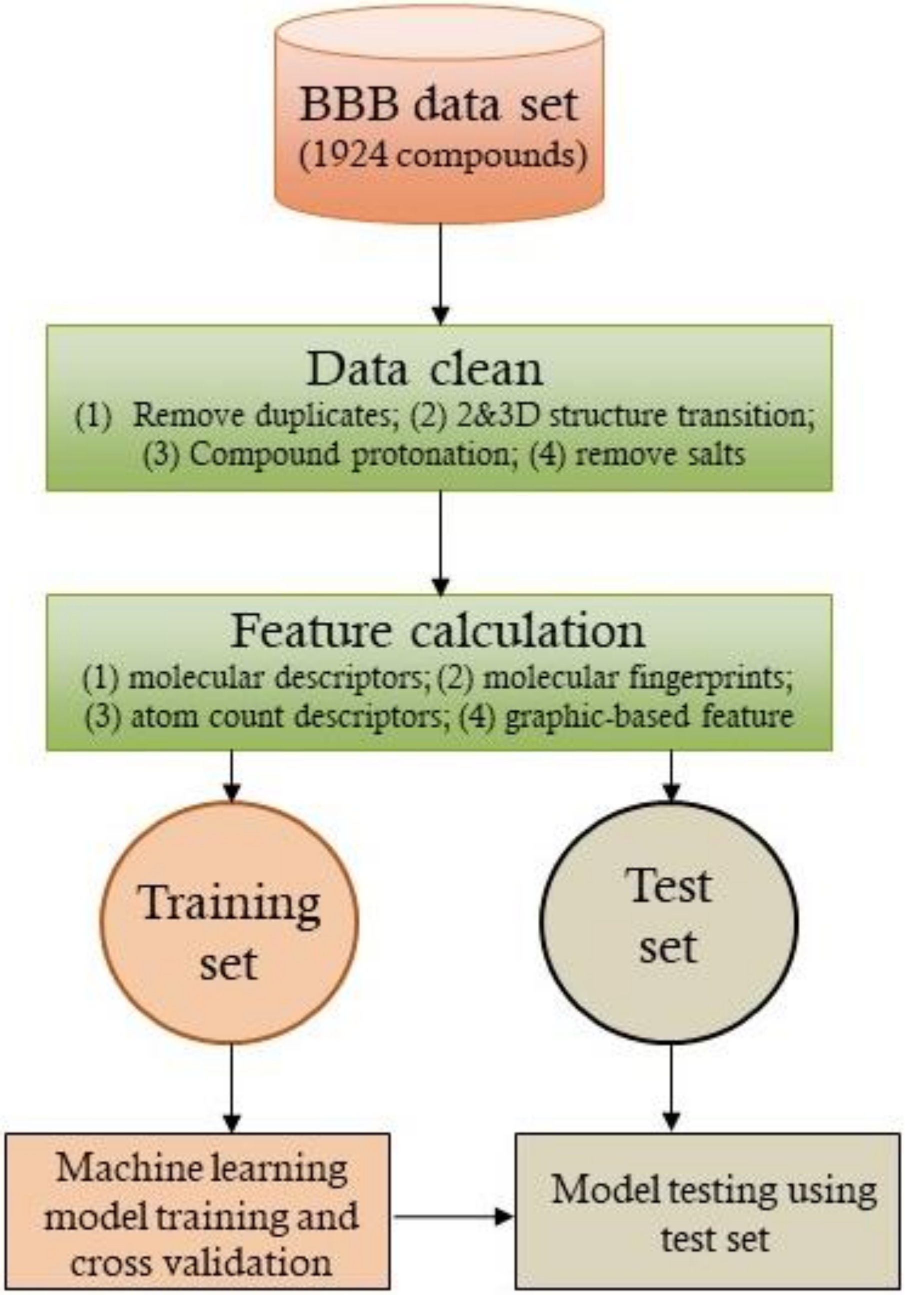
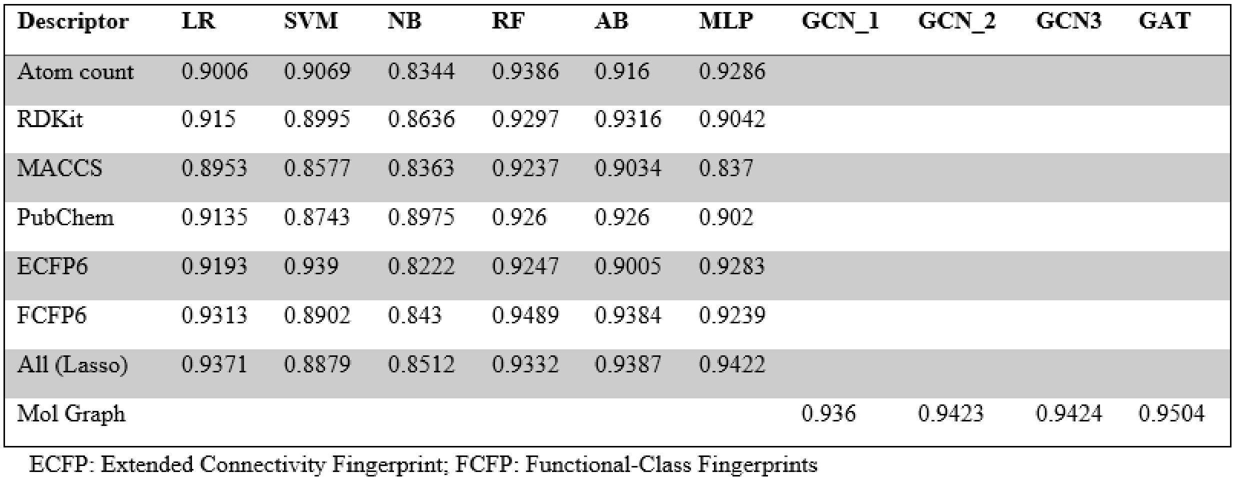

BBB data set
(1924 compounds)
Data clean
(1) Remove duplicates; (2) 2&3D structure transition;
(3) Compound protonation; (4) remove salts
Feature calculation
(1) molecular descriptors; (2) molecular fingerprints;
(3) atom count descriptors; (4) graphic-based feature
Training set
Test set
Machine learning model training and cross validation
Model testing using test set

Figure 3.2.2 Overall workflow for data processing. First, the entire database which contained 1924 (1465 BBB+ and 353 BBB− compounds) were cleaned and then denaturalized using 4 types of molecule features. The cleaned dataset was split into a training set and test set with the ratio 0.75: 0.25, with the training set, used to train and validate the ML/DL model and the test set used to evaluate the model performance.

3.2.3.2 Prediction results in cross-validation

The ROC AUC value of all the machine learning models for the cross-validation is summarized in Tables 3. Models gave relatively consistent performances when using the same types of molecular descriptors. Overall, the four GNN models (GCN_1, GCN_2, GCN3, and GAT) outperformed all the other algorithms, with the highest predictive accuracy reaching 0.95. For traditional machine learning methods, RF models and MLP models performed better than other methods. From the perspective of molecular descriptors, besides graph-based molecular representation used for GNN, FCFP molecular fingerprints provided more information helping the machine learning algorithms to learn a better model. Notably, the RF model combined with FCFP descriptors achieved a comparative performance to the GNN models. Figure 3.2.3 shows the ROC plot for four of the best models from cross-validation. Two of them use GNN methods (GCN3 and GAT), and two models use traditional machine learning methods (RF and MLP). The GAT model achieved a predictive accuracy of 0.95 in the ROC AUC in the cross-validation. The ROC curve is shown in Figure 3.2.3 also provides a sight of sensitivity and specificity of the prediction, which further indicates that this model made a reliable prediction for predicting whether a compound can transmit across BBB or not.

Table 3.3 ROC AUC of all machine learning models on cross-validation



Descriptor	LR	SVM	NB	RF	AB	MLP	GCN_1	GCN_2	GCN3	GAT
Atom count	0.9006	0.9069	0.8344	0.9386	0.916	0.9286				
RDKit	0.915	0.8995	0.8636	0.9297	0.9316	0.9042				
MACCS	0.8953	0.8577	0.8363	0.9237	0.9034	0.837				
PubChem	0.9135	0.8743	0.8975	0.926	0.926	0.902				
ECFP6	0.9193	0.939	0.8222	0.9247	0.9005	0.9283				
FCFP6	0.9313	0.8902	0.843	0.9489	0.9384	0.9239				
All (Lasso)	0.9371	0.8879	0.8512	0.9332	0.9387	0.9422				
Mol Graph							0.936	0.9423	0.9424	0.9504

ECFP: Extended Connectivity Fingerprint; FCFP: Functional-Class Fingerprints

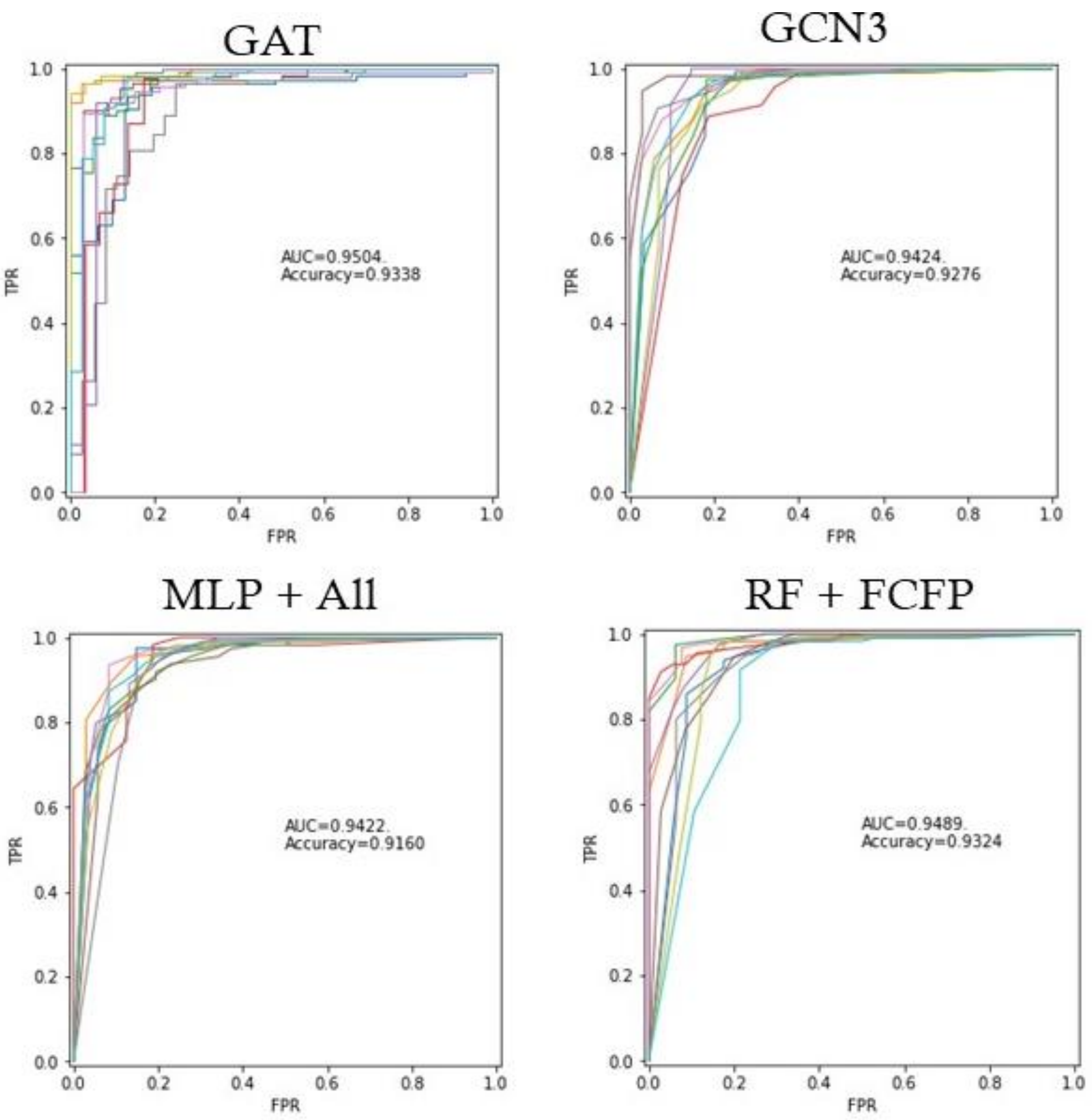


Figure 3.2.3 The ROC plot of the selected best models from cross-validation**.** This figure shows the results from the ROC analysis on the four best ML/DL classification models of BBB. Each figure is a ROC curve from the analysis, with the X-axis assigned as FPR (false positive rate, 1-Specificity) and Y-axis assigned as TPR (true positive rate, Sensitivity, recall)

3.2.2.3 Model Evaluation

Instead of using only the ROCAUC score, a series of metrics were calculated to further explore the performance of each machine learning algorithm on different feature types. Metrics functions assess prediction errors for specific purposes and evaluate the model performance from various aspects. The other metrics involved in this study are F1 score, Accuracy, MCC, precision, and recall. For each algorithm, the best-performed model was selected for the evaluation using the test set.

The results were shown in **Figure 3.2.4**, and the test set performance was relatively consistent with the cross-validation. GNN models made the most accurate prediction on the test set evaluation, while the RF model also performed well. Most of the statistical evaluation metrics aligned with the ROC AUC when comparing different algorithms. Surprisingly, the linear regression model (LR) combined with lasso regularization achieved a relatively good predictive performance over the test set. However, the two simpler GNN models (GCN1 and GCN2) did not perform as well as their performance in the training process.

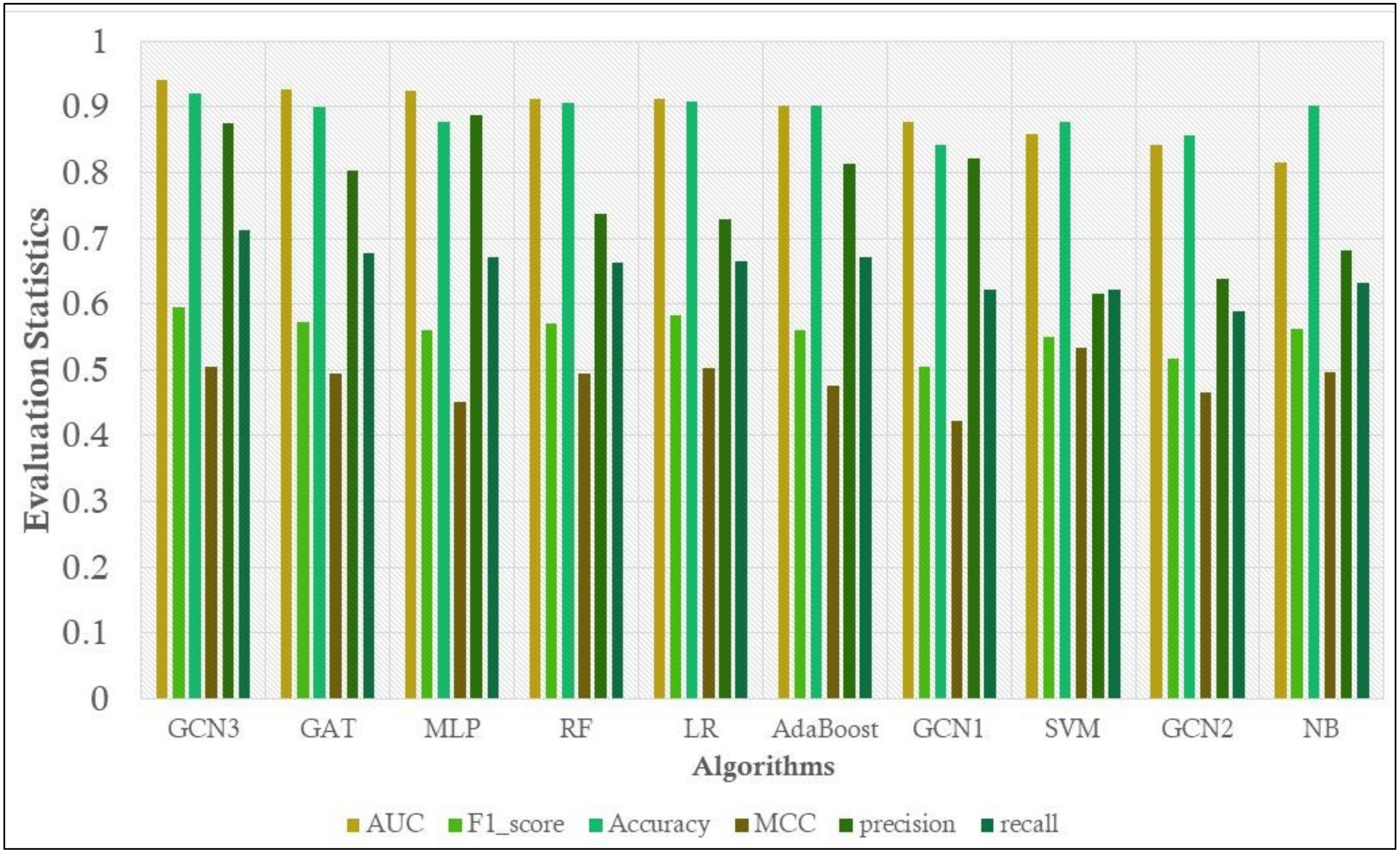


Figure 3.2.4 Predictive performance of the test set over all the machine learning models

3.2.4 Conclusion

In this study, an ML/DL classification framework in predicting the drug permeability of BBB is introduced. Significantly, the effectivity of the deep graphic neural network methods in predicting the drug's BBB penetration was examined and compared with the existing methods. The prediction accuracy of the best mode over the external test datasets already achieved 0.93, and the average AUC is 0.94, the F1 score is 0.60. Furthermore, the accuracy, AUC, and F1 scores of our machine learning methods with both training set and test set are relatively consistent, indicating that our model is stable and not overfitting. In the future, we will further evaluate our model using more experimental data and hope our method can be applied to facilitate CNS drug discovery.

CHAPTER 4. AI/ML PHARMACO-ANALYTICS IN CLINICAL OUTCOMES RESEARCH

4.1 Machine Learning-based Prediction of Substance Use Disorder in Children

4.1.1 Background and Significance

Substance use disorder (SUD) exact enormous societal cost, estimated in the United States to exceed 740 billion dollars annually (NIDA 2020). Considering that first exposure to legal substances (for adults), as well as illegal drugs, usually begins during adolescence, and frequently leads to SUD before thirty years of age (Bose et al. 2018), often co-occurring with psychiatric and medical disorders, physical disability, socioeconomic decline, long-term incarceration and social maladjustment (Organization and Unit 2014), it is important to efficiently detect and implement timely prevention for at-risk youths. Whereas a strong genetic contribution to SUD liability is documented in many population studies (McGue, Elkins, and Iacono 2000), it is not possible to measure the magnitude of SUD risk in the individual. At the phenotypic level, externalizing behaviors such as aggression, impulsivity, and sensation-seeking are well-established childhood antecedents of SUD (Verdejo-Garcia, Lawrence, and Clark 2008a; Iacono et al. 1999). Relatedly, externalizing disturbances qualify for a psychiatric disorder, specifically, attention-deficit/hyperactivity disorder (ADHD) and conduct disorder (CD) also heighten SUD risk. In addition, internalizing disturbances evinced clinically primarily as anxiety and depressive spectrum disorders are also at risk for SUD (King, Iacono, and McGue 2004b). Since externalizing and internalizing characteristics are correlated (Krueger and Markon 2006) and their respective disorders frequently co-occur (Grant et al. 2004), it can be concluded that psychological dysregulation has the cardinal features of encompassing behavior under-control and disturbed modulation of emotions (Tarter et al. 2003) is an integral component of the SUD liability phenotype.

Other behavioral characteristics that are not subsumed within internalizing and externalizing dimensions also impact SUD risk. For example, disinclination for physical exercise and more broadly sedentary lifestyle presages substance use onset (Nelson and Gordon-Larsen 2006). In effect, standard psychological constructs incompletely characterize the liability phenotype which, from the practical standpoint, diminishes the accuracy of detecting high-risk youths. Machine learning (ML) often associated with "big data" potentially enables constructing a computer program to predict SUD. A major advantage of ML methodology is that it is entirely empirical and thus free of investigator assumptions and biases (Wernick et al. 2010). Furthermore, ML is equipped with automatic feature selection functions (i.e., identifying the most salient variables) (Liu and Zhao 2012), using algorithms that use linear as well as nonlinear methods (Kotsiantis, Zaharakis, and Pintelas 2007). These advantages have led to its applications to address diverse medical issues (Chen and Asch 2017; Jing et al. 2018c) as well as detecting SUD peripheral biomarkers (Bough and Pollock 2018) and predicting SUD treatment outcomes (Acion et al. 2017).

To date, ML has seldom been applied to characterize SUD liability owing in part to the paucity of longitudinal studies having multiple datasets spanning multiple assessment waves. The present study employs ML accessed waves of data consisting of approximately 1,000 variables at each timepoint accrued by the NIDA-funded Center for Education and Drug Abuse Research (CEDAR). Previous research conducted on CEDAR's dataset has yielded the transmissible liability index (TLI); that is the psychological characteristics having intergenerational continuity that predispose to SUD (Vanyukov et al. 2003). This prospective study employing ML extends their line of research by identifying the specific characteristics that portend SUD during five developmental tripoints between 10-22 years of age without reference etiology. All the identified characteristics can be ranked according to their strength of association with SUD outcome; hence, the final constellation of items informs SUD risk, and their ranked salience informs intervention targets. Selected items were further used to develop instruments to predict the liability of SUD with a high degree of accuracy using ML algorithm.

4.1.2 Methods

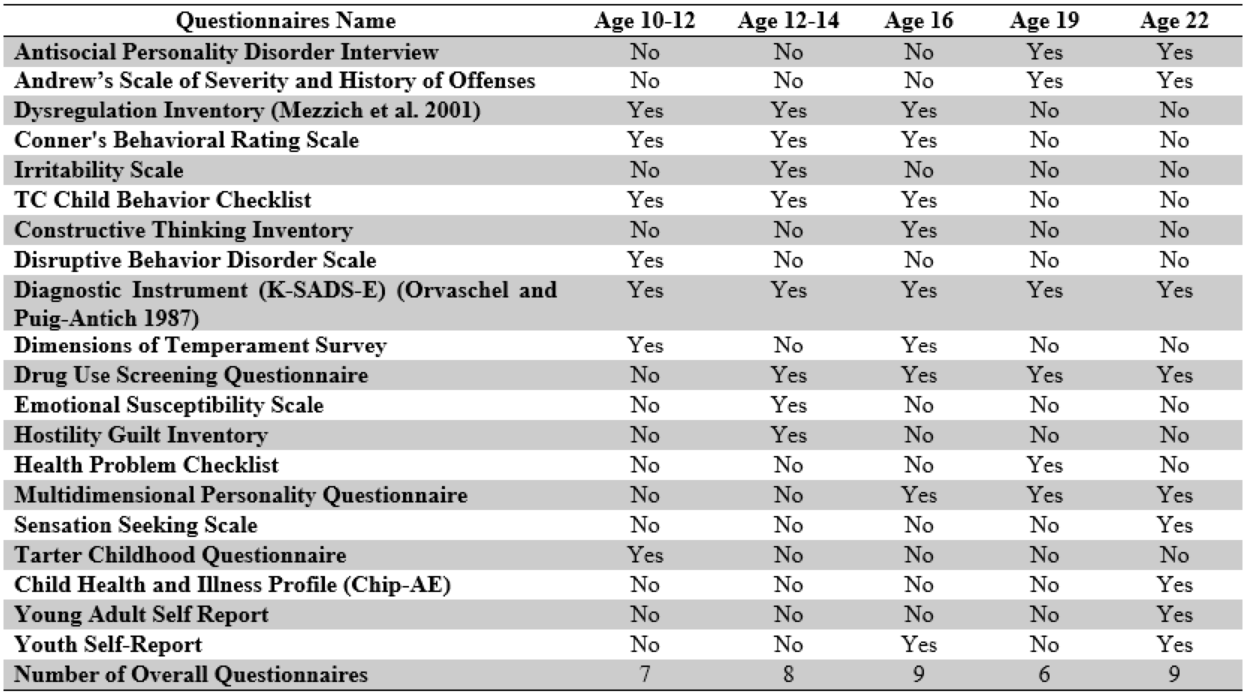
4.1.2.1 Participants

Males (N=494) and females (N=206) between 10-12 years of age (baseline) who had biological fathers (probands) qualifying for a lifetime diagnosis of SUD consequent to use of illicit drugs or no adult psychiatric disorder were re-evaluated at 12-14, 16, 19, 22 years of age with a large number of questionnaires and interviews items measuring psychological and psychiatric characteristics. The men were identified primarily via newspaper and radio advertisements, public service announcements, and random digit telephone calls. Approximately 25% of men with SUD were recruited from addiction. The sample was middle class and consisted of 75.6% European-American, 21.2% African-American, and 3.2% who self-identified having another ethnicity (Vanyukov et al. 2009b). Previous reports have detailed the characteristics of the sample (Vanyukov et al. 2009b). In brief, they scored in the average range of intelligence. None had a history of neurologic injury or disease, physical disability, or chronic medical illness.

4.1.2.2 Measures and variables

Informed consent was obtained from the parents, and children provided written assent before data collection. At 18 years of age and thereafter the participants signed informed consent forms. At the outset, breath alcohol, and urine drug screens were conducted to ensure that the results are not confounded by acute effects of substances. Questionnaires and interviews measured psychopathology, personality, family/social functioning, health, and neurocognition (Table 4.1) (Mezzich et al. 2001; Orvaschel and Puig-Antich 1987). The responses (approx. 1000 at each visit), hereafter termed features, commensurate with ML research, where the data inputs forecast SUD. The outcome variable, termed class label in ML research, is a diagnosis in any SUD category based on DSM-IV criteria (Spitzer et al. 1992). The diagnosis was formulated by a clinical committee based on the results of the Structured Clinical Interview for Diagnosis (SCID) (Spitzer et al. 1992) in conjunction with information obtained in other aspects of the research protocol and medical records.

Table 4.1 Questionnaires summary for different visits



Questionnaires Name	Age 10-12	Age 12-14	Age 16	Age 19	Age 22
Antisocial Personality Disorder Interview	No	No	No	Yes	Yes
Andrew's Scale of Severity and History of Offenses	No	No	No	Yes	Yes
Dysregulation Inventory (Mezzich et al. 2001)	Yes	Yes	Yes	No	No
Conner's Behavioral Rating Scale	Yes	Yes	Yes	No	No
Irritability Scale	No	Yes	No	No	No
TC Child Behavior Checklist	Yes	Yes	Yes	No	No
Constructive Thinking Inventory	No	No	Yes	No	No
Disruptive Behavior Disorder Scale	Yes	No	No	No	No
Diagnostic Instrument (K-SADS-E) (Orvaschel and Puig-Antich 1987)	Yes	Yes	Yes	Yes	Yes
Dimensions of Temperament Survey	Yes	No	Yes	No	No
Drug Use Screening Questionnaire	No	Yes	Yes	Yes	Yes
Emotional Susceptibility Scale	No	Yes	No	No	No
Hostility Guilt Inventory	No	Yes	No	No	No
Health Problem Checklist	No	No	No	Yes	No
Multidimensional Personality Questionnaire	No	No	Yes	Yes	Yes
Sensation Seeking Scale	No	No	No	No	Yes
Tarter Childhood Questionnaire	Yes	No	No	No	No
Child Health and Illness Profile (Chip-AE)	No	No	No	No	Yes
Young Adult Self Report	No	No	No	No	Yes
Youth Self-Report	No	No	Yes	No	Yes
Number of Overall Questionnaires	7	8	9	6	9

4.1.2.3 Data pre-processing and missing data imputation

At the outset, a feature (i.e., item) was eliminated if 1) the percentage of missing responses was 70% or more, 2) its response had a variance of <0.1, 3) the item directly queried substance use, or 4) the item was answered by an informant (e.g., teacher) other than a parent or their child.

