Universidade do Minho
Escola de Engenharia

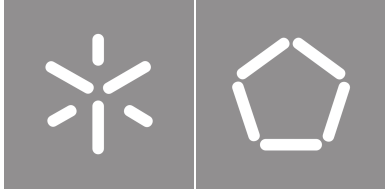
Delora Soeiro Baptista

A framework for predicting drug sensitivity and synergy in cancer cells using deep learning

December 2021

A framework for predicting drug sensitivity and synergy in cancer cells using deep learning
Delora Soeiro Baptista
UMinho | 2021

FCT Fundação para a Ciência e a Tecnologia
REPÚBLICA PORTUGUESA
CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR
NORTE2020 PROGRAMA OPERACIONAL REGIONAL DO NORTE
PORTUGAL 2020
UNIÃO EUROPEIA
Fundo Social Europeu



Universidade do Minho
Escola de Engenharia

Delora Soeiro Baptista

A framework for predicting drug sensitivity and synergy in cancer cells using deep learning

Doctoral Thesis
Doctoral Program in Biomedical Engineering

Work developed under the supervision of:
Professor Miguel Rocha
Professor Pedro G. Ferreira

December 2021

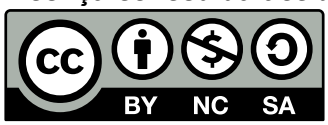
DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositoriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho



Creative Commons Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional
CC BY-NC-SA 4.0

https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pt

Acknowledgements

I would first like to thank Fundação para a Ciência e Tecnologia for the PhD scholarship I was awarded (SFRH/BD/130913/2017), without which this work would not be possible. I also thank my main host institution, the Centre of Biological Engineering at the University of Minho, and my secondary host institution, i3S.

I also wish to express my gratitude to my supervisors, Professor Miguel Rocha and Professor Pedro G. Ferreira. I thank them for offering me this opportunity and for all their support. I am immensely grateful for the guidance and feedback they provided throughout this project.

I would also like to thank all of my colleagues from the BISBII research group for the friendly and collaborative working environment, and for their useful comments and suggestions. I would especially like to thank João Correia for his collaboration on Chapter 4, and Ruben Rodrigues for his technical help with the servers.

Finally, a special thank you to my family and my boyfriend for their encouragement and support over the past four years.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the Universidade do Minho.

Resumo

Uma plataforma para a previsão de sensibilidade e sinergia de fármacos em linhas celulares de cancro utilizando aprendizagem profunda

Cancro é um termo aplicado a um conjunto diverso de doenças, sendo caracterizado pela proliferação anormal de células e sua potencial disseminação. Devido à heterogeneidade inter- e intratumoral, a descoberta de tratamentos eficazes para todos os tipos de cancro é uma tarefa complexa. A medicina de precisão é uma abordagem inovadora na área da saúde que consiste em estratificar doentes com base nas suas características genómicas e fenotípicas. Ao caracterizar extensivamente os tumores ao nível molecular utilizando tecnologias ómicas, torna-se possível desenvolver tratamentos direcionados a subtipos moleculares específicos de cancro. Os algoritmos e as ferramentas de inteligência artificial poderão auxiliar na identificação dos fármacos mais promissores ao prever a sensibilidade das células cancerígenas a fármacos individuais ou a combinações de compostos, reduzindo assim o esforço experimental necessário.

Neste trabalho, métodos de aprendizagem profunda foram desenvolvidos para prever a resposta de células tumorais a compostos químicos com base em dados ómicos e propriedades químicas dos compostos. Foram criados diversos modelos para vários problemas de previsão da resposta a medicamentos, incluindo a previsão da sensibilidade de linhas celulares de cancro e de amostras tumorais derivadas de pacientes e a previsão dos efeitos de combinações de medicamentos em linhas celulares tumorais. O impacto de diferentes escolhas metodológicas na capacidade preditiva dos modelos foi avaliado sistematicamente. Testaram-se diferentes estratégias de pré-processamento, métodos de representação dos dados, técnicas de seleção de atributos e de redução de dimensionalidade, e diversas arquiteturas de redes neuronais. Com base nestas análises, foi possível identificar as estratégias que produzem modelos com melhor desempenho preditivo. Demonstrou-se que os algoritmos de aprendizagem profunda são adequados a este tipo de problema, tendo superado o desempenho de outros algoritmos de aprendizagem máquina. Além disso, utilizando um método de interpretabilidade *post hoc*, demonstrou-se que os modelos de aprendizagem profunda têm a capacidade de associar corretamente a resposta celular a características biológicas e químicas importantes. Apesar da existência de limitações na abordagem descrita nesta tese, os resultados conseguidos serão úteis para guiar o desenvolvimento de novos métodos de aprendizagem profunda para a previsão da resposta de células tumorais a compostos químicos.

Palavras-chave: Aprendizagem profunda, cancro, medicina de precisão, sensibilidade a fármacos, sinergia farmacológica

Abstract

A framework for predicting drug sensitivity and synergy in cancer cells using deep learning

Cancer is a label applied to a diverse group of diseases, marked by abnormal cell proliferation and the potential for metastasis. Due to intertumoral and intratumoral heterogeneity, finding successful treatments for all types of cancers is a difficult endeavor. Precision medicine is an innovative approach to healthcare focused on stratifying patients for treatment based on their genomic and phenotypic characteristics. By extensively characterizing tumors at the molecular level using high-throughput omics technologies, it is possible to design targeted treatments aimed at specific molecular subtypes of cancer. Artificial intelligence algorithms and tools can help identify the most promising drug candidates by predicting the sensitivity of cancer cells to specific drugs or their combinations, reducing the experimental effort required.

In this work, deep learning (DL) methods were developed to predict drug response based on omics data characterizing tumoral cells and chemical features of anti-cancer drug candidates. Models were built for several drug response prediction problems, including the prediction of drug sensitivity for cancer cell lines and patient-derived samples, and estimating the effects of drug combinations in cancer cell lines. The impact of different methodological choices on the performance of these models was systematically evaluated. Different preprocessing strategies, feature representation methods, feature selection and dimensionality reduction techniques, and neural network architectures were tested. Based on these analyses, we were able to identify the strategies that generally lead to models with better predictive performance. We also show that DL is a suitable choice for this type of problem, as the performance of the DL models developed in this work surpasses that of other machine learning (ML) algorithms. In addition, using a *post hoc* interpretability method, we also demonstrate that DL is able to correctly associate important biological and chemical features with drug response. Despite some limitations of the proposed approach, our findings will be useful to guide the development of new DL-based drug response prediction methods.

Keywords: Cancer, deep learning, drug sensitivity, drug synergy, precision medicine

Contents

List of Figures **xi**

List of Tables **xiii**

Acronyms **xiv**

1 Introduction **1**

1.1 Context and Motivation 1

1.2 Research Objectives 2

1.3 Outline 3

2 Background **5**

2.1 Machine Learning 5

2.1.1 Definitions 5

2.1.2 Machine Learning Models and Algorithms 6

2.1.3 Artificial Neural Networks 8

2.1.4 Deep Learning 10

2.2 Cancer Treatment and Drug Response 11

2.2.1 Cancer Treatment 11

2.2.2 Drug Resistance and Drug Combinations 12

2.2.3 Drug Response Screening 13

3 Deep Learning for Drug Response Prediction in Cancer **15**

3.1 Introduction 15

3.2 Deep Learning 19

3.3 Data Resources 23

3.4 Implementing Deep Learning Workflows for Drug Response Prediction 27

3.5 Drug Response Prediction Models 30

3.5.1 Predicting Single-Drug Sensitivity . . . 33
3.5.2 Predicting Drug Combination Effects . . . 39
3.5.3 The Potential for Drug Repurposing . . . 41
3.6 Model Evaluation . . . 42
3.7 Improving Model Performance . . . 44
3.8 Model Interpretability . . . 47
3.9 Conclusion . . . 48

4 A comparison of different compound representations for drug sensitivity prediction 50
4.1 Introduction . . . 50
4.2 Methods . . . 52
4.2.1 Data sets . . . 52
4.2.2 Models . . . 53
4.2.3 Model training and evaluation . . . 55
4.2.4 Model ensembling . . . 55
4.2.5 Feature importance . . . 55
4.2.6 DeepMol . . . 56
4.3 Results and Discussion . . . 56
4.3.1 Individual models . . . 56
4.3.2 Ensemble models vs. individual models . . . 59
4.3.3 Feature importance . . . 60
4.4 Conclusion . . . 64

5 Predicting the ex vivo drug sensitivity of acute myeloid leukemia specimens 66
5.1 Introduction . . . 66
5.2 Methods . . . 67
5.2.1 Dataset . . . 67
5.2.2 Preprocessing . . . 68
5.2.3 Deep learning models . . . 69
5.2.4 Machine learning models . . . 71
5.2.5 Heterogeneous ensembles . . . 71
5.2.6 Feature importance . . . 71
5.2.7 Code availability . . . 72
5.3 Results . . . 72
5.3.1 Challenge solution . . . 72
5.3.2 Internal validation results . . . 73
5.3.3 Comparison with other ML algorithms . . . 76

5.3.4 Hetereogeneous ensemble . . . 77
5.3.5 Feature importance . . . 77
5.4 Discussion . . . 79
5.5 Conclusions . . . 80

6 Predicting drug synergy in cancer cell lines 81
6.1 Introduction . . . 81
6.2 Results . . . 84
6.2.1 Testing the impact of different methodological variables . . . 84
6.2.2 Baseline models . . . 88
6.2.3 Different gene expression subnetworks . . . 88
6.2.4 Drug encoding networks . . . 90
6.2.5 Additional cell line features . . . 91
6.2.6 Comparison with traditional machine learning models . . . 92
6.2.7 Heterogeneous ensemble . . . 93
6.2.8 Feature importance . . . 93
6.2.9 Selecting hyperparameter values . . . 100
6.3 Discussion . . . 101
6.4 Methods . . . 102
6.4.1 Dataset & preprocessing . . . 102
6.4.2 Models . . . 104
6.4.3 Model hyperparameters & hyperparameter optimization . . . 104
6.4.4 Model evaluation . . . 105
6.4.5 Feature importance . . . 105
6.4.6 Code availability . . . 105

7 Conclusions and Future Work 106
7.1 Main contributions . . . 106
7.2 Publications . . . 108
7.3 Future work . . . 109

Bibliography 110

Appendices

A Supplementary Figures 132
A.1 Supplementary Figures for Chapter 5 . . . 132

B Supplementary Tables 133

B.1 Supplementary Tables for Chapter 5 133
B.2 Supplementary Tables for Chapter 6 138

List of Figures

1 A general workflow for supervised machine learning tasks. 5
2 An example of an artificial neural network with a single hidden layer. 9
3 Common activation functions. 9
4 An example of a dose-response curve. 14
5 An example of a dose-response surface for the combination of two drugs. 14

6 Deep learning architectures that have been used in drug response prediction models. . . 21
7 The usual steps in a deep learning-based drug response prediction workflow. 28
8 Input and output data types commonly used when building deep learning-based drug response prediction models. 29
9 Different model validation schemes. 43

10 Traditional representations of compounds for machine learning tasks. 51
11 Learned representations of compounds. 52
12 General data preprocessing and machine learning steps followed in this work. 55
13 Performance (ROC-AUC) of different deep learning models on the NCI 1 and NCI 109 classification tasks. 57
14 Performance (RMSE) of different deep learning models on the PC-3, CCRF-CEM and NCI-60 A549 regression tasks. 58
15 Global feature importance for the ECFP4 model trained on the NCI 1 dataset. 60
16 Global feature importance for the AtomPair model trained on the CCRF-CEM dataset. . . 61
17 SHAP explanation of the prediction made by the ECFP4 model for sample 537 from the NCI 1 test set. 62
18 Molecular fragments corresponding to the ten ECFP4 bits that contribute the most towards the prediction for sample 537 from the NCI 1 test set. 63
19 SHAP explanation of the prediction made by the AtomPair model for sample 90 from the CCRF-CEM test set. 64

20 A schematic representation of a multimodal model for drug sensitivity prediction similar to the models developed in this work. 70
21 Final results for subchallenge 1 of the CTD^2 Beat AML DREAM Challenge. 73
22 Average per-inhibitor Spearman correlation scores for models trained on different input type combinations. 74
23 Average per-inhibitor Spearman correlation scores for models trained on different gene expression features selected using different feature selection methods. 75
24 Average per-inhibitor Spearman correlation scores for the model configuration that had been submitted to the challenge and two additional models that were developed after the challenge. 76
25 Comparison between the best DL model developed in this study and other ML algorithms. 77
26 Top 20 most important features ranked by mean absolute SHAP values. 78
27 A representation of the general architecture of the multimodal DL models developed in this study. 85
28 The different types of DL and ML models developed in this study. 87
29 Performance scores of models with different gene expression feature-encoding subnetworks. 89
30 Performance scores of models with different drug encoding subnetworks. 90
31 Performance scores of models trained with and without mutation (mut) and copy number variation (cnv) data in addition to gene expression and drug features. 91
32 Performance scores of traditional ML models compared with one of the best DL models developed in this study. 92
33 Top 20 most important features. 94
34 Top 20 features for test set sample 11835. 95
35 Top 20 drugA features for test set sample 11835. 96
36 Top 20 drugB features for test set sample 11835. 97
37 2D depiction of the top 15 most important ECFP4 “ON” bits for drugA (Topotecan) in test set sample 11835. 98
38 2D depiction of the top 15 most important ECFP4 “ON” bits for drugB (Gefitinib) in test set sample 11835. 98
39 Top 20 gene expression features for test set sample 11835. 99
40 Top 20 most important gene expression features, ranked according to their mean absolute SHAP values. 132

List of Tables

1 Some of the main classes of chemotherapeutic and targeted anti-cancer drugs. 12

2 Terminology box. 17
3 Datasets from single-drug pan-cancer screening studies. 25
4 Datasets from large-scale pan-cancer drug combination screening studies. 26
5 Published studies that have used deep learning for drug response prediction. 30
6 Performance scores for some of the drug response prediction models for single compounds referred to in this review. 34
7 Performance scores for some of the drug response prediction models for drug combinations referred to in this review. 40

8 Details on the datasets used in this study. 53
9 Ensemble results for classification tasks. 59
10 Ensemble results for regression tasks. 59

11 Selected clinical features. 69
12 Mean absolute SHAP values for features grouped by input type. 79

13 Average per-inhibitor Spearman correlation scores for models before guaranteeing that the dataset was always split the same way. 133
14 Model performance scores. 134
15 Top 20 features (ranked by Gini importance) for the random forest model. 138

Acronyms

Adam	adaptive moment estimation
AE	autoencoder
ALMANAC	A Large Matrix of Anti-Neoplastic Agent Combinations
AML	acute myeloid leukemia
API	Application Programming Interface
AtomPair	atom pair fingerprints
ATP	adenosine triphosphate
AUC	area under the dose-response curve
AUROC	area under the receiver operating characteristic curve
BOHB	Bayesian Optimization with HyperBand
CART	classification and regression trees
CCLE	Cancer Cell Line Encyclopedia
CMap	Connectivity Map
CNN	convolutional neural network
CNV	copy number variation
COSMIC	Catalogue Of Somatic Mutations In Cancer
CTD^2	Cancer Target Discovery and Development
CTRP	Cancer Therapeutics Response Portal
DBN	deep belief network
DeepLIFT	Deep Learning Important FeaTures
DepMap	Cancer Dependency Map
DGI	drug-gene interaction
DGIdb	The Drug-gene Interaction Database
DL	deep learning
DNN	deep neural network
DREAM	Dialogue on Reverse Engineering Assessment and Methods

DTP	Developmental Therapeutics Program
EC_{50}	half maximal effective concentration
ECFP	extended connectivity fingerprint
EGFR	epidermal growth factor receptor
FCNN	fully-connected neural network
FPKM	Fragments Per Kilobase of transcript per Million mapped reads
GAN	generative adversarial network
GAT	graph attention network
GB	gradient boosting
GCN	graph convolutional network
GDC	Genomic Data Commons
GDSC	Genomics of Drug Sensitivity in Cancer
GI_{50}	growth inhibition of 50% of cells
GISTIC	Genomic Identification of Significant Targets in Cancer
GNN	graph neural network
IC_{50}	half maximal inhibitory concentration
ICGC	International Cancer Genome Consortium
IGTD	Image Generator for Tabular Data
ILP	inductive logic programming
LASSO	least absolute shrinkage and selection operator
LayeredFP	RDKit layered fingerprints
LGBM	light gradient boosting machine
LIME	Local Interpretable Model-agnostic Explanations
LINCS	Library of Integrated Network-Based Cellular Signatures
LSTM	long short-term memory
MACCS	Molecular ACCess System
MAF	mutation annotation format
ML	machine learning
MSE	mean squared error
MSigDB	Molecular Signatures Database
MTE	Molecular Transformer Embedding
NCG	Network of Cancer Genes
NCI	National Cancer Institute
NCI-60	National Cancer Institute 60 Human Cancer Cell Line Screen
NLP	natural language processing
NN	neural network
NSC	National Service Center

PCA principal component analysis
PharmGKB Pharmacogenomics Knowledgebase
pReLU parametric rectified linear unit
QSAR quantitative structure-activity relationship
R^2 coefficient of determination
RBF radial basis function
RBM restricted Boltzmann machine
RDKitFP RDKit fingerprints
REFINED REpresentation of Features as Images with NEighborhood Dependencies
ReLU rectified linear unit
RF random forest
RMSE root mean squared error
RNN recurrent neural network
ROC-AUC area under the receiver operating characteristic curve
RSS residual sum of squares
SDF structure-data file
SGD stochastic gradient descent
SHAP SHapley Additive exPlanations
SMILES Simplified Molecular-Input Line-Entry System
STITCH search tool for interactions of chemicals
SVM support vector machine
SVR support vector regression
tanh hyperbolic tangent
TNR true negative rate
TPR true positive rate
UMAP Uniform Manifold Approximation and Projection
VAE variational autoencoder
WGCNA Weighted Gene Co-expression Network Analysis
XGBoost extreme gradient boost

1 Introduction

1.1 Context and Motivation

Precision medicine represents a paradigm shift in drug development. Instead of trying to find a general, "one-size-fits-all" solution for a given disease, it aims to tailor treatments to specific groups of patients, according to their individual characteristics, in order to improve patient outcomes [1]. One of the most prominent endeavors in the field has been the search for personalized cancer treatments.

The success of precision oncology relies on the extensive characterization of tumors at the molecular level, and the effective translation of genomic and other "omics" data into prognostic and predictive biomarkers [2]. Another relevant task is to generate drug sensitivity profiles, spanning many drugs and genomic/epigenomic variants. The effectiveness of drug candidates can be evaluated through high-throughput screening. In these experiments, multiple compounds are screened at different concentrations across a panel of cell lines, and the cellular response to the drugs is measured, resulting in dose-response relationships [3]. In recent years, several datasets from large-scale drug screening projects have been made publicly available [3]–[8].

Computational methods are essential to gather insights from these large datasets. Additionally, although high-throughput drug screening technologies allow many drug candidates to be tested at once, experimentally screening all potential drug candidates would be impossible, for both practical and financial reasons [8], [9]. Computational methods can help limit the search space by identifying the most promising candidates, before any experimental work is undertaken. Publicly available anti-cancer drug screening datasets have already allowed the development of computational methods to identify putative biomarkers, using statistical methods [3], [10] and ML models that predict drug sensitivity based on cell line omics data and drug chemical features [11], [12].

A more complex problem is the prediction of drug synergy. The efficacy of single-drug therapies is often diminished due to tumor drug resistance, either innate or arising in response to treatment [13]. Drug combinations are a common strategy to overcome resistance. Drug synergy is the enhancement of combination effects beyond expected, and it is a highly desirable outcome, as it may increase treatment efficacy without requiring an increase in dosage [14]. Novel drug combinations are also usually discovered

through high-throughput screening experiments. Once again, given the vast number of drug combinations that are possible, screening all conceivable drug pairs would be unfeasible. Therefore, computational methods that accurately predict drug synergy could also greatly reduce the experimental effort required to develop effective combination therapies.

The Dialogue on Reverse Engineering Assessment and Methods (DREAM) Challenges crowdsourcing initiative has launched several challenges over the past few years to accelerate the development of drug response prediction methods. In the National Cancer Institute (NCI)-DREAM Drug Sensitivity Challenge [15], competing teams were asked to predict drug response based on genomics, epigenomics, and proteomics data from several breast cancer cell-lines. The NCI-DREAM Drug Synergy Challenge [9] was focused on predicting the effects of drug combinations on a lymphoma cell-line, with participants having to rank compound pairs according to the predicted response. The more recent AstraZeneca-Sanger Drug Combination Prediction Challenge was aimed at the development of ML methods to predict the effects of drug combinations on an extensive panel of cancer cell lines [8]. Despite some encouraging results from these challenges and other studies, the proposed solutions are still far from optimal and there is still much room for improvement in terms of predictive performance, model generalizability [16], and model interpretability.

DL is a subclass of ML based on artificial neural networks. Deep neural networks (DNNs) are composed of multiple processing layers, which enables them to learn complex functions [17] and allows them to discover suitable higher-order representations to handle raw data [18]. In addition, DL is able to handle large volumes of high-dimensional and noisy data, such as omics datasets. By integrating several data sources, smoothing and weighting different data, DL may be able to capture the complex biological relationships present in the data. These characteristics make DL algorithms interesting candidates for drug sensitivity and synergy prediction from large-scale omics and chemical datasets. Nevertheless, the application of DL requires significant expertise and resources, which has hampered its use in this field. As a result, the use of DL to model drug response has been underexplored until very recently.

In this work, the development of a framework based on DNN is proposed, aiming to address drug sensitivity and synergy prediction from cancer cell omics data and drug chemical features. Methods will be developed to train deep models using distinct data types and combine them to build effective predictors. Different preprocessing techniques and DL architectures will be systematically evaluated. In addition, state-of-the-art methods for model interpretability will also be explored.

1.2 Research Objectives

The main goal of this work is to develop a DL-based computational framework capable of predictng the response of cancer cells when exposed to single drugs or drug combinations, based on distinct types of data, including multiomics (genomics and transcriptomics) data characterizing the cancer cells and chemical features of the drugs.

This encompasses a set of specific scientific and technological objectives:

- To review the state-of-the-art of drug sensitivity and drug synergy prediction approaches, as well as relevant literature on DL models and training/optimization algorithms, methods for handling overfitting and ways to handle multimodal learning, databases and portals with relevant data on cancer cells and drugs, and other relevant resources.

- To design and implement algorithms to train DL models to predict drug sensitivity and synergy based on publicly available drug screening data, as well as omics data characterizing the cancer cells and the chemical properties of the drug candidates. Given the modularity of DL models, this will result in sets of models and model parts that can be reused by our team and the community;

- To validate the methods and the models developed above considering case studies including different types of cancer and different tasks (e.g. predicting drug sensitivity of cell line-drug pairs; predicting drug combination effects for cell line-drug A-drug B triplets);

- To make the developed pipelines and models open to the community to foster their reuse and the reproducibility of the studies;

- To write articles for peer-reviewed international journals and conferences with the main results of the work.

1.3 Outline

This manuscript is structured as follows:

In Chapter 1 (the present chapter), the research problem is identified and the main research objectives are set forth. At the end of the chapter, a brief overview of the structure of the manuscript is provided.

Chapter 2 provides the theoretical background for this work. The first part of the chapter introduces ML and DL, from the basic concepts underlying these algorithms to common problems and the strategies that have been proposed to overcome them. In the second part, cancer and drug-based cancer treatments are briefly described. In addition, drug resistance is explained, the concepts of drug sensitivity and drug synergy are defined, and the methodology behind high-throughput drug screens is described.

Chapter 3 is a review of recently published studies that have employed DL methods to predict drug sensitivity and drug synergy in cancer. A brief description of DL and the main types of architectures that have been used in these studies is provided. Additionally, a selection of publicly available drug screening data resources that can be used to develop drug response prediction models are presented. Finally, we also address the limitations of these approaches and provide a discussion on possible paths for further improvement.

Chapter 4 presents a study that was undertaken to determine which compound representation strategies are most suitable for drug response prediction in cancer cell lines. Twelve different compound representation methods were benchmarked on 5 anti-cancer drug sensitivity datasets. We also explore the use of ensemble learning to improve performance, and show that our DL models can be explained using a gradient-based feature attribution method.

Chapter 5 describes the process behind the development of a solution for a recent drug response prediction DREAM challenge. The goal of this challenge was to predict the sensitivity of tumor specimens obtained from acute myeloid leukemia (AML) patients to anti-cancer drugs, using compound features, omics data and clinical features. This chapter provides a description of the different multimodal DL models that were tested, and an analysis of the effects of using different combinations of input data sources and different compound representations.

Chapter 6 reports the analysis of the impact of different methodological variables on the performance of DL-based drug synergy prediction models, including different preprocessing steps, types of input data, and DL architectures. We discuss the main results and identify promising modeling strategies that may be useful for the development of new drug synergy prediction models in the future. We also show that DL models are able learn to associate drug and cell line features with drug response in a biologically meaningful way, and confirm that DL approaches may be better at predicting drug synergy than classical ML approaches.

The main conclusions of this thesis are summarized in Chapter 7. The work undertaken in this thesis is briefly analyzed and future perspectives and some ideas for improvement are also proposed in this chapter.

2 Background

2.1 Machine Learning

2.1.1 Definitions

The field of ML is focused on the design and application of algorithms that can learn information directly from data, and make accurate predictions using a model that is inferred from input data alone [19].

ML can be divided into two main types of approaches, unsupervised learning and supervised learning, although semi-supervised and reinforcement learning approaches also exist. In unsupervised learning, the goal is for the ML algorithm to find structure in unlabeled data. In supervised learning, an estimator is initially trained on a dataset that contains a set of input features and the corresponding response (output) values. The goal is to obtain a function that maps the input variables to the output variable(s). Supervised learning can be used for classification tasks (if the response variable is categorical), ranking tasks (if it is ordered), or regression problems (when the output variable is continuous). A general workflow for supervised machine learning tasks is shown in Figure 1.

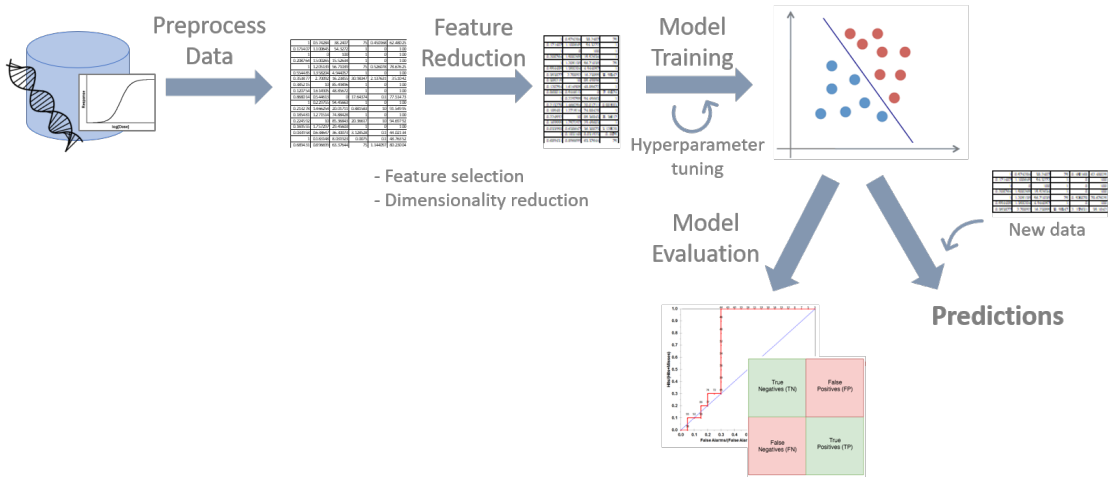


Figure 1: A general workflow for supervised machine learning tasks.

In supervised learning, model training is accomplished by optimizing a particular objective function -

usually minimizing a loss function - in order to estimate the model parameters given a training dataset. A loss function is a measure of the penalty associated with prediction errors [20]; thus, the loss function gives an idea of how well the model fits the data, while it may also encompass components measuring the model's capacity (or complexity). Once constructed, a supervised machine learning model can be applied to previously unseen data to predict the corresponding output values.

A common problem in ML is overfitting. It occurs when the complexity of the model is high and there is insufficient training data. The resulting model fits the training data well, but is unable to generalize to unseen data. One common solution to this problem is to increase the size of the training set, but this is not always possible. Another way to improve model generalization is to use a regularization method, which penalizes complex models in the loss function.

Feature selection techniques can also reduce model complexity by considering only subsets of the original variables. Feature selection also makes the learning process faster, and it simplifies the interpretation of both the model and the data [21]. An alternative strategy to reduce model complexity is the use of unsupervised dimensionality reduction techniques, such as principal component analysis (PCA).

To assess the quality of a given predictive model, it must be tested on a set of examples that were not used to train the model and for which the output values are known. This test set can be obtained by simply dividing the input data into two sets, and holding out one of the sets when training.

Other ways to generate training and test sets exist, such as cross-validation or bootstrapping. K-fold cross-validation randomly partitions the input dataset into k subsets that are equal in size and mutually exclusive. The estimator is evaluated k times, using one of the folds as the test set and the remaining folds as the training set. The cross-validated model performance is the average of the performance score calculated for the k iterations [22]. Leave-one-out cross-validation is the case where k is equal to the number of samples in the training set. On the other hand, bootstrapping creates a validation set by sampling from the original dataset with replacement.

While model parameters are optimized during model fitting, estimator hyperparameters must be optimized before training, as they must be set before the model fitting step. Hyperparameter optimization can be achieved through manual tuning, or by performing an exhaustive search (grid search) across all (or a set of) possible combinations of user-specified values or a random search that only evaluates a number of hyperparameter combinations sampled from the entire search space. Cross-validation or other unbiased estimation methods are used to compute performance metrics to evaluate each proposed set of hyperparameters. The chosen set of hyperparameters will be the one that maximizes the predictive performance of the model. Alternative methods for hyperparameter tuning also exist, such as Bayesian hyperparameter optimization [23] or optimization metaheuristics.

2.1.2 Machine Learning Models and Algorithms

In this section, some of the most popular machine learning models are briefly reviewed, focusing on those that are most used in drug discovery applications.

Linear regression methods are one of the simplest machine learning regression algorithms. Linear regression assumes that the relationship between the input features and the output variable is linear, and attempts to find the linear function that best fits the data. In ordinary least squares linear regression, the model parameters are estimated by minimizing the residual sum of squares (RSS) (Eq 2.1). Assuming that N is the number of samples, p is the number of features, x_{ij} are the inputs, y_i are the output values, and β_j are the regression coefficients (model parameters), the RSS is defined as:

$$RSS = \sum_{i=1}^{N}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 \tag{2.1}$$

If the determined model is excessively complex, overfitting may occur, causing the model to lose generalization capacity. To avoid overfitting, a regularization term that penalizes the size of the regression coefficients can be added. Ridge regression [24] is a linear least squares regression method with regularization which uses L2 regularization, which corresponds to adding the sum of the squares of the parameters as a penalty (Eq. 2.2).

$$Ridge\ loss = \sum_{i=1}^{N}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{2.2}$$

In this loss function, λ is the regularization coefficient, a parameter that represents the importance given to the penalization term.

Least absolute shrinkage and selection operator (LASSO) [25] uses L1 regularization, adding a penalization term to the loss function that corresponds to the sum of the absolute values of the parameters (Eq. 2.3).

$$LASSO\ loss = \sum_{i=1}^{N}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{2.3}$$

Elastic net regression [26] adopts the penalties of both of the previous methods.

Logistic regression [27] is a statistical method for linear classification that is used when the dependent variable is binary. It uses a sigmoid (logistic) function to model the probability of a given example belonging to a given class. Similar to linear regression, the estimation of the parameters of the model is an optimization problem. The parameters are estimated by minimizing classification errors through the maximization of the log-likelihood. This optimization problem can be solved using analytical methods or gradient descent algorithms.

Decision trees are recursive data structures including decision rules, which may be inferred from the input data through appropriate algorithms. Each internal node (decision node) of the tree represents a decision that is made regarding a particular feature, while branches represent the possible values for a given feature and leaf nodes are the final predicted values [28]. There are several tree building algorithms, one of which is the classification and regression trees (CART) algorithm [29]. To reduce overfitting and improve model performance, decision tree algorithms can include a pruning step, which is guided by the error on a validation set.

Support vector machines (SVMs) [30] are kernel-based machine learning methods. To be able to model non-linear problems, SVMs start by mapping the input features to a new, high-dimensional feature space using a kernel function. In this new feature space, the algorithm searches for the optimal linear hyperplane that maximizes the separation (margin) between classes. The input examples that define the margin constitute the support vectors; thus, the SVM model is only dependent upon a subset of the training data [30].

To improve model performance, multiple estimators can be combined in ensembles. A simple ensemble is one where predictions from several different estimators are averaged, giving the final prediction values. Stacking is another ensemble strategy, where predictions from various models are used as input for another estimator that makes the final prediction [31].

Random forests (RFs) [32] and gradient boosting (GB) [33] are more elaborate ensembles. Both typically use decision trees as their base estimator.

RFs apply the bootstrap aggregating (bagging) method to create different models. Bagging generates a number of distinct training subsets by sampling from the training set with replacement (bootstrapping). Each tree is grown using a modified version of the CART algorithm, which tests a random selection of features at each node instead of all the remaining features.

The RF predictions are an average of the predictions from the individual trees. The forest growth process is guided by the out-of-bag error, an estimate of the generalization error on each training sample, when predicted using the trees where bagging did not include the sample in their training set. This internal estimate is also used to determine feature importance.

GB also usually uses CART trees as a base estimator. In GB, the gradient descent optimization method is used to minimize an arbitrary differentiable loss function. The base estimators are trained in a sequential manner (boosting), guided by the negative gradient of the loss function being minimized. This strategy is used to iteratively transform weak learners into more predictive estimators. For each iteration, the base estimator is fit with a subset of samples that is randomly chosen without replacement from the full training dataset [34].

2.1.3 Artificial Neural Networks

Artificial neural networks (NNs) are a group of machine learning techniques, which are inspired by their biological counterparts. Both supervised and unsupervised NN models exist.

Supervised NNs are comprised of several connected layers which contain several units (also called “neurons”). The most used NNs are fully-connected networks composed of an input layer, a single hidden layer, and an output layer (Figure 2), also called a multilayer perceptron (MLP). However, NNs may have more than one hidden layer. These models are feed-forward networks, which means that they constitute an acyclic graph in which information flows in only one direction, from input nodes to output nodes [35, p. 164].

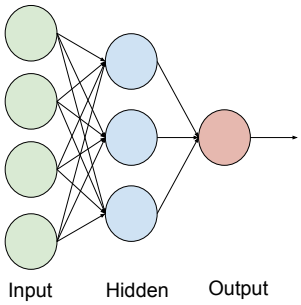


Figure 2: An example of an artificial neural network with a single hidden layer.

In hidden or output layers, each unit receives its inputs from the preceding layer. The connections between nodes in adjacent layers, called edges, each have an associated weight, reflecting the relative importance of a given input. Each unit applies an activation function, typically non-linear, to the weighted sum of its inputs to calculate its output value [17]. Some of the most widely adopted non-linear activation functions for hidden units are the sigmoid function, the hyperbolic tangent (tanh) function, and, in recent years, the rectified linear unit (ReLU) function (see Figure 3). This forward propagation of information is continued until the final output value is predicted.

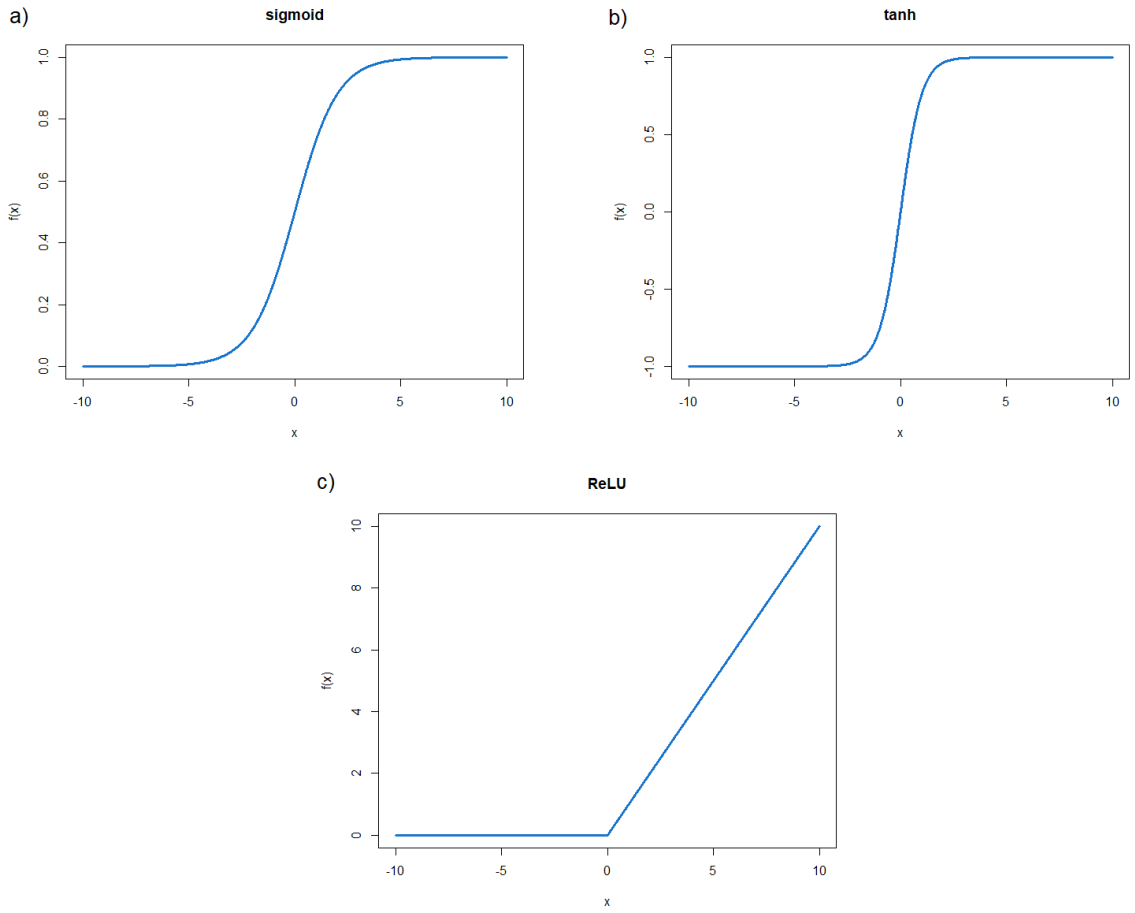


Figure 3: Common activation functions: a) sigmoid function; b) hyperbolic tangent (tanh) function; c) rectified linear unit (ReLU) function.

Training a NN is an optimization problem, where the goal is to minimize the difference between the predicted output values and the real values, that is, to minimize the error. Once an input case has been forward propagated until the output layer, the predicted and real output values are compared, and the error is determined using a loss function.

To train the network, the gradient of the loss function can be calculated, and then the backpropagation algorithm can be applied so that the error is propagated backwards, from the output layer to the input layer [17]. In this manner, the gradients with respect to the weights can be computed. The weights can then be adjusted using gradient methods, such as stochastic gradient descent (SGD), or variants such as the Adam algorithm [36]. Therefore, learning is achieved by iteratively modifying the weights.

2.1.4 Deep Learning

DL refers to a set of computational models that are based on the principles described above, but are more complex, having multiple hierarchical processing layers [17]. Due to this, DL models are capable of learning complex functions and, unlike most traditional machine learning methods, they typically do not require feature selection before training, since they have the ability to learn representations directly from raw input data [18].

There are many different types of DL architectures. Specific types of DL architectures that have been used for drug response prediction will be described in more detail in the following chapters.

DL approaches are well-suited to and perform better when trained on large amounts of data. In some cases, this can be a disadvantage, as obtaining sufficiently large sets of complete and labeled data for certain prediction tasks may be difficult. With smaller datasets, it may even be preferable to use more traditional machine learning methods. Unsupervised pre-training on a larger set of unlabeled data and then fine-tuning the model for a more specific prediction task using smaller datasets can help to overcome this issue.

Since DL models are complex and have many parameters to learn, they have a tendency to overfit. To avoid this, regularization techniques can be applied. The dropout technique temporarily eliminates randomly selected units during the training step [37]. Early stopping of model training, when a validation metric stops improving, can also avoid over-training the network.

Multi-task learning may also help reduce overfitting. In this approach, a single network is built to predict several different outcomes, therefore forcing it to learn features relevant for a variety of prediction tasks instead of a single, specific task.

The complexity of deep learning models poses additional challenges, including the more extensive hyperparameter optimization that is required, as well as the difficulty involved in training these models, which calls for greater computational capacity when compared to shallow methods.

Finally, one characteristic of DL models that has limited their application to biological and health related problems is their lack of interpretability. However, a few methods to improve the interpretability of NNs have recently been described in the literature. Some of the proposed methods include assigning

an importance score to input features by backpropagating an importance signal throughout the network [38], using an additional ML model to learn important features based on the predictions of DL models [39], [40], or using the concept of dropout applied to the input layer to rank input features [41].

2.2 Cancer Treatment and Drug Response

2.2.1 Cancer Treatment

Cancer is a label applied to a group of many diseases that may affect a variety of organs and tissues, all characterized by abnormal cell proliferation. Other defining characteristics, progressively acquired by many tumors and considered to be the hallmarks of cancer, include the ability to maintain proliferative signaling, evasion of growth suppression, activation of invasion and metastasis, enabling replicative immortality, inducing angiogenesis, cell death resistance, the deregulation and modification of cellular metabolism, and evading immunological destruction [42]. These progressive changes at the phenotype level are driven by changes at the genetic and epigenetic levels.

Despite these common features, tumors are very diverse, not only in terms of tissue of origin, but also at the molecular level. This diversity is reflected in the variety of strategies that are currently used to treat cancer. The main types of cancer treatment modalities are surgery, radiation therapy, chemotherapy, targeted therapies, hormone therapy, targeted therapies (molecularly targeted drugs), and immunotherapy. Cancer treatment plans frequently include a combination of different treatment modalities [43].

While chemotherapy may also encompass molecularly targeted drugs, it is usually used as a synonym for cytotoxic drugs. Cytotoxic drugs are those that kill all rapidly proliferating cells, tumoral or otherwise [44]. Due to this, chemotherapy may cause negative side effects. Recent anti-cancer drug discovery efforts have been focusing on more targeted forms of treatment. Molecularly targeted drugs specifically target proteins that are altered in tumors, with the aim of inhibiting the biochemical pathways that are crucial for tumor maintenance [45], and that underlie the previously mentioned cancer hallmarks. Targeted therapy may also refer to the use of immunotherapeutic agents such as monoclonal antibodies to stimulate an immune response against the tumor [45]. The goal with targeted therapy is to affect the tumor cells, while having little or no effect on normal cells, thus reducing the negative side effects of the treatment. Table 1 lists some of the main types of chemotherapeutic and targeted drugs.

In recent years, there has also been a shift towards more personalized treatments, thanks to the appearance of next-generation sequencing and other high-throughput “omics” technologies. Cancer cell lines and tumors have been studied at the genomics, transcriptomics, epigenomics, proteomics and metabolomics levels. These data have revealed that there is a great degree of heterogeneity even among tumors of the same histological subtype [47], and new molecular subtypes have been described [48], [49]. These findings highlight the complexity of developing successful cancer treatments. The hope with personalized medicine is that, by extensively characterizing the tumors at the molecular level, it will be

Table 1: Some of the main classes of chemotherapeutic and targeted anti-cancer drugs [46].

Drug Type	Examples
Chemotherapy	
Alkylating agents	Temozolomide, Cyclophosphamide
Platinum coordination complexes	Cisplatin, Carboplatin
Antimetabolites	Gemcitabine, Fluorouracil (5-fluorouracil)
Mitotic inhibitors	Vinorelbine, Paclitaxel
Topoisomerase inhibitors	Topotecan, Irinotecan
Anti-tumor antibiotics	Doxorubicin, Dactinomycin
Targeted Therapies	
Tyrosine kinase inhibitors	Imatinib, erlotinib
mTOR inhibitors	Everolimus, Vemurafenib
PARP inhibitors	Olaparib
Proteosome inhibitors	Bortezomib
Monoclonal antibodies	Trastuzumab, Bevacizumab

possible to design treatments specifically adapted to the characteristics of a particular molecular subgroup of tumors, increasing treatment success.

2.2.2 Drug Resistance and Drug Combinations

After an initial period of responsiveness, the efficacy of drug-based anti-cancer therapies is often reduced, due to the existence of tumor drug resistance mechanisms [13]. Resistance may be intrinsic or acquired. Intrinsic resistance is caused by the presence of resistance-conferring characteristics in the tumor cells prior to therapy, while acquired resistance develops during treatment as an adaptive response of the tumor. Furthermore, owing to intratumoral heterogeneity, drug-resistant cells may be positively selected during treatment [13], [47], which can lead to future relapse.

A variety of mechanisms have been implicated in drug resistance in cancerTumors undergoing treatment may adapt by activating pathways that are functionally similar to the targeted pathway [13]. Survival signaling pathways may also be activated, and death signaling pathways may be suppressed [13]. DNA repair mechanisms may be able to reverse the harmful effects of DNA-damaging drugs [13], [50]. Drug target alterations, such as mutations and or overexpression, may reduce the effectiveness of the treatment [13], [50]. Epigenetic modifications, such as DNA methylation and histone modification, can also contribute to drug resistance, by altering the expression levels of drug targets and other important proteins, such as those that are necessary for drug activation or DNA repair [50]. Other mechanisms involved in drug resistance include a lack of drug activation or complete inactivation of the drug before it can affect its target [50], altered drug transport across cell membranes [13], [50], and the protective role of the tumor microenvironment [13].

