Evaluating the use of structural and drug-protein feature representations for the predictive modelling of drug toxicology in the Toxric database

by

James Kin Man Ngai

Supervised by

A/Prof Fatemeh Vafaee

A thesis submitted as partial fulfilment

of the requirement for the degree of

Bachelor of Science (Honours)

School of Biotechnology and Biomolecular Sciences

University of New South Wales

October 2024

ORIGINALITY STATEMENT

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Signed: _____

Date: 29/10/24

ABSTRACT

During the drug development pipeline, one-third of all candidate molecules are knocked out due to toxicological concerns. Much of this attrition occurs during the latter stages of the pipeline, incurring large resource and time costs on unfeasible candidates. To alleviate these issues, predictive models can be implemented that can identify a drug's toxicity from their identifying information. These models perform *in silico*, and, by utilising these models early in drug development, researchers can either remove or investigate toxic candidate molecules before the cost-intensive tests that occur throughout the pipeline. With the ever-increasing availability of computational hardware, predictive modelling has become a commonly used technique for molecular property prediction tasks such as toxicological profiling.

This project employed the use of four machine learning methods for the creation of toxicity predicting models: random forests, support vector machines, K-nearest neighbours, and neural networks. Each of these algorithms have seen prominent success with predictive modelling tasks in the context of molecules. However, these methods require large quantities of data, and rich data representation methods to effectively create accurate models.

Toxric is a 2023 database that contains information pertaining to various molecules, and their toxicological profiles. This project extracted 22 datasets from Toxric, each associated with a unique toxicological endpoint such as cardiotoxicity or carcinogenicity. Across these datasets, a total of 17,773 unique molecules were found. These compounds had various structure-based information extracted, such as their hydrophobicity, or functional groups using Morgan fingerprints, ChemoPy and PaDEL descriptors. Along with this, an interaction-based representation was created by linking Toxric compounds to their polypeptide drug targets found in the DrugBank database. Furthermore, a fusion-based representation was created that amalgamated DrugBank and Morgan fingerprinting data. With these representations, predictive models can extract relevant patterns in drugs to identify their toxicity.

By applying these 4 predictive models onto 5 different forms of feature representations in 22 toxicological endpoints, a total of 440 predictive models were created, assessed and validated. This resulted in 10 models above a quality threshold of 0.65 MCC with a peak MCC value of 0.83. The most prominent of these models can then be applied by researchers during drug discovery as a diagnostic tool to identify possibly toxic molecules, without the need for extensive testing.

TABLE OF CONTENTS

1 INTRODUCTION 1

1.1 DRUG DEVELOPMENT 1

1.2 DRUG TOXICITY 2

1.2.1 Mechanisms of toxicity 2

1.2.2 Modern toxicology 4

1.3 MACHINE LEARNING 4

1.3.1 Model creation pipeline 5

1.3.2 Drug databases 5

1.3.3 Feature representation 6

1.3.4 Quality assurance 7

1.4 MACHINE LEARNING MODEL TYPES 7

1.4.1 Neural networks 8

1.4.2 Support vector machine 10

1.4.3 Random forest 11

1.4.4 K-nearest neighbours 12

1.5 PROJECT OUTLINE 13

1.5.1 Hypotheses 13

1.5.2 Aims 13

REPRODUCIBILITY STATEMENT 14

2 MATERIALS AND METHODS 15

2.1 DATASET EXTRACTION 15

2.1.1 Dataset exploration 15

2.2 FEATURE EXTRACTION 16

2.2.1 Molecular fingerprinting 16

2.2.2 ChemoPy Features *17*

2.2.3 PaDEL Descriptor *19*

2.2.4 DrugBank targets *21*

2.2.5 Feature fusion *23*

2.3 MODEL DEVELOPMENT 24

2.3.1 Machine learning hyperparameters *24*

2.3.2 Neural networks *24*

2.4 MODEL VALIDATION 27

2.4.1 Class imbalance *27*

2.4.2 Metric selection *28*

2.4.3 K-fold cross validation *30*

3 RESULTS 32

3.1 TOXRIC DATASETS 32

3.2 FEATURE EXTRACTION 32

3.2.1 Structural representations *32*

3.2.2 Interaction & fusion representations *33*

3.3 MODEL QUALITY THRESHOLD CREATION 34

3.4 CLASS IMBALANCE OBSERVATIONS 35

3.5 MODEL EVALUATION 36

3.6 FEATURE REPRESENTATIONS SUMMARY 37

3.6.1 Morgan Fingerprints analysis *38*

3.6.2 ChemoPy predictors analysis *39*

3.6.3 PaDEL descriptors analysis *40*

3.6.4 DrugBank Polypeptide targets analysis *41*

3.6.5 Early fusion analysis *42*

3.7 IMPLEMENTED MACHINE LEARNING METHODS 43

3.7.1 Support vector machines assessment *44*

3.7.2 Random forests assessment *45*

3.7.3 K-nearest neighbours assessment *46*

3.7.4 Neural network assessment *47*

3.8 TOXICITY ENDPOINT DATASETS 48

3.9 ANALYSIS OF VARIANCE BETWEEN TESTED MODEL ATTRIBUTES 49

4 DISCUSSION 50

4.1 ASSESSED TOXICITY ENDPOINTS 50

4.1.1 Developmental & Reproductive toxicity *50*

4.1.2 Endocrine Disruption *51*

4.1.3 Hepatotoxicity *51*

4.1.4 Cardiotoxicity *52*

4.1.5 Respiratory toxicity *52*

4.1.6 Ames Mutagenicity & Carcinogenicity *53*

4.2 TESTED DATA REPRESENTATIONS 53

4.2.1 Morgan fingerprinting *54*

4.2.2 ChemoPy *54*

4.2.3 PaDEL descriptors *55*

4.2.4 DrugBank polypeptide targets *55*

4.2.5 MorganFP-DrugBank early fusion *56*

4.3 EVALUATED MODEL TYPES 57

4.3.1 Support vector machines *57*

4.3.2 Random forests *58*

4.3.3 K-nearest neighbours *58*

4.3.4 Neural networks *58*

4.4 STATISTICAL TESTS 59

4.5 LIMITATIONS & FUTURE DIRECTIONS 59

4.5.1 Class imbalance 59

4.5.2 Feature representation expansion 60

4.5.3 Package creation 60

5 CONCLUSION 61

6 REFERENCES 62

7 APPENDIX 69

7.1 SUPPLEMENTARY TABLES 69

TABLE OF TABLES

Table 1: Example Toxric dataset entry.....16
Table 2: 'Ames Mutagenicity'-MorganFP dataset snippet.....17
Table 3: ChemoPy descriptors & feature counts.....18
Table 4: Sample ChemoPy dataset snippet.....19
Table 5: Example DrugBank dataset entry.....21
Table 6: Early fusion example dataset entry.....23
Table 7: Structural representation feature counts.....33
Table 8: Interaction and fusion representation feature counts.....33
Table 9: Kruskal-Wallis H test of feature representations and model types.....49

TABLE OF EQUATIONS

Equation 1: Huang first layer node count 25
Equation 2: Huang second layer node count 25
Equation 3: Rectified linear unit equation 25
Equation 4: Sigmoid function 25
Equation 5: Matthew's Correlation Coefficient 28

TABLE OF FIGURES

Figure 1: Drug development overview 2
Figure 2: The many toxicological endpoints of the human body.....3
Figure 3: Predictive model pipeline 5
Figure 4: Perceptron functional diagram.....9
Figure 5: Support vector machine diagram 10
Figure 6: RF decision tree of hepatotoxicity prediction.....11
Figure 7: KNN example diagram 12
Figure 8: PaDEL descriptor settings 20
Figure 9: Example DrugBank dataset creation workflow 22
Figure 10: Gradient descent diagram 26
Figure 11: MCC vs accuracy though experiment 29
Figure 12: K-fold cross validation diagram 30
Figure 13: MCC vs accuracy comparison under ideal conditions 34
Figure 14: Developmental and Reproductive toxicity class balancing.....35
Figure 15: Total model quality distribution chart.....36
Figure 16: Median & maximum MCC of feature representations across models 37
Figure 17: MorganFP model performance heatmap 38
Figure 18: Heatmap of ChemoPy model performance.....39
Figure 19: PaDEL performance heatmap.....40
Figure 20: DrugBank model performance heatmap 41
Figure 21: MorganFP-DrugBank polypeptide target fusion performance heatmap.....42
Figure 22: Median and maximum MCC values for machine learning algorithms.....43
Figure 23: SVM performance heatmap 44
Figure 24: RF performance heatmap.....45
Figure 25: KNN performance heatmap.....46
Figure 26: Neural network performance heatmap.....47
Figure 27: Heatmap of performance across all models.....48

ACKNOWLEDGMENTS

First, I must thank my supervisor Fatemeh Vafaee for the huge amount of support you've given me throughout the year. Every meeting we'd have would reinvigorate me with direction and motivate me towards being the best bioinformatician I can be. Even though it has just been a year, I won't forget the guidance and sage advice you've provided to me. Thanks for letting me explore my own ideas while still reeling me back when I start drawing nonsensical conclusions.

Parvin and Mohammad, thanks for the invaluable advice. Any time I'd be stuck banging against my wall trying to code or decipher some Python package you two would be there to point me in the right direction. Also, thanks for all the advice on feature extraction and molecular representations, without that it would've taken twice as long for me to get anything done.

Bryan, even though you were only here for the tail end of the year, thanks for the chats and for proofreading this thesis. No doubt your head's been hurting from all the grammatical mistakes. Without your aesthetic eye this thesis would have the appeal of a lead brick.

Abby, your energy and genuine advice throughout the year has emboldened me to strive for to be the best writer and presenter I can be. Thanks so much for taking the time to bear with me as I presented the most half-baked PowerPoints, and for providing the earnest feedback I needed to raise my presenting skills up to scratch.

For the rest of Vafaee Lab, Mona, thanks for organizing our meetings even though I rarely replied, Coco, thanks for always being a friendly face I can chat to, John, thanks for all the coffee and talks that always keep me hopeful, Ricky and Tom, thanks for hearing me out when got sick of learning theory. And, thanks to all those in the lab I couldn't fit in here! There's just so many of us and only one acknowledgements page but I couldn't make it here without each and every one of you.

Of course, I have to thank my friends who took time out of their busy schedules to help proofread my code and writing. Sarah Costello, Andrew Lin, Ellie Yang, Alice Gao, Nina Chen and Damjan Stevanoski, without you guys I'd still be unable to tell 'were' from 'where', and my conclusions wouldn't even have a conclusion.

Last, but not least, a formal thanks to my family and friends who have no doubt grown sick of me ranting about how annoying coding is. Without the countless days and nights of fun and levity there is no doubt I would have gone insane over the course of this year.

ABBREVIATIONS

ACToR	aggregated computational toxicity resource
AhR	aryl hydrocarbon-acceptor
AR	androgen receptor
ARE	antioxidant response element
ChEMBL	chemical database by the European molecular biology lab
COX	cyclooxygenase
CPDB	carcinogenicity potency database
DILIRank	drug induced liver injury ranking
EPA	Environmental protection agency
FN	false negative
FNN	feed-forward neural network
HSE	heat shock sequence
IC50	half-maximal inhibitory concentration
InChiKey	international chemical identifier key
IUPAC	international union of pure and applied chemistry
KNN	K-nearest neighbours
LBD	ligand binding domain
LTKB	liver toxicity knowledge base
MCC	Matthew's correlation coefficient
ML	machine learning
MMP	mitochondrial membrane potential
MorganFP	Morgan fingerprinting/fingerprints

NaN	not a value
NN	neural network(s)
NR	nuclear receptor
p53	p53 protein pathway
PPAR	peroxisome profilerator-activated receptor
QSAR	quantitative structure-activity relationship
ReLu	rectified linear unit
RF	random forest
SMILES	simplified molecular line input entry system
SVM	support vector machine(s)
TAID	Toxric identifier number
TN	false negative
Tox21	toxicology in the 21st century
TP	true positive

1 INTRODUCTION

The resources required to develop a single drug for clinical use totals over a billion dollars of funding used across a decade-long timeframe[1]. Over this period 10,000 to 25,000 candidate molecules are analysed, with a third being knocked out due to toxicological concerns such as harm to an organ or bodily process[2]. Much of this occurs in the latter stages of drug discovery[3] which causes large resource sinks into infeasible candidates. This project aims to develop predictive models that can diagnose a candidate molecule's toxicity in the initial stages of drug discovery, allowing researchers to eliminate or test these candidates before the expensive latter stages of drug development.

1.1 Drug development

The drug development pipeline is a multi-stage process that acts to identify, discover, and test candidate molecules for their clinical applications. Throughout this process, the candidate molecules' efficacy is continuously assessed in increasingly resource-intensive methods[2] to ensure an effective product for clinical use.

Initially, in the target validation phase, researchers act to identify possible endpoints for a drug to disrupt or accentuate, such as a gene or pathway implicated with a disease[1]. Large quantities (10,000 to 25,000)[2] of drugs are then tested against such an endpoint during compound screening utilising high throughput screening[1]. This technique assesses drug-target interactions on a large scale with automated biochemical assays, allowing for the analysis of over 100,000 samples per day. The 1%[3] of candidates that demonstrate a desirable interaction with the target in these assays enter lead optimization, where they are modified or combined with other compounds for better efficacy, delivery, excretion, or uptake within the body[4]. Beyond this, these optimised candidates undergo preclinical trials to be tested within animal models such as mice[5], or zebrafish[6] for efficacy and safety within an *in vivo* context. Finally, these compounds are tested under three phases of clinical trials that last a total of 4.5 years[3]. Here, they are investigated on human test groups for efficacy, toxicity and long-term effects[7].

This pipeline totals over a decade, with the brunt of time and resources being spent during the preclinical and clinical testing phases. As a result, it is essential that all drugs that enter the latter stages of drug discovery are safe and efficacious to the identified endpoint.

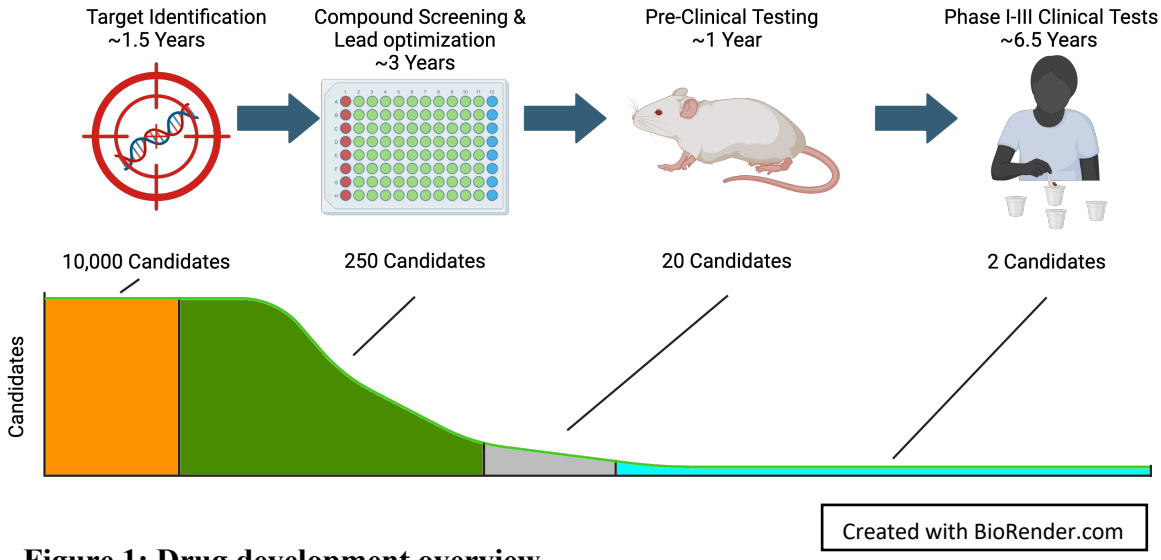

Figure 1: Drug development overview

The bulk of the time and resources spent in drug development is in the clinical trial phases. It is integral to ensure that candidates which enter this stage are free of toxicological concerns.