Imputation of missing data was performed using the k-nearest-neighbors algorithm. This algorithm assumes that missing data can be substituted with values informed by the closest cases (neighbors) from the entire sample (Beretta and Santaniello 2016). First, all the variables were normalized using the conjunction of three neighbors (k = 3) based on examination of the data reflecting the most appropriate balance between imputation error and preservation of the data structure true (Beretta and Santaniello 2016). Next, the proximity between features was calculated according to the equation:

$$S_{ij} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}\left[w_k(v_{ik} - v_{jk})\right]^2}\ ,$$

where *n* is the number of features without missing data for Subjects *i* and *j*, w_k is the weight of feature *k*, v_{ik}, and v_{jk} are the normalized values of feature *k*. The following two criteria must be satisfied during the difference score calculations: (1) *n* must be no smaller than 40% of total features, and (2) a feature is disqualified if the missing data is larger than 30%. If the *k-th* feature of subject *i*, v_{ik}, is missing, three subjects whose profiles are most similar to the subject *i* are first identified, that is, their difference score S_{ij} is the smallest. Lastly, the mean of the three v_{jk} values is assigned to v_{ik}.

4.1.2.4 Feature selection

Feature selection is the ML technique of removing irrelevant and redundant features and selecting a subset of features for developing a good parsimonious prediction model. In addition, feature selection could also simplify the data description and improve the comprehensibility of the models (Liu and Zhao 2012). There are three general types of feature selection methods: “filter”, “wrapper”, and “embedding” (Guyon and Elisseeff 2003). The filter methods select features based on their feature importance score calculated independently from any model developing algorithms (e.g., $\chi2$ values). These methods run very fast, but at the cost of inferior results. Wrapper methods search among different feature subsets iteratively to find the one that maximizes the predictive accuracy of the model. Those methods usually provide much better results, but they are very slow and more computational demanding (Bouaguel 2016). Embedded methods are built-in methods in the algorithms, which simultaneously perform feature selection and model training, and they often provide a good balance between performance and computational cost (Saeys, Abeel, and Van de Peer 2008). The random forest-based feature selection is one of the most commonly used embedded methods (Genuer, Poggi, and Tuleau-Malot 2010). It can provide multivariate feature importance scores which are relatively cheap to obtain, and it has been successfully applied to diverse types of high dimensional data (Genuer, Poggi, and Tuleau-Malot 2010). In this study, this method was utilized to select the best features. The information gain (Shannon 1948) of each item calculated by RF was used as the importance score for predicting SUD. The items were ranked based on their importance scores and sequentially entered into the predictive model until reaching the maximum (ROC AUC) (Hanley and McNeil 1982). Pearson’s $\chi2$ test assessed the relationship between each feature and the outcome class (SUD+/-).

4.1.2.5 Model construction using machine learning algorithms

To keep consistent with the feature selection method, RF algorithm (Ho 1998) was primary used to construct prediction models. RF is a commonly used tree-based ensemble learning algorithm used for a variety of tasks, including classification and regression (Breiman 2001b). It consists of a multitude of decision tree models, each of which is trained using a subset of the sample and features are randomly drawn from the original dataset, and the final prediction is based on the majority voting from all decision tree models (Ho 1998). Each decision tree is a recursive partitioning model (Magerman 1995), in which the entire dataset is divided into smaller subsets recursively based on one feature value at each split until all the sample in the subsets has the same class label. In the current work, the RF models were optimized using the out-of-bag (OOB) estimates, which is the estimation using the left-out samples after randomly subsampling the entire dataset (Breiman 2001b).

In addition to the RF algorithm, other ML algorithms were briefly tested and compared, which include logistic regression, adaptive boosting (AdaBoost) [ref]. naïve Bayes(Domingos and Pazzani 1997), support vector machine (SVM) (Steinwart and Christmann 2008), k nearest neighbor (kNN)(Stewart and Willett 1987), and deep neural network (DNN)(Lippmann 1989). AdaBoost is an ensemble tree-based algorithm that conjugates multiple decision tree models, and the final prediction is the weighted sum up of the output from those weak classifiers (Ma, Wang, and Xie 2011b). The Naïve Bayes algorithm (Domingos and Pazzani 1997) calculates the posterior probability of each class at the condition of given features, and the outcome class with the highest probability is the predictive outcome. The strategy of SVM is to find a decision boundary (hyperplane) that maximizes the geometric margin between the two classes in the feature space(Steinwart and Christmann 2008). This hyperplane can be either linear or nonlinear, depending on kernel methods(Elisseeff and Weston 2002). The principle behind KNN, introduced in the missing data section, is to predict a new case using a pre-defined number of samples that are similar to it (Stewart and Willett 1987). DNNs are network-based methods that are composed of one input layer, one output layer, and several hidden layers in between(Lippmann 1989). Notably, DNNs are considered to enable deep learning that shows more tolerance to multiple levels, nonlinearity, and complexity of big data (Cabitza, Rasoini, and Gensini 2017). Sci-kit-learn python package(Pedregosa et al. 2011b) was used to develop models for these ML algorithms. The selected features were compared with the entire set using each ML algorithm.

4.1.2.6 Validation of predictive models

We performed 10-fold cross-validation to evaluate the forecasting accuracy of the six models (Shao 1993). The dataset was randomly divided into 10 approximately equally sized subsets. Nine subsets were combined to form the training set with the remaining subset was used to evaluate each model. This process was repeated ten times accompanied by ROC analysis to determine each model’s sensitivity, specificity, and overall classification accuracy.

4.1.3 Results

4.1.3.1 Selected features for predicting SUD individuals

To determine the optimal number of features needed for building robust models at each age point, we tracked the change in prediction accuracy (ROC AUC) of the random forest model constructed with an increasing number of features. Features were sequentially added into the model based on their importance ranking. As illustrated in **Figure 4.1.1**, accuracy at every visit for predicting SUD reached peak values when the size of the feature (items) set was approximately 30. Thus, the top 30 features (i.e., prediction of SUD outcome) were selected to generate the final models to predict SUD. The table in Appendix A (a) lists the features at 10-12 years of age. Almost half (N=14) were ratings or responses provided by a parent. In the subsequent visits (Appendix A), all of the best features were provided by the children. This finding concurs with the observation that young children are not the most accurate informants about themselves. Overall, the best features at 10-12 years of age are indicators of psychological self-regulation spanning behavior control, emotion modulation, daily routine, and mental concentration in conjunction with social interaction problems at later ages, although different indicators of suboptimal psychological self-regulation with additional indicators of social maladjustment, particularly decrease proneness. Considered from the ontogenetic perspective, the features most prognostic of SUD thus advance from psychological dysregulation during childhood to the persistence of this disposition accompanied by marked non-normative socialization in adolescence and adulthood.

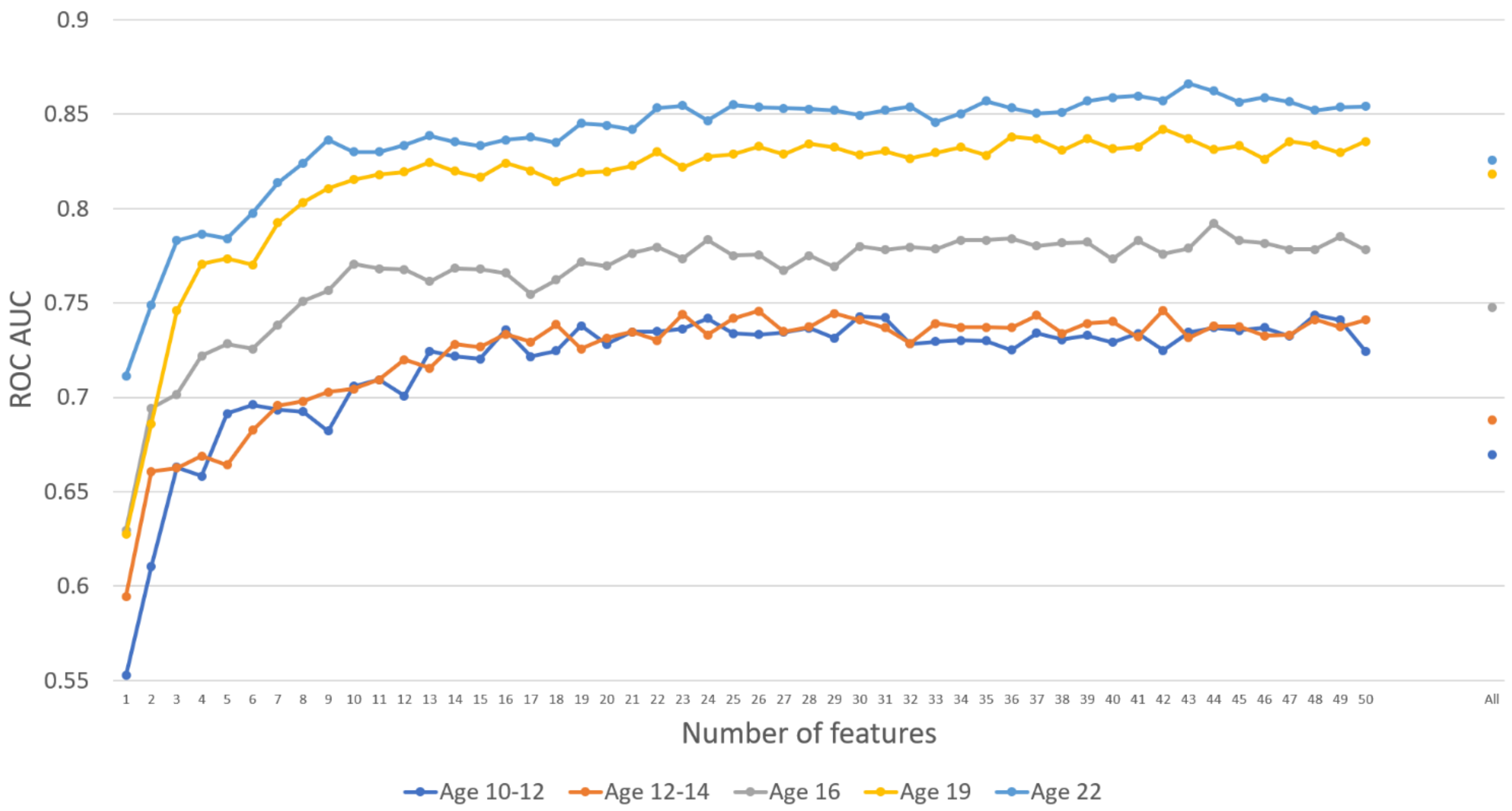


Figure 4.1.1 The relationship between the number of features and the predictive power of the model using RF algorithms for all visits. The predictive power was scaled using ROC AUC in the 10-fold cross-validation

4.1.3.2 Model performance, selection, and validation

The accuracy of RF models at different ages for predicting SUD is depicted in **Figure 4.1.3**. At 10-12 and 12-14 years of age, the ROC AUC is 0.74. With increasing age, the forecasting improves to excellence. The ROC AUC is respectively 0.78, 0.83, and 0.86 at ages 16, 19, and 22. Notably, RF and Naïve Bayes models are superior to the other models (**Figure 4.1.2**). However, regardless of the particular model, the accuracy of forecasting SUD increases with chronological age. It is also noteworthy that the standard deviation of the ROC AUC in the 10-fold cross-validation is somewhat high in all six models indicating that the algorithms are interchangeable concerning this measure of model quality. Overall, the models consisting of thirty features (orange bars in **Figure 4.1.2**) are superior to the models using the entire dataset (blue bars in **Figure 4.1.2**).

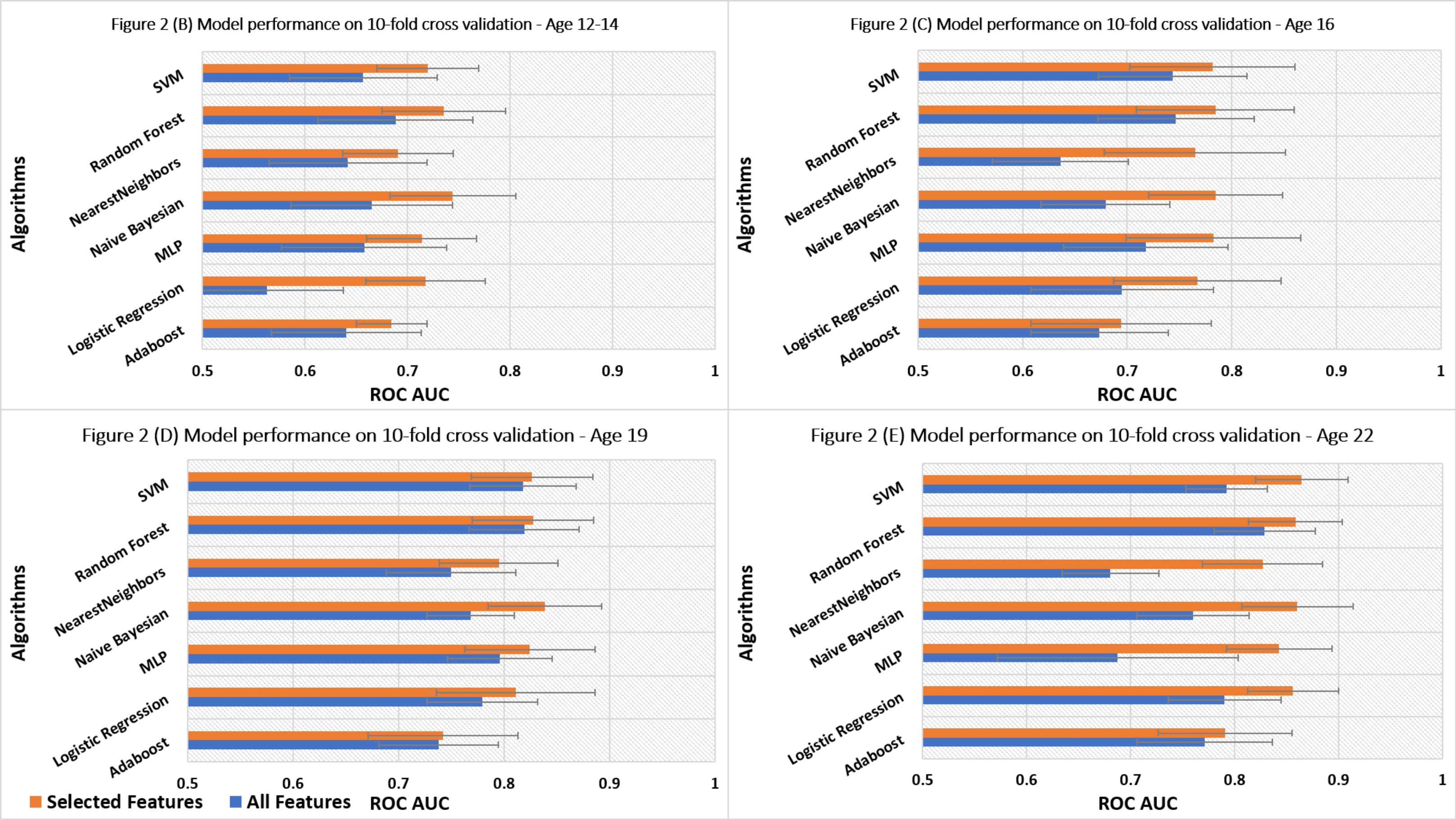


Figure 2 (B) Model performance on 10-fold cross validation - Age 12-14

Figure 2 (C) Model performance on 10-fold cross validation - Age 16

Figure 2 (D) Model performance on 10-fold cross validation - Age 19

Figure 2 (E) Model performance on 10-fold cross validation - Age 22

Figure 4.1.2 Performance of different prediction algorithms at different ages

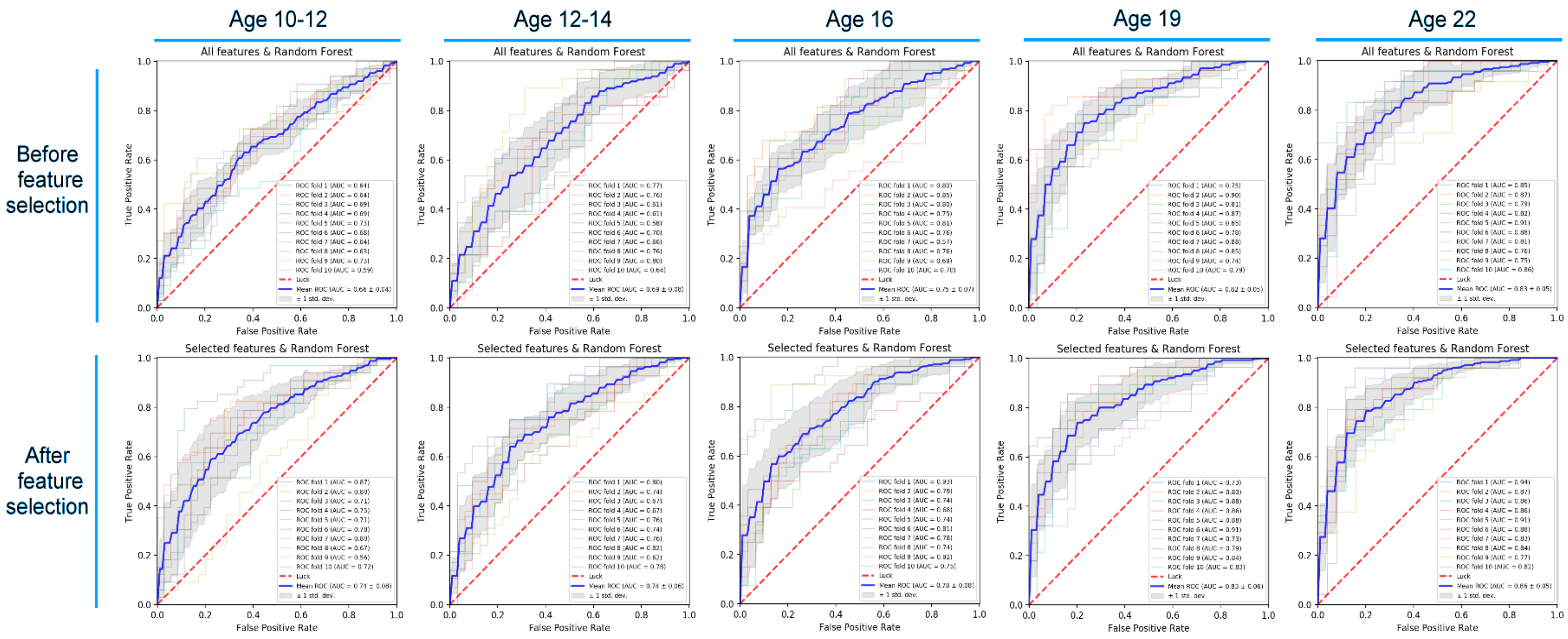


Figure 4.1.3 Random Forest prediction before and after feature selection. The ROC AUC for all ten RF models were shown in Figure 4.2. Top figures were models generated using all the features prepared before feature selection. The bottom figures were the performances of models using selected features. Each column was assigned for a visit. In each chart, the blue line shows the average ROC curve in the 10-fold cross-validation and the gray areas shew the standard deviation. ROC curves in other colors shew the detailed performances of the models in the cross-validation

4.1.4 Discussion and Conclusion

To briefly recapitulate, the results of this prospective study demonstrate that models constructed using machine learning (ML) methodology predict SUD with increasing accuracy across five-time points spanning from 10-12 to 22 years of age. Accuracy of forecasting SUD points to the feasibility of using the derived algorithms in routine screening of youths for timely determination of the need for intervention. Considering that the thirty items require about five minutes to administer and score on the Web platform, screening can be conducted expeditiously, inexpensively, and unobtrusively since none of the prognostic features query substance use. Overall, these results demonstrate the feasibility of detecting high-risk youths. However, in contrast to prior research focusing on a particular etiological component such as intergenerational transmissibility (Vanyukov et al. 2003), psychological orientation (e.g. externalizing behavior) (King, Iacono, and McGue 2004b), or normative socialization (Whiteman, Becerra, and Killoren 2009), ML models revealed additional important characteristics associated with SUD risk. For example, the 3rd, 12th, 18th, and 24th best predictor variables consist of daily routines, particularly eating and sleeping. These latter findings raise the prospect that interventions targeting these aspects of SUD liability may increase the likelihood of non-SUD outcomes.

Whereas psychological dysregulation such as symptoms of ADHD and CD SUD forecast SUD (King, Iacono, and McGue 2004b; Krueger et al. 2005), this study also revealed other highly salient SUD risk features that are readily observed and amenable to intervention. Notably, the use of foul language is the most prominent feature. Poor play behavior ranks second and irritability ranks fourth. Additionally, it is observed that the array of predictors is not confined to individual characteristics, but also includes environmental factors (neighborhood and school).

These factors are well-known to impact SUD risks, especially in the later visits at ages of 16, 19, and 22. Future research directed at clarifying the SUD risk-promoting environ type (Kirisci et al. 2009) using ML methodology may, therefore, enhance forecasting accuracy as part of comprehensive prevention targets.

The best prognostic features are transdiagnostic. Since these features were identified independently of assumptions or biases of the investigator, their importance resides in their salience for prioritization of intervention tactics. It is also noteworthy that the portending SUD features may be detectable before ten years of age, the baseline in this study. Longitudinal studies show that temperament disturbance (Horner et al. 2015) before five years of age indicates disrupted psychological-self-regulation and daily rhythmicity forecasts SUD two decades later. These findings point to the feasibility of risk detection and intervention in early childhood.

Several limitations in this study are noted. First, because the high-risk paradigm was used, oversampling children prone to SUD prevalence in the general population may have biased the results. Accordingly, the results of this study require replication employing random sampling of youths. Second, it is noteworthy that the standard deviations in the 10-fold cross-validation are large, indicating that while models are adequate, they need improvement. Furthermore, although the performance of the models ranged from satisfactory to excellent with increasing age, improvement may be achieved by expanding the spectrum of variables available for feature selection and removing redundant items using such methods as a lasso, ridge regressions, and genetic algorithms [51]. Notwithstanding these limitations, this study demonstrates the heuristic value of ML methodology to derive a cost-efficient scalable screening tool to identify youths at high risk for SUD.

In conclusion, ML techniques identified the characteristics between late childhood and adulthood that forecast SUD can be predicted by these characteristics with sufficient accuracy to justify application in routine screening to detect high-risk individuals. Moreover, ranking the components of risk according to their contribution to SUD development informs prioritizing intervention targets. Hence, reviewing an individual's protocol quickly provides insight into the resources needed to ameliorate the particular risk factor to lower the risk for SUD.

4.2 CausalSUD: a Causality-based Method for Predicting SUD in Childhood using Causal Machine Learning

4.2.1 Background and Significance

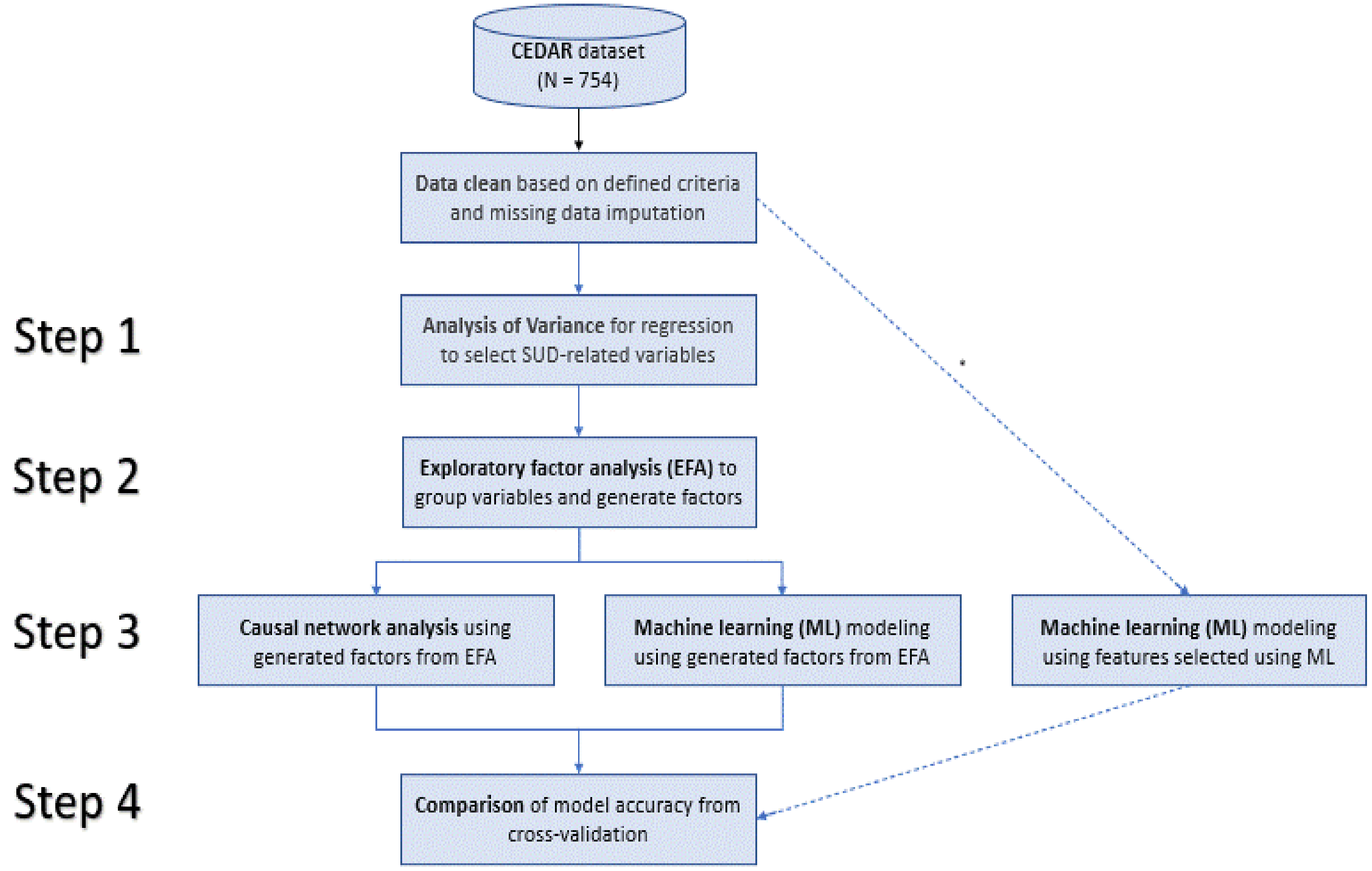
Consumption of substances having dependence liability exacts an enormous cost to the U.S. economy, with estimates ranging up to $740 billion annually (NIDA 2020). Chronic medical disease, social decline, incarceration, mental health disorders, and violence victimization consequent to habitual use of abusable substances contributing to this fiscal burden are especially likely to occur among early age onset consumers who develop substance use disorder (SUD). Notably, the prevalence of consumption has remained high and stable in the youth population for over two decades (Administration 2020). Considering that substance use onset at a young age heightens risk (Lipari et al. 2017), it is important to accurately and efficiently develop tools that detect youths with a high probability of advancing to SUD (Conrod 2016).