A common strategy to overcome drug resistance is the administration of two or more drugs in combination. These can be a mixture of chemotherapy drugs, a mixture of targeted therapies, or a mixture of both targeted and chemotherapeutic drugs. Some examples of drug combinations that include targeted

drugs and are currently being evaluated in clinical trials are the pertuzumab/trastuzumab/docetaxel drug combination [51], which is a combination of two targeted immunotherapeutic agents with a chemotherapeutic drug (docetaxel), and the everolimus/trastuzumab/vinorelbine combination [52], a combination of a molecularly targeted drug (everolimus), a monoclonal antibody (trastuzumab) and a chemotherapeutic drug (vinorelbine).

Drug combinations may circumvent pre-existing resistance mechanisms more easily, as well as prevent the development of acquired resistance mechanisms [44]. Combinations of targeted therapies accomplish this by inhibiting the same target through multiple mechanisms, by targeting different proteins within the same pathway, or by targeting parallel pathways [44]. Furthermore, certain combinations may be more effective than would be expected when taking into account the effects of each of the constituent compounds on their own, a phenomenon called drug synergy. Drug synergy increases treatment efficacy without requiring an increase in drug dosage, potentially avoiding an increase in toxicity as well [14]. Given this, ideal drug combinations are those that produce synergistic effects.

2.2.3 Drug Response Screening

High-throughput screening is a common first step in drug development to determine which compounds will potentially result in effective treatments. *In vitro* screening experiments can be biochemistry-based or cell-based assays [53]. Several recent anti-cancer drug discovery efforts have used cell-based screening [3], [4]. In these cell viability assays, many candidate drugs are screened across a panel of cancer cell lines. The compounds are administered at different concentrations and the cellular response to the drug is measured [3].

Screening results can be represented as dose-response curves, one for each compound-cell line pair tested. A dose-response curve plots the measured effect of a given drug as a function of the dose (concentration) that was administered. Drug dose is usually plotted on a logarithmic scale, producing a sigmoidal dose-response curve [46] (Figure 4). A mathematical model, such as the widely used Hill equation/four parameter logistic model [54], [55], is then fit to the dose-response data. Drug effects can be summarized by the parameters derived from these models, such as the half maximal effective concentration (EC_{50}), half maximal inhibitory concentration (IC_{50}), or the drug concentration that causes growth inhibition of 50% of cells (GI_{50}), among others.

The effect of drug combinations is also assessed in a similar manner. In this case, multiple compounds are administered simultaneously at several different concentrations and different ratios, and the resulting combination effect is described by a dose-response surface (Figure 5), from which the single-drug dose-response curves can then be derived [57].

When multiple drugs are jointly administered, the combination effect can be classified based on the difference between the observed response and the response that would be expected assuming that the drugs do not interact with each other [58]. If the resulting combination effect is the expected effect, it is considered additive. A drug combination is considered synergistic if the drug response is enhanced

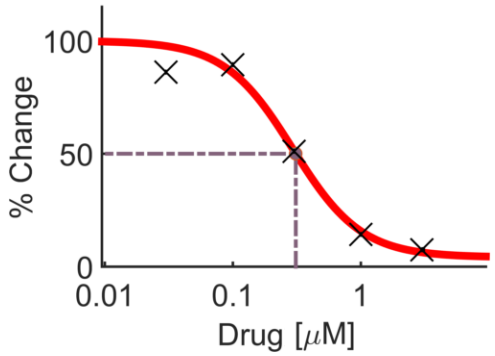


Figure 4: An example of a dose-response curve. Figure generated using Combenefit [56].

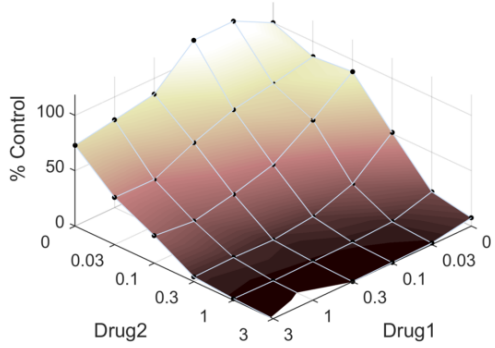


Figure 5: An example of a dose-response surface for the combination of two drugs. Figure generated using Combenefit [56].

beyond expected [14], whereas if the response is reduced when compared to the anticipated response, the combined effect is antagonistic.

Several reference models have been proposed to quantify drug combination effects. The definition of expected combination response differs between reference models, and, therefore, the identification of synergy/antagonism also varies between models. One of the most widely used reference models is the Loewe additivity model [59]. Loewe additivity postulates that a compound cannot have a synergistic interaction with itself, and that both compounds have similar mechanisms of action [60]. Another popular reference model is the Bliss independence model [61], which is based on the assumption that the drugs in a combination act in an independent manner, through different mechanisms [60].

3

Deep Learning for Drug Response Prediction in Cancer

This chapter is an unmodified version of a review article [62] published in Briefings in Bioinformatics, written in co-authorship with Pedro G. Ferreira and Miguel Rocha. The manuscript was up-to-date at the time of submission (September 2019).

3.1 Introduction

Precision medicine represents a challenge for this century, with the search for personalized cancer treatments being one of the most prominent endeavors in the field. The hope with precision medicine is that by profiling tumors at the molecular level it will be possible to design treatments specifically adapted to the characteristics of a particular molecular subgroup of tumors or even individual patients, improving treatment outcomes. The success of precision medicine relies, therefore, on effectively translating the combination of clinical data with genomics and other "omics" data into prognostic and predictive biomarkers.

Besides the characterization of tumors at the molecular level, another relevant task for precision oncology is to generate drug response profiles, spanning a wide range of drugs and cancer subtypes. In recent years, data from several large-scale drug screening initiatives [3]–[5], [63] have been made publicly available, helping to further the field of precision oncology. These projects have screened known and candidate anti-cancer drugs against cancer cell lines which have been extensively characterized at the molecular level. The data from these initiatives have already enabled the identification of putative drug response biomarkers using computational methods (elastic net regression) [3], [10] and the development of predictive models, such as elastic net and RF models to predict drug sensitivity [12].

Indeed, computational methods are crucial to make sense of these large drug screening datasets. Although high-throughput screening is a common first step in drug discovery, experimentally screening all possible candidate drugs or drug combinations is not feasible, for both practical and financial reasons [8], [9]. Therefore, the development of computational strategies to predict drug response is essential to limit the search space and guide the discovery process, reducing the experimental effort required. Since performing screening assays in cell cultures is currently the only alternative enabling high-throughput drug screening, computational methods are also needed to translate the knowledge obtained from cell line-based screens to drug response profiles of specific patients.

A variety of computational methods for drug response prediction and the discovery of drug response biomarkers have already been reported in the literature, including ML-based approaches such as SVMs [64], Bayesian multitask multiple kernel learning [15], [65], RFs [12], [66]–[68], and neural network [11] models. Nevertheless, there is still much room for improvement of these computational models in terms of predictive performance and model generalizability [16].

A particular subclass of machine learning algorithms referred to collectively as DL might be well-suited to the prediction of drug response based on pharmacological and cell line omics data. Since DL methods can handle large volumes of high-dimensional and noisy data, they may be able to capture the nonlinear and complex relationships typical of biological data better than other types of ML algorithms. Furthermore, DL has already been successfully applied to a wide range of other drug discovery-related tasks. For most of these tasks, the predictive performance of DL-based models is at least on par with other ML approaches, if not better [69]. For instance, DL has been shown to outperform traditional ML approaches in the prediction of compound activity [70]–[72], compound toxicity [73], and other compound properties [74]. Nevertheless, the application of DL to drug response prediction problems has been under-explored until very recently.

In this article, we review the state-of-the-art of DL-based drug response prediction methods. First, we introduce the concept of deep learning and describe the main types of DL architectures (Table 2 defines some of the technical terms that will be used throughout the manuscript). We then present a selection of publicly available drug screening data resources that can be used to develop drug response prediction models. Next, we briefly describe and comment on the DL models for drug response prediction that have been reported in the literature. We address both the strengths and limitations of these models, and discuss suggestions for further improvement. The insights gleaned from these studies will undoubtedly be useful in guiding the future development of computational methods for the rational design of effective anti-cancer treatments.

Table 2: Terminology box.

Term	Definition
Activation function	A function that each unit in a neural network applies to the weighted sum of its inputs to calculate its output value [17]
Attention mechanism	A method from the field of machine translation that identifies which parts of an input sequence are relevant to the output [75]
Backpropagation	An algorithm which propagates the prediction error of a neural network backwards, from the output layer to the input layer so that the gradients with respect to the weights of each unit can be computed [17]
Drug response biomarker	A biological characteristic that is predictive of the response of a tumor to a given treatment [76]
Classification	A supervised learning task where the output variable is categorical
Cross-validation	A model validation technique where data is divided into several subsets (folds) which are successively held out from the training set and used to estimate model performance
Data augmentation	Increasing the number of training data points
Discriminative model	Learns the conditional probability of the output given the inputs [77]
Dose-response relationship	The relationship between the observed effect (response) of a drug and its concentration (dose) [46]
Dropout	A technique that "drops" some neurons from a neural network in each iteration to reduce overfitting [37]
Drug sensitivity	The susceptibility of a cell line/tumor to a drug
Drug synergy	A phenomenon where the response to a combination is enhanced, going beyond the effect that would be expected based on the responses to each individual drug [14]
End-to-end learning	A learning approach where feature learning/extraction and outcome prediction are automatically performed by a single neural network instead of requiring multiple steps
Ensemble learning	Methods that combine the predictions of multiple machine learning models (base models), forming a single model

Table 2 continued from previous page

Term	Definition
Feature selection	A method to reduce model complexity by only considering smaller subsets of the original variables.
Generative model	Generative models learn the joint probability distribution of input and output variables being able to generate new inputs for a given distribution [77]
High-throughput screening	In the context of this paper, high-throughput screening refers to experiments where many candidate drugs are screened at varying concentrations across a panel of cancer cell lines and response to the drug is measured [3]
Hyperparameter	A model parameter that is not learned during training and must be set beforehand
Loss function	A function that measures the penalty associated with prediction errors [20]
Machine learning	A subfield of artificial intelligence that refers to algorithms that can learn information directly from data, and make accurate predictions using a model that is inferred from input data alone [19]
Molecular descriptors	Experimentally determined or theoretical properties of compounds summarized in numerical form [78]
Molecular fingerprints	A representation of the structure in the form of numerical vectors (e.g. binary fingerprints represent the presence of absence of certain chemical substructures within a molecule) [78]
Molecular graph	A representation of the structure of a compound in the form of a graph, where nodes represent atoms and edges represent bonds [79]
Multimodal learning	Using models that can relate and learn from data from different modalities (i.e. different input data types) [80]
Omics data	Fields of study in biology that focus on characterizing particular biological entities and interactions. Genomics studies genomes of organisms; transcriptomics the set of RNA transcripts; epigenomics the epigenetic modifications; proteomics the proteins of an organism; and metabolomics the concentration of compounds
One-hot encoding	Converting categorical variables into a numerical (binary) form

Table 2 continued from previous page

Term	Definition
Overfitting	Overfitting occurs when a model fits the training data well, but is unable to generalize to unseen data
Regression	A supervised learning task where the output variable is continuous
Representation learning	The process through which relevant features are learned automatically from the input data
Semi-supervised learning	A type of machine learning task where an estimator is trained on both labeled and unlabeled data to learn a function mapping input variables to output variables.
SMILES strings	A representation of compound structures in the form of ASCII strings
Supervised learning	A type of machine learning task where an estimator is trained on labeled data (a dataset that contains a set of input features and the corresponding output values) to predict the output for unseen samples of input data
(Model) Training (fitting)	The process through which the parameters of a model (in DL, the weights in a neural networks) are estimated
Transfer learning	A machine learning method where a model developed for one prediction task (a pre-trained network) is reused as the starting point for a similar task [35]
Unsupervised learning	A type of machine learning where the goal is for the algorithm to find structure in unlabeled data

3.2 Deep Learning

DL refers to a distinct class of machine learning algorithms based on artificial NNs, which, as the name suggests, are inspired by their biological counterparts. Artificial NNs consist of several connected layers, each containing multiple units, also called *neurons*. A shallow NN is composed of an input layer, a single hidden layer, and an output layer, while DNN are typically composed of multiple processing layers [17]. This characteristic allows these models to learn complex non-linear functions. Furthermore, unlike most traditional ML methods, DL approaches typically do not require extensive feature selection before training, since they have the ability to learn higher-order representations directly from raw input data [18].

In the hidden or output layers, each unit receives its inputs from the preceding layer. The connections between nodes in adjacent layers, called edges, each have an associated weight, reflecting the relative

importance of a given input. Each unit applies an activation function to the weighted sum of its inputs to calculate its output value [17]. This forward propagation of information is continued until the final output values (last layer) are predicted.

Training a NN is an optimization problem, where the goal is to minimize the difference between the predicted output values and the real values, that is, to minimize the error defined by a suitable loss function. Once an input example has been forward propagated until the output layer, the predicted and real output values are compared, and the error is determined using the defined loss function. To train the network, the gradient of the loss function can be calculated, and then the backpropagation algorithm can be applied so that the error is propagated backwards, from the output layer to the input layer [17]. In this manner, the gradients with respect to the weights can be computed. The weights can then be adjusted using gradient methods, such as SGD, or variants such as the Adam algorithm [36]. Therefore, learning is achieved by iteratively modifying the weights.

There is a wide variety of DL architectures, some of which are illustrated in Figure 6. In the following paragraphs, we will describe the main types DL models that have been applied to the problem of drug response prediction.

The simplest DL models are fully connected DNNs (Figure 6a). They are similar to the previously described shallow (single hidden layer) artificial NNs, but have a greater number of hidden layers. These networks are feedforward networks, which means that they constitute an acyclic graph in which information flows in only one direction, from input nodes to output nodes [35, p. 164]. DNNs have already been used to estimate drug response in cancer cell lines [82]. More complex models composed of multiple subnetworks (described in section 3.5) also usually use a DNN as the final subnetwork that predicts drug response.

Convolutional neural networks (CNNs) (Figure 6d) represent another type of feedforward DL model [83]. As the name implies, these NNs apply convolutions in some of their layers, usually in the initial layers of the network. Other common operations in CNNs include pooling and normalization, while the final layers of these networks are usually fully connected layers to allow for supervised classification or regression. Unlike DNNs, CNNs have sparse or local connectivity, which means that units in one layer are only directly connected to certain units in the previous layer [17]. This characteristic helps to preserve the local structure of the data [84]. CNNs also have comparatively fewer parameters to learn as the weights are shared, making them easier to train [17]. 2D CNNs are particularly well-suited to handle input data in the form of a grid (multi-dimensional arrays), such as images [17], and they can be used for drug response prediction if the input data is represented in the required format (e.g. using compound images as input [85]). 1D CNNs are more appropriate for data in the form of sequences, such as compound Simplified Molecular-Input Line-Entry System (SMILES) strings.

Recurrent neural networks (RNNs) (Figure 6b) are a distinct class of NNs characterized by the existence of cycles in the networks, typically formed due to edges that connect adjacent time steps (recurrent edges) [86]. Nodes with incoming recurrent connections can receive as input not only the current data point, but also the values of hidden units from previous time steps. This makes RNNs suitable to model data that is sequential in nature, such as natural language or time series. An RNN can be unfolded in time and

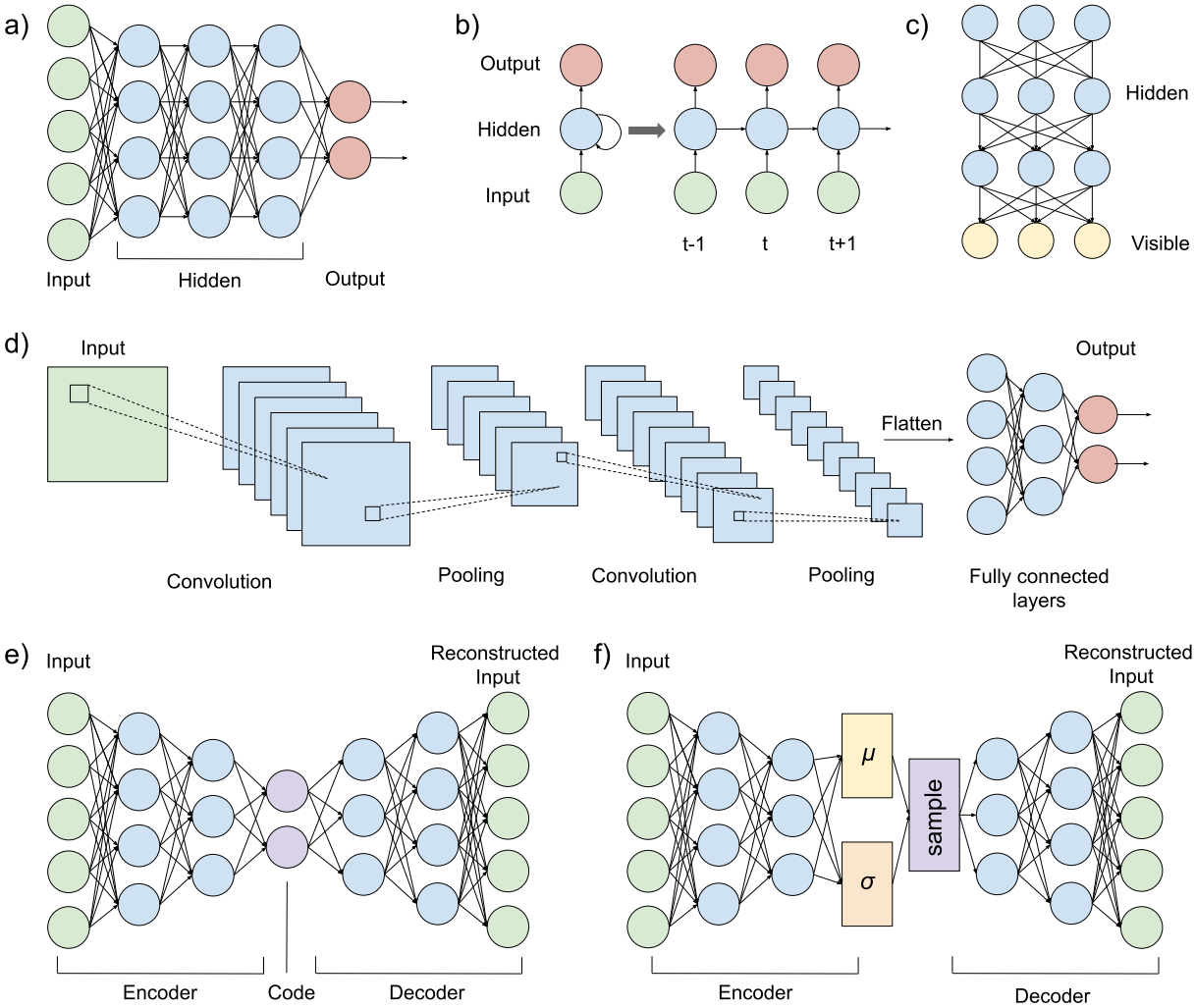


Figure 6: Deep learning architectures that have been used in drug response prediction models. a) A fully-connected feedforward deep neural network (DNN). b) A recurrent neural network (RNN) and the corresponding computational graph unfolded in time (t-1, t, and t+1 denote different time steps); adapted from [17]. c) A deep belief network (DBN); adapted from [81]. d) A convolutional neural network (CNN). e) A deep autoencoder (AE). f) A variational autoencoder (VAE); μ and σ are vectors of parameters defining the distributions of the latent variables.

represented as deep feedforward networks with the same weights being shared among layers [17]. In drug response prediction models, feature encoders with recurrent layers can be used to learn representations from SMILES strings, for example [87].

Unsupervised DL methods also exist. Autoencoders (AEs) are typically used for dimensionality reduction and feature representation learning before using other ML or DL methods for prediction (Figure 6e). An AE is a NN that learns to reconstruct its inputs [88]. By restricting the number of units in the hidden layers of the network and creating a bottleneck, the AE can learn a low-dimensional latent representation of the original input data, called a "code". A basic AE is therefore composed of two parts: an encoder, which produces the code; and a decoder, which attempts to reconstruct the input from the code [89]. Deep AEs are formed by stacking several AEs. AEs can be used to encode both compounds and omics data, and these learned representations can be fed into a predictive model to estimate drug response [90]–[92].

There are other variants of AEs, such as variational autoencoders (VAEs) [93], for instance (Figure 6f). A VAE is a generative model based on approximate inference. A VAE models the underlying probability distribution of the latent representations of the inputs. Like other AEs, VAEs are also composed of two networks. The encoder network learns a Gaussian distribution of the possible values of the latent representation from which a given sample could have been generated, while the decoder learns the distribution of the possible values of a sample that could be produced given a certain latent representation [93].

A less common type of DL model is the deep belief network (DBN) [94] (Figure 6c). A DBN is a generative model that consists of several layers of latent variables. The connections between the first two layers are undirected, while the connections between the remaining layers are all directed. DBNs can be considered stacks of restricted Boltzmann machines (RBMs) [95], which are composed of a visible layer and a hidden layer, with undirected connections between the two layers and without connections between units in the same layer. They can thus be trained layer by layer [94]. With the addition of a discriminative fine-tuning step at the end, DBNs can also be transformed into discriminative models [94]. DBN models have already been used to predict the response of cancer cell lines to drug combinations [96].

DL models can be implemented using one of the many open source DL libraries that are available. Some of the most popular DL libraries are Python-based, including PyTorch [97] and the TensorFlow [98] Python Application Programming Interface (API). Keras [99] is a higher-level DL API written in Python that facilitates the use of other lower-level libraries such as TensorFlow. Most of the models described in this review were implemented using one of these Python libraries. Another Python library that may be of interest to researchers in this field is DeepChem [100], a DL framework built on top of TensorFlow that offers implementations of chemistry-specific DL architectures and featurization techniques.

3.3 Data Resources

The high-throughput screening of compounds is a common step in the drug discovery process. Its purpose is to determine which compounds or combinations of compounds will potentially result in effective treatments. In recent years, several large-scale anti-cancer drug screens have been undertaken, the results of which have been made available through public repositories. Projects such as Genomics of Drug Sensitivity in Cancer (GDSC) [4], Cancer Cell Line Encyclopedia (CCLE) [3], Cancer Therapeutics Response Portal (CTRP) [5], [63] and NCI-60 [101] provide access to drug sensitivity profiles for a wide variety of cancer cell lines. An extended list of single-drug screening datasets is provided in Table 3. Large pan-cancer drug combination screening datasets have also been made available to the public (Table 4). These datasets can serve as the basis for the development of DL-based drug response prediction models.

In addition to the dose-response data gathered from the high-throughput screening experiments, some of these databases also provide access to omics data characterizing the cancer cell lines that the compounds were screened against. The cancer cell lines used in the National Cancer Institute 60 Human Cancer Cell Line Screen (NCI-60), CCLE, and GDSC screening panels, for example, have all been extensively characterized at the molecular level. All three projects provide genomic, transcriptomic, and epigenomic data characterizing the cell lines, and proteomics [102], [103] and metabolomics [104] data is available for the NCI-60 and CCLE cell lines. If there is a considerable overlap between cell lines, the omics data available from these databases can also be useful to extend other screening datasets that only include pharmacological dose-response data, such as the CTRPv2 dataset, for example.

The CCLE, GDSC and NCI-60 projects only offer baseline cell line data, that is, data obtained before treatment. The Connectivity Map (CMap) [105], [106] and the Library of Integrated Network-Based Cellular Signatures (LINCS) [107], [108] projects are resources that provide data on the transcriptional responses of cancer cells after treatment with small molecules. Another project recently analyzed the response of cells to treatment with small molecules at the proteomic and epigenomic levels [109]. These cellular response signatures can complement the data from other drug screening initiatives and may be a very valuable source of information when building drug response prediction models.

Smaller-scale and more specific (single cancer) drug screening datasets that were not included in Table 3 or Table 4 may cover other cell lines or compounds that were not contemplated by the larger screening initiatives. These datasets may be used to fine-tune models trained on larger pan-cancer datasets, making them more specific to certain tumor types.

Ideally, the datasets used to train drug response prediction models would come from patient cohorts, as cancer cell lines may not be representative of their tumors of origin [110], [111]. Although high-throughput screening has already been attempted using patient-derived xenografts [112] and organoids [113], the vast majority of anti-cancer drug response datasets are currently from cell line-based screening experiments. As a result, DL models are usually trained on the more abundant cell line-based screening data. Nevertheless, patient-derived data can be particularly useful to validate and refine drug response

prediction models trained on *in vitro* data, improving their clinical applicability. For example, patient-level data can be incorporated into DL models of cellular drug response by using the molecular data characterizing patient tumors in a pre-training step, as proposed in a recent study [92]. Large-scale patient-level omics datasets can be obtained from resources such as the Genomic Data Commons (GDC) Portal [114] and the International Cancer Genome Consortium (ICGC) [115].

Prior knowledge from additional data sources can complement and enrich the data from these publicly available drug screening datasets. Further information on the compounds screened in these experiments can be retrieved from PubChem [116], ChEMBL [117] or DrugBank [118], for instance. Information on drug targets and the pathways underlying drug response can be acquired from public databases, such as search tool for interactions of chemicals (STITCH) [119] and the Guide to Pharmacology [120].

External data sources can also be useful when exploring cell line omics data. The Catalogue Of Somatic Mutations In Cancer (COSMIC) database [121] is an important source of information on somatic mutations in cancer. Project Achilles [122], now a part of Cancer Dependency Map (DepMap) [123], is a project that aims to provide information on gene essentiality for cancer cell lines that have been extensively characterized at the molecular level. The Pharmacogenomics Knowledgebase (PharmGKB) [124] provides information regarding the influence of genetic variation on drug response. Besides these, many other sources of prior biological knowledge can be leveraged to enrich the drug response datasets and create more biologically informed models.

The DREAM challenges initiative is a community effort aimed at developing new computational approaches to address important biological and human health questions. In recent years, the DREAM community has proposed several challenges which can be useful sources of drug response data. In 2012, the community launched the NCI-DREAM Drug Sensitivity [15] and Drug Synergy [9] Challenges. The Drug Sensitivity challenge was aimed at building models to predict and rank the sensitivity of breast cancer cell lines to individual compounds, while the goal of the Drug Synergy challenge [9] was to predict the effect of drug combinations on a diffuse large B-cell lymphoma cell line. In September 2015, the DREAM community launched the AstraZeneca-Sanger Drug Combination Prediction challenge [8] to gather more insight on the factors underlying drug synergy and to accelerate the development of methods to predict drug combination effects. Challenge participants had access to a drug combination screening dataset provided by AstraZeneca which has now been published [8].

Other similar, but not cancer-specific, DREAM challenges can also be important sources of data that can complement the large anti-cancer drug screening datasets. For instance, the NIEHS-NCATS-UNC DREAM Toxicogenetics Challenge [125], which was aimed at predicting the cytotoxicity of compounds in humans, provides access to data that can be used to determine the toxic effects of anti-cancer drugs on patients.

Table 3: Datasets from single-drug pan-cancer screening studies.

Dataset	Website	# Compounds	# Cell Lines	PHARM	MUT	CNV	mRNA EXP	miRNA EXP	METHYL	PROT	METAB
NCI-60[101]	https://dtp.cancer.gov/discovery_development/nci-60/	~50,000	60	x	x	x	x	x	x	x	x
GlaxoSmithKline [126]		19	311	x							
CCLE [3], [103], [104]	https://portals.broadinstitute.org/ccle	24	479*	x	x	x	x	x	x	x	x
GDSC1 [4]	https://www.cancerrxgene.org/	320	988	x	x	x	x		x		
GDSC2 [4]	https://www.cancerrxgene.org/	175	810	x	x	x	x		x		
CTRPv1 [63]	https://portals.broadinstitute.org/ctrp.v1/	354	242	x							
CTRPv2 [5]	https://portals.broadinstitute.org/ctrp/	481	860	x							
gCSI [127]	http://research-pub.gene.com/gCSI-cellline-data/	16	410	x	x	x	x				
FIMM [128]		52	50	x							

*1,457 characterized cell lines in total

PHARM: Pharmaceutical data; MUT: mutation data; CNV: copy number variation data; mRNA EXP: mRNA expression data; miRNA EXP: microRNA expression data; METHYL: methylation data; PROT: proteomics data; METAB: metabolomics data

Table 4: Datasets from large-scale pan-cancer drug combination screening studies.

Dataset	Website	# Compounds	# Combinations	# Cell Lines	PHARM	MUT	CNV	mRNA EXP	miRNA EXP	METHYL	PROT	METAB
Merck Compound Screen [6]		38	583	39	x							
NCI-ALMANAC [7]	https://dtp.cancer.gov/ncialmanac	104	5,232	60	x	x	x	x	x	x	x	x
AstraZeneca Sanger DREAM [8]	https://www.synapse.org/#!Synapse:syn4231880	118	910	85	x							

PHARM: Pharmaceutical data; MUT: mutation data; CNV: copy number variation data; mRNA EXP: mRNA expression data; miRNA EXP: microRNA expression data; METHYL: methylation data; PROT: proteomics data; METAB: metabolomics data

3.4 Implementing Deep Learning Workflows for Drug Response Prediction

Most drug response prediction workflows follow the same general steps, which are shown in Figure 7. These include:

1. Selecting a DL framework to implement the model.
2. Defining the prediction problem (drug sensitivity vs. drug synergy prediction).
3. Selecting the dataset(s) used to train the model (see Tables 3 and 4 for lists of datasets for each type of problem);
4. Defining which data types will be used as inputs (Figure 8 provides an overview of the most common types of input data and output variables used in these models, as well as some of the most common preprocessing steps).
5. Deciding how multiple data types will be handled. Users must decide whether they will concatenate all features irrespective of data type, or if they will use a multimodal strategy with separate feature-encoding subnetworks for each data type.
6. Defining the model architecture. Input data type will determine the types of DL architectures that can be used for each network, as well as the preprocessing methods that need to be applied. For example, RNN-based architectures will require input data that is sequential in nature, such as SMILES strings.
7. Preprocessing and selecting appropriate data representations.
8. Training the model and tuning hyperparameters. Hyperparameter optimization can be achieved through manual tuning, by performing a search across all or a subset of possible combinations of user-specified values, or by using other optimization techniques.
9. Evaluating model performance. Researchers need to select appropriate scoring metrics, define how the screening dataset will be split into training and validation/test sets, and select any external datasets that will be used to validate the model.
10. Interpreting and explaining model predictions. Many of the model explanation methods mentioned in section 3.8 have been implemented as Python packages and can be easily incorporated into the workflow. Other *post hoc* analyses of the predictions can also be performed, as described in many of the studies reviewed here.

To illustrate these steps, we present a hypothetical drug response prediction workflow for the problem of drug sensitivity prediction. The first steps could be selecting the GDSC dataset for model training and

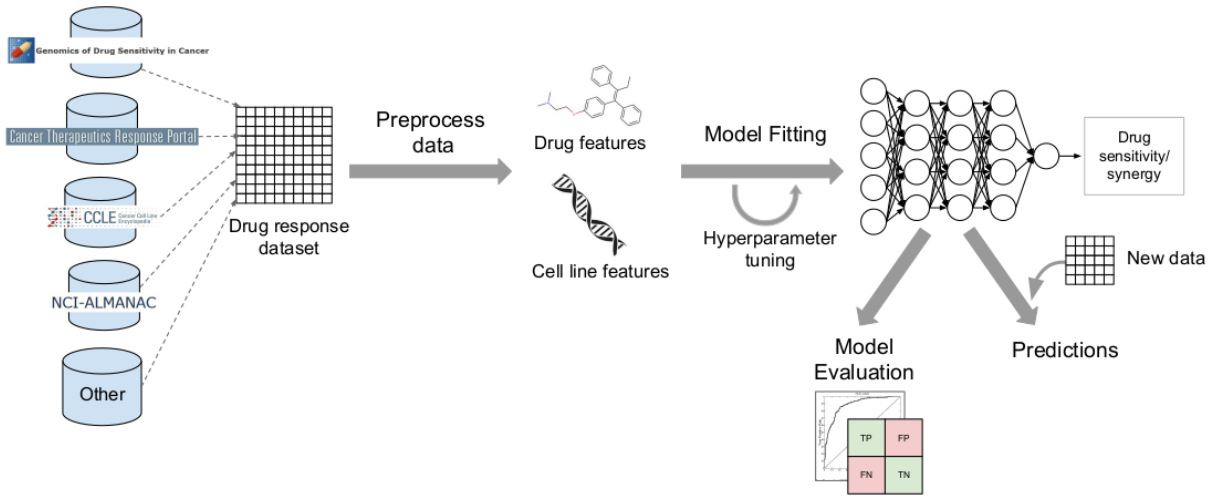


Figure 7: The usual steps in a deep learning-based drug response prediction workflow. A drug screening dataset is obtained from Genomics of Drug Sensitivity in Cancer (GDSC), Cancer Cell Line Encyclopedia (CCLE) or other resources. Complementary data and prior knowledge from other databases can also be collected at this stage. The data is then preprocessed according to data type. This results in a set of drug features and multiple omics features characterizing the cell lines. Afterwards, these distinct sets of features are either merged into a single training set or fed into separate encoding subnetworks. The model is then fit on the training set. Model hyperparameters can be optimized during the training stage. Different model validation techniques are used to evaluate model performance and select the best model. After training, the model can be used to predict drug response for new samples.

deciding to use gene expression data and one-hot encoded SMILES strings as input data. We could opt to use a multimodal model with a subnetwork for gene expression composed of fully connected layers and an encoding subnetwork for compounds that is a 1D CNN, both linked to a final prediction DNN. Such a network could be implemented using the Keras Python package. Hyperparameter optimization could be performed using grid search with cross-validation. After training the model, a model interpretability tool such as the shap Python package could be used to explain the model and determine feature importance. Model performance could be evaluated using the leave-drug-out and leave-cell-line out approaches, and the model could be further validated on the CCLE dataset. Considering that the problem is a regression task, we would use regression-specific scoring metrics such as R^2 to measure model performance.

The workflow would be very similar for a hypothetical drug synergy prediction problem. The main differences would be the screening datasets used for training and external validation, the output variable (measuring drug combination effects instead of drug sensitivity), and the model architecture would have to be adapted to allow for more than one drug. Since the hypothetical NN described in the previous example is modular and already includes a compound-encoding subnetwork, it could simply be extended by, for example, adding a subnetwork for the second compound which could share weights with the first compound subnetwork.

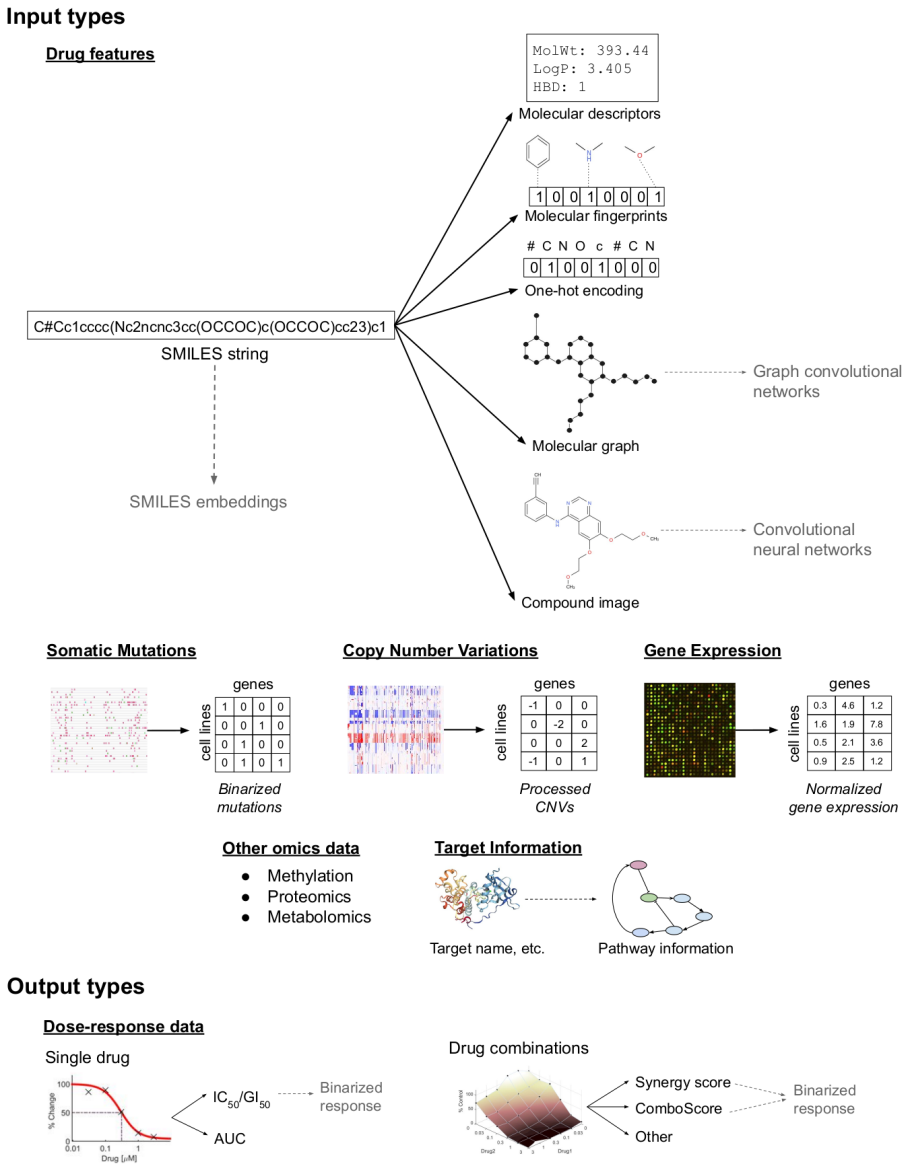


Figure 8: Input and output data types commonly used when building deep learning-based drug response prediction models. Compound structures are usually represented as Simplified Molecular-Input Line-Entry System (SMILES) strings in drug screening datasets. SMILES can be fed directly into deep learning models that learn embeddings, or they can be used to calculate molecular descriptors or molecular fingerprints. SMILES strings can also be one-hot encoded, transformed into the corresponding molecular graphs, which can then be used as input to graph convolutional networks, or converted into an image of the compound, which can be used as input to regular convolutional neural networks. Cell line input features are usually somatic mutations, copy number variations and gene expression data, although other omics data (epigenomics, proteomics, etc.) can also be incorporated into the models. Somatic mutations are usually binarized (presence/absence of an alteration). Copy number variations can be summarized as binary features, as scores (e.g. G-scores [129]), or in some other form. Gene expression features are usually continuous features which have undergone some form of normalization. Target information, when used, is usually used to derive features that reflect the pathways a particular target is associated with. The outputs of drug response prediction models are values describing the dose-response relationships. For single drugs, this is usually half maximal inhibitory concentration (IC_{50}), growth inhibition of 50% of cells (GI_{50}) or area under the dose-response curve (AUC). For drug combinations, the output variable is usually a score that quantifies drug combination effects based on a given reference model, such as the Loewe additivity [130] or Bliss independence [61] models.

3.5 Drug Response Prediction Models

Until recently, DL had seldom been applied to pharmacogenomics problems such as drug response prediction, but interest in DL approaches has greatly increased in the last few years. In this section, we first introduce readers to the main steps involved in DL-based workflows for drug response prediction, and then we briefly review the DL models for drug response prediction that have been published so far. For the most part, we did not consider preprints that have not been peer-reviewed yet. Table 5 summarizes the first few studies that have included DL in some form in their drug response prediction workflows. Tables 6 and 7 summarize the performance scores achieved by DL-based drug response prediction models for single drugs and drug combinations, respectively, as reported by the original studies.

Table 5: Published studies that have used deep learning for drug response prediction.

Study	Model	Training Datasets	Input Data Types	Prediction Task
Ding et al. [90]	Deep AEs + elastic nets/SVMs	GDSC	MUT, CNV, mRNA EXP	Drug sensitivity
Dr.VAE [131]	VAE + logistic regression	Cmap & CTRPv2	mRNA EXP (before & after treatment)	Joint modeling of drug sensitivity & drug perturbation signatures
DeepDSC [91]	Stacked AE + DNN	CCLE & GDSC	mRNA EXP, FP	Drug sensitivity
DeepDR [92]	DNN with separate feature-encoding subnetworks for each data type (encoders pre-trained on patient data)	GDC & CCLE & GDSC	MUT, mRNA EXP	Drug sensitivity

Table 5 continued from previous page

Study	Model	Training Datasets	Input Data Types	Prediction Task
PaccMann [87]	DL models with a gene expression encoder with an attention mechanism & a compound encoder (bRNN, SCNN, SA, CA, or MCA)	GDSC	mRNA EXP, SMILES	Drug sensitivity
MOLI [132]	DNN with separate feature-encoding subnetworks for each data type	GDSC	MUT, CNV, mRNA EXP	Drug sensitivity
tCNNS [133]	1D CNN, with separate encoders for drugs and genomic data	GDSC	MUT, CNV, SMILES	Drug sensitivity
KekuleScope [85]	CNN models pre-trained on ImageNet	ChEMBL	Compound images (Kekulé structures)	Drug sensitivity
Matlock et al. [134]	Heterogeneous ensembles that include DNNs	CCLE, GDSC, or synthetic data	mRNA EXP and/or DSCRPTR, target information	Drug sensitivity

Table 5 continued from previous page

Study	Model	Training Datasets	Input Data Types	Prediction Task
ELDAP [135]	Heterogeneous ensembles that include DNNs	CCLE, GDSC, LINCS	mRNA EXP, drug activity and cell line sensitivity signatures derived from drug-induced gene expression profiles	Drug sensitivity
CDRScan [136]	Ensemble of 5 CNNs	COSMIC, GDSC	MUT, FP & DSCRPTR	Drug sensitivity
DeepSynergy [82]	DNN	GDSC & Merck Compound Screen	mRNA EXP, FP & other compound features	Drug synergy
Xia et al. [137]	DNN with separate feature-encoding subnetworks for each data type	NCI-ALMANAC & NCI-60	mRNA EXP, PROT, miRNA EXP, DSCRPTR	Drug synergy
Chen et al. [96]	DBN	AstraZeneca-Sanger DREAM & GDSC	mRNA EXP, ontology fingerprints	Drug synergy
"DMIS new model" [8]	DL model with four different feature-encoding modules	AstraZeneca-Sanger DREAM & GDSC	MUT, target, drug & cell line-related features	Drug synergy

MUT: mutation data; CNV: copy number variation data; mRNA EXP: mRNA expression data; miRNA EXP: microRNA expression data; PROT: proteomics data; SMILES: Simplified Molecular-Input Line-Entry System strings; FP: molecular fingerprints; DSCRPTR: molecular descriptors; bRNN: bidirectional recurrent neural network; SCNN: stacked convolutional neural network; SA: self-attention; CA: contextual attention; MCA: multichannel convolutional attentive

3.5.1 Predicting Single-Drug Sensitivity

Unlike many traditional machine learning algorithms, DL can be used not only to predict the outcome of single-drug or combination anti-cancer treatments, but also to directly learn internal, lower-dimensional representations of the input data. This reduces the need for the explicit calculation of molecular features or extensive feature selection prior to training, as the features that are most predictive of drug response are automatically learned during the training process.

Some research groups have investigated the use of AEs for unsupervised feature learning and dimensionality reduction as a first step in the drug response prediction workflow. Ding et al. [90] recently showed that deep AEs are able to capture relevant information on the state of tumor cells prior to treatment. In this study, deep AEs were used to derive compressed representations from input data consisting of somatic mutations, copy number variations (CNVs), and gene expression data. The learned features were subsequently used to train elastic net classifiers to predict drug sensitivity in cancer cell lines. The encoded representations improved model performance, especially for drugs that were not well modeled when using either the original features or a selection of known genomic markers of drug sensitivity as input. The high sensitivity (true positive rate (TPR)) and specificity (true negative rate (TNR)) scores (Table 6) show that the model was able to correctly identify drug sensitivity/non-sensitivity, but the modest area under the receiver operating characteristic curve (AUROC) score achieved on an external test set suggests that the model had more difficulty generalizing to data from a different screen experiment.

Dr.VAE [131] is a semi-supervised approach that simultaneously trains a generative model of drug-induced changes in gene expression and a predictor that estimates drug response. As the name suggests, the method uses a VAE to create latent representations of the pre-treatment gene expression profiles. This latent representation is used to predict a latent representation of the corresponding post-treatment gene expression profile. Both latent representations are then fed into a logistic regression classifier to predict drug response. Dr. VAE was compared to other machine learning methods that were not trained on the same latent representations, and was shown to have achieved superior cross-validated AUROC scores. The joint modeling of drug perturbation signatures was key to improving the predictive performance of the drug response classifier.

Other research groups have used DL in a more end-to-end manner to predict drug response. Many of these models take advantage of the modular nature of DL networks when integrating multiple data types. Instead of concatenating different feature types into a single training dataset, distinct subnetworks for each type of data are constructed, and then the learned representations are fed into a final prediction subnetwork.

Table 6: Performance scores for some of the drug response prediction models for single compounds referred to in this review.