1.2 Drug toxicity

All drugs are toxic given a high enough dosage, this train of thought has been echoed since the 16th century when Swiss alchemist Paracelsus proposed that, 'The right dose differentiates a poison and a remedy'[8]. This duality reigns true, even for seemingly innocuous medicines such as Ibuprofen, which reduces inflammation in regulated dosages, but, is associated with liver injuries when overdosed[9]. The study of this medicine-poison duality is known as toxicology, with the toxicity targets of a given drug being its toxicological endpoints.

1.2.1 Mechanisms of toxicity

There is a plethora of methods in which drugs can interact with the human body, ranging from the disruption of target genes to the accentuation of biochemical processes[8].

One prominent mechanism of drug toxicity is known as on-target toxicity. This refers to disruptive effects occurring from a drug interacting with its intended target[2]. For instance,

statins are a drug class that lowers cholesterol by inhibiting 3-hydroxy-3-methylglutaryl CoA reductase. This enzyme is also responsible for regulating muscle activity, and its inhibition can cause muscle damage, inherently linking the positive and detrimental effects of statins via on-target toxicity[2]. Contrastingly, off-target toxicity occurs when a drug binds onto an alternate, unintended target, resulting in detrimental effects and hypersensitivity is characterised when drugs induce an unsolicited immune response in individuals[8], also known as an allergic reaction. These forms of toxicity, amongst others, are commonly measured with their half-maximal inhibitory concentration (IC50)[1], which indicates the concentration of a molecule required to limit the function of its target by half.

To categorise the toxicity of drugs, it is common to group drugs through the aspect of the body they are toxic to, also known as their toxicological endpoints. This includes traits like cardiotoxicity or hepatotoxicity, which relates to a drug's disruption of heart or liver functionality respectively.

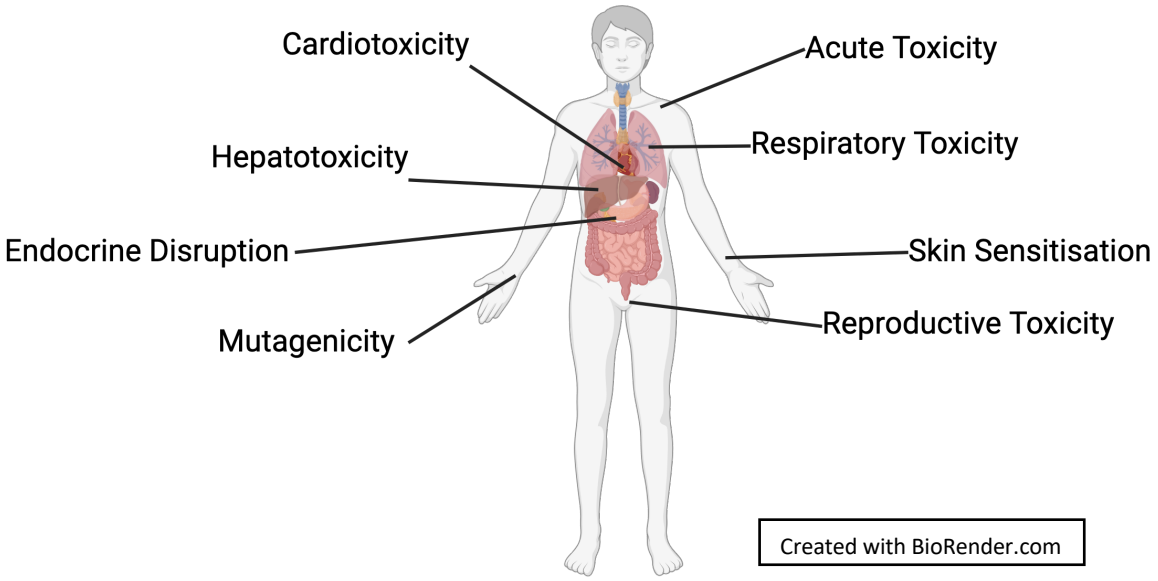

Figure 2: The many toxicological endpoints of the human body

Within the human body toxicity can be categorised by the area or organ it disrupts. Through this characterisation drugs can be linked together via the areas of the body they are detrimental to.

1.2.2 Modern toxicology

Toxicology is integral to the process of drug development, where the toxicological profile of candidate drugs must be thoroughly assessed before being verified for clinical use[3]. Across this pipeline, drug toxicity is continually tested but is particularly emphasised in the preclinical and clinical trial phases[2]. However, ideally, toxicity is to be identified as early in the pipeline as possible, to avoid the expenditure of resources onto human incompatible compounds.

The initial round of toxicological analysis occurs during compound screening. Here, drugs that show a favourable interaction with a validated target are tested against a plethora of possible toxicological endpoints via biochemical assays[1]. In doing this, possible off-target toxicity can be identified, and overtly toxic drugs can be eliminated. Despite this, most candidates that pass this round of testing are still knocked out later in the pipeline due to undiscovered toxicity. This occurs due to the sheer quantity of mechanisms in which drugs can influence the body, which are often too numerous to test for in assay work[8].

Borrowed from the field of computer science, predictive modelling is a method that can be implemented in drug development to predict the toxicological profiles of candidate molecules *in silico*. This technique has been gaining popularity in modern toxicology due to its capacity to diagnose toxic endpoints accurately without the need for experimentation[10]. Amongst other areas, the technology has seen strong success in the prediction of cardiotoxicity[11], mutagenicity[12] and endocrine disruption[13].

1.3 Machine learning

Machine learning (ML) is a form of predictive modelling that can be thought of as algorithms that learn patterns from existing data to predict the outcome of unknown data[14]. The first implementation of this was in 1957 when psychologist Frank Rosenblatt built a perceptron[15]. This ML model acts as a multi-layered apparatus capable of identifying handwritten letters based on principles of human neuron functionality[16]. Since then, ML techniques have seen extensive use in the drug discovery field for predicting the function of a molecule based on its structure through quantitative structure activity relationship (QSAR) analysis[10,14]. As a

subcategory of QSAR, ML models have also managed to consistently link the structure of compounds to their toxicological profile within the human body[17,18].

1.3.1 Model creation pipeline

The creation of a machine learning model involves three primary steps, data curation, feature extraction and model creation. Initially, a dataset containing large quantities of ML-appropriate data must be created. In the context of toxicity prediction, this would entail a selection of molecules and their respective toxicity endpoints. Afterwards, relevant attributes of a molecule can be extracted such as its structural features. This is a process known as feature extraction, and it aids ML models to focus on specific representations of a molecule or sample. Finally, a model type is chosen to be trained with the feature-represented data before its quality is validated. These three core components of ML model creation have a high degree of flexibility in choice, and the synergies between each of these aspects can have a major impact on the quality of the overall model.

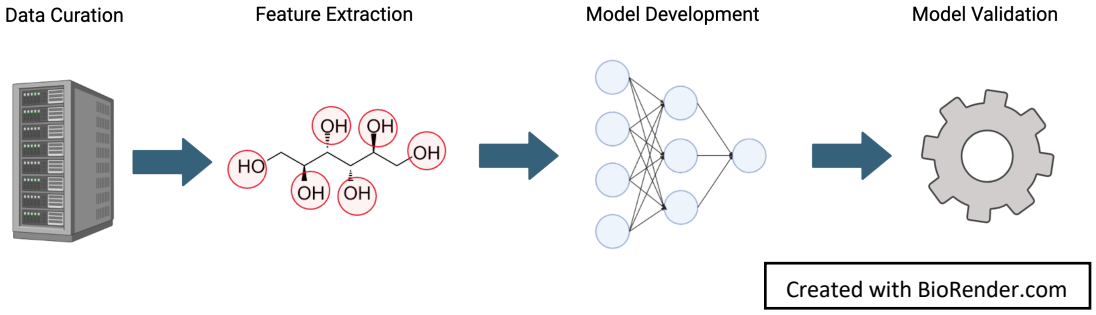

Figure 3: Predictive model pipeline

Model creation proceeds in 3 primary stages: finding data and endpoints, extracting features, then developing the model. Afterwards, it is to be verified with a metric and validation technique. Each choice made during these three stages has a profound impact on model integrity and quality.

1.3.2 Drug databases

The creation of QSAR models through ML principles requires large quantities of data to train and validate[14]. This may come in the form of molecules labelled with the presence or absence

of a trait like cardiotoxicity. A major source of this data comes from databases that collate molecular information from multiple sources (e.g. scientific literature) in a standardised format[19]. Data from multiple sources typically need to be scaled for ML as datapoints of differing magnitudes cannot be directly compared[20].

1.3.2.1 Toxric

Published in 2023, the Toxric[21] database is a repository for compounds and their respective toxicological information. The database categorises 113,372 molecules into 1,474 toxicity endpoints, such as cardiotoxicity and hepatotoxicity, among others. Much of these endpoints are labelled binarily (0,1), for example, the drug mannitol is categorised as not carcinogenic, and is thus labelled as 0 in that dataset, while hydroquinone is carcinogenic and as a result, has a toxicity value of 1 here. Drug data for each dataset in Toxric is sourced either from preexisting literature[22] or other repositories[23]. Toxric data has seen effective use in toxicity prediction tasks[24], particularly its performance in the prediction of endpoints like mutagenicity with an accuracy of 0.9[12].

1.3.3 Feature representation

Compounds can be converted to their 2D or 3D representations[25] and directly fed into ML models for QSAR analysis to great effect[26]. However, it is also common to extract salient features, or characteristics relevant to a molecule's operation such as its functional groups[27] or charge counts. By using features rather than whole molecules, ML models can focus their pattern recognition on certain aspects of a drug. This saves heavily on processing power and reduces the impact of potential noise that may be present in a molecule's whole chemistry[27].

A common form of feature representation for QSAR analysis is molecular fingerprinting[28]. This technique represents a molecule as a series of binary, or one-hot encoded categories that correspond to the presence or absence of certain atomic groups. By doing this, a model can correlate the functionality or toxicity of a molecule to its constituent structures[28]. Furthermore, another feature representation for drugs is their drug-protein and drug-drug interactions[29]. This representation hinges on the concept that drugs with similar targets may have similar functionality. Feature representations such as these have seen strong success in models for QSAR and toxicity prediction[30,31].

1.3.4 Quality assurance

It is integral to ensure that a model can apply its learnt patterns onto unknown data, rather than just predicting already learned, training data. To ensure this quality, two primary concerns must be addressed during model creation: overfitting and class imbalance.

Overfitting is when a model learns patterns from its training data that do not apply to unseen datasets[32]. This is akin to a student who repeatedly studies the same set of practice questions for an exam without learning the lecture content. While they may perform well on the practice test given enough attempts, their performance on the actual exam will be poor. While some degree of overfitting is unavoidable, drastically overfit models are invalid and cannot be applied to unknown data.

To catch this problem, ML methods typically employ a training/test split. Here a dataset is split into two primary subsets, a training and a test set. By training models using the training data and gauging their performance onto the test set the extent of overfitting can be calculated. Models that perform poorly when attempting to predict test set data are likely overfit while models that perform well can be considered robust.

On the other hand, class imbalance occurs when the ratio of prediction categories is overly varied. For example, in the context of binary toxicity prediction, there would two classes (toxic/non-toxic). The larger the sample count disparity is between these classes, the more biased a predictive model will become as they disproportionately learn characteristics on the dominant class. The simplest method of resolving this issue is to remove samples of the greater class until equality is reached, a technique known as undersampling. In doing this, bias in model creation can be minimised.

1.4 Machine learning model types

Given the longevity of ML as a field, a plethora of algorithms and model types have been developed over the years. Each of these models excel in different feature and data sets, so it is integral to compare a variety of models in predictive model development.

1.4.1 Neural networks

Rosenblatt's implementation of the perceptron can be thought of as a primitive version of a neural network, a ML architecture that has seen extensive use in QSAR prediction and drug discovery[14,33,34]. These networks are highly customisable, but conventionally consist of layers containing information-carrying nodes: an input, a hidden, and an output layer[35]. In the case of Rosenblatt's perceptron that identified letters, the input layer treated an input image as a 1D array with binary cells corresponding to the presence of ink[15]. The hidden layer then utilises matrix multiplication to extract patterns from the input layer. Finally, the output layer consists of a singular node that determines the network's prediction of the letter's identity.

Simply feeding an algorithm data is not always sufficient for creating an accurate predictive model[35]. In the case of the perceptron, its learnt patterns are initially randomised but can be changed with internal parameters known as 'weights', which are assigned to each node[36]. Through rerunning the algorithm and assigning different weights to each node, the learnt patterns, and therefore, the output of the model can be altered[35]. By continually rerunning the model, better weights are kept, and inferior ones are discarded in a process known as iteration[15]. Rosenblatt's initial perceptron required these weights to be manually altered, but newer, conventional neural networks have mechanisms to automatically adjust such weights[35].

Furthermore, by adjusting the number of nodes in the hidden layer, the number of patterns the model learns can be manipulated. Here, the node count is a 'hyperparameter'[35], a setting that is not altered by the machine but is instead changed by the operator.

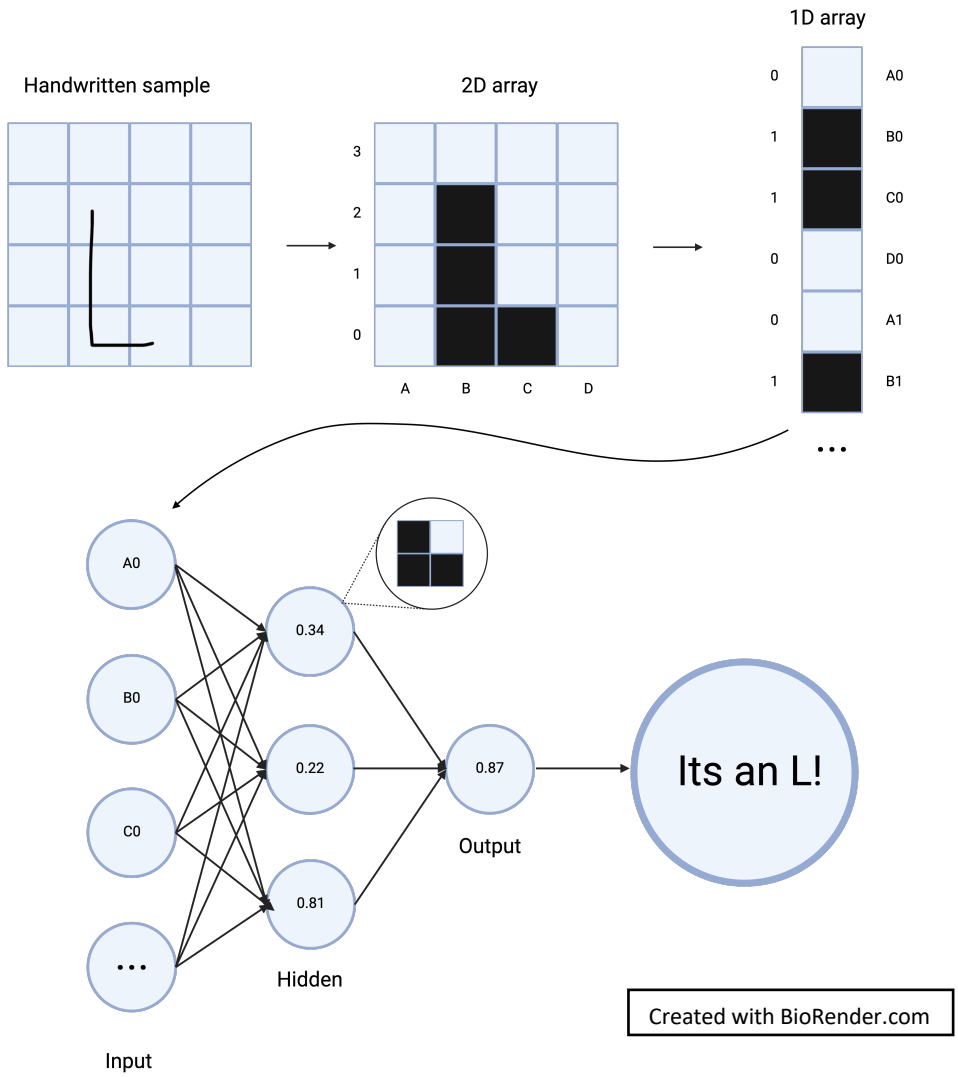

Figure 4: Perceptron functional diagram

First, an input is converted to a 2D array. Here, a 4x4 matrix overlays the handwriting, with cells containing ink being filled. The array is then squashed into 1 dimension, with black cells being labelled '1' and empty cells labelled '0'.