Toward this goal, we have published machine learning methods to identify the characteristics associated with elevated risk for SUD (Hu et al. 2020; Jing et al. 2020). Whereas measures have been shown to predict SUD based on theoretical suppositions regarding etiology and its intergenerational transmissibility (Schulenberg et al. 2014; King, Iacono, and McGue 2004a; Verdejo-Garcia, Lawrence, and Clark 2008b; Vanyukov et al. 2009a), machine learning procedures are theoretical and have the advantage of taking into account in the forecasting the correlations among all the predictors (Jordan and Mitchell 2015). In the study, we have shown that thirty characteristics, largely reflecting psychological disposition in 10–12-year-old children, forecasted SUD before 30 years of age with 74% overall accuracy (Jing et al. 2020; Hu et al. 2020).

Even though traditional machine learning methods, such as SVM and RF, can provide accurate predictions to support decision-making tasks, they have several weaknesses. Firstly, those methods may only specify association rather than causality between variables. Secondly, those methods may not be appropriate to identify the potential latent confounders from numerous variables. Thirdly, models developed using those methods are inherently less interpretable and run as a “black box”. Those weaknesses of machine learning limit its further development on offering physicians causal-based suggestions that accurately inform a personalized approach to recommend therapy for individual patients. (Rudin 2019).

Within the family of machine learning methodologies, causal Bayesian network methods is especially promising for identifying the variables involved in the etiology of psychopathology (Heckerman, Geiger, and Chickering 1995; McNally 2016; Conrod 2016). In other words, this method has the potential to elucidate the psychological and behavioral variables that have a causal effect on the etiology of SUD (Kramer et al. 2014; Rhemtulla et al. 2016). Depicted in a graphic display, the derived causal network model is featured by nodes (representing the random variable) and their connections (representing potential causal influence) (Glymour, Zhang, and Spirtes 2019a). Moreover, causal network theory aligns with the conceptualization of SUD by focusing on the dynamic relationships among the symptoms traceable to their causal effect (Borsboom 2017; Jones, Heeren, and McNally 2017). This stands in contrast to the current conceptualization of psychiatric disorders, including SUD, portrayed as syndromes having latent (i.e., biological) causation (Jang et al. 2020). Additionally, the combination of machine learning and causal network analysis may help to achieve a balance between the predictability and the interpretability of the model and is becoming widely used in clinical outcomes research (Richens, Lee, and Johri 2020; Nogueira, Gama, and Ferreira 2020).

This prospective study examined the heuristic utility of causal network analysis to 1) delineate the causal relationships among psychological and psychopathological characteristics contributing to the etiology of SUD; 2) build graphic models based on psychological and psychopathological characteristics to forecast SUD during adolescence and young adulthood; and 3) elucidate its forecasting accuracy compared to other commonly used ML methods. **Figure 4.2.1** illustrates the progress of the steps in this study.


CEDAR dataset
(N = 754)
Data clean based on defined criteria and missing data imputation
Analysis of Variance for regression to select SUD-related variables
Exploratory factor analysis (EFA) to group variables and generate factors
Causal network analysis using generated factors from EFA
Machine learning (ML) modeling using generated factors from EFA
Machine learning (ML) modeling using features selected using ML
Comparison of model accuracy from cross-validation
Step 1
Step 2
Step 3
Step 4
*

*This section is accomplished in our published paper *Analysis of substance use and its outcomes by machine learning* [8]

Figure 4.2.1 The overall research procedure of the causal machine learning analysis

4.2.2 Methods

4.2.2.1 Data preparation and pre-processing

The participants, measures, and variables are the same as in chapter 4.1.2.1. The data pre-processing and missing data imputation process are the same with chapter 4.1.2.2. Details can be found in section 4.1.

4.2.2.2 Analysis of variance with the regression model

In Step 1 of the study, analysis of variance (ANOVA) was performed to estimate the correlation between each variable and SUD label and eliminate the less correlated variables (Glantz and Slinker 2001). Specifically, a logistic regression model was constructed between each variable and SUD label. The p-value from the significance test was applied to interpret the correlation. In addition, to minimize type I error consequent to multiple hypothesis testing, the p-values from the original analysis were corrected using the false discovery rate (FDR) approach (Benjamini and Hochberg 1995)**.** This entire analysis was performed using the *statsmodels* package in Python (Seabold and Perktold 2010).

4.2.2.3 Exploratory factor analysis

In Step 2, an exploratory factor analysis (EFA) was conducted to reveal the constructs underlying the questionnaire items that depict the behavioral and psychopathological characteristics. The goals of this analysis are to 1) better understand the internal structure of the measured variables; 2) discover patterns of relations (common factors) among the variables; and 3) reduce the dimensionality of the data and prepare optimized data for the subsequent analysis.

EFA is a statistical method widely used to identify factors or aggregates of variables within a set of measured variables. The factors also describe the magnitude of the correlations of the measured variables within each factor (Norris and Lecavalier 2010). In sum, EFA is used in psychological research to identify latent constructs (Fabrigar et al. 1999).

In this exploratory factor analysis, the correlation matrix was used to undertake the analysis. To determine how many factors should be included, the Kaiser criterion was adopted by eliminating factors with eigenvalues less than 1 (Yeomans and Golder 1982). Maximum likelihood estimation was employed to fit the factor analysis model. To increase interpretability while allowing them to correlate, the factors were rotated using the *Promax* (oblique) rotation method after the model-fitting procedure. Any variable with loading less than 0.4 was ignored and removed from the factor. The remainder reflects a common theme informing the factor label. The entire analysis was performed using the *factor_analyzer* package in Python (Biggs 2019).

4.2.2.4 Causal network analysis

Step 3 of this study involved a comprehensive causal analysis for each assessment wave (10-12, 12-14, 16, 19, and 22 years of age). The results depict the causal relationship, if any, between psychopathological characteristics with SUD outcomes at specific ages. This analysis was performed using two procedures. The first procedure involved performing an extended Markov blanket search to adaptively identify the causal-related factors of SUD as well as the parent nodes of those factors. The second procedure involved doing causal inference based on the causal order learned and discovering an optimized causal network between causal factors and SUD.

Markov blanket search used in the first step is a standard method to learn the Bayesian network and feature selection (Aliferis et al. 2010). Under the faithfulness assumption, the Markov blanket of a node (T) in the network graph is the minimal set of nodes conditioned on which all other nodes are independent of T (Fu and Desmarais 2010). In this step, both the PC algorithms (Spirtes, Glymour, Scheines, and Heckerman 2000) and the Fast Greedy Equivalence Search (FGES) algorithms (Ramsey et al. 2017) were adopted to catch as many causal factors as possible. The PC algorithm is a type of constraint-based algorithm that involves a set of statistical tests of conditional independence. The rule of the PC algorithm is restricted to just the variables in the Markov blanket of the target node (T) and its output is the graph that is a pattern over the variables. The FGES is a score-based search algorithm that heuristically performs searches over a large number of different network models by iteratively adding or removing edges in the network graph, and finally returns the graph model with the highest score, such as the Bayesian information criterion (BIC) score. Such a method is fast and accurate; however, this method cannot deal with unobserved confounders because of the Causal Markova and Faithfulness Assumptions.

The second procedure of this causal analysis consisted of performing a causal inference to search the optimal graphic causal network structure using the causal factors learned in the Markov blanket of SUD. To ensure the network extends to follow the causal series, tiers of nodes (variables) in the causal networks were pre-defined in each analysis, with one tier representing one certain age group. This step was achieved using Greedy Fast Causal Inference (GFCI) algorithm (Ogarrio, Spirtes, and Ramsey 2016), a hybrid causal inference algorithm that combines the FGES algorithm and Fast Causal Inference (FCI) algorithm (Spirtes 2001; Glymour, Zhang, and Spirtes 2019a). The FCI algorithm is a variation of the PC algorithms but tolerates and sometimes discovers unknown confounding variables. Causal network structures which were compatible with the result of the statistical tests were selected out and the common features of those structures are extracted to generate the graphical object called a Partial Ancestral Graph (PAG). In this case, FCI could tolerate and sometimes discover unobserved confounding variables. However, one of the major disadvantages of the FCI algorithms is that the number of statistical tests in FCI will exponentially increase with the number of variables included in the data set. Therefore, the accuracy, computing speed, and reliability of such methods are always criticized when using FCI. GFCI takes advantage of those two algorithms by separating the causal analysis into two steps. First, GFCI uses the FGES algorithm to perform a quick search over the model space and generate a more accurate initial graph. GFCI uses the FCI algorithms by performing a set of conditional independence tests to refine the preliminary search results and find the orientations for the edges in the graph networks.

This causal analysis was conducted using the Tetrad software package, version 6.7.1 (Ramsey et al. 2018; 'Tetrad Manual' 2019). The Fisher Z test was employed in the statistical test to judge the conditional independence if the conditional correlation is zero, with the p-value threshold parameter (alpha) set to 0.05. Tetrad-specific Bayesian information criterion (BIC) score was adopted in the scoring metrics to evaluate the networks generated from the search procedure, with the additional penalty parameter (penalty discount) set to be the standard BIC value of 1. To ensure the stability of the causal network, we performed the subset bootstrap procedure. Each time a random subset of the input data set was resampled, a causal graph network was computed for that sample. Then the edge and edge type probabilities in the graph, which mean how often the edge and edge type appeared in the 100 bootstrapped networks, were calculated. The final output graph will be the ensemble of the voting results from all the graphs. Specifically, the ensemble method adopted here is the ‘highest’ option, which only returns edge orientation that the highest percentage of sample graphs returns.

4.2.2.5 Machine learning for SUD

In Step 4 of this study, we tried to predict SUD by integrating machine learning methods causal factors obtained from the causal analysis. The predictive accuracy from causal-ML models was evaluated and compared with traditional machine learning-based models. In addition, the results from our previous published study using machine learning methods were also added to the comparison (Jing et al. 2020). Specific for machine learning (Mohri, Rostamizadeh, and Talwalkar 2018), six commonly used algorithms were adopted including 1) logistic regression (Kleinbaum et al. 2002), 2) random forests (Breiman 2001b), 3) adaptive boosting (AdaBoost) (Solomatine and Shrestha 2004), 4) naïve Bayes (Rish 2001), 5) support vector machine (SVM) (Steinwart and Christmann 2008), and 6) deep neural network (DNN) (LeCun, Bengio, and Hinton 2015). The dataset was split into a training set and a test set with a ratio of 10:1. During the training process, the cross-validation (CV) using 10 folds (Browne 2000) was performed for validating the models using the area under the receiver operating characteristic curve (ROC AUC) (Hajian-Tilaki 2013). All the models were developed utilizing the *Scikit-learn* Python package (Pedregosa et al. 2011a).

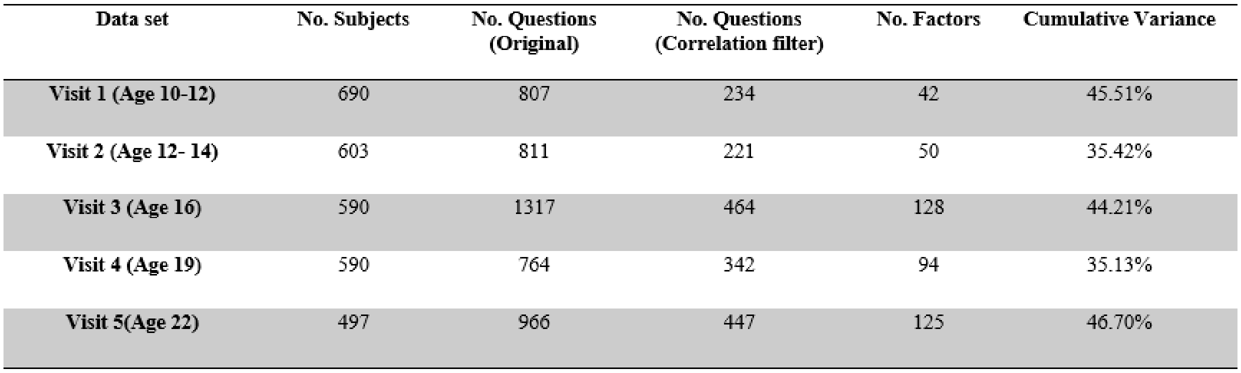
4.2.3 Results

4.2.3.1 Exploratory factor analysis and exploratory factor analysis

Variables (question items) uncorrelated with SUD were eliminated after the variance analysis. The number of selected variables for all age groups is listed in Table 4.2. More than half of the questions (items) were filtered out from the original data set due to lacking correlation with SUD.

The number of factors and associated information for all age groups of ages is shown in Table 4.2. As can be seen, approximately a third to half of the total cumulative variance is explained by each factor. The final number of remained factors was determined using the Kaiser criterion (Yeomans and Golder 1982). This helped with keeping more information (variance) for the next step of causal network analysis and providing a more conceptually coherent and meaningful solution. As introduced in the methods section, factor loadings less than 0.4 were reset to be zero. The factor loadings can be interpreted as correlation coefficients, and regardless of sign (positive or negative), the absolute value was considered using this threshold.

Table 4.2 Data structure of substance use disorder (SUD) data



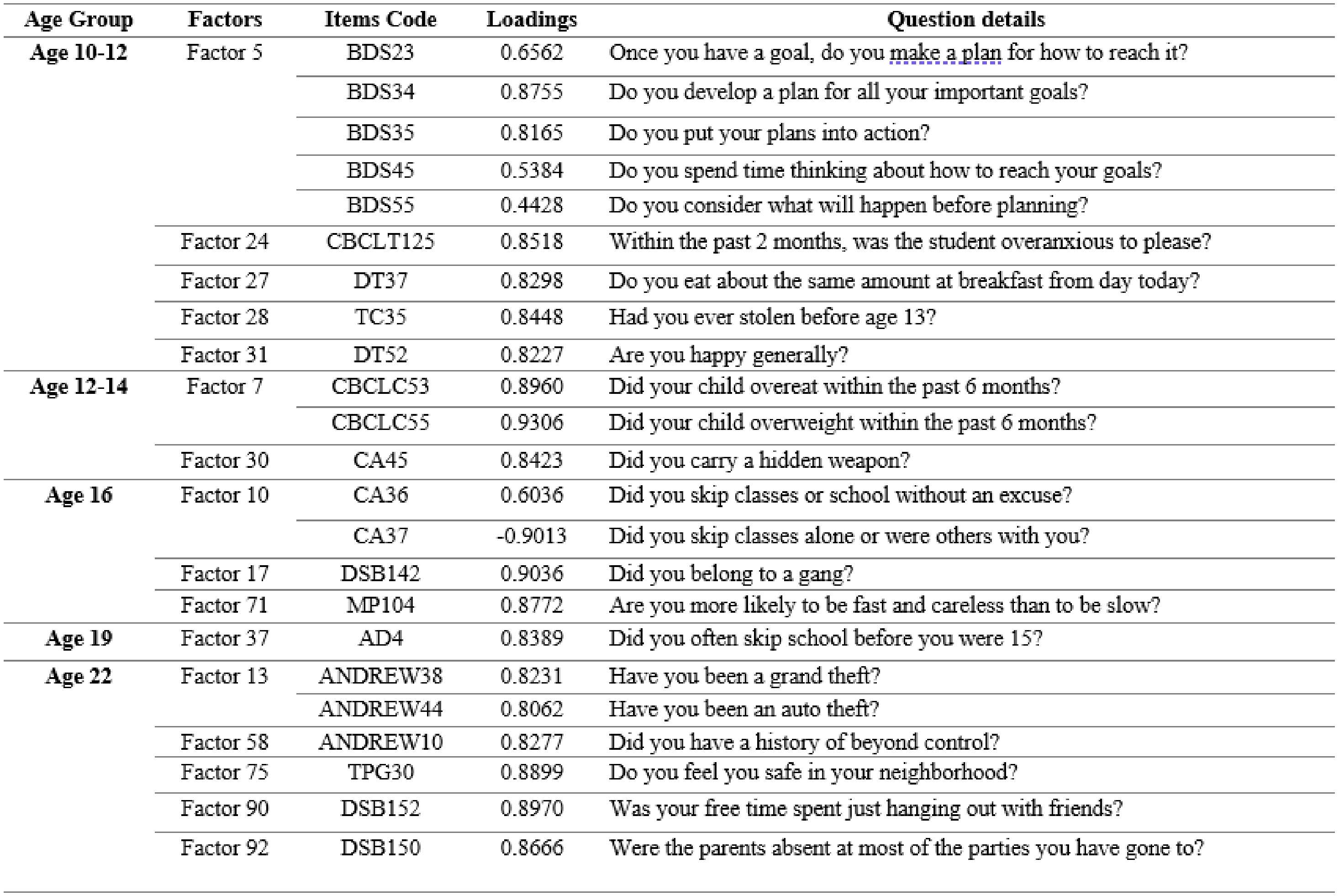
Data set	No. Subjects	No. Questions (Original)	No. Questions (Correlation filter)	No. Factors	Cumulative Variance
Visit 1 (Age 10-12)	690	807	234	42	45.51%
Visit 2 (Age 12- 14)	603	811	221	50	35.42%
Visit 3 (Age 16)	590	1317	464	128	44.21%
Visit 4 (Age 19)	590	764	342	94	35.13%
Visit 5(Age 22)	497	966	447	125	46.70%

4.2.3.2 Causal Bayesian networks analysis for each age group

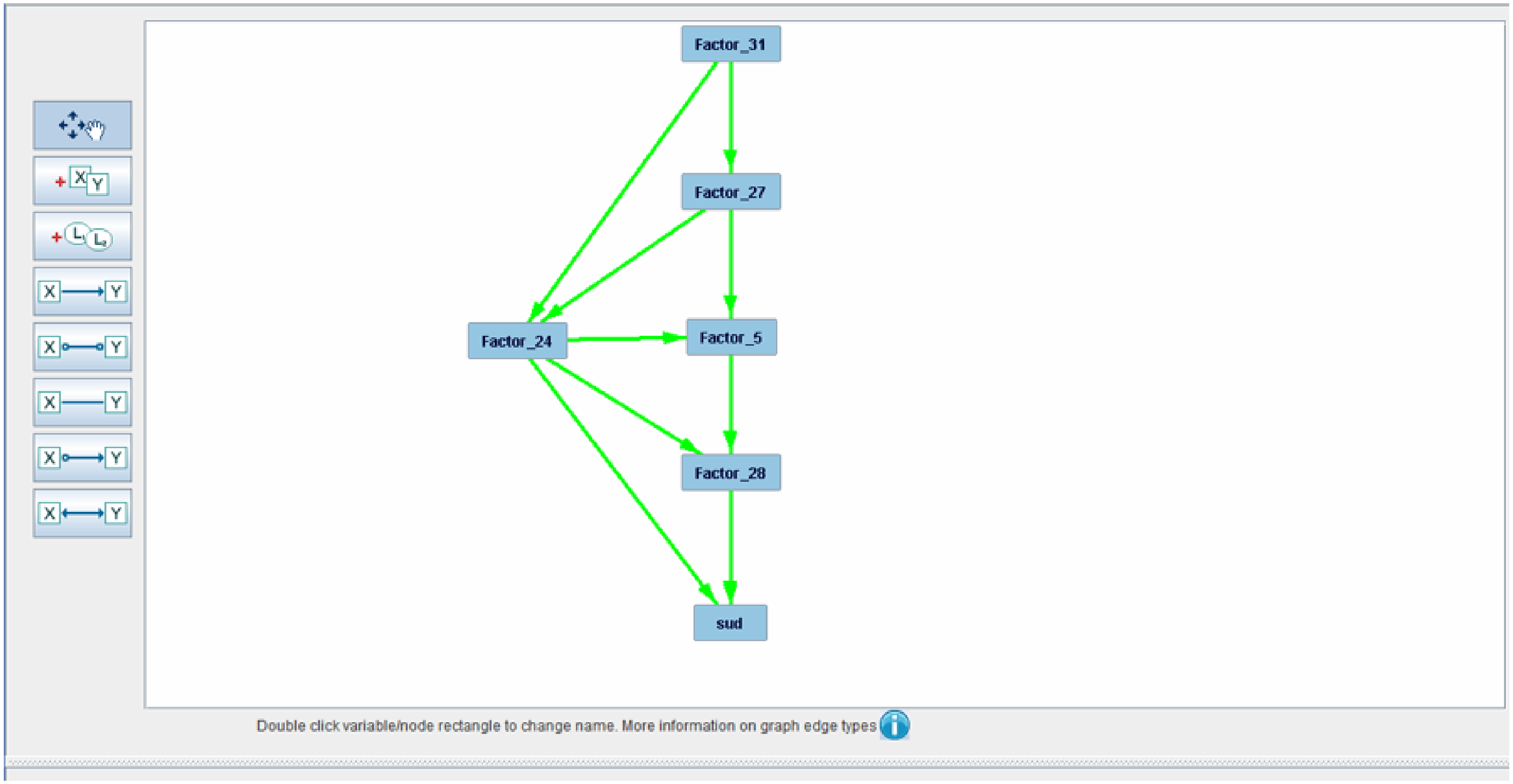
To examine the causation between the psychopathological factors and SUD, we conducted causal analysis for each assessment wave. As mentioned in the methods section 2.6, we first performed the Markov blanket search to find the causal factors of SUD for each assessment wave separately. The results showed a limited number of causal factors contributing to the SUD-associated networks, with only a few factors having a direct causal relationship with SUD (**Table 4.3**). **Figures 4.2.2 (A-E)** depicted the PAGs generated using the GFCI algorithm for all five age groups. The edge thickness signifies edge type probabilities, and the edge direction indicates causal relationships.

For children at age 10-12, five factors directly causally connect with SUD (**Figure 4.2.2A**), which include factor5, factor24, factor27, factor28, factor31. The top factor (factor 31) investigated how happy the child was, suggesting that the happiness might impact the mental health of the child, leading to psychological problems such as anxiety (factor 24) and behavior such as stealing (factor 28). For children at age 12-14, two factors were directly causing SUD. Factor 7 investigates children's capability of self-control, which may further impact their dangerous behaviors such as keeping weapons (factor 30). For children at age 16 and 19, school attendance was a very important causal factor for SUD, and especially for the age 19 model, the only causal factor (factor 37) asked the child's school attendance before age 15, which was the same with factor 10 for age 16 model. The causal factors for the age 22 model correlated with anti-social behavior (factor 13, factor 58) and the surrounding environment of the child (factor 75, factor 90, factor 92). And the top factor (factor 92) in the causal network (**Figure 4.2.2E**) suggested that parents and families could play an extremely important role in preventing the development of SUD for their children.

Table 4.3 Causal factors relevant to SUD at each age group

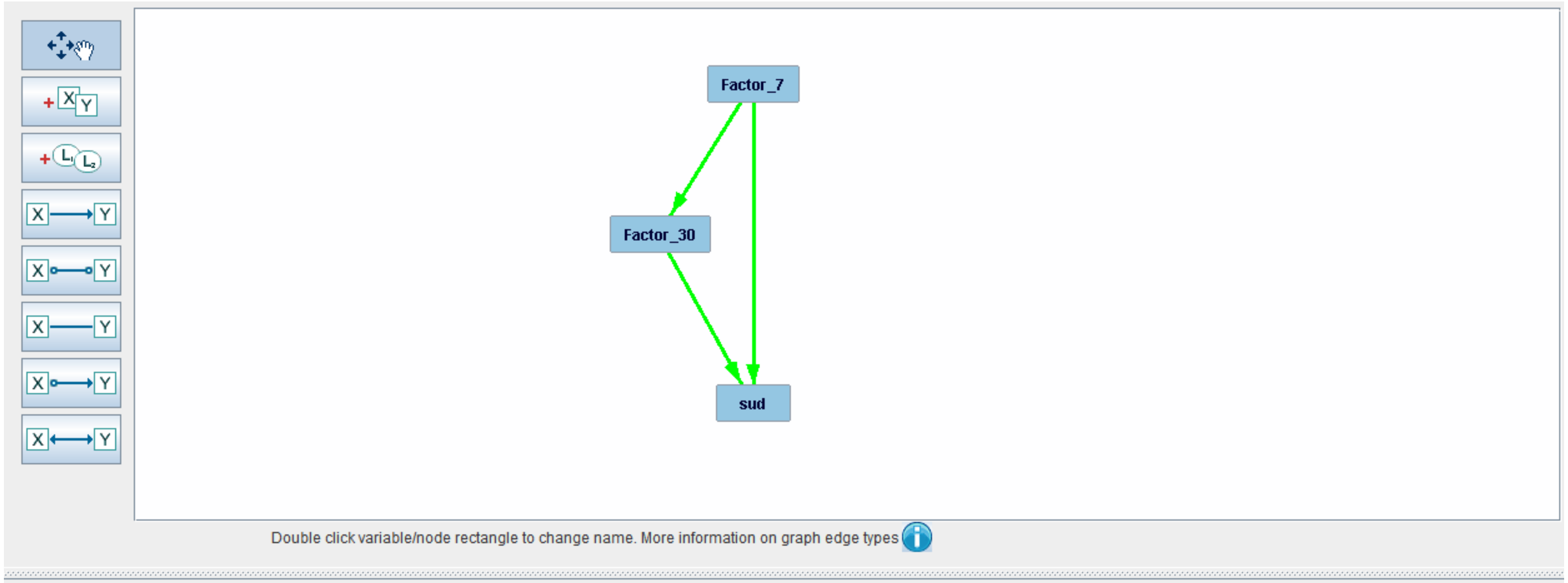


Age Group	Factors	Items Code	Loadings	Question details
Age 10-12	Factor 5	BDS23	0.6562	Once you have a goal, do you make a plan for how to reach it?
		BDS34	0.8755	Do you develop a plan for all your important goals?
		BDS35	0.8165	Do you put your plans into action?
		BDS45	0.5384	Do you spend time thinking about how to reach your goals?
		BDS55	0.4428	Do you consider what will happen before planning?
	Factor 24	CBCLT125	0.8518	Within the past 2 months, was the student overanxious to please?
	Factor 27	DT37	0.8298	Do you eat about the same amount at breakfast from day today?
	Factor 28	TC35	0.8448	Had you ever stolen before age 13?
	Factor 31	DT52	0.8227	Are you happy generally?
Age 12-14	Factor 7	CBCLC53	0.8960	Did your child overeat within the past 6 months?
		CBCLC55	0.9306	Did your child overweight within the past 6 months?
	Factor 30	CA45	0.8423	Did you carry a hidden weapon?
Age 16	Factor 10	CA36	0.6036	Did you skip classes or school without an excuse?
		CA37	-0.9013	Did you skip classes alone or were others with you?
	Factor 17	DSB142	0.9036	Did you belong to a gang?
	Factor 71	MP104	0.8772	Are you more likely to be fast and careless than to be slow?
Age 19	Factor 37	AD4	0.8389	Did you often skip school before you were 15?
Age 22	Factor 13	ANDREW38	0.8231	Have you been a grand theft?
		ANDREW44	0.8062	Have you been an auto theft?
	Factor 58	ANDREW10	0.8277	Did you have a history of beyond control?
	Factor 75	TPG30	0.8899	Do you feel you safe in your neighborhood?
	Factor 90	DSB152	0.8970	Was your free time spent just hanging out with friends?
	Factor 92	DSB150	0.8666	Were the parents absent at most of the parties you have gone to?