Model	Screening Dataset	Validation scheme	MSE	RMSE	R^2	r	r_s	AUROC	AUPRC	TPR	TNR
Ding et al. [90] (per-drug results)	GDSC	25-fold CV	1.96							0.82	0.82
Ding et al. [90] (per-cell line results)	GDSC	25-fold CV								0.80	0.82
Ding et al. [90]	CCLE	External test set						0.67			
Dr. VAE (per-drug models)	CTRPv2	100 train-validation-test splits (20 x 5-fold CV)						0.706	0.718		
DeepDSC [91]	CCLE	10-fold CV		0.23	0.78						
DeepDSC [91]	GDSC	10-fold CV		0.52	0.78						
DeepDSC [91]	CCLE	Leave-one-tissue-out		0.28	0.73						
DeepDSC [91]	GDSC	Leave-one-tissue-out		0.64	0.66						
DeepDSC [91]	CCLE	Leave-one-drug-out		0.61	0.05						
DeepDSC [91]	GDSC	Leave-one-drug-out		1.24	0.04						

Table 6 continued from previous page

Model	Screening Dataset	Validation scheme	MSE	RMSE	R^2	r	r_s	AUROC	AUPRC	TPR	TNR
DeepDR [92]	GDSC	100 train-validation-test splits	1.96 (median)								
PaccMann [87] (bRNN)	GDSC	25-fold CV (only different cell lines & drugs in test fold)		0.118							
PaccMann [87] (SCNN)	GDSC	25-fold CV (only different cell lines & drugs in test fold)		0.133							
PaccMann [87] (SA)	GDSC	25-fold CV (only different cell lines & drugs in test fold)		0.112							
PaccMann [87] (CA)	GDSC	25-fold CV (only different cell lines & drugs in test fold)		0.110							

Table 6 continued from previous page

Model	Screening Dataset	Validation scheme	MSE	RMSE	R^2	r	r_s	AUROC	AUPRC	TPR	TNR
PaccMann [87] (MCA)	GDSC	25-fold CV (only different cell lines & drugs in test fold)		0.120							
tCNNS [133]	GDSC	50 train-validation-test splits (80%/10%/10%) (different interaction pairs in each set)		0.027	0.826	0.909					
tCNNS [133]	GDSC	Leave-one-tissue-out		0.039	0.665	0.818					
CDRScan [136]	GDSC	train-test split (95%/5%)		1.069	0.843			0.98			
CDRScan [136]	GDSC	5-fold CV			0.847						

MSE: mean squared error; RMSE:root mean squared error; MAE: mean absolute error; R^2: coefficient of determination; r: Pearson correlation coefficient; r_s: Spearman's rank correlation coefficient; AUROC: area under the receiver operating characteristic curve; AUPRC: area under the precision-recall curve; TPR: true positive rate/recall/sensitivity; TNR: true negative rate/specificity; CV: cross-validation; bRNN: bidirectional recurrent neural network; SCNN: stacked convolutional neural network; SA: self-attention; CA: contextual attention; MCA: multichannel convolutional attentive

The DeepDSC model [91] uses DL methods both for dimensionality reduction and to predict drug sensitivity. It first employs a stacked deep AE to encode gene expression data characterizing the cancer cell lines as compressed representations. The encoded features were merged with pre-computed molecular fingerprints and then fed into a fully connected predictive network. DeepDSC was trained and evaluated on both GDSC and CCLE. For both datasets, they found that model performance decreased drastically when using the leaving drugs out of the training set, showing that the model is not generalizing well to compounds it has never seen before, despite having achieved relatively high R^2 scores when a 10-fold cross-validation scheme was used.

The modular nature of the models also allows for transfer learning, by reusing parts of networks that have been pre-trained on other datasets. DeepDR [92] is a DL model that predicts drug sensitivity based on the mutation and expression profiles of cancer cell lines or tumors. It consists of three subnetworks, the first two of which are a mutation encoder and a gene expression encoder, both pre-trained using data retrieved from the GDC database. The final subnetwork integrates the previous two and predicts IC_{50} values. The model was fully trained on cell line data from CCLE and GDSC, and then used to predict drug response in cancer patients. The DeepDR model outperformed linear regression models, SVMs, and DNNs without pre-training, having achieved lower mean squared error (MSE) scores (around an 80% improvement over linear regression and SVM models). Furthermore, by analyzing the genomic profiles of GDC patients predicted to be highly sensitive or resistant to a given drug, the authors showed that DeepDR was able to uncover the mechanisms underlying the response of cancer patients to well-known anti-cancer drugs, and was also capable of discovering potential therapeutic indications for novel compounds.

PaccMann [87], presented at the NIPS 2018 “Machine Learning for Molecules and Materials” workshop, is another end-to-end DL method for drug sensitivity prediction. It uses baseline gene expression data and SMILES representations of compounds to predict IC_{50}. SMILES enumeration [138], which takes advantage of the fact that several different SMILES strings can represent a given compound, was employed as a data augmentation strategy. The Paccmann model is composed of a gene expression encoder, a compound encoder and a prediction subnetwork which takes as input the encoded features. The gene expression encoder includes an attention mechanism, a method borrowed from the machine translation field which assigns higher weights to the most informative input features. For the compound encoder, the authors evaluated five different methods, having concluded that the encoders that include an attention mechanism performed best. The validation results for each of these models are shown in Table 6.

MOLI [132] is another DL model for drug response prediction that uses distinct encoding subnetworks for each data type. Three separate encoders learn representations for somatic mutations, CNVs and gene expression data, and a final subnetwork uses the concatenated features to classify the response of cancer cells to a given drug. MOLI was trained on cancer cell line data from GDSC, and validated on data from patient derived xenografts and patient samples (GDC). When training the network, the authors adopted a unique combined loss function consisting of a binary cross-entropy loss and a triplet loss, which helped improve model performance. The authors also observed that using multiple omics data was preferable to single omics data, and that integrating the different data types after representation learning led to

better performance when compared to approaches that merge different feature types before training. MOLI models were originally drug-specific, but the authors found that training on a dataset consisting of multiple drugs with the same target improved model performance.

The tCNNS model [133] also consists of separate encoding subnetworks - one which receives one-hot encoded representations of compound SMILES and another for genomic features. The two encoders are linked to a final prediction subnetwork. Both encoding subnetworks are 1D convolutional networks instead of the DNNs used in the previously described models. tCNNS performed well when predicting drug sensitivity for unknown drug-cell line pairs, having achieved high R^2 and Pearson correlation scores (Table 6) that were better than the previously reported models that it was compared to. However, the authors noted that the model performed much worse when making predictions for unknown drugs achieving R^2 and Pearson correlation values close to zero. This is similar to what has been observed for other models [91], and it is a problem that should be more carefully assessed in future studies.

KekuleScope [85] is a unique end-to-end method for drug response prediction. It uses transfer learning by employing CNNs that have been pre-trained on the unrelated ImageNet dataset [139]. These pre-trained CNNs were minimally modified and then trained on images of compounds (Kekulé structures) to predict drug sensitivity for eight cell line-specific cytotoxicity datasets from ChEMBL. KekuleScope was able to achieve similar performance to DNNs and RFs trained using Morgan fingerprints as input. The authors found that data augmentation by using modified versions of the compound images greatly improved the predictive performance of the models. The authors also observed that performance increased when the CNNs and the RF fingerprint-based model were joined in simple averaging ensemble. They suggest that this is an indication that the pre-trained CNNs were able to extract compound features containing information that is distinct from that encoded in Morgan fingerprints.

Combining multiple, complementary machine learning models is a strategy that has been shown to improve the predictive performance of drug response prediction models [15], even if only slightly [8]. Additionally, it increases model robustness [15]. Some recent drug response studies have included DL-based models in ensembles.

Matlock et al. [134] evaluated a variety of ensembles that were trained to predict drug sensitivity, some of which included DNNs. The authors found that an ensemble of five different base models, including a DL model trained on gene expression data, achieved the best performance overall.

Another heterogeneous ensemble, called ELDAP [135] is composed of four different base models, including a multitask DNN with two hidden layers. Apart from the ELDAP ensemble, this study also evaluated the performance of single models and ensembles composed of a single base learner. The results of this assessment showed that ensembles of multitask NNs were able to achieve significantly lower MSE values than single NN models on both the GDSC and CCLE datasets.

CDRscan is an ensemble entirely composed of DL base learners [136]. It consists of five CNNs, four of which adopt a "dual convergence architecture". In the dual convergence models, a series of convolutions are performed on each input data type (mutations and molecular fingerprints) separately. The two sets of convoluted features are then merged and convolution is applied again before predicting

the IC_{50} values for cell line-drug pairs from the GDSC. CDRscan achieved high predictive performance, showing a considerable improvement in terms of R^2 when compared to RF and SVM models (21% and 50% improvement, respectively). The authors found that base models that adopted the dual convergence architecture performed better. They also noted that two of the base models performed better than the ensemble, which suggests that individual DL models might sometimes be sufficient. However, they argue that the ensemble approach improves the robustness and generalizability of the model.

3.5.2 Predicting Drug Combination Effects

After an initial period of responsiveness, the efficacy of drug-based anti-cancer therapies is often reduced due to the existence of intrinsic or acquired tumor drug resistance mechanisms [13]. A common strategy to overcome drug resistance is the administration of two or more drugs in combination. Drug synergy, the enhancement of combination effects beyond expected, is a highly desirable outcome, as it may increase treatment efficacy without requiring an increase in drug dosage [14]. Once again, computational methods could greatly reduce the experimental effort required, but predicting drug combination effects is an even more complex problem than the prediction of sensitivity to single drugs.

A recent study [82] was the first in the literature to propose the use of DL to predict drug combination effects. The model, named DeepSynergy, is a DNN that uses drug response data, chemical features and gene expression data to predict a drug synergy score. The model was trained using pharmacological data from the Merck Compound Screen [6] and omics data from GDSC [12]. The DNN achieved relatively high performance scores. Wilcoxon signed rank-sum tests confirmed that DeepSynergy performed significantly better than the gradient boosting, RFs, SVMs, and elastic net models it was compared to, in terms of several different scoring metrics. It was able to achieve a MSE that was 7.2% lower than the second-best method (gradient boosting), and a 5.8% improvement in terms of Pearson correlation. Nevertheless, the authors noted that DeepSynergy, as well as all of the other evaluated methods, had difficulties predicting drug synergy when presented with data from previously unseen cell lines or compounds [82]. DeepSynergy was also evaluated as a classifier, and results for a variety of scoring metrics were reported. While the high AUROC score seems to be an indication of good predictive performance, other metrics show that this might be overly optimistic. This makes clear that assessing multiple scoring metrics is necessary to fully evaluate the performance of a model.

Table 7: Performance scores for some of the drug response prediction models for drug combinations referred to in this review.

Model	Drug Response Dataset	Validation scheme	MSE	RMSE	MAE	R^2	r	r_s	AUROC	AUPRC	TPR	TNR	ACC	BACC	PREC	κ	F1
DeepSynergy [82]	Merck Compound Screen	5-fold stratified nested CV (leave-combinations-out)	255.49	15.91			0.73		0.90	0.59	0.57	0.95	0.92	0.76	0.56	0.51	
DeepSynergy [82]	Merck Compound Screen	Leave-drugs-out	435.92	19.90			0.48										
DeepSynergy [82]	Merck Compound Screen	Leave-cell-lines-out	405.40	18.74			0.57										
Xia et al. [137]	NCI-ALMANAC	5-fold stratified CV	0.0158		0.0833	0.9440	0.972	0.965									
Chen et al. [96]	AstraZeneca-Sanger DREAM challenge	Leave-one-out									0.602				0.715		0.654

MSE: mean squared error; RMSE:root mean squared error; MAE: mean absolute error; R^2: coefficient of determination; r: Pearson correlation coefficient; r_s: Spearman's rank correlation coefficient; AUROC: area under the receiver operating characteristic curve; AUPRC: area under the precision-recall curve; TPR: true positive rate/recall/sensitivity; TNR:true negative rate/specificity; ACC: accuracy; BACC: balanced accuracy; PREC: precision; κ: Cohen's kappa; F1: F1 score; CV: cross-validation

Another promising DL model for the prediction of drug combination effects was proposed by Xia et al. [137]. The model was trained on data from the large A Large Matrix of Anti-Neoplastic Agent Combinations (ALMANAC) [7] dataset. Similarly to many of the previously described drug sensitivity models, this model consists of separate feature encoding subnetworks for each type of input data (drug descriptors, gene expression, microRNA, and proteomics data), and a final prediction subnetwork that estimates growth inhibition. All of the weights and layers in the drug descriptor subnetwork are shared between the two drugs in a given drug combination, allowing data from single-drug experiments to also be used as input. The model was able to achieve R^2, Pearson correlation and Spearman correlation values that were quite high (see Table 7) when compared to models built for other drug discovery-related prediction tasks. Furthermore, the model was able to correctly identify the majority of the most promising drug combinations. Unfortunately, the model was only evaluated through 5-fold cross-validation. It would have been interesting to see if the model could maintain its high predictive performance when applied to previously unseen drugs or cell lines. A performance comparison to other state-of-the-art methods trained on the same dataset is also missing.

The supplementary information provided alongside the recently published AstraZeneca-Sanger Drug Combination DREAM challenge paper [8] also briefly mentions the use of a multimodal DL model to predict drug combination effects. The post-challenge model developed by the DMIS team included four subnetworks that separately encode mutations, target proteins, chemical and pharmacological features, and other cell-line related features, and a final prediction network that predicts a continuous synergy score. When evaluated on an external dataset, the model outperformed the original model developed by the team and other top-ranking challenge submissions evaluated on the same test set.

Chen et al. used a completely different DL architecture, DBNs, to classify the effects of drug combination experiments as synergistic or non-synergistic [96]. The study used baseline gene expression data and drug target information from the AstraZeneca-Sanger Drug Combination DREAM challenge [8]. Unlike many of the previously mentioned studies, drug target information played a essential role, having been used to derive ontology fingerprints [140] for the target genes and information on the pathways targeted by each drug. The authors claim that their approach outperformed other ML models submitted to the DREAM challenge, having achieved higher precision, recall and F1 scores. We note, however, that the DBN model was evaluated using the leave-one-out approach, while the challenge submissions were evaluated on an external dataset to which the participants did not have access.

3.5.3 The Potential for Drug Repurposing

Drug repurposing refers to the act of discovering new therapeutic indications for existing drugs that were originally intended for other purposes [141]. The main advantage of this strategy is that these compounds have already been well-studied in terms of their pharmacokinetic and safety profiles, which can help accelerate drug development. Several DL models for *in silico* drug repurposing have already been reported in the literature [142]–[144].

Besides predicting how cells respond to novel drugs, the drug response models described in this review can also be used for drug repurposing. The large screening initiatives mentioned in section 3.3 have screened both experimental compounds and drugs that have already been approved. Therefore, if a high-performing model trained on one of these large screening datasets predicts that a given cell type is sensitive to a drug that was not approved for that particular cancer type, it may be an indication of the repurposing potential of that particular drug. Furthermore, databases of approved drugs can be "screened" *in silico*, using the drug sensitivity models to predict how cancer cell lines would respond to these compounds. The authors of CDRScan [136] studied the drug repurposing potential of their model by using it to predict the response of 787 cancer cell lines to over a thousand approved drugs from the DrugBank database [118]. CDRScan predicted that 23 known anti-cancer drugs were also active against at least one other cancer type besides their approved indications. It was also able to uncover potential anti-cancer indications for 27 non-oncological drugs [136]. A similar approach could be employed to determine the drug repurposing potential of other DL-based drug sensitivity prediction models.

3.6 Model Evaluation

When evaluating the performance of a drug response prediction model, the choice of an adequate validation strategy is essential to ensure that the model is able to generalize well to new drugs and cell lines. Many of the models surveyed in this review used some form of cross-validation to evaluate model performance and generalizability, but not all of them guarantee that the validation set only includes previously unseen cell lines and drugs/drug combinations (see Tables 6 and 7. Preuer et al. [82] and found that their DeepSynergy model performed poorly when predicting drug response for test cases with drugs or cell lines that were distinct from those seen during the training phase. This highlights the importance of using more rigorous validation schemes, such as "leave-drug-out" or "leave-cell-line-out" (Figure 9).

When using a "leave-drug-out" cross-validation scheme, one or more compounds are held out from the training set. This is repeated until all of the drugs have been held out once. The results of this type of validation more accurately reflect how a model would behave when presented with a new compound.

The "leave-cell-line-out" validation strategy leaves out cell lines from the training set, while a "leave-tissue-out" validation scheme leaves out all cell lines of a particular tissue type. Leaving specific cell lines or tissue out from training is important to assess whether the model is just learning to identify the cell line or the tissue type based on the omics features and associating it with a certain response [90]. A useful drug response prediction model is one that is able to uncover tumor-specific associations between omics features and response, reflecting the heterogeneity of responses [12] that is often seen even among tumors of the same type.

Many of the studies also used external datasets from different screening initiatives with some degree of overlap with the training dataset to further validate the models. Since different datasets may have been generated using different screening methodologies, this overlap can be particularly useful to assess the

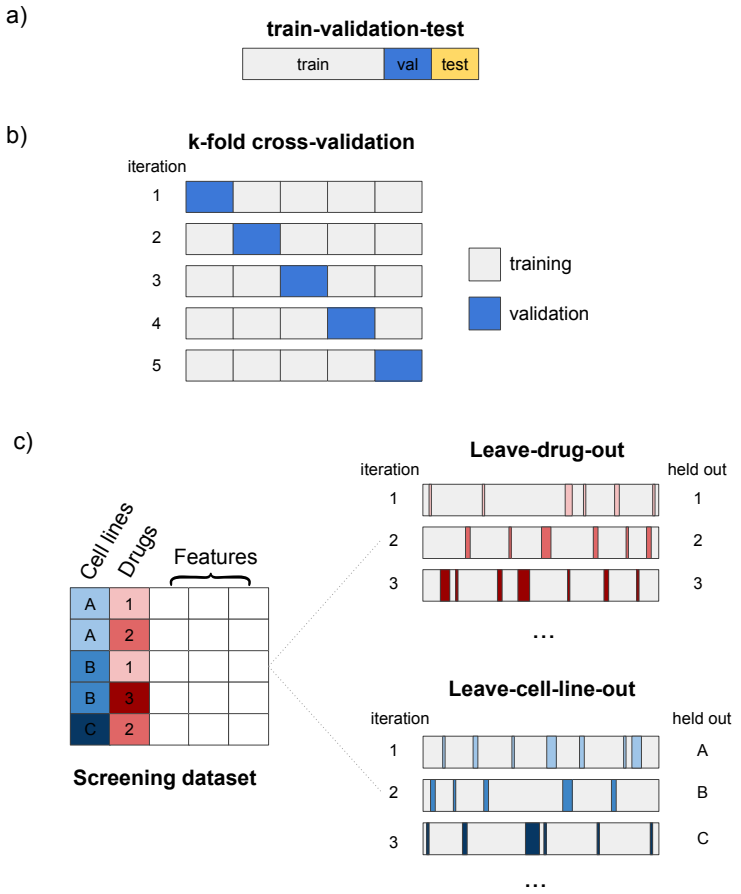


Figure 9: Different model validation schemes. a) A dataset can be split once or multiple times into training, validation and test sets. b) In k-fold cross-validation, a dataset is split into k different folds, and in each iteration a different fold is used as the validation set. In these cases, the held out validation sets consist of randomly selected drug-cell line interaction pairs. c) The leave-cell-line-out and leave-drug-out validation methods hold out one or more specific cell lines or drugs when training and use them as the validation set.

robustness of the models with respect to different experimental conditions. A model can be considered robust if it is able make similar predictions for the same interaction pairs in different datasets, and if the same candidate drug response biomarkers are identified irrespective of the experimental setup.

Another specific validation step reported in some of the studies is evaluating if a model is able to identify gene-drug associations that are have already been well-studied in the literature. This step helps to confirm the clinical usefulness of the response biomarkers identified by the model.

The choice of adequate scoring metrics to evaluate and compare different drug response prediction models is also crucial. Some commonly used metrics such as MSE/root mean squared error (RMSE) are dataset-specific and may not be the best for comparing between models trained on different datasets, for instance. Problem-tailored scoring metrics such as the weighted scoring metrics proposed in past DREAM challenges [8], [15] may provide a more accurate idea of the predictive performance and generalization capacity of the models. In any case, the use of multiple scoring metrics is recommended, as different scoring metrics may offer different insights.

Most of the reviewed studies also compared their proposed DL models to traditional ML models. However, the hyperparameter space for the ML models is usually not as extensively explored and the models may be trained using slightly different input data, as feature selection techniques may need to be employed. Furthermore, some studies do not directly compare the proposed models to the state-of-the-art ML-based drug response prediction models. This may lead to optimistic results showing that DL models perform better, when in fact the DL models are not being compared to the best possible traditional ML alternatives.

Now that several DL-based drug response models have been published, it would also be interesting to compare them with one another. However, as can be seen in Tables 6 and 7, different studies use different scoring metrics and validation strategies, with little overlap between studies. This highlights the importance of defining a benchmarking strategy for the community, so that performance can be easily compared across methods.

3.7 Improving Model Performance

Although these initial studies have demonstrated that DL drug response models are usually able to outperform traditional ML models, there is still room for improvement. For instance, although several DL models have already achieved R^2 scores greater than 0.80 (see Tables 6 and 7) when predicting drug response for unknown drug-cell line pairs, drug response prediction models in general still struggle to generalize well to novel cell lines and drugs.

Diverse neural network architectures have been used to build drug response prediction models, from simple DNNs, to CNNs or natural language processing (NLP)-inspired approaches. One of the main advantages of using DL to predict drug response is that DL models can learn to learn higher-order representations directly from input data. Many of these models have already taken advantage of this unique characteristic of DL models. However, none have tested the use of graph convolutions [79], [145]. Graph convolutional networks are able to learn representations of compound structures represented as molecular graphs, and have been used in various other drug discovery prediction tasks [146]. Besides graph convolutions, other DL methods to learn continuous representations of compounds, such as the NLP-inspired Seq2seq fingerprint [147] for example, have yet to be explored.

As demonstrated in some of the studies reviewed in this article, DL can also be used to learn representations of cell line omics data. The development of suitable DL methods for feature extraction from these types of data should continue to be investigated.

It is important to note, however, that it is still unclear if features learned directly from raw input data always perform better than manually-engineered features. Two recent studies [148], [149] have reached different conclusions when comparing the performance of learned representations of compounds and pre-computed molecular descriptors and fingerprints. Another study found that learned representations

of compounds do not perform well when the training dataset is small or highly imbalanced [146]. Furthermore, manually-engineered features may be more interpretable.

In addition, previous drug response prediction studies [8], [15] have observed that the type of ML algorithm used does not have a considerable effect on the predictive performance of the models, as opposed to the way the input data is processed and integrated into the model.

Most of the models referred to in this review still use prior biological knowledge such as pathway data to filter out less relevant features and to refine the models. Some preprocessing and filtering of the input datasets is still necessary due to the fact that these drug response datasets are multidimensional and very noisy, especially the omics datasets characterizing the cell lines. The success of DL models in this field will probably depend on the successful combination of learned representations of the original data and carefully selected manually-engineered features.

Drug response prediction models are usually trained on heterogeneous datasets, comprising both pharmacological data and various types of omics data. Integrating multiple input data types is a complex task. Each feature type must be preprocessed differently before being fed into the network, and variable types and value ranges may greatly differ between different data types. The problem of biological data integration has been addressed in more detail in another review [150]. Several DL drug response models [87], [92], [132], [133], [136], [137] have adopted a multimodal learning approach, in which separate subnetworks with distinct architectures are used to model different input data types. In the future, most, if not all, DL-based drug response models will probably follow a similar multimodal, modular approach.

Besides the choice of network architecture, assessing which combinations of different data types are the most predictive is essential. Gene expression data is by far the most commonly used input data type for drug response models. Several studies have found that gene expression is indeed the most informative data type [12], [15], but a recent DL-based study found that drug descriptors were more predictive than cell line features [137]. One study also found that, in a tissue-specific setting, genomic features were more predictive of drug response than gene expression [12]. Furthermore, certain types of data may be more readily available in a clinical setting than others, and it is important to also take this into consideration when selecting the input data types for these models.

There are many sources of data that could be valuable when building drug response prediction models, but leveraging these data may not always be feasible. For instance, the lack of target information for compounds may limit the use of pathway data. Integrating additional data may also be difficult due to the lack of sufficient overlap between different resources. Tan et al. [135] observed that the lack of overlap (in terms of compounds) between LINCS and the drug screening datasets from GDSC and CCLE was a limiting factor. Semi-supervised learning can be used to overcome this problem in some cases [131].

Since DL models are complex and have many learnable parameters, they have a tendency to overfit. Due to this, DL models perform better when trained on large amounts of data [69], [84], [151]. The scarcity of sufficiently large anti-cancer drug screening datasets has limited the use of DL for drug response prediction. However, it is important to note that the need for large, high-quality drug screening datasets is common to all types of ML-based drug response prediction models [16], and not just DL models. Larger

and more diverse training sets are therefore necessary to be able to obtain drug response models that can generalize well to a wider range of drugs and cell lines than is currently possible.

Integrating several drug screening datasets could be a way to create datasets with a greater amount of samples representing a wider variety of drugs and cell lines. However, this may not be a simple task. Data from different screening initiatives may have been obtained using different experimental methods and under different conditions, and the way drug response is quantified may be different. It may also be necessary to standardize gene and compound identifiers to guarantee that they are uniform across the different datasets. Furthermore, some authors have expressed concerns regarding the inconsistency between different drug screening datasets [152], [153], although others are more optimistic [127], [154], [155].

An alternative way to increase the size of the training set may be to employ data augmentation techniques, such as the SMILES enumeration technique [138]. DL methods such as generative adversarial networks (GANs) [156] could also be used for data augmentation purposes by generating artificial data based on the original training set [157].

Transfer learning can also help mitigate the effects of small training datasets. Pre-training compound encoder subnetworks using larger, more general compound datasets may help to create more robust compound encoders. Omics feature encoders can also be pre-trained in a similar manner. In addition, using patient-level data to pre-train parts of the networks can aid in the extraction of clinically relevant features. This type of approach might help overcome the differences between the cell type contexts in cell lines and patients that prevent translatability to the clinic. Nevertheless, it is still unclear if DL models will be able to fully resolve this issue.

The majority of the models described in this review are not cell line or drug-specific models, but integrate drug response data from a variety of compounds screened against of variety of cancer cell lines representing different types of tumors. Training pan-drug and pan-cancer drug response prediction models is necessary because the amount of data available for certain cancer cell lines or drugs that are underrepresented in the available screening datasets may be insufficient to train robust DL models for these cases. It may also help the model to learn more general features, increasing model generalizability. These general models can be subsequently fine-tuned using smaller, more specific datasets (limited to a specific tumor type, or a certain class of drugs), enabling the creation of models that are more specific and clinically useful.

Finally, as the complexity of DL drug response prediction models increases, the financial and environmental costs associated with building these models will also increase substantially [158], [159]. In the future, balancing model performance and clinical applicability with these additional factors will become increasingly important.

3.8 Model Interpretability

One of the main characteristics of DL models that has limited their application to biological and health-related problems in general is their lack of interpretability [69], [84], [151], although some of the traditional ML models commonly used in pharmaceutical research also have this problem [69], and the quality of the input data itself can also affect the biological interpretability of model predictions [16]. While the ability to extract relevant features directly from raw data may be considered an advantage of using DL, the abstract learned representations may be very difficult for humans to interpret.

DL models are generally considered "black boxes", but the ability to easily to interpret the results of a drug response model is essential for its acceptance and application in the pharmaceutical industry or in the clinic. Ensuring that these models are interpretable would improve our understanding of the factors underlying drug sensitivity/resistance or drug synergy, rather than merely predicting drug response. Only a few of the studies mentioned in this review addressed the issue of interpretability, but if DL-based drug response models are to be relevant, techniques to make these types of models more interpretable must continue to be explored.

A step in this direction is to build models trained on different combinations of input data to find out which data types are the most predictive, as described in the work of Xia et al [137], for example. However, although this provides an idea of which datasets might be the most informative, it still does not reveal how specific features influence prediction.

Determining which input features contributed most to a certain prediction would help to uncover potential drug response biomarkers or allow the identification of structural features of drugs that are associated with improved drug response. Discovering new biomarkers of drug response is particularly important for precision medicine, as its main purpose is to help to stratify patients for treatment [12]. Furthermore, putative biomarkers identified in cell lines and confirmed in patients could be a way to translate the insights gained from cell line models to the clinic. Confirming that a well-known biomarker-drug association is still observed when a given model trained on cell lines predicts drug responses for patients (e.g. greater sensitivity predicted for patients with a known biomarker than those without) would also be an additional confirmation of the translatability of the model.

The field of "explainable artifical intelligence" is a very active one, and numerous strategies to increase the explainability of DL models have been proposed. Including one of these methods into the drug response prediction workflow may help identify the most predictive features. We refer readers to a recent article on model interpretability for a more comprehensive review on the topic [160].

Some of these strategies determine feature importance by perturbing specific input units and evaluating the effects on the outputs. Local Interpretable Model-agnostic Explanations (LIME), for instance, uses an additional ML model to learn important features based on the predictions of DL models [40]. Another perturbation-based method applies dropout, a technique which ignores certain units in a network during training, to the input layer to rank input features [41]. Another class of model interpretability methods are backpropagation-based approaches, which propagate importance backwards through the network using

gradients [161]. Examples include the Integrated Gradients method [162] and DeepLIFT [161].

SHapley Additive exPlanations (SHAP) [163] unifies several popular model explanation methods, including LIME and DeepLIFT, and Shapley value estimation methods from the field of game theory under a single framework. It approximates Shapley values [164] to determine the contribution of each feature to a given prediction [163].

Incorporating attention mechanisms within the network itself is a more direct way of making a DL model more explainable. Attention mechanisms assign weights to the input features during training. In PaccMann [87], learned attention weights allowed the identification of the specific genes or the specific atoms and bonds within a molecule that were most predictive of drug response.

Methods to increase the interpretability of DL models can also be borrowed from other subfields of artificial intelligence. For example, given input examples and background knowledge, inductive logic programming (ILP) models learn logic programs [165] which are much more humanly interpretable than NNs. Combining ILP and DL [166] may therefore produce models that are more easily understood.

The increased complexity of DL models can lead to greater predictive performance, but it is also what makes these models very difficult to explain. Due to this trade-off between accuracy and interpretability, better-performing, complex models will probably always be more difficult to explain than simpler models, even if the previously suggested methods to improve interpretability are employed. Some authors argue that designing models that are inherently interpretable should be preferred over attempting to explain "black box" DL models a posteriori [167]. However, recent efforts in the field of explainable artificial intelligence, such as self-explaining neural networks [168], have shown that it may be possible to build DL models that are interpretable by design.

3.9 Conclusion

In this article, we reviewed some of the first studies that have employed DL to predict the effects of single drugs and drug combinations on cancer cell lines. The results of these initial studies are promising, demonstrating that DL-based models are able to perform as well as, or even better than, traditional ML-based models on drug response prediction tasks. Nevertheless, there is still room for improvement, and the future success of DL in this field will depend on the improvement of both the generalizability and the interpretability of these models.

Many of the studies reviewed here highlight the potential clinical applicability of the DL-based drug response models they report on, but direct application to patients in a clinical setting is still far from being a reality. Few of these studies have been able to demonstrate how cell line models could actually translate to patients. Due to the lack of interpretability of the models and the fact that most were trained on cell line data only, the current results do not substantiate these claims. Furthermore, we must also consider that the variety of omics data types characterizing cancer cell lines may not be readily available when it comes to patients, as obtaining samples from tumors or performing certain analyses may not always be

possible.

Although they are still far from being clinically applicable, these models help us get closer to achieving the goal of precision medicine in the clinic. The results of these studies will be useful to direct future efforts towards the development of computational methods for the rational design of novel effective anti-cancer treatments.

4

A comparison of different compound representations for drug sensitivity prediction

This chapter is an extended version of a paper [169] (written in co-authorship with João Correia, Bruno Pereira and Miguel Rocha) that was presented at the 15th edition of the Practical Applications of Computational Biology & Bioinformatics (PACBB) conference.

4.1 Introduction

In recent years, ML has become an important tool for computer-aided drug design. Quantitative structure-activity relationship (QSAR) modeling, for example, typically uses ML models to predict molecular properties or the bioactivity of compounds. ML has been an especially popular choice for the modeling of drug response in cancer [170], [171].

One of the first steps in a ML workflow for drug discovery is the selection of suitable input features. Chemical compounds are typically represented using molecular descriptors or molecular fingerprints (Figure 10). Molecular descriptors are the experimentally obtained or theoretical physical and chemical properties of a compound. Molecular fingerprints encode molecules as bit or count vectors. The type of information that is encoded depends on the type of fingerprint. Substructure key-based fingerprints set the bits of the bit vector to one or zero depending on the presence or absence in the compound of certain substructures or features from a list of predefined structural keys [172]. Path-based fingerprints enumerate the different linear paths between atoms in a molecule to determine which types of fragments are present. Circular fingerprints describe the surrounding environment of each atom in the molecule up to a predefined radius [172].

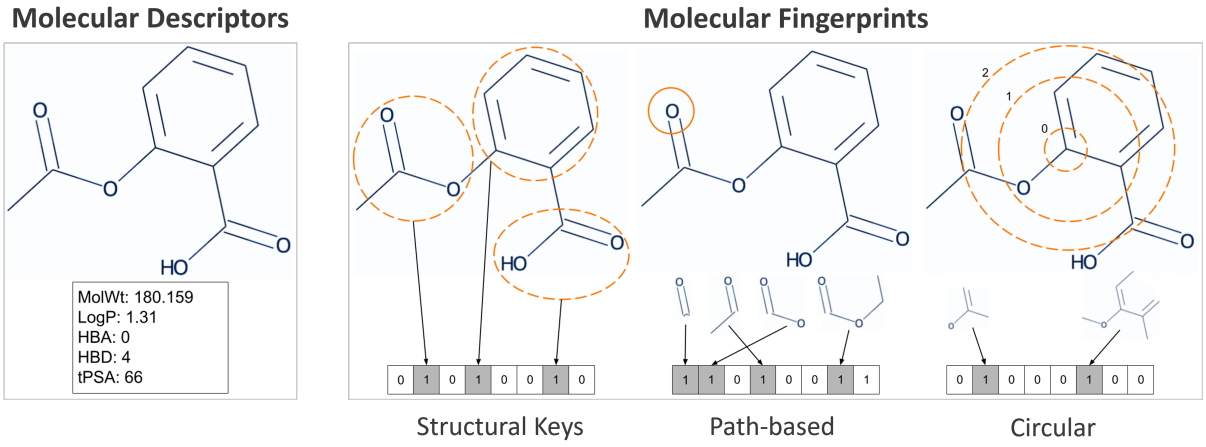


Figure 10: Traditional representations of compounds for machine learning tasks.

Another alternative is to use end-to-end DL approaches that can learn relevant features directly from raw input data, eliminating the need for explicitly calculated descriptors and fingerprints (Figure 11). Certain types of DL algorithms are able to learn directly from line notations, such as SMILES strings. RNNs and 1D CNNs are particularly well-suited to this type of sequential input. Additionally, natural language processing methods can be applied to sequential inputs like SMILES strings to create continuous embeddings of molecules. Graph neural networks (GNNs) are able to learn features from compound structures represented as molecular graphs.

In recent years, several benchmarking studies have been undertaken to determine whether learned representations of compounds perform better than traditional descriptors and fingerprints. Hop et al. found that GNNs outperformed fully-connected neural networks (FCNNs) trained on molecular fingerprints on a majority of benchmark tasks [149]. In contrast, another study [148] reached the opposite conclusion: learned representations performed worse than traditional features. An extensive benchmarking study using the MoleculeNet collection of datasets concluded that learned representations do not perform well when the training dataset is small or highly imbalanced [146]. Therefore, the most suitable representation for a given prediction problem probably depends on the type of problem itself, as well as other factors such as dataset size, making it essential to evaluate this for each specific application.

In this study, we tested several compound featurization methods and DL algorithms to try to understand which representations are the most suitable for predicting drug sensitivity in cancer cell lines. We used a new Python package, developed in-house and named DeepMol, to perform this analysis. All of the datasets and scripts used for this study are available online at https://github.com/BioSystemsUM/DeepMol/tree/pacbb21/pacbb21_paper.

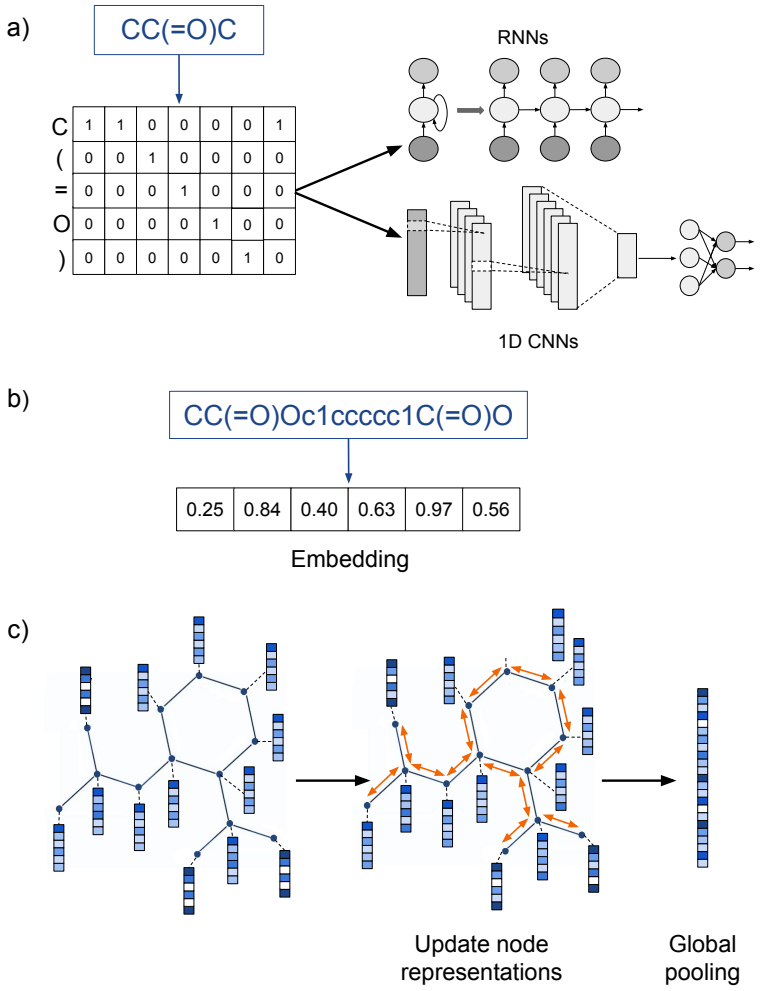


Figure 11: Learned representations of compounds. a) Representations can be learned directly from SMILES strings using DL architectures such as RNNs or 1D CNNs. b) Natural language processing-inspired methods can be used to generate continuous embeddings of molecules. c) GNNs can be used to learn representations directly from molecular graphs.

4.2 Methods

4.2.1 Data sets

We benchmarked a selection of DL models on several human cancer cell line drug screening datasets (Table 8). Single-cell line datasets were used so that it would be possible to study the effect of different compound representations without having to take cell line features into account.

The NCI 1 and NCI 109 human tumor cell line growth inhibition datasets were used to develop binary classification models. These datasets were chosen because they have been widely used in the literature to validate graph classification algorithms. In our case, we opted to use balanced versions of these datasets [173], available from https://github.com/shiruipan/graph_datasets. In each dataset, the output variable indicates whether a given compound is active or inactive in a specific cell line.

To develop and evaluate regression models, we used two single-cell line cytotoxicity datasets (PC-3

Table 8: Details on the datasets used in this study.

Dataset	Compounds	Output Variable	Task Type
NCI 1	3466	sensitive/not sensitive	classification
NCI 109	3431	sensitive/not sensitive	classification
PC-3	4294	$-\log(\mathrm{IC}_{50})$	regression
CCRF-CEM	3047	$-\log(\mathrm{IC}_{50})$	regression
A549/ATCC	20730	$-\log(\mathrm{GI}_{50})$	regression

and CCRF-CEM) from a recent drug sensitivity prediction study [85]. The authors of this study obtained the original datasets from ChEMBL [174], performed some filtering and data cleaning steps, and transformed the original IC_{50} values into $-\log(\mathrm{IC}_{50})$ values (pIC_{50}). These pIC_{50} values were used as the output variable in our regression models.

The previously mentioned datasets are all relatively small, each comprised of less than 5,000 compounds. However, these smaller datasets were preferred because we assumed that they would more closely reflect the behavior of the compound representation methods when used in drug sensitivity prediction models trained on publicly available anti-cancer screening datasets, which usually have data for even fewer compounds. Indeed, the original version of the popular GDSC resource [4], for example, provides access to a dataset (GDSC1) containing screening data for only 367 compounds, while the CTRPv2 [5] dataset has data for only 481 compounds.

Nevertheless, we also evaluated DL models on a larger dataset derived from the NCI-60 dataset. We selected the A549/ATCC cell line and, after removing low quality experiments, compounds without sensitivity data, and compounds that we were unable to map to SMILES strings using the files provided by the Developmental Therapeutics Program (DTP), we obtained a dataset with sensitivity values, measured as $-\log\mathrm{GI}_{50}$ (growth inhibition of 50% of cells) for 20,730 compounds.

Prior to modeling, all SMILES strings were preprocessed using the ChEMBL Structure Pipeline [175].

4.2.2 Models

4.2.2.1 Pre-computed features

We evaluated six different types of molecular fingerprints in this work: extended connectivity fingerprint (ECFP) (ECFP4 and ECFP6), Molecular ACCess System (MACCS) keys, atom pair fingerprints (AtomPair), RDKit fingerprints (RDKitFP) and RDKit layered fingerprints (LayeredFP). ECFP fingerprints [176] are a popular circular fingerprint based on the Morgan algorithm [177]. ECFP4 fingerprints use a radius of 2 to define the circular neighborhood surrounding each atom, while ECFP6 fingerprints use a radius of 3. MACCS is a type of substructure key-based fingerprint which uses 166 predefined keys [178]. The AtomPair fingerprint is a topological fingerprint based on determining the shortest distance between all pairs of atoms within a molecule. [179]. The RDKitFP is another topological fingerprint that was developed by the RDKit [180] project. The algorithm finds all subgraphs in a molecule containing a number of bonds

within a predefined range, hashes the subgraphs, and then uses these hashes to generate a bit vector of fixed length. The LayeredFP [180] uses the same algorithm as the RDKitFP to identify subgraphs, but different bits are set in the final fingerprint based on different "layers" (different atom and bond type definitions).

Compound structures were encoded as bit vectors using each fingerprinting algorithm and the resulting fingerprints were used as inputs to FCNNs. With the exception of MACCS fingerprints, which have a fixed length, we limited the size of all fingerprints to 1024 bits.

4.2.2.2 Mol2vec embeddings

Mol2vec is an unsupervised method that generates continuous vectors representing molecules using the Word2vec word embedding algorithm [181]. Each molecule is considered a "sentence" and molecular substructures (calculated using the Morgan algorithm) are considered "words". We used a pre-trained Mol2vec model to generate 300-dimensional embeddings for the molecules in each dataset, and fed these embeddings into FCNNs.

4.2.2.3 TextCNN

TextCNN is a 1D CNN that was originally developed for sentence classification [182]. We used a modified version of this algorithm (implemented in DeepChem [100]) which uses one-hot encoded SMILES strings as inputs instead of words. It applies several 1D convolutional filters, followed by a max-over-time pooling operation, which summarizes each filter using its maximum value. These learned features are then fed into fully-connected layers to predict the output.

4.2.2.4 Graph Neural Networks

The structure of a chemical compound can be represented as a molecular graph, where nodes are atoms and edges represent bonds. Conventional NN architectures, such as FCNNs or CNNs, are unable to learn directly from this type of data.

GNNs generalize deep neural networks to graph-structured data. Inputs to a GNN are usually node features (e.g. atom type) and adjacency matrices encoding the structure of the graph, and sometimes can also include edge features as well. The node and edge features are used to initialize the graph. In general, GNNs apply learnable functions to update the node-level representations, progressively incorporating information about the neigborhood of a node into its representation. After several rounds of updates, a pooling operation can be used to obtain a graph-level (molecular-level) representation. The graph-level representations can then be fed into fully-connected layers to predict a given output.

In this work, we benchmarked four different GNN algorithms: neural fingerprints (GraphConv) [79], graph convolutional network (GCN) [183], graph attention network (GAT) [184] and the AttentiveFP algorithm [185].

4.2.3 Model training and evaluation

The modeling workflow that was followed in this study is shown in Figure 12.

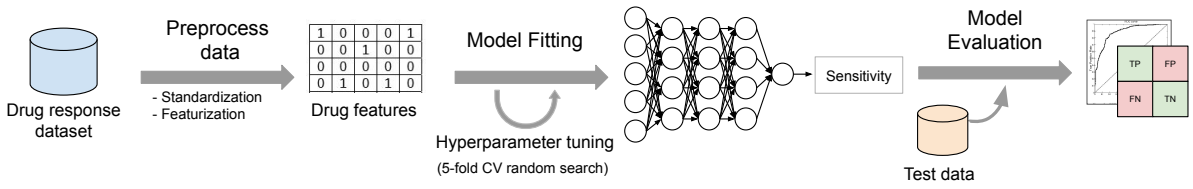


Figure 12: General data preprocessing and machine learning steps followed in this work.