The resulting 1D array can be utilised as an input layer for the neural network, with each node containing the value of a cell. These cells are interconnected to the hidden layer, which extracts patterns from the input. Unlike in the diagram, in practice, most patterns are nonsensical to the human eye. Finally, the hidden layer is further squeezed into an output layer. Based on the output the conclusion of the model can be concluded. E.g. whether the input was an 'L' or not.

1.4.2 Support vector machine

Initially conceived in the mid-20th century, and later developed in 1995 by Vapnik[37,38], support vector machines (SVM) aim to find the hyperplane that best separates classes in multidimensional data. This is done by plotting data onto N dimensions, where N represents the number of feature categories in a dataset, also known as a dataset's dimensionality. The algorithm then finds a plane in N-1 dimensions that best separates classes (i.e. toxic/non-toxic). This method has seen proficient use in differentiating healthy and seizure-induced brain signals[39], as well as in QSAR analysis[40].

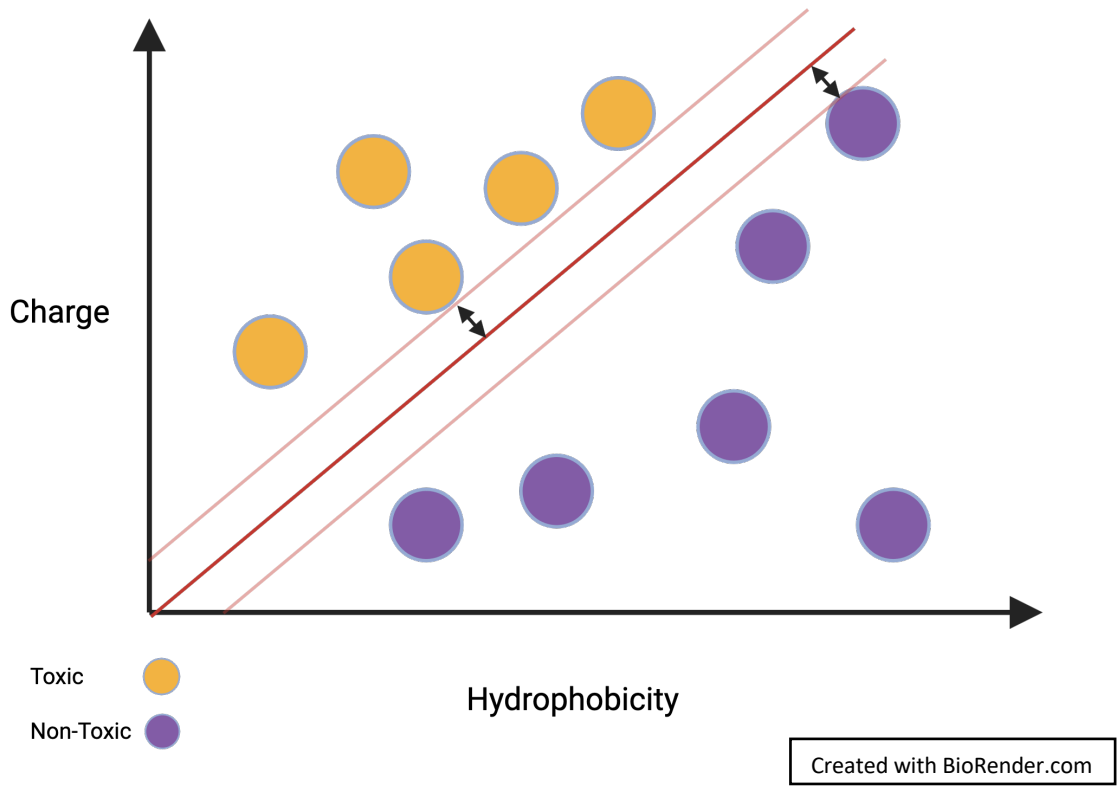

Figure 5: Support vector machine diagram

An example of how SVM functions. In the 2D dimensionality of this dataset each node is a drug with its charge and hydrophobicity represented on the X and Y axes respectively. As N is 2, the plane is in 1 dimensional space and its position is optimised with the titular

support vectors, identified as the closest points either side of the plane. While here the plane is a linear line, differing functions can be used such as multi-degree polynomials.

1.4.3 Random forest

First published in 1995 by T.K. Ho[41], random forest (RF) is a machine learning technique that is generally applicable to most datasets. This method involves the use of multiple decision trees that ask and answer questions about a dataset to conclude its class. The consensus of these trees acts as the final predictor. As a method, RF has seen extensive use in the fields of bioinformatics and chemoinformatics, ranging from applications in postoperative pain prediction[42], QSAR analysis[43] and biomarker detection[44].

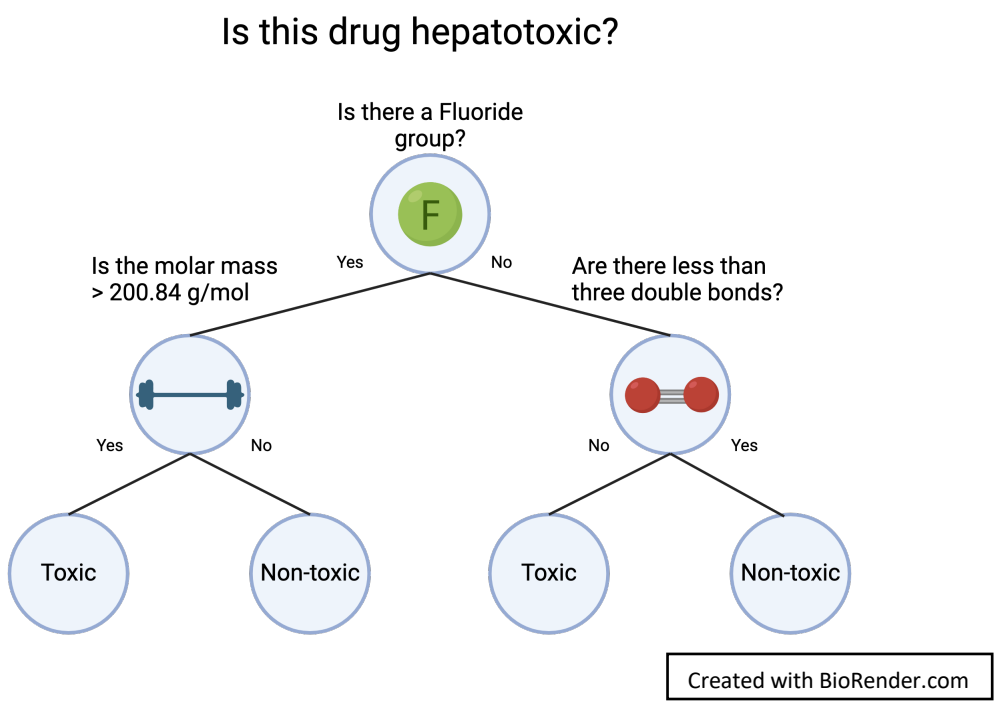

Figure 6: RF decision tree of hepatotoxicity prediction

The constituent decision trees that make up a random forest, act to answer a question about a dataset. Here, the question is whether a drug is hepatotoxic, with the tree consisting of a series of relevant yes/no questions that culminate in a conclusion. Each tree contains randomised questions that are altered with each iteration, and by combining multiple of these trees, a consensus can be reached. The number of features that are sampled at each node can be altered e.g. is there a Fluoride group? AND is there a Nitrogen group?

1.4.4 K-nearest neighbours

K-nearest neighbours (KNN) is a non-iterative machine learning method outlined by Fix and Hodges in 1951[45]. It involves plotting data onto Nth dimensional space and identifying the class of a new datapoint based on the categories of its surrounding, already classified datapoints[46]. KNN is known to be a relatively simple and generally applicable ML method for classification tasks[47]. In particular, it has shown prevalence in QSAR prediction[48], as well as in the prediction of acute toxicity in rats[17].

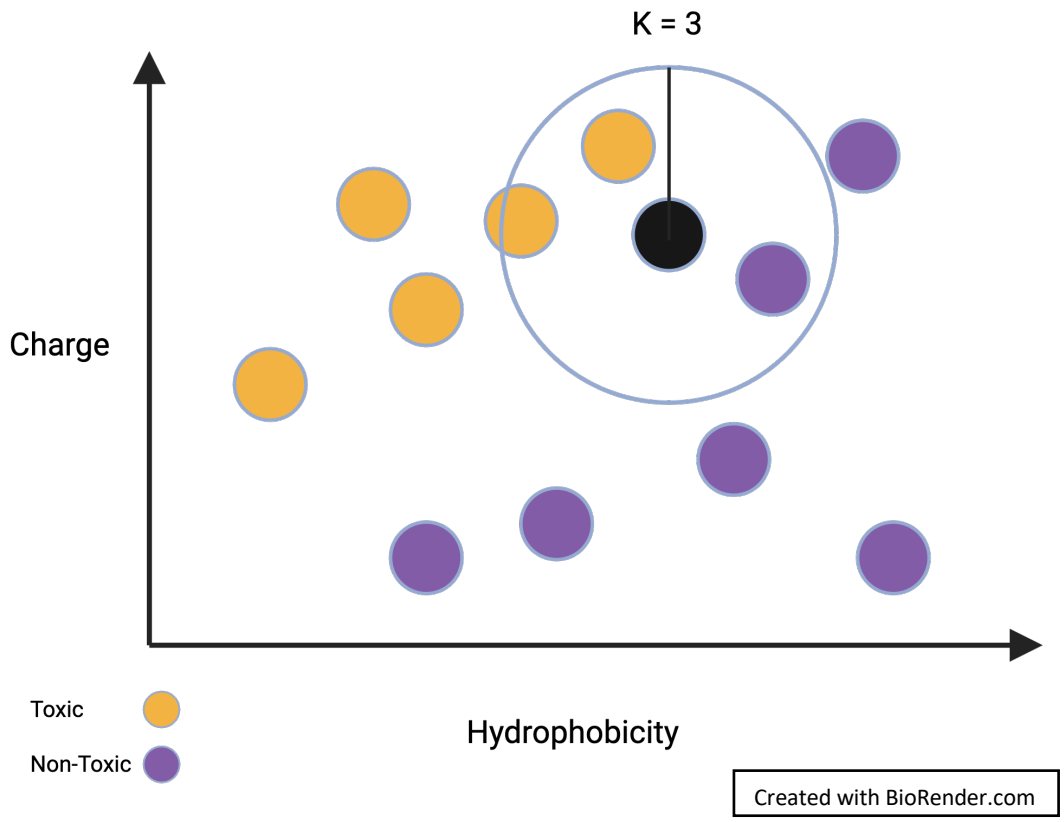

Figure 7: KNN example diagram

An example of how KNN functions. Given a new datapoint (black), a circle of radius 'K' is generated, and based on the identified samples within the circle, the new data is classified. In this case, as there are more toxic than non-toxic nodes inside the radius, the unknown data will be classified as toxic.

1.5 Project outline

This project involves developing a series of predictive models that can draw the toxicity endpoints of a given molecule or set of molecules. Each model corresponds to a toxicity endpoint, a form of feature representation, and a model type. For example, one model may predict the hepatotoxicity of a compound from its Morgan fingerprints using a random forest architecture. As a result, the input for each model would be a feature represented molecule, with the output being a binary prediction of the compound's toxicity to that endpoint.

While predictive modelling has already seen extensive use in toxicological analysis, this project aims to utilise the Toxric database for data curation. In addition to this, a plethora of differing feature representations and model types are tested to uncover significant synergies in the context of toxicity prediction. In doing this, strong predictive models can be created to knock out toxic molecules early in drug development, saving resources that would have been otherwise utilised on a toxic drug candidate.

1.5.1 Hypotheses

Machine learning methods can be effectively used to identify toxicological endpoints within molecules. By varying the used dataset, feature representation method, and machine learning architecture, the quality of developed models will drastically vary, with certain combinations of these variables producing significantly stronger models.

1.5.2 Aims

1. To apply and assess the quality of Toxric datasets and their toxicological endpoints in a machine learning context as information for predictive model creation.
2. To evaluate the effectiveness of different structural and protein-drug interactions as feature representations for drug toxicity prediction.
3. To contrast machine learning model types in the context of toxicological analysis.
4. To find synergies within these components of a predictive model and compare them to those documented in existing literature.

REPRODUCIBILITY STATEMENT

All scripts, code and figures for data preprocessing, feature extraction, model creation and analysis can be found at my Github (https://github.com/jngai23/Honours). All random states or seeds for randomised methods were set to '81'.

2 MATERIALS AND METHODS

2.1 Dataset extraction

22 datasets were taken from the Toxric database's download webpage (found at (https://toxric.bioinforai.tech/download))[21]. This totalled 17,773 unique compounds across 8 endpoints and 22 datasets (Supplementary Table 1).

2.1.1 Dataset exploration

Each dataset initially contained seven pieces of identifying information for each drug:

'TAID' (Toxric's unique identifier number)

'Name' (If applicable)

'IUPAC Name' (International Union of Pure and Applied Chemistry nomenclature)

'PubChemCID (PubChem Chemical Identifier for the PubChem database)

'Canonical SMILES' (Simplified Molecular Line Input Entry System, a computer-readable string notation for molecular structure)

'InChIKey' (International Chemical Identifier Key an encoded string format for structure)

'Toxicity Value' (Binary, indicates toxicity to the dataset's endpoint)

For example, the commonly used diuretic mannitol has the following entry for the 'Carcinogenicity' dataset:

Table 1: Example Toxric dataset entry

Each dataset consisted of multiple of these entries, collated in a comma separated values (.csv) file. Each field is separated with a comma (,) and entries are delimited with a new line (\n).

TAID	Tox-19
Name	Mannitol
IUPAC Name	(2R,3R,4R,5R)-hexane-1,2,3,4,5,6-hexol
PubChemCID	6251
Canonical SMILES	OC[C@@H](O)[C@@H](O)[C@H](O)[C@H](O)CO
InChIKey	FBPFZTCFMRRESA-KVTDHHQDSA-N
Toxicity Value	0

2.2 Feature extraction

To focus model training on certain aspects of drugs such as structural or interaction-based properties, differing methods of feature extraction were utilised. This included three structural representations: Morgan fingerprinting (MorganFP), ChemoPy and PaDEL descriptors along with one interaction-based attribute using drug-polypeptide interactions catalogued by DrugBank. In addition to these, a mixed structure and interaction descriptor was created by combining MorganFP and DrugBank feature sets.