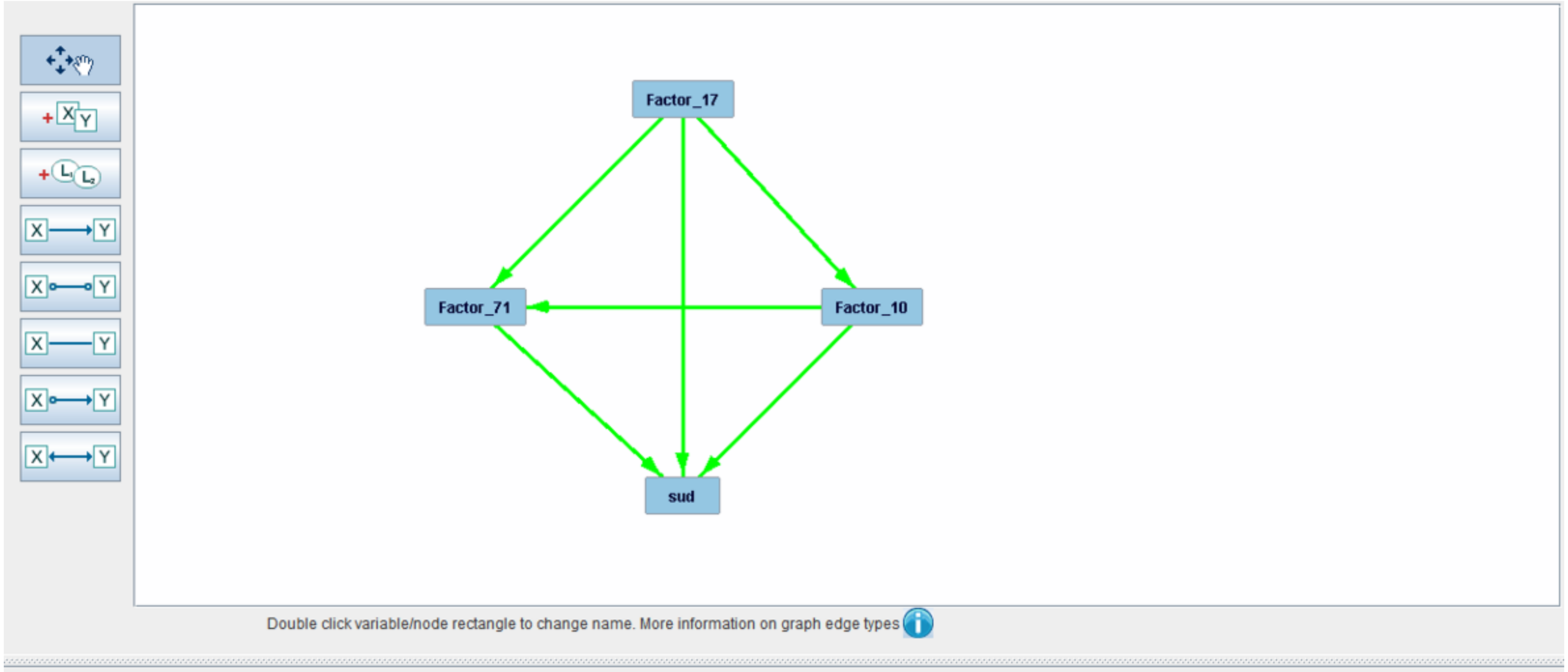
Double click variable/node rectangle to change name. More information on graph edge types

Figure 4.2.2A Causal B networks for children at age10-12



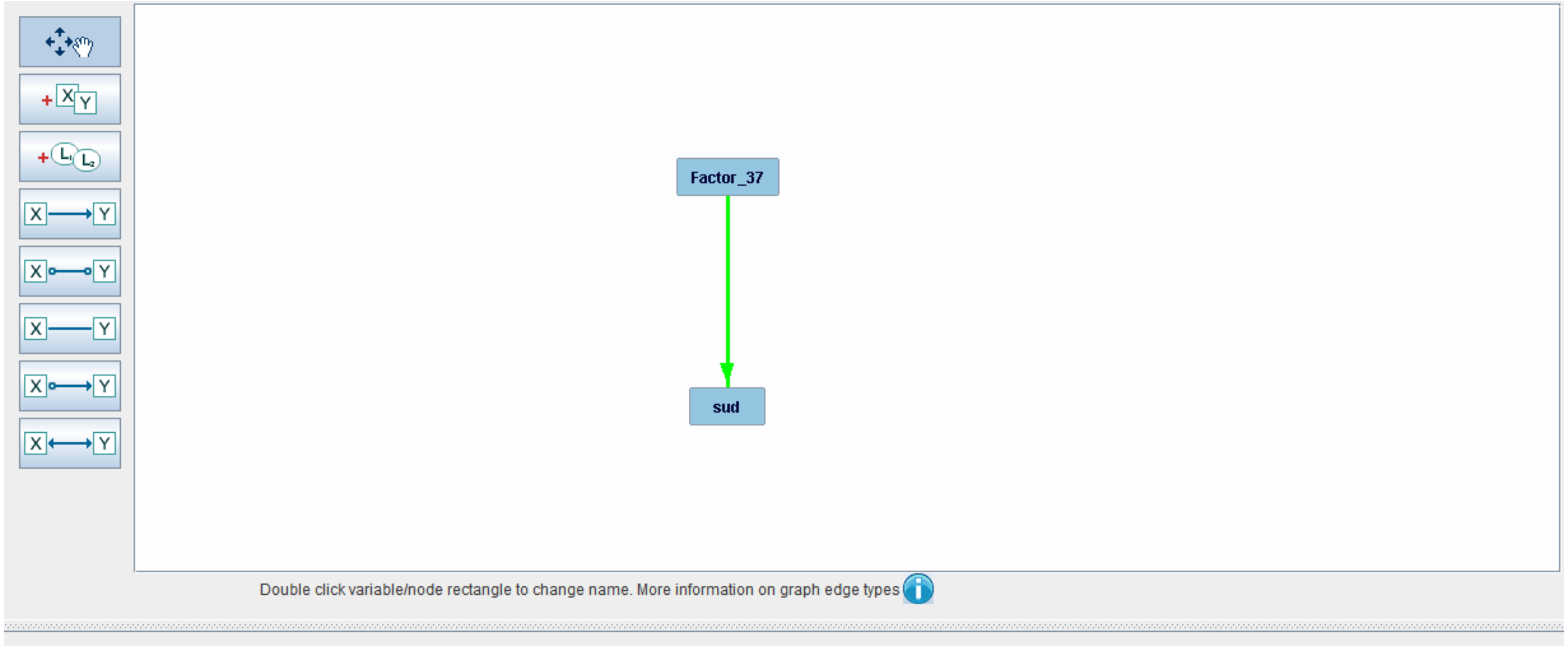
Double click variable/node rectangle to change name. More information on graph edge types

Figure 4.2.2B Causal networks for children at age12-14



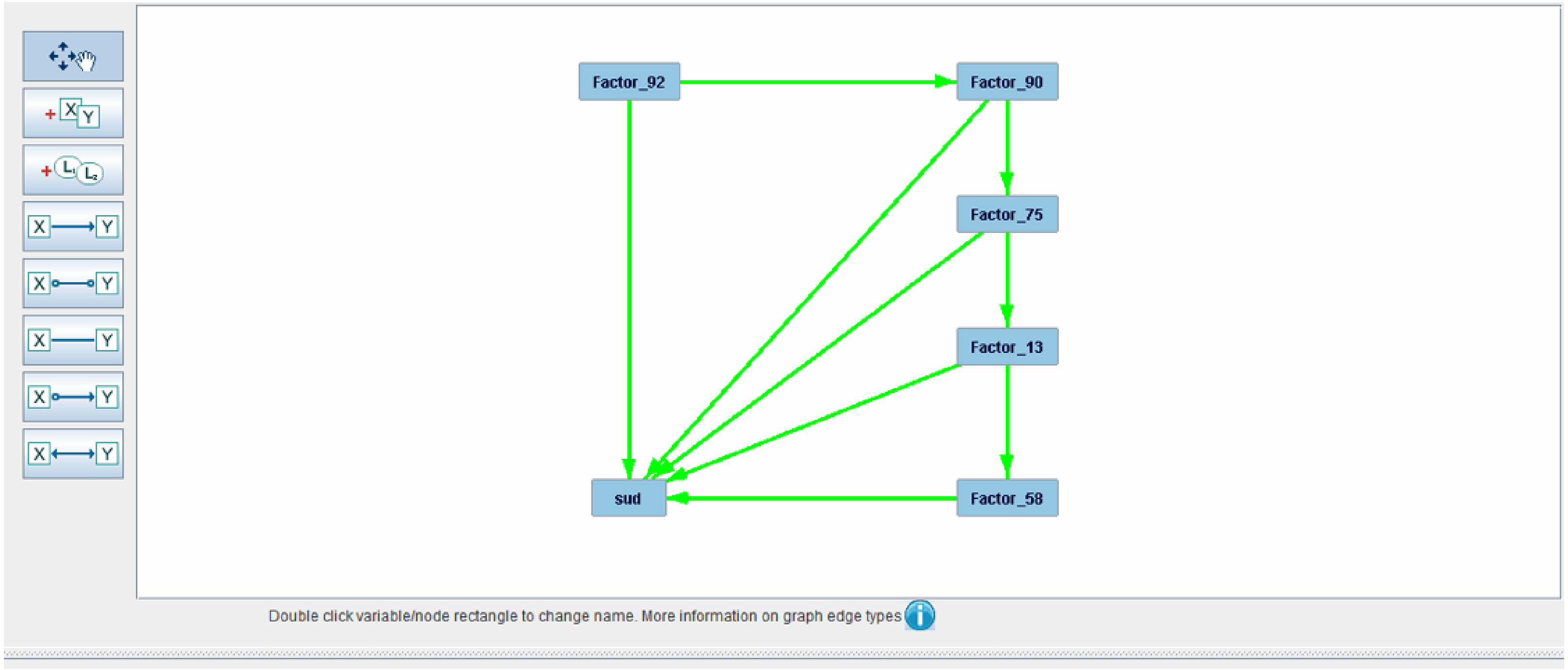
Double click variable/node rectangle to change name. More information on graph edge types

Figure 4.2.2C Causal networks for children at age16



Double click variable/node rectangle to change name. More information on graph edge types

Figure 4.2.2D Causal networks for children at age19



Double click variable/node rectangle to change name. More information on graph edge types

Figure 4.2.2E Causal networks for children at age22

4.2.3.3 Model performance evaluation

The causal-ML models were validated to predict SUD using 10-folds cross-validation. For comparison, traditional ML-based models generated using all factors from EFA were trained and then evaluated using 10-folds cross-validation. In addition, published ML-based models using original question items as features were also compared. The results of the comparison between causal-ML models and traditional ML-based models were presented in **Figure 4.2.3**.

The overall performance of causal-ML models (Orange bars in **Figure 4.2.3**) was generally as good as traditional machine learning models (Blue bars in **Figure 4.2.3**) in terms of predictive accuracy (ROC AUC). The predicted accuracy of causal-ML models in the cross-validation achieved a reliable performance across all the assessment waves. The predictive accuracy reaches 0.73 for 10-12 years of age and 0.87 for 22 years of age. We also made a comparison between this causal-ML model and our published ML-based work (Gray bars in **Figure 4.2.3**) (Jing et al. 2020), which used ML-based feature selection to select important variables. The results show that the causal-ML models are comparable to the published models, especially in the models for younger ages (i.e., at age 10-12, 12-14). In addition, the random forest method, which performed best in our previous study, did not dominate for this task. Regardless of the algorithms used for generating models, the accuracy of forecasting SUD increases with chronological age. It is also noteworthy that the standard deviation of the ROC AUC in the 10-fold cross-validation is somewhat high in all ML-based models indicating that the algorithms are interchangeable with respect to this measure of model quality. Further evaluation of the models was performed using the extra test set (**Table 4.4**). The performance on the test set aligned well with the performance in the cross-validation. The general predictive accuracy of causal-ML models on the test set is not as good as the training performance but still promising. Moreover, the comparison between training and test performance showed that there is no significant overfitting issue. Overall, the models constructed using causal-ML algorithms were reliable for predicting SUD.

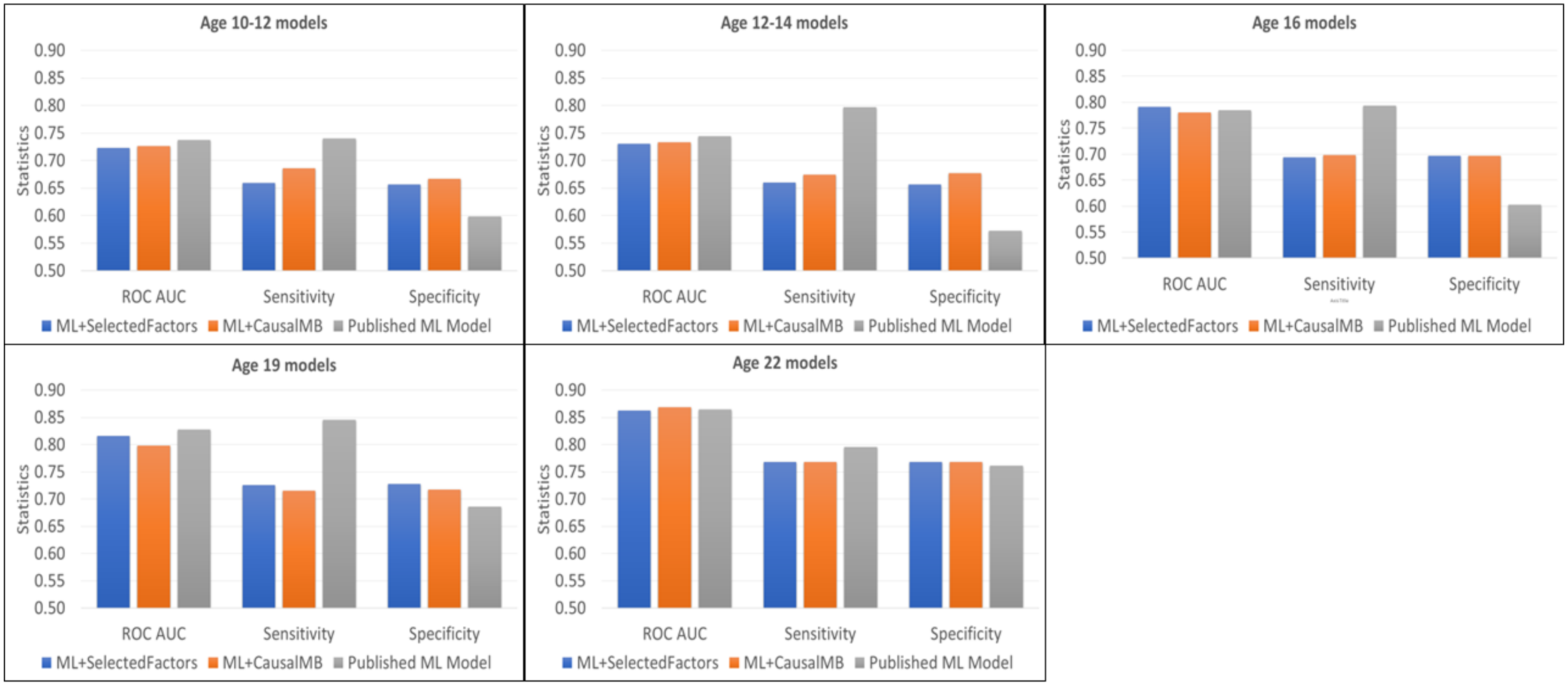
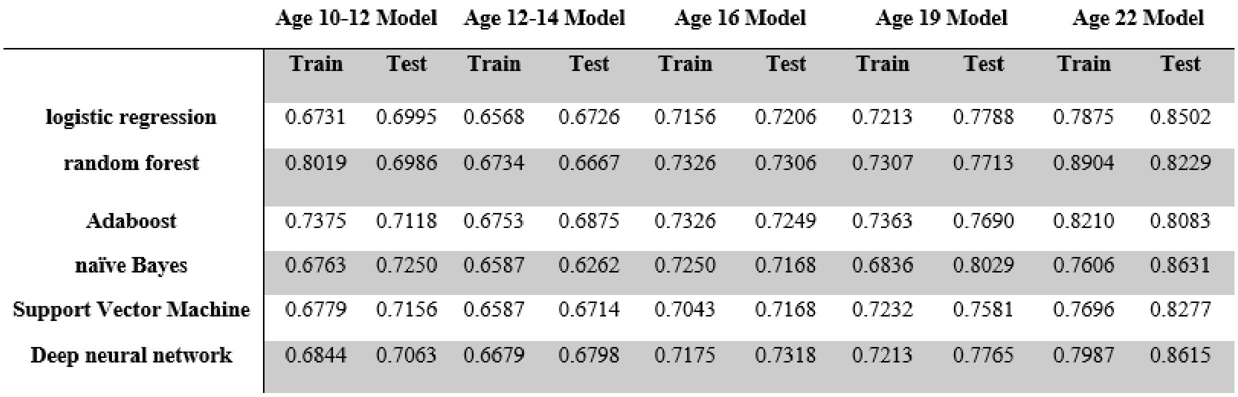


Figure 4.2.3 Cross-validation evaluation of comparative models for predicting SUD**.** Each figure describes the comparison between different models for predict SUD which is introduced in the methods section. The X axis is different evaluation matrix, and the Y axis is the calculated value of those matrix. ML: machine learning; ROC: Receiver operating characteristic; AUC: area under curve

Table 4.4 Training and test accuracy of causal machine learning models



	Age 10-12 Model		Age 12-14 Model		Age 16 Model		Age 19 Model		Age 22 Model	
	Train	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**	**Train**	**Test**
logistic regression	0.6731	0.6995	0.6568	0.6726	0.7156	0.7206	0.7213	0.7788	0.7875	0.8502
random forest	0.8019	0.6986	0.6734	0.6667	0.7326	0.7306	0.7307	0.7713	0.8904	0.8229
Adaboost	0.7375	0.7118	0.6753	0.6875	0.7326	0.7249	0.7363	0.7690	0.8210	0.8083
naïve Bayes	0.6763	0.7250	0.6587	0.6262	0.7250	0.7168	0.6836	0.8029	0.7606	0.8631
Support Vector Machine	0.6779	0.7156	0.6587	0.6714	0.7043	0.7168	0.7232	0.7581	0.7696	0.8277
Deep neural network	0.6844	0.7063	0.6679	0.6798	0.7175	0.7318	0.7213	0.7765	0.7987	0.8615

4.2.4 Discussion and Conclusion

To briefly recapitulate, the results from the presented analysis using causal networks analysis demonstrate the feasibility of examining the potential pattern cluster as well as detecting high-risk youths. For children in early adolescence, it should be pointed out that in the causal factor at the age of 10-12 (factor 31), there was a question item (DT52) owning a very high negative coefficient. That question exams the happiness emotion of the child, which gives the parents and society notice of how important it is to create a peaceable environment to contribute to children's happiness. Moreover, it's important for parents to help children find their milestones, making plans, and motivating them to do things effectively are also important. For young children in middle adolescence, their school attendance and performance may causally relate to their future development, which has been considered as a predictor of SUD for young children in other studies (Crum et al. 2006), and our results confirmed this from the causal aspect, not just correlation. That is the same as the causality between risk and illegal behaviors and SUD. Inasmuch as these causal factors were identified independently of assumptions or biases of the investigator, their importance resides in their salience for prioritization of intervention tactics. It is also noteworthy that the portending SUD factors may be detectable before ten years of age, the baseline in this study. These findings point to the feasibility of risk detection and intervention in early childhood.

From the perspective of making predictions, the results of this prospective study demonstrated that models constructed using causal-ML predict SUD with satisfying accuracy across five-time points spanning from 10-12 to 22 years of age. Accuracy of forecasting SUD points to the feasibility of using the combination of causal analysis and ML algorithms in routine screening of youths for the timely determination of the need for intervention. Considering that no matter which methods are used to develop models, the selected items require only a few minutes to administer and score on the Web platform. Screening can be conducted expeditiously, inexpensively, and unobtrusively since none of the prognostic features query substance use.

Several limitations in this study are noted. First, the procedure of EFA derived a continuous score from original discrete variables, which was an advantage for developing a scale, as many psychiatric studies have done. However, this step reduced the total variance of the entire dataset and manually introduced noise into the indeterminacy, and finally, limited the predictability of the predictive models. Second, it is noteworthy that the sample size was relatively small compared to the number of variables and/or factors, especially when modeling all five assessment waves together, which influenced the reliability of the heuristic study from the causal analysis. Furthermore, although the performance of the models ranged from satisfactory to excellent as age increased, improvement may be achieved by expanding the spectrum of variables available for feature selection and removing redundant information using other methods. Notwithstanding these limitations, this study demonstrates the heuristic value of causal network analysis to identify youths at high risk for SUD.

In conclusion, causal network analysis identified the causal relationship between psychopathological and behavioral cluster patterns and the risk of SUD from late childhood to adulthood. The development of SUD can be forecasted by these patterns with sufficient accuracy to justify application in routine screening to detect high-risk individuals. Moreover, ranking the components of risk according to their contribution to SUD development informs prioritizing intervention targets. Hence, reviewing an individual’s protocol quickly provides insight into the potential etiology of the development of SUD as well as the resources needed to ameliorate the particular risk factor to lower the risk for SUD.

CHAPTER 5. SUMMARY AND PERSPECTIVE

Over the past decades, AI/ML techniques, including state-of-art DL methods, have provided opportunities to accelerate the drug discovery and development pipeline. As shown in this study, the applications of those novel Pharmaco-Analytics technologies enhance the entire cycle, from drug-target identification in the early discovery stage, to preclinical modeling, and finally to clinical outcomes. With the size of biomedical data becoming 'bigger' and computers becoming more powerful, AI/ML methods will inevitably produce ever better performance. Accordingly, it is expected that the number of applications in Pharmaco-Analytics in the coming years will continue to increase. We have described some AI/ML/DL method developments at different stages of drug design and development, which can also be readily applied to other fields such as healthcare research. Combining these methods with drug discovery will most likely lead to significant advances in personalized medicine.

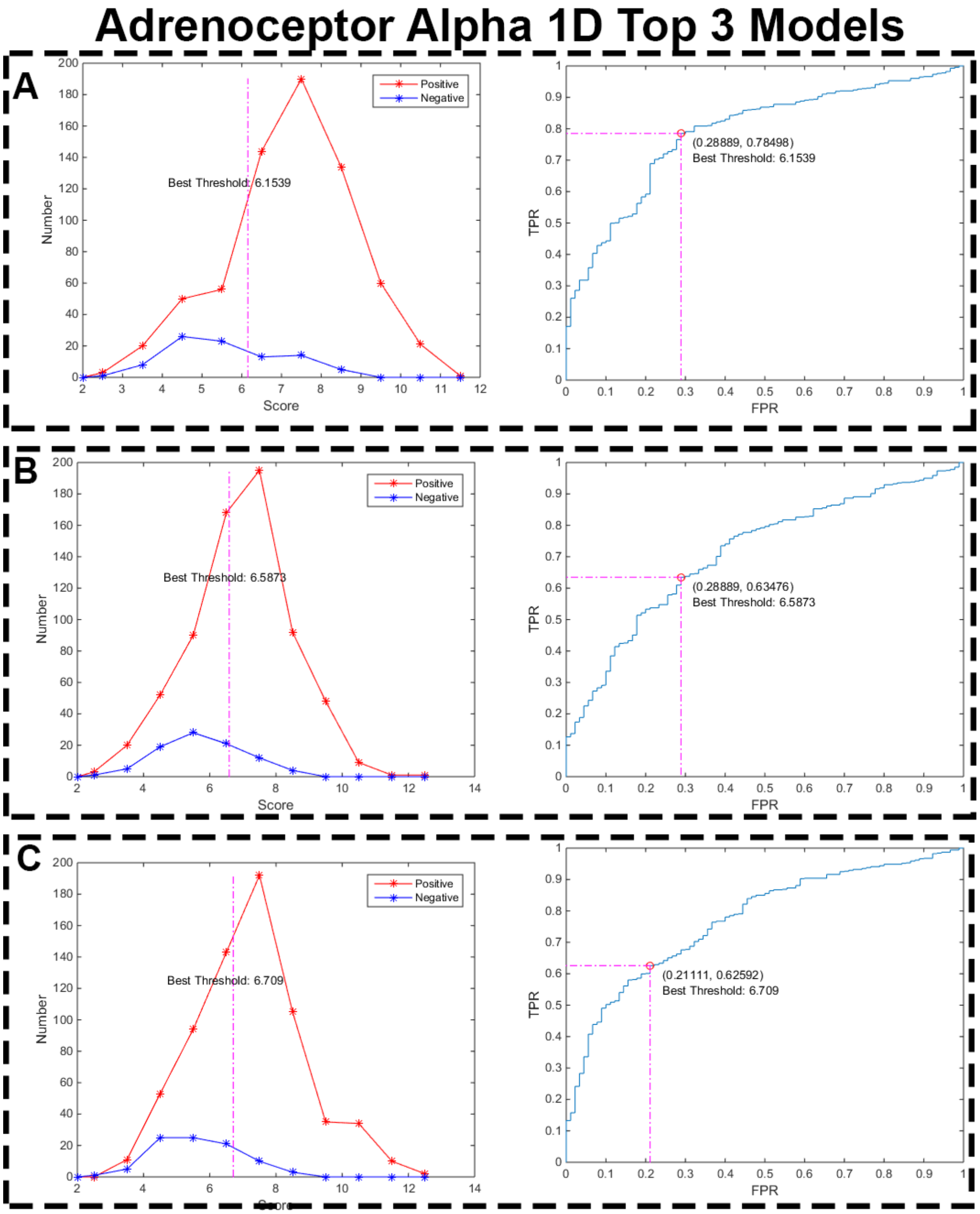
Even though ML/DL methods have been successfully applied in diverse areas, challenges remain applying them in Pharmaco-Analytics research. For example, ML/DL methods strongly rely on the training data, whereas the corresponding experimental datasets are less than optimal. Compared to the amount of 'Big Data' for training the DL models in general AI such as the AlphaGo, the size of the biomedical database for Pharmaco-Analytics modeling lags far behind. Despite that the size of a major database like ChEMBL has reached a magnitude of a million, there is still a dearth of available data for building a specific model (Gaulton et al. 2012). Moreover, experimental data are usually sparse and unbalanced, and beset with noise. It is the responsibility of AI/ML scientists to clean the data and comprehensively learn the internal structure of data distribution in order to construct a reliable model.