Each dataset was split into a training set (70%) and a test set (30%). All models were trained and evaluated using the same splits for each dataset.

All models were trained for 100 epochs with a batch size of 256 samples, and used the Adam [36] optimization algorithm. Binary cross-entropy was used as the loss function for all classification models, while the mean squared error was used for regression models. Other model-specific hyperparameters were tuned using a 5-fold cross-validated randomized search, in which 30 different hyperparameter combinations were tested. These included the number of hidden layers and hidden units and the use of regularization methods, such as L2 weight regularization and dropout [37], among others. The best model that was found for each algorithm was then refit on the entire training set and evaluated on the held-out test set. Additional details on the models (including the full search space and the best hyperparameters found for each type of model) are available online (https://github.com/BioSystemsUM/DeepMol/blob/pacbb21/pacbb21_paper/supplementary_material.pdf).

4.2.4 Model ensembling

Besides evaluating individual models, simple voting ensembles were also built to determine if combining predictions from multiple models could further improve performance. For classification tasks, majority voting ensembles were built, where the final prediction is the label predicted by the majority of the individual classifiers. For regression tasks, the ensembles simply averaged the individual predictions to obtain the final prediction. The hyperparameter values that were used for each individual model were the optimal values that had been previously tuned using randomized search.

4.2.5 Feature importance

Feature importance was determined using the Deep SHAP method implemented in the SHAP Python package (version 0.39.0) [163]. Deep SHAP approximates SHAP values for DL models by using a modified version of the DeepLIFT [161] algorithm.

4.2.6 DeepMol

All preprocessing, featurization and modeling steps were implemented using DeepMol, a newly developed chemoinformatics package developed from our host group. DeepMol is a python-based machine and deep learning framework for drug discovery. It offers a variety of functionalities that enable a smoother approach to many drug discovery and chemoinformatics problems. This framework uses Tensorflow [98], Keras [99], Scikit-learn [186] and DeepChem [100] to either build custom ML and DL models or make use of pre-built models. It also uses the RDKit [180] framework to perform operations on molecular data.

Regarding compound standardization, it allows users to use the ChEMBL Structure Pipeline [175] or apply custom standardization steps using RDKit [180] standardization methods. Some of these steps include standardization of some non-standard valence states, molecule sanitization, charge neutralization, stereochemistry removal, the removal of smaller fragments, kekulization, among others.

DeepMol also offers several featurization methods including molecular fingerprints, molecular embeddings and graph-based featurizers.

In summary, DeepMol offers a complete workflow to perform machine and deep learning tasks for molecules represented as SMILES strings. It has modules that perform standard tasks, such as loading and standardizing data, computing molecular features, performing feature selection and data splitting. It also provides methods to deal with unbalanced datasets and to do unsupervised exploration of the data. This way, DeepMol provides a common platform to treat the data and build, train, optimize and evaluate ML and DL models using different ML frameworks.

4.3 Results and Discussion

4.3.1 Individual models

In this section, we report and discuss the performance of 12 DL algorithms benchmarked on 5 drug response datasets of variable size and with different output variables. Figure 13 reports the results for classification tasks, with model performance quantified using the area under the receiver operating characteristic curve (ROC-AUC). For regression problems, model performance scores are reported in Figure 14, using the RMSE values. The full results tables and plots for additional scoring metrics are available from https://github.com/BioSystemsUM/DeepMol/tree/pacbb21/pacbb21_paper/results.

ECFP4 fingerprints outperformed other fingerprints and end-to-end DL models on the NCI 1 classification task, having achieved a ROC-AUC score of 0.83. The LayeredFP model achieved a very similar ROC-AUC score (also 0.83, when rounded), while the best end-to-end DL model (TextCNN) reached a score of 0.81. On the NCI 109 dataset, the GCN algorithm ranked first in terms of performance, but other methods such as LayeredFP, TextCNN and GraphConv were not far behind (Figure 13). With the exception of the LayeredFP model, fingerprint-based methods generally performed worse than most of the end-to-end DL models on this dataset.

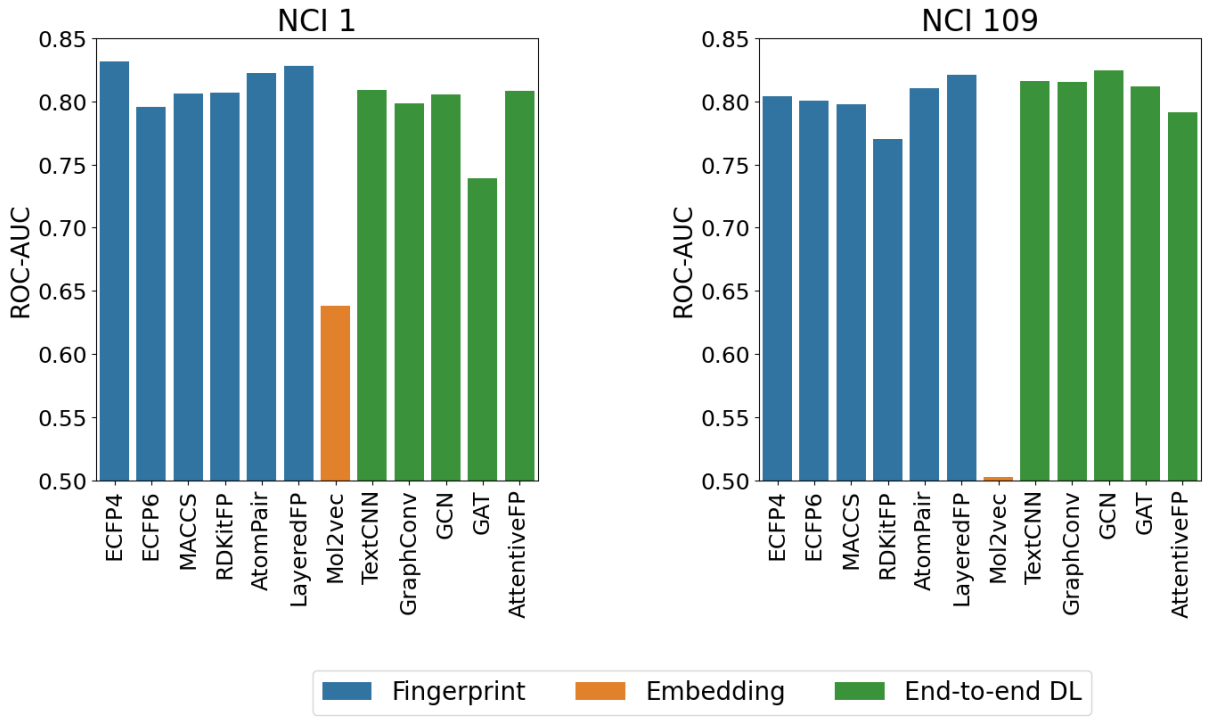


Figure 13: Performance (ROC-AUC) of different deep learning models on the NCI 1 and NCI 109 classification tasks. Higher scores mean better performance.

On the PC-3 dataset, the TextCNN model achieved the the lowest RMSE (0.61), followed by the LayeredFP model and the GraphConv model, both with a RMSE of 0.65 (Figure 14). Other GNNs did not perform as well as GraphConv, having been surpassed by most of the fingerprint-based models. In the CCRF-CEM regression task, several fingerprint-based models (AtomPair, RDKitFP and MACCS) outperformed the best end-to-end DL model, which was once again the TextCNN model (RMSE = 0.76) (Figure 14). On the larger A549 dataset (Figure 14), TextCNN was the best model (RMSE = 0.79), followed by AtomPair and GCN which both reached a RMSE value of 0.81. The increase in dataset size did not benefit all of the end-to-end DL models, however.

In general, performance scores were usually similar between many of the models. This finding is in agreement with the MoleculeNet benchmarking study, which was also unable to find clear differences between models when benchmarking on smaller datasets (less than 3000 compounds) [146]. The best type of compound representation method seems to depend on the dataset itself, even when the prediction tasks are similar. Other factors, such as the particular data split that was used or the limited number of hyperparameter combinations that were explored using random search, could also have influenced the results. Surprisingly, results were consistently worse when using Mol2vec embeddings, which were generated using a model that had been pre-trained on a dataset with over 19 million molecules [181]. The Mol2vec models performed poorly on the training sets as well, indicating that these models were underfitting. Although the original Mol2vec study does not mention the need for scaling the embeddings before using them as inputs to ML models, some of the values in the embeddings are outside the ideal

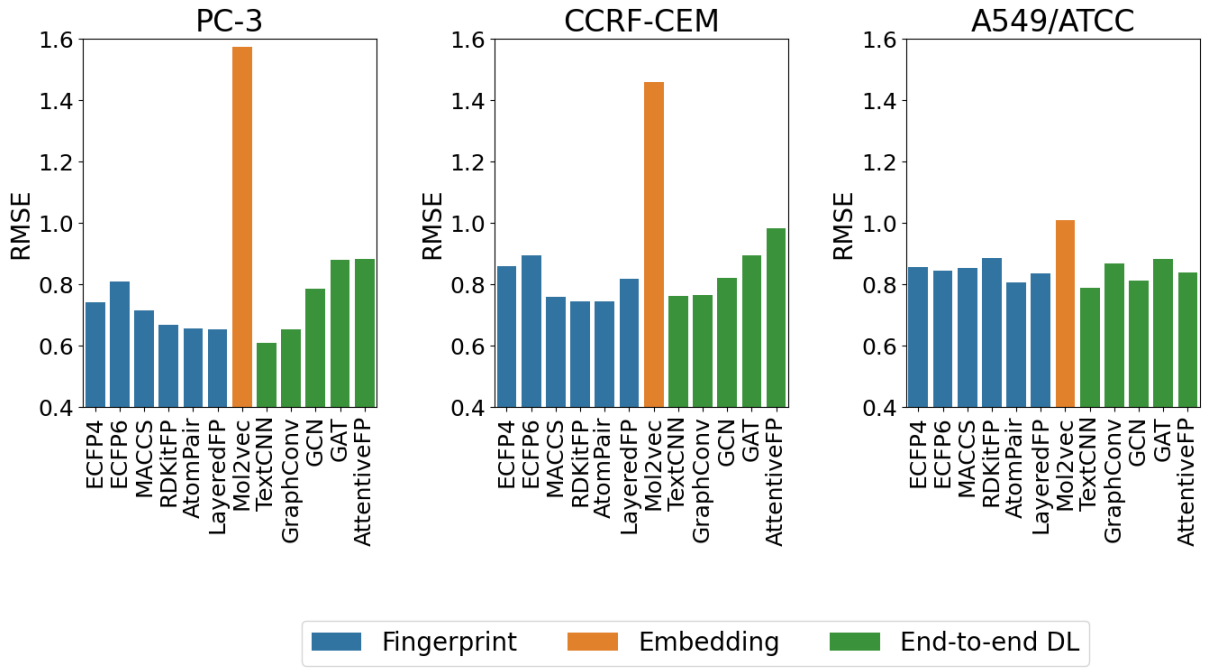


Figure 14: Performance (RMSE) of different deep learning models on the PC-3, CCRF-CEM and NCI-60 A549 regression tasks. Lower scores mean better performance.

range (between 0 and 1) for DL models. Scaling the embeddings prior to learning might be necessary to improve learning. In addition, dataset-specific fine-tuning of the embedding model might also help improve the performance of FCNNs trained on these embeddings.

End-to-end DL models performed as well as, and at times even surpassed, models trained on pre-computed features. TextCNN models, in particular, ranked highly across all datasets. GNNs also performed relatively well on some of the prediction tasks. This is contrary to what would be expected given the limited number of compounds and the fact that these models were not pre-trained on larger chemical datasets beforehand. However, we found that more complex GNNs with attention mechanisms (GAT and AttentiveFP) did not always perform as well as other end-to-end methods on the datasets that were used in this study, indicating that applying self-attention to the molecular graph nodes does not seem to be particularly advantageous in this case.

Regarding molecular fingerprints, LayeredFP consistently performed well across all datasets, and models trained using atom pair fingerprints also performed relatively well, both outperforming the more popular ECFP fingerprints in 4 out of 5 datasets. LayeredFPs are able to encode information about larger subsets of the molecular graphs than ECFPs, capturing more information on the global structure of the molecules. AtomPair fingerprints also encode more global features since all pairs of atoms are taken into account. The results suggest that global molecular features might be important for the prediction of drug sensitivity. Therefore, these less well-known fingerprints might be interesting alternatives to some of the more commonly used options, at least for drug response prediction tasks.

Despite the use of regularization methods, most models still had a tendency to overfit. In the future,

an early stopping mechanism could be implemented similar to the Keras EarlyStopping callback for all models in DeepMol to try to mitigate this.

4.3.2 Ensemble models vs. individual models

For each prediction task, the best individual models were also compared to two different ensembles: an 11-model ensemble comprising all models except Mol2Vec models, and an ensemble comprising only the 5 best individual models for a specific task. Results for the classification tasks and the regression tasks are provided in Tables 9 and 10, respectively.

Table 9: Ensemble results for classification tasks.

Task	Model	ROC-AUC	Accuracy	PRC-AUC	Precision	Recall
NCI 1	ECFP4	0.831	0.831	0.870	0.826	0.831
	11-model ensemble	0.862	0.862	0.898	0.881	0.831
	5-model ensemble	0.851	0.852	0.888	0.866	0.825
NCI 109	GCN	0.824	0.824	0.864	0.809	0.842
	11-model ensemble	0.856	0.856	0.894	0.878	0.822
	5-model ensemble	0.845	0.845	0.882	0.847	0.838

Using the 11-model ensembles improved performance scores across all prediction tasks relative to the best individual models for nearly all of the scoring metrics considered. The 5-model ensembles also outperformed single models on all tasks, and performed better than the larger ensembles on the regression tasks, having achieved lower RMSE scores. The improvement in performance, when using ensembles instead of single models, is likely due to the fact that different representations capture different molecular features. Therefore, combining several drug representations is a strategy that should be considered when developing drug response prediction models in the future.

Table 10: Ensemble results for regression tasks.

Task	Model	RMSE	Pearson	R^2	Spearman
PC-3	TextCNN	0.607	0.806	0.641	0.773
	11-model ensemble	0.581	0.831	0.671	0.797
	5-model ensemble	0.555	0.837	0.700	0.811
CCRF-CEM	AtomPair	0.742	0.798	0.629	0.761
	11-model ensemble	0.706	0.840	0.664	0.817
	5-model ensemble	0.645	0.849	0.720	0.825
A549/ATCC	TextCNN	0.787	0.667	0.421	0.593
	11-model ensemble	0.741	0.715	0.487	0.641
	5-model ensemble	0.736	0.711	0.494	0.637

4.3.3 Feature importance

We were able to determine which features were the most important for the test set predictions by analyzing the values computed by SHAP. Features with greater absolute SHAP values will be the most important, and global feature importance can be determined by averaging the absolute SHAP values across all samples in the dataset. The top 20 most important features (calculated for the test sets) for the best individual models for the NCI 1 classification task (ECFP4 model) and the CCRF-CEM regression task (AtomPair model) are shown in figures 15 and 16, respectively. Through the use of color, these plots also allow us to visualize how the value of a given feature impacts the model prediction.

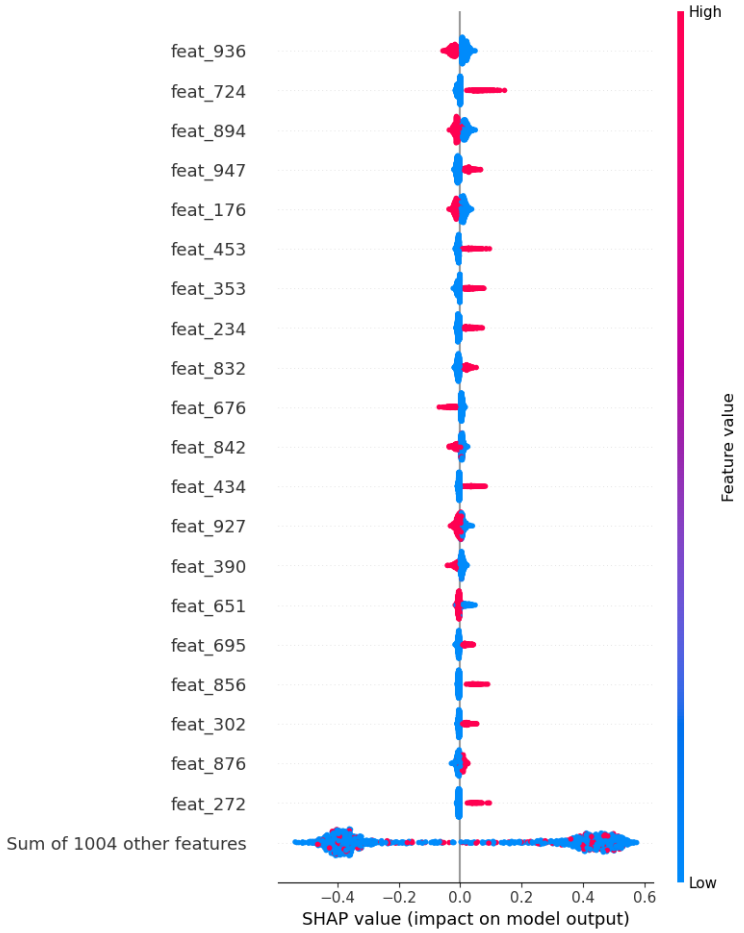


Figure 15: Global feature importance for the ECFP4 model trained on the NCI 1 dataset. Only the 20 most important features are shown, and features are ordered according to their importance. Each point represents a sample in the test set, and the color of the points represents the value of a particular feature for that sample.

For NCI 1, we can use the SHAP values to determine which fingerprint bits generally lead the model to predict whether the cell line will be sensitive to a given compound or not when the bits are set to 1. For example, ECFP4 bits 724, 947, 453, 353, 324, 832, 434, 695, 856, 302, and 272 have a positive effect on the model output, indicating that the presence of these fragments in a compound is associated

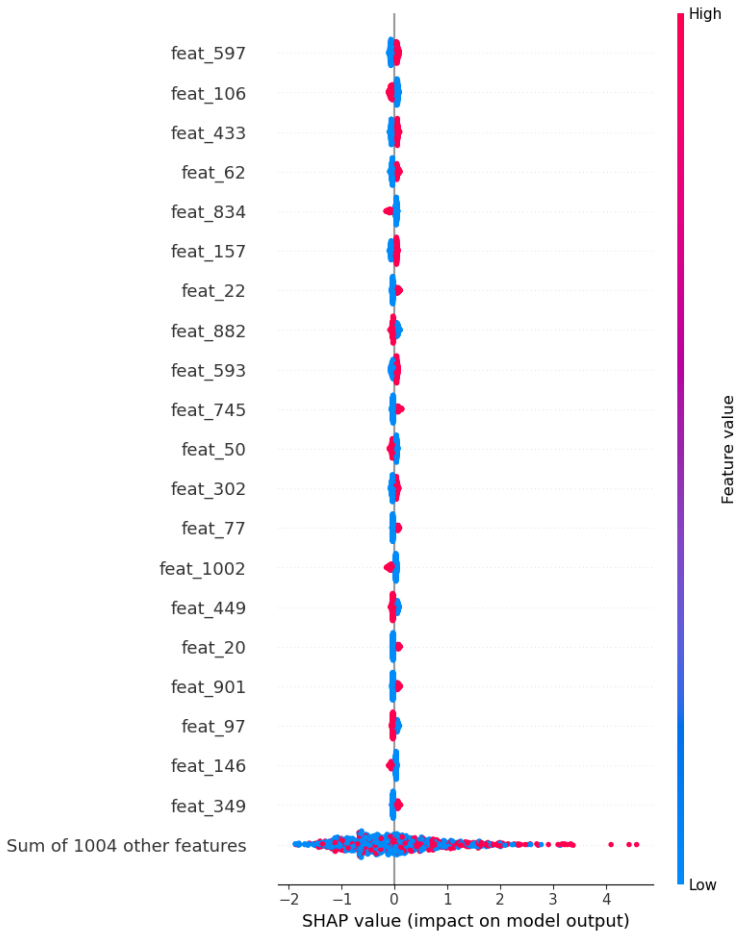


Figure 16: Global feature importance for the AtomPair model trained on the CCRF-CEM dataset. Only the 20 most important features are shown, and features are ordered according to their importance. Each point represents a sample in the test set, and the color of the points represents the value of a particular feature for that sample.

with drug sensitivity. Likewise, in the CCRF-CEM task, fingerprint bits that have higher SHAP values are compound fragments that, when present, contribute to increase the pIC_{50} values. This indicates that the sensitivity of the CCRF-CEM cell to treatment is greater when these substructures are present in the screened compounds.

SHAP values can also be analyzed at the sample level, allowing to explain how specific features affect a single prediction. Figure 17 shows how each feature contributes to increase or decrease the model prediction from a base value of 0.46 for sample 537 from the NCI 1 test set, which had been correctly predicted as “sensitive” (class label=1) by the ECFP4 model. Fingerprint bits that are set to, 1 and move the predicted value towards 1, are features present in the compound that explain the sensitivity of the cell line used in the NCI 1 assay towards this compound.

Figure 18 depicts the top ten ECFP4 bits and the corresponding substructures that most influenced the model prediction for sample 537. The compound in question is (7-Acetamido-1,2,10-trimethoxy-9-oxo-6,7-dihydro-5H-benzo[a]heptalen-3-yl) ethyl carbonate, a colchicine analog. Colchicine and its analogs

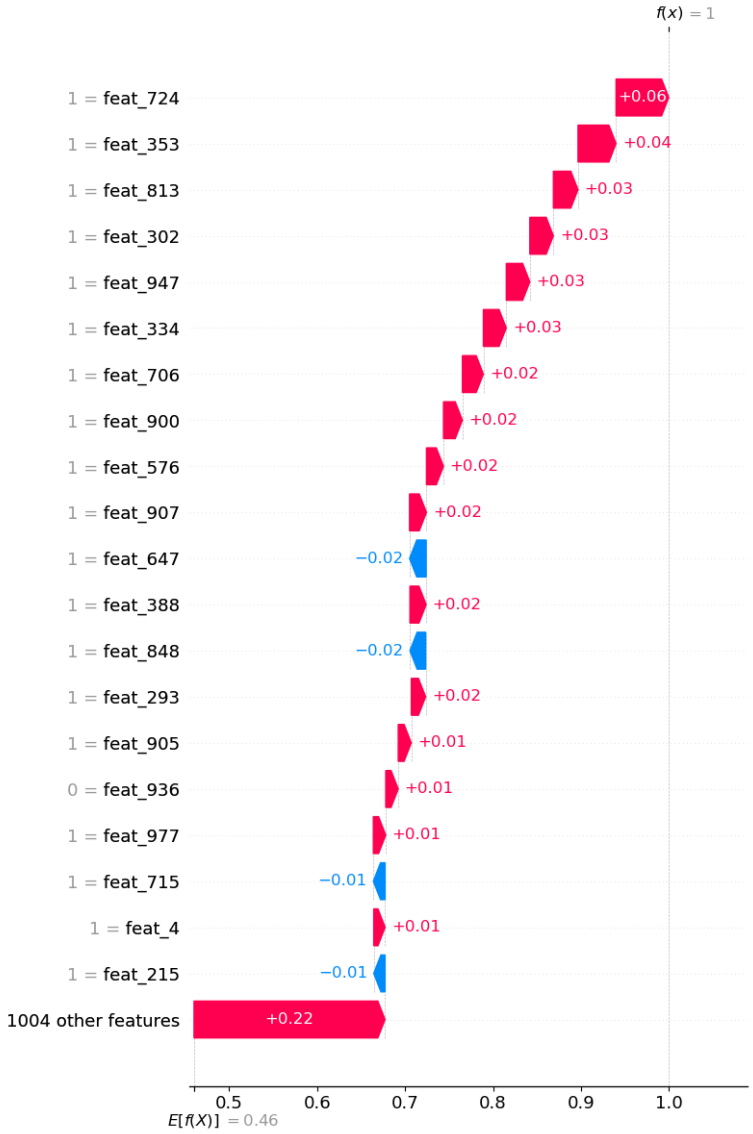


Figure 17: SHAP explanation of the prediction made by the ECFP4 model for sample 537 from the NCI 1 test set, represented as a waterfall plot. Features are ordered by their impact on the model output. Only the top 20 features that contribute the most to the prediction are shown.

bind to tubulin [187]. The top 10 most important ECFP4 bits identified for sample 537 correspond to substructures that have been identified as essential for the bioactivity of colchicine binding site inhibitors [187], [188]. Bit 899 corresponds to the trimethoxyphenyl ring that is essential for interaction with the binding site [187]. Bits 575 and 705 also represent parts of the trimethoxyphenyl moiety, and are centered on atoms that are part of the methoxy groups where drug-protein interactions are more likely to occur [187]. Bits 812, 723 and 333 are centered on atoms belonging to the phenyl ring that is part of the trimethoxyphenyl group as well, which has been shown to be essential for maintaining the geometry of the molecule required at the binding site [187]. Bit 906 is centered on a keto group where an important drug-tubulin interaction can occur in the form of a hydrogen bond [187], [188], and bit 946 is centered

on a carbon atom that is linked to a methoxy group, which also might potentially be involved in drug-protein interactions [187]. Bit 301 is more difficult to interpret, but it appears to be representing part of the central 7-membered ring. These results indicate that the DL model was capable of learning which molecular features are the most relevant for predicting drug response in this particular case.

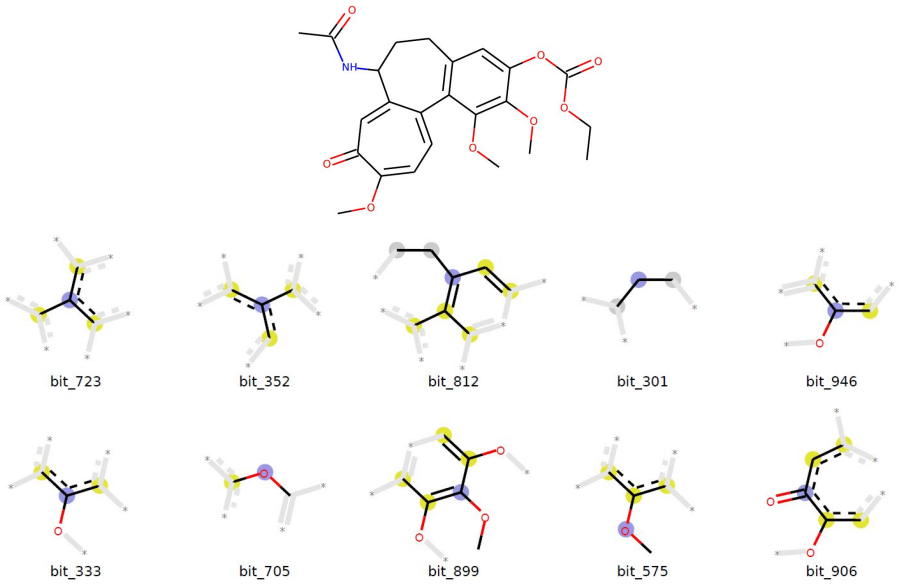


Figure 18: Molecular fragments corresponding to the ten ECFP4 bits that contribute the most towards the prediction for sample 537 from the NCI 1 test set.

Using SHAP values, we were also able to determine which AtomPair fingerprint bits are most predictive of drug sensitivity for sample 90 in the CCRF-CEM test set (Figure 19), which has a predicted pIC50 value that is very close to the true pIC50 value. Unfortunately, due to the use of a hashed version of AtomPair fingerprints with a fixed length, we were unable to obtain more information on the specific fragments that are associated with drug sensitivity for the CCRF-CEM task, making these fingerprints more difficult to interpret than ECFP4 fingerprints.

SHAP values were only calculated for the NCI 1 and CCRF-CEM tasks because we were unable to apply SHAP directly to TextCNN (best type of model for the PC-3 and A549/ATCC datasets) and GCN (best model on the NCI 109 task). More explanation methods, including graph-specific methods such as GNNExplainer [189], should be added to DeepMol in the future.

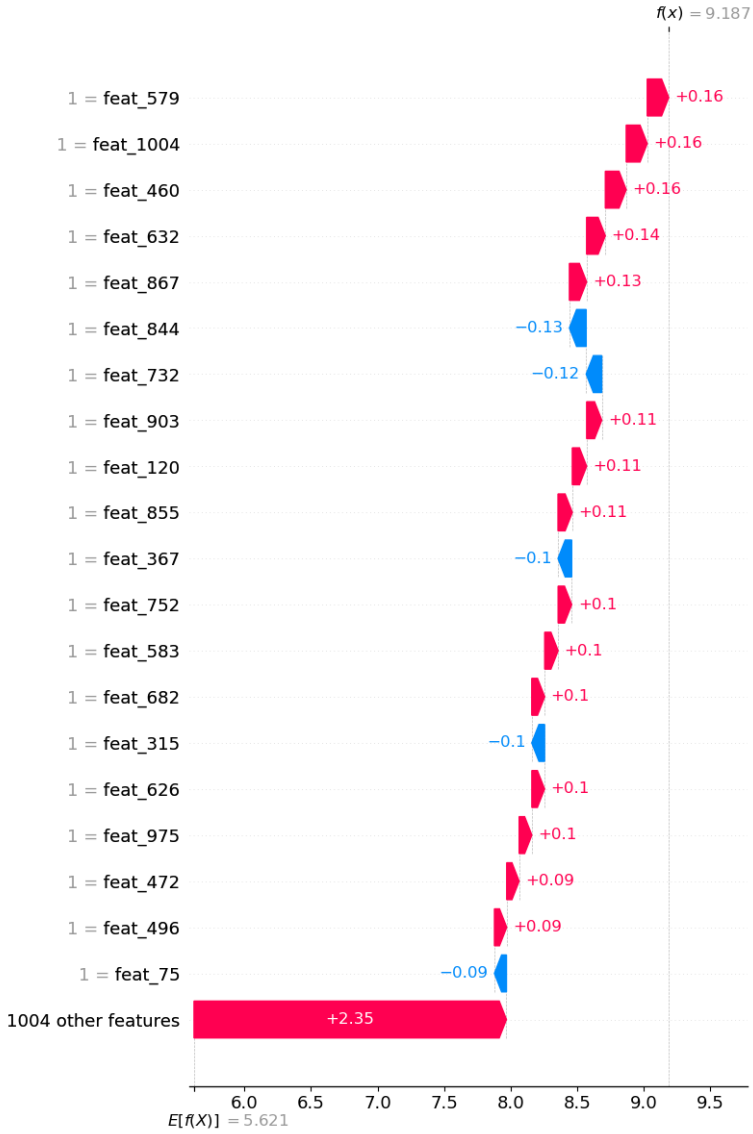


Figure 19: SHAP explanation of the prediction made by the AtomPair model for sample 90 from the CCRF-CEM test set, represented as a waterfall plot. Features are ordered by their impact on the model output. Only the top 20 features that contribute the most to the prediction are shown.

4.4 Conclusion

In this study, we used DeepMol, a new Python package developed in-house for chemoinformatics, to evaluate the performance of different compound representation methods and DL algorithms on several cancer drug sensitivity prediction tasks. We found that end-to-end DL models are capable of outperforming traditional fingerprint-based models even on small datasets. Additionally, less well-known molecular fingerprints may be interesting alternatives to some of the more popular types of molecular fingerprints. We also assessed the effect of combining multiple compound representation methods into ensembles of models and found that this strategy led to improved predictive performance. Finally, we also showed that SHAP can be used to increase the interpretability of some of the models. These findings can help guide

the development of new DL-based drug response prediction models trained on screening data from other screening projects.

The DeepMol framework is under constant development and it is currently at a pre-release version. New models and features will be added in the future.

5

Predicting the ex vivo drug sensitivity of acute myeloid leukemia specimens

Part of this work was used in a submission to the CTD^2 Beat AML DREAM Challenge.

5.1 Introduction

AML is a rapidly progressing hematological cancer. Prognosis is poor, especially among older patients [190], [191]. Recurrence is common [190] and long term survival rates are low [191].

AML is a complex disease and, therefore, difficult to treat. Few treatment options are currently available for AML, and the standard of care has remained relatively unchanged for several decades [192]. The standard therapy consists of remission induction with cytarabine and an anthracycline, followed by post-remission therapy [192]. This is an intensive treatment regimen that some patients, particularly the elderly, may not be able to tolerate [193]. Targeted drugs, designed against specific proteins that are frequently mutated in AML, have only begun to emerge in recent years [194]–[196]. These newly approved therapies have shown promising results, including an improvement in patient survival [197]. However, disease recurrence remains a challenge [194], [197]. Accelerating the discovery of improved targeted treatments tailored to specific patients, and determining which patients will benefit most from these therapies is, therefore, essential.

Next generation sequencing has enabled the extensive molecular characterization of AML tumors, providing deeper insight into the complexity of this disease. Large-scale sequencing studies have identified important mutational events and co-mutation patterns [198]–[200], revealing several distinct molecular subtypes [199]. Many of these genetic alterations are prognostic [199], and can be particularly useful for the therapeutic stratification of patients.

A recent study published by the Beat AML initiative examined the relationship between changes at the genetic and transcriptional levels and drug sensitivity in AML tumors [200]. Tumor samples obtained from AML patients were characterized using RNA and whole-exome sequencing, and *ex vivo* drug screens were performed. The project identified several mutational patterns and gene expression signatures that are correlated with drug sensitivity and resistance [200].

In late 2019, the CTD^2 Beat AML DREAM challenge was launched with the goal of discovering new drug sensitivity biomarkers for AML patients and, ultimately, identifying patient subpopulations that may benefit from certain treatments [201]. The challenge was divided into two subchallenges. Subchallenge 1 was focused on the development of computational methods to predict the sensitivity of patient-derived tumor specimens to anti-cancer drugs using genomics, transcriptomics, and clinical data from the Beat AML project. The goal of the second subchallenge was to predict patient survival. The solutions were evaluated on an unpublished cohort of patients from the Beat AML project.

In this study, we describe the process behind the development of a method to predict drug sensitivity, using the data provided by the Cancer Target Discovery and Development (CTD^2) Beat AML challenge. The proposed solution is a multimodal DL-based model that requires minimal preprocessing and feature selection prior to training. DL was chosen because, unlike traditional ML algorithms, it can be used not only to predict an outcome such as drug sensitivity, but also to directly learn internal, lower-dimensional representations of the input data [18]. Furthermore, the modular nature of DL models also makes it easier to integrate multiple data types [80]. Several recently published DL-based models for drug response prediction have employed this multimodal approach [92], [132], [202].

Lack of interpretability is one of the main drawbacks of DL methods. In this study, through the use of a state-of-the-art feature attribution method, we demonstrate that our DL model can learn to associate important AML biomarkers with drug sensitivity.

5.2 Methods

5.2.1 Dataset

The datasets used for the analyses described in this chapter were contributed by Oregon Health & Science University (research supported by the National Cancer Institute (NCI) grant number U01 CA217862). They were obtained as part of the CTD^2 Beat AML DREAM Challenge through Synapse [syn209405].

Drug sensitivity data were provided in the form of area under the dose-response curve (AUC) values for specimen-inhibitor pairs. The challenge also provided access to several additional files containing omics data (high-confidence somatic mutations and RNA-Seq gene expression data) and clinical data (categorical and numerical features) characterizing the patient-derived specimens. The only information on the screened inhibitors that was provided were their names.

In the Beat AML challenge, the drug sensitivity data were split into three different sets: a training set; a leaderboard set, to which challenge participants only had access at a later stage; and a final validation

set, which was not made public. The training set comprised a total of 20,242 specimen-inhibitor pairs, corresponding to 213 specimens and 122 drugs. The leaderboard dataset had a total of 7,850 specimen-inhibitor pairs, corresponding to 80 new specimens and the same 122 drugs. In this work, the challenge training and leaderboard datasets were merged into a single dataset (merged dataset), giving a total of 28,092 specimen-inhibitor pairs.

5.2.2 Preprocessing

The inhibitor names were used to retrieve the corresponding canonical SMILES strings from PubChem using the pubchempy (version 1.0.4) Python package. The SMILES strings were then used to compute 1024-bit ECFP4 or ECFP6 molecular fingerprints (using the RDKit python package, version 2019.09.3.0), Mol2vec embeddings [181] (mol2vec package, version 0.1), or they were one-hot encoded or label-encoded using scikit-learn (version 0.22.1), depending on the type of architecture that was chosen for the drug subnetwork.

The omics and clinical datasets were further preprocessed using the pandas (version 1.0.1) and scikit-learn (version 0.22.1) python packages. The gene expression dataset (*expr*), in which the expression of a gene in a given specimen was originally reported as $\log_2$(CPM), was rescaled per gene so that each feature has zero-mean and unit variance. Due to the high dimensionality of the gene expression dataset (63,677 features), feature selection methods were also employed. Initially, we reduced the number of features by only selecting the N most variable genes before rescaling the dataset ($expr_{N\ most\ variable}$ models). Since this method did not produce satisfactory results, we also tried selecting features using univariate statistical tests (after rescaling the dataset), using the SelectKBest class from scikit-learn ($expr_{SelectKBest}$ models). In both of these strategies, the number of features to be kept was considered a tunable hyperparameter. As an alternative to these approaches, we also used a file obtained from Ensembl to select genes by gene type, keeping 20,408 protein-coding genes in the final gene expression dataset ($expr_{protein\ coding}$ models).

When using a 1D CNN for the gene expression-encoding subnetwork ($expr_{1D\ CNN,\ protein\ coding}$ + *mut* + *drug* models), hierarchical clustering was used to find a new order for the features so that the position of the genes in the input tensor would be biologically meaningful. A dendrogram was created based on the clustering results, and the positions of the leaves were used to rearrange the features. Using this method, genes positioned close to each other in the input tensor will have similar expression patterns across the samples.

The mutation data provided by the challenge were binarized and summarized at the gene level, with '1' indicating that a given sample had at least one alteration in a certain gene and zero indicating the absence of mutations. Since the merged (training + leaderboard) dataset and the challenge validation dataset might not have the same genes, only the genes included in the merged dataset were kept as features, and any missing values were set to zero. The final mutation dataset had a total of 302 features.

When using clinical data, non-binary categorical features were one-hot encoded, and any missing values were ignored. For binary features, all missing values were set to zero. Missing values in the

numerical clinical features were either replaced with the median value or with zero. We tested models trained on all of the clinical features (*expr*$_{protein\ coding}$ + *mut* + *clinical*$_{all}$ + *drug* models), as well as models that only used a small subset of the clinical features (*expr* + *clinical*$_{selected}$ + *drug* models) that provided information on the mutation status of important AML genes and the molecular subtypes of the specimens belong (Table 11) as an alternative to using the mutation dataset.

Table 11: Selected clinical features.

Feature	Description
FLT3-ITD	Presence (1) or absence (0) of internal tandem duplication (ITD) in the FLT3 gene.
NPM1	Presence (1) or absence (0) of mutations in the NPM1 gene.
finalFusion	The type of fusion genes detected in the specimens, if any exist.
specificDxAtAcquisition	Specific diagnosis at time of earliest specimen inclusion by collection date. Includes information on the molecular subtype of the tumors.
FAB/Blast.Morphology	FAB classification of the specimen if noted in the electronic medical record.

After input-specific preprocessing, each of the input datasets was separately merged with the data in the drug sensitivity file to obtain the corresponding features for each specimen-inhibitor pair. The dataset was then split into three subsets – a test set comprising 15% of the merged dataset, which was used for internal validation of the models, a validation set corresponding to another 15% of the data, and the remaining 70% of the data were used as the training set.

5.2.3 Deep learning models

A multimodal approach was used to build the DL models. Each data type was input to its corresponding feature-encoding subnetwork, and then the learned representations from each subnetwork were concatenated into a single tensor before being fed into a final fully connected AUC prediction subnetwork. An example of such a model is shown in Figure 20. All of the models were implemented using the Keras subpackage in Tensorflow [98] (version 2.0.0).

Throughout this work, we tested 4 different combinations of the input data types that were provided by the challenge (expression data (*expr*); mutations (*mut*); *clinical* features; and *drug* data):

- *expr* + *drug*
- *expr* + *mut* + *drug*
- *expr* + *clinical*$_{selected}$ + *drug*
- *expr* + *mut* + *clinical*$_{all}$ + *drug*

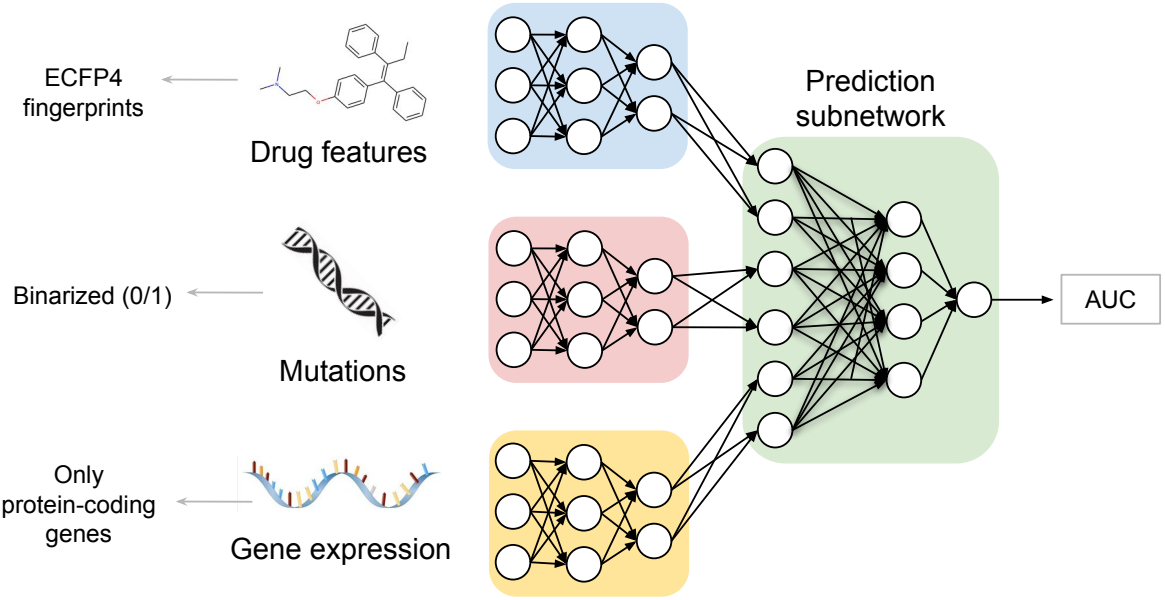


Figure 20: A schematic representation of a multimodal model for drug sensitivity prediction similar to the models developed in this work.

The mutation and clinical subnetworks were always composed of fully connected layers. The gene expression subnetwork was either a fully connected neural network or a 1D CNN ($expr_{1D\ CNN,\ protein\ coding}$ subnetworks). When the drugs were featurized using molecular fingerprints (ECFP4/ECFP6) or pre-computed Mol2vec embeddings, the drug subnetwork was also a fully connected neural network ($drug_{ECFP4}$, $drug_{ECFP6}$, and $drug_{Mol2vec}$ subnetworks)). We also evaluated the use of 1D CNN subnetworks, such as the $drug_{1D\ CNN}$ subnetwork, which is based on the drug encoding subnetwork in the DeepDTA model [203] and the $drug_{TextCNN}$ subnetwork [182], as implemented in the DeepChem [100] package (version 2.4.0). In addition, a subnetwork ($drug_{SmilesLSTM}$) that learns directly from one-hot or label encoded SMILES strings using long short-term memory (LSTM) layers, taken from a recent large-scale benchmarking study [148], was also tested.

Since the prediction of the continuous AUC values is a regression problem, the output unit always had a linear activation function and the MSE was used as the loss function. The Adam algorithm was used as the optimization method, with a learning rate of 0.001. All of the models were trained for a maximum of 500 epochs, using the EarlyStopping callback with a patience of 15 epochs to stop the learning process once the loss stopped improving. A mini-batch size of 32 was used for all models. The validation dataset was used to tune additional hyperparameters using a randomized search approach, with a maximum of 60 combinations of hyperparameter values being tested before selecting the best model.

After optimizing the hyperparameters, the models were evaluated on the held out test set. Several scoring metrics were used to evaluate model performance, including the MSE, Spearman correlation, Pearson correlation, and coefficient of determination (R^2) scores. The main scoring metric, however, was the metric defined by the challenge, which is calculated as the average of the per-inhibitor Spearman correlations. This metric was used both to optimize the hyperparameter values and to evaluate the final

models.

The model architecture that was selected for submission to the challenge was retrained on the full dataset (merged dataset), and then it was saved so that it could be used to make new predictions in the future. All models submitted to the challenge were further evaluated on a private validation dataset that participants did not have access to at any time during the challenge. Submissions were evaluated using the average per-inhibitor Spearman correlation as the primary scoring metric and the average per-inhibitor Pearson correlation as a secondary metric. The challenge organizers used bootstrap sampling of the test set (1000 bootstrap samples) to calculate the final performance scores and to determine if the scores were significantly different between submissions.

5.2.4 Machine learning models

We also evaluated several ML models to determine if the use of DL provided any benefit. Our DL models were compared against elastic net [26], support vector regression (SVR) [204], RF [32], extreme gradient boost (XGBoost) [205] and light gradient boosting machine (LGBM) [206] models. The XGBoost model was implemented using the xgboost Python package (version 1.1.1), LGBM was implemented using the lightgbm package (version 2.3.1), and the remaining models were developed using scikit-learn [186] (version 0.22.1).