2.2.1 Molecular fingerprinting

Morgan fingerprints are a form of molecular fingerprinting consisting of various one-hot encoded functional groups commonly used in drug property prediction[49,50]. To extract this data, each of the 22 datasets had their respective compounds converted to a 2D structural representation via RDKit's[51] 'AddMoleculeColumnToFrame' tool. Afterwards, the 'GetMorganFingerprintAsBitVect' function can be applied to each molecule to retrieve its

Morgan fingerprint information in the form of 2048 binary categories. This is then recombined with each molecule's SMILES and toxicity values into datasets.

Table 2: 'Ames Mutagenicity'-MorganFP dataset snippet

Each of the 22 Toxric datasets were translated into their respective Morgan fingerprint representations. Below is a snippet of the Ames mutagenicity Morgan fingerprint data. Each of these sets consists of 2050 fields: A molecule's SMILES, followed by their toxicity then the presence or absence of each of the 0-indexed, one-hot encoded fingerprint groups.

SMILES	Toxicity_Value	0	1	2	…	2046	2047
OCC(O)CO	0	0	1	0	…	0	0
O=C1CCO1	0	0	0	0	…	0	0
CCCCCCCC1CCC(=O)O1	0	1	0	0	…	0	0

2.2.2 ChemoPy Features

Released in 2013, ChemoPy[52] is a Python package designed to calculate a variety of chemical properties within a compound. This package has seen success in the development of structure-activity relationship (SAR) models[53] , as well as in predicting flavour retention in beer[54].

While many feature sets are available in ChemoPy, including several forms of molecular fingerprinting, for the purposes of this project, the 2D, as well as chemoinformatic descriptors were taken.

Table 3: ChemoPy descriptors & feature counts

The feature count totals to 201 per ChemoPy dataset. Individual features are binary (0, 1), numeric (1, 2, 3…) or continuous (-0.01, 0, 0.01…).

Descriptor name	No. of features	Description
Constitution	30	Constituent atoms, molecular weight etc.
Connectivity	44	Indices on how constituent atoms are connected
Kappa	7	Describes the shape and topology of molecules
Basak	21	A variety of topological, geometric and quantum indices outlined by Basak, Grut and Grunwald[52].
Electro-topological state (EState)	8	Values based on how the charge of an atom is influenced by other atoms in a compound
Charge	25	Concerns charge of molecule and constituents
Molecular property (molproperty)	6	Various miscellaneous molecular properties such as hydrophobicity
Molecular orbital energy (MOE)	59	Concerns molecular orbits and surface areas

To apply this form of feature representation first a molecule must be represented in its 2D structure. This was done via RDKit's 'Chem.MolFromSmiles()' function which extracts a molecule's 2D representation from its corresponding SMILES. Afterwards the descriptors

were extracted from each molecule using the ChemoPy package. This process was repeated for each Toxric endpoint, resulting in 22 total ChemoPy datasets.

Table 4: Sample ChemoPy dataset snippet

Taken from the 'Ames Mutagenicity' ChemoPy dataset.

Weight: molecular weight nH: number of hydrogens, nHal: number of halogens, VASEstate(7/8/9): van der Waals surface area electrotopological-state, a value based on how forces dependant on an atom's size (van der Waals forces) interact with a molecules EState.

Toxicity_Value	Weight	nH	nHal	...	VSAEstate7	VSAEstate8	VSAEstate9
0	168.060	14	0	...	-1.451636	0	0
0	84.030	8	0	...	-0.729167	0	0
1	222.045	5	0	...	-1.690833	0	0

2.2.3 PaDEL Descriptor

Described in a paper published in 2010, PaDEL Descriptor[55] is a JavaScript program that calculates 1D, 2D, 3D and fingerprint-based structural features for a given molecule. As a feature representation, PaDEL has shown proficiency in the prediction of antimalarial properties in molecules[56], as well as in the prediction of a molecule's acid dissociation constant[33]. Through its applications in the prediction of various chemical properties, it can be reasoned that PaDEL Descriptor is a suitable feature extraction tool for the prediction of drug properties such as toxicity.

Each individual SMILES string was converted into a text file and input into PaDEL's JavaScript program with the following settings:

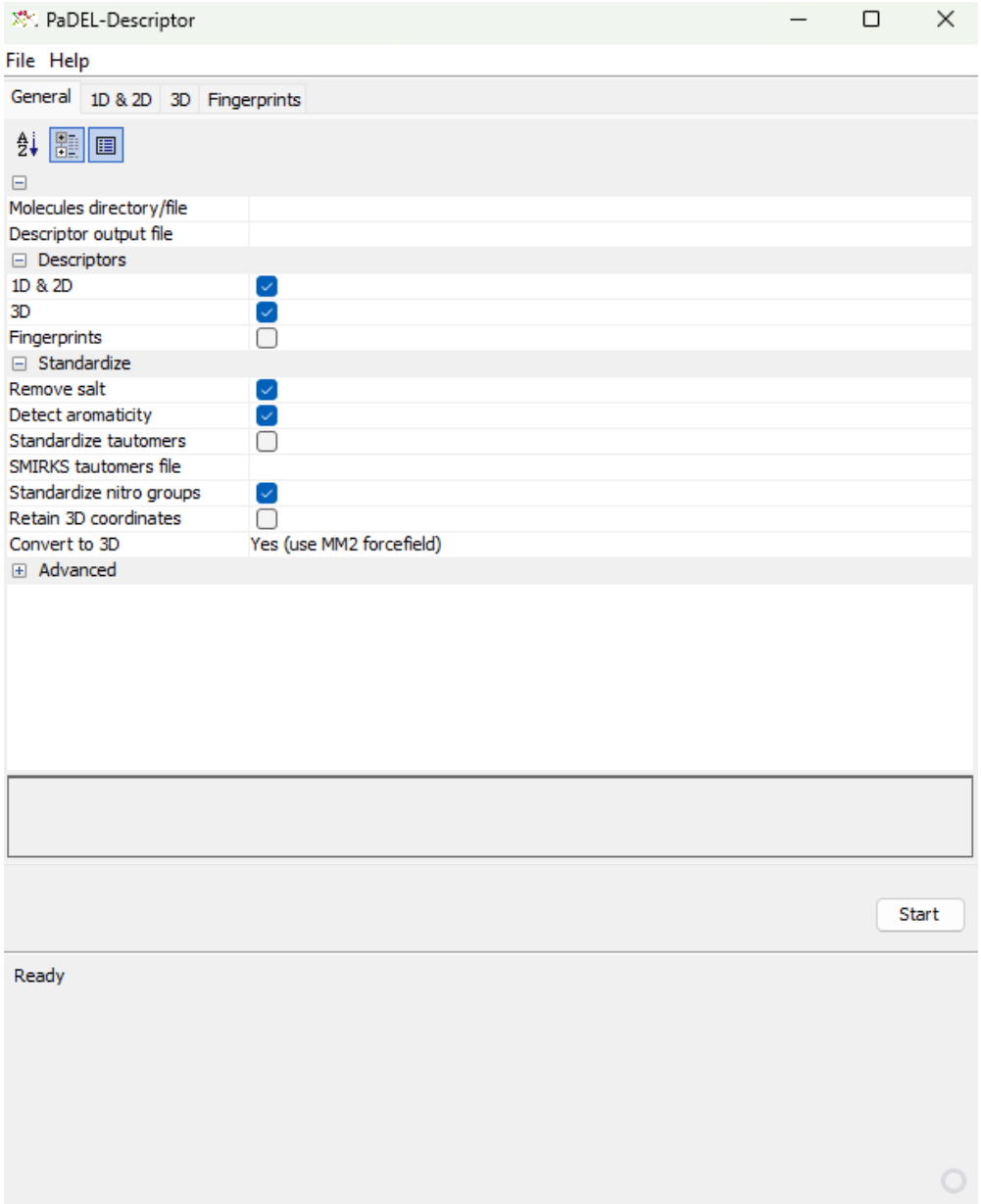

Figure 8: PaDEL descriptor settings

Only 1D, 2D and 3D descriptors were extracted. MM2 forcefield refers to a method of molecule 2D to 3D structure conversion that is specialised for small organic molecules[57]. All other settings were set to their defaults.

The resulting output was a file containing all 17,773 unique SMILES from the Toxric datasets and their 1,876 PaDEL features. All NaN values were then set to '0' to prevent errors from occurring in the model development phase. They were then reassigned to their respective endpoints for prediction.

2.2.4 DrugBank targets

To gauge the effectiveness of drug-protein interactions as a feature representation in this project, known drug targets were extracted from the DrugBank database. This resource has seen success in the creation of predictive models for drug-drug interaction prediction[30], making it suitable for procuring valid drug-target information. As DrugBank's API is not publicly available, instead a dataset concerning polypeptide-drug interactions were downloaded from (https://go.drugbank.com/releases/latest#full). This dataset contained 5,450 polypeptide targets, each linked to one or more of 7,485 unique molecules.

Table 5: Example DrugBank dataset entry

Below is the first entry in the DrugBank Polypeptide targets dataset.

ID: index in dataset, PDB: protein database, HGNC: Hugo gene nomenclature committee,

ID	1
Name	Peptidoglycan synthase FtsI
Gene Name	ftsI
GenBank Protein ID	1574687
GenBank Gene ID	L42023
UniProt ID	P45059
Uniprot Title	FTSI_HAEIN
PDB ID	
GeneCard ID	
GenAtlas ID	
HGNC ID	
Species	Haemophilus influenzae (strain ATCC 51907 / DSM 11121 / KW20 / Rd)
Drug IDs	DB00303

As drugs are encoded as DrugBank IDs which are not present in Toxric data, these IDs must be translated into a format used in the procured datasets. By utilising the UniChem API[58] DrugBank IDs can be converted to InChIKeys which are an identifier found in Toxric data. Afterwards, a dataset can be created containing molecules as well as their toxicity and known targets.

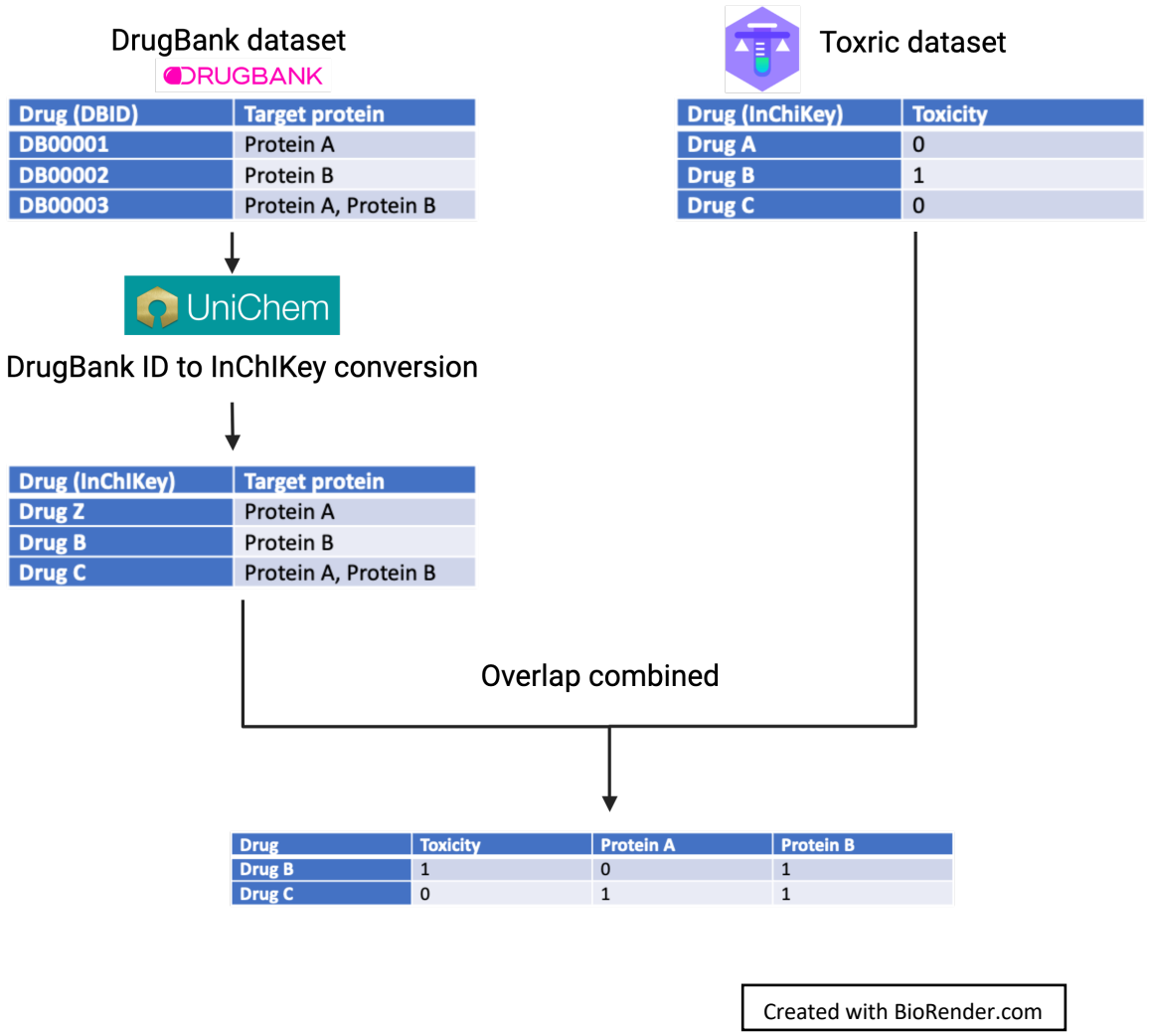

Figure 9: Example DrugBank dataset creation workflow

Initially, DrugBank IDs are converted to InChIKeys for compatability with Toxric data via the UniChem API. Afterwards, the drug overlap between the Toxric and DrugBank data is kept (Drug B, Drug C) with molecules absent in either dataset being removed (Drug A, Drug Z). Toxicity values and drug targets are kept and one-hot encoded.

2.2.5 Feature fusion

Feature extraction aims to focus a ML technique onto a specific aspect of data such as a compound's structure. By merging multiple differing and robust features a more comprehensive representation of a molecule can be created. Combining features prior to prediction is a process known as early fusion and is commonly in predictive modelling to create rich feature sets while minimising the introduction of noise[59]. In the context of this project, the MorganFP and DrugBank polypeptide targets methods were merged into an amalgam dataset after extraction but before prediction. Due to the data loss in DrugBank this, the fusion dataset also only contains compounds found in the overlap between the Toxric and DrugBank datasets.

Table 6: Early fusion example dataset entry

In each fusion dataset a molecule's Morgan fingerprints are listed followed by their DrugBank polypeptide targets, both one-hot encoded. While a relatively simple merge, the combination of two differing forms of feature representation may allow for more salient patterns to be drawn from data.

InChIKey	CVBMAZKKCSYWQR-WCGOZPBSSA-N
Toxicity_Value	0
MorganFP 0	0
MorganFP 1	1
...	...
MorganFP 2047	1
DrugBank 0	1
DrugBank 1	0
...	...
DrugBank 5448	0
DrugBank 5449	0

2.3 Model development

While a multitude of machine learning model types are available for this binary classification task, four primary model types were chosen to encompass a variety of machine learning methodologies. This included: support vector machines (SVM), random forests (RF), K-nearest neighbours (KNN), neural networks (NN). Each of these model types were applied to each feature representation-toxicity endpoint combination, resulting in 4 model types by 5 feature representations by 22 datasets for 440 final models.