Additionally, ML/DL systems are considered as 'black box' systems. Thus, they are hard to interpret and have limited power to engage in logical reasoning. Those factors limit the application and approbation of ML/DL in many domains such as clinical data analysis. In cases of small molecule drug discovery, interpretation of a structure-activity relationship (SAR) study is more practical from the descriptor perspective. However, regular features commonly used by traditional machine learning models in current cheminformatics studies to describe the small molecules, such as molecular fingerprints (Rogers and Hahn 2010; Myint and Xie 2015; Myint et al. 2012; Wang et al. 2013), physicochemical properties, topological properties, and thermodynamics properties (Yao and Parkhill 2016), are not fully appropriate to be used in some complex ML/DL architecture (Kearnes et al. 2016). Thus, the development of more interpretable descriptors is dire. Specifically, since ML/DL methods belong to representation learning which can automatically abstract features from raw data, there are two very important problems to ML/DL modeling: 1) how to optimize ML/DL architectures to abstract useful features; 2) how to interpret those features. The lack of interpretability may hinder scientists, even in situations in which ML/DL perform better than human experts. The lack of interpretability of the approaches makes it more difficult to troubleshoot when they unexpectedly fail on new unseen data sets. Another important issue for ML/DL is repeatability, since ML/DL outputs sometimes are highly dependent on the initial values or weights of the network parameters or even the order in which training samples are presented to the model, as all of them are typically chosen at random. The situation that different ML methods may have totally different results will add uncertainty to the adoption of these methods.

AI/ML has also been applied to electronic health records and real-world evidence to facilitate clinical outcomes research. Current applications of AI in clinical outcome analysis mainly include the use of ML/DL methods to support clinical decision-making such as diagnosis and prediction using various types of data such as electronic medical records. For example, many studies have been conducted using the medical image of the brain for the detection of AD (Tanveer et al. 2020). However, limited studies focus on giving physician causal-inference suggestions on personalized medication or combinational therapy for individual patients. However, how to model intervention using observational data to discover causal connections between observations and outcomes is still a challenge. Causal inference using Bayesian networks is an alternative method for addressing this problem. Therefore, it is a powerful tool for explanatory analysis, which is expected to enable the modern ML/DL to become explainable (Kuang et al. 2020). In comparison with causal networks, current ML/DL technologies in AI have limitations. Models built using those methods usually perform better than models of traditional methods such as logistic regression, however, those methods have limited capability of understanding and interpreting the predicted results. As clinical decision support systems are more frequently applied in intervening in practice, it is critical to correctly predict and understand the causal effects of these predictions and interventions in the medical domain, especially when informing clinical decision support (Holzinger et al. 2019). Conventional ML methods build predictive models based on pattern recognition and correlational analyses without any statistical test, which are insufficient for causal reasoning. Therefore, the development of explainable Artificial Intelligence is warranted to establish trust in models for clinical decision-making, and more and more AI research come to focus on using causal inference to explore the causality and explain-ability of AI technology in clinical research (Pearl 2019; Ohlsson and Kendler 2019; Demmer and Papapanou 2020)

APPENDIX

Appendix A. Supplemental Data for Chapter 2



Supplementary Figure S2.1 Statistical results of 3 models of adrenoceptor alpha 1d (ADRA1D). TPR: true positive rate, FPR: false positive rate, Number: numbers of ligand, Score: docking score

Supplementary Table S2.1 DA-related GPCR targets and data information

Target Name	Total Compound	Active Compd	Inactive Compd	Acc_LR	Acc_RF	Acc_svm	Acc_mlp
ADORA1	5373	2530	2843	0.65	0.64	0.64	0.53
ADORA2A	6264	2937	3327	0.65	0.64	0.64	0.53
ADORA2B	3211	1421	1790	0.68	0.66	0.68	0.56
ADORA3	5661	2647	3014	0.7	0.72	0.71	0.69
ADRA1D	2469	1057	1412	0.69	0.7	0.69	0.57
ADRA2A	1168	550	618	0.64	0.67	0.66	0.63
ADRA2B	790	349	441	0.65	0.66	0.64	0.56
ADRA2C	914	365	549	0.65	0.66	0.67	0.65
ADRB1	1724	799	925	0.65	0.65	0.67	0.54
ADRB2	2014	989	1025	0.68	0.68	0.69	0.51
ADRB3	2993	1301	1692	0.76	0.78	0.77	0.57
AGTR2	690	337	353	0.8	0.81	0.8	0.51
AVPR1A	1354	656	698	0.8	0.81	0.79	0.52
BDKRB2	829	403	426	0.78	0.8	0.78	0.51
CALCR	19	9	10	na	na	na	na
CCKAR	571	285	286	0.84	0.84	0.84	0.5
CCKBR	1921	957	964	0.75	0.74	0.76	0.5
CCR1	1072	536	536	0.76	0.77	0.78	0.5
CCR2	2668	1334	1334	0.78	0.77	0.78	0.5
CCR4	417	208	209	0.7	0.71	0.72	0.7
CCR5	3447	1723	1724	0.86	0.85	0.86	0.5
CHRM1	2535	1091	1444	0.65	0.64	0.63	0.57
CHRM2	2190	1049	1141	0.68	0.68	0.68	0.56
CHRM3	2600	1284	1316	0.69	0.7	0.69	0.51
CHRM4	872	425	447	0.62	0.62	0.62	0.51
CHRM5	577	288	289	0.53	0.63	0.63	0.5
CNR1	5587	2793	2794	0.68	0.67	0.68	0.5
CNR2	7357	3679	3678	0.67	0.66	0.67	0.5
CXCR1	302	147	155	0.8	0.8	0.83	0.51
CXCR2	1153	561	592	0.73	0.74	0.75	0.51
CXCR3	1501	747	754	0.81	0.81	0.82	0.5
CXCR4	568	283	285	0.76	0.75	0.74	0.5
CYSLTR1	370	185	185	0.89	0.91	0.92	0.5
DRD1	1507	753	754	0.67	0.67	0.67	0.5
DRD2	8298	4521	3777	0.66	0.66	0.67	0.54
DRD3	6617	3375	3242	0.7	0.72	0.72	0.63
DRD4	3923	1942	1981	0.63	0.64	0.65	0.5
DRD5	283	138	145	0.59	0.6	0.64	0.51
EDNRA	2260	1116	1144	0.76	0.75	0.77	0.51

GABBR1	5	1	4	na	na	na	na
GRM1	1052	526	526	0.73	0.72	0.73	0.5
GRM2	768	390	378	0.68	0.73	0.7	0.55
GRM3	143	68	75	0.66	0.65	0.66	0.52
GRM4	393	158	235	0.77	0.75	0.77	0.6
GRM5	2681	1341	1340	0.64	0.64	0.66	0.5
GRM6	69	7	62	na	na	na	na
GRM7	64	1	63	na	na	na	na
GRM8	63	9	54	na	na	na	na
HRH1	1774	884	890	0.67	0.67	0.68	0.5
HRH2	217	108	109	0.58	0.63	0.63	0.5
HRH3	6197	3081	3116	0.62	0.64	0.65	0.5
HRH4	1610	803	807	0.72	0.72	0.72	0.5
HTR1A	6471	3233	3238	0.69	0.7	0.7	0.5
HTR1B	1810	904	906	0.75	0.75	0.74	0.5
HTR1D	2132	1065	1067	0.71	0.72	0.72	0.5
HTR1E	36	18	18	0.6	0.6	0.55	0.5
HTR1F	213	106	107	0.76	0.73	0.76	0.5
HTR2A	6235	3116	3119	0.68	0.68	0.7	0.5
HTR2B	2158	1080	1078	0.63	0.64	0.66	0.5
HTR2C	4641	2319	2322	0.62	0.65	0.66	0.5
HTR4	917	458	459	0.77	0.78	0.76	0.5
HTR5A	387	184	203	0.73	0.73	0.72	0.52
HTR6	5191	2596	2595	0.69	0.68	0.68	0.5
HTR7	2751	1372	1379	0.69	0.69	0.7	0.5
MC1R	966	470	496	0.76	0.78	0.76	0.51
MC2R	1	0	1	na	na	na	na
MC3R	826	413	413	0.66	0.66	0.62	0.5
MC4R	4306	2113	2193	0.84	0.84	0.85	0.51
MC5R	627	300	327	0.77	0.79	0.79	0.52
NPBWR1	169	87	82	0.8	0.83	0.79	0.51
NPFFR1	120	60	60	0.79	0.85	0.81	0.5
NPSR1	134	67	67	0.78	0.82	0.8	0.5
NPY1R	1016	495	521	0.81	0.82	0.83	0.51
NPY2R	575	269	306	0.77	0.78	0.79	0.53
NPY4R	62	26	36	0.95	0.97	0.97	0.58
NPY5R	2085	1042	1043	0.79	0.78	0.78	0.5
OPRD1	5310	2587	2723	0.73	0.72	0.73	0.51
OPRK1	5711	2852	2859	0.73	0.72	0.73	0.5
OPRL1	2539	1263	1276	0.74	0.73	0.74	0.5
OPRM1	5494	2746	2748	0.74	0.72	0.73	0.5
PTAFR	628	314	314	0.75	0.78	0.75	0.5
TAAR1	383	190	193	0.69	0.68	0.68	0.5

TACR1	4492	2245	2247	0.83	0.83	0.84	0.5
TACR2	1517	758	759	0.87	0.87	0.87	0.5
TSHR	25	11	14	na	na	na	na

Appendix B. Supplemental Data for Chapter 3

Supplementary Table S3.1 Atom type (AMBER) used for graph-based small molecular feature calculation.

Atom	Class	Atom type	Description
C	Basic	c	sp2 C in C=O, C=S
		c1	sp1 C
		c2	sp2 C, aliphatic
		c3	sp3 C
		ca	sp2 C, aromatic
	Special	cc(cd)	inner sp2 C in conj. ring systems
		ce(cf)	inner sp2 C in conj. chain systems
		cp(cq)	bridge aromatic C
		cu	sp2 C in three-memberred rings
		cv	sp2 C in four-memberred rings
		cx	sp3 C in three-memberred rings
		cy	sp3 C in four-memberred rings
N	Basic	n	sp2 N in amide
		n1	sp1 N
		n2	sp2 N with 2 subst. readl double bond
		n3	sp3 N with 3 subst.
		n4	sp3 N with 4 subst.
		na	sp2 N with 3 subst
		nh	amine N connected to the aromatic rings
		no	N in nitro group
	Special	n	aromatic nitrogen
		nb	inner sp2 N in conj. ring systems
		nc(nd)	inner sp2 N in conj. chain systems
O	Basic	o	sp2 O in C=O, COO-
		oh	sp3 O in hydroxyl group
		os	sp3 O in ether and ester
S	Basic	s2	sp2 S (p=S, C=S etc)
		sh	sp3 S in thiol group
		ss	sp3 S in -SR and SS
		s4	hypervalent S, 3 subst.
		s6	hypervalent S, 4 subst.
	Special	sx	conj. S, 3 subst.
		sy	conj. S, 4 subst.
P	Basic	p2	sp2 P (C=P etc)
		p3	sp3 P, 3 subst.

		p4	hypervalent P, 3 subst.
		p5	hypervalent P, 4 subst.
	Special	pb	aromatic phosphorus
		pc(pd)	inner sp2 P in conj. ring systems
		pe(pf)	inner sp2 P in conj. chain systems
		px	conj. P, 3 subst.
		py	conj. P, 4 subst.
H	Basic	hc	H on aliphatic C
		ha	H on aromatic C
		hn	H on N
		ho	H on O
		hs	H on S
		hp	H on P
	Special	h1, h2, h3, h4, h5	H on aromatic C with 1-5 EW group;
halogen	Basic	f	any F
		cl	any Cl
		br	any Br
		i	any I

Table S3.2 Model optimization and hyper-parameter tuning in DeepGhERG training process.

GNN types	n_GCN	n_MLP	batch_size	lr	dropout	K	optimizer	bestEpoch	ValAcc	TrainAcc
GCN_Adj	5	5	100	0.001	0.1	N/A	rmsprop	99	0.8036	0.8143
	5	3	100	0.001	0.1	N/A	adam	84	0.8036	0.8143
	1	5	100	0.001	0	N/A	adam	92	0.7988	0.8571
	4	3	100	0.001	0	N/A	adam	92	0.7988	1.0000
	5	3	100	0.001	0.2	N/A	rmsprop	99	0.7975	0.8571
	5	1	100	0.001	0.2	N/A	adam	71	0.7974	0.8571
	1	1	100	0.001	0.1	N/A	adam	57	0.7925	0.8571
	6	5	100	0.001	0.1	N/A	adam	81	0.7924	0.8571
	2	5	100	0.001	0	N/A	adam	68	0.7911	0.8571
	6	3	100	0.001	0.1	N/A	adam	58	0.7900	0.5714
	1	3	100	0.001	0.2	N/A	adam	83	0.7886	1.0000
	2	3	100	0.001	0.1	N/A	adam	98	0.7875	1.0000
	3	3	100	0.001	0.2	N/A	adam	84	0.7875	1.0000
	6	3	100	0.001	0	N/A	adam	90	0.7874	1.0000
	3	5	100	0.001	0	N/A	adam	63	0.7874	0.7143
	1	3	100	0.001	0.2	N/A	rmsprop	94	0.7863	0.8571
	6	3	100	0.001	0	N/A	rmsprop	91	0.7850	0.8571
	6	1	100	0.001	0	N/A	adam	72	0.7850	0.5714
	2	5	100	0.001	0.2	N/A	adam	90	0.7849	1.0000
	4	3	100	0.001	0.1	N/A	adam	89	0.7838	1.0000
	6	1	500	0.001	0.1	N/A	rmsprop	96	0.7836	0.7420
	3	5	100	0.001	0.2	N/A	rmsprop	79	0.7825	1.0000
	6	5	100	0.001	0	N/A	adam	69	0.7825	0.8571
	4	3	500	0.01	0	N/A	adam	99	0.7820	0.8256
	2	5	100	0.01	0	N/A	adam	95	0.7811	1.0000
	4	1	100	0.001	0	N/A	adam	93	0.7788	0.7143
	4	3	500	0.001	0.1	N/A	adam	90	0.7781	0.8034
	1	1	100	0.001	0	N/A	adam	74	0.7775	0.7143
	5	1	100	0.001	0.2	N/A	rmsprop	93	0.7775	0.8571
	3	5	500	0.001	0.2	N/A	adam	77	0.7773	0.7617
GCN_ Laplacian	4	1	100	0.001	0.2	N/A	rmsprop	99	0.8100	0.8571
	4	1	100	0.001	0	N/A	adam	73	0.7913	0.4286
	4	3	500	0.001	0.2	N/A	adam	96	0.7897	0.7617
	3	1	100	0.001	0.5	N/A	adam	76	0.7888	0.7143
	1	5	100	0.001	0.1	N/A	rmsprop	70	0.7886	1.0000
	3	5	100	0.001	0	N/A	adam	64	0.7861	0.4286
	4	5	100	0.001	0	N/A	adam	46	0.7825	1.0000
	6	1	100	0.001	0	N/A	rmsprop	80	0.7824	1.0000
	2	3	500	0.001	0	N/A	adam	98	0.7802	0.7617

GATConv	3	3	100	0.001	0.5	N/A	adam	91	0.7800	0.7143
	6	3	100	0.001	0	N/A	adam	91	0.7799	1.0000
	3	1	100	0.001	0.1	N/A	rmsprop	92	0.7799	0.7143
	5	3	500	0.001	0.2	N/A	adam	96	0.7797	0.7936
	2	5	1000	0.01	0	N/A	adam	75	0.7792	0.8329
	4	3	100	0.001	0.2	N/A	adam	99	0.7788	0.7143
	3	3	100	0.001	0.1	N/A	adam	74	0.7786	0.7143
	2	5	100	0.001	0	N/A	rmsprop	58	0.7786	1.0000
	2	5	100	0.001	0	N/A	adam	94	0.7763	0.7143
	3	5	100	0.001	0.2	N/A	rmsprop	93	0.7763	0.8571
	4	3	100	0.001	0	N/A	adam	96	0.7761	1.0000
	4	1	500	0.001	0.1	N/A	rmsprop	94	0.7752	0.7076
	2	5	100	0.001	0.1	N/A	adam	60	0.7750	0.8571
	6	3	500	0.01	0	N/A	adam	97	0.7750	0.8010
	2	1	100	0.001	0.2	N/A	adam	81	0.7749	0.7143
	3	1	100	0.001	0.1	N/A	adam	93	0.7735	0.8571
	4	3	500	0.001	0.1	N/A	adam	69	0.7735	0.7666
	1	1	100	0.001	0.2	N/A	adam	60	0.7725	1.0000
	1	3	100	0.001	0.1	N/A	rmsprop	59	0.7725	0.7143
	5	3	100	0.001	0	N/A	adam	96	0.7725	0.8571
	1	5	100	0.001	0	N/A	adam	84	0.7713	1.0000
	2	3	100	0.001	0	N/A	adam	70	0.7825	0.8571
	1	5	100	0.001	0.2	N/A	adam	86	0.7813	0.5714
	2	5	100	0.001	0	N/A	rmsprop	83	0.7725	1.0000
	2	5	100	0.001	0	N/A	adam	63	0.7713	0.7143
	1	3	100	0.001	0	N/A	adam	55	0.7686	1.0000
	1	1	100	0.001	0.1	N/A	adam	89	0.7674	1.0000
	1	3	100	0.001	0.1	N/A	rmsprop	63	0.7648	0.7143
	1	3	500	0.01	0.1	N/A	rmsprop	80	0.7645	0.7862
	1	3	500	0.001	0	N/A	rmsprop	98	0.7640	0.8256
	1	3	100	0.01	0.1	N/A	adam	95	0.7637	0.8571
	1	3	100	0.001	0	N/A	rmsprop	87	0.7636	0.8571
	1	5	100	0.001	0	N/A	adam	73	0.7624	0.8571
	1	1	100	0.001	0	N/A	adam	76	0.7613	0.8571
	3	5	100	0.001	0	N/A	adam	66	0.7611	0.7143
	1	3	100	0.01	0.5	N/A	adagrad	95	0.7600	0.7143
	2	5	100	0.001	0.1	N/A	adam	67	0.7588	0.7143
	1	3	100	0.001	0.2	N/A	adam	74	0.7588	0.8571
	1	5	100	0.01	0	N/A	adam	99	0.7586	0.7143
	3	5	100	0.001	0	N/A	rmsprop	67	0.7586	1.0000
	1	5	500	0.001	0	N/A	adam	97	0.7584	0.8698
	1	5	100	0.001	0.1	N/A	adam	69	0.7563	0.8571
	2	5	500	0.001	0.1	N/A	adam	81	0.7558	0.7445

ChebConv	1	5	1000	0.01	0	N/A	adam	90	0.7553	0.8771
	1	5	100	0.001	0.1	N/A	rmsprop	44	0.7550	0.8571
	2	3	100	0.001	0.1	N/A	adam	93	0.7549	0.5714
	1	1	100	0.001	0	N/A	rmsprop	64	0.7538	1.0000
	2	5	500	0.001	0	N/A	adam	82	0.7536	0.8059
	1	1	500	0.01	0.1	N/A	adam	47	0.7535	0.7150
	2	5	500	0.01	0	N/A	adam	58	0.7527	0.7764
	1	1	100	0.001	0.1	N/A	rmsprop	52	0.7525	0.8571
	3	3	100	0.001	0.5	4	adam	96	0.8138	0.8571
	6	3	100	0.001	0	4	adam	72	0.8138	1.0000
	5	1	500	0.001	0.1	4	adam	68	0.8132	0.8428
	5	3	500	0.001	0.1	4	adam	94	0.8083	0.8477
	5	1	100	0.001	0	4	rmsprop	97	0.8075	1.0000
	5	1	100	0.001	0.2	4	adam	95	0.8075	1.0000
	3	5	100	0.001	0	4	rmsprop	70	0.8063	1.0000
	5	1	100	0.001	0	4	adam	79	0.8062	1.0000
	5	3	100	0.001	0	4	adam	84	0.8038	0.8571
	2	3	100	0.001	0	4	adam	52	0.8025	0.8571
	6	3	100	0.001	0	3	rmsprop	97	0.8024	1.0000
	5	3	500	0.001	0	4	rmsprop	63	0.8015	0.8108
	4	3	100	0.001	0.1	4	rmsprop	66	0.8011	1.0000
	3	5	500	0.001	0.1	4	adam	89	0.8008	0.8231
	3	3	100	0.001	0	4	adam	67	0.8000	0.7143
	1	1	500	0.001	0.1	3	adam	92	0.7994	0.7322
	4	5	100	0.001	0.1	4	adam	96	0.7988	1.0000
	6	1	100	0.001	0.2	2	adam	99	0.7988	0.7143
	5	3	100	0.001	0.2	4	adam	72	0.7988	1.0000
	2	5	100	0.001	0.1	4	rmsprop	87	0.7986	1.0000
	4	1	100	0.001	0.1	3	adam	91	0.7975	1.0000
	3	1	100	0.001	0.2	4	adam	65	0.7975	0.7143
	3	5	100	0.001	0.5	3	adam	88	0.7974	0.4286
	3	5	100	0.0001	0	4	adam	64	0.7963	0.7143
	2	1	100	0.001	0	3	adam	58	0.7961	0.8571
	4	3	100	0.001	0	4	adam	48	0.7950	0.8571
	5	3	100	0.0001	0	4	rmsprop	79	0.7950	0.8571
	4	3	100	0.001	0.2	4	adam	87	0.7950	0.8571
	6	5	100	0.001	0.5	4	adam	91	0.7938	0.7143
	6	1	100	0.001	0.2	4	rmsprop	68	0.7938	0.8571

Appendix C. Supplemental Data for Chapter 4

Supplementary Table S4.1. Selected 30 features for predicting SUD in age group 1 (Age 10-12)

Questions	Feature Importance	Importance rank	chi2	p-value
Do you often swear or use bad language?	0.0069	1	42.3804	0.0000
Do you have difficulty playing quietly?	0.0055	2	39.0885	0.0000
My child eats about the same amount at breakfast from day to day (Parents)	0.0047	3	10.9055	0.0010
Are you touchy or easily annoyed by others?	0.0043	4	26.8952	0.0000
About how many times a week does your child do things with any friends outside of regular school hours? (Parents)	0.0042	5	4.7421	0.0294
Do you have difficulty staying in line in the supermarket or waiting for your turn while you were playing with other children?	0.0042	6	38.1248	0.0000
Do you deliberately refuse adults, or do you refuse to do your chores at home or disobey rules a lot?	0.0041	7	34.7887	0.0000
Do you often argue with adults?	0.0038	8	30.2260	0.0000
How many jobs, chores do your child has? (Parents)	0.0036	9	3.6153	0.0573
Is your child hard to be distracted? (Parents)	0.0033	10	7.8940	0.0050
Does your child get very restless If he/she has to stay in one place for a long time? (Parents)	0.0032	11	11.0743	0.0009
Does your child get hungry about the same time each day? (Parents)	0.0031	12	5.4537	0.0195
Do you get very fidgety after a few minutes if you're supposed to sit still?	0.0029	13	14.0798	0.0002
Does your child get very fidgety after a few minutes Even when he/she is supposed to be still? (Parents)	0.0028	14	9.3343	0.0022
How many organizations, clubs, teams or groups does your child belong to? (Parents)	0.0028	15	9.0944	0.0026
Within the past 6 months, does your child, hang around with others who get in trouble? (Parents)	0.0027	16	19.4536	0.0000
Compared to others of his/her age, how well does your child play and work alone? (Parents)	0.0027	17	3.2823	0.0700
No matter when your child goes to sleep, does he/she wake up at the same time the next morning? (Parents)	0.0027	18	8.0267	0.0046

Does your child have difficulty following through on instructions from others (not due to oppositional behavior or failure of comprehension), e.g., fails to finish chores? (Parents)	0.0027	19	42.7693	0.0000
Does failure at a task or in school make your work harder?	0.0026	20	3.7005	0.0544
Can you read a book for half an hour before you get restless?	0.0026	21	6.6266	0.0100
Do you get into trouble because you would do things without thinking about them first, for example running into the street without looking?	0.0025	22	29.7495	0.0000
Do you get very restless when you have to stay in one place for a long time?	0.0025	23	8.9215	0.0028
Does your child wake up at the same time each day when he/she is away from home? (Parents)	0.0024	24	8.0571	0.0045
Do your heart beats fast for a long time when you get stirred up?	0.0023	25	4.4068	0.0358
Do you have so much energy that you just can't stop moving?	0.0023	26	8.2014	0.0042
Do you get so excited that I remain very excited for a long time after watching an action show?	0.0023	27	6.5546	0.0105
Are you easily distracted?	0.0023	28	6.9223	0.0085
Compared to others of the same age, about how much time does your child spend in hobbies, activities and games other than sports? (Parents)	0.0023	29	0.6293	0.4276
Do you develop a plan for all your important goals?	0.0022	30	3.3211	0.0684

Supplementary Table S4.2. Selected 30 features for predicting SUD in age group 2 (Age 12-14)

Questions	Feature Importance	Importance rank	chi2	pval
Have you been suspended from school?	0.0116	1	34.1238	0.0000
Have your friends stolen anything from a store or damaged property on purpose?	0.0086	2	25.3322	0.0000
Have any of your friends been in trouble with the law?	0.0050	3	22.4370	0.0000
Is there anyone who would wish to harm you?	0.0047	4	22.2547	0.0000
Do you swear or use dirty language a lot?	0.0041	5	17.8792	0.0000
Are you sure nobody really would wish to harm you?	0.0040	6	4.4896	0.0341
Do you think the people who don't do the work should feel very guilty?	0.0040	7	6.0113	0.0142
Do your friends cut school a lot?	0.0040	8	24.3936	0.0000
Do you think the people are always bugging you deserve a punch in the nose?	0.0039	9	16.7450	0.0000
Were you bothered by problems you were having with a friend?	0.0038	10	12.3399	0.0004
Are your grades below average?	0.0036	11	18.6292	0.0000
Do you get into fights?	0.0035	12	39.6817	0.0000
Have you ever felt tempted to steal something?	0.0034	13	14.6926	0.0001
Do you get into trouble because you would do things without thinking about them first, for example running into the street without looking?	0.0033	14	40.6352	0.0000
Are you often worried that you will lose control of your feeling?	0.0033	15	24.3527	0.0000
Do you think whoever insults you, or your family is looking for trouble?	0.0033	16	22.7151	0.0000
Are your grades in school worse than they used to be?	0.0032	17	16.1606	0.0001
Do you have trouble concentrating in school or when studying?	0.0032	18	12.1541	0.0005
Are you Inattentive, easily distracted?	0.0029	19	23.1874	0.0000
Have you ever been talked into doing something you didn't want to do?	0.0029	20	11.0709	0.0009
Can you tell us the number your favorite hobbies, activities, and games, other than sports?	0.0028	21	3.4236	0.0643
Will little things or distractions throw you off?	0.0028	22	15.6035	0.0001

Is it very hard for you to get used to a new situation?	0.0028	23	4.4760	0.0344
Do you often lose your temper?	0.0028	24	37.1608	0.0000
Do you lose control over your actions sometimes when you're angry?	0.0028	25	23.2534	0.0000
Do your parents or guardians dislike your friends?	0.0028	26	14.6298	0.0001
Have any of your friends cheated on school tests?	0.0027	27	9.7745	0.0018
Sex	0.0027	28	2.7875	0.0950
Do you hit someone when you really get mad?	0.0027	29	25.6019	0.0000
Will you be a little rude to people you don't like?	0.0026	30	14.8206	0.0001

Supplementary Table S4.3. Selected 30 features for predicting SUD in age group 3 (Age 16)