All of the ML models were trained on gene expression, mutation and drug features. The gene expression features were limited to protein coding genes only. After concatenating the different input types into a single dataset, features were selected using univariate statistical tests (using the SelectKBest class from scikit-learn), and only the k best features were kept.

Hyperparameters were optimized using the same random search approach as described for the DL models. Besides model-specific hyperparameters, the number of features (k) that were kept was also considered a tunable hyperparameter.

5.2.5 Heterogeneous ensembles

A heterogeneous ensemble was also built to determine if combining different models could further improve performance. The ensemble included the top 5 DL models developed in this work as well as the top 3 ML models. The predictions of the individual models were averaged to obtain the final prediction. The hyperparameter values used for each individual model were the optimal values that had been previously found using randomized search.

5.2.6 Feature importance

The SHAP Python package (version 0.39.0) [163] was used to analyze feature importance. More specifically, we used the Deep SHAP method, which uses an adaptation of the Deep Learning Important FeaTures

(DeepLIFT) [161] feature attribution algorithm to approximate SHAP values. The SHAP value for a given feature reflects the change in the expected model prediction when conditioning on that feature [163].

5.2.7 Code availability

All of the code developed in this study is available online at https://gitlab.bio.di.uminho.pt/dlrsb/beat_aml. The full results tables and additional details on the models (including the full hyperparameter search spaces and the best hyperparameters found for each type of model) are also provided in this repository.

5.3 Results

5.3.1 Challenge solution

Our submission to the challenge was a multi-input ($expr_{protein\ coding}$ + mut + $drug_{ECFP4}$) DL model comprising three input-specific feature-encoding subnetworks (expression, mutation and drug subnetworks) and a final drug sensitivity prediction subnetwork. The model was trained on gene expression data limited to protein coding genes, binarized mutations, and drug data in the form of ECFP4 molecular fingerprints. The drug-encoding subnetwork has three hidden layers containing 1000, 500, and 250 hidden units. The mutation-encoding subnetwork has two hidden layers with 150 and 75 hidden units. The gene expression subnetwork also has two hidden layers, with 625 and 312 hidden units. The prediction subnetwork consists of a single hidden layer with 1000 hidden units, and an output layer with a single unit. All of the layers were fully connected, and all hidden units used the ReLU activation function. Batch normalization was performed after each hidden layer, and a dropout rate of 0.3 was used during training.

This model was selected because it had achieved the highest average per-inhibitor Spearman value (0.656) out of all of the models that we had evaluated using our internal validation workflow (for the full results, see Supplementary Table 13). However, after the challenge we found that the data splitting process had not been implemented properly, meaning that the models that had been tested up until that point were not comparable as they had been evaluated on different test sets. For this reason, after the challenge was over we repeated our analysis, guaranteeing that the training, validation and test sets were always the same for all models. The results of this reanalysis are described in more detail in the next subsection.

The challenge results are shown in Figure 21. Our submission (UM-BISBII-AML) achieved an average per-inhibitor Spearman correlation of 0.312 and an average per-inhibitor Pearson correlation of 0.331. It was the 13th best model out of a total of 28 submissions in terms of the bootstrapped average per-inhibitor Spearman correlation, and 11th best in terms of the average per-inhibitor Pearson correlation. For comparison, the winning solution achieved average per-inhibitor Spearman and Pearson correlation values of 0.369 and 0.378, respectively.

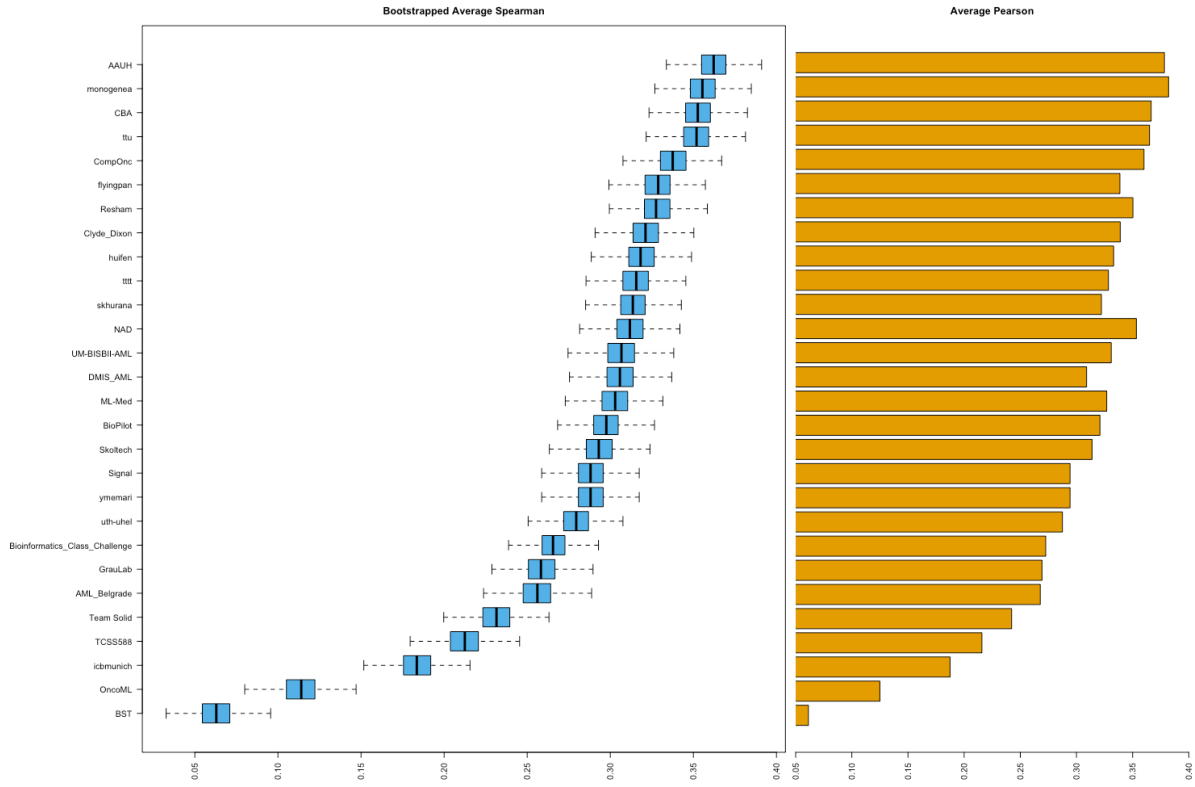


Figure 21: Final results for subchallenge 1 of the CTD^2 Beat AML DREAM Challenge [207]. Our submission is named 'UM-BISBII-AML'.

These scores are lower than our internal validation results. The final validation dataset used by the challenge had data for 65 new specimens, all of which had not been previously seen by our model. In retrospect, creating training, validation and test sets with non-overlapping specimens instead of randomly splitting the data would have been a more rigorous validation scheme for our internal validation workflow.

5.3.2 Internal validation results

As mentioned in the previous section, after discovering that the data splitting process was not being performed correctly, we repeated our analysis so that the models would be comparable.

First, the effect of using different input combinations was evaluated. Three types of input combinations were tested: *expr + drug*, *expr + mut + drug* and *expr + clinical*$_{selected}$ *+ drug*. In our preliminary study, the first solutions that were developed all used the "N most variable genes" feature selection approach for the expression dataset, so this was maintained when repeating the analysis. The results of these tests are shown in Figure 22.

Models trained on expression, mutation and drug features clearly outperformed models trained without using mutation data as input. Substituting mutations with clinical features that refer to known molecular subtypes or genetic alterations in AML slightly improved model performance when compared with models that only used expression and drug features. Nevertheless, these models still performed considerably

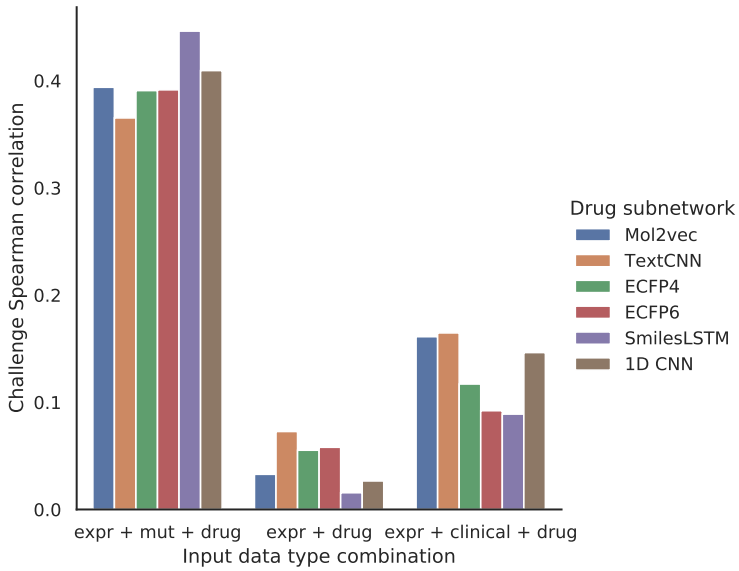


Figure 22: Average per-inhibitor Spearman correlation scores for models trained on different input type combinations. All models used the "N most variable genes" feature selection approach to reduce the number of features in the expression dataset prior to training.

worse than those that had used mutation data instead. The selected clinical features contained information on the mutational status of a few AML-related genes, but this was not sufficient to adequately predict drug sensitivity. Combinatorial mutational events are important in AML and different combinations produce different response to drugs [200]. The full mutation dataset may contain additional information regarding co-mutation patterns that the selected clinical features do not provide. Given these results, the models that were developed afterwards always included mutation features as inputs.

Since the performance scores of the models were quite low when using the "N most variable genes" feature selection method, we decided to test two alternatives - feature selection based on univariate statistical tests (SelectKBest) and only using protein coding genes. The use of these feature selection methods greatly improved model performance (Figure 23). When using the new splits to train and validate the models, the model configuration that we had chosen to submit to the challenge ($expr_{protein\ coding}$ + *mut* + $drug_{ECFP4}$) still achieved a similar average per-inhibitor Spearman correlation value (0.660). However, with the new splits we found that two SelectKBest models ($expr_{SelectKBest}$ + *mut* + $drug_{ECFP6}$ and $expr_{SelectKBest}$ + *mut* + $drug_{ECFP4}$) also had average per-inhibitor Spearman correlation scores similar to the $expr_{protein\ coding}$ + *mut* + $drug_{ECFP4}$ model (Figure 23 and Supplementary Table 14).

Additionally, in an attempt to improve model performance after the challenge was over, two new types of models were also tested: $expr_{1D\ CNN,\ protein\ coding}$ + *mut* + *drug* and $expr_{protein\ coding}$ + *mut* + $clinical_{all}$ + *drug* models. Six variations of these were developed, with different drug subnetworks (ECFP4, ECFP6, Mol2vec, TextCNN, 1D CNN or SmilesLSTM). Figure 24 compares the results of the $expr_{1D\ CNN,\ protein\ coding}$ + *mut* + $drug_{ECFP4}$ and $expr_{protein\ coding}$ + *mut* + $clinical_{all}$ + $drug_{ECFP4}$ models with the model configuration that had been submitted to the challenge. Results for the other drug subnetworks are reported in

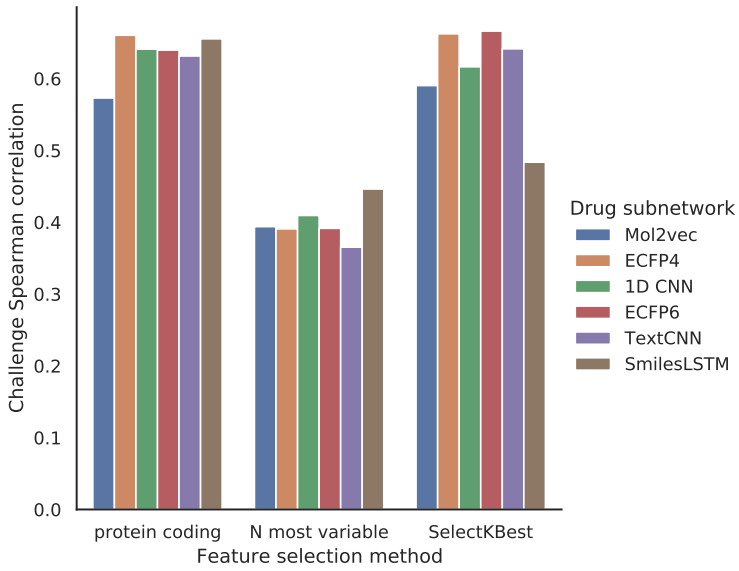


Figure 23: Average per-inhibitor Spearman correlation scores for models trained on different gene expression features selected using different feature selection methods.

Supplementary Table 14.

Using a 1D CNN for the gene expression subnetwork instead of fully connected layers decreased the average per-inhibitor Spearman correlation score. Although clustering was used to rearrange the genes in a more biologically meaningful order, this result is not surprising, as the expression data is not sequential in nature and the position of the genes in the input tensor may not be relevant to the prediction task. Therefore, it does not seem like there is any advantage in using this approach for this type of data.

Including clinical features in addition to the expression, mutation and drug features slightly improved performance when compared to the $expr_{protein\ coding}$ + mut + $drug_{ECFP4}$ approach. The $expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{ECFP4}$ model was the best model overall, with an average per-inhibitor Spearman correlation value of 0.666. The addition of clinical features also improved the performance of most of the other models with different drug subnetworks (Supplementary Table 14). Thus, it might be beneficial to include clinical variables when developing drug response prediction models for patients, if these features are available.

Throughout this work, both traditional molecular fingerprints (ECFP4 and ECFP6) and learned representations of compounds (Mol2vec, TextCNN, 1D CNN and SmilesLSTM subnetworks) were tested. In general, our internal validation results show that the type of architecture used for the drug subnetwork had less impact on the predictive performance of the models than the types of input features that were used, as can be seen in Figures 22 and 23.

Finally, there is a discrepancy between the challenge score (average of the per-inhibitor Spearman correlations) and the Spearman correlation values that were calculated using our internal validation setup (Supplementary Table 14). This is an indication that our models are not accurately predicting drug sensitivity for some of the inhibitors.

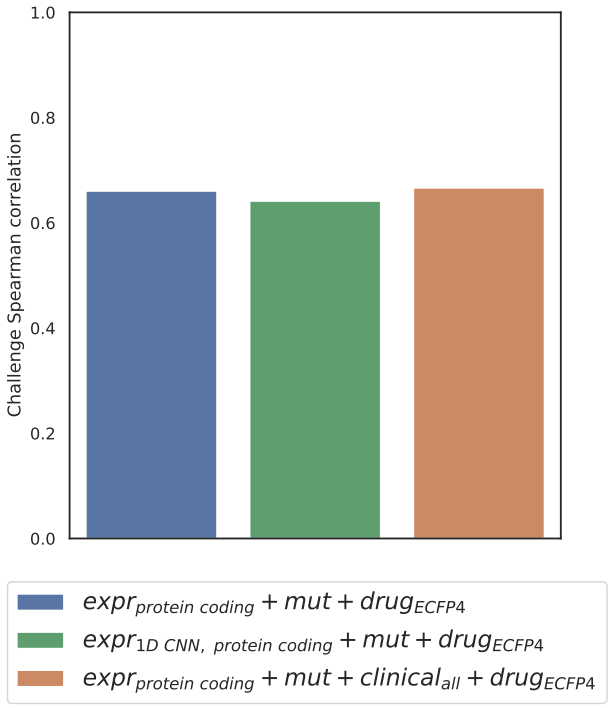


Figure 24: Average per-inhibitor Spearman correlation scores for the model configuration that had been submitted to the challenge ($expr_{protein\ coding} + mut + drug$) and two additional models that were developed after the challenge ($expr_{1D\ CNN,\ protein\ coding} + mut + drug$ and $expr_{protein\ coding} + mut + clinical_{all} + drug$).

5.3.3 Comparison with other ML algorithms

Our best model ($expr_{protein\ coding} + mut + clinical_{all} + drug_{ECFP4}$) outperformed all of the traditional machine learning models by a wide margin (Figure 25). The best ML model was LGBM, which achieved an average per-inhibitor Spearman correlation value of 0.527. Most of the DL models that either used SelectKBest or a list of protein coding genes to filter the expression data performed better than the LGBM model, including the $expr_{protein\ coding} + mut + drug_{ECFP4}$ model (Supplementary Table 14).

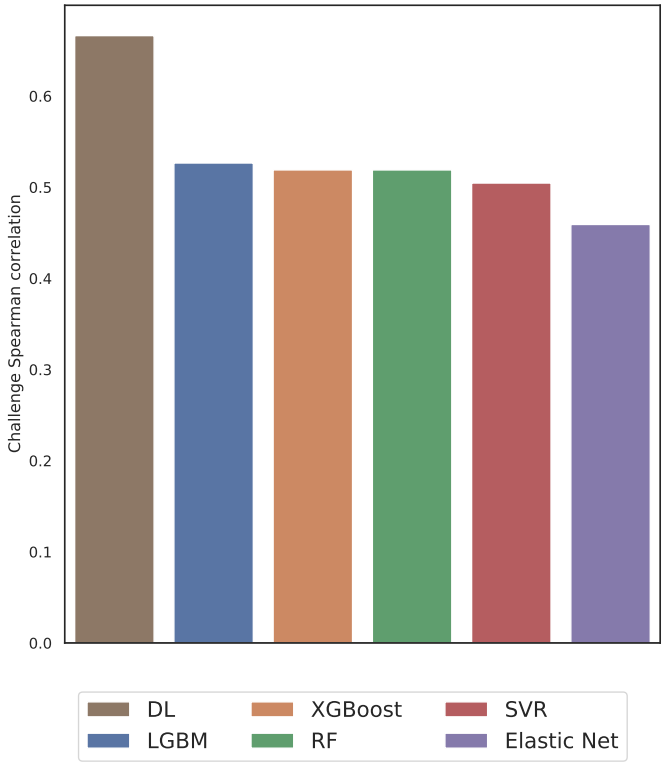


Figure 25: Comparison between the best DL model developed in this study and other ML algorithms.

5.3.4 Hetereogeneous ensemble

The simple heterogeneous ensemble (top 5 DL + top 3 ML models) achieved an average per-inhibitor Spearman correlation value of 0.687, which is an improvement over the best individual DL model (average per-inhibitor Spearman correlation=0.666).

5.3.5 Feature importance

Feature importance was determined based on the mean absolute SHAP values calculated for each feature. The top 20 most important features for the $expr_{protein\ coding} + mut + clinical_{all} + drug_{ECFP4}$ model, are shown in Figure 26. These are the variables that had the greatest impact on the output of the model. Among the top features are several drug features (ECFP4 bits) and genes from the mutation dataset, as well as two clinical features that are key prognostic factors in AML (%.Blasts.in.PB and WBC.count).

The most important mutation features correspond to some of the most frequently mutated genes in AML [198]–[200]. Many are known prognostic biomarkers [199], and mutations in these genes can influence drug response [200]. In the original Beat AML study, mutations in the KRAS and NRAS proto-oncogenes were associated with resistance to most drugs [200]. Alterations in the FLT3 gene were also highly associated with drug response [200]. IDH1 mutations were linked to resistance to most drugs, while mutations in IDH2 were associated with enhanced sensitivity to many of the drugs screened in the Beat AML study [200]. These findings reveal that our DL model correctly learned to associate important

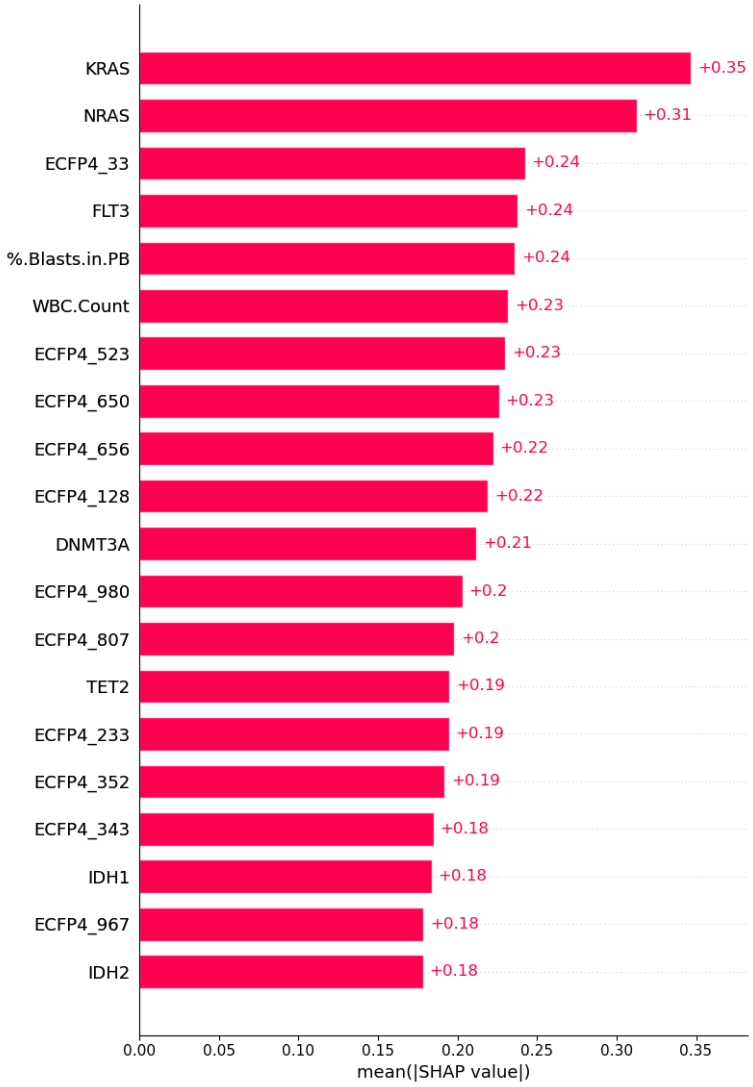


Figure 26: Top 20 most important features ranked by mean absolute SHAP values.

mutational events in AML with drug response.

None of the gene expression features appear in the top 20 and in general the SHAP values for individual expression features were relatively low (Supplementary Figure 40). However, when analyzing the mean absolute SHAP values for features grouped by input type, we found that the expression dataset as a whole was the second most important group of features (Table 12).

Table 12: Mean absolute SHAP values for features grouped by input type. SHAP values for groups of features were calculated by summing the SHAP values of all of the individual features, and then the mean absolute SHAP value across all samples was calculated for the combined features.

Input type	Mean absolute SHAP value for grouped features
expr	18.86
mut	3.62
clinical	0.62
drug	30.53

5.4 Discussion

Although our internal validation results seem promising, the DL models developed in this work are still far from being clinically applicable and there is still much room for improvement. The discrepancy between our internal validation results and the challenge scoring of our submission highlights the importance of choosing an adequate validation scheme to estimate the real generalization capacity of the models when faced with new patients or drugs.

DL models perform best when training data is abundant. The Beat AML challenge dataset is a moderately-sized screening dataset, with over 20 thousand experiments, but the number of unique compounds (122) and specimens (293) in the dataset is relatively small, which makes it more difficult for the models to generalize well to new scenarios. Therefore, pre-training the feature-specific encoding subnetworks on larger datasets might be beneficial. The drug subnetwork could be pretrained using large compound datasets from PubChem [208] or ChEMBL [174], for example. Additional mutational and transcriptomics data could be obtained from other AML patient cohorts, such as the TCGA AML study [198]. Alternatively, the omics subnetworks could be pre-trained on larger, pan-cancer datasets and then fine-tuned using the Beat AML dataset.

Using a list of protein-coding genes or univariate statistical tests to select gene expression features produced models with similar results. Although DL models usually require less feature selection than traditional ML models, some filtering of the transcriptomics dataset is still necessary, as the data are noisy and very high-dimensional. In the future, alternative feature reduction strategies should be explored to filter out less relevant features, including selection strategies based on biological prior knowledge. For example, the expression dataset could be filtered to include only genes that have been implicated in AML and other cancer-related genes. Further limiting the amount of gene expression features could also help reduce overfitting.

According to our feature importance analysis, individual gene expression features did not have much impact on the model, but as a group the gene expression dataset is important for the prediction of drug sensitivity. These results indicate that patterns of expression across multiple genes are predictive of drug response. This is in line with the original Beat AML study, which had uncovered specific gene expression signatures that are correlated with drug sensitivity in AML patients [200].

5.5 Conclusions

The main goal of this study was to develop DL models capable of predicting drug sensitivity in tumor samples obtained from AML patients, using data provided by the CTD^2 Beat AML DREAM challenge. Throughout the development of our solution for the challenge, a variety of preprocessing strategies and model architectures were explored and the impact of these choices on model performance was evaluated. Through this analysis we were able to identify the feature selection methods, input data combinations and drug representations that seemed to produce the best results for this prediction task. Although the performance of these models indicates that they are still insufficient for clinical decision making, we were able to demonstrate that DL is capable of associating important AML features with drug response and clearly outperforms other traditional ML models.

6

Predicting drug synergy in cancer cell lines

At the time of writing, this chapter is being prepared for submission to a scientific journal.

6.1 Introduction

The phenomenon of drug resistance is one of the greatest challenges in the fight against cancer. Although many tumors initially respond well to a given treatment, the efficacy of single-drug anti-cancer therapies is often diminished due to the existence of tumor drug resistance mechanisms. Resistance-conferring characteristics may already be present in the tumor cells prior to therapy, or they may arise as an adaptive response of the tumor to the treatment itself [13]. One of the main drivers of resistance is intratumoral heterogeneity. Genomic instability in cancer leads to the emergence of subpopulations of cells within a tumor with distinct characteristics and different sensitivity to drugs. Treatment may exert selective pressure on the cells and select subpopulations possessing characteristics that favor drug resistance, leading to future relapse [209].

Combining multiple treatments instead of administering a single drug can help to reduce drug resistance [210]. Drug combinations may circumvent pre-existing resistance mechanisms more easily and prevent the development of acquired resistance mechanisms [44]. In addition, certain combinations may be more effective than would be expected when taking into account the effects of each of the constituent compounds on their own, a phenomenon called drug synergy. Drug synergy increases treatment efficacy without requiring an increase in drug dosage, potentially avoiding an increase in toxicity as well [14]. Therefore, optimal drug combinations will be those that produce synergistic effects.

Novel effective anti-cancer drug combinations can be discovered using high-throughput cell viability assays. In these assays, a large number of candidate drug combinations are screened at different concentrations across a panel of cancer cell lines and the cellular response to the drug is measured. In recent years, several datasets from large-scale drug screening initiatives have been made publicly available [6]–

[8]. The largest of these is the NCI ALMANAC project [7], which screened a total of 5,232 pairs of FDA-approved drugs against NCI-60, a panel of 60 tumor cell lines that have been extensively characterized at the molecular level [101]. The project uncovered several synergistic drug pairs, including two clinically novel combinations that are currently being evaluated in phase I clinical trials [7].

Despite the existence of these high-throughput technologies, screening all conceivable drug combinations is still infeasible, for both practical and financial reasons [8], [9]. Computational methods could greatly reduce the search space, thus minimizing the experimental effort required to find truly effective anti-cancer drug combinations. ML methods, for example, can be used to learn functional mappings between very high-dimensional input data and a score that reflects drug combination effects. This makes it a powerful approach to develop models that are able to predict drug synergy based on drug combination screening experiments and other relevant data. Several ML models for drug synergy prediction have been described in the literature [8], [68], [211]–[214]. Many of these studies used tree-based ML methods, such as RFs [68], [211], [213] or gradient boosting [211], [212], [214].

One particular subset of ML that has attracted great interest from researchers in this field is DL. DL models are composed of multiple processing layers [17], giving them the ability to learn complex, non-linear functions. Furthermore, unlike most traditional ML methods, DL approaches typically do not require extensive feature selection before training, since they have the ability to learn higher-order representations directly from raw input data [18]. Since DL models can handle large amounts of high-dimensional and noisy data, they may be good candidates for the development of drug synergy prediction models.

In 2018, the first model to use DL to predict drug combination effects was published [82]. The model, named DeepSynergy, is a feedforward, fully-connected deep neural network that uses chemical features and gene expression data to predict drug synergy. Shortly after, Xia et al. developed a multimodal DL model to predict the growth inhibition of cell lines from the NCI ALMANAC project [137]. This model includes separate feature-encoding subnetworks for each input data type (drug descriptors, gene expression, microRNA and proteomics data) and a cell line growth prediction network. Several other DL-based drug synergy prediction models have since been reported in the literature. Similar to the model proposed in 2018 by Xia et al., many of these more recent models adopt a multimodal architecture [215]–[218].

The majority of the published DL-based drug synergy prediction models are entirely composed of fully connected layers [82], [137], [216], [217], [219], but several innovative architectures have also been proposed. Zhang et al. [220] developed a sparsely-connected deep belief network constrained by biological prior knowledge. The recent architecture of the TranSynergy model [221] includes a transformer [222] component as well as fully connected layers. A method called REpresentation of Features as Images with NEighborhood Dependencies (REFINED) was developed to transform drug descriptors into images so that CNNs could be used to model drug synergy instead of the typical fully connected networks [223]. Another study used GNNs for drug-specific subnetworks to learn drug representations directly from the compound structures in an end-to-end manner instead of using fully connected layers trained on molecular descriptors or fingerprints [218]. Several studies published within the last year have used GNNs trained on graphs containing information on interactions between the drugs in a combination, between drugs and

their targets, and/or interactions between genes or proteins in the cell lines [224]–[227].

Most drug synergy prediction models use drug and gene expression features, with some models using these feature types exclusively [82], [216]. Some models include additional cell line information, such as genetic data (somatic mutations and/or CNVs [215], [220] proteomics data [137], [219], among others. Several studies also explicitly include drug target-specific features [217], [220], [221]. Since adding more features increases the complexity of the models, assessing which types of input data are more informative and predictive of drug synergy is important.

Many of the models use precomputed molecular descriptors or fingerprints as chemical features to represent the drugs instead of using end-to-end DL methods to learn the relevant compound features directly from the compound structures. Given that the screening datasets that are currently available only contain a very limited number of compounds, it is still unclear whether there is any benefit in using learned representations instead of traditional fingerprints and descriptors. A recent study benchmarked several compound representations on a large drug synergy dataset and found that DL-based representations were able to outperform traditional fingerprints [214], but the authors also noted that the difference between the top performing DL-based methods and the best fingerprints were not substantial and that other concerns, such as interpretability, may be more important.

Some models forgo chemical features entirely, and only use drug-target interaction features to represent the compounds [221] or heterogeneous graphs to represent the interactions between the compounds and between drugs and targets or cell lines [224]–[227]. This provides a way to integrate more biological prior knowledge directly into the model. However, this type of approach might not work well for novel compounds with unknown targets or targets that have been poorly characterized. Furthermore, excluding chemical features precludes the identification of structural features that may be enriched in synergistic compounds that could be used as starting points for the design of novel compounds with similar effects.

In most DL-based synergy prediction studies, some type of feature reduction method is usually applied to the cell line omics data, either by using specific gene lists based on biological prior knowledge [218], [220], [221], or by employing unsupervised data dimensionality reduction techniques, such as PCA [219], [228] or autoencoders [215], [219]. Using predefined gene lists to select features provides greater control over the selection process and might make the models easier to interpret biologically. However, certain approaches, such as limiting the gene features to known drug targets, for instance, may limit the generalization capacity of the models. The selected genes will be determined by the compounds present in the training set, and models trained using these features might have difficulty inferring drug synergy for new drugs with different drug targets. Data-driven approaches avoid this problem.

Another advantage of using data-driven dimensionality reduction techniques is that they may be trained using much larger datasets with data from more cell lines than those used in the screening datasets [219] or even patient data from resources such as The Cancer Genome Atlas [229], as proposed in a recent study [215]. This may improve the learning process and increase the generalizability of the models. Nevertheless, a limitation of using feature reduction methods like PCA is that model interpretability becomes harder to attain, requiring more steps to link model predictions with the original inputs [219].

Therefore, evaluating which feature reduction methods are capable of achieving satisfactory performance, as well as simultaneously facilitating model interpretability, is an essential step when building drug synergy prediction models.

Although several DL-based methodologies have been proposed to tackle the problem of drug synergy prediction, with promising results, the impact of different methodological aspects on the performance of the methods is still unclear. In this work, we set out to investigate the impact of different methodological variables on the performance of drug synergy prediction DL methods using the ALMANAC drug combination screening dataset. Namely, we will evaluate the impact of different preprocessing steps, types of input data, and DL architectures on the final performance of the methods. Prior biological knowledge was used to select cell line features and to facilitate model interpretation.

Interpretability is an important requirement of biomedical predictive systems. We further explored recent methodologies to determine the importance of features and how they can be used to provide insights and interpretability of the prediction mechanisms.

Through these analyses, we were able to identify the types of input data that are more predictive of drug response, as well as the feature selection and data representation methods that produce the best results. We also found that combining different models improves performance. Additionally, we demonstrate that the decisions made by the DL models are driven by biologically meaningful features.

6.2 Results

6.2.1 Testing the impact of different methodological variables

We developed several multimodal DL models (Figure 27) to predict drug combination effects summarized as ComboScores, using the ALMANAC dataset [7]. The ComboScore for a given <cell line-drugA-drugB> triplet is the sum of the differences between the expected and observed cell line growth calculated for each dose combination tested in the screen, with expected effects being determined using a modified version of the Bliss independence reference model [61].

First, we built several baseline models: a random baseline model that always predicts the average ComboScore value of the training set, and baseline models where one-hot encoded identifiers were used instead of cell line features and/or drug features.

We then built models trained on two different combinations of input data types:

1. *expr + drugA + drugB* models - use of RNAseq gene expression data (*expr*) and drug features for both drugs in a combination (drugA, drugB);

2. *expr + mut + cnv + drugA + drugB* - use of expression data (*expr*), mutation data (*mut*), copy number variation features (*cnv*) and drug features (*drugA*, *drugB*).

We began with type 1 models and tested the impact of the format of the transcriptomics data. We evaluated the influence of the type of architecture used to process the gene expression values (fully connected,

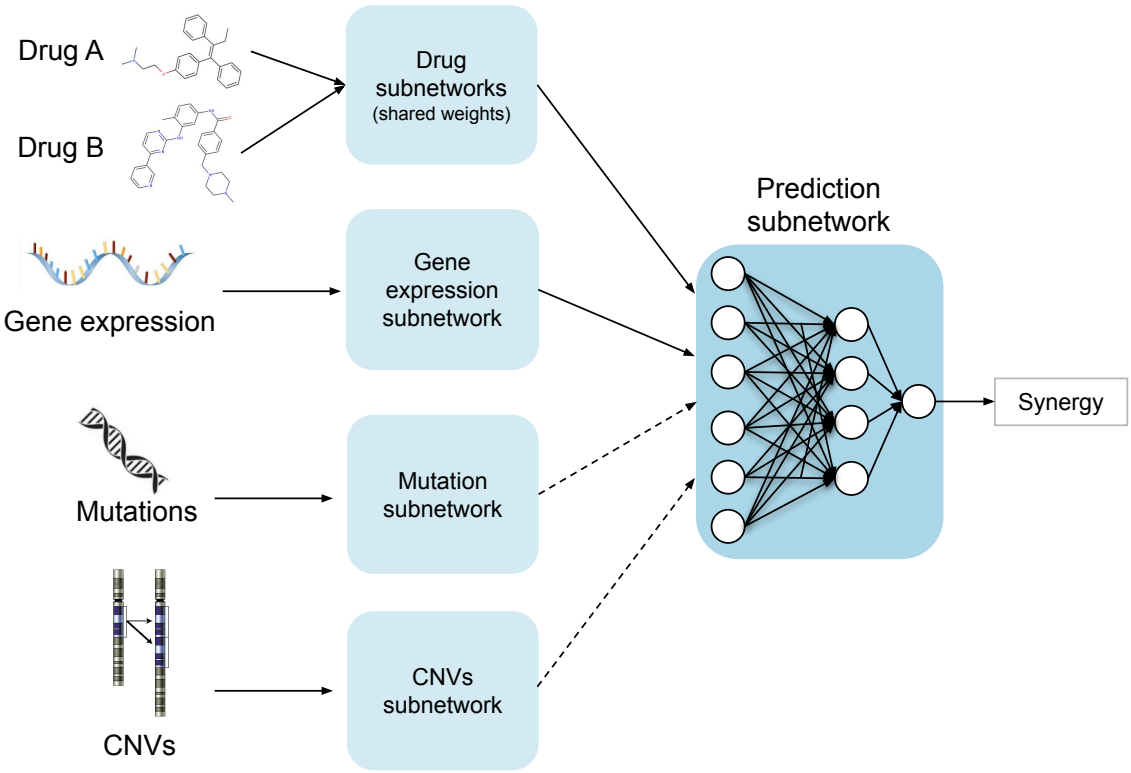


Figure 27: A representation of the general architecture of the multimodal DL models developed in this study. Here, we define a model as a combination of the learning algorithm itself and the data preparation steps required beforehand.

1D CNNs and 2D CNNs) and the influence of the arrangement and order of the genes (*chromosome position order* vs. *clustering order*).

We then tested the extension of the gene sets, by using more comprehensive or more selective lists of genes. We also tested if using directly normalized expression values or compact representations that capture the variability and correlation showed any differences in performance. The following feature selection/dimensionality reduction methods were evaluated in this study:

- $expr_{protein\ coding}$: the original dataset was filtered so that only protein coding genes were kept, leaving a total of 18,779 genes;

- $expr_{landmark}$: a list of "landmark genes" as defined by the L1000 project was used to reduce the gene expression dataset to 978 genes that are considered to be representative of the transcriptome as a whole [106];

- $expr_{DGI}$: Drug-gene interaction (DGI) claims were obtained from The Drug-gene Interaction Database (DGIdb) [230] (version 4.2.0) and then used to build a list of 994 DGI genes for the compounds screened in the ALMANAC project. 976 of these genes were present in the gene expression dataset;

- $expr_{COSMIC}$: A list of 723 cancer driver genes was obtained from the COSMIC [121] (version 94) Cancer Gene Census [231]. After filtering, the gene expression dataset contained 713 genes;
- $expr_{NCG}$: A different cancer-specific gene list containing 2,372 genes was obtained from the Network of Cancer Genes (NCG) (version 6.0) [232]. It includes genes identified in the COSMIC Cancer Gene Census as well as other genes that have been implicated in cancer. After filtering, 2,362 genes remained in the expression dataset;
- Combined gene lists: $expr_{DGI\ +\ landmark}$ (1,815 features) and $expr_{DGI\ +\ NCG}$ (3,037 features);
- $expr_{UMAP}$: Uniform Manifold Approximation and Projection (UMAP) [233] was used to capture the non-linear structure in the high-dimensional gene expression data, projecting it into a low-dimensional representation (50 components);
- $expr_{WGCNA}$: Weighted Gene Co-expression Network Analysis (WGCNA) [234] was used to find network modules that capture the correlated expression of a set of genes. Module eigengenes summarize the expression of genes in each module and can be used to reduce the dimensionality of the gene expression data. The reduced expressed dataset had a total of 76 module eigengenes.

Next, we kept the expression-encoding subnetwork fixed and modified the drug-encoding subnetworks to assess the impact of different drug representations:

- $drugs_{ECFP4}$: 1024-bit Morgan fingerprints with a radius of 2 (equivalent to ECFP4 fingerprints [176]) fed into fully connected drug subnetworks;
- $drugs_{LayeredFP}$: 1024-bit layered fingerprints fed into fully connected drug subnetworks;
- $drugs_{TextCNN}$: one-hot encoded SMILES strings fed into TextCNN [100], [182] subnetworks;
- $drugs_{GCN}$: using GCNs [183] for the drug subnetworks;
- $drugs_{GAT}$: using GATs [184] for the drug subnetworks;
- $drugs_{MTE}$: 512-dimensional Molecular Transformer Embeddings (MTEs) [235], fed into fully connected drug subnetworks.

Afterwards, we examined if adding more cell line data (mutation and CNVs) would lead to improved predictive performance (type 2 models). We tested two different models, one trained on mutation data summarized at the gene level and CNVs, both datasets being filtered to only include DGI genes ($expr_{protein\ coding}$ + $mut_{DGI,\ gene\text{-}level}$ + cnv_{DGI} + $drugs_{ECFP4}$ model) and another trained using mutations summarized at the pathway level and CNVs ($expr_{protein\ coding}$ + $mut_{pathway\text{-}level}$ + cnv_{DGI} + $drugs_{ECFP4}$ model).

We also compared our DL models against several ML algorithms.

In total, we developed 24 different DL models and 6 ML models, which are summarized in Figure 28. The results of these tests will be described in the following subsections.

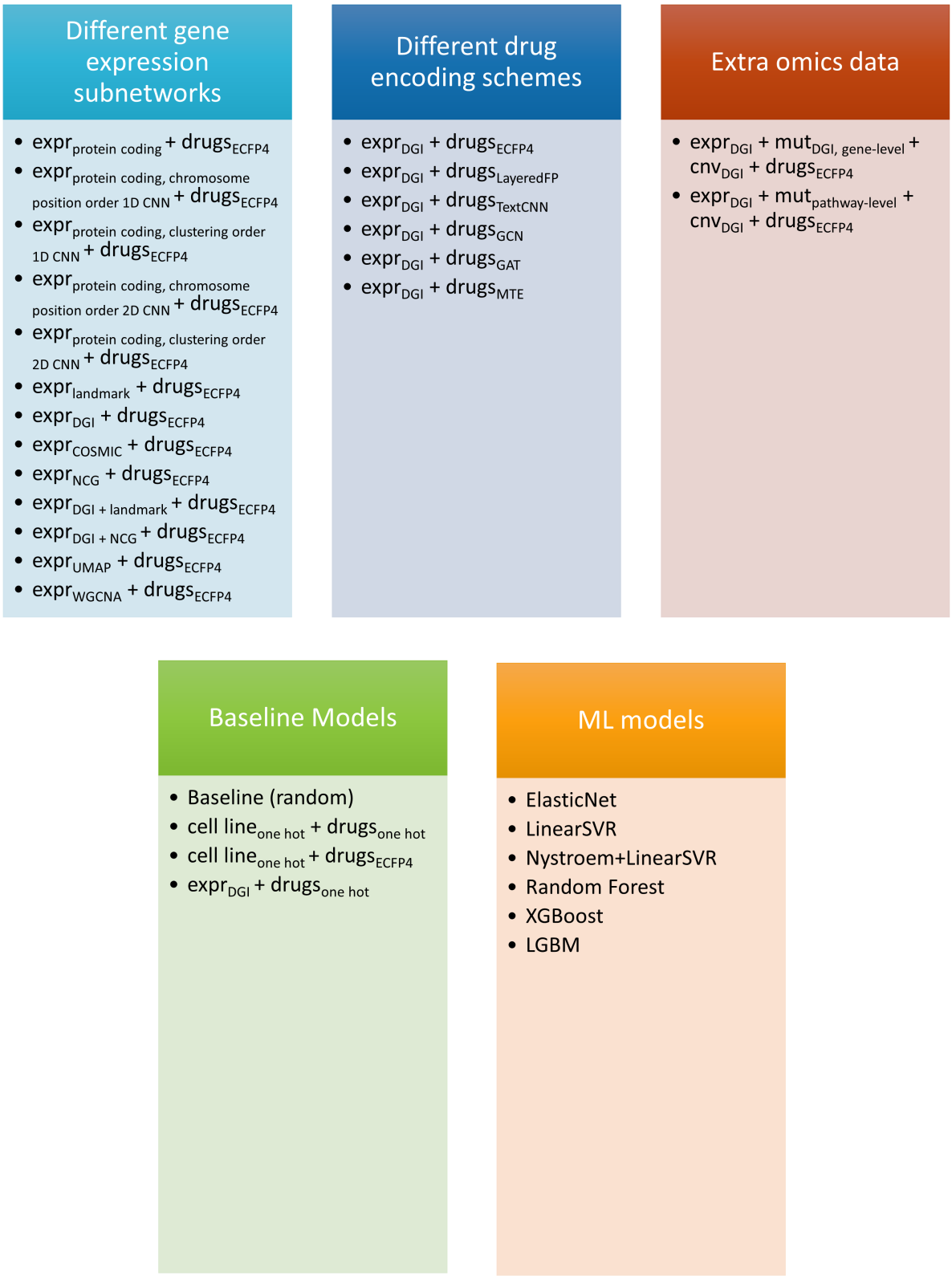


Figure 28: The different types of DL and ML models developed in this study.

6.2.2 Baseline models

The baseline models that were built served as references for subsequent models. All of the models developed in this study performed better than a random baseline model that always predicts the average ComboScore value of the training set (Figures 29 and 30).

To analyze the importance of including a certain input data type, we analyzed the performance of models trained on one-hot encoded identifiers instead of actual cell line or drug features:

- $cell\ line_{one\ hot}$ + $drugs_{one\ hot}$ - a model trained exclusively with one-hot encoded cell line and drug identifiersm which was used to determine if omics and chemical features include information that is relevant for the prediction of drug synergy, beyond the information contained in drug or cell line identifiers;
- $cell\ line_{one\ hot}$ + $drugs_{ECFP4}$ - to determine the impact of removing cell line omics features on the performance of the model;
- $expr_{DGI}$ + $drugs_{one\ hot}$ - to determine the impact of removing drug features;

All of the models performed better than the $cell\ line_{one\ hot}$ + $drugs_{one\ hot}$ model (Figures 29 and 30). Using one-hot encoded cell line names instead of omics features greatly decreases model performance (Figure 29). Nevertheless, the scores are comparable to those of some of the worst models that had been trained on gene expression data (WGCNA and 2D CNN models), revealing that the ECFP4 drug features contribute to the predictive capacity of the model. When using one-hot encoded drug identifiers instead of chemical features, model performance drops even more dramatically, nearly equaling the scores of the $cell\ line_{one\ hot}$ + $drugs_{one\ hot}$ model (Figure 30). This finding reveals that drug features are more predictive of drug combination effects than cell line features, at least for the ALMANAC dataset. A study by Xia et al. which used the ALMANAC data to predict growth inhibition had also reached this conclusion [137].