SVM, RF and KNN models were created through the use of scikit-learn's various packages[60], Neural networks were generated via Keras' framework[61].

2.3.1 Machine learning hyperparameters

For the conventional ML algorithms of SVM, RF and KNN the non-default hyperparameters were as follows:

SVM hyperparameters

kernel: 'poly'

Degree: 1-4

Coef0: 0.04

RF hyperparameters

n_estimators: 1-50

criterion: 'entropy'

KNN hyperparameters

n_neighbors: 1-100

weights: 'distance'

leaf_size: '100'

2.3.2 Neural networks

Neural networks are a broad term that encompasses many different techniques for machine learning. For this project, a feedforward neural network (FNN) architecture was utilised, which

involves passing information unidirectionally through layers until the output layer is reached. This architecture is structured similarly to the perceptron, and, despite its simplicity, FNN architecture has still seen considerable success in QSAR prediction[34,62].

In this project, for all models two hidden layers were utilised as outlined by Huang[63,64] with the node count per layer found with the equations of:

$$\sqrt{(m+2)N} + 2\sqrt{N/(m+2)}$$

Equation 1: Huang first layer node count

$$m\sqrt{N/(m+2)}$$

Equation 2: Huang second layer node count

In the Huang layer equations 'm' represents the number of output neurons, 'N' refers to the number of samples, or, in this case, the number of input nodes.

These hidden layer nodes have their values represented with a rectified linear unit (ReLu) function which represents data in a non-negative format. This was followed by a sigmoid function-defined output layer to compress the output into a float ranging from 0 to 1 with higher values indicating a higher chance of toxicity. The threshold to determine toxicity is then adaptively chosen best on what values have the best performance for the test set. As a result, each FNN model consists of N input layers, 2 hidden layers, and 1 output layer.

$$R(z) = max(0, z)$$

Equation 3: Rectified linear unit equation

The ReLu function takes the face value of a number unless it is less than 0, in those cases it is set to 0. E.g. for a value of 4, the ReLu output is 4 whereas a value of -4 will output 0.

$$S(x) = \frac{1}{1 + e^{-x}}$$

Equation 4: Sigmoid function

A sigmoid function is a type of logistic function that outputs values between 0 and 1. Its key feature that makes it valuable in machine learning contexts is its ability to transform any real number into a float within this range. This is particularly useful for output layers

as the range can be used to differentiate classes. E.g. a value less than 0.5 could be classed as toxic.

2.3.2.1 Gradient descent

Unlike the perceptron, this FNN is an iterative model that features a learning process known as gradient descent to automatically optimise its internal weights[35]. This technique involves characterising each training cycle as an 'epoch'. In the first epoch, random weights are initialised within the model and the quality of the iteration is measured by the amount of information 'loss' present, calculated via a loss function[35]. In the second epoch, the weights are randomly shifted up or down depending on the 'learning rate', and the model is retrained. If the changes decreased the amount of loss present, they are more likely to be kept. In the third epoch and onward this process is repeated until the change in loss is minimal between iterations.[35]

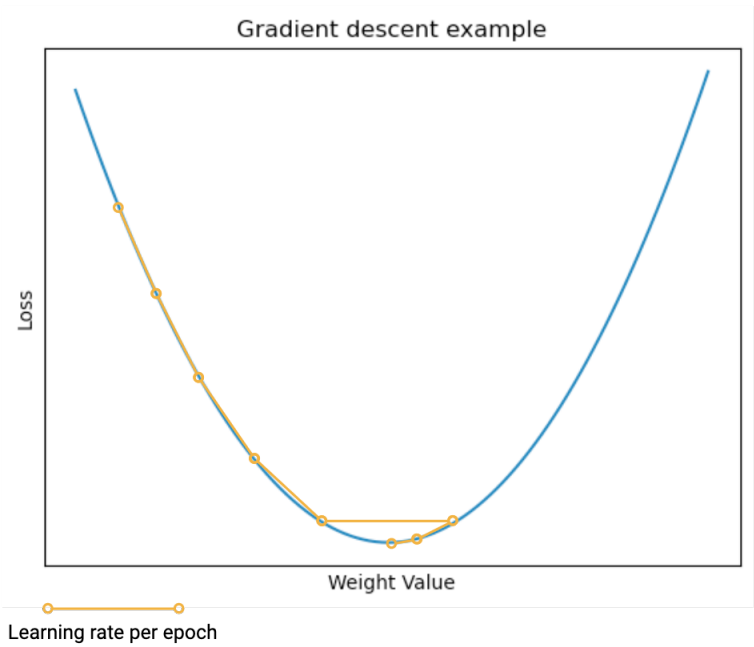

Figure 10: Gradient descent diagram

In each epoch the gradient descent algorithm attempts to decrease loss. However, as loss approaches its minimum, the learning rate may cause weight values to overshoot their optimum. When this occurs, the adaptive learning rate allows the model to readjust its values to best fit the dataset and provide minimum information loss.

2.3.2.2 Neural network hyperparameters

NN hyperparameters

Optimizer algorithm: 'Adam'

Learning rate: 0.0001

Epoch count: 500

Batch size: 16

Layer count: 4

Loss function: 'binary_crossentropy'

Input layer node: Varies (input of sample)

Hidden layer node function: Rectified linear unit (ReLu)

Output layer node function: Sigmoid

Adam is a gradient descent algorithm commonly used for its efficiency and low memory requirements[65]. Batch size refers to the number of samples that are fed into and analysed by a model per cycle. A larger batch size reduces run time but makes the model more prone to generalisations.

2.4 Model validation

Quality assurance is a vital step in the context of predictive model development to ensure that models are robust and valid in their predictions. This involves ensuring that models are trained on high quality data and gauged with a metric that is representative of their effectiveness. Furthermore, the patterns learned by a model must be applicable to unseen molecules for practical application in real world toxicology prediction.

2.4.1 Class imbalance

In this project, datasets with a class ratio exceeding 1:4 are considered imbalanced. In datasets where this is the case, prior to model training, each dataset has samples of the dominant class randomly removed until at least a 5:6 ratio is reached. As a result, the impact of class imbalance is minimised while still preserving a large amount of information for a model to conclude from.

2.4.2 Metric selection

As there is a class imbalance in even the most even datasets in this project, accuracy is not a valid metric[66]. In its place, Matthew's correlation coefficient (MCC) will be utilised. This metric is commonly used in binary classification tasks[66,67] and considers a prediction's: true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). These values constitute the 'confusion matrix' with the true positive and negative values indicating correct predictions whereas the false positive and negatives track incorrect predictions. The values that make up the confusion matrix effectively encompass all aspects of prediction, and, by reviewing all these areas, MCC is considered a valid metric.

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

Equation 5: Matthew's correlation coefficient

MCC is a measure of correlation between 2 sets. In the case of a binary classification task, it measures how robust a model's predictions are compared to a known set of test values. By using the whole confusion matrix (TP, TN, FP, FN), it functions as a metric independent of imbalanced data.

MCC typically ranges as a float value from -1 to 1, but for this project, any values less than 0 will be treated as 0. For cases where any of a model's TN, TP, FN or FP values equal 0, MCC will provide a null value, as a result, in these cases the model's MCC will be set to 0.

2.4.2.1 MCC threshold creation

Conventionally in the context of QSAR and toxicity prediction, a quality model will have an accuracy of 0.8 or higher[11,68] whereas models with an accuracy less than 0.6 can be considered functionally guessing. However, accuracy and MCC are not directly relatable as MCC considers each individual portion of the confusion matrix, however accuracy only looks at correct predictions divided by total predictions. By conducting a thought experiment where true positives and true negatives are considered the same, and false positive and false negatives are equal MCC, and accuracy can be compared.

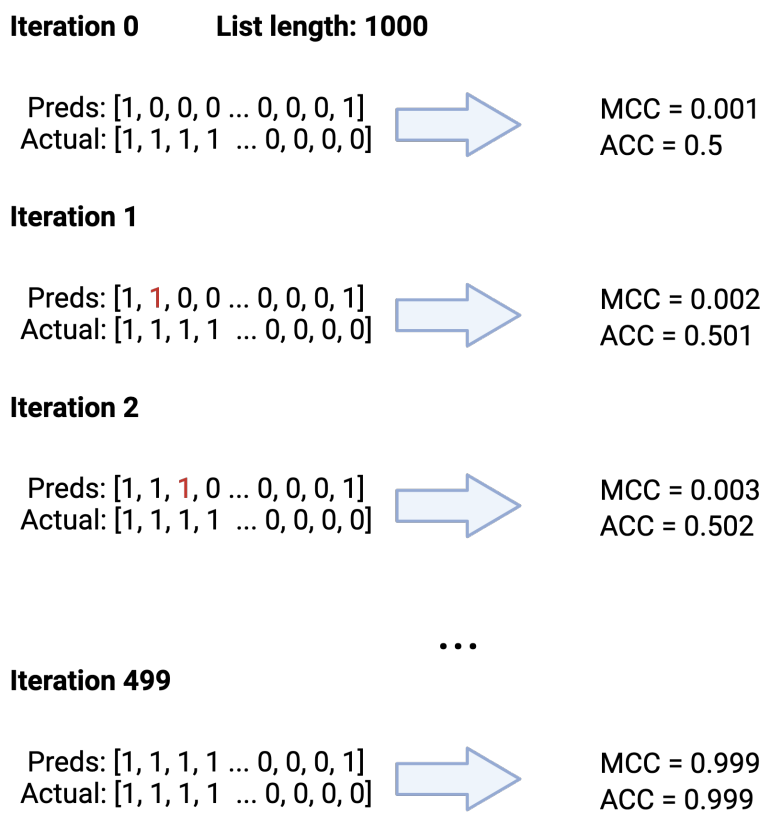

Figure 11: MCC vs accuracy though experiment

This thought experiment is a binary classification task consisting of 1000 samples. Here, the actual values of the samples are 500 '1' values followed by 500 '0' values. Initially, the predicted values of the samples consist of one '1' value, followed by 998 '0' values and tailed with another '1' value. When the predicted and actual sets are compared, the MCC value between the two comes out as 0.001 with an accuracy of 0.5. Next, in the first iteration, one of the incorrect values in the prediction set is corrected. This increases the MCC to 0.002, and accuracy to 0.501. Each iteration then adds an additional correct value to the prediction set until all, but 1 prediction is correct. True convergence cannot be reached as the MCC equation fails if there is not a single FP or FN value. By plotting the MCC and accuracy values at each iteration the correlation between the metrics can be found.

2.4.3 K-fold cross validation

K-fold cross validation is a ML method to minimise the impact of overfitting in predictive models. This process involves initially splitting the overall dataset into a training and a validation set[32]. Then, the training set is further split into K equal folds. A model is trained on K-1 of these folds and optimised for the remaining fold, which acts as the test set[69]. This is then repeated so that each fold has an opportunity to be both tested, and trained on, with the resulting K models having their performance measured against the initial validation set. As this validation set is unseen by all the developed models, it can differentiate overfit from robust predictions.

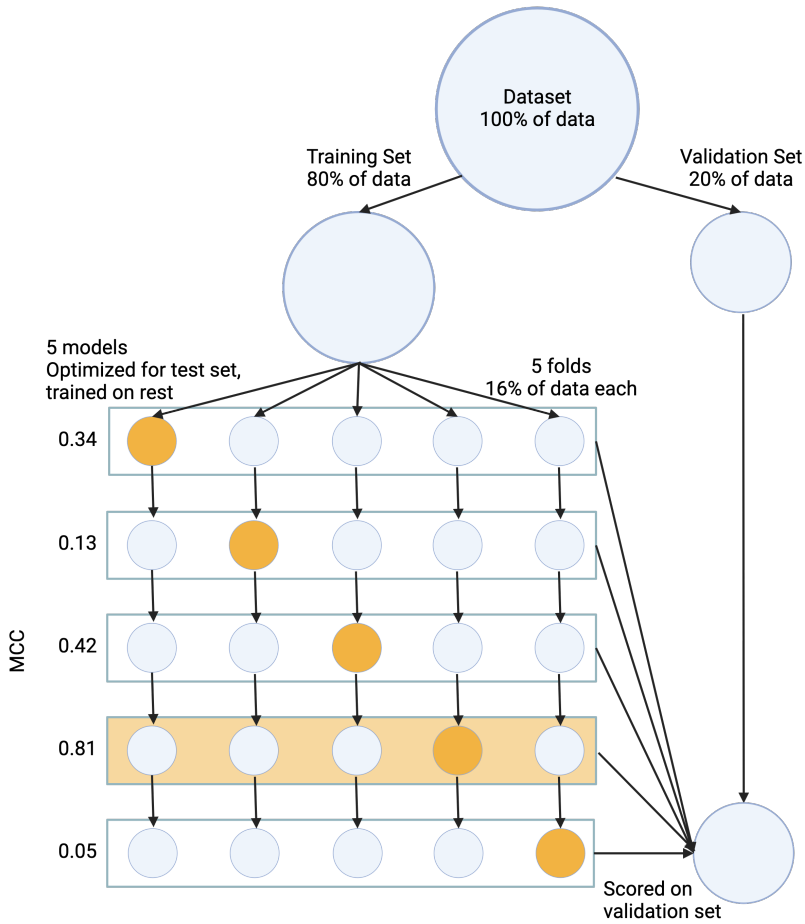

Figure 12: K-fold cross validation diagram

Created with BioRender.com

In the above diagram, K = 5, and thus there are 5 folds. Here, the models are rated based on their Matthew's correlation coefficient. The model with the best performance on the

validation set is kept as the most robust predictor (orange box), with the remaining, weaker models being discarded.

In this project, K is 5, and instead of taking the best model of the set, all models are kept with the consensus of the 5 models acting as the final prediction for each sample. For example, if 3 models were to predict a sample is toxic and 2 do not, the consensus would be that the sample is toxic. By using a consensus-based architecture the information each data point provides is maximised and relevant patterns that otherwise weaker models may possess are retained.

3 RESULTS

For brevity, not all model outputs are present in this document. For a full repository of results figures and data, see my Github (https://github.com/jngai23/Honours). All plots were created using Matplotlib[70] in Python.

3.1 Toxric datasets

22 datasets were extracted from the Toxric database containing 17773 molecules across 8 toxicity endpoints (Supplementary Table 1). In each dataset, all molecules were assigned either 1 (toxic) or 0 (non-toxic) for their toxicity values for that given endpoint. Class imbalance was present in 16 of the 22 datasets, prompting the balancing protocol to remove samples of the dominant class prior to model creation.

3.2 Feature extraction

To extract feature sets for ML algorithms, different methodologies were utilised consisting of structure-based and interaction-based representations. Structural representations describe molecules based on their unique chemical makeup, and the properties their constitution facilitates. On the other hand, interaction-based attributes aim to correlate molecules together based on their similarity in targets; compounds with similar targets may have similar functionality. The structural representations included Morgan fingerprinting, ChemoPy features, and PaDEL descriptors, whereas the drug-protein descriptors originated from a DrugBank drug-polypeptide target dataset. In addition to these representations, a fusion of MorganFP and DrugBank targets was created to determine if combining structural and interaction-based features could result in models greater than the sum of their parts.

3.2.1 Structural representations

Across all structural representation methods, no sample loss was found after feature extraction. These representations contained varying quantities of feature categories and value types.

Table 7: Structural representation feature counts

Binary feature types refer to the presence or absence of a feature or property, numerical features are integers like 0,1,2,3 etc. continuous features are floats and can be negative e.g. -0.532.