Questions	Feature Importance	Importance rank	chi2	pval
Do you skip classes or school without an excuse?	0.0100	1	90.8518	0.0000
Are the parents absent at most of the parties you have gone to?	0.0086	2	32.1471	0.0000
Have you been suspended from school?	0.0064	3	35.2659	0.0000
Have you ever made money doing something that was against the law?	0.0059	4	39.4514	0.0000
Do you lie about your age to get into some place or to buy something, for example lying about your age to get into a movie or to buy alcohol?	0.0057	5	63.9847	0.0000
Do you often not do your school assignments?	0.0055	6	26.4756	0.0000
Have you ever felt tempted to steal something?	0.0054	7	22.1189	0.0000
Do you deliberately refuse adults, or do you refuse to do your chores at home or disobey rules a lot?	0.0050	8	71.6345	0.0000
Will your friends get bored at parties when there was no alcohol served?	0.0049	9	32.0085	0.0000
Do you prefer to be fast and careless than to be slow and plodding?	0.0048	10	14.4702	0.0001
Do you swear or use dirty language?	0.0046	11	22.5124	0.0000
Do you go out for fun on school nights without permission?	0.0041	12	32.8435	0.0000
Do you like to watch a good, vicious fight?	0.0040	13	17.9742	0.0000
Do you swear or use dirty language a lot?	0.0037	14	18.1234	0.0000
Do you skip classes alone or were others with you?	0.0036	15	4.1741	0.0410
Have you stolen things?	0.0034	16	28.0122	0.0000
Do you think it is pointless spending time on a task that is probably too difficult?	0.0034	17	15.9235	0.0001
Has a member of your family ever been arrested?	0.0033	18	24.8175	0.0000
Do you often swear or use bad language?	0.0033	19	49.8315	0.0000
Are your grades below average?	0.0033	20	26.0356	0.0000
Do you do risky or dangerous things a lot?	0.0030	21	22.7663	0.0000
Do you think it is pointless sticking with a problem if success is unlikely?	0.0030	22	12.8763	0.0003

Are your parents or guardians often unaware of where you were and what you were doing?	0.0027	23	22.1863	0.0000
Do you often go on working on a problem long after others would have given up?	0.0026	24	7.7619	0.0053
Are you often late for class?	0.0025	25	23.6823	0.0000
Do you think most people stay friendly only as long as it is to their advantage?	0.0024	26	11.5344	0.0007
Do your friends cut school a lot?	0.0024	27	28.5453	0.0000
Do you have frequent arguments with your parents/guardians who involved yelling and screaming?	0.0024	28	19.1145	0.0000
Have you stolen or attempted to steal things worth $5 or less?	0.0022	29	54.3203	0.0000
How is your performance in academic subjects?	0.0022	30	3.4943	0.0616

Supplementary Table S4.4. Selected 30 features for predicting SUD in age group 4 (Age 19)

Questions	**Feature Importance**	**Importance rank**	**chi2**	**pval**
Do you do things a lot without first thinking about the consequences?	0.0106	1	46.1616	0.0000
Do you swear or use dirty language a lot?	0.0105	2	28.8758	0.0000
Have you failed to conform to social norms with respect to lawful behavior, as indicated by repeatedly?	0.0103	3	69.6168	0.0000
Do you go out for fun on school nights without permission?	0.0101	4	31.4849	0.0000
Have you ever receiving stolen property (or possession of stolen property)?	0.0098	5	50.3877	0.0000
Do you carry a knife?	0.0097	6	42.8425	0.0000
Have you ever shoplifted?	0.0094	7	41.5452	0.0000
Have you ever made money doing something that was against the law?	0.0091	8	56.9989	0.0000
Have you been suspended from school?	0.0086	9	36.8688	0.0000
Are your parents or guardians often unaware of where you were and what you were doing?	0.0086	10	36.0509	0.0000
Have you ever played truant?	0.0083	11	43.4836	0.0000
Have you ever been arrested after age of 15?	0.0079	12	43.4478	0.0000
Have you ever flighted on school grounds?	0.0078	13	34.9638	0.0000
Do your friends cut school a lot?	0.0073	14	38.6300	0.0000
Have your friends stolen anything from a store or damaged property on purpose?	0.0069	15	35.7664	0.0000
Have you ever been involved into verbal fights, verbally assaultive?	0.0066	16	30.8590	0.0000
Have any of your friends been in trouble with the law?	0.0063	17	30.8590	0.0000
Do you often not do your school assignments?	0.0061	18	32.4980	0.0000
Do you often swear or use obscene language?	0.0059	19	70.1197	0.0000
Do you agree that most mornings the day ahead looks bright?	0.0056	20	10.7965	0.0010
Do you like to watch a good, vicious fight?	0.0055	21	25.5635	0.0000
Are you reckless regarding your own or others' personal safety, as indicated by	0.0055	22	53.1566	0.0000

driving while intoxicated, and/or recurrent speeding?				
Do you have temper?	0.0051	23	26.1920	0.0000
Have any of your friends cheated on school tests?	0.0050	24	27.5915	0.0000
Have you ever taking parent's car?	0.0050	25	34.5161	0.0000
Do you think you are often not as cautious as you should be?	0.0049	26	19.7519	0.0000
Do you do petty theft?	0.0049	27	33.0313	0.0000
Have you ever cut school more than two days a month?	0.0046	28	30.2571	0.0000
Have you ever convicted of a crime after age 15?	0.0046	29	72.2338	0.0000
Do you often truant?	0.0046	30	31.3284	0.0000

Supplementary Table S4.5. Selected 30 features for predicting SUD in age group 5 (Age 22)

Questions	Feature Importance	Importance rank	chi2	pval
Have any of your friends been in trouble with the law?	0.0217	1	59.1422	0.0000
Do your friends get bored at parties when there was no alcohol served?	0.0122	2	35.2838	0.0000
Have you ever been curfewed?	0.0120	3	22.6362	0.0000
Do you like to watch a good, vicious fight?	0.0101	4	31.6867	0.0000
Have you ever been involved into verbal fights, verbally assaultive?	0.0091	5	23.4821	0.0000
Have you ever done shoplifting?	0.0087	6	32.1019	0.0000
Have you ever made money doing something that was against the law?	0.0079	7	44.8398	0.0000
Will you try to retaliate (get even) when someone hurts you?	0.0071	8	29.5773	0.0000
Will you be very embarrassed to tell people that you had spent your vacation at a nudist camp?	0.0068	9	24.6081	0.0000
Have you ever been sexual misbehaved?	0.0061	10	27.7401	0.0000
Have you ever had sexual intercourse (made love or gone all the way)?	0.0061	11	2.4758	0.1156
Have you ever received stolen properly (or possession of stolen property)?	0.0057	12	34.3720	0.0000
Have you ever done trespassing?	0.0057	13	21.2388	0.0000
Do you prefer quiet parties with good conversation or "wild" uninhibited parties?	0.0057	14	16.2933	0.0001
Do you enjoy a good brawl?	0.0056	15	28.5904	0.0000
Have you ever stolen?	0.0055	16	32.1314	0.0000
Do you do risky or dangerous things a lot?	0.0053	17	35.5660	0.0000
Have you ever taken parent's car?	0.0052	18	26.3156	0.0000
Have you ever stopped working at a job because you just didn't care?	0.0052	19	34.1459	0.0000
Do you swear or use dirty language a lot?	0.0051	20	20.3407	0.0000
When was the last time you carried a weapon, such as a gun, razor, or big knife, for protection?	0.0048	21	115.8488	0.0000
Are you satisfied with your educational situation?	0.0042	22	10.3296	0.0013
Do you disturb the peace?	0.0042	23	30.5305	0.0000

When was the last time you destroyed something belong to someone else?	0.0041	24	62.1470	0.0000
At any time in the past 6 months, did you live with your spouse or with a partner?	0.0041	25	20.0053	0.0000
Do you have temper?	0.0040	26	21.1296	0.0000
Do you do things a lot without first thinking about the consequences?	0.0040	27	33.5189	0.0000
Do you prefer to date members of the opposite sex who are physically exciting or who share your values?	0.0040	28	15.8746	0.0001
Do you do things that may cause you to fail?	0.0038	29	28.8233	0.0000
Do you think heavy drinking usually ruins a party because some people get loud and boisterous or keeping the drinks full is the key to a good party?	0.0038	30	14.6957	0.0001

BIBLIOGRAPHY

(IHME), Institute for Health Metrics and Evaluation. 2019. ' GBD Compare data visualization', Accessed October 8. https://vizhub.healthdata.org/gbd-compare/.

Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, and Michael Isard. 2016. "Tensorflow: A system for large-scale machine learning." In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, 265-83.

Acion, Laura, Diana Kelmansky, Mark van der Laan, Ethan Sahker, DeShauna Jones, and Stephan Arndt. 2017. 'Use of a machine learning framework to predict substance use disorder treatment success', *PLoS ONE*, 12: e0175383.

Adenot, M., and R. Lahana. 2004. 'Blood-brain barrier permeation models: discriminating between potential CNS and non-CNS drugs including P-glycoprotein substrates', *J Chem Inf Comput Sci*, 44: 239-48.

Administration, Substance Abuse and Mental Health Services. 2020. "Key Substance Use and Mental Health Indicators in the United States: Results from the 2019 National Survey on Drug Use and Health." In.

Aliferis, Constantin F, Alexander Statnikov, Ioannis Tsamardinos, Subramani Mani, and Xenofon D Koutsoukos. 2010. 'Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: algorithms and empirical evaluation', *Journal of Machine Learning Research*, 11.

Alsenan, Shrooq, Isra Al-Turaiki, and Alaaeldin Hafez. 2020. 'A Recurrent Neural Network model to predict blood–brain barrier permeability', *Computational Biology and Chemistry*, 89: 107377.

Aronov, A. M. 2005. 'Predictive in silico modeling for hERG channel blockers', *Drug Discov Today*, 10: 149-55.

'Artificial intelligence: Google's AlphaGo beats Go master Lee Se-dol'. 2016. BCC News, Accessed May 20th, 2016.

Bagchi, Sounak, Tanya Chhibber, Behnaz Lahooti, Angela Verma, Vivek Borse, and Rahul Dev Jayant. 2019. 'In-vitro blood-brain barrier models for drug screening and permeation studies: an overview', *Drug design, development and therapy*, 13: 3591-605.

Banks, William A. 2009. 'Characteristics of compounds that cross the blood-brain barrier', *BMC Neurology*, 9: S3.

Baskin, II, D. Winkler, and I. V. Tetko. 2016. 'A renaissance of neural networks in drug discovery', *Expert Opin Drug Discov*, 11: 785-95.

Bengio, Yoshua. 2009. 'Learning deep architectures for AI', *Foundations and trends® in Machine Learning*, 2: 1-127.

Benjamini, Yoav, and Yosef Hochberg. 1995. 'Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing', *Journal of the Royal Statistical Society: Series B (Methodological)*, 57: 289-300.

Beretta, Lorenzo, and Alessandro Santaniello. 2016. 'Nearest neighbor imputation algorithms: a critical evaluation', *BMC Medical Informatics and Decision Making*, 16: 74.

Berman, HM, J Westbrook, Z Feng, G Gilliland, TN Bhat, H Weissig, IN Shindyalov, and PE Bourne. 2000. 'The protein data bank nucleic acids research, 28: 235–242', *URL: www. rcsb. org Citation*.

Biggs, Jeremy. 2019. 'Factor_analyzer documentation', *Release 0.3*, 1.

Birnbaum, H. G., A. G. White, M. Schiller, T. Waldman, J. M. Cleveland, and C. L. Roland. 2011. 'Societal Costs of Prescription Opioid Abuse, Dependence, and Misuse in the United States', *Pain Medicine*, 12: 657-67.

Bjerrum, Esben Jannik. 2017. 'SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules', *arXiv:1703.07076*.

Bors, Luca Anna, and Franciska Erdő. 2019. 'Overcoming the blood–brain barrier. challenges and tricks for CNS drug delivery', *Scientia Pharmaceutica*, 87: 6.

Borsboom, Denny. 2017. 'A network theory of mental disorders', *World Psychiatry*, 16: 5-13.

Bose, Jonaki, Sarra L. Hedden, Rachel N. Lipari, and Eunice Park-Lee. 2018. "Key Substance Use and Meental Health Indicators in the United States: Results from the 2017 National Survey on Drug Use and Health." In.: Center for Behavioral Health Statistics and Quality.

Bouaguel, Waad. 2016. "A New Approach for Wrapper Feature Selection Using Genetic Algorithm for Big Data." In *Intelligent and Evolutionary Systems*, edited by Kittichai Lavangnananda, Somnuk Phon-Amnuaisuk, Worrawat Engchuan and Jonathan H. Chan, 75-83. Cham: Springer International Publishing.

Bough, Kristopher J., and Jonathan D. Pollock. 2018. 'Defining Substance Use Disorders: The Need for Peripheral Biomarkers', *Trends in Molecular Medicine*, 24: 109-20.

Bradbury, M. W. 1993. 'The blood-brain barrier', *Exp Physiol*, 78: 453-72.

Breiman, L. 2001a. 'Random forests', *Machine Learning*, 45: 5.

Breiman, Leo. 2001b. 'Random forests', *Machine learning*, 45: 5-32.

Brooks, B. R., C. L. Brooks, 3rd, A. D. Mackerell, Jr., L. Nilsson, R. J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caflisch, L. Caves, Q. Cui, A. R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoscek, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R. W. Pastor, C. B. Post, J. Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D. M. York, and M. Karplus. 2009. 'CHARMM: the biomolecular simulation program', *J Comput Chem*, 30: 1545-614.

Browne, Michael W. 2000. 'Cross-Validation Methods', *Journal of Mathematical Psychology*, 44: 108-32.

Brunette, E. S., R. C. Flemmer, and C. L. Flemmer. 2009. "A review of artificial intelligence." In *2009 4th International Conference on Autonomous Robots and Agents*, 385-92.

Cabitza, F., R. Rasoini, and G. Gensini. 2017. 'Unintended consequences of machine learning in medicine', *JAMA*, 318: 517-18.

Cai, C. P., P. F. Guo, Y. D. Zhou, J. W. Zhou, Q. Wang, F. X. Zhang, J. S. Fang, and F. X. Cheng. 2019. 'Deep Learning-Based Prediction of Drug-Induced Cardiotoxicity', *Journal of Chemical Information and Modeling*, 59: 1073-84.

Cai, Hongyun, Vincent W Zheng, and Kevin Chen-Chuan Chang. 2018. 'A comprehensive survey of graph embedding: Problems, techniques, and applications', *IEEE Transactions on Knowledge and Data Engineering*, 30: 1616-37.

Cano, Gaspar, Jose Garcia-Rodriguez, Alberto Garcia-Garcia, Horacio Perez-Sanchez, Jón Atli Benediktsson, Anil Thapa, and Alastair Barr. 2017. 'Automatic selection of molecular descriptors using random forest: Application to drug discovery', *Expert Systems with Applications*, 72: 151-59.

Capuzzi, Stephen J., Regina Politi, Olexandr Isayev, Sherif Farag, and Alexander Tropsha. 2016. 'QSAR Modeling of Tox21 Challenge Stress Response and Nuclear Receptor Signaling Toxicity Assays', *Frontiers in Environmental Science*, 4.

Casey, W. 2013. 'Tox21 Overview and Update.', *In Vitro Cellular & Developmental Biology-Animal*, 49: S7-S8.

Casini, Lorenzo, and Michael Baumgartner. 2020. 'The PC Algorithm and the Inference to Constitution'.

Cavalli, A., E. Poluzzi, F. De Ponti, and M. Recanatini. 2002. 'Toward a pharmacophore for drugs inducing the long QT syndrome: Insights from a CoMFA study of HERG K+ channel blockers', *Journal of Medicinal Chemistry*, 45: 3844-53.

CDC. 2016. 'https://www.cdc.gov/features/costsofdrinking/ ', Centers for Disease Control and Prevention, Accessed April 21. https://www.cdc.gov/features/costsofdrinking/

Chaudhary, K. W., J. M. O'Neal, Z. L. Mo, B. Fermini, R. H. Gallavan, and A. Bahinski. 2006. 'Evaluation of the rubidium efflux assay for preclinical identification of hERG blockade', *Assay and Drug Development Technologies*, 4: 73-82.

Chen, Jonathan H, and Steven M %J The New England journal of medicine Asch. 2017. 'Machine learning and prediction in medicine—beyond the peak of inflated expectations', 376: 2507.

Chen, M., Y. Jing, L. Wang, Z. Feng, and X. Q. Xie. 2019. 'DAKB-GPCRs: An Integrated Computational Platform for Drug Abuse Related GPCRs', *Journal of Chemical Information and Modeling*, 59: 1283-89.

Chen, Yushi, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. 2014. 'Deep learning-based classification of hyperspectral data', *IEEE Journal of Selected topics in applied earth observations and remote sensing*, 7: 2094-107.

Cheng, H., A. Li, A. A. Koenigsberger, C. F. Huang, Y. Wang, J. H. Sheng, and S. D. Newman. 2017. 'Pseudo-Bootstrap Network Analysis-an Application in Functional Connectivity Fingerprinting', *Frontiers in Human Neuroscience*, 11.

Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. 'Learning phrase representations using RNN encoder-decoder for statistical machine translation', *arXiv preprint arXiv:1406.1078*.

Chothia, C., and A. M. Lesk. 1986. 'The Relation between the Divergence of Sequence and Structure in Proteins', *Embo Journal*, 5: 823-26.

Clark, D. E. 2003. 'In silico prediction of blood-brain barrier permeation', *Drug Discov Today*, 8: 927-33.

Coley, Connor W, Dale A Thomas, Justin AM Lummiss, Jonathan N Jaworski, Christopher P Breen, Victor Schultz, Travis Hart, Joshua S Fishman, Luke Rogers, and Hanyu Gao. 2019. 'A robotic platform for flow synthesis of organic compounds informed by AI planning', *Science*, 365.

Collaborators, Gbd Us Neurological Disorders, V. L. Feigin, T. Vos, F. Alahdab, A. M. L. Amit, T. W. Barnighausen, E. Beghi, M. Beheshti, P. P. Chavan, M. H. Criqui, R. Desai, S. Dhamminda Dharmaratne, E. R. Dorsey, A. Wilder Eagan, I. Y. Elgendy, I. Filip, S. Giampaoli, G. Giussani, N. Hafezi-Nejad, M. K. Hole, T. Ikeda, C. Owens Johnson, R. Kalani, K. Khatab, J. Khubchandani, D. Kim, W. J. Koroshetz, V. Krishnamoorthy, R. V. Krishnamurthi, X. Liu, W. D. Lo, G. Logroscino, G. A. Mensah, T. R. Miller, S.

Mohammed, A. H. Mokdad, M. Moradi-Lakeh, S. D. Morrison, V. K. N. Shivamurthy, M. Naghavi, E. Nichols, B. Norrving, C. M. Odell, E. Pupillo, A. Radfar, G. A. Roth, A. Shafieesabet, A. Sheikh, S. Sheikhbahaei, J. I. Shin, J. A. Singh, T. J. Steiner, L. J. Stovner, M. T. Wallin, J. Weiss, C. Wu, J. R. Zunt, J. D. Adelson, and C. J. L. Murray. 2021. 'Burden of Neurological Disorders Across the US From 1990-2017: A Global Burden of Disease Study', *JAMA Neurol*, 78: 165-76.

Conrod, Patricia J. 2016. 'Personality-Targeted Interventions for Substance Use and Misuse', *Current Addiction Reports*, 3: 426-36.

Coordinators, Ncbi Resource. 2017. 'Database resources of the National Center for Biotechnology Information', *Nucleic Acids Res*.

Cortes, C., and V. Vapnik. 1995. 'Support-Vector Networks', *Mach. Learn.*, 20: 273.

Cox, D. R. 1958. 'The Regression-Analysis of Binary Sequences', *Journal of the Royal Statistical Society Series B-Statistical Methodology*, 20: 215-42.

Crum, Rosa M, Hee-Soon Juon, Kerry M Green, Judith Robertson, Kate Fothergill, and Margaret Ensminger. 2006. 'Educational achievement and early school behavior as predictors of alcohol-use disorders: 35-year follow-up of the Woodlawn Study', *Journal of Studies on Alcohol*, 67: 75-85.

Dahl, George E, Navdeep Jaitly, and Ruslan Salakhutdinov. 2014. 'Multi-task neural networks for QSAR predictions', *arXiv preprint arXiv:1406.1231*.

Das, Sumit, Aritra Dey, Akash Pal, and Nabamita Roy. 2015. 'Applications of artificial intelligence in machine learning: review and prospect', *International Journal of Computer Applications*, 115.

'Data Science and its Relationship to Big Data and Data-Driven Decision Making'. 2013. *Big Data*, 1: 51-59.

Dealmakers, Biopharma. 2020. 'A view into the central nervous system disorders market', Nature, Accessed March 2 https://www.nature.com/articles/d43747-020-01119-8#.

Demmer, Ryan T., and Panos N. Papapanou. 2020. 'Causal Inference and Assessment of Risk in the Health Sciences.' in Iain L. C. Chapple and Panos N. Papapanou (eds.), *Risk Assessment in Oral Health: A Concise Guide for Clinical Application* (Springer International Publishing: Cham).

Ding, Chris, and Xiaofeng He. 2004. "K-nearest-neighbor consistency in data clustering: incorporating local information into global optimization." In *Proceedings of the 2004 ACM symposium on Applied computing*, 584-89.

Domingos, P., and M. Pazzani. 1997. 'On the optimality of the simple Bayesian classifier under zero-one loss', *Machine learning*, 29: 103-30.

Dong, X. 2018. 'Current Strategies for Brain Drug Delivery', *Theranostics*, 8: 1481-93.

Elisseeff, André, and Jason Weston. 2002. "A kernel method for multi-labelled classification." In *Advances in neural information processing systems*, 681-87.

Ermondi, G., S. Visentin, and G. Caron. 2009. 'GRIND-based 3D-QSAR and CoMFA to investigate topics dominated by hydrophobic interactions: The case of hERG K+ channel blockers', *European Journal of Medicinal Chemistry*, 44: 1926-32.

Ertl, P. 2010. 'Molecular structure input on the web', *Journal of Cheminformatics*, 2.

Fabrigar, Leandre R, Duane T Wegener, Robert C MacCallum, and Erin J Strahan. 1999. 'Evaluating the use of exploratory factor analysis in psychological research', *Psychological methods*, 4: 272.

Fu, Shunkai, and Michel C Desmarais. 2010. "Markov blanket based feature selection: a review of past decade." In *Proceedings of the world congress on engineering*, 321-28. Citeseer.

Gao, B., and Y. Xu. 1993. 'Univariant Approximation by Superpositions of a Sigmoidal Function', *Journal of Mathematical Analysis and Applications*, 178: 221-26.

Gardner, Matt W, and SR Dorling. 1998. 'Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences', *Atmospheric environment*, 32: 2627-36.

Gaulton, A., P. Bento, J. Chambers, E. Felix, A. Hersey, D. Mendez, J. Mosquera, P. Mutowo, M. Nowotka, and A. Leach. 2018. 'ChEMBL - encouraging deposition of drug discovery data', *Abstracts of Papers of the American Chemical Society*, 256.

Gaulton, Anna, Louisa J Bellis, A Patricia Bento, Jon Chambers, Mark Davies, Anne Hersey, Yvonne Light, Shaun McGlinchey, David Michalovich, and Bissan Al-Lazikani. 2012. 'ChEMBL: a large-scale bioactivity database for drug discovery', *Nucleic Acids Research*, 40: D1100-D07.

Geiger, Dan, Thomas Verma, and Judea Pearl. 1990. 'd-separation: From theorems to algorithms.' in, *Machine Intelligence and Pattern Recognition* (Elsevier).

Genuer, Robin, Jean-Michel Poggi, and Christine Tuleau-Malot. 2010. 'Variable selection using random forests', *Pattern Recognition Letters*, 31: 2225-36.

Ghasemi, Fahimeh, Alireza Mehridehnavi, Alfonso Pérez-Garrido, and Horacio Pérez-Sánchez. 2018. 'Neural network and deep-learning algorithms used in QSAR studies: merits and drawbacks', *Drug Discovery Today*, 23: 1784-90.

Gintant, G., P. T. Sager, and N. Stockbridge. 2016. 'Evolution of strategies to improve preclinical cardiac safety testing', *Nat Rev Drug Discov*, 15: 457-71.

Glantz, Stanton, and Bryan Slinker. 2001. *Primer of Applied Regression & Analysis of Variance, ed* (McGraw-Hill, Inc., New York).

Glymour, C., K. Zhang, and P. Spirtes. 2019a. 'Review of Causal Discovery Methods Based on Graphical Models', *Front Genet*, 10: 524.

Glymour, Clark, Kun Zhang, and Peter Spirtes. 2019b. 'Review of causal discovery methods based on graphical models', *Frontiers in genetics*, 10: 524.

Goh, G. B., N. O. Hodas, and A. Vishnu. 2017. 'Deep learning for computational chemistry', *J Comput Chem*, 38: 1291-307.

Goh, Garrett B., Charles Siegel, Abhinav Vishnu, Nathan O. Hodas, and Nathan Baker. 2017a. 'Chemception: A Deep Neural Network with Minimal Chemistry Knowledge Matches the Performance of Expert-developed QSAR/QSPR Models', *arXiv:1706.06689*.

———. 2017b. 'How Much Chemistry Does a Deep Neural Network Need to Know to Make Accurate Predictions?', *arXiv:1710.02238*.

Goldberg, Yoav, and Omer Levy. 2014. 'word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method', *arXiv preprint arXiv:1402.3722*.

Goodfellow, Ian J., Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. 'Generative Adversarial Networks', *arXiv:1406.2661*.

'Google supercharges machine learning tasks with TPU custom chip'. 2016. Google, Accessed May 20th.

Grant, B. F., F. S. Stinson, D. A. Dawson, and et al. 2004. 'Prevalence and co-occurrence of substance use disorders and independentmood and anxiety disorders: Results from the national epidemiologic survey on alcohol and relatedconditions', *Archives of General Psychiatry*, 61: 807-16.

Gray, K. A., B. Yates, R. L. Seal, M. W. Wright, and E. A. Bruford. 2015. 'Genenames.org: The HGNC Resources in 2015', *Nucleic Acids Res.*, 43: D1079.

Gribkoff, V. K., and L. K. Kaczmarek. 2017. 'The need for new approaches in CNS drug discovery: Why drugs have failed, and what can be done to improve outcomes', *Neuropharmacology*, 120: 11-19.

Group, ICH Expert Working. 2005. "S7B nonclinical evaluation of the potential for delayed ventricular repolarization (QT interval prolongation) by human pharmaceuticals." In *Int. Conf. Harmon. Tech. Requir. Regist. Pharm. Hum. Use*, 61133-34.

Guimaraes, Gabriel Lima, Benjamin Sanchez-Lengeling, Carlos Outeiral, Pedro Luis Cunha Farias, and Alán Aspuru-Guzik. 2017. 'Objective-Reinforced Generative Adversarial Networks (ORGAN) for Sequence Generation Models', *arXiv:1705.10843*.

Gupta, SP. 1989. 'QSAR studies on drugs acting at the central nervous system', *Chemical Reviews*, 89: 1765-800.

Guyon, Isabelle, and André Elisseeff. 2003. "An introduction to variable and feature selection." In, edited by Leslie Pack Kaelbling, 1157-82. US: MIT Press.