6.2.3 Different gene expression subnetworks

For the most part, models with fully connected gene expression subnetworks trained using the original gene expression values (tabular format) outperformed 1D and 2D convolutional gene expression subnetworks (Figure 29). This was independent of the order of genes obtained by clustering or by chromosome position. In light of these results, we only used fully connected layers for the omics subnetworks in subsequent tests.

The best gene expression subnetworks were fully connected subnetworks trained on the log-transformed and min-max scaled Fragments Per Kilobase of transcript per Million mapped reads (FPKM) values, with genes being selected according to predefined gene lists (Figure 29). The top three models were trained on expression data limited to landmark genes, COSMIC cancer genes or genes with known or potential interactions with drugs screened in the ALMANAC project ($expr_{DGI}$). Working with the full (> 18,000) set of protein coding genes present in this dataset was not shown to be particularly advantageous. Most of the feature selection methods that used smaller gene lists produced models with higher performance scores.

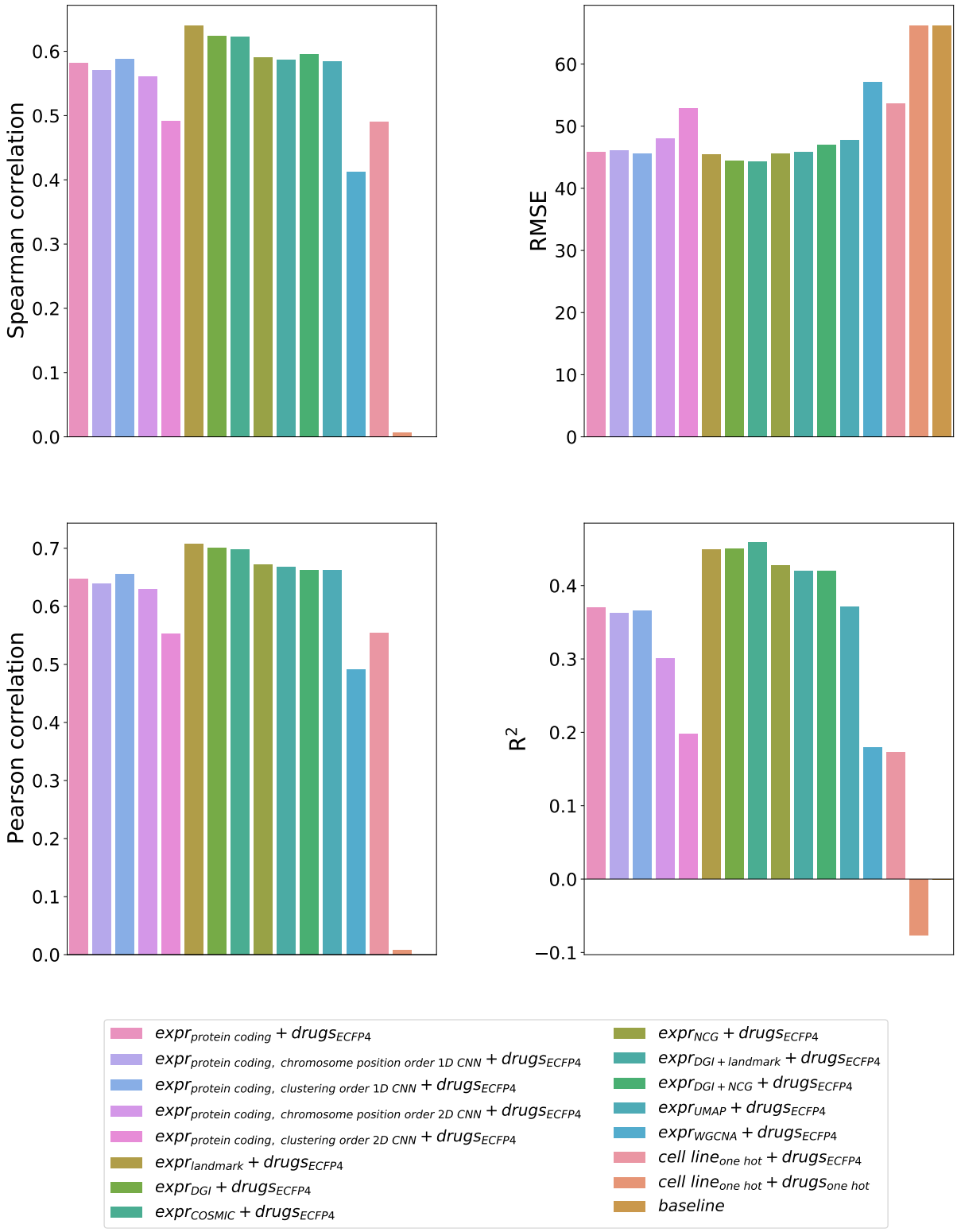


Figure 29: Performance scores of models with different gene expression feature-encoding subnetworks.

Using dimensionality reduction led to worse performance (Figure 29). UMAP embeddings were able to perform similar to some of the gene list selection methods, while using WGCNA as a dimensionality reduction method resulted in poor performance, even worse than a baseline model trained on one-hot

encoded cell line names.

In the following sections, we will only focus on models using $expr_{DGI}$ as the feature-encoding subnetwork for gene expression data. This gene selection method ranked second in terms of scoring metrics (RMSE; R^2, Pearson correlation and Spearman correlation), and might also provide better interpretability.

6.2.4 Drug encoding networks

The model trained on ECFP4 fingerprints outperformed all of the other subnetworks in terms of all of the scoring metrics considered in this study (Figure 30). Another fingerprint-based drug encoding scheme, LayeredFP, achieved the second-best RMSE and R^2 scores, while the GCN subnetwork achieved the second-best Pearson and Spearman correlation scores. The other GNN-based subnetwork, the GAT subnetwork, showed the worst performance in terms of all scoring metrics.

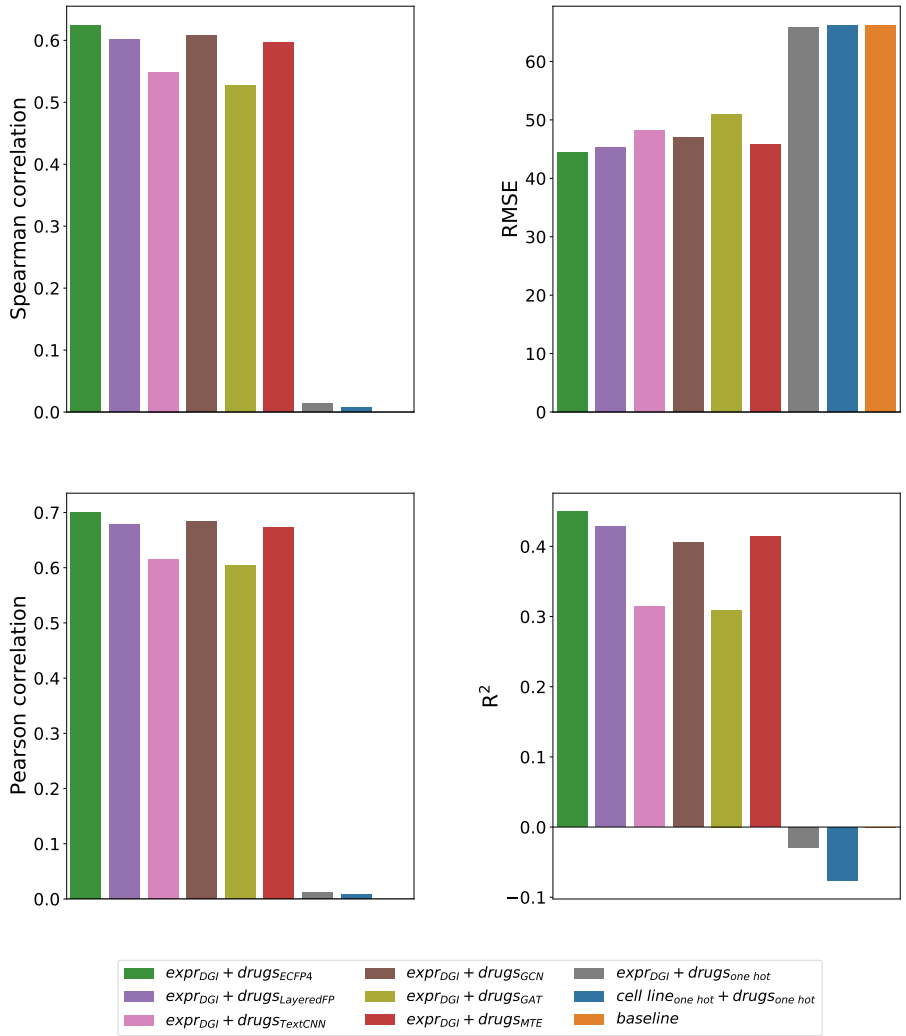


Figure 30: Performance scores of models with different drug encoding subnetworks.

The performance scores of the LayeredFP, MTE and GCN models were all very similar, and the difference in performance between these models and the ECFP4-based model was not substantial. This finding

is in agreement with a recent study by our group that compared the performance of different compound representations across different drug response prediction tasks and concluded that most representations perform similarly [169].

6.2.5 Additional cell line features

Including mutation and CNV data slightly decreased model performance (Figure 31). Previous studies have also reported that gene expression features are more predictive of cellular drug response than genetic features [12], [15]. Using pathway-level mutation features instead of gene-level features improved performance. This may possibly be due to the fact that more genes were taken into consideration when summarizing the mutation data at the pathway level, which would be an indication that relevant genetic information is being lost when only DGI genes are considered. The mutation dataset summarized at the pathway-level is also less sparse than the gene-level dataset.

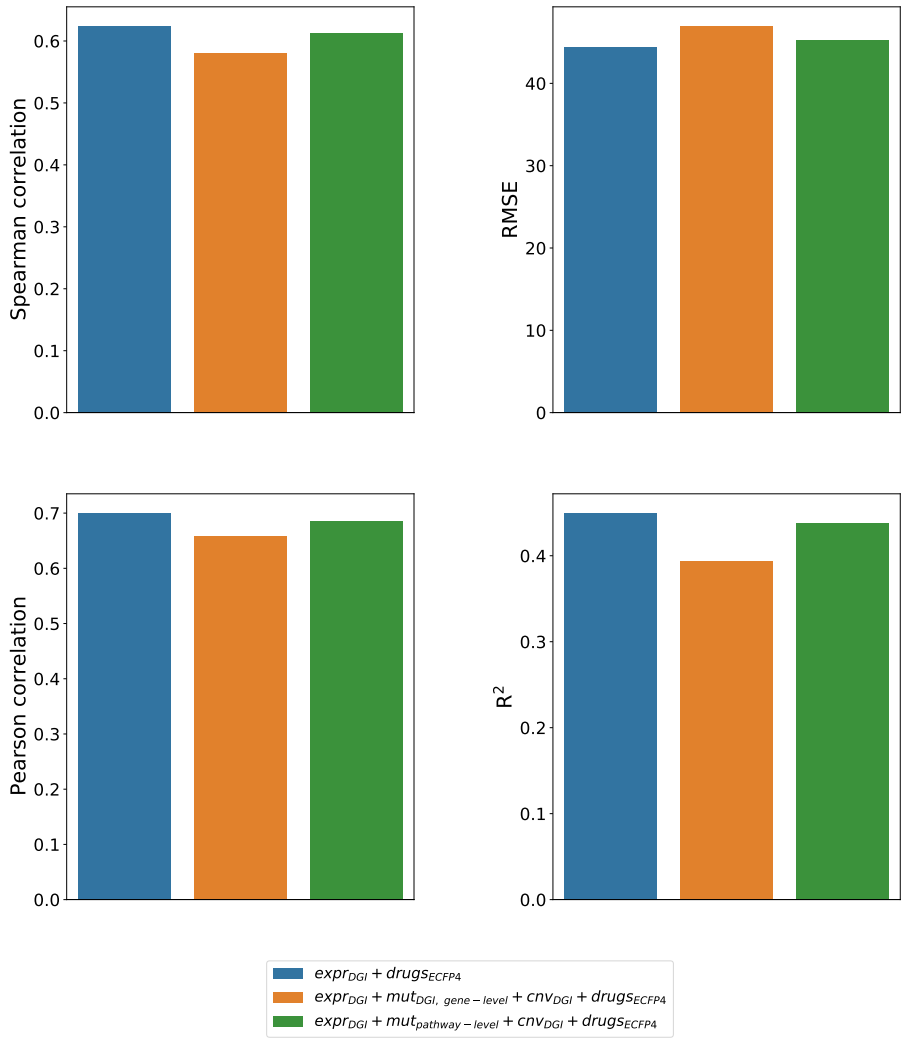


Figure 31: Performance scores of models trained with and without mutation (mut) and copy number variation (cnv) data in addition to gene expression and drug features.

6.2.6 Comparison with traditional machine learning models

To determine if there is any advantage in using DL instead of traditional ML to model drug combination effects, we compared our DL model trained on gene expression features of DGI genes and ECFP4 fingerprints ($expr_{DGI}$ + $drugs_{ECFP4}$ model) against 5 different ML models trained on the same features (Figure 32). We tested elastic net, linear SVR and linear SVR preceded by a radial basis function (RBF) kernel approximation method, RF, XGBoost and LGBM models. The DL model outperformed all of the ML models. The best traditional ML models were LGBM (in terms of RMSE, Pearson correlation and R^2) and RF (in terms of Spearman correlation). The performance of all of the tree-based models (RF, XGBoost and LGBM) is on par with some of the other DL models described in previous sections. The elastic net and SVR models did not perform as well, with performance scores that were not much better than those achieved by the baseline models.

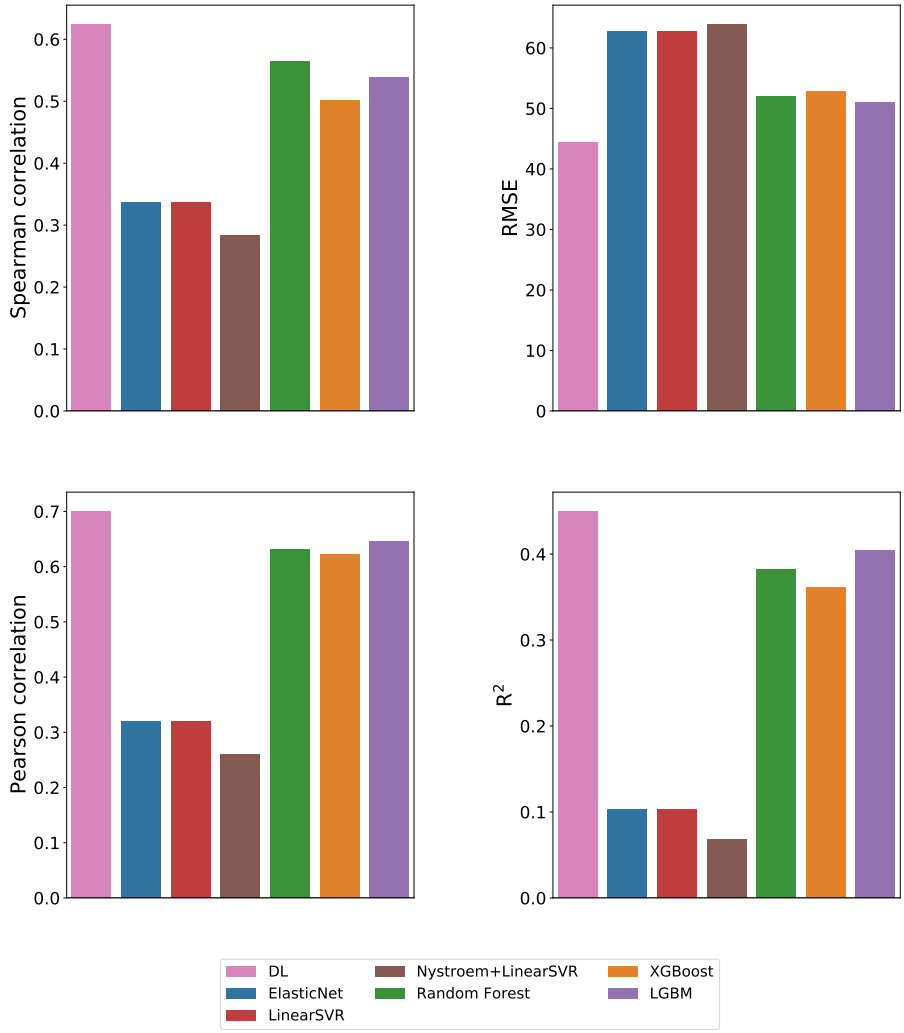


Figure 32: Performance scores of traditional ML models compared with one of the best DL models developed in this study. All of the models were trained on gene expression data filtered by drug-gene interactions present in DGIdb ($expr_{DGI}$) and ECFP4 fingerprints.

6.2.7 Heterogeneous ensemble

We also created a simple heterogeneous ensemble to determine if combining different DL approaches as well as other ML models could lead to better results. The ensemble included the top 10 DL models developed in this study and the RF, XGBoost and LGBM models. To obtain the ensemble prediction, the predictions from each of the individual models were simply averaged.

The heterogeneous ensemble outperformed the best individual DL models across all scoring metrics (RMSE=42.754, R^2=0.582, Pearson correlation=0.767, Spearman correlation=0.672 vs. RMSE=44.438, R^2=0.450, Pearson correlation=0.700, Spearman correlation=0.624 for the $expr_{DGI}$ + $drugs_{ECFP4}$ model, for example). These results show that combining different DL architectures and different learning algorithms can improve the generalizability of drug synergy prediction models.

6.2.8 Feature importance

One of the main disadvantages of using DL models to predict drug response is that they are difficult to interpret. The SHAP [163] interpretability framework was used to determine which features were the most important for the $expr_{DGI}$ + $drugs_{ECFP4}$ model. SHAP values reflect the contribution of each feature to a prediction.

Figure 33 shows the top 20 most important features. Important features are those that have greater absolute SHAP values. Each point in the figure represents an instance in the test set. The points are distributed along the X-axis according to the SHAP values. The colors represent the feature values. This reveals how the value of a feature influences the model output for a given sample. For example, when ECFP4 bit 250 is "ON" (value=1), the impact on the model output is usually positive, pushing the predicted ComboScore higher.

All of the top 20 features are drug features. All of the top 20 features (ranked by Gini importance) for the RF, one of the best ML models, were drug features as well, and many of these features overlap with the top 20 features determined for the DL model (Supplementary Table 15). This confirms that drug features are more predictive of drug response than individual gene expression features in the case of the ALMANAC dataset.

SHAP can also be used to provide explanations for specific examples. We analyzed the SHAP values obtained for test set example 11835. This experiment involved the SR cell line, derived from an anaplastic large cell lymphoma [103]. The drugs screened in this experiment were Topotecan hydrochloride, a topoisomerase I inhibitor [236], and Gefitinib, which inhibits the epidermal growth factor receptor (EGFR) tyrosine kinase domain [237]. This experiment was selected for further analysis since this combination was found to have greater-than-additive effects in the SR cell line in the ALMANAC screen (ComboScore=108.00), and because the ComboScore predicted by our model (ComboScore=108.06) was close to the true value.

Figure 34 displays the explanation of the prediction for test set instance 11835. The plot starts at a base value of -22.754, which is the expected model output that was determined based on a background

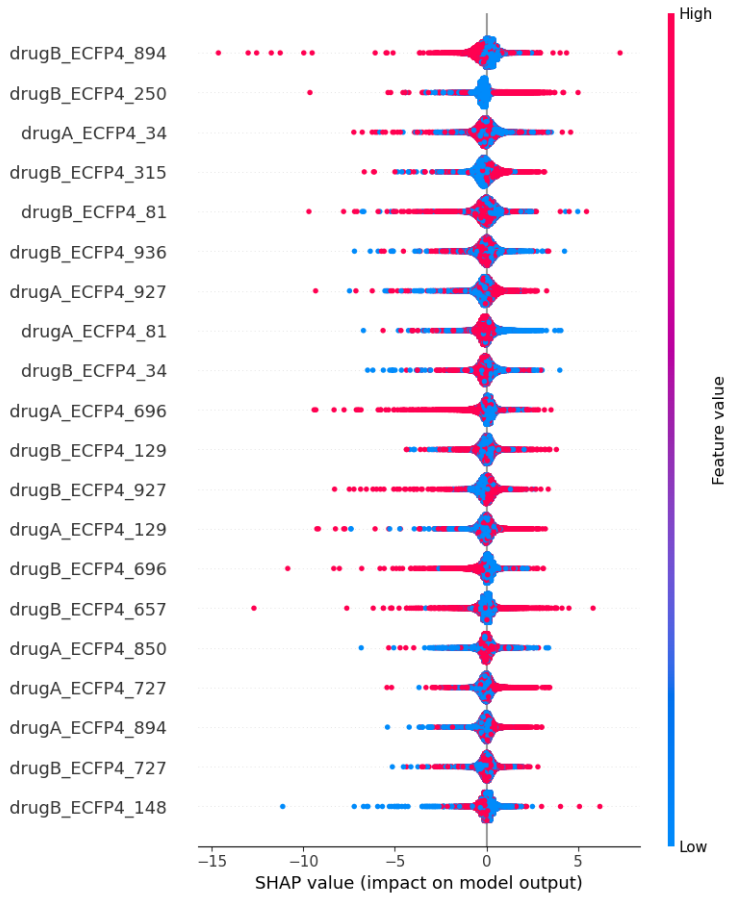


Figure 33: Top 20 most important features.

dataset. Each row shows how each feature contributed to moving the model output from the base value towards the predicted ComboScore value of 108.06. For this specific example, all of the top 20 features were, once again, ECFP4 fingerprint bits from drugA and drugB.

The SHAP values for each of the different input types were also analyzed separately. Figures 35 and 36 show the 20 most important ECFP4 bits for drugA (Topotecan hydrochloride) and drugB (Gefitinib), respectively, along with the corresponding SHAP values. These SHAP values provide information on how each specific bit influences the model output for example 11835. The 2D structures of 15 of the most important "ON" bits (value=1) that contributed to moving the predicted value towards the final prediction (in this case, those with positive SHAP values) are shown in Figure 37 for Topotecan and Figure 38 for Gefitinib.

As shown in Figure 37, bit 249 captures a part of the essential lactone ring, which establishes an important interaction with topoisomerase via hydrogen bonding [236]. Bit 658 also captures part of this ring. Topotecan targets the topoisomerase I-DNA complex, interacting with the complex at the site where the topoisomerase-mediated cleavage of the DNA occurs. Many of the remaining bits capture parts of the planar ring system of topotecan that structurally resembles a DNA base pair. This ring system is important as it allows topotecan to intercalate between two DNA base pairs and establish pi stacking interactions



Figure 34: Top 20 features (ranked by mean absolute SHAP values) for test set sample 11835 (all features).

with the neighboring DNA bases [236]. These results show that our model was able to correctly identify which substructures are most important for the bioactivity of topotecan.

The DL model was also able to identify features in Gefitinib that are more predictive of drug response and are consistent with the literature (Figure 38). Bits 795, 490 and 203 capture parts of the important quinazoline core that is involved in an important drug-protein interaction, with bit 203 focusing on the exact part of the quinazoline group (the nitrogen atom) that interacts with the adenosine triphosphate (ATP) binding site of EGFR via a hydrogen bond [238]. The majority of the bits in Figure 38, however, correspond to substituents that, according to the literature, are not involved in interactions between Gefitinib and EGFR. Many of the bits capture parts of the morpholine substituent that interacts with the solvent and not with the protein [238]. Bits 905 and 466 are centered on atoms belonging to a 3-chloro-4-fluoro aniline substituent that is positioned in the ATP-binding site when Gefitinib binds to EGFR [238], but is also not involved in



Figure 35: Top 20 drugA features (ranked by mean absolute SHAP values) for test set sample 11835.

drug-protein interactions. The DL model may have identified these features as more important because they might be structures that distinguish Gefitinib from other compounds in the ALMANAC dataset instead of being important for drug activity per se.

The top 20 gene expression features are shown in Figure 39. Interestingly, several of the most important genes have some kind of connection to the immune system. None of the genes that encode the primary targets of Topotecan and Gefitinib are found among the top 20 gene expression features. The TOP1 gene, which encodes DNA Topoisomerase I, was not present in the list of drug-gene interactions for Topotecan hydrochloride, which was the name provided by DTP for drug A and is the identifier that was used to obtain this information from the DGIdb database. The gene encoding DNA Topoisomerase I Mitochondrial (TOP1MT) was present in the list of drug-gene interactions, but it was only the 246th most important gene expression feature (527th most important feature overall). EGFR ranked higher, being the



Figure 36: Top 20 drugB features (ranked by mean absolute SHAP values) for test set sample 11835.

46th most important gene expression feature (184th overall) in this experiment. The expression of EGFR is low in this sample (log2(FPKM+1)) = 0.099), which was expected for this type of tissue. Additionally, none of the genes involved in the drug-gene interactions with Topotecan listed in DGIdb are among the top 20 most important gene expression features. Four of the top 20 genes are involved in drug-gene interactions with Gefitinib, however.

The most important gene expression feature was BLK, a proto-oncogene belonging to the Src family of tyrosine kinases that plays a role in B-cell development and B-cell receptor signaling and is normally not expressed in T-cells [239]. BLK has a drug-gene interaction with Gefitinib listed in DGIdb, and Gefitinib has been experimentally shown to be capable of binding to BLK [240], although with a much lower affinity than for EGFR. Since the expression of BLK in the SR cell line is high (min-max scaled log2(FPKM+1) value of 1), while the expression of EGFR is low in this cell line, the high SHAP value of BLK might be an

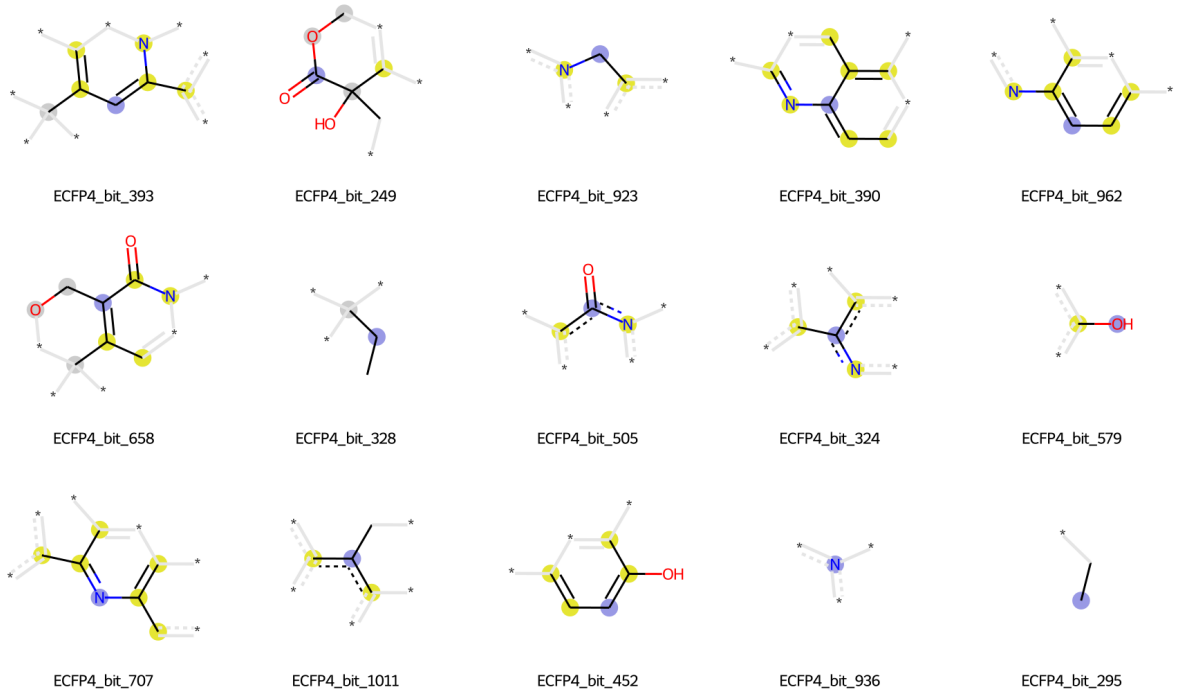


Figure 37: 2D depiction of the top 15 most important ECFP4 "ON" bits for drugA (Topotecan) in test set sample 11835.

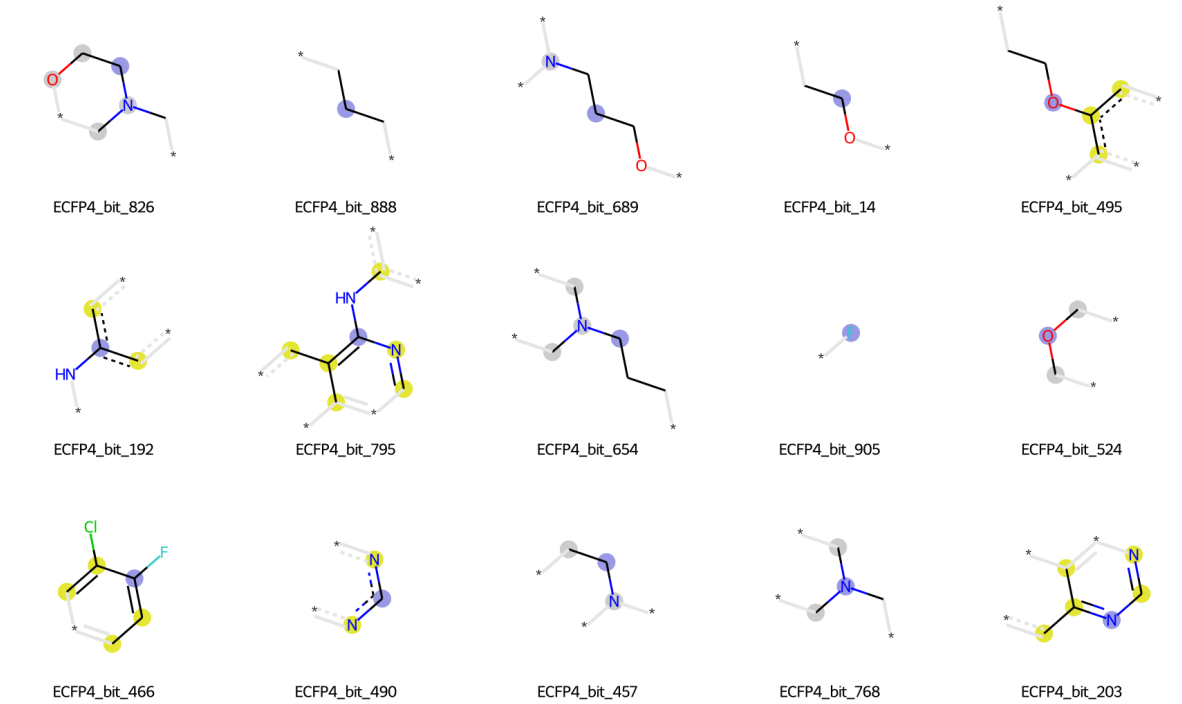


Figure 38: 2D depiction of the top 15 most important ECFP4 "ON" bits for drugB (Gefitinib) in test set sample 11835.



Figure 39: Top 20 gene expression features (ranked by mean absolute SHAP values) for test set sample 11835.

indication that this drug-gene interaction may be important for the response of the SR cell line to Gefitinib. CXCL10 and CXCL12 are both chemokines that are involved in the recruitment of lymphocytes [241], [242], and are highly expressed in lymphoid tissues [243], [244], according to The Human Protein Atlas [245]. Expression of TYMP is also abundant in lymphoid tissues, although it is not highly expressed in T-cells [246]. The TLR7 gene is involved in immune response [247] and is also expressed in lymphoid tissues, but not T-cells [248]. ALCAM also plays a role in immunity [249], but has low expression in T-cells [250], as is the case of the SR cell line. The ADORA2A gene plays a role in the regulation of the immune system [251] and is abundantly expressed in lymphoid tissues and T-cells [252].

These findings seem to indicate that the DL model identified genes associated with the immune system as being essential for the prediction of drug synergy in for sample 11835 and more important than the expression of the primary drug targets, possibly because the expression of these genes uniquely identifies the cell line in some way and not necessarily due to the relevance of these genes for drug response.

The model also gave more importance to genes that typically have lower expression in immune tissues/cells (NQO1, TPT, COMT, CCND1, MAPK1, GSTP1, MET, ERBB3), considering them essential to distinguish between the SR cell line and the rest of the NCI 60 cell lines. According to DGIdb, three of these underexpressed genes (MET, ERBB3 and CRKL) are potentially involved in drug-gene interactions with Gefitinib and are possible off-targets of the drug. Amplification of MET [253] or CRKL [254] has been linked to acquired resistance to EGFR inhibitors, and ERBB3 has also been implicated in resistance to EGFR tyrosine kinase inhibitors [253]. Lower expression of these genes is, therefore, probably essential for the SR cell line to be responsive to treatment with Gefitinib.

6.2.9 Selecting hyperparameter values

We analyzed the tuned hyperparameters to determine if it is possible to establish some useful guidelines for future drug synergy prediction models.

In fully-connected subnetworks, there was a preference for ReLU-based activation functions that allow for a small slope for negative values, especially the parametric rectified linear unit (pReLU) activation function. The majority of the models either used the recommended default learning rate value for the adaptive moment estimation (Adam) optimizer (0.001) or a lower learning rate (0.0001). Most models did not make use of dropout or used only small dropout rates. The top 3 DL models all used a dropout rate of 0.1, for example. Most of the models also applied a small l2 weight penalty. The number of hidden layers and number of units in each hidden layer varied greatly between models, even among the top 3 best models. Most models had 2 or 3 fully connected hidden layers in the prediction subnetwork, and only 1-2 hidden layers in the fully-connected feature-encoding subnetworks.

6.3 Discussion

Despite training the models on a relatively large screening dataset, comprising over 200,000 screening experiments, there are only 59 cell lines and 104 drugs in ALMANAC. The number of unique compounds screened in the ALMANAC project is relatively small, a fact that might explain why fingerprint-based models performed slightly better than learned representations in this case, as learned representations are known to struggle when faced with smaller compound sets [146]. Considering the small amount of unique cell lines and drugs, pre-training the feature encoding subnetworks using larger sets of compounds and larger omics datasets obtained from other sources could help improve model performance. Training models on larger and more diverse drug combination datasets from databases such as DrugCombDB [133] or DrugComb [255], which integrate data from several high-throughput screening experiments, could be another strategy to obtain models with better generalization capacity. However, it is important to note that different screening protocols might make it difficult to compare the synergy scores calculated for different screening datasets and to use them jointly to train models. Furthermore, integrating omics datasets from different studies requires the selection of adequate batch correction methods, which may not be straightforward [256].

Several of the compound representation methods tested in this work produced models with similar performance. Since different drug representations may capture different characteristics of the compounds that may be equally predictive of drug response, the combined use of different types of drug representations might lead to an improvement in model performance and could be an interesting strategy to explore in the future.

Using prior biological knowledge for feature selection proved to be beneficial. Besides the cancer-specific and DGI gene lists that were utilized in this study, other cancer-specific gene lists and combinations of lists might provide even better results. Pathway propagation methods have been employed in other drug response prediction studies to simulate response to treatment [257] and to extend the selection of genes beyond known drug targets [221]. This strategy was not explored in the current study, but it might also be a way to improve the predictive capacity of the models. It is important to note that using drug targets or DGIs to select genes may limit the generalizability of the model, as new compounds may have DGIs that are different from the ones that were considered relevant for the training set and with this approach these genes will be missed.

It is interesting to note that the 1D convolutional subnetworks performed better than the 2D convolutional subnetworks, which may be an indication that the reshaping involved in transforming the gene expression vectors into images may have forced unrelated genes into closer proximity, leading to worse results. Other factors, such as the architectural choices made when constructing these subnetworks, may impact the performance of the 2D CNN subnetworks. In the future, it might be interesting to test more elaborate methods of transforming tabular omics data into images, such as REFINED [223] or Image Generator for Tabular Data (IGTD) [258].

Another approach that was not explored in this study is directly including biological knowledge in the

neural network by representing the experiments as heterogeneous graphs and modeling drug response using GNNs. Since the number of unique compounds and cell lines in the ALMANAC dataset is relatively small, it is unclear if the models would benefit from the added complexity of this approach.

Although SHAP has allowed us to gain more insight into the model and to try to interpret its predictions in a biologically meaningful way, it is still difficult to explain how all of the most important features interact in a particular screening experiment and give rise to drug synergy. Interpretability methods allow us to explain how a DL model works, but they do not necessarily explain the underlying biology. It is therefore unlikely that the methodology used in our study will be able to uncover the mechanisms underlying drug synergy for a particular <cell line-drugA-drugB> triplet, but our findings do confirm that DL models are able to learn to associate important structural features of the compounds and tissue-specific gene expression patterns with drug response.

Sparse-input neural networks have been proposed as a way to improve the biological interpretability of DL models and would be another way to incorporate prior knowledge directly within the network. Some studies have shown that sparse architectures may also improve predictive performance [259], [260], although other authors have reached the opposite conclusion [261]. Zhang et al applied this strategy to model ALMANAC ComboScores [220] and did not see an improvement in model performance. Furthermore, even though model connections were limited, the authors still found it difficult to associate feature importance scores with the biological mechanisms underlying drug synergy [220]. Therefore, we did not explore this approach in this work.

Finally, we found that creating simple heterogeneous ensembles comprising both DL and ML models can improve performance. Preto et al. had previously observed that heterogeneous ensembles produced better classifiers than individual models when using the ALMANAC dataset [219], and other synergy prediction studies using different datasets also found that combining multiple models is beneficial [8], [9].

6.4 Methods

6.4.1 Dataset & preprocessing

ALMANAC drug response data in the form of ComboScores for <cell line, drugA, drugB> triplets were downloaded from CellMinerCDB [262].

Since ComboScore predictions should be independent of the order of the drugs in a given combination, we considered reverse drug order <cell line, drugB, drugA> triplets to be duplicates and the ComboScores for experiments involving the same cell line and same drug combination were averaged. The MDA-MB-468 cell line that is present in the original ALMANAC dataset was not considered in our analysis as it does not appear in the omics datasets we used in this study. The MDA-N cell line present in the omics datasets was removed as it was not included in the ALMANAC project.

National Service Center (NSC) compound identifiers were used to map compounds to their respective SMILES strings using a compound structure-data file (SDF) file provided by NCI DTP for the ALMANAC

dataset. For the compounds that were not successfully mapped using the previously described method, PubChemPy was used to obtain canonical SMILES strings from PubChem [208] based on drug names. The SMILES strings were then preprocessed using the ChEMBL Structure Pipeline [175], which removed salts and standardized the molecules according to ChEMBL-defined rules. The preprocessed SMILES were then used to compute different compound representations, depending on the type of architecture that was chosen for the drug subnetwork.

RNA-Seq gene expression data (log2(FPKM+1)) for the NCI-60 cell lines were also downloaded from CellMinerCDB [262]. Genes with constant expression values across all cell lines were removed. The expression dataset was then filtered using specific gene lists, as detailed in the Results section.

Besides the use of specific gene sets to reduce the number of features in the gene expression dataset, several other dimensionality reduction methods were also tested. UMAP embeddings with 50 components were generated using the UMAP Python package (version 0.5.1) [233]. The algorithm was fit on the training data and then used to transform the training, validation and test sets. A total of 20 neighbors were considered when learning the manifold structure of the data. Pearson correlation was used as the distance metric, and the minimum distance between two points in the embedding was kept at the default value of 0.1.

We also evaluated the use of WGCNA [234] as a dimensionality reduction technique for gene expression data. The WGCNA R package (version 1.69) was used to find a total of 76 gene co-expression modules. Module eigengenes, which summarize the expression of genes in each module, were then used as input features instead of the original gene expression values.

When preparing the gene expression data to be fed into 1D or 2D CNN, two different methods were used to rearrange the genes so that their locations within the input tensors might be more biologically meaningful. The first method sorted the genes according to their chromosome positions, an approach similar to the method described in [263]. The second method employed hierarchical clustering to find a new order for the genes. The clustering algorithm used Pearson correlation as the distance metric and complete linkage. A dendrogram was created based on the clustering results, and the positions of the leaves (genes) was used to reorder the gene expression features. The gene expression data were then reshaped into the required formats for 1D and 2D CNNs.

A mutation annotation format (MAF) file containing mutation data from whole-exome sequencing for NCI-60 cell lines was downloaded from CBioPortal [264], [265]. Silent mutations and mutations in non-coding regions were excluded. The remaining data were then binarized and summarized at the gene level, with '1' indicating that a given cell line had at least one alteration in a particular gene and zero indicating the absence of mutations. To create the gene-level mutation dataset, the data were further filtered using a list of drug-gene interactions from DGIdb, leaving a total of 636 genes. To create the pathway-level dataset, a new binary event matrix was created, where a '1' indicates that a given cell line had at least one alteration in a given pathway. Genes were mapped to pathways using Molecular Signatures Database (MSigDB) [266], [267] (version 7.2) gene sets derived from the Reactome [268] database, which have been filtered to reduce the redundancy between gene sets. The pathway-level mutation dataset had a total

of 1,510 features.

Genomic Identification of Significant Targets in Cancer (GISTIC)2 putative CNVs for the NCI-60 cell lines were also obtained from CBioPortal [264], [265]. Only genes that were present in the drug-gene interactions list obtained from DGIdb were kept, leaving a total of 952 features.

The omics datasets were individually merged with the drug response dataset to create the final omics datasets containing the corresponding cell line data for each experiment. Afterwards, the data were split into three subsets - a training set comprising around 80% of the data (239,943 examples), and validation and test sets each corresponding to approximately 10% of the original data (30,007 and 30,001 examples, respectively). After splitting the expression dataset into training/validation/test sets, features were scaled to a range between zero and one (min-max scaling).

6.4.2 Models

The DL models were built using a multimodal approach, in which each data type has its own corresponding feature-encoding subnetwork. The learned representations from each subnetwork were then concatenated into a single tensor before being fed into a final fully connected drug synergy prediction subnetwork.

Several different architectures were tested for each of the subnetworks. The gene expression subnetworks were either fully connected networks, 1D CNNs, or 2D CNNs. When included, the mutation and CNV subnetworks were always fully connected. For the drug subnetworks we tested fully connected networks (when using ECFP4 fingerprints, LayeredFP or MTEs as inputs), GCN [183] subnetworks, GAT [184] subnetworks, and TextCNN [182] subnetworks (as implemented in DeepChem [100]). Each drug in a combination had a separate feature-encoding subnetwork, with weights being shared between the two drug subnetworks. The prediction subnetwork was always entirely composed of fully connected layers, ending in a single output unit with a linear activation function, given that the prediction task was approached as a regression problem.

All of the models were implemented using the Keras subpackage in Tensorflow [98] (version 2.2.0). The GCN and GAT subnetworks were built using graph layers implemented in the Spektral package [269], version 1.0.

6.4.3 Model hyperparameters & hyperparameter optimization

All models used the MSE as the loss function, and the Adam algorithm as the optimization method. Models were trained for a maximum of 500 epochs, using the EarlyStopping callback with a patience of 15 epochs to stop the learning process once the loss stopped improving. The mini-batch size was 64.

A selection of model hyperparameters, including the learning rate, the hidden layer activation function, the dropout rate and the L2 regularization penalty, as well as subnetwork-specific hyperparameters, were optimized for each combination of input data types and subnetwork architectures that we evaluated in this work. Hyperparameters were tuned using the validation set and the Bayesian Optimization with

HyperBand (BOHB) algorithm [270] as implemented in Ray Tune (version 1.0.1). The hyperparameter search space was explored using Bayesian optimization [271] and the HyperBand [272] algorithm was used to stop underperforming trials early. A total of 50 configurations were evaluated for each model. The set of hyperparameters that minimized the validation loss was considered the best configuration.

6.4.4 Model evaluation

After tuning the model hyperparameters, the models were evaluated on the independent test set. Model performance was evaluated using several scoring metrics, including the MSE, R^2, Pearson's correlation coefficient and Spearman's rank correlation coefficient.

In addition to comparing DL models with different omics and drug subnetworks, our models were also compared against traditional ML models. Elastic net linear regression [26], SVR [204], and RF [32] models were implemented using the scikit-learn Python package (version 0.23.2) [186]. Since kernelized SVR does not scale well to larger datasets, we used the Nystroem method [273] to approximate the feature mappings of a RBF kernel before training a linear SVR on these approximations. An XGBoost [205] model was implemented using the xgboost Python package (version 1.4.0), and a LGBM [206] model was implemented using the lightgbm package (version 3.2.1). The ML models were trained on ECFP4 fingerprints and the gene expression values of DGI genes based on information from DGIdb. The input features were concatenated into a single dataset before training. Hyperparameters were optimized using Bayesian optimization.

The models were also compared against a baseline model that always predicts the average ComboScore value of the training set, and against models where the cell line and/or drug features were substituted with the corresponding one-hot encoded identifiers (cell line (one-hot encoded) + drugs (one-hot encoded), cell line (one-hot encoded) + drugs (ECFP4), and expr (targets) + drugs (one-hot encoded) models).