Representation	Feature count	Feature Types
MorganFP	2048	Binary
ChemoPy	201	Binary, numerical, continuous
PaDEL	1876	Binary, numerical, continuous

3.2.2 Interaction & fusion representations

As not all Toxric compounds were present in the DrugBank dataset, any molecules not in the overlap were removed (Supplementary Table 3).

Table 8: Interaction and fusion representation feature counts

DrugBank targets consist of one-hot encoded features that correspond to a given polypeptide target. The fusion dataset is a concatenation of DrugBank and MorganFP representations, combining the two feature sets into one with their total feature counts (2048 + 5450). Class imbalance ratios are identical between the DrugBank and Fusion datasets.

Representation	Feature count	Feature Types
DrugBank targets	5450	Binary
MorganFP-DrugBank fusion	7498	Binary

3.3 Model quality threshold creation

In established literature, quality models for QSAR and toxicity prediction tasks like this typically have at least the equivalent accuracy value of 0.8. To directly compare this to the metric used in this project (Matthew's correlation coefficient), a thought experiment was created to correlate accuracy in ideal conditions to MCC. By graphing the accuracy and MCC at each iteration the MCC at an accuracy of 80% can be found. Furthermore, models with less than a 60% converted accuracy can be considered to have little to no correlation and thus are of very low quality.

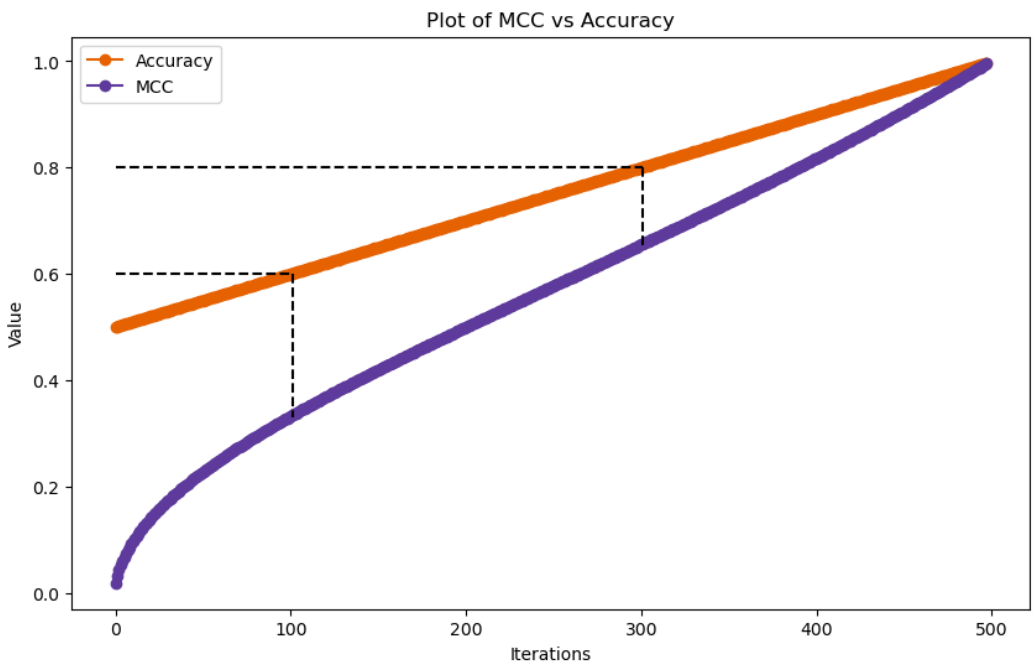

Figure 13: MCC vs accuracy comparison under ideal conditions

For the outlined model, initially, in iteration 1 there are 499 TP and 499 FN values. Additionally, there is one FP and one TN value to prevent a math error when calculating MCC. This results in an accuracy of 0.5 and an MCC of 0.001. In each succeeding iteration, one of the FN values is converted to a TP before the MCC and accuracy is calculated. This process is repeated until convergence. The MCC value when accuracy is 0.8 can then be found as 0.65. Furthermore, an accuracy of 0.6 represents an MCC of 0.33

3.4 Class Imbalance observations

Grossly uneven class quantities can cause a severe loss of efficacy in predictive models, particularly, this is the case for binary classification tasks such as toxicity prediction. To rectify this issue, datasets with class imbalance had samples of the dominant class systematically removed to preserve an even spread of samples. A greater than 1:4 class imbalance was present in 16 of the 22 datasets. As a result, these datasets had samples removed to match at least a 5:6 ratio (Supplementary Table 2).

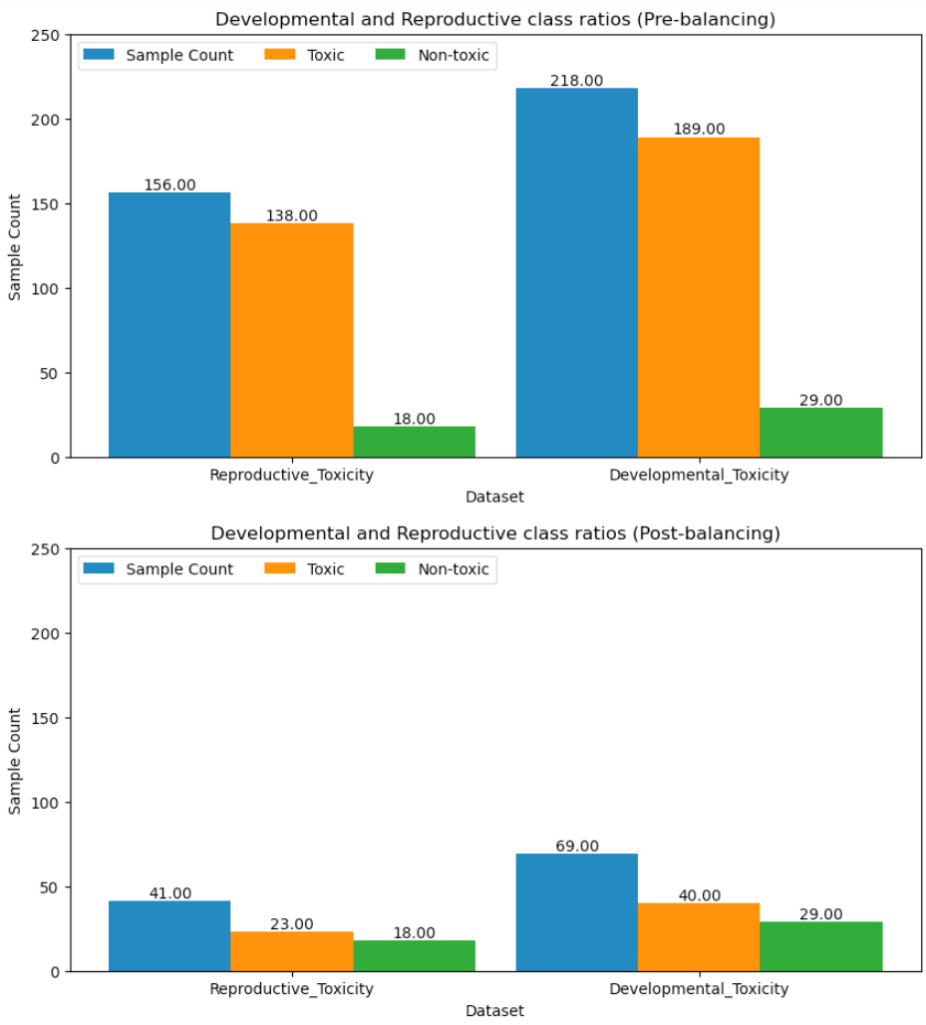

Figure 14: Developmental and Reproductive toxicity class balancing

An example of unbalanced data, initially both these datasets contain a class ratio exceeding 1:4. As a result, the class balancing function triggers, removing samples of the dominant class until datasets a class ratio at most of 5:6 is reached.

3.5 Model evaluation

Across 22 endpoints, 5 feature representations, and 4 model types a total of 440 predictive models were created. Each of these models act to determine whether an unknown molecule is toxic to an endpoint (e.g. cardiotoxicity), given the molecule is input in a set representation (e.g. its Morgan fingerprints). The validity of the models was ensured with a consensus K-fold validation method and their quality was gauged using MCC as a metric. Models with an MCC greater than 0.65 were considered high-quality models and are comparable to those found in existing literature. On the other hand, models with a metric value of less than 0.33 can be seen as barely superior to guessing.

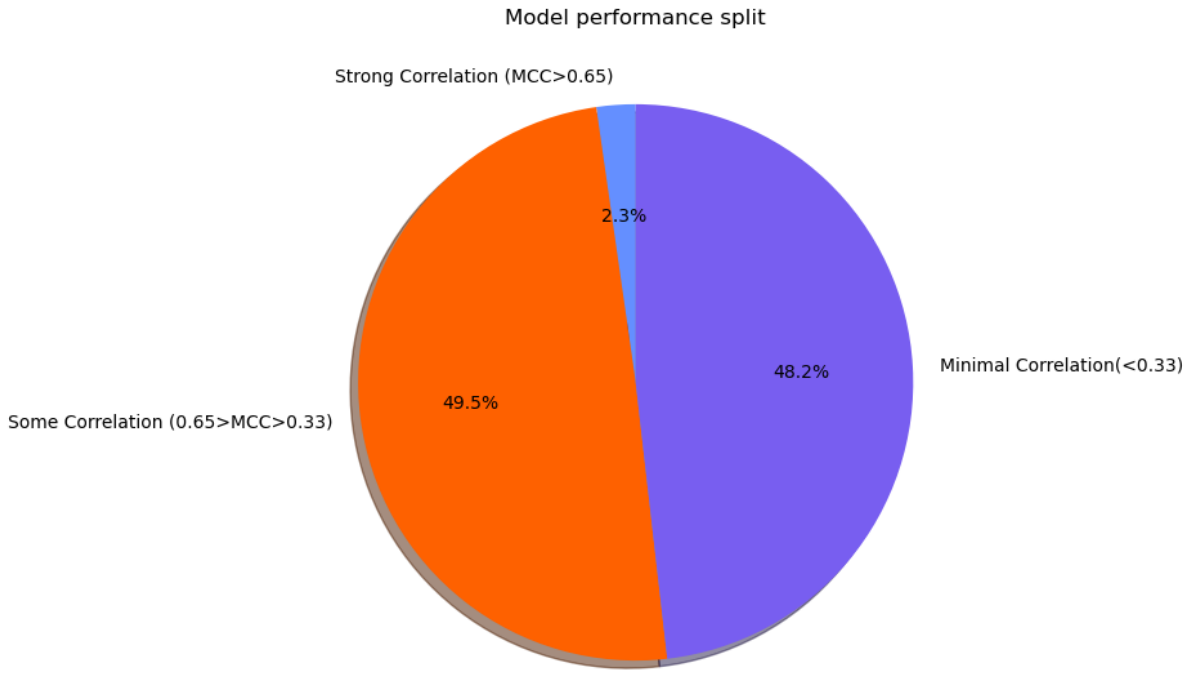

Figure 15: Total model quality distribution chart

Of the 440 created models across all representations, datasets and data types, 48.2% (212) were considered weak with an MCC less than 0.33. Contrasting this, a mere 2.3% (10) models exceeded the 0.65 MCC benchmark for strong models with the remaining 49.5% (218) models being between these extremities.

3.6 Feature representations summary

Different feature representations explore differing aspects of a dataset. In the case of this binary classification task, each structural feature representation extracts key constitutional information for molecule functoriality. On the other hand, the interaction-based representations aim to determine if molecules with similar targets have similar properties.

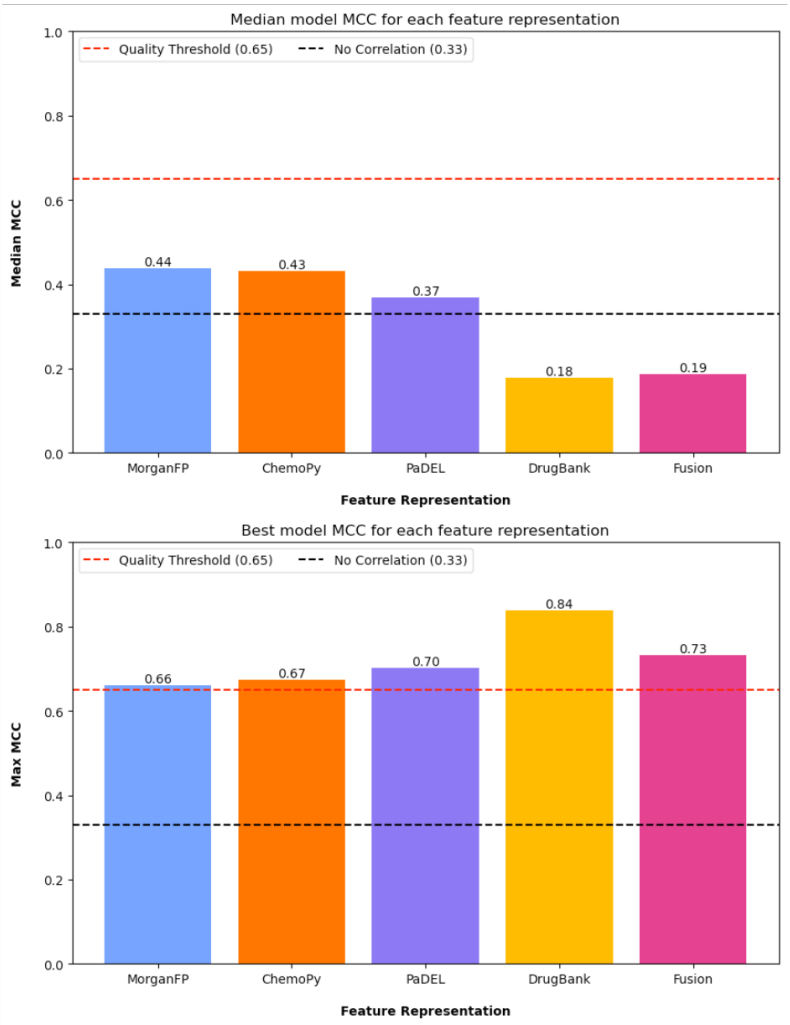

Figure 16: Median & maximum MCC of feature representations across models

Among all models, the structural representations (MorganFP, ChemoPy, PaDEL) perform marginally better than the interaction and fusion representations in terms of median values. MorganFP and ChemoPy have the greatest medians, followed by PaDEL, Fusion and DrugBank representations. Contrasting the median performance of each representation, the maximum MCC ranking of each of the feature extraction methods were reversed.

3.6.1 Morgan Fingerprints analysis

A structural feature consisting of one-hot encoded atomic groups, Morgan fingerprinting has seen extensive use in literature for QSAR and toxicity prediction. In the context of Toxric data, MorganFP representations consisted of 2048 features calculated with RDKit. Compared to other feature representations, MorganFP data had the highest median but the weakest peak model. This indicates that the representation is the most consistent in performance across model types and datasets.