Hajian-Tilaki, Karimollah. 2013. 'Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation', *Caspian journal of internal medicine*, 4: 627-35.

Hanley, J. A., and B. J. McNeil. 1982. 'The meaning and use of the area under a receiver operating characteristic (ROC) curve', *Radiology*, 143: 29-36.

Harris, Naftali, and Mathias Drton. 2013. 'PC algorithm for nonparanormal graphical models', *Journal of Machine Learning Research*, 14.

Hart, T., and L. Xie. 2016. 'Providing data science support for systems pharmacology and its implications to drug discovery', *Expert Opin Drug Discov*, 11: 241-56.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. 'Unsupervised learning.' in, *The elements of statistical learning* (Springer).

Hausman, Daniel M, and James Woodward. 1999. 'Independence, invariance and the causal Markov condition', *The British journal for the philosophy of science*, 50: 521-83.

Heckerman, David, Dan Geiger, and David M Chickering. 1995. 'Learning Bayesian networks: The combination of knowledge and statistical data', *Machine learning*, 20: 197-243.

Hertzberg, Robert P., and Andrew J. Pope. 2000. 'High-throughput screening: new technology for the 21st century', *Current Opinion in Chemical Biology*, 4: 445-51.

Hinton, G. E., S. Osindero, and Y. W. Teh. 2006. 'A fast learning algorithm for deep belief nets', *Neural Computation*, 18: 1527.

Hinton, G. E., and R. R. Salakhutdinov. 2006. 'Reducing the Dimensionality of Data with Neural Networks', *Science*, 313: 504.

Hinton, Geoffrey E. 1984. 'Distributed representations'.

Ho, T. K. 1998. 'The random subspace method for constructing decision forests', *Ieee Transactions on Pattern Analysis and Machine Intelligence*, 20: 832-44.

Hochreiter, S. 1998. 'The vanishing gradient problem during learning recurrent neural nets and problem solutions', *International Journal of Uncertainty Fuzziness and Knowledge-Based Systems*, 6: 107-16.

Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. 'Long short-term memory', *Neural Computation*, 9: 1735-80.

Holzinger, Andreas, Georg Langs, Helmut Denk, Kurt Zatloukal, and Heimo Müller. 2019. 'Causability and explainability of artificial intelligence in medicine', *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9: e1312.

Hopkins, A. L. 2008. 'Network pharmacology: the next paradigm in drug discovery', *Nat Chem Biol*, 4: 682-90.

Horner, Michelle S., Maureen Reynolds, Betty Braxter, Levent Kirisci, and Ralph E. Tarter. 2015. 'Temperament disturbances measured in infancy progress to substance use disorder 20 years later', *Personality and individual differences*, 82: 96-101.

Horrobin, David F. 2001. 'Realism in drug discovery—could Cassandra be right?', *Nature biotechnology*, 19: 1099-100.

Houtmann, S., B. Schombert, C. Sanson, M. Partiseti, and G. A. Bohme. 2017. 'Automated Patch-Clamp Methods for the hERG Cardiac Potassium Channel', *Methods Mol Biol*, 1641: 187-99.

Hu, Z., Y. Jing, Y. Xue, P. Fan, L. Wang, M. Vanyukov, L. Kirisci, J. Wang, R. E. Tarter, and X. Q. Xie. 2020. 'Analysis of substance use and its outcomes by machine learning: II. Derivation and prediction of the trajectory of substance use severity', *Drug Alcohol Depend*, 206: 107604.

Huang, Kexin, Tianfan Fu, Lucas M Glass, Marinka Zitnik, Cao Xiao, and Jimeng Sun. 2020. 'DeepPurpose: a deep learning library for drug–target interaction prediction', *Bioinformatics*, 36: 5545-47.

Hubel, David H, and Torsten N Wiesel. 1959. 'Receptive fields of single neurones in the cat's striate cortex', *The Journal of physiology*, 148: 574-91.

———. 1962. 'Receptive fields, binocular interaction and functional architecture in the cat's visual cortex', *The Journal of physiology*, 160: 106-54.

Iacono, W. G., S. R. Carlson, J. Taylor, I. J. Elkins, and M. McGue. 1999. 'Behavioral disinhibition and the development of substance-case disorders: Findings from the Minnesota Twin Family Study', *Development and Psychopathology*, 11: 869-900.

Iorio, F., R. Bosotti, E. Scacheri, V. Belcastro, P. Mithbaokar, R. Ferriero, L. Murino, R. Tagliaferri, N. Brunetti-Pierri, A. Isacchi, and D. di Bernardo. 2010. 'Discovery of drug mode of action and drug repositioning from transcriptional responses', *Proc Natl Acad Sci U S A*, 107: 14621-6.

Isberg, V., S. Mordalski, C. Munk, K. Rataj, K. Harpsoe, A. S. Hauser, B. Vroling, A. J. Bojarski, G. Vriend, and D. E. Gloriam. 2017. 'GPCRdb: an information system for G protein-coupled receptors', *Nucleic Acids Res*, 45: 2936.

Iskar, M., G. Zeller, P. Blattmann, M. Campillos, M. Kuhn, K. H. Kaminska, H. Runz, A. C. Gavin, R. Pepperkok, V. van Noort, and P. Bork. 2013. 'Characterization of Drug-Induced Transcriptional Modules: Towards Drug Repositioning and Functional Understanding', *Mol. Syst. Biol.*, 9: 662.

Jang, Seon-Kyeong, Gretchen Saunders, MengZhen Liu, Yu Jiang, Dajiang J. Liu, and Scott Vrieze. 2020. 'Genetic correlation, pleiotropy, and causal associations between substance use and psychiatric disorder', *Psychological Medicine*: 1-11.

Jing, Y., Y. Bian, Z. Hu, L. Wang, and X. Q. Xie. 2018a. 'Correction to: Deep Learning for Drug Design: an Artificial Intelligence Paradigm for Drug Discovery in the Big Data Era', *AAPS J*, 20: 79.

Jing, Y., Z. Hu, P. Fan, Y. Xue, L. Wang, R. E. Tarter, L. Kirisci, J. Wang, M. Vanyukov, and X. Q. Xie. 2020. 'Analysis of substance use and its outcomes by machine learning I.

Childhood evaluation of liability to substance use disorder', *Drug Alcohol Depend*, 206: 107605.

Jing, Y. K., A. Easter, D. Peters, N. Kim, and I. J. Enyedy. 2015. 'In silico prediction of hERG inhibition', *Future Medicinal Chemistry*, 7: 571-86.

Jing, Yankang, Yuemin Bian, Ziheng Hu, Lirong Wang, and Xiang-Qun Xie. 2018b. 'Correction to: Deep Learning for Drug Design: an Artificial Intelligence Paradigm for Drug Discovery in the Big Data Era', *The AAPS Journal*, 20: 79.

Jing, Yankang, Yuemin Bian, Ziheng Hu, Lirong Wang, and Xiang-Qun Sean Xie. 2018c. 'Deep Learning for Drug Design: an Artificial Intelligence Paradigm for Drug Discovery in the Big Data Era', *The AAPS Journal*, 20: 58.

Jones, P. J., A. Heeren, and R. J. McNally. 2017. 'Commentary: A network theory of mental disorders', *Front Psychol*, 8: 1305.

Jordan, M. I., and T. M. Mitchell. 2015. 'Machine learning: Trends, perspectives, and prospects', *Science*, 349: 255-60.

Jumper, John, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. 2021. 'Highly accurate protein structure prediction with AlphaFold', *Nature*, 596: 583-89.

Kadurin, Artur, Alexander Aliper, Andrey Kazennov, Polina Mamoshina, Quentin Vanhaelen, Kuzma Khrabrov, and Alex Zhavoronkov. 2017. 'The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology', *Oncotarget*, 8: 10883.

Kalé, Laxmikant, Robert Skeel, Milind Bhandarkar, Robert Brunner, Attila Gursoy, Neal Krawetz, James Phillips, Aritomo Shinozaki, Krishnan Varadarajan, and Klaus Schulten. 1999. 'NAMD2: Greater Scalability for Parallel Molecular Dynamics', *Journal of Computational Physics*, 151: 283-312.

Kalyaanamoorthy, S., and K. H. Barakat. 2018. 'Development of Safe Drugs: The hERG Challenge', *Medicinal Research Reviews*, 38: 525-55.

Kanehisa, M., and S. Goto. 2000. 'KEGG: kyoto encyclopedia of genes and genomes', *Nucleic Acids Res*, 28: 27-30.

Kearnes, Steven, Kevin McCloskey, Marc Berndl, Vijay Pande, and Patrick Riley. 2016. 'Molecular graph convolutions: moving beyond fingerprints', *Journal of computer-aided molecular design*, 30: 595-608.

Kelder, J., P. D. Grootenhuis, D. M. Bayada, L. P. Delbressine, and J. P. Ploemen. 1999. 'Polar molecular surface as a dominating determinant for oral absorption and brain penetration of drugs', *Pharm Res*, 16: 1514-9.

Kelley, Henry J. 1960. 'Gradient theory of optimal flight paths', *Ars Journal*, 30: 947-54.

Kenter, Tom, Alexey Borisov, and Maarten De Rijke. 2016. 'Siamese cbow: Optimizing word embeddings for sentence representations', *arXiv preprint arXiv:1606.04640*.

Kerns, Edward H., and Li Di. 2008. 'Chapter 28 - Blood-Brain Barrier Methods.' in Edward H. Kerns and Li Di (eds.), *Drug-like Properties: Concepts, Structure Design and Methods* (Academic Press: San Diego).

Khanna, Ish. 2012. 'Drug discovery in pharmaceutical industry: productivity challenges and trends', *Drug Discovery Today*, 17: 1088-102.

Kim, Sunghwan, Paul A. Thiessen, Evan E. Bolton, Jie Chen, Gang Fu, Asta Gindulyte, Lianyi Han, Jane He, Siqian He, Benjamin A. Shoemaker, Jiyao Wang, Bo Yu, Jian Zhang, and Stephen H. Bryant. 2016. 'PubChem Substance and Compound databases', *Nucleic Acids Research*, 44: D1202-D13.

King, S. M., W. G. Iacono, and M. McGue. 2004a. 'Childhood externalizing and internalizing psychopathology in the prediction of early substance use', *Addiction*, 99: 1548-59.

———. 2004b. 'Childhood externalizing and internalizing psychopathology in the prediction of early substance use', *Addiction*, 99: 1548-59.

Kirisci, Levent, Ralph Tarter, Ada Mezzich, Ty Ridenour, Maureen Reynolds, and Michael Vanyukov. 2009. 'Prediction of Cannabis Use Disorder between Boyhood and Young Adulthood: Clarifying the Phenotype and Environtype', *The American Journal on Addictions*, 18: 36-47.

Kitchen, D. B., H. Decornez, J. R. Furr, and J. Bajorath. 2004. 'Docking and scoring in virtual screening for drug discovery: Methods and applications', *Nature Reviews Drug Discovery*, 3: 935-49.

Kleinbaum, David G, K Dietz, M Gail, Mitchel Klein, and Mitchell Klein. 2002. *Logistic regression* (Springer).

Konda, Leela Sarath Kumar, S. Keerthi Praba, and Rajendra Kristam. 2019. 'hERG liability classification models using machine learning techniques', *Computational Toxicology*, 12: 100089.

Kortagere, Sandhya, Dmitriy Chekmarev, William J Welsh, and Sean Ekins. 2008. 'New predictive models for blood–brain barrier permeability of drug-like molecules', *Pharmaceutical research*, 25: 1836-45.

Kotsiantis, Sotiris B, I Zaharakis, and P %J Emerging artificial intelligence applications in computer engineering Pintelas. 2007. 'Supervised machine learning: A review of classification techniques', 160: 3-24.

Kramer, A., J. Green, J. Pollard, Jr., and S. Tugendreich. 2014. 'Causal analysis approaches in Ingenuity Pathway Analysis', *Bioinformatics*, 30: 523-30.

Krueger, Robert F., and Kristian E. Markon. 2006. 'Understanding Psychopathology: Melding Behavior Genetics, Personality, and Quantitative Psychology to Develop an Empirically Based Model', *Current directions in psychological science*, 15: 113-17.

Krueger, Robert F., Kristian E. Markon, Christopher J. Patrick, and William G. Iacono. 2005. 'Externalizing psychopathology in adulthood: a dimensional-spectrum conceptualization and its implications for DSM-V', *Journal of abnormal psychology*, 114: 537-50.

Kuang, Kun, Lian Li, Zhi Geng, Lei Xu, Kun Zhang, Beishui Liao, Huaxin Huang, Peng Ding, Wang Miao, and Zhichao Jiang. 2020. 'Causal Inference', *Engineering*.

Kunwittaya, Sarun, Chanin Nantasenamat, Lertyot Treeratanapiboon, Apapan Srisarin, Chartchalerm Isarankura-Na-Ayudhya, and Virapong Prachayasittikul. 2013. 'Influence of logBB cut-off on the prediction of blood-brain barrier permeability', *Biomedical and Applied Technology Journal*, 1: 16-34.

Landrum, Greg. 2013. "RDKit: A software suite for cheminformatics, computational chemistry, and predictive modeling." In.: Academic Press.

Lawrence, S., and C. L. Giles. 2000. "Overfitting and neural networks: conjugate gradient and backpropagation." In *Proceedings of the IEEE-INNS-ENNS International Joint*

Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium, 114-19 vol.1.

LeCun, Y., Y. Bengio, and G. Hinton. 2015. 'Deep Learning', *Nature*, 521: 436.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. 'Gradient-based learning applied to document recognition', *Proceedings of the IEEE*, 86: 2278-324.

LeCun, Yann, LD Jackel, Leon Bottou, A Brunot, Corinna Cortes, JS Denker, Harris Drucker, I Guyon, UA Muller, and Eduard Sackinger. 1995. "Comparison of learning algorithms for handwritten digit recognition." In *International conference on artificial neural networks*, 53-60. Perth, Australia.

Lenselink, E. B., N. Ten Dijke, B. Bongers, G. Papadatos, H. W. T. van Vlijmen, W. Kowalczyk, I. Jzerman AP, and G. J. P. van Westen. 2017. 'Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set', *J Cheminform*, 9: 45.

Leo, Albert, Corwin Hansch, and David Elkins. 1971. 'Partition coefficients and their uses', *Chemical Reviews*, 71: 525-616.

Li, H., C. W. Yap, C. Y. Ung, Y. Xue, Z. W. Cao, and Y. Z. Chen. 2005. 'Effect of selection of molecular descriptors on the prediction of blood-brain barrier penetrating and nonpenetrating agents by statistical learning methods', *Journal of Chemical Information and Modeling*, 45: 1376-84.

Li, Yuxi. 2017. 'Deep reinforcement learning: An overview', *arXiv preprint arXiv:1701.07274*.

Lingineni, K., V. Belekar, S. R. Tangadpalliwar, and P. Garg. 2017. 'The role of multidrug resistance protein (MRP-1) as an active efflux transporter on blood-brain barrier (BBB) permeability', *Mol Divers*, 21: 355-65.

Lipari, R. N., R. D. Ahrnsbrak, M. R. Pemberton, and J. D. Porter. 2017. 'Risk and Protective Factors and Estimates of Substance Use Initiation: Results from the 2016 National Survey on Drug Use and Health.' in, *CBHSQ Data Review* (Substance Abuse and Mental Health Services Administration (US): Rockville (MD)).

Lippmann, R. P. 1989. 'Pattern classification using neural networks', *IEEE Communications Magazine*, 27: 47-64.

Liu, H. B., L. R. Wang, M. L. Lv, R. R. Pei, P. B. Li, Z. Pei, Y. G. Wang, W. W. Su, and X. Q. Xie. 2014. 'AlzPlatform: An Alzheimer's Disease Domain-Specific Chemogenornics Knowledgebase for Polypharmacology and Target Identification Research', *Journal of Chemical Information and Modeling*, 54: 1050-60.

Liu, Huan, and Zheng Zhao. 2012. 'Manipulating Data and Dimension Reduction Methods: Feature Selection.' in Robert A. Meyers (ed.), *Computational Complexity: Theory, Techniques, and Applications* (Springer New York: New York, NY).

Liu, K., X. Sun, L. Jia, J. Ma, H. Xing, J. Wu, H. Gao, Y. Sun, F. Boulnois, and J. Fan. 2019. 'Chemi-Net: A Molecular Graph Convolutional Network for Accurate Drug Property Prediction', *Int J Mol Sci*, 20.

Liu, Miao, Li Zhang, Shimeng Li, Tianzhou Yang, Lili Liu, Jian Zhao, and Hongsheng Liu. 2020. 'Prediction of hERG potassium channel blockage using ensemble learning methods and molecular fingerprints', *Toxicology Letters*, 332: 88-96.

Lu, J., D. Lu, Z. Fu, M. Zheng, and X. Luo. 2018. 'Machine Learning-Based Modeling of Drug Toxicity', *Methods Mol Biol*, 1754: 247-64.

Lungarella, Max, Fumiya Iida, Josh Bongard, and Rolf Pfeifer. 2007. *50 Years of Artificial Intelligence: Essays Dedicated to the 50th Anniversary of Artificial Intelligence* (Springer).

Lusci, Alessandro, Gianluca Pollastri, and Pierre Baldi. 2013. 'Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-Like Molecules', *Journal of Chemical Information and Modeling*, 53: 1563-75.

Ma, Chao, Lirong Wang, and Xiang-Qun Xie. 2011a. 'GPU Accelerated Chemical Similarity Calculation for Compound Library Comparison', *Journal of Chemical Information and Modeling*, 51: 1521-27.

———. 2011b. 'Ligand classifier of adaptively boosting ensemble decision stumps (LiCABEDS) and its application on modeling ligand functionality for 5HT-subtype GPCR families', *Journal of Chemical Information and Modeling*, 51: 521-31.

Ma, Chao, Lirong Wang, Peng Yang, Kyaw Z. Myint, and Xiang-Qun Xie. 2013. 'LiCABEDS II. Modeling of Ligand Selectivity for G-protein Coupled Cannabinoid Receptors', *Journal of Chemical Information and Modeling*, 53: 11-26.

Ma, Junshui, Robert P. Sheridan, Andy Liaw, George E. Dahl, and Vladimir Svetnik. 2015. 'Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships', *Journal of Chemical Information and Modeling*, 55: 263-74.

Magerman, David M. 1995. "Statistical decision-tree models for parsing." In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, 276-83. Association for Computational Linguistics.

Maggiora, G. M. 2006. 'On outliers and activity cliffs--why QSAR often disappoints', *Journal of Chemical Information and Modeling*, 46: 1535.

Mahesh, Batta. 2020. 'Machine Learning Algorithms-A Review', *International Journal of Science and Research (IJSR).[Internet]*, 9: 381-86.

Małkiewicz, Marta A., Arkadiusz Szarmach, Agnieszka Sabisz, Wiesław J. Cubała, Edyta Szurowska, and Paweł J. Winklewski. 2019. 'Blood-brain barrier permeability and physical exercise', *Journal of Neuroinflammation*, 16: 15.

McCulloch, Warren S, and Walter Pitts. 1943. 'A logical calculus of the ideas immanent in nervous activity', *The bulletin of mathematical biophysics*, 5: 115-33.

McGue, M., I. Elkins, and W. G. Iacono. 2000. 'Genetic and environmental influences on adolescent substance use and abuse', *American Journal of Medical Genetics*, 96: 671-77.

McNally, R. J. 2016. 'Can network analysis transform psychopathology?', *Behav Res Ther*, 86: 95-104.

Metz, Charles E. 1978. 'Basic principles of ROC analysis', *Seminars in Nuclear Medicine*, 8: 283-98.

Mezzich, Ada C, Ralph E Tarter, Peter R Giancola, and Levent Kirisci. 2001. 'The dysregulation inventory: A new scale to assess the risk for substance use disorder', *Journal of Child & Adolescent Substance Abuse*, 10: 35-43.

Miao, Rui, Liang-Yong Xia, Hao-Heng Chen, Hai-Hui Huang, and Yong Liang. 2019. 'Improved Classification of Blood-Brain-Barrier Drugs Using Deep Learning', *Scientific Reports*, 9: 8802.

Mohri, Mehryar, Afshin Rostamizadeh, and Ameet Talwalkar. 2018. *Foundations of machine learning* (MIT press).

Myint, Kyaw-Zeyar, Lirong Wang, Qin Tong, and Xiang-Qun Xie. 2012. 'Molecular fingerprint-based artificial neural networks QSAR for ligand biological activity predictions', *Molecular pharmaceutics*, 9: 2912-23.

Myint, Kyaw Z., and Xiang-Qun Xie. 2015. 'Ligand Biological Activity Predictions Using Fingerprint-Based Artificial Neural Networks (FANN-QSAR)', *Methods in molecular biology (Clifton, N.J.)*, 1260: 149-64.

Mysinger, Michael M., Michael Carchia, John J. Irwin, and Brian K. Shoichet. 2012. 'Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking', *Journal of Medicinal Chemistry*, 55: 6582-94.

Nair, Vinod, and Geoffrey E Hinton. 2010. "Rectified linear units improve restricted boltzmann machines." In *Proceedings of the 27th international conference on machine learning (ICML-10)*, 807-14.

National Drug Intelligence Center (NDIC). 2017. "National Drug Threat Assessment." In.

Nelder, John Ashworth, and Robert WM Wedderburn. 1972. 'Generalized linear models', *Journal of the Royal Statistical Society: Series A (General)*, 135: 370-84.

Nelson, Melissa C., and Penny Gordon-Larsen. 2006. 'Physical Activity and Sedentary Behavior Patterns Are Associated With Selected Adolescent Health Risk Behaviors', *Pediatrics*, 117: 1281.

NIDA. 2020. 'Costs of Substance Abuse', Accessed 2 July. https://www.drugabuse.gov/drug-topics/trends-statistics/costs-substance-abuse.

Nogueira, Ana Rita, João Gama, and Carlos Abreu Ferreira. 2020. "Improving prediction with causal probabilistic variables." In *International Symposium on Intelligent Data Analysis*, 379-90. Springer.

Norris, M., and L. Lecavalier. 2010. 'Evaluating the use of exploratory factor analysis in developmental disability psychological research', *J Autism Dev Disord*, 40: 8-20.

O'Boyle, Noel M., Michael Banck, Craig A. James, Chris Morley, Tim Vandermeersch, and Geoffrey R. Hutchison. 2011. 'Open Babel: An open chemical toolbox', *Journal of Cheminformatics*, 3: 33.

Ogarrio, Juan Miguel, Peter Spirtes, and Joe Ramsey. 2016. "A hybrid causal search algorithm for latent variable models." In *Conference on Probabilistic Graphical Models*, 368-79.

Ohlsson, Henrik, and Kenneth S Kendler. 2019. 'Applying causal inference methods in psychiatric epidemiology: A review', *JAMA psychiatry*.

Olivecrona, Marcus, Thomas Blaschke, Ola Engkvist, and Hongming Chen. 2017. 'Molecular De Novo Design through Deep Reinforcement Learning', *arXiv:1704.07555*.

Olurotimi, O. 1994. 'Recurrent neural network training with feedforward complexity', *IEEE Trans Neural Netw*, 5: 185-97.

Organization, World Health, and World Health Organization. Management of Substance Abuse Unit. 2014. *Global status report on alcohol and health, 2014* (World Health Organization).

Orvaschel, Helen, and Joaquim Puig-Antich. 1987. *Schedule for affective disorder and schizophrenia for school-age children: Epidemiologic version: Kiddie-SADS-E (K-SADS-E)*.

Österberg, Thomas, and Ulf Norinder. 2000. 'Prediction of Polar Surface Area and Drug Transport Processes Using Simple Parameters and PLS Statistics', *Journal of Chemical Information and Computer Sciences*, 40: 1408-11.

Ottaviani, Giorgio, Sophie Martel, and Pierre-Alain Carrupt. 2006. 'Parallel Artificial Membrane Permeability Assay: A New Membrane for the Fast Prediction of Passive Human Skin Permeability', *Journal of Medicinal Chemistry*, 49: 3948-54.

Pajouhesh, Hassan, and George R. Lenz. 2005. 'Medicinal chemical properties of successful central nervous system drugs', *NeuroRX*, 2: 541-53.

Pardridge, William M. 1998. 'CNS drug design based on principles of blood-brain barrier transport', *Journal of neurochemistry*, 70: 1781-92.

Pardridge, William M. 2005. 'The blood-brain barrier: Bottleneck in brain drug development', *NeuroRX*, 2: 3-14.

Pastur-Romay, A. Lucas, Francisco Cedrón, Alejandro Pazos, and B. Ana Porto-Pazos. 2016. 'Deep Artificial Neural Networks and Neuromorphic Chips for Big Data Analysis: Pharmaceutical and Bioinformatics Applications', *International Journal of Molecular Sciences*, 17.

Patrick, Edward A, and Frederick P Fischer III. 1970. 'A generalized k-nearest neighbor rule', *Information and control*, 16: 128-52.

Pearl, Judea. 1995. 'Causal diagrams for empirical research', *Biometrika*, 82: 669-88.

———. 1998. 'Graphs, causality, and structural equation models', *Sociological Methods & Research*, 27: 226-84.

———. 2019. 'The seven tools of causal inference, with reflections on machine learning', *Communications of the ACM*, 62: 54-60.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Vincent Dubourg. 2011a. 'Scikit-learn: Machine learning in Python', *the Journal of machine Learning research*, 12: 2825-30.

———. 2011b. 'Scikit-learn: Machine learning in Python', *Journal of machine learning research*, 12: 2825-30.

Pereira, J. C., E. R. Caffarena, and C. N. Dos Santos. 2016. 'Boosting Docking-Based Virtual Screening with Deep Learning', *Journal of Chemical Information and Modeling*, 56: 2495-506.

Piper, D. R., S. R. Duff, H. C. Eliason, W. J. Frazee, E. A. Frey, M. Fuerstenau-Sharp, C. Jachec, B. D. Marks, B. A. Pollok, M. S. Shekhani, D. V. Thompson, P. Whitney, K. W. Vogel, and S. D. Hess. 2008. 'Development of the predictor hERG fluorescence polarization assay using a membrane protein enrichment approach', *Assay and Drug Development Technologies*, 6: 213-23.

Plisson, Fabien, and Andrew M. Piggott. 2019. 'Predicting Blood⁻Brain Barrier Permeability of Marine-Derived Kinase Inhibitors Using Ensemble Classifiers Reveals Potential Hits for Neurodegenerative Disorders', *Marine drugs*, 17: 81.

Polton, D. J. 1982. 'Installation and Operational Experiences with Maccs (Molecular Access System)', *Online Review*, 6: 235-42.

Praveena, M, and V Jaiganesh. 2017. 'A literature review on supervised machine learning algorithms and boosting process', *International Journal of Computer Applications*, 169: 32-35.

Prosperi, Mattia, Yi Guo, Matt Sperrin, James S. Koopman, Jae S. Min, Xing He, Shannan Rich, Mo Wang, Iain E. Buchan, and Jiang Bian. 2020. 'Causal inference and counterfactual prediction in machine learning for actionable healthcare', *Nature Machine Intelligence*, 2: 369-75.

Pugsley, M. K., and M. J. Curtis. 2006. 'Safety pharmacology in focus: new methods developed in the light of the ICH S7B guidance document', *J Pharmacol Toxicol Methods*, 54: 94-8.