6.4.5 Feature importance

The SHAP Python package (version 0.39.0) [163] was used to explain predictions made by the best DL models developed in this work. More specifically, we used Deep SHAP, which uses the DeepLIFT [161] additive feature attribution method to approximate SHAP values. Feature importance was then determined based on these values. The most important fingerprint bits were visualized using drawing functions available in RDKit (version 2020.09.1.0).

6.4.6 Code availability

All of the code developed in this study is available online at https://gitlab.bio.di.uminho.pt/dlrsb/drug_response_pipeline. The full results tables and more details on the hyperparameters are also provided in this repository.

7

Conclusions and Future Work

The overall aim of this thesis was the development of DL methods to predict drug sensitivity and synergy from multiomics datasets characterizing cancer cells and chemical features of drugs. A variety of models were built and evaluated for 6 different drug sensitivity prediction tasks and a drug synergy prediction problem. Although the models developed in this work are still far from being clinically applicable, the work described here will be useful to guide the development of future computational methods for the rational design of novel cancer treatments.

In the following sections, the main contributions of this thesis will be summarized and suggestions for further improvement will be proposed.

7.1 Main contributions

One of the main contributions of this work was the systematic analysis of the impact of different methodological choices on the predictive performance of DL-based drug response prediction models. A variety of preprocessing and modeling strategies were tested, in an attempt to understand which approaches produce the best results. In chapter 4, single-cell line drug sensitivity datasets were used to benchmark the performance of different compound representations. In Chapter 5, DL was used to model drug response in patient-derived tumor samples. During the development of a solution for this problem, the influence of different input data types, feature selection methods and model architectures on the performance of the models was assessed. In chapter 6, using a large drug combination screening dataset, a similar analysis was performed for the problem of drug synergy prediction.

The following general conclusions can be drawn from these analyses:

- It was confirmed that DL algorithms are well-suited to these types of prediction tasks.

- The most important types of inputs for drug response prediction depend on the problem itself, as well as the quality of the data. For instance, genetic variants were considered important inputs in Chapter 5, while in Chapter 6 the addition of these features decreased model performance.

One explanation for this may be that the Beat AML mutation dataset only included high-confidence mutational events, whereas the ALMANAC mutation and CNV datasets were less filtered.

- Despite the ability of DL models to learn relevant features directly from raw data, the use of prior biological knowledge to select features is still useful to reduce the size of high-dimensional omics datasets, particularly RNA-seq datasets.

- Most drug representation methods perform similarly. End-to-end methods are capable of outperforming traditional molecular fingerprints when trained on datasets with a few thousands of compounds. However, for datasets with a very limited number of unique drugs using learned representations might not be the best approach and the use of molecular fingerprints might be preferable.

- Since many of the different methodological choices tested in this work produced models with similar performance, combining multiple models into ensembles appears to be a good strategy to improve performance. Distinct models may learn different aspects of the problem, and thus combining them may increase the generalization capacity of the final model.

In this thesis, the issue of model interpretability was also addressed. Using a state-of-the-art feature attribution method, we were able to demonstrate that the decisions made by our DL models are driven by biologically relevant features.

Although software development was not the main focus of this thesis, the code that was developed throughout this work is made freely available and can be reused by the scientific community to reproduce and extend our work. The work undertaken in Chapter 4 contributed to extend and validate the DeepMol software package developed by our research group. In this thesis, support for regression models and cross-validated hyperparameter optimization for DeepChem models was improved and DL-specific interpretability methods were added to the package. All of the scripts developed for Chapter 5 were made publicly available. In Chapter 6, a drug response modeling pipeline was created. The code repository offers access to reusable model architectures, as well as separate feature-encoding subnetworks that can be combined in different ways to create new models. It also provides classes to load multi-input data for DL models, to preprocess drug response, compound, and omics data, to perform hyperparameter optimization for both DL and ML models, and to interpret DL models using DL-specific feature attribution algorithms. The code can be installed as a Python package. In addition, scripts to reproduce our analyses for the ALMANAC dataset are also provided. To guarantee the reproducibility of our analyses, we also provide Docker images replicating the environments in which our analyses were performed.

Finally, Chapter 3 is an in-depth review of recently published studies that have employed DL methods to predict drug response in cancer cell lines. The article addresses some of the main limitations of these models, and gives suggestions for further improvement. In addition, it provides brief descriptions of the main types of DL architectures that have been used in these studies and presents a selection of publicly available data resources that can be used in the development of drug response prediction models.

7.2 Publications

Several chapters from this thesis have either already been published, or are currently being prepared for submission to scientific journals:

- Chapter 3 was published as a review paper in Briefings in Bioinformatics: D. Baptista, P. G. Ferreira, and M. Rocha, "Deep learning for drug response prediction in cancer", *Brief. Bioinform.*, vol. 22, no. 1, pp. 360–379, Jan. 2021. doi: 10.1093/bib/bbz171

- Chapter 4 is an extended version of a published article: D. Baptista, J. Correia, B. Pereira, *et al.*, "A Comparison of Different Compound Representations for Drug Sensitivity Prediction", in *Practical Applications of Computational Biology & Bioinformatics, 15th International Conference (PACBB 2021)*, M. Rocha, F. Fdez-Riverola, M. S. Mohamad, *et al.*, Eds., Cham: Springer International Publishing, 2022, pp. 145–154. doi: 10.1007/978-3-030-86258-9_15. This extended version is currently being prepared for submission to a scientific journal.

- Part of the work described in Chapter 5 will be included in an overview paper on the results from the CTD^2 Beat AML DREAM Challenge. The data and results from the challenge are currently under a publication embargo, but challenge participants will be allowed to publish their work after the publication of the overview paper or at the end of the 2 year embargo, whichever comes first.

- At the time of writing, Chapter 6 is being prepared for submission to a scientific journal.

In addition, the author of this thesis also contributed to the following scientific publications during the PhD:

- S. Cardoso, D. Baptista, R. Santos, *et al.*, "A Review on Metabolomics Data Analysis for Cancer Applications", in *Practical Applications of Computational Biology and Bioinformatics, 12th International Conference*, F. Fdez-Riverola, M. S. Mohamad, M. Rocha, *et al.*, Eds., Cham: Springer International Publishing, 2019, pp. 157–165. doi: 10.1007/978-3-319-98702-6_19

- J. Correia, T. Resende, D. Baptista, *et al.*, "Artificial Intelligence in Biological Activity Prediction", in *Practical Applications of Computational Biology and Bioinformatics, 13th International Conference*, F. Fdez-Riverola, M. Rocha, M. S. Mohamad, *et al.*, Eds., Cham: Springer International Publishing, 2020, pp. 164–172. doi: 10.1007/978-3-030-23873-5_20

- Z. Xiong, M. Jeon, R. J. Allaway, *et al.*, "Crowdsourced identification of multi-target kinase inhibitors for RET- and TAU- based disease: The Multi-Targeting Drug DREAM Challenge", *PLOS Comput. Biol.*, vol. 17, no. 9, E. Wang, Ed., e1009302, Sep. 2021. doi: 10.1371/journal.pcbi.1009302 (as a member of the Multi-Targeting Drug DREAM Challenge community)

7.3 Future work

The insights gained from this research can be used as a starting point for the development of better drug response prediction models in the future. In this section, we suggest ways to extend and improve upon the work described in this thesis.

A major limitation of our work was the use of screening datasets with few unique compounds and/or cell lines. We propose two ways of overcoming this that could possibly improve the predictive performance of the models: 1) combining multiple high-throughput screening datasets to have access to more training data, for example by using existing integrative databases; 2) pretraining the feature-encoding subnetworks using unlabeled compound and cell line data obtained from other sources, and then fine-tuning them for each specific problem. These approaches will require methods to reduce potential batch effects and adequately integrate data from experiments performed under different conditions.

Another strategy that could increase generalizability would be the use of multitask models that can simultaneously predict both drug sensitivity and synergy. This would allow the models to learn from data from different screening projects and to share information across different tasks by learning a common representation, which could facilitate generalization to new scenarios.

The use of prior biological knowledge for feature selection proved to be beneficial in Chapter 6. In the future, other ways of leveraging this knowledge should be explored, including the use of alternative gene lists for feature selection and methods that can incorporate biological knowledge directly within the neural networks.

Ensemble learning was shown to improve performance in Chapters 4, 5 and 6. Thus, we recommend the use of larger and more elaborate ensembles in the future. Instead of using simple averaging methods, more complex ensemble methods could be used, such as stacked generalization, which uses the predictions of individual models as inputs to another model that computes the final predictions [31].

Although the feature attribution method that was employed in Chapters 4, 5 and 6 allowed us to determine the importance of the input features and their impact on the predictions of the models, it did not provide enough information to explain higher order interactions among features. In the future, alternative interpretability methods may be explored, with a special focus on developing new approaches that can provide insights into how features interact and give rise to drug response.

Finally, the development of a more general and easy-to-use software package for drug response modeling with DL that could be applied to any kind of drug response dataset would be an important contribution to the field and would allow our code to be more widely used by the community.

Bibliography

[1] D. C. Collins, R. Sundar, J. S. Lim, and T. A. Yap, "Towards Precision Medicine in the Clinic: From Biomarker Discovery to Novel Therapeutics", *Trends Pharmacol. Sci.*, vol. 38, no. 1, pp. 25–40, Jan. 2017. doi: 10.1016/j.tips.2016.10.012.

[2] D. Senft, M. D. Leiserson, E. Ruppin, and Z. A. Ronai, "Precision Oncology: The Road Ahead", *Trends Mol. Med.*, vol. 23, no. 10, pp. 874–898, Oct. 2017. doi: 10.1016/j.molmed.2017.08.003.

[3] J. Barretina, G. Caponigro, N. Stransky, *et al.*, "The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity", *Nature*, vol. 483, no. 7391, pp. 603–307, Mar. 2012. doi: 10.1038/nature11003.

[4] W. Yang, J. Soares, P. Greninger, *et al.*, "Genomics of Drug Sensitivity in Cancer (GDSC): a resource for therapeutic biomarker discovery in cancer cells", *Nucleic Acids Res.*, vol. 41, no. D1, pp. D955–D961, Jan. 2013. doi: 10.1093/nar/gks1111.

[5] B. Seashore-Ludlow, M. G. Rees, J. H. Cheah, *et al.*, "Harnessing connectivity in a large-scale small-molecule sensitivity dataset", *Cancer Discov.*, vol. 5, no. 11, pp. 1210–1223, 2015. doi: 10.1158/2159-8290.CD-15-0235.

[6] J. O'Neil, Y. Benita, I. Feldman, *et al.*, "An Unbiased Oncology Compound Screen to Identify Novel Combination Strategies", *Mol. Cancer Ther.*, vol. 15, no. 6, pp. 1155–1162, Jun. 2016. doi: 10.1158/1535-7163.MCT-15-0843.

[7] S. L. Holbeck, R. Camalier, J. A. Crowell, *et al.*, "The National Cancer Institute ALMANAC: A Comprehensive Screening Resource for the Detection of Anticancer Drug Pairs with Enhanced Therapeutic Activity", *Cancer Res.*, vol. 77, no. 13, pp. 3564–3576, Jul. 2017. doi: 10.1158/0008-5472.CAN-17-0489.

[8] M. P. Menden, D. Wang, M. J. Mason, *et al.*, "Community assessment to advance computational prediction of cancer drug combinations in a pharmacogenomic screen", *Nat. Commun.*, vol. 10, no. 1, p. 2674, Dec. 2019. doi: 10.1038/s41467-019-09799-2.

[9] M. Bansal, J. Yang, C. Karan, *et al.*, "A community computational challenge to predict the activity of pairs of compounds", *Nat. Biotechnol.*, vol. 32, no. 12, pp. 1213–1222, Dec. 2014. doi: 10.1038/nbt.3052.

[10] M. J. Garnett, E. J. Edelman, S. J. Heidorn, *et al.*, "Systematic identification of genomic markers of drug sensitivity in cancer cells", *Nature*, vol. 483, no. 7391, pp. 570–575, Mar. 2012. doi: 10.1038/nature11005.

[11] M. P. Menden, F. Iorio, M. Garnett, *et al.*, "Machine Learning Prediction of Cancer Cell Sensitivity to Drugs Based on Genomic and Chemical Properties", *PLoS One*, vol. 8, no. 4, G. P. S. Raghava, Ed., e61318, Apr. 2013. doi: 10.1371/journal.pone.0061318.

[12] F. Iorio, T. A. Knijnenburg, D. J. Vis, *et al.*, "A Landscape of Pharmacogenomic Interactions in Cancer", *Cell*, vol. 166, no. 3, pp. 740–754, Jul. 2016. doi: 10.1016/j.cell.2016.06.017.

[13] C. Holohan, S. Van Schaeybroeck, D. B. Longley, and P. G. Johnston, "Cancer drug resistance: an evolving paradigm", *Nat. Rev. Cancer*, vol. 13, no. 10, pp. 714–726, Oct. 2013. doi: 10.1038/nrc3599.

[14] R. J. Tallarida, "Quantitative Methods for Assessing Drug Synergism", *Genes Cancer*, vol. 2, no. 11, pp. 1003–1008, Nov. 2011. doi: 10.1177/1947601912440575.

[15] J. C. Costello, L. M. Heiser, E. Georgii, *et al.*, "A community effort to assess and improve drug sensitivity prediction algorithms", *Nat. Biotechnol.*, vol. 32, no. 12, pp. 1202–1212, Dec. 2014. doi: 10.1038/nbt.2877.

[16] A. Kalamara, L. Tobalina, and J. Saez-Rodriguez, "How to find the right drug for each patient? Advances and challenges in pharmacogenomics", *Curr. Opin. Syst. Biol.*, vol. 10, pp. 53–62, Aug. 2018. doi: 10.1016/j.coisb.2018.07.001.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015. doi: 10.1038/nature14539.

[18] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013. doi: 10.1109/TPAMI.2013.50.

[19] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. MIT Press, 2012, p. 1.

[20] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning", in ser. Springer Series in Statistics, New York, NY: Springer New York, 2009, p. 18. doi: 10.1007/978-0-387-84858-7.

[21] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics", *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, Oct. 2007. doi: 10.1093/bioinformatics/btm344.

[22] R. Kohavi and F. Provost, “Glossary of Terms”, *Mach. Learn.*, vol. 30, no. 2/3, pp. 271–274, 1998. doi: 10.1023/A:1017181826899.

[23] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian Optimization of Machine Learning Algorithms”, Jun. 2012.

[24] A. E. Hoerl and R. W. Kennard, “Ridge Regression: Biased Estimation for Nonorthogonal Problems”, *Technometrics*, 1970. doi: 10.1080/00401706.1970.10488634.

[25] R. Tibshirani, *Regression Selection and Shrinkage via the Lasso*, 1996. doi: 10.2307/2346178.

[26] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net”, *J. R. Stat. Soc. Ser. B (Statistical Methodol.*, vol. 67, no. 2, pp. 301–320, Apr. 2005. doi: 10.1111/j.1467-9868.2005.00503.x.

[27] D. R. Cox, “The Regression Analysis of Binary Sequences”, *J. R. Stat. Soc. Ser. B*, vol. 20, no. 2, pp. 215–242, 1958.

[28] S. B. Kotsiantis, “Decision trees: a recent overview”, *Artif. Intell. Rev.*, vol. 39, no. 4, pp. 261–283, Apr. 2013. doi: 10.1007/s10462-011-9272-4.

[29] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, “Classification and regression trees”, *Wadsworth Int. Gr.*, 1984.

[30] C. Cortes and V. Vapnik, “Support-vector networks”, *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995. doi: 10.1007/BF00994018.

[31] D. H. Wolpert, “Stacked generalization”, *Neural Networks*, vol. 5, no. 2, pp. 241–259, Jan. 1992. doi: 10.1016/S0893-6080(05)80023-1.

[32] L. Breiman, “Random forests”, *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001. doi: 10.1023/A:1010933404324.

[33] J. H. Friedman, “Greedy function approximation: a gradient boosting machine”, *Ann. Stat.*, pp. 1189–1232, 2001.

[34] ——, “Stochastic gradient boosting”, *Comput. Stat. Data Anal.*, vol. 38, no. 4, pp. 367–378, Feb. 2002. doi: 10.1016/S0167-9473(01)00065-2.

[35] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[36] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization”, *Proc. 3rd Int. Conf. Learn. Represent.*, Dec. 2014.

[37] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”, *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[38] A. Shrikumar, P. Greenside, and A. Kundaje, “Learning Important Features Through Propagating Activation Differences”, Apr. 2017.

[39] Z. Che, S. Purushotham, R. Khemani, and Y. Liu, "Distilling Knowledge from Deep Networks with Applications to Healthcare Domain", pp. 1–13,

[40] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why Should I Trust You?": Explaining the Predictions of Any Classifier", *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD '16*, pp. 1135–1144, Feb. 2016. doi: 10.1145/2939672.2939778.

[41] C.-H. Chang, L. Rampasek, and A. Goldenberg, "Dropout Feature Ranking for Deep Learning Models", *arXiv*, Dec. 2017.

[42] D. Hanahan and R. A. Weinberg, "Hallmarks of Cancer: The Next Generation", *Cell*, vol. 144, no. 5, pp. 646–674, Mar. 2011. doi: 10.1016/j.cell.2011.02.013.

[43] K. D. Miller, R. L. Siegel, C. C. Lin, *et al.*, "Cancer treatment and survivorship statistics, 2016", *CA. Cancer J. Clin.*, vol. 66, no. 4, pp. 271–289, Jul. 2016. doi: 10.3322/caac.21349.

[44] B. Al-Lazikani, U. Banerji, and P. Workman, "Combinatorial drug therapy for cancer in the post-genomic era", *Nat. Biotechnol.*, vol. 30, no. 7, pp. 679–692, Jul. 2012. doi: 10.1038/nbt.2284.

[45] M. Vanneman and G. Dranoff, "Combining immunotherapy and targeted therapies in cancer treatment", *Nat. Rev. Cancer*, vol. 12, no. 4, pp. 237–251, Apr. 2012. doi: 10.1038/nrc3237.

[46] L. Brunton, B. A. Chabner, and B. Knollman, *Goodman and Gilman's The Pharmacological Basis of Therapeutics, Twelfth Edition*. McGraw-Hill Education, 2011.

[47] R. Fisher, L. Pusztai, and C. Swanton, "Cancer heterogeneity: implications for targeted therapeutics", *Br. J. Cancer*, vol. 108, no. 3, pp. 479–485, Feb. 2013. doi: 10.1038/bjc.2012.581.

[48] K. A. Hoadley, C. Yau, T. Hinoue, *et al.*, "Cell-of-Origin Patterns Dominate the Molecular Classification of 10,000 Tumors from 33 Types of Cancer", *Cell*, vol. 173, no. 2, 291–304.e6, Apr. 2018. doi: 10.1016/j.cell.2018.03.022.

[49] K. A. Hoadley, C. Yau, D. M. Wolf, *et al.*, "Multiplatform Analysis of 12 Cancer Types Reveals Molecular Classification within and across Tissues of Origin", *Cell*, vol. 158, no. 4, pp. 929–944, Aug. 2014. doi: 10.1016/j.cell.2014.06.049.

[50] G. Housman, S. Byler, S. Heerboth, *et al.*, "Drug Resistance in Cancer: An Overview", *Cancers (Basel).*, vol. 6, no. 3, pp. 1769–1792, Sep. 2014. doi: 10.3390/cancers6031769.

[51] S. M. Swain, J. Baselga, S.-B. Kim, *et al.*, "Pertuzumab, Trastuzumab, and Docetaxel in HER2-Positive Metastatic Breast Cancer", *N. Engl. J. Med.*, vol. 372, no. 8, pp. 724–734, Feb. 2015. doi: 10.1056/NEJMoa1413513.

[52] A. E. D. Van Swearingen, M. B. Siegel, A. M. Deal, *et al.*, "LCCC 1025: a phase II study of everolimus, trastuzumab, and vinorelbine to treat progressive HER2-positive breast cancer brain metastases", *Breast Cancer Res. Treat.*, vol. 171, no. 3, pp. 637–648, Oct. 2018. doi: 10.1007/s10549-018-4852-5.

[53] W. P. Janzen, "Screening Technologies for Small Molecule Discovery: The State of the Art", *Chem. Biol.*, vol. 21, no. 9, pp. 1162–1170, Sep. 2014. doi: 10.1016/j.chembiol.2014.07.015.

[54] A. V. Hill, "The possible effects of the aggregation of the molecules of haemoglobin on its dissociation curves", *J. Physiol.*, vol. 40, no. 0, pp. iv–vii, 1910. doi: 10.1038/nrn2055.

[55] G. S. Sittampalam, N. P. Coussens, K. Brimacombe, *et al.*, Eds., *Assay guidance manual*. Eli Lilly & Company and the National Center for Advancing Translational Sciences, 2004.

[56] G. Y. Di Veroli, C. Fornari, D. Wang, *et al.*, "Combenefit: an interactive platform for the analysis and visualization of drug combinations", *Bioinformatics*, vol. 32, no. 18, pp. 2866–2868, Sep. 2016. doi: 10.1093/bioinformatics/btw230.

[57] C. T. Keith, A. a. Borisy, and B. R. Stockwell, "Multicomponent therapeutics for networked systems", *Nat. Rev. Drug Discov.*, vol. 4, no. 1, pp. 71–78, Jan. 2005. doi: 10.1038/nrd1609.

[58] J. Tang, K. Wennerberg, and T. Aittokallio, "What is synergy? The Saariselkä agreement revisited", *Front. Pharmacol.*, vol. 6, no. September, pp. 1–5, Sep. 2015. doi: 10.3389/fphar.2015.00181.

[59] S. Loewe, "The problem of synergism and antagonism of combined drugs", *Arzneimittelforschung.*, vol. 3, no. 6, pp. 285–90, Jun. 1953.

[60] J. B. Fitzgerald, B. Schoeberl, U. B. Nielsen, and P. K. Sorger, "Systems biology and combination therapy in the quest for clinical efficacy", *Nat. Chem. Biol.*, vol. 2, no. 9, pp. 458–466, Sep. 2006. doi: 10.1038/nchembio817.

[61] C. I. Bliss, "The Toxicity of Poisons Applied Jointly", *Ann. Appl. Biol.*, vol. 26, no. 3, pp. 585–615, Aug. 1939. doi: 10.1111/j.1744-7348.1939.tb06990.x.

[62] D. Baptista, P. G. Ferreira, and M. Rocha, "Deep learning for drug response prediction in cancer", *Brief. Bioinform.*, vol. 22, no. 1, pp. 360–379, Jan. 2021. doi: 10.1093/bib/bbz171.

[63] A. Basu, N. E. Bodycombe, J. H. Cheah, *et al.*, "An interactive resource to identify cancer genetic and lineage dependencies targeted by small molecules", *Cell*, vol. 154, no. 5, pp. 1151–1161, 2013. doi: 10.1016/j.cell.2013.08.003.

[64] C. Huang, R. Mezencev, J. F. McDonald, and F. Vannberg, "Open source machine-learning algorithms for the prediction of optimal cancer drug therapies", *PLoS One*, vol. 12, no. 10, pp. 1–14, 2017. doi: 10.1371/journal.pone.0186906.

[65] M. Gönen and A. A. Margoliny, "Drug susceptibility prediction against a panel of drugs using kernelized Bayesian multitask learning", *Bioinformatics*, vol. 30, no. 17, pp. 556–563, 2014. doi: 10.1093/bioinformatics/btu464.

[66] I. Cortés-Ciriano, G. J. Van Westen, G. Bouvier, *et al.*, "Improved large-scale prediction of growth inhibition patterns using the NCI60 cancer cell line panel", *Bioinformatics*, vol. 32, no. 1, pp. 85–95, 2015. doi: 10.1093/bioinformatics/btv529.

[67] S. Naulaerts, C. C. Dang, and P. J. Ballester, "Precision and recall oncology: combining multiple gene mutations for improved identification of drug-sensitive tumours", *Oncotarget*, vol. 8, no. 57, pp. 97 025–97 040, 2017. doi: 10.18632/oncotarget.20923.

[68] K. M. Gayvert, O. Aly, J. Platt, M. W. Bosenberg, D. F. Stern, and O. Elemento, "A Computational Approach for Identifying Synergistic Drug Combinations", *PLOS Comput. Biol.*, vol. 13, no. 1, K. Chen, Ed., e1005308, Jan. 2017. doi: 10.1371/journal.pcbi.1005308.

[69] G. B. Goh, N. O. Hodas, and A. Vishnu, "Deep learning for computational chemistry", *J. Comput. Chem.*, vol. 38, no. 16, pp. 1291–1307, Jun. 2017. doi: 10.1002/jcc.24764.

[70] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, "Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships", *J. Chem. Inf. Model.*, vol. 55, no. 2, pp. 263–274, Feb. 2015. doi: 10.1021/ci500747n.

[71] E. B. Lenselink, N. ten Dijke, B. Bongers, *et al.*, "Beyond the hype: deep neural networks outperform established methods using a ChEMBL bioactivity benchmark set", *J. Cheminform.*, vol. 9, no. 1, p. 45, Dec. 2017. doi: 10.1186/s13321-017-0232-0.

[72] A. Koutsoukas, K. J. Monaghan, X. Li, and J. Huan, "Deep-learning: investigating deep neural networks hyper-parameters and comparison of performance to shallow methods for modeling bioactivity data", *J. Cheminform.*, vol. 9, no. 1, p. 42, Dec. 2017. doi: 10.1186/s13321-017-0226-y.

[73] A. Mayr, G. Klambauer, T. Unterthiner, and S. Hochreiter, "DeepTox: Toxicity Prediction using Deep Learning", *Front. Environ. Sci.*, vol. 3, Feb. 2016. doi: 10.3389/fenvs.2015.00080.

[74] A. Korotcov, V. Tkachenko, D. P. Russo, and S. Ekins, "Comparison of Deep Learning with Multiple Machine Learning Methods and Metrics Using Diverse Drug Discovery Data Sets", *Mol. Pharm.*, vol. 14, no. 12, pp. 4462–4475, 2017. doi: 10.1021/acs.molpharmaceut.7b00578.

[75] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate", *arXiv*, 2014.

[76] U. A. Meyer, U. M. Zanger, and M. Schwab, "Omics and Drug Response", *Annu. Rev. Pharmacol. Toxicol.*, vol. 53, no. 1, pp. 475–502, 2013. doi: 10.1146/annurev-pharmtox-010510-100502.

[77] A. Y. Ng and M. I. Jordan, "On Discriminative vs. Generative Classifiers: A comparison of logistic regression and naive Bayes", in *Adv. Neural Inf. Process. Syst. 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds., Cambridge, MA: MIT Press, 2002, pp. 841–848.

[78] Y.-C. Lo, S. E. Rensi, W. Torng, and R. B. Altman, "Machine learning in chemoinformatics and drug discovery", *Drug Discov. Today*, vol. 23, no. 8, pp. 1538–1546, Aug. 2018. doi: 10.1016/j.drudis.2018.05.010.

[79] D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, *et al.*, "Convolutional Networks on Graphs for Learning Molecular Fingerprints", *J. Chem. Inf. Model.*, vol. 56, no. 2, pp. 399–411, Sep. 2015.

[80] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal Machine Learning: A Survey and Taxonomy", *arXiv*, May 2017.

[81] R. Salakhutdinov and G. Hinton, "Deep boltzmann machines", in *Artif. Intell. Stat.*, 2009, pp. 448–455.

[82] K. Preuer, R. P. I. Lewis, S. Hochreiter, A. Bender, K. C. Bulusu, and G. Klambauer, "DeepSynergy: predicting anti-cancer drug synergy with Deep Learning", *Bioinformatics*, vol. 34, no. 9, J. Wren, Ed., pp. 1538–1546, May 2018. doi: 10.1093/bioinformatics/btx806.

[83] Y. Le Cun, L. Jackel, B. Boser, *et al.*, "Handwritten digit recognition: applications of neural network chips and automatic learning", *IEEE Commun. Mag.*, vol. 27, no. 11, pp. 41–46, Nov. 1989. doi: 10.1109/35.41400.

[84] T. Ching, D. S. Himmelstein, B. K. Beaulieu-Jones, *et al.*, "Opportunities and obstacles for deep learning in biology and medicine", *J. R. Soc. Interface*, vol. 15, no. 141, p. 20 170 387, Apr. 2018. doi: 10.1098/rsif.2017.0387.

[85] I. Cortés-Ciriano and A. Bender, "KekuleScope: prediction of cancer cell line sensitivity and compound potency using convolutional neural networks trained on compound images", *J. Cheminform.*, vol. 11, no. 1, p. 41, Dec. 2019. doi: 10.1186/s13321-019-0364-5.

[86] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A Critical Review of Recurrent Neural Networks for Sequence Learning", *arXiv*, May 2015.

[87] A. Oskooei, J. Born, M. Manica, V. Subramanian, J. Sáez-Rodríguez, and M. R. Martínez, "PaccMann: Prediction of anticancer compound sensitivity with multi-modal attention-based neural networks", *arXiv*, Nov. 2018.

[88] Y. Bengio, "Learning Deep Architectures for AI", *Found. Trends® Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009. doi: 10.1561/2200000006.

[89] G. E. Hinton, "Reducing the Dimensionality of Data with Neural Networks", *Science (80-.).*, vol. 313, no. 5786, pp. 504–507, Jul. 2006. doi: 10.1126/science.1127647.

[90] M. Q. Ding, L. Chen, G. F. Cooper, J. D. Young, and X. Lu, "Precision Oncology beyond Targeted Therapy: Combining Omics Data with Machine Learning Matches the Majority of Cancer Cells to Effective Therapeutics", *Mol. Cancer Res.*, vol. 16, no. 2, pp. 269–278, Feb. 2018. doi: 10.1158/1541-7786.MCR-17-0378.

[91] M. Li, Y. Wang, R. Zheng, *et al.*, "DeepDSC: A Deep Learning Method to Predict Drug Sensitivity of Cancer Cell Lines", *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, pp. 1–1, 2019. doi: 10.1109/TCBB.2019.2919581.

[92] Y.-C. Chiu, H.-I. H. Chen, T. Zhang, *et al.*, "Predicting drug response of tumors from integrated genomic profiles by deep neural networks", *BMC Med. Genomics*, vol. 12, no. S1, p. 18, Jan. 2019. doi: 10.1186/s12920-018-0460-9.

[93] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes”, *arXiv*, Dec. 2013.

[94] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets”, *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[95] P. Smolensky, “Information processing in dynamical systems: Foundations of harmony theory”, Colorado Univ at Boulder Dept of Computer Science, Tech. Rep., 1986.

[96] G. Chen, A. Tsoi, H. Xu, and W. J. Zheng, “Predict effective drug combination by deep belief network and ontology fingerprints”, *J. Biomed. Inform.*, vol. 85, pp. 149–154, Sep. 2018. doi: 10.1016/j.jbi.2018.07.024.

[97] A. Paszke, S. Gross, S. Chintala, *et al.*, “Automatic differentiation in PyTorch”, 2017.

[98] M. Abadi, A. Agarwal, P. Barham, *et al.*, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems”, *Nat. Neurosci.*, vol. 16, no. 4, pp. 486–492, Mar. 2016.

[99] F. Chollet *et al.*, *Keras*, \url{https://keras.io}, 2015.

[100] B. Ramsundar, P. Eastman, P. Walters, V. Pande, K. Leswing, and Z. Wu, *Deep Learning for the Life Sciences*. O’Reilly Media, 2019.

[101] R. H. Shoemaker, “The NCI60 human tumour cell line anticancer drug screen”, *Nat. Rev.*, vol. 6, no. 10, pp. 813–823, 2006. doi: 10.1038/nrc1951.

[102] A. M. Gholami, H. Hahne, Z. Wu, *et al.*, “Global Proteome Analysis of the NCI-60 Cell Line Panel”, *Cell Rep.*, vol. 4, no. 3, pp. 609–620, Aug. 2013. doi: 10.1016/j.celrep.2013.07.018.

[103] M. Ghandi, F. W. Huang, J. Jané-Valbuena, *et al.*, “Next-generation characterization of the Cancer Cell Line Encyclopedia”, *Nature*, vol. 569, no. 7757, pp. 503–508, May 2019. doi: 10.1038/s41586-019-1186-3.

[104] H. Li, S. Ning, M. Ghandi, *et al.*, “The landscape of cancer cell line metabolism”, *Nat. Med.*, vol. 25, no. 5, pp. 850–860, May 2019. doi: 10.1038/s41591-019-0404-8.

[105] J. Lamb, “The Connectivity Map: Using Gene-Expression Signatures to Connect Small Molecules, Genes, and Disease”, *Science (80-.).*, vol. 313, no. 5795, pp. 1929–1935, 2006. doi: 10.1126/science.1132939.

[106] A. Subramanian, R. Narayan, S. M. Corsello, *et al.*, “A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles”, *Cell*, vol. 171, no. 6, 1437–1452.e17, Nov. 2017. doi: 10.1016/j.cell.2017.10.049.

[107] A. B. Keenan, S. L. Jenkins, K. M. Jagodnik, *et al.*, “The Library of Integrated Network-Based Cellular Signatures NIH Program: System-Level Cataloging of Human Cells Response to Perturbations”, *Cell Syst.*, vol. 6, no. 1, pp. 13–24, 2018. doi: 10.1016/j.cels.2017.11.001.

[108] A. Koleti, R. Terryn, V. Stathias, *et al.*, "Data Portal for the Library of Integrated Network-based Cellular Signatures (LINCS) program: integrated access to diverse large-scale cellular perturbation response data", *Nucleic Acids Res.*, vol. 46, no. D1, pp. D558–D566, Jan. 2018. doi: 10.1093/nar/gkx1063.

[109] L. Litichevskiy, R. Peckner, J. G. Abelin, *et al.*, "A Library of Phosphoproteomic and Chromatin Signatures for Characterizing Cellular Responses to Drug Perturbations", *Cell Syst.*, vol. 6, no. 4, 424–443.e7, Apr. 2018. doi: 10.1016/j.cels.2018.03.012.

[110] J.-P. Gillet, S. Varma, and M. M. Gottesman, "The Clinical Relevance of Cancer Cell Lines", *JNCI J. Natl. Cancer Inst.*, vol. 105, no. 7, pp. 452–458, Apr. 2013. doi: 10.1093/jnci/djt007.

[111] A. Goodspeed, L. M. Heiser, J. W. Gray, and J. C. Costello, "Tumor-Derived Cell Lines as Molecular Models of Cancer Pharmacogenomics", *Mol. Cancer Res.*, vol. 14, no. 1, pp. 3–13, 2016. doi: 10.1158/1541-7786.MCR-15-0189.

[112] H. Gao, J. M. Korn, S. Ferretti, *et al.*, "High-throughput screening using patient-derived tumor xenografts to predict clinical trial drug response", *Nat. Med.*, vol. 21, no. 11, pp. 1318–1325, Nov. 2015. doi: 10.1038/nm.3954.

[113] M. van de Wetering, H. E. Francies, J. M. Francis, *et al.*, "Prospective Derivation of a Living Organoid Biobank of Colorectal Cancer Patients", *Cell*, vol. 161, no. 4, pp. 933–945, May 2015. doi: 10.1016/j.cell.2015.03.053.

[114] R. L. Grossman, A. P. Heath, V. Ferretti, *et al.*, "Toward a Shared Vision for Cancer Genomic Data", *N. Engl. J. Med.*, vol. 375, no. 12, pp. 1109–1112, Sep. 2016. doi: 10.1056/NEJMp1607591.

[115] J. Zhang, J. Baran, A. Cros, *et al.*, "International Cancer Genome Consortium Data Portal–a one-stop shop for cancer genomics data", *Database*, vol. 2011, bar026–bar026, Sep. 2011. doi: 10.1093/database/bar026.

[116] S. Kim, P. A. Thiessen, E. E. Bolton, *et al.*, "PubChem Substance and Compound databases", *Nucleic Acids Res.*, vol. 44, no. D1, pp. D1202–D1213, Jan. 2016. doi: 10.1093/nar/gkv951.

[117] A. Gaulton, L. J. Bellis, A. P. Bento, *et al.*, "ChEMBL: A large-scale bioactivity database for drug discovery", *Nucleic Acids Res.*, vol. 40, no. D1, pp. 1100–1107, 2012. doi: 10.1093/nar/gkr777.

[118] D. S. Wishart, Y. D. Feunang, A. C. Guo, *et al.*, "DrugBank 5.0: a major update to the DrugBank database for 2018", *Nucleic Acids Res.*, vol. 46, no. D1, pp. D1074–D1082, Jan. 2018. doi: 10.1093/nar/gkx1037.

[119] D. Szklarczyk, A. Santos, C. von Mering, L. J. Jensen, P. Bork, and M. Kuhn, "STITCH 5: augmenting protein–chemical interaction networks with tissue and affinity data", *Nucleic Acids Res.*, vol. 44, no. D1, pp. D380–D384, 2016. doi: 10.1093/nar/gkv1277.

[120] S. D. Harding, J. L. Sharman, E. Faccenda, *et al.*, "The IUPHAR/BPS Guide to PHARMACOLOGY in 2018: updates and expansion to encompass the new guide to IMMUNOPHARMACOLOGY", *Nucleic Acids Res.*, vol. 46, no. D1, pp. D1091–D1106, Jan. 2018. doi: 10.1093/nar/gkx1121.

[121] J. G. Tate, S. Bamford, H. C. Jubb, *et al.*, "COSMIC: the Catalogue Of Somatic Mutations In Cancer", *Nucleic Acids Res.*, vol. 47, no. D1, pp. D941–D947, Jan. 2019. doi: 10.1093/nar/gky1015.

[122] G. S. Cowley, B. A. Weir, F. Vazquez, *et al.*, "Parallel genome-scale loss of function screens in 216 cancer cell lines for the identification of context-specific genetic dependencies", *Sci. Data*, vol. 1, p. 140 035, 2014. doi: 10.1038/sdata.2014.35.

[123] A. Tsherniak, F. Vazquez, P. G. Montgomery, *et al.*, "Defining a Cancer Dependency Map", *Cell*, vol. 170, no. 3, 564–576.e16, Jul. 2017. doi: 10.1016/j.cell.2017.06.010.

[124] M. Whirl-Carrillo, E. M. McDonagh, J. M. Hebert, *et al.*, "Pharmacogenomics Knowledge for Personalized Medicine", *Clin. Pharmacol. Ther.*, vol. 92, no. 4, pp. 414–417, Oct. 2012. doi: 10.1038/clpt.2012.96.

[125] F. Eduati, L. M. Mangravite, T. Wang, *et al.*, "Prediction of human population responses to toxic compounds by a collaborative competition", *Nat. Biotechnol.*, vol. 33, no. 9, pp. 933–940, Sep. 2015. doi: 10.1038/nbt.3299.

[126] J. Greshock, K. E. Bachman, Y. Y. Degenhardt, *et al.*, "Molecular Target Class Is Predictive of In vitro Response Profile", *Cancer Res.*, vol. 70, no. 9, pp. 3677–3686, May 2010. doi: 10.1158/0008-5472.CAN-09-3788.

[127] P. M. Haverty, E. Lin, J. Tan, *et al.*, "Reproducible pharmacogenomic profiling of cancer cell line panels", *Nature*, vol. 533, no. 7603, pp. 333–337, 2016. doi: 10.1038/nature17987.

[128] J. P. Mpindi, B. Yadav, P. Östling, *et al.*, "Consistency in drug response profiling", *Nature*, vol. 540, no. 7631, E5–E6, Dec. 2016. doi: 10.1038/nature20171.

[129] C. H. Mermel, S. E. Schumacher, B. Hill, M. L. Meyerson, R. Beroukhim, and G. Getz, "GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers", *Genome Biol.*, vol. 12, no. 4, R41, 2011. doi: 10.1186/gb-2011-12-4-r41.

[130] S. Loewe, "Effect of combinations: mathematical basis of problem", *Arch. Exp. Pathol. Pharmakol.*, vol. 114, pp. 313–326, 1926.

[131] L. Rampášek, D. Hidru, P. Smirnov, B. Haibe-Kains, and A. Goldenberg, "Dr.VAE: improving drug response prediction via modeling of drug perturbation effects", *Bioinformatics*, R. Schwartz, Ed., Mar. 2019. doi: 10.1093/bioinformatics/btz158.

[132] H. Sharifi-Noghabi, O. Zolotareva, C. C. Collins, and M. Ester, "MOLI: multi-omics late integration with deep neural networks for drug response prediction", *Bioinformatics*, vol. 35, no. 14, pp. i501–i509, Jul. 2019. doi: 10.1093/bioinformatics/btz318.

[133] Y. Liu, Y. Zhou, X. Liu, F. Dong, C. Wang, and Z. Wang, "Wasserstein GAN-Based Small-Sample Augmentation for New-Generation Artificial Intelligence: A Case Study of Cancer-Staging Data in Biology", *Engineering*, vol. 5, no. 1, pp. 156–163, Feb. 2019. doi: 10.1016/j.eng.2018.11.018.

[134] K. Matlock, C. De Niz, R. Rahman, S. Ghosh, and R. Pal, "Investigation of model stacking for drug sensitivity prediction", *BMC Bioinformatics*, vol. 19, no. S3, p. 71, Mar. 2018. doi: 10.1186/s12859-018-2060-2.

[135] M. Tan, O. F. Özgül, B. Bardak, I. Ekşioğlu, and S. Sabuncuoğlu, "Drug response prediction by ensemble learning and drug-induced gene expression signatures", *Genomics*, vol. 111, no. 5, pp. 1078–1088, Sep. 2019. doi: 10.1016/j.ygeno.2018.07.002.

[136] Y. Chang, H. Park, H.-J. Yang, *et al.*, "Cancer Drug Response Profile scan (CDRscan): A Deep Learning Model That Predicts Drug Effectiveness from Cancer Genomic Signature", *Sci. Rep.*, vol. 8, no. 1, p. 8857, Dec. 2018. doi: 10.1038/s41598-018-27214-6.

[137] F. Xia, M. Shukla, T. Brettin, *et al.*, "Predicting tumor cell line response to drug pairs with deep learning", *BMC Bioinformatics*, vol. 19, no. S18, p. 486, Dec. 2018. doi: 10.1186/s12859-018-2509-3.

[138] E. J. Bjerrum, "SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules", *arXiv*, Mar. 2017.

[139] J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database", in *2009 IEEE Conf. Comput. Vis. Pattern Recognit.*, IEEE, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[140] T. Qin, L. C. Tsoi, K. J. Sims, X. Lu, and W. Zheng, "Signaling network prediction by the Ontology Fingerprint enhanced Bayesian network", *BMC Syst. Biol.*, vol. 6, no. Suppl 3, S3, 2012. doi: 10.1186/1752-0509-6-S3-S3.

[141] J. M. Pulley, J. P. Rhoads, R. N. Jerome, *et al.*, "Using What We Already Have: Uncovering New Drug Repurposing Strategies in Existing Omics Data", *Annu. Rev. Pharmacol. Toxicol.*, vol. 60, no. 1, annurev–pharmtox–010919–023 537, Jan. 2020. doi: 10.1146/annurev-pharmtox-010919-023537.

[142] A. Aliper, S. Plis, A. Artemov, A. Ulloa, P. Mamoshina, and A. Zhavoronkov, "Deep Learning Applications for Predicting Pharmacological Properties of Drugs and Drug Repurposing Using Transcriptomic Data", *Mol. Pharm.*, vol. 13, no. 7, pp. 2524–2530, Jul. 2016. doi: 10.1021/acs.molpharmaceut.6b00248.

[143] Y. Donner, S. Kazmierczak, and K. Fortney, "Drug Repurposing Using Deep Embeddings of Gene Expression Profiles", *Mol. Pharm.*, vol. 15, no. 10, pp. 4314–4325, Oct. 2018. doi: 10.1021/acs.molpharmaceut.8b00284.

[144] X. Zeng, S. Zhu, X. Liu, Y. Zhou, R. Nussinov, and F. Cheng, "deepDR: a network-based deep learning approach to in silico drug repositioning", *Bioinformatics*, L. Cowen, Ed., May 2019. doi: 10.1093/bioinformatics/btz418.

[145] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: moving beyond fingerprints", *J. Comput. Aided. Mol. Des.*, vol. 30, no. 8, pp. 595–608, Aug. 2016. doi: 10.1007/s10822-016-9938-8.

[146] Z. Wu, B. Ramsundar, E. N. Feinberg, *et al.*, "MoleculeNet: a benchmark for molecular machine learning", *Chem. Sci.*, vol. 9, no. 2, pp. 513–530, 2018. doi: 10.1039/C7SC02664A.

[147] Z. Xu, S. Wang, F. Zhu, and J. Huang, "Seq2seq Fingerprint", in *Proc. 8th ACM Int. Conf. Bioinformatics, Comput. Biol. Heal. Informatics - ACM-BCB '17*, New York, New York, USA: ACM Press, 2017, pp. 285–294. doi: 10.1145/3107411.3107424.

[148] A. Mayr, G. Klambauer, T. Unterthiner, *et al.*, "Large-scale comparison of machine learning methods for drug target prediction on ChEMBL", *Chem. Sci.*, vol. 9, no. 24, pp. 5441–5451, 2018. doi: 10.1039/C8SC00148K.

[149] P. Hop, B. Allgood, and J. Yu, "Geometric Deep Learning Autonomously Learns Chemical Features That Outperform Those Engineered by Domain Experts", *Mol. Pharm.*, vol. 15, no. 10, pp. 4371–4377, Oct. 2018. doi: 10.1021/acs.molpharmaceut.7b01144.