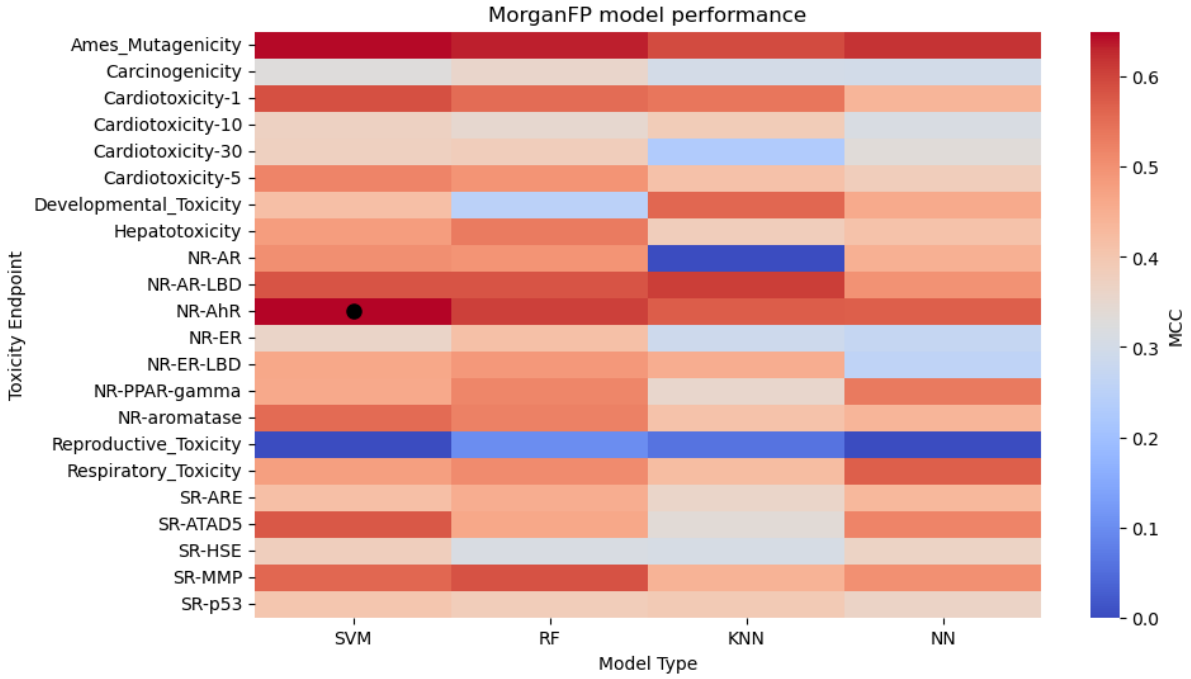

Figure 17: MorganFP model performance heatmap

MorganFP models have a generally middling (0.65 > MCC > 0.33) performance in most datasets. Symbolised by the black dot, a singular strong (MCC > 0.65) model was present in present in the 'NR-AhR' toxicity endpoint when coupled with the SVM model type. However, very poor performance (< 0.33 MCC) was seen in the 'Reproductive Toxicity' dataset across all model types. Furthermore, across model types, MorganFP performed consistently with KNN models slightly lacking compared to its peers.

3.6.2 ChemoPy predictors analysis

Generated via the ChemoPy Python package, these structural descriptors have one-hot encoded, numerical and continuous features corresponding to a molecule's atomic groups, charge and orbital energy amongst other factors. In terms of median performance, ChemoPy was second strongest, slightly behind MorganFP, contrasting its ranking as second worst at peak model performance.

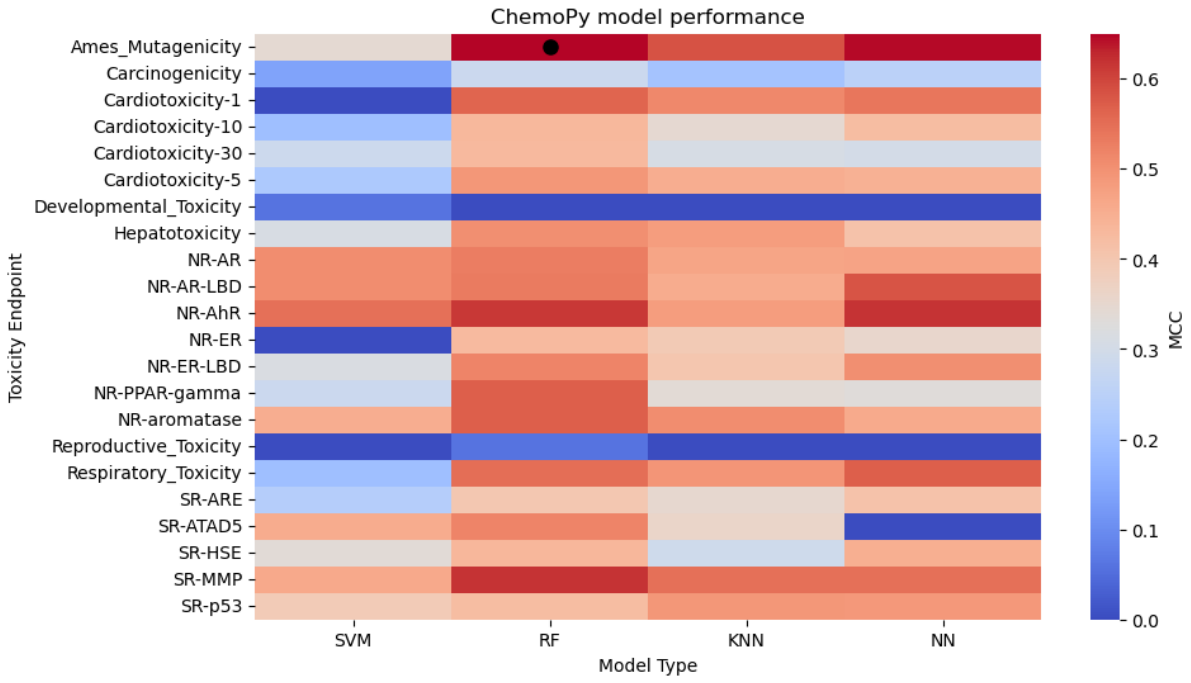

Figure 18: Heatmap of ChemoPy model performance

For ChemoPy models, a strong performance (MCC > 0.65) was seen in the RF model of the 'SR-MMP' and 'Ames_Mutagenicity' endpoints as well in the neural network model of 'Ames_Mutagencitity'. On the other hand, multiple very poor-performing models with next to no correlation (MCC < 0.33) were present across models in both 'Developmental_Toxicity' and 'Reproductive_Toxicity' datasets. Furthermore, SVM models consistently score unfavourably compared to other model types for ChemoPy represented data with near 0 MCC values in 'Carcinogenicity', 'Cardiotoxicity' and 'NR-ER' endpoints.

3.6.3 PaDEL descriptors analysis

Created through PaDEL descriptor's JavaScript file, this representation extracts features from both 2D and 3D representations of a molecule for a total of 1876 categories. These features can be one-hot encoded, numerical or continuous with all NaN values set to 0 and all infinite values set to the largest float value available in Python ($1.8*10^{308}$). In terms of both median and peak performance, PaDEL is in the middle of the pack compared to the other tested representations.

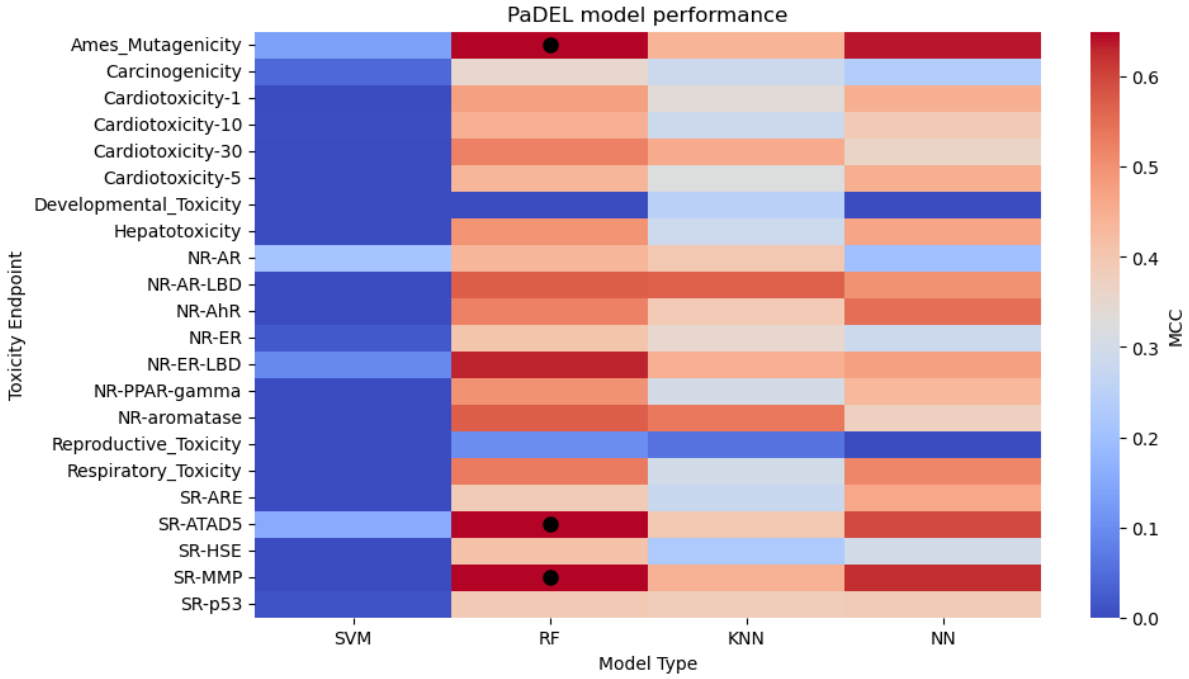

Figure 19: PaDEL performance heatmap

Immediately, it is evident that SVM models lacked immensely in model quality relative to other model types with this feature representation, with all its created models nearing 0 MCC. Contrasting this weak performance is a generally even degree of model quality amongst other model types, with KNN slightly underperforming compared to RF and NN models. Strong models can be seen in RF models for the 'SR-MMP', 'SR-ATAD5' and 'Ames_Mutagenicity' toxicity endpoints.

3.6.4 DrugBank Polypeptide targets analysis

Adapted from DrugBank's drug-polypeptide interaction data, this dataset consists of 5450 one-hot encoded drug targets identified as UniprotIDs. As not all molecules seen in Toxric data were present in the DrugBank data, compounds not in the overlap were removed, resulting in large quantities of sample removal in datasets (Supplementary Table 3). The dataset's median performance was the lowest of the cohort, directly contrasting its peak model that exceeded all other feature representations with an MCC of 0.83.

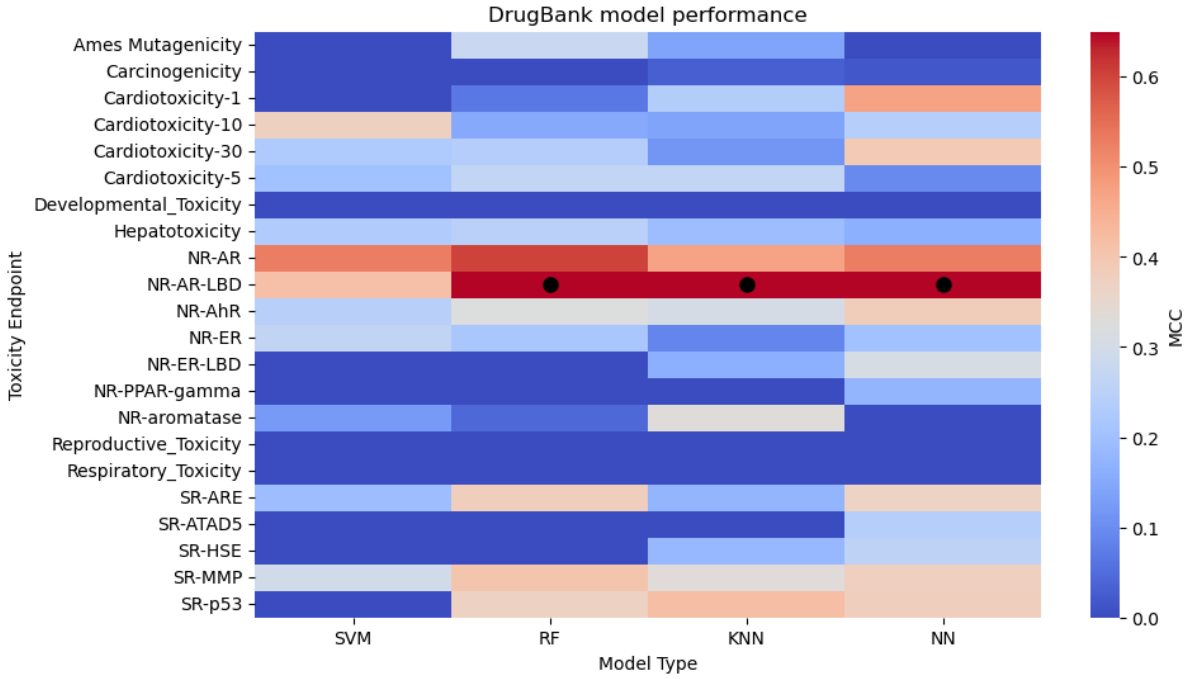

Figure 20: DrugBank model performance heatmap

In general, most of the developed models for the DrugBank data were very poor with most created models scoring a near 0 MCC. Strong MCC values were only seen in the KNN, SVM, and NN models of the 'NR-AR-LBD' endpoint which were the highest scored by any of the developed models. Notably, SVM models tended to struggle in performance, whereas the neural network models performed well compared to other model types when combined with this feature representation.

3.6.5 Early fusion analysis

Generated by combining MorganFP and DruBank feature sets, the fusion data consists of 7498 total feature categories. Through linking a structural and an interaction-based interaction, models are provided with a wealth of information across differing contexts that may aid in finding patterns for predicting toxicity. In terms of median performance, the representation was almost identical to the DrugBank data, with its best model performing better than all the structural representations while also being inferior to DrugBank's peak.

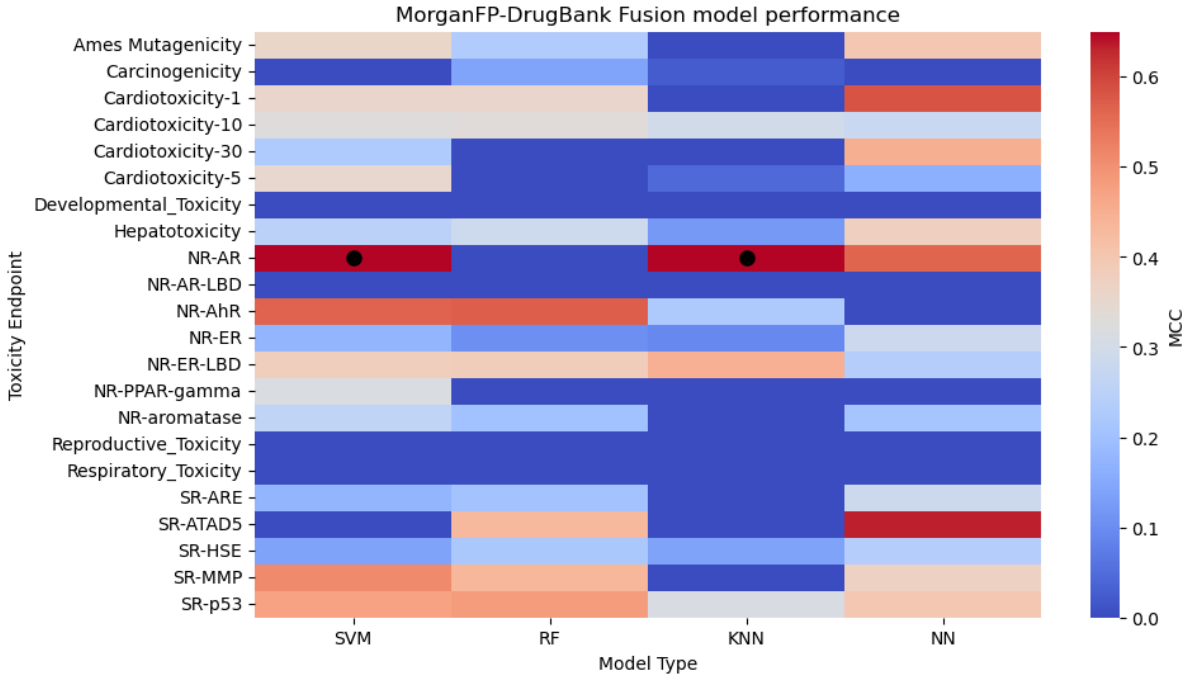

Figure 21: MorganFP-DrugBank polypeptide target fusion performance heatmap

Performing closer to the DrugBank rather than the MorganFP representations, the fusion features generally produce poor models with strong MCC values found only in the KNN and SVM 'NR-AR' models. Outside of this strong performance on the 'NR-AR' endpoint, K-nearest neighbours had in general the lowest compatibility with the fusion representation compared to other model types. Contrasting this, support vector machines had the best synergy with these datasets with the model type demonstrating some success in datasets like 'NR-AhR' and 'SR-MMP' where other model types faltered.