Quinlan, J Ross. 1996. "Bagging, boosting, and C4. 5." In *Aaai/iaai, Vol. 1*, 725-30.

Raghib, H., M. J. Stebbing, and H. Majewski. 2006. 'Validation of established non-animal HERG testing system using a rubidium assay.', *Acta Pharmacologica Sinica*, 27: 247-47.

Ramsey, J., M. Glymour, R. Sanchez-Romero, and C. Glymour. 2017. 'A million variables and more: the Fast Greedy Equivalence Search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images', *Int J Data Sci Anal*, 3: 121-29.

Ramsey, Joseph D, Kun Zhang, Madelyn Glymour, Ruben Sanchez Romero, Biwei Huang, Imme Ebert-Uphoff, Savini Samarasinghe, Elizabeth A Barnes, and Clark Glymour. 2018. "TETRAD—A toolbox for causal discovery." In *8th International Workshop on Climate Informatics*.

Ramsundar, Bharath, Steven Kearnes, Patrick Riley, Dale Webster, David Konerding, and Vijay Pande. 2015. 'Massively multitask networks for drug discovery', *arXiv preprint arXiv:1502.02072*.

Redzic, Zoran. 2011. 'Molecular biology of the blood-brain and the blood-cerebrospinal fluid barriers: similarities and differences', *Fluids and Barriers of the CNS*, 8: 3.

Release, Schrödinger. 2017. '2: LigPrep, Schrödinger, LLC, New York, NY, 2017', *New York, NY*.

Rhemtulla, M., E. I. Fried, S. H. Aggen, F. Tuerlinckx, K. S. Kendler, and D. Borsboom. 2016. 'Network analysis of substance abuse and dependence symptoms', *Drug Alcohol Depend*, 161: 230-7.

Richens, Jonathan G, Ciarán M Lee, and Saurabh Johri. 2020. 'Improving the accuracy of medical diagnosis with causal machine learning', *Nature communications*, 11: 1-9.

Rish, Irina. 2001. "An empirical study of the naive Bayes classifier." In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, 41-46.

Rogers, David, and Mathew Hahn. 2010. 'Extended-connectivity fingerprints', *Journal of Chemical Information and Modeling*, 50: 742-54.

Rosenblatt, Frank. 1957. *The perceptron, a perceiving and recognizing automaton Project Para* (Cornell Aeronautical Laboratory).

Roy, Dipankar, Vijaya Kumar Hinge, and Andriy Kovalenko. 2019. 'To Pass or Not To Pass: Predicting the Blood–Brain Barrier Permeability with the 3D-RISM-KH Molecular Solvation Theory', *ACS Omega*, 4: 16774-80.

Roy, Kunal, Supratik Kar, and Rudra Narayan Das. 2015. *A primer on QSAR/QSPR modeling: fundamental concepts* (Springer).

Rubio, Doris McGartland, Ellie E. Schoenbaum, Linda S. Lee, David E. Schteingart, Paul R. Marantz, Karl E. Anderson, Lauren Dewey Platt, Adriana Baez, and Karin Esposito. 2010. 'Defining Translational Research: Implications for Training', *Academic medicine : journal of the Association of American Medical Colleges*, 85: 470-75.

Rudin, Cynthia. 2019. 'Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead', *Nature Machine Intelligence*, 1: 206-15.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams. 1986. 'Learning representations by back-propagating errors', *Nature*, 323: 533.

Ryu, Jae Yong, Mi Young Lee, Jeong Hyun Lee, Byung Ho Lee, and Kwang-Seok Oh. 2020. 'DeepHIT: a deep learning framework for prediction of hERG-induced cardiotoxicity', *Bioinformatics*, 36: 3049-55.

Sachs, Karen, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. 2005. 'Causal protein-signaling networks derived from multiparameter single-cell data', *Science*, 308: 523-29.

Saeys, Yvan, Thomas Abeel, and Yves Van de Peer. 2008. "Robust Feature Selection Using Ensemble Feature Selection Techniques." In *Machine Learning and Knowledge Discovery in Databases*, edited by Walter Daelemans, Bart Goethals and Katharina Morik, 313-25. Berlin, Heidelberg: Springer Berlin Heidelberg.

Sakai, Miyuki, Kazuki Nagayasu, Norihiro Shibui, Chihiro Andoh, Kaito Takayama, Hisashi Shirakawa, and Shuji Kaneko. 2021. 'Prediction of pharmacological activities from chemical structures with graph convolutional neural networks', *Scientific Reports*, 11: 525.

Saxena, D., A. Sharma, M. H. Siddiqui, and R. Kumar. 2019. 'Blood Brain Barrier Permeability Prediction Using Machine Learning Techniques: An Update', *Curr Pharm Biotechnol*, 20: 1163-71.

Schadt, Eric E, John Lamb, Xia Yang, Jun Zhu, Steve Edwards, Debraj GuhaThakurta, Solveig K Sieberts, Stephanie Monks, Marc Reitman, and Chunsheng Zhang. 2005. 'An integrative genomics approach to infer causal associations between gene expression and disease', *Nature genetics*, 37: 710-17.

Schmidhuber, Jürgen. 2015. 'Deep learning in neural networks: An overview', *Neural Networks*, 61: 85-117.

Schulenberg, John, Megan E. Patrick, Julie Maslowsky, and Jennifer L. Maggs. 2014. 'The Epidemiology and Etiology of Adolescent Substance Use in Developmental Perspective.' in Michael Lewis and Karen D. Rudolph (eds.), *Handbook of Developmental Psychopathology* (Springer US: Boston, MA).

Seabold, Skipper, and Josef Perktold. 2010. "Statsmodels: Econometric and statistical modeling with python." In *Proceedings of the 9th Python in Science Conference*, 61. Austin, TX.

Segler, Marwin H. S., Thierry Kogej, Christian Tyrchan, and Mark P. Waller. 2017. 'Generating Focussed Molecule Libraries for Drug Discovery with Recurrent Neural Networks', *arXiv:1701.01329*.

Sellwood, Matthew A, Mohamed Ahmed, Marwin HS Segler, and Nathan Brown. 2018. 'Artificial intelligence in drug discovery', *Future Medicinal Chemistry*, 10: 2025-28.

Services, U.S. Department of Health and Human. "The Health Consequences of Smoking—50 Years of Progress: A Report of the Surgeon General." In.

Shannon, C. E. 1948. 'A Mathematical Theory of Communication', *Bell System Technical Journal*, 27: 623-56.

Shao, Jun. 1993. 'Linear Model Selection by Cross-Validation', *Journal of the American Statistical Association*, 88: 486-94.

Sharif, Y., F. Jumah, L. Coplan, A. Krosser, K. Sharif, and R. S. Tubbs. 2018. 'Blood brain barrier: A review of its anatomy and physiology in health and disease', *Clin Anat*, 31: 812-23.

Sharifi, Mohsen, Dan Buzatu, Stephen Harris, and Jon Wilkes. 2017. 'Development of models for predicting Torsade de Pointes cardiac arrhythmias using perceptron neural networks', *BMC bioinformatics*, 18: 497-97.

Shin, Moonshik, Dongjin Jang, Hojung Nam, Kwang Hyung Lee, and Doheon Lee. 2016. 'Predicting the Absorption Potential of Chemical Compounds through a Deep Learning Approach', *IEEE/ACM transactions on computational biology and bioinformatics*.

Shobha, Gangadhar, and Shanta Rangaswamy. 2018. 'Machine learning.' in, *Handbook of statistics* (Elsevier).

Shuker, S. B., P. J. Hajduk, R. P. Meadows, and S. W. Fesik. 1996. 'Discovering high-affinity ligands for proteins: SAR by NMR', *Science*, 274: 1531-4.

Si, S, CJ Hsieh, and I Dhillon. 2014. 'Proceedings of The 31st International Conference on Machine Learning'.

Silver, David, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, and Marc Lanctot. 2016. 'Mastering the game of Go with deep neural networks and tree search', *Nature*, 529: 484-89.

Singh, Manvi, Reshmi Divakaran, Leela Sarath Kumar Konda, and Rajendra Kristam. 2020. 'A classification model for blood brain barrier penetration', *Journal of Molecular Graphics and Modelling*, 96: 107516.

Sinoquet, Christine. 2014. *Probabilistic graphical models for genetics, genomics, and postgenomics* (OUP Oxford).

Snoek, Jasper, Hugo Larochelle, and Ryan P Adams. 2012. 'Practical bayesian optimization of machine learning algorithms', *arXiv preprint arXiv:1206.2944*.

Solomatine, Dimitri P, and Durga L Shrestha. 2004. "AdaBoost. RT: a boosting algorithm for regression problems." In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*, 1163-68. IEEE.

Soman, KP, R Loganathan, and V Ajay. 2009. *Machine learning with SVM and other kernel methods* (PHI Learning Pvt. Ltd.).

Spirtes, Pater, Clark Glymour, Richard Scheines, Stuart Kauffman, Valerio Aimale, and Frank Wimberly. 2000. 'Constructing Bayesian network models of gene expression networks from microarray data'.

Spirtes, Peter. 2001. "An Anytime Algorithm for Causal Inference." In *AISTATS*.

Spirtes, Peter, Clark N Glymour, Richard Scheines, and David Heckerman. 2000. *Causation, prediction, and search* (MIT press).

Spitzer, Robert L, Janet BW Williams, Miriam Gibbon, and Michael B %J Archives of general psychiatry First. 1992. 'The structured clinical interview for DSM-III-R (SCID): I: history, rationale, and description', 49: 624-29.

Steinwart, Ingo, and Andreas Christmann. 2008. *Support vector machines* (Springer Science & Business Media).

Sterling, Teague, and John J. Irwin. 2015. 'ZINC 15 – Ligand Discovery for Everyone', *Journal of Chemical Information and Modeling*, 55: 2324-37.

Stewart, M., and P. Willett. 1987. 'Nearest neighbor searching in binary search trees. Simulation of a multiprocessor system', *Journal of Documentation*, 43: 93-111.

Stockbridge, N., J. Morganroth, R. R. Shah, and C. Garnett. 2013. 'Dealing with Global Safety Issues Was the Response to QT-Liability of Non-Cardiac Drugs Well Coordinated?', *Drug Safety*, 36: 167-82.

Svetnik, V., A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston. 2003. 'Random forest: a classification and regression tool for compound classification and QSAR modeling', *J. Chem. Inf. Comput. Sci.*, 43: 1947.

Sweeney, M. D., Z. Zhao, A. Montagne, A. R. Nelson, and B. V. Zlokovic. 2019. 'Blood-Brain Barrier: From Physiology to Disease and Back', *Physiol Rev*, 99: 21-78.

"SYBYL-X." In. 2010. Tripos International.

Tanveer, M, B Richhariya, RU Khan, AH Rashid, P Khanna, M Prasad, and CT Lin. 2020. 'Machine learning techniques for the diagnosis of Alzheimer's disease: A review', *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16: 1-35.

Tarter, Ralph E., Levent Kirisci, Ada Mezzich, Jack R. Cornelius, Kathleen Pajer, Michael Vanyukov, William Gardner, Timothy Blackson, and Duncan Clark. 2003. 'Neurobehavioral Disinhibition in Childhood Predicts Early Age at Onset of Substance Use Disorder', *American Journal of Psychiatry*, 160: 1078-85.

'Tetrad Manual'. 2019. Accessed July 2nd. http://cmu-phil.github.io/tetrad/manual/.

The UniProt, Consortium. 2017. 'UniProt: the universal protein knowledgebase', *Nucleic Acids Res*, 45: D158-D69.

Triantafillou, Sofia, and Ioannis Tsamardinos. 2016. "Score-based vs Constraint-based Causal Learning in the Presence of Confounders." In *CFA@ UAI*, 59-67.

Tucci, Robert R. 2013. 'Introduction to Judea Pearl's Do-Calculus', *arXiv preprint arXiv:1305.5506*.

Tunyasuvunakool, Kathryn, Jonas Adler, Zachary Wu, Tim Green, Michal Zielinski, Augustin Žídek, Alex Bridgland, Andrew Cowie, Clemens Meyer, Agata Laydon, Sameer Velankar, Gerard J. Kleywegt, Alex Bateman, Richard Evans, Alexander Pritzel, Michael Figurnov, Olaf Ronneberger, Russ Bates, Simon A. A. Kohl, Anna Potapenko, Andrew J. Ballard, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Ellen Clancy, David Reiman, Stig Petersen, Andrew W. Senior, Koray Kavukcuoglu, Ewan Birney, Pushmeet Kohli, John Jumper, and Demis Hassabis. 2021. 'Highly accurate protein structure prediction for the human proteome', *Nature*, 596: 590-96.

Unterthiner, Thomas, Andreas Mayr, Günter Klambauer, and Sepp Hochreiter. 2015. 'Toxicity prediction using deep learning', *arXiv preprint arXiv:1503.01445*.

Unterthiner, Thomas, Andreas Mayr, Günter Klambauer, Marvin Steijaert, Jörg K Wegner, Hugo Ceulemans, and Sepp Hochreiter. 2014. 'Deep learning as an opportunity in virtual screening', *Advances in neural information processing systems*, 27.

van Amsterdam, WAC, JJC Verhoeff, PA de Jong, T Leiner, and MJC Eijkemans. 2019. 'Eliminating biasing signals in lung cancer images for prognosis predictions with deep learning', *NPJ digital medicine*, 2: 1-6.

van de Waterbeemd, H., G. Camenisch, G. Folkers, J. R. Chretien, and O. A. Raevsky. 1998. 'Estimation of blood-brain barrier crossing of drugs using molecular size and shape, and H-bonding descriptors', *J Drug Target*, 6: 151-65.

van Laarhoven, Twan, Sander B. Nabuurs, and Elena Marchiori. 2011. 'Gaussian interaction profile kernels for predicting drug–target interaction', *Bioinformatics*, 27: 3036-43.

van Westen, Gerard J. P., Jorg K. Wegner, Adriaan P. Ijzerman, Herman W. T. van Vlijmen, and A. Bender. 2011. 'Proteochemometric modeling as a tool to design selective compounds and for extrapolating to novel targets', *MedChemComm*, 2: 16-30.

Vanyukov, M. M., L. Kirisci, L. Moss, R. E. Tarter, M. D. Reynolds, B. S. Maher, G. P. Kirillova, T. Ridenour, and D. B. Clark. 2009a. 'Measurement of the risk for substance use disorders: phenotypic and genetic analysis of an index of common liability', *Behav Genet*, 39: 233-44.

Vanyukov, M. M., R. E. Tarter, L. Kirisci, G. P. Kirillova, B. S. Maher, and D. B. Clark. 2003. 'Liability to substance use disorders: 1. Common mechanisms and manifestations', *Neurosci Biobehav Rev*, 27: 507-15.

Vanyukov, Michael M., Levent Kirisci, Lisa Moss, Ralph E. Tarter, Maureen D. Reynolds, Brion S. Maher, Galina P. Kirillova, Ty Ridenour, and Duncan B. Clark. 2009b. 'Measurement of the Risk for Substance Use Disorders: Phenotypic and Genetic Analysis of an Index of Common Liability', *Behavior genetics*, 39: 233-44.

Vasimuddin, Md, and Srinivas Aluru. 2017. "Parallel exact dynamic bayesian network structure learning with application to gene networks." In *2017 IEEE 24th International Conference on High Performance Computing (HiPC)*, 42-51. IEEE.

Vastag, M., and G. M. Keseru. 2009. 'Current in vitro and in silico models of blood-brain barrier penetration: a practical view', *Curr Opin Drug Discov Devel*, 12: 115-24.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention is all you need." In *Advances in neural information processing systems*, 5998-6008.

Venter, J. C. 2001. 'The sequence of the human genome (vol 292, pg 1304, 2001)', *Science*, 292: 1838-38.

Verdejo-Garcia, A., A. J. Lawrence, and L. Clark. 2008a. 'Impulsivity as a vulnerability marker for substance-use disorders: Review of findings from high-risk research, problem gamblers and genetic association studies', *Neuroscience and Biobehavioral Reviews*, 32: 777-810.

———. 2008b. 'Impulsivity as a vulnerability marker for substance-use disorders: review of findings from high-risk research, problem gamblers and genetic association studies', *Neurosci Biobehav Rev*, 32: 777-810.

Vicini, P, and P H van der Graaf. 2013. 'Systems Pharmacology for Drug Discovery and Development: Paradigm Shift or Flash in the Pan?', *Clinical Pharmacology & Therapeutics*, 93: 379-81.

Wager, Travis T., Xinjun Hou, Patrick R. Verhoest, and Anabella Villalobos. 2010. 'Moving beyond Rules: The Development of a Central Nervous System Multiparameter Optimization (CNS MPO) Approach To Enable Alignment of Druglike Properties', *ACS Chemical Neuroscience*, 1: 435-49.

Wallach, Izhar, Michael Dzamba, and Abraham Heifets. 2015. 'AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery', *CoRR*, abs/1510.02855.

Wan, Fangping, and Jianyang Zeng. 2016. 'Deep learning with feature embedding for compound-protein interaction prediction', *bioRxiv*: 086033.

Wang, C., W. Caihua, L. Juan, L. Fei, T. Yafang, D. Zixin, and H. Qian-Nan. 2014. *Pairwise Input Neural Network for Target-Ligand Interaction Prediction*.

Wang, J., W. Wang, P. A. Kollman, and D. A. Case. 2006. 'Automatic atom type and bond type perception in molecular mechanical calculations', *J Mol Graph Model*, 25: 247-60.

Wang, L., C. Ma, P. Wipf, H. Liu, W. Su, and X. Q. Xie. 2013. 'TargetHunter: an in silico target identification tool for predicting therapeutic potential of small organic molecules based on chemogenomic database', *AAPS J*, 15: 395-406.

Wang, Lipo. 2005. *Support vector machines: theory and applications* (Springer Science & Business Media).

Wang, W. W., and R. MacKinnon. 2017. 'Cryo-EM Structure of the Open Human Ether-a-go-go-Related K+ Channel hERG', *Cell*, 169: 422-+.

Wang, Yan-Bin, Zhu-Hong You, Shan Yang, Hai-Cheng Yi, Zhan-Heng Chen, and Kai Zheng. 2020. 'A deep learning-based method for drug-target interaction prediction based on long short-term memory neural network', *BMC Medical Informatics and Decision Making*, 20: 49.

Wang, Yiwei, Lei Huang, Siwen Jiang, Yifei Wang, Jun Zou, Hongguang Fu, and Shengyong Yang. 2020. 'Capsule Networks Showed Excellent Performance in the Classification of hERG Blockers/Nonblockers', *Frontiers in Pharmacology*, 10.

Wang, Yuhao, and Jianyang Zeng. 2013. 'Predicting drug-target interactions using restricted Boltzmann machines', *Bioinformatics*, 29: i126-i34.

Wang, Z., H. Yang, Z. Wu, T. Wang, W. Li, Y. Tang, and G. Liu. 2018. 'In Silico Prediction of Blood-Brain Barrier Permeability of Compounds by Machine Learning and Resampling Methods', *ChemMedChem*, 13: 2189-201.

Webb, Benjamin, and Andrej Sali. 2016. 'Comparative Protein Structure Modeling Using MODELLER', *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, 54: 5.6.1-5.6.37.

Weininger, D. 1988. 'Smiles, a Chemical Language and Information-System .1. Introduction to Methodology and Encoding Rules', *Journal of Chemical Information and Computer Sciences*, 28: 31-36.

Wernick, Miles N., Yongyi Yang, Jovan G. Brankov, Grigori Yourganov, and Stephen C. Strother. 2010. 'Machine Learning in Medical Imaging', *IEEE signal processing magazine*, 27: 25-38.

Whiteman, Shawn D., Julia M. Becerra, and Sarah E. Killoren. 2009. 'Mechanisms of sibling socialization in normative family development', *New Directions for Child and Adolescent Development*, 2009: 29-43.

Winkler, David A, and Tu C Le. 2017. 'Performance of Deep and Shallow Neural Networks, the Universal Approximation Theorem, Activity Cliffs, and QSAR', *Molecular informatics*, 36.

Wishart, David S, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, and Zinat Sayeeda. 2018. 'DrugBank 5.0: a major update to the DrugBank database for 2018', *Nucleic Acids Research*, 46: D1074-D82.

Wong, K. H., M. K. Riaz, Y. Xie, X. Zhang, Q. Liu, H. Chen, Z. Bian, X. Chen, A. Lu, and Z. Yang. 2019. 'Review of Current Strategies for Delivering Alzheimer's Disease Drugs across the Blood-Brain Barrier', *Int J Mol Sci*, 20.

Wu, Zhenqin, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. 2017. 'MoleculeNet: A Benchmark for Molecular Machine Learning', *arXiv:1703.00564*.

Wu, Zonghan, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. 'A comprehensive survey on graph neural networks', *IEEE transactions on neural networks and learning systems*.

Xiao, L., J. Diao, D. Greene, J. Wang, and R. Luo. 2017. 'A Continuum Poisson-Boltzmann Model for Membrane Channel Proteins', *J Chem Theory Comput*, 13: 3398-412.

Xie, Xiang-Qun, Lirong Wang, Qin Ouyang, Cheng Fang, and Weiwei Su. 2014. 'Chemogenomics knowledgebased polypharmacology analyses of drug abuse related G-protein coupled receptors and their ligands', *Frontiers in Pharmacology*, 5: 3.

Xie, Xiang-Qun, Lirong Wang, Junmei Wang, Zhaojun Xie, Peng Yang, and Qin Ouyang. 2016. 'In Silico Chemogenomics Knowledgebase and Computational System Neuropharmacology Approach for Cannabinoid Drug Research.' in, *Neuropathology of Drug Addictions and Substance Misuse* (Elsevier).

Xu, Lei, Xiaoqing Ru, and Rong Song. 2021. 'Application of Machine Learning for Drug-Target Interaction Prediction', *Frontiers in genetics*, 12: 680117-17.

Xu, X., E. E. Bishop, S. M. Kennedy, S. A. Simpson, and T. F. Pechacek. 2015. 'Annual Healthcare Spending Attributable to Cigarette Smoking An Update', *American Journal of Preventive Medicine*, 48: 326-33.

Xu, X. M., S. F. Ma, Z. W. Feng, G. X. Hu, L. R. Wang, and X. Q. Xie. 2016. 'Chemogenomics knowledgebase and systems pharmacology for hallucinogen target identification-Salvinorin A as a case study', *Journal of Molecular Graphics & Modelling*, 70: 284-95.

Yao, Kun, and John Parkhill. 2016. 'Kinetic Energy of Hydrocarbons as a Function of Electron Density and Convolutional Neural Networks', *Journal of Chemical Theory and Computation*, 12: 1139-47.

Yeomans, Keith A., and Paul A. Golder. 1982. 'The Guttman-Kaiser Criterion as a Predictor of the Number of Common Factors', *Journal of the Royal Statistical Society. Series D (The Statistician)*, 31: 221-29.

You, Jiaying, Robert D. McLeod, and Pingzhao Hu. 2019. 'Predicting drug-target interaction network using deep learning model', *Computational Biology and Chemistry*, 80: 90-101.

Yu, Kun-Hsing, Andrew L. Beam, and Isaac S. Kohane. 2018. 'Artificial intelligence in healthcare', *Nature Biomedical Engineering*, 2: 719-31.

Yu, Z., E. Klaasse, L. H. Heitman, and A. P. Ijzerman. 2014. 'Allosteric modulators of the hERG K(+) channel: radioligand binding assays reveal allosteric characteristics of dofetilide analogs', *Toxicol Appl Pharmacol*, 274: 78-86.

Yuan, Y., F. Zheng, and C. G. Zhan. 2018. 'Improved Prediction of Blood-Brain Barrier Permeability Through Machine Learning with Combined Use of Molecular Property-Based Descriptors and Fingerprints', *AAPS J*, 20: 54.

Zeiler, Matthew D., and Rob Fergus. 2014. "Visualizing and Understanding Convolutional Networks." In, 818-33. Cham: Springer International Publishing.

Zerbino, D. R., P. Achuthan, W. Akanni, M. R. Amode, D. Barrell, J. Bhai, K. Billis, C. Cummins, A. Gall, C. G. Giron, L. Gil, L. Gordon, L. Haggerty, E. Haskell, T. Hourlier, O. G. Izuogu, S. H. Janacek, T. Juettemann, J. K. To, M. R. Laird, I. Lavidas, Z. Liu, J. E. Loveland, T. Maurel, W. McLaren, B. Moore, J. Mudge, D. N. Murphy, V. Newman, M. Nuhn, D. Ogeh, C. K. Ong, A. Parker, M. Patricio, H. S. Riat, H. Schuilenburg, D. Sheppard, H. Sparrow, K. Taylor, A. Thormann, A. Vullo, B. Walts, A. Zadissa, A. Frankish, S. E. Hunt, M. Kostadima, N. Langridge, F. J. Martin, M. Muffato, E. Perry, M. Ruffier, D. M. Staines, S. J. Trevanion, B. L. Aken, F. Cunningham, A. Yates, and P. Flicek. 2017. 'Ensembl 2018', *Nucleic Acids Res*.

Zhang, L., H. Zhu, T. I. Oprea, A. Golbraikh, and A. Tropsha. 2008. 'QSAR modeling of the blood-brain barrier permeability for diverse organic compounds', *Pharm Res*, 25: 1902-14.

Zhang, Y., L. R. Wang, Z. W. Feng, H. Z. Cheng, T. F. McGuire, Y. H. Ding, T. Cheng, Y. D. Gao, and X. Q. Xie. 2016. 'StemCellCKB: An Integrated Stem Cell-Specific Chemogenomics KnowledgeBase for Target Identification and Systems-Pharmacology Research', *Journal of Chemical Information and Modeling*, 56: 1995-2004.

Zhang, Y. Y., H. Liu, S. G. Summerfield, C. N. Luscombe, and J. Sahi. 2016. 'Integrating in Silico and in Vitro Approaches To Predict Drug Accessibility to the Central Nervous System', *Mol Pharm*, 13: 1540-50.

Zhang, Y., J. Zhao, Y. Wang, Y. Fan, L. Zhu, Y. Yang, X. Chen, T. Lu, Y. Chen, and H. Liu. 2019. 'Prediction of hERG K+ channel blockage using deep neural networks', *Chem Biol Drug Des*, 94: 1973-85.

Zhao, Peng, and Bin Yu. 2006. 'On model selection consistency of Lasso', *the Journal of machine Learning research*, 7: 2541-63.

Zhao, Y. H., M. H. Abraham, A. Ibrahim, P. V. Fish, S. Cole, M. L. Lewis, M. J. de Groot, and D. P. Reynolds. 2007. 'Predicting penetration across the blood-brain barrier from simple descriptors and fragmentation schemes', *Journal of Chemical Information and Modeling*, 47: 170-5.

Zhou, Jie, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. 'Graph neural networks: A review of methods and applications', *AI Open*, 1: 57-81.

Zhou, Wei, Yonghua Wang, Aiping Lu, and Ge Zhang. 2016. 'Systems Pharmacology in Small Molecular Drug Discovery', *International Journal of Molecular Sciences*, 17: 246.

Zhou, Yadi, Fei Wang, Jian Tang, Ruth Nussinov, and Feixiong Cheng. 2020. 'Artificial intelligence in COVID-19 drug repurposing', *The Lancet Digital Health*, 2: e667-e76.