[150] Y. Li, F.-X. Wu, and A. Ngom, "A review on machine learning principles for multi-view biological data integration", *Brief. Bioinform.*, bbw113, Dec. 2016. doi: 10.1093/bib/bbw113.

[151] P. Mamoshina, A. Vieira, E. Putin, and A. Zhavoronkov, "Applications of Deep Learning in Biomedicine", *Mol. Pharm.*, vol. 13, no. 5, pp. 1445–1454, May 2016. doi: 10.1021/acs.molpharmaceut.5b00982.

[152] B. Haibe-Kains, N. El-Hachem, N. J. Birkbak, *et al.*, "Inconsistency in large pharmacogenomic studies", *Nature*, vol. 504, no. 7480, pp. 389–393, 2013. doi: 10.1038/nature12831.

[153] Z. Safikhani, P. Smirnov, M. Freeman, *et al.*, "Revisiting inconsistency in large pharmacogenomic studies", *F1000Research*, vol. 5, p. 2333, Sep. 2016. doi: 10.12688/f1000research.9611.1.

[154] Cancer Cell Line Encyclopedia Consortium and Genomics of Drug Sensitivity in Cancer Consortium, "Pharmacogenomic agreement between two cancer cell line data sets.", *Nature*, vol. 528, no. 7580, pp. 84–7, Dec. 2015. doi: 10.1038/nature15736.

[155] P. Geeleher, E. R. Gamazon, C. Seoighe, N. J. Cox, and R. S. Huang, "Consistency in large pharmacogenomic studies", *Nature*, vol. 540, no. 7631, E1–E2, Dec. 2016. doi: 10.1038/nature19838.

[156] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative Adversarial Nets", in *Adv. Neural Inf. Process. Syst. 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014, pp. 2672–2680.

[157] H. Liu, W. Zhang, B. Zou, J. Wang, Y. Deng, and L. Deng, "DrugCombDB: a comprehensive database of drug combinations toward the discovery of combinatorial therapy", *Nucleic Acids Res.*, Oct. 2019. doi: 10.1093/nar/gkz1007.

[158] E. Strubell, A. Ganesh, and A. McCallum, "Energy and Policy Considerations for Deep Learning in NLP", *arXiv*, Jun. 2019.

[159] R. Schwartz, J. Dodge, N. A. Smith, and O. Etzioni, "Green AI", *arXiv*, Jul. 2019.

[160] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning", in *2018 IEEE 5th Int. Conf. Data Sci. Adv. Anal.*, IEEE, Oct. 2018, pp. 80–89. doi: 10.1109/DSAA.2018.00018.

[161] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences", in *Int. Conf. Mach. Learn.*, PMLR, 2017, pp. 3145–3153.

[162] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic Attribution for Deep Networks", in *Proc. 34th Int. Conf. Mach. Learn. - Vol. 70*, ser. ICML'17, JMLR.org, 2017, pp. 3319–3328.

[163] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions", in *Adv. Neural Inf. Process. Syst. 30*, I. Guyon, U. V. Luxburg, S. Bengio, *et al.*, Eds., Curran Associates, Inc., 2017, pp. 4765–4774.

[164] L. S. Shapley, "A Value for n-Person Games", in *Contrib. to Theory Games (AM-28), Vol. II*, H. W. Kuhn and A. W. Tucker, Eds., Princeton: Princeton University Press, Dec. 1953, pp. 307–318. doi: 10.1515/9781400881970-018.

[165] S. Muggleton, "Inductive logic programming", *New Gener. Comput.*, vol. 8, no. 4, pp. 295–318, Feb. 1991. doi: 10.1007/BF03037089.

[166] R. Evans and E. Grefenstette, "Learning Explanatory Rules from Noisy Data", *J. Artif. Intell. Res.*, vol. 61, pp. 1–64, Jan. 2018. doi: 10.1613/jair.5714.

[167] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead", *Nat. Mach. Intell.*, vol. 1, no. 5, pp. 206–215, May 2019. doi: 10.1038/s42256-019-0048-x.

[168] D. Alvarez Melis and T. Jaakkola, "Towards Robust Interpretability with Self-Explaining Neural Networks", in *Adv. Neural Inf. Process. Syst. 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Curran Associates, Inc., 2018, pp. 7775–7784.

[169] D. Baptista, J. Correia, B. Pereira, and M. Rocha, "A Comparison of Different Compound Representations for Drug Sensitivity Prediction", in *Practical Applications of Computational Biology & Bioinformatics, 15th International Conference (PACBB 2021)*, M. Rocha, F. Fdez-Riverola, M. S. Mohamad, and R. Casado-Vara, Eds., Cham: Springer International Publishing, 2022, pp. 145–154. doi: 10.1007/978-3-030-86258-9_15.

[170] M. Ali and T. Aittokallio, "Machine learning and feature selection for drug response prediction in precision oncology applications", *Biophys. Rev.*, vol. 11, no. 1, pp. 31–39, Feb. 2019. doi: 10.1007/s12551-018-0446-z.

[171] G. Adam, L. Rampášek, Z. Safikhani, P. Smirnov, B. Haibe-Kains, and A. Goldenberg, "Machine learning approaches to drug response prediction: challenges and recent progress", *npj Precis. Oncol.*, vol. 4, no. 1, p. 19, Dec. 2020. doi: 10.1038/s41698-020-0122-1.

[172] A. Cereto-Massagué, M. J. Ojeda, C. Valls, M. Mulero, S. Garcia-Vallvé, and G. Pujadas, "Molecular fingerprint similarity search in virtual screening", *Methods*, vol. 71, pp. 58–63, Jan. 2015. doi: 10.1016/j.ymeth.2014.08.005.

[173] S. Pan, J. Wu, X. Zhu, G. Long, and C. Zhang, "Finding the best not the most: regularized loss minimization subgraph selection for graph classification", *Pattern Recognit.*, vol. 48, no. 11, pp. 3783–3796, Nov. 2015. doi: 10.1016/j.patcog.2015.05.019.

[174] D. Mendez, A. Gaulton, A. P. Bento, *et al.*, "ChEMBL: towards direct deposition of bioassay data", *Nucleic Acids Res.*, vol. 47, no. D1, pp. D930–D940, Jan. 2019. doi: 10.1093/nar/gky1075.

[175] A. P. Bento, A. Hersey, E. Félix, *et al.*, "An open source chemical structure curation pipeline using RDKit", *J. Cheminform.*, vol. 12, no. 1, p. 51, Dec. 2020. doi: 10.1186/s13321-020-00456-1.

[176] D. Rogers and M. Hahn, "Extended-Connectivity Fingerprints", *J. Chem. Inf. Model.*, vol. 50, no. 5, pp. 742–754, May 2010. doi: 10.1021/ci100050t.

[177] H. L. Morgan, "The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service.", *J. Chem. Doc.*, vol. 5, no. 2, pp. 107–113, May 1965. doi: 10.1021/c160017a018.

[178] J. L. Durant, B. A. Leland, D. R. Henry, and J. G. Nourse, "Reoptimization of MDL Keys for Use in Drug Discovery", *J. Chem. Inf. Comput. Sci.*, vol. 42, no. 6, pp. 1273–1280, Nov. 2002. doi: 10.1021/ci010132r.

[179] R. E. Carhart, D. H. Smith, and R. Venkataraghavan, "Atom pairs as molecular features in structure-activity studies: definition and applications", *J. Chem. Inf. Comput. Sci.*, vol. 25, no. 2, pp. 64–73, May 1985. doi: 10.1021/ci00046a002.

[180] G. Landrum *et al.*, "RDKit: Open-source cheminformatics", 2006.

[181] S. Jaeger, S. Fulle, and S. Turk, "Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition", *J. Chem. Inf. Model.*, vol. 58, no. 1, pp. 27–35, Jan. 2018. doi: 10.1021/acs.jcim.7b00616.

[182] Y. Kim, "Convolutional Neural Networks for Sentence Classification", in *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process.*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2014, pp. 1746–1751. doi: 10.3115/v1/D14-1181.

[183] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks", in *5th Int. Conf. Learn. Represent. {ICLR} 2017, Toulon, Fr. April 24-26, 2017, Conf. Track Proc.*, OpenReview.net, 2017.

[184] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks", in *6th Int. Conf. Learn. Represent. {ICLR} 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conf. Track Proc.*, OpenReview.net, 2018.

[185] Z. Xiong, D. Wang, X. Liu, *et al.*, "Pushing the Boundaries of Molecular Representation for Drug Discovery with the Graph Attention Mechanism", *J. Med. Chem.*, vol. 63, no. 16, pp. 8749–8760, Aug. 2020. doi: 10.1021/acs.jmedchem.9b00959.

[186] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, "Scikit-learn: Machine Learning in Python", *J. Mach. Learn. Res.*, vol. 12, no. Oct, pp. 2825–2830, Jan. 2012.

[187] E. C. McLoughlin and N. M. O'Boyle, "Colchicine-Binding Site Inhibitors from Chemistry to Clinic: A Review", *Pharmaceuticals*, vol. 13, no. 1, p. 8, Jan. 2020. doi: 10.3390/ph13010008.

[188] T. L. Nguyen, C. McGrath, A. R. Hermone, *et al.*, "A Common Pharmacophore for a Diverse Set of Colchicine Site Inhibitors Using a Structure-Based Approach", *J. Med. Chem.*, vol. 48, no. 19, pp. 6107–6116, Sep. 2005. doi: 10.1021/jm050502t.

[189] R. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "Gnnexplainer: Generating explanations for graph neural networks", *Adv. Neural Inf. Process. Syst.*, vol. 32, p. 9240, 2019.

[190] H. Döhner, D. J. Weisdorf, and C. D. Bloomfield, "Acute Myeloid Leukemia", *N. Engl. J. Med.*, vol. 373, no. 12, D. L. Longo, Ed., pp. 1136–1152, Sep. 2015. doi: 10.1056/NEJMra1406184.

[191] R. M. Shallis, R. Wang, A. Davidoff, X. Ma, and A. M. Zeidan, "Epidemiology of acute myeloid leukemia: Recent progress and enduring challenges", *Blood Rev.*, vol. 36, pp. 70–87, Jul. 2019. doi: 10.1016/j.blre.2019.04.005.

[192] S. Bertoli, S. Tavitian, A. Huynh, *et al.*, "Improved outcome for AML patients over the years 2000–2014", *Blood Cancer J.*, vol. 7, no. 12, p. 635, Dec. 2017. doi: 10.1038/s41408-017-0011-1.

[193] H. Döhner, E. Estey, D. Grimwade, *et al.*, "Diagnosis and management of AML in adults: 2017 ELN recommendations from an international expert panel", *Blood*, vol. 129, no. 4, pp. 424–447, 2017. doi: 10.1182/blood-2016-08-733196.

[194] R. M. Stone, S. J. Mandrekar, B. L. Sanford, *et al.*, "Midostaurin plus Chemotherapy for Acute Myeloid Leukemia with a FLT3 Mutation", *N. Engl. J. Med.*, vol. 377, no. 5, pp. 454–464, Aug. 2017. doi: 10.1056/NEJMoa1614359.

[195] E. M. Stein, C. D. DiNardo, D. A. Pollyea, *et al.*, "Enasidenib in mutant IDH2 relapsed or refractory acute myeloid leukemia", *Blood*, vol. 130, no. 6, pp. 722–731, Aug. 2017. doi: 10.1182/blood-2017-04-779405.

[196] C. D. DiNardo, E. M. Stein, S. de Botton, *et al.*, "Durable Remissions with Ivosidenib in IDH1 -Mutated Relapsed or Refractory AML", *N. Engl. J. Med.*, vol. 378, no. 25, pp. 2386–2398, Jun. 2018. doi: 10.1056/NEJMoa1716984.

[197] L. K. Schmalbrock, A. Dolnik, S. Cocciardi, *et al.*, "Clonal evolution of acute myeloid leukemia with FLT3 -ITD mutation under treatment with midostaurin", *Blood*, vol. 137, no. 22, pp. 3093–3104, Jun. 2021. doi: 10.1182/blood.2020007626.

[198] The Cancer Genome Atlas Research Network, "Genomic and Epigenomic Landscapes of Adult De Novo Acute Myeloid Leukemia", *N. Engl. J. Med.*, vol. 368, no. 22, pp. 2059–2074, May 2013. doi: 10.1056/NEJMoa1301689.

[199] E. Papaemmanuil, M. Gerstung, L. Bullinger, *et al.*, "Genomic Classification and Prognosis in Acute Myeloid Leukemia", *N. Engl. J. Med.*, vol. 374, no. 23, pp. 2209–2221, Jun. 2016. doi: 10.1056/NEJMoa1516192.

[200] J. W. Tyner, C. E. Tognon, D. Bottomly, *et al.*, "Functional genomic landscape of acute myeloid leukaemia", *Nature*, vol. 562, no. 7728, pp. 526–531, Oct. 2018. doi: 10.1038/s41586-018-0623-z.

[201] CTD-squared BeatAML DREAM Challenge. "CTD-squared BeatAML DREAM Challenge". (2019), [Online]. Available: https://www.synapse.org/%7B%5C#%7D!Synapse:syn20940518/wiki/596265 (visited on 12/09/2021).

[202] M. Manica, A. Oskooei, J. Born, V. Subramanian, J. Sáez-Rodríguez, and M. Rodríguez Martínez, "Toward Explainable Anticancer Compound Sensitivity Prediction via Multimodal Attention-Based Convolutional Encoders", *Mol. Pharm.*, vol. 16, no. 12, acs.molpharmaceut.9b00520, Dec. 2019. doi: 10.1021/acs.molpharmaceut.9b00520.

[203] H. Öztürk, A. Özgür, and E. Ozkirimli, "DeepDTA: deep drug–target binding affinity prediction", *Bioinformatics*, vol. 34, no. 17, pp. i821–i829, Sep. 2018. doi: 10.1093/bioinformatics/bty593.

[204] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support vector regression machines", in *Adv. Neural Inf. Process. Syst.*, 1997, pp. 155–161.

[205] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system", in *Proc. 22nd acm sigkdd Int. Conf. Knowl. Discov. data Min.*, 2016, pp. 785–794.

[206] G. Ke, Q. Meng, T. Finley, *et al.*, "Lightgbm: A highly efficient gradient boosting decision tree", *Adv. Neural Inf. Process. Syst.*, vol. 30, pp. 3146–3154, 2017.

[207] CTD-squared BeatAML DREAM Challenge, "Results", 2019.

[208] S. Kim, J. Chen, T. Cheng, *et al.*, "PubChem in 2021: new data content and improved web interfaces", *Nucleic Acids Res.*, vol. 49, no. D1, pp. D1388–D1395, Jan. 2021. doi: 10.1093/nar/gkaa971.

[209] I. Dagogo-Jack and A. T. Shaw, "Tumour heterogeneity and resistance to cancer therapies", *Nat. Rev. Clin. Oncol.*, vol. 15, no. 2, pp. 81–94, Feb. 2018. doi: 10.1038/nrclinonc.2017.166.

[210] N. Chatterjee and T. G. Bivona, "Polytherapy and Targeted Cancer Drug Resistance", *Trends in Cancer*, vol. 5, no. 3, pp. 170–182, Mar. 2019. doi: 10.1016/j.trecan.2019.02.003.

[211] R. Celebi, O. Bear Don't Walk, R. Movva, S. Alpsoy, and M. Dumontier, "In-silico Prediction of Synergistic Anti-Cancer Drug Combinations Using Multi-omics Data", *Sci. Rep.*, vol. 9, no. 1, p. 8949, Dec. 2019. doi: 10.1038/s41598-019-45236-6.

[212] P. Sidorov, S. Naulaerts, J. Ariey-Bonnet, E. Pasquier, and P. J. Ballester, "Predicting Synergism of Cancer Drug Combinations Using NCI-ALMANAC Data", *Front. Chem.*, vol. 7, Jul. 2019. doi: 10.3389/fchem.2019.00509.

[213] T. Nakano and B. JB, "Prediction of Compound Cytotoxicity Based on Compound Structures and Cell Line Molecular Characteristics", *J. Comput. Aided Chem.*, vol. 21, pp. 1–10, 2020.

[214] B. Zagidullin, Z. Wang, Y. Guan, E. Pitkänen, and J. Tang, "Comparative analysis of molecular fingerprints in prediction of drug combination effects", *Brief. Bioinform.*, Aug. 2021. doi: 10.1093/bib/bbab291.

[215] T. Zhang, L. Zhang, P. R. O. Payne, and F. Li, "Synergistic Drug Combination Prediction by Integrating Multiomics Data in Deep Learning Models", in 2021, pp. 223–238. doi: 10.1007/978-1-0716-0849-4_12.

[216] H. I. Kuru, O. Tastan, and E. Cicek, "MatchMaker: A Deep Learning Framework for Drug Synergy Prediction", *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, pp. 1–1, 2021. doi: 10.1109/TCBB.2021.3086702.

[217] Y. Kim, S. Zheng, J. Tang, W. Jim Zheng, Z. Li, and X. Jiang, "Anticancer drug synergy prediction in understudied tissues using transfer learning", *J. Am. Med. Informatics Assoc.*, vol. 28, no. 1, pp. 42–51, Jan. 2021. doi: 10.1093/jamia/ocaa212.

[218] J. Wang, X. Liu, S. Shen, L. Deng, and H. Liu, "DeepDDS: deep graph neural network with attention mechanism to predict synergistic drug combinations", *Brief. Bioinform.*, Sep. 2021. doi: 10.1093/bib/bbab390.

[219] A. J. Preto, P. Matos-Filipe, J. Mourão, and I. S. Moreira, "SynPred: Prediction of Drug Combination Effects in Cancer using Full-Agreement Synergy Metrics and Deep Learning", *Preprints*, 2021.

[220] H. Zhang, J. Feng, A. Zeng, P. R. Payne, and F. Li, "Predicting Tumor Cell Response to Synergistic Drug Combinations Using a Novel Simplified Deep Learning Model", *bioRxiv*, 2020.

[221] Q. Liu and L. Xie, "TranSynergy: Mechanism-driven interpretable deep neural network for the synergistic prediction and pathway deconvolution of drug combinations", *PLOS Comput. Biol.*, vol. 17, no. 2, A. Schlessinger, Ed., e1008653, Feb. 2021. doi: 10.1371/journal.pcbi.1008653.

[222] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need", in *Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[223] O. Bazgir, S. Ghosh, and R. Pal, "Investigation of REFINED CNN ensemble learning for anti-cancer drug sensitivity prediction", *Bioinformatics*, vol. 37, no. Supplement_1, pp. i42–i50, Aug. 2021. doi: 10.1093/bioinformatics/btab336.

[224] P. Jiang, S. Huang, Z. Fu, Z. Sun, T. M. Lakowski, and P. Hu, "Deep graph embedding for prioritizing synergistic anticancer drug combinations", *Comput. Struct. Biotechnol. J.*, vol. 18, pp. 427–438, 2020. doi: 10.1016/j.csbj.2020.02.006.

[225] Z. Dong, H. Zhang, Y. Chen, and F. Li, "Interpretable Drug Synergy Prediction with Graph Neural Networks for Human-AI Collaboration in Healthcare", May 2021.

[226] J. Yang, Z. Xu, W. K. K. Wu, Q. Chu, and Q. Zhang, "GraphSynergy: a network-inspired deep learning model for anticancer drug combination prediction", *J. Am. Med. Informatics Assoc.*, vol. 28, no. 11, pp. 2336–2345, Oct. 2021. doi: 10.1093/jamia/ocab162.

[227] B. Rozemberczki, A. Gogleva, S. Nilsson, G. Edwards, A. Nikolov, and E. Papa, "MOOMIN: Deep Molecular Omics Network for Anti-Cancer Drug Combination Therapy", Oct. 2021.

[228] J. Ma and A. Motsinger-Reif, "Prediction of synergistic drug combinations using PCA-initialized deep learning", *BioData Min.*, vol. 14, no. 1, p. 46, Dec. 2021. doi: 10.1186/s13040-021-00278-3.

[229] J. N. Weinstein, E. A. Collisson, G. B. Mills, *et al.*, "The Cancer Genome Atlas Pan-Cancer analysis project", *Nat. Genet.*, vol. 45, no. 10, pp. 1113–1120, Oct. 2013. doi: 10.1038/ng.2764.

[230] S. L. Freshour, S. Kiwala, K. C. Cotto, *et al.*, "Integration of the Drug–Gene Interaction Database (DGIdb 4.0) with open crowdsource efforts", *Nucleic Acids Res.*, vol. 49, no. D1, pp. D1144–D1151, Jan. 2021. doi: 10.1093/nar/gkaa1084.

[231] Z. Sondka, S. Bamford, C. G. Cole, S. A. Ward, I. Dunham, and S. A. Forbes, "The COSMIC Cancer Gene Census: describing genetic dysfunction across all human cancers", *Nat. Rev. Cancer*, vol. 18, no. 11, pp. 696–705, Nov. 2018. doi: 10.1038/s41568-018-0060-1.

[232] D. Repana, J. Nulsen, L. Dressler, *et al.*, "The Network of Cancer Genes (NCG): a comprehensive catalogue of known and candidate cancer genes from cancer sequencing screens", *Genome Biol.*, vol. 20, no. 1, p. 1, Dec. 2019. doi: 10.1186/s13059-018-1612-0.

[233] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction", Feb. 2018.

[234] P. Langfelder and S. Horvath, "WGCNA: an R package for weighted correlation network analysis", *BMC Bioinformatics*, vol. 9, no. 1, p. 559, Dec. 2008. doi: 10.1186/1471-2105-9-559.

[235] P. Morris, R. St. Clair, W. E. Hahn, and E. Barenholtz, "Predicting Binding from Screening Assays with Transformer Network Embeddings", *J. Chem. Inf. Model.*, vol. 60, no. 9, pp. 4191–4199, Sep. 2020. doi: 10.1021/acs.jcim.9b01212.

[236] B. L. Staker, K. Hjerrild, M. D. Feese, C. A. Behnke, A. B. Burgin, and L. Stewart, "The mechanism of topoisomerase I poisoning by a camptothecin analog", *Proc. Natl. Acad. Sci.*, vol. 99, no. 24, pp. 15 387–15 392, 2002.

[237] A. E. Wakeling, S. P. Guy, J. R. Woodburn, *et al.*, "ZD1839 (Iressa): an orally active inhibitor of epidermal growth factor signaling with potential for cancer therapy", *Cancer Res.*, vol. 62, no. 20, pp. 5749–5754, 2002.

[238] C.-H. Yun, T. J. Boggon, Y. Li, *et al.*, "Structures of Lung Cancer-Derived EGFR Mutants and Inhibitor Complexes: Mechanism of Activation and Insights into Differential Inhibitor Sensitivity", *Cancer Cell*, vol. 11, no. 3, pp. 217–227, Mar. 2007. doi: 10.1016/j.ccr.2006.12.017.

[239] I. V. Litvinov, E. Netchiporouk, B. Cordeiro, *et al.*, "Ectopic expression of embryonic stem cell and other developmental genes in cutaneous T-cell lymphoma", *Oncoimmunology*, vol. 3, no. 11, e970025, Nov. 2014. doi: 10.4161/21624011.2014.970025.

[240] M. I. Davis, J. P. Hunt, S. Herrgard, *et al.*, "Comprehensive analysis of kinase inhibitor selectivity", *Nat. Biotechnol.*, vol. 29, no. 11, pp. 1046–1051, Nov. 2011. doi: 10.1038/nbt.1990.

[241] L. F. Neville, G. Mathiak, and O. Bagasra, "The immunobiology of interferon-gamma inducible protein 10 kD (IP-10): A novel, pleiotropic member of the C-X-C chemokine superfamily", *Cytokine Growth Factor Rev.*, vol. 8, no. 3, pp. 207–219, Sep. 1997. doi: 10.1016/S1359-6101(97)00015-4.

[242] C. C. Bleul, R. C. Fuhlbrigge, J. M. Casasnovas, A. Aiuti, and T. A. Springer, "A highly efficacious lymphocyte chemoattractant, stromal cell-derived factor 1 (SDF-1)", *J. Exp. Med.*, vol. 184, no. 3, pp. 1101–1109, Sep. 1996. doi: 10.1084/jem.184.3.1101.

[243] The Human Protein Atlas. "Tissue Expression of CXCL10". (2021), [Online]. Available: https://www.proteinatlas.org/ENSG00000169245-CXCL10/tissue (visited on 11/11/2021).

[244] ——, "Tissue expression of CXCL12". (2021), [Online]. Available: https://www.proteinatlas.org/ENSG00000107562-CXCL12/tissue (visited on 11/11/2021).

[245] M. Uhlén, L. Fagerberg, B. M. Hallström, *et al.*, "Tissue-based map of the human proteome", *Science (80-.).*, vol. 347, no. 6220, Jan. 2015. doi: 10.1126/science.1260419.

[246] The Human Protein Atlas. “Tissue expression of TYMP”. (2021), [Online]. Available: https://www.proteinatlas.org/ENSG00000025708-TYMP/tissue (visited on 11/11/2021).

[247] S. S. Diebold, T. Kaisho, H. Hemmi, S. Akira, and C. Reis e Sousa, “Innate Antiviral Responses by Means of TLR7-Mediated Recognition of Single-Stranded RNA”, *Science (80-.).*, vol. 303, no. 5663, pp. 1529–1531, Mar. 2004. doi: 10.1126/science.1093616.

[248] The Human Protein Atlas. “Tissue expression of TLR7”. (2021), [Online]. Available: https://www.proteinatlas.org/ENSG00000196664-TLR7/tissue (visited on 11/11/2021).

[249] M. A. Bowen, D. D. Patel, X. Li, *et al.*, “Cloning, mapping, and characterization of activated leukocyte-cell adhesion molecule (ALCAM), a CD6 ligand.”, *J. Exp. Med.*, vol. 181, no. 6, pp. 2213–2220, Jun. 1995. doi: 10.1084/jem.181.6.2213.

[250] The Human Protein Atlas. “Tissue expression of ALCAM”. (2021), [Online]. Available: https://www.proteinatlas.org/ENSG00000170017-ALCAM/tissue (visited on 11/11/2021).

[251] A. Ohta and M. Sitkovsky, “Extracellular Adenosine-Mediated Modulation of Regulatory T Cells”, *Front. Immunol.*, vol. 5, Jul. 2014. doi: 10.3389/fimmu.2014.00304.

[252] The Human Protein Atlas. “Tissue expression of ADORA2A”. (2021), [Online]. Available: https://www.proteinatlas.org/ENSG00000128271-ADORA2A/tissue (visited on 11/11/2021).

[253] J. A. Engelman, K. Zejnullahu, T. Mitsudomi, *et al.*, “MET Amplification Leads to Gefitinib Resistance in Lung Cancer by Activating ERBB3 Signaling”, *Science (80-.).*, vol. 316, no. 5827, pp. 1039–1043, May 2007. doi: 10.1126/science.1141478.

[254] H. W. Cheung, J. Du, J. S. Boehm, *et al.*, “Amplification of CRKL Induces Transformation and Epidermal Growth Factor Receptor Inhibitor Resistance in Human Non–Small Cell Lung Cancers”, *Cancer Discov.*, vol. 1, no. 7, pp. 608–625, Dec. 2011. doi: 10.1158/2159-8290.CD-11-0046.

[255] B. Zagidullin, J. Aldahdooh, S. Zheng, *et al.*, “DrugComb: an integrative cancer drug combination data portal”, *Nucleic Acids Res.*, vol. 47, no. W1, W43–W51, Jul. 2019. doi: 10.1093/nar/gkz337.

[256] W. W. B. Goh, W. Wang, and L. Wong, “Why Batch Effects Matter in Omics Data, and How to Avoid Them”, *Trends Biotechnol.*, vol. 35, no. 6, pp. 498–507, Jun. 2017. doi: 10.1016/j.tibtech.2017.02.012.

[257] H. Li, T. Li, D. Quang, and Y. Guan, “Network Propagation Predicts Drug Synergy in Cancers”, *Cancer Res.*, canres.0740.2018, Jul. 2018. doi: 10.1158/0008-5472.CAN-18-0740.

[258] Y. Zhu, T. Brettin, F. Xia, *et al.*, “Converting tabular data into images for deep learning with convolutional neural networks”, *Sci. Rep.*, vol. 11, no. 1, p. 11 325, Dec. 2021. doi: 10.1038/s41598-021-90923-y.

[259] H. A. Elmarakeby, J. Hwang, R. Arafeh, *et al.*, “Biologically informed deep neural network for prostate cancer discovery”, *Nature*, vol. 598, no. 7880, pp. 348–352, Oct. 2021. doi: 10.1038/s41586-021-03922-4.

[260] J. Hao, Y. Kim, T.-K. Kim, and M. Kang, "PASNet: pathway-associated sparse deep neural network for prognosis prediction from high-throughput data", *BMC Bioinformatics*, vol. 19, no. 1, p. 510, Dec. 2018. doi: 10.1186/s12859-018-2500-z.

[261] T. Gaudelet, N. Malod-Dognin, J. Sánchez-Valle, V. Pancaldi, A. Valencia, and N. Pržulj, "Unveiling new disease, pathway, and gene associations via multi-scale neural network", *PLoS One*, vol. 15, no. 4, C. V. Cannistraci, Ed., e0231059, Apr. 2020. doi: 10.1371/journal.pone.0231059.

[262] V. N. Rajapakse, A. Luna, M. Yamade, *et al.*, "CellMinerCDB for Integrative Cross-Database Genomics and Pharmacogenomics Analyses of Cancer Cell Lines", *iScience*, vol. 10, pp. 247–264, Dec. 2018. doi: 10.1016/j.isci.2018.11.029.

[263] B. Lyu and A. Haque, "Deep Learning Based Tumor Type Classification Using Gene Expression Data", in *Proc. 2018 ACM Int. Conf. Bioinformatics, Comput. Biol. Heal. Informatics*, New York, NY, USA: ACM, Aug. 2018, pp. 89–96. doi: 10.1145/3233547.3233588.

[264] E. Cerami, J. Gao, U. Dogrusoz, *et al.*, "The cBio Cancer Genomics Portal: An Open Platform for Exploring Multidimensional Cancer Genomics Data: Figure 1.", *Cancer Discov.*, vol. 2, no. 5, pp. 401–404, May 2012. doi: 10.1158/2159-8290.CD-12-0095.

[265] J. Gao, B. A. Aksoy, U. Dogrusoz, *et al.*, "Integrative Analysis of Complex Cancer Genomics and Clinical Profiles Using the cBioPortal", *Sci. Signal.*, vol. 6, no. 269, pl1–pl1, Apr. 2013. doi: 10.1126/scisignal.2004088.

[266] A. Subramanian, P. Tamayo, V. K. Mootha, *et al.*, "Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles", *Proc. Natl. Acad. Sci.*, vol. 102, no. 43, pp. 15 545–15 550, Oct. 2005. doi: 10.1073/pnas.0506580102.

[267] A. Liberzon, A. Subramanian, R. Pinchback, H. Thorvaldsdottir, P. Tamayo, and J. P. Mesirov, "Molecular signatures database (MSigDB) 3.0", *Bioinformatics*, vol. 27, no. 12, pp. 1739–1740, Jun. 2011. doi: 10.1093/bioinformatics/btr260.

[268] B. Jassal, L. Matthews, G. Viteri, *et al.*, "The reactome pathway knowledgebase", *Nucleic Acids Res.*, Nov. 2019. doi: 10.1093/nar/gkz1031.

[269] D. Grattarola and C. Alippi, "Graph Neural Networks in TensorFlow and Keras with Spektral [Application Notes]", *IEEE Comput. Intell. Mag.*, vol. 16, no. 1, pp. 99–106, 2021.

[270] S. Falkner, A. Klein, and F. Hutter, "BOHB: Robust and efficient hyperparameter optimization at scale", in *Int. Conf. Mach. Learn.*, PMLR, 2018, pp. 1437–1446.

[271] J. Močkus, "On Bayesian Methods for Seeking the Extremum", in *Optim. Tech. IFIP Tech. Conf.* Berlin, Heidelberg: Springer Berlin Heidelberg, 1975, pp. 400–404. doi: 10.1007/978-3-662-38527-2_55.

[272] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization", *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6765–6816, 2017.

[273] C. K. I. Williams and M. Seeger, "Using the Nyström Method to Speed up Kernel Machines", in *Proc. 13th Int. Conf. Neural Inf. Process. Syst.*, ser. NIPS'00, Cambridge, MA, USA: MIT Press, 2000, pp. 661–667.

[274] S. Cardoso, D. Baptista, R. Santos, and M. Rocha, "A Review on Metabolomics Data Analysis for Cancer Applications", in *Practical Applications of Computational Biology and Bioinformatics, 12th International Conference*, F. Fdez-Riverola, M. S. Mohamad, M. Rocha, J. F. De Paz, and P. González, Eds., Cham: Springer International Publishing, 2019, pp. 157–165. doi: 10.1007/978-3-319-98702-6_19.

[275] J. Correia, T. Resende, D. Baptista, and M. Rocha, "Artificial Intelligence in Biological Activity Prediction", in *Practical Applications of Computational Biology and Bioinformatics, 13th International Conference*, F. Fdez-Riverola, M. Rocha, M. S. Mohamad, N. Zaki, and J. A. Castellanos-Garzón, Eds., Cham: Springer International Publishing, 2020, pp. 164–172. doi: 10.1007/978-3-030-23873-5_20.

[276] Z. Xiong, M. Jeon, R. J. Allaway, *et al.*, "Crowdsourced identification of multi-target kinase inhibitors for RET- and TAU- based disease: The Multi-Targeting Drug DREAM Challenge", *PLOS Comput. Biol.*, vol. 17, no. 9, E. Wang, Ed., e1009302, Sep. 2021. doi: 10.1371/journal.pcbi.1009302.

A

Supplementary Figures

A.1 Supplementary Figures for Chapter 5

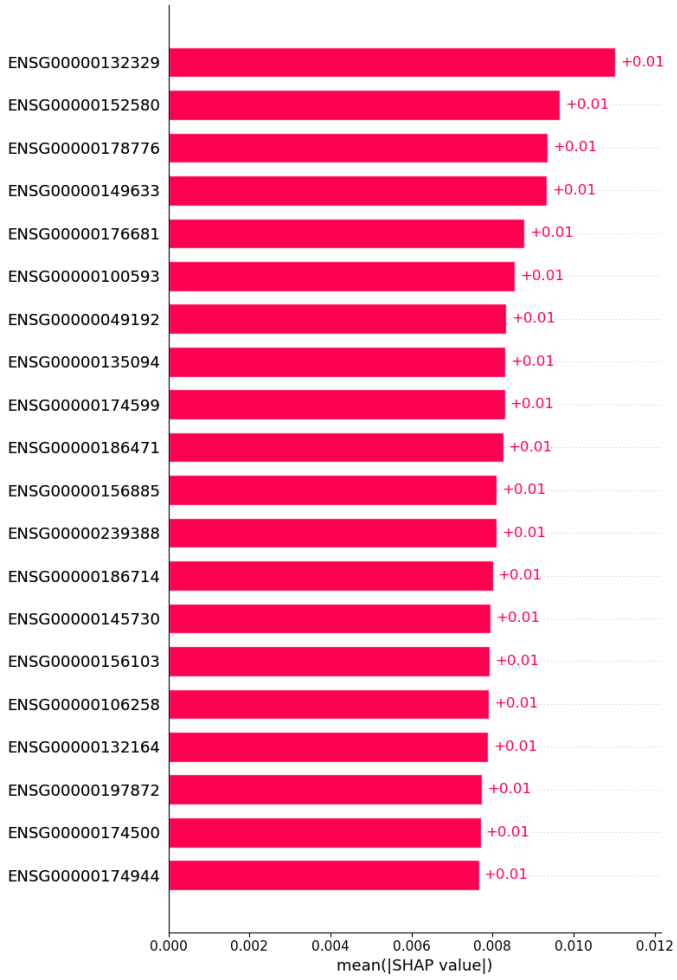


Figure 40: Top 20 most important gene expression features, ranked according to their mean absolute SHAP values.

B

Supplementary Tables

B.1 Supplementary Tables for Chapter 5

Table 13: Average per-inhibitor Spearman correlation scores for models before guaranteeing that the dataset was always split the same way.

Model	Average per-inhibitor Spearman
$expr_{\text{N most variable}}$ + mut + $drug_{\text{ECFP4}}$	0.345
$expr_{\text{N most variable}}$ + mut + $drug_{\text{ECFP6}}$	0.394
$expr_{\text{N most variable}}$ + mut + $drug_{\text{Mol2vec}}$	0.432
$expr_{\text{N most variable}}$ + mut + $drug_{\text{1D CNN}}$	0.379
$expr_{\text{N most variable}}$ + mut + $drug_{\text{TextCNN}}$	0.409
$expr_{\text{N most variable}}$ + mut + $drug_{\text{SmilesLSTM}}$	0.339
$expr_{\text{N most variable}}$ + $drug_{\text{ECFP4}}$	0.043
$expr_{\text{N most variable}}$ + $drug_{\text{ECFP6}}$	0.018
$expr_{\text{N most variable}}$ + $drug_{\text{Mol2vec}}$	0.048
$expr_{\text{N most variable}}$ + $drug_{\text{1D CNN}}$	-0.061
$expr_{\text{N most variable}}$ + $drug_{\text{TextCNN}}$	0.096
$expr_{\text{N most variable}}$ + $drug_{\text{SmilesLSTM}}$	-0.069
$expr_{\text{N most variable}}$ + $clinical_{\text{selected}}$ + $drug_{\text{ECFP4}}$	0.073
$expr_{\text{N most variable}}$ + $clinical_{\text{selected}}$ + $drug_{\text{ECFP6}}$	-0.012
$expr_{\text{N most variable}}$ + $clinical_{\text{selected}}$ + $drug_{\text{Mol2vec}}$	0.091
$expr_{\text{N most variable}}$ + $clinical_{\text{selected}}$ + $drug_{\text{1D CNN}}$	0.15

Table 13 continued from previous page

Model	Average per-inhibitor Spearman
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{TextCNN}$	0.117
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{SmilesLSTM}$	0.087
$expr_{SelectKBest}$ + mut + $drug_{ECFP4}$	0.648
$expr_{SelectKBest}$ + mut + $drug_{ECFP6}$	0.637
$expr_{SelectKBest}$ + mut + $drug_{Mol2vec}$	0.589
$expr_{SelectKBest}$ + mut + $drug_{1D\ CNN}$	0.649
$expr_{SelectKBest}$ + mut + $drug_{TextCNN}$	0.652
$expr_{SelectKBest}$ + mut + $drug_{SmilesLSTM}$	0.649
$expr_{protein\ coding}$ + mut + $drug_{ECFP4}$	0.656
$expr_{protein\ coding}$ + mut + $drug_{ECFP6}$	0.634
$expr_{protein\ coding}$ + mut + $drug_{Mol2vec}$	0.581
$expr_{protein\ coding}$ + mut + $drug_{1D\ CNN}$	0.618
$expr_{protein\ coding}$ + mut + $drug_{TextCNN}$	0.626
$expr_{protein\ coding}$ + mut + $drug_{SmilesLSTM}$	0.63

Table 14: Model performance scores.

Model	Average per-inhibitor Spearman	Spearman	R^2	MSE	Pearson
$expr_{N\ most\ variable}$ + mut + $drug_{ECFP4}$	0.391	0.654	0.393	2215.954	0.650
$expr_{N\ most\ variable}$ + mut + $drug_{ECFP6}$	0.391	0.657	0.405	2170.488	0.651
$expr_{N\ most\ variable}$ + mut + $drug_{Mol2vec}$	0.394	0.719	0.505	1806.690	0.711
$expr_{N\ most\ variable}$ + mut + $drug_{1D\ CNN}$	0.409	0.691	0.442	2037.070	0.680
$expr_{N\ most\ variable}$ + mut + $drug_{TextCNN}$	0.365	0.657	0.419	2122.100	0.653

Table 14 continued from previous page

Model	Average per-inhibitor Spearman	Spearman	R^2	MSE	Pearson
$expr_{N\ most\ variable}$ + mut + $drug_{SmilesLSTM}$	0.446	0.730	0.529	1717.435	0.730
$expr_{N\ most\ variable}$ + $drug_{ECFP4}$	0.055	0.468	0.103	3272.571	0.455
$expr_{N\ most\ variable}$ + $drug_{ECFP6}$	0.058	0.465	0.149	3106.547	0.463
$expr_{N\ most\ variable}$ + $drug_{Mol2vec}$	0.033	0.542	0.204	2903.621	0.484
$expr_{N\ most\ variable}$ + $drug_{1D\ CNN}$	0.027	0.548	0.116	3228.181	0.454
$expr_{N\ most\ variable}$ + $drug_{TextCNN}$	0.073	0.422	0.078	3364.353	0.409
$expr_{N\ most\ variable}$ + $drug_{SmilesLSTM}$	0.016	0.420	-0.030	3760.165	0.392
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{ECFP4}$	0.117	0.494	0.223	2837.084	0.496
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{ECFP6}$	0.092	0.505	0.188	2962.112	0.483
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{Mol2vec}$	0.161	0.655	0.392	2218.659	0.654
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{1D\ CNN}$	0.146	0.575	0.335	2427.007	0.586
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{TextCNN}$	0.165	0.597	0.276	2643.169	0.593

Table 14 continued from previous page

Model	Average per-inhibitor Spearman	Spearman	R^2	MSE	Pearson
$expr_{N\ most\ variable}$ + $clinical_{selected}$ + $drug_{SmilesLSTM}$	0.089	0.585	0.299	2558.891	0.576
$expr_{SelectKBest}$ + mut + $drug_{ECFP4}$	0.662	0.843	0.710	1058.219	0.847
$expr_{SelectKBest}$ + mut + $drug_{ECFP6}$	0.666	0.844	0.705	1076.492	0.847
$expr_{SelectKBest}$ + mut + $drug_{Mol2vec}$	0.590	0.785	0.617	1396.940	0.790
$expr_{SelectKBest}$ + mut + $drug_{1D\ CNN}$	0.616	0.818	-21.149	80844.058	0.154
$expr_{SelectKBest}$ + mut + $drug_{TextCNN}$	0.641	0.831	0.654	1261.425	0.821
$expr_{SelectKBest}$ + mut + $drug_{SmilesLSTM}$	0.484	0.724	0.522	1743.709	0.735
$expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{ECFP4}$	0.666	0.846	0.729	989.854	0.854
$expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{ECFP6}$	0.664	0.842	0.712	1049.855	0.845
$expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{Mol2vec}$	0.570	0.791	0.633	1340.726	0.796
$expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{1D\ CNN}$	0.645	0.829	0.254	2723.491	0.677
$expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{TextCNN}$	0.652	0.829	0.688	1139.902	0.829

Table 14 continued from previous page

Model	Average per-inhibitor Spearman	Spearman	R^2	MSE	Pearson
$expr_{protein\ coding}$ + mut + $clinical_{all}$ + $drug_{SmilesLSTM}$	0.619	0.818	0.645	1294.615	0.812
$expr_{protein\ coding}$ + mut + $drug_{ECFP4}$	0.660	0.841	0.715	1038.640	0.846
$expr_{protein\ coding}$ + mut + $drug_{ECFP6}$	0.640	0.831	0.700	1095.328	0.837
$expr_{protein\ coding}$ + mut + $drug_{Mol2vec}$	0.573	0.790	0.444	2031.084	0.725
$expr_{protein\ coding}$ + mut + $drug_{1D\ CNN}$	0.641	0.829	-0.605	5859.075	0.539
$expr_{protein\ coding}$ + mut + $drug_{TextCNN}$	0.631	0.830	0.688	1140.278	0.831
$expr_{protein\ coding}$ + mut + $drug_{SmilesLSTM}$	0.655	0.834	0.679	1171.389	0.833
$expr_{1D\ CNN,\ protein\ coding}$ + mut + $drug_{ECFP4}$	0.641	0.831	0.695	1113.174	0.837
$expr_{1D\ CNN,\ protein\ coding}$ + mut + $drug_{ECFP6}$	0.648	0.827	0.668	1212.004	0.821
$expr_{1D\ CNN,\ protein\ coding}$ + mut + $drug_{Mol2vec}$	0.553	0.772	0.508	1795.142	0.779
$expr_{1D\ CNN,\ protein\ coding}$ + mut + $drug_{1D\ CNN}$	0.620	0.818	0.637	1326.368	0.810
$expr_{1D\ CNN,\ protein\ coding}$ + mut + $drug_{TextCNN}$	0.574	0.790	0.598	1468.337	0.795
$expr_{1D\ CNN,\ protein\ coding}$ + mut + $drug_{SmilesLSTM}$	0.633	0.822	0.564	1592.348	0.775

B.2 Supplementary Tables for Chapter 6

Table 15: Top 20 features (ranked by Gini importance) for the random forest model.

Feature	Importance
drugA_ECFP4_71	0.0034
drugA_ECFP4_440	0.0027
drugA_ECFP4_589	0.0026
drugA_ECFP4_238	0.0025
drugB_ECFP4_315	0.0024
drugA_ECFP4_867	0.0024
drugA_ECFP4_357	0.0023
drugA_ECFP4_997	0.0023
drugB_ECFP4_250	0.0021
drugA_ECFP4_81	0.0020
drugA_ECFP4_696	0.0020
drugA_ECFP4_34	0.0019
drugA_ECFP4_129	0.0019
drugA_ECFP4_927	0.0019
drugB_ECFP4_559	0.0018
drugA_ECFP4_894	0.0018
drugB_ECFP4_927	0.0018
drugA_ECFP4_850	0.0018
drugB_ECFP4_34	0.0018
drugA_ECFP4_815	0.0017