3.7 Implemented machine learning methods

Model types each utilise differing methods to classify data which allows them to excel in different datasets and feature representations. By evaluating a range of model types for this project, each dataset and feature representation can be better explored, and relationships between certain data types and model types can be discovered.

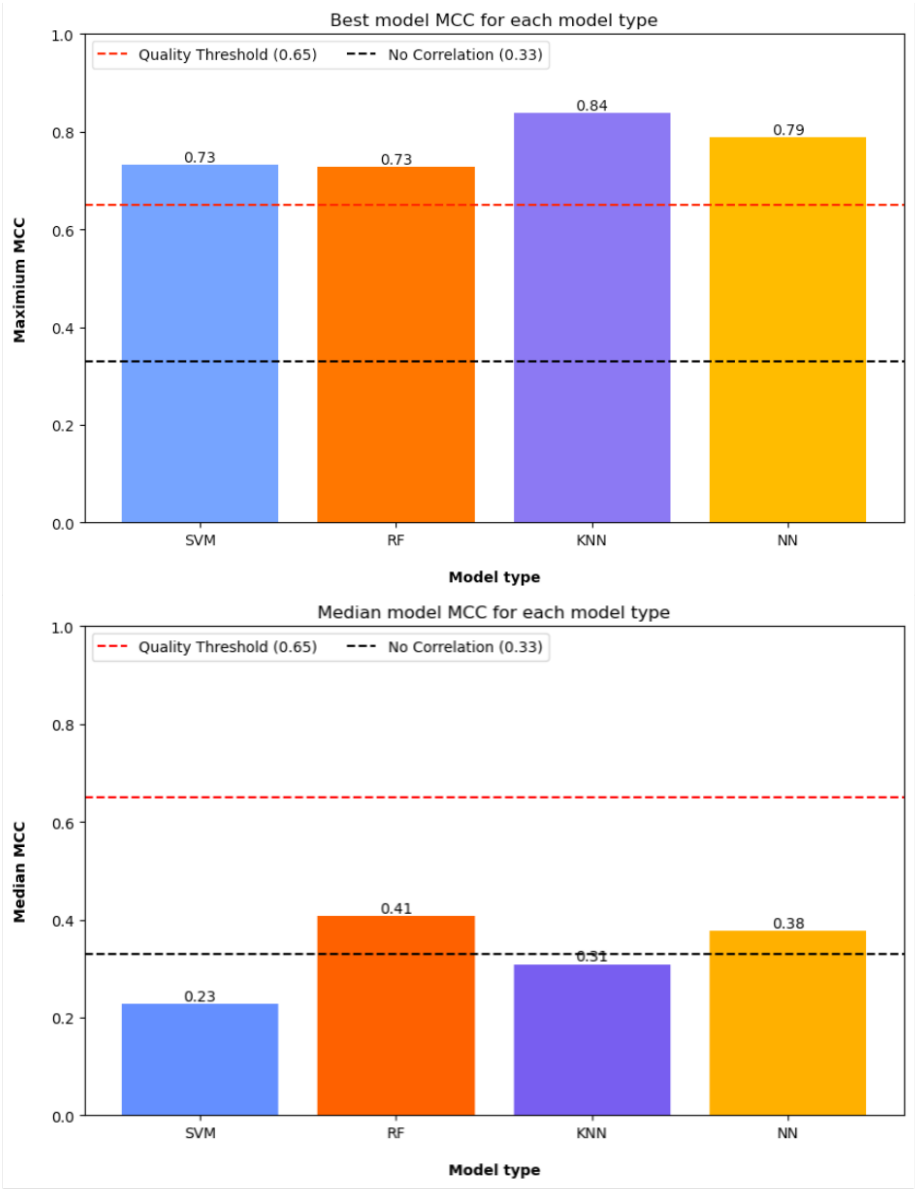

Figure 22: Median and maximum MCC values for machine learning algorithms

In terms of median performance, RF and neural network techniques are the most consistent with their median model containing at least some sort of correlation (MCC > 0.33). On the

other hand, KNN and SVM algorithms tended to generate models weaker than this threshold. Contrasting this, in terms of peak performance, K-nearest neighbour's best model performed far above its contemporaries, followed by neural networks, then SVM and RF models. Overall, all model types are comparable and consistent when it comes to model quality.

3.7.1 Support vector machines assessment

A model-generating algorithm that involves finding a hyperplane to separate classes, the SVM algorithm was selected for its prevalence in binary classification tasks. This project utilised a polynomial kernel for SVM model creation with the primary hyperparameter being the degree of the polynomial, ranging from 1-4. The method tends to succeed best with lower dimensionality data and tends to struggle in complex and large datasets. Concerning its median and best models, this ML method has both the lowest peak and median values.

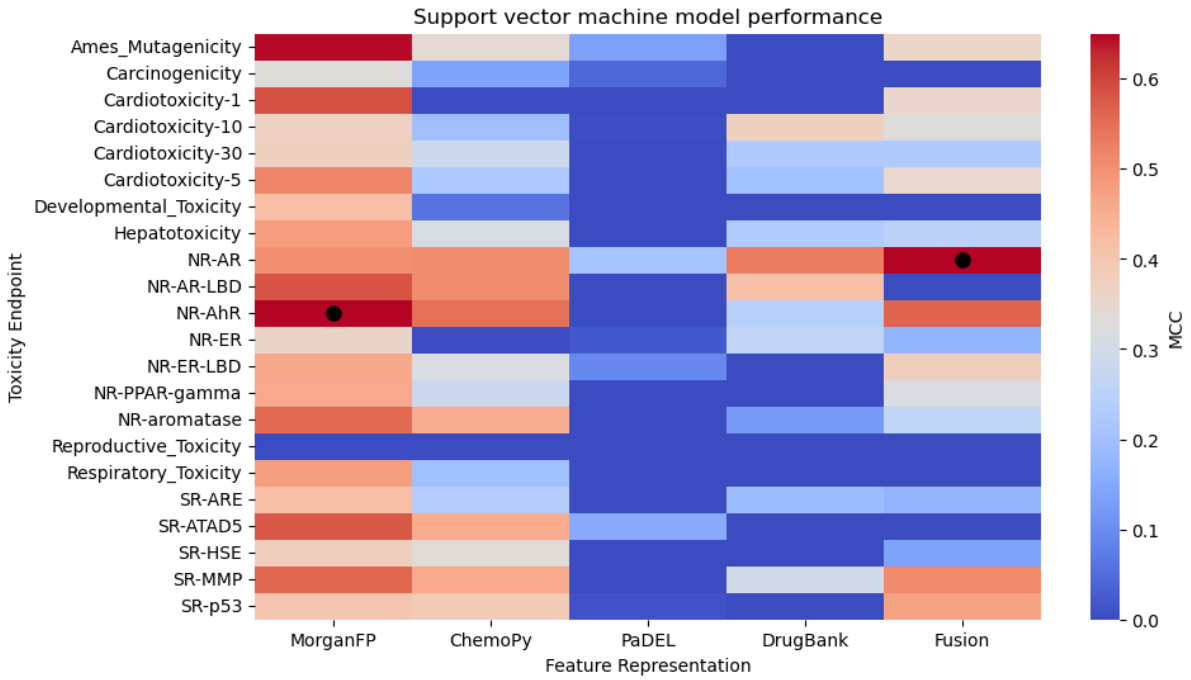

Figure 23: SVM performance heatmap

Support vector machine synergises best with MorganFP feature representations with many models of moderate success (0.33 < MCC < 0.65). On the other hand, when utilised with PaDEL data the SVM algorithm produced zero models with any sort of success in toxicity prediction, which indicates that the two are a dysfunctional feature-model combination.

3.7.2 Random forests assessment

A ML method consisting of multiple decision trees, random forest is known as a generally applicable algorithm for binary classification tasks, particularly in the field of QSAR analysis such as toxicity prediction. The median and maximum performance of RF models in this project are contrasting, while random forest-based models have the highest median of the ML methods, they also have the lowest maximum MCC value. The primary hyperparameter for this model type is the number of features assessed at each node, which was tested for values between 1 and 50.

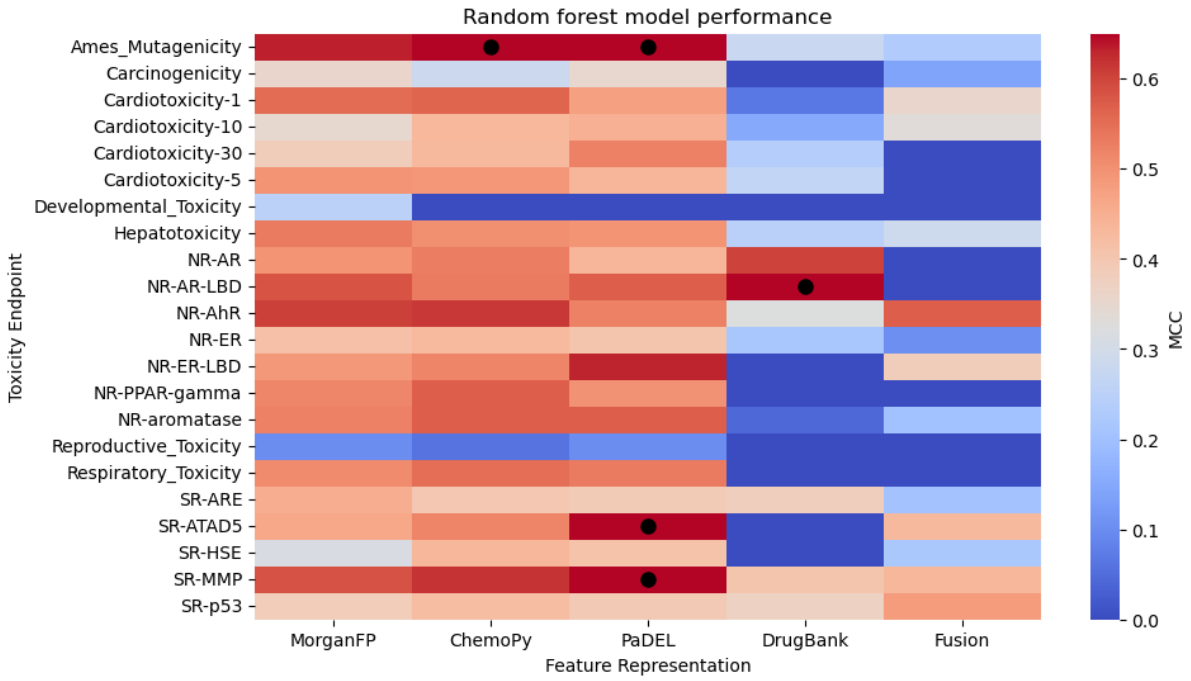

Figure 24: RF performance heatmap

Consistent with its median performance RF algorithm-based models typically have at least some links between predictions and actual outcomes (0.33 < MCC < 0.65). There are multiple strong (MCC > 0.65) models present, most of which are in conjunction with PaDEL feature representations, indicating a synergy between this algorithm-representation combination. On the other hand, there was poor performance in DrugBank and Fusion-based datasets.

3.7.3 K-nearest neighbours assessment

A straightforward ML algorithm that classifies unknown data based on the class of its surrounding samples, KNN models have seen success in toxicity prediction tasks. The model type's median scores were second lowest amongst its peers but its peak MCC among generated models value ranked first at 0.84. This indicates that the models are inconsistent and as a result dependant on dataset and representation quality to provide accurate predictions.

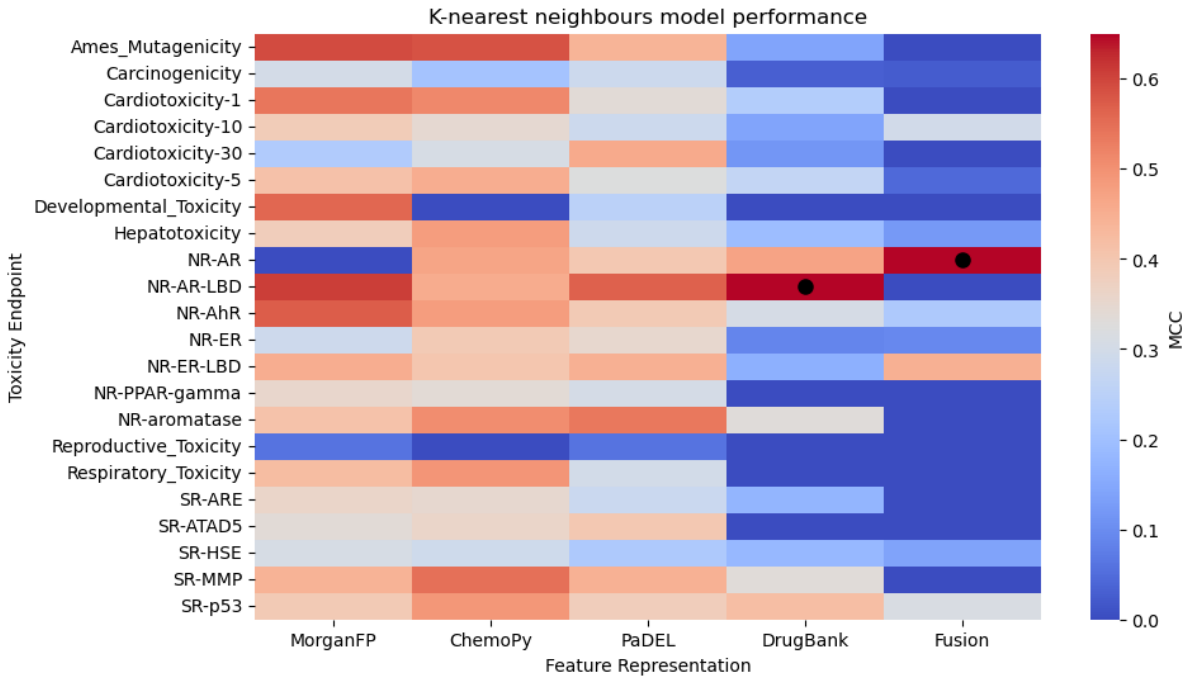

Figure 25: KNN performance heatmap

Correlating to its median and maximum performance, KNN models are overall low in quality, with little correlation between predicted and actual toxicity values. This lack of model quality is especially prevalent in DrugBank and Fusion data, but in 2 cases ('NR-AR-LBD' and 'NR-AR' endpoints) the algorithm produced exceptional models. Amongst the structural representations, KNN models found less success in PaDEL representations in comparison to ChemoPy and MorganFP datasets.

3.7.4 Neural network assessment

An extremely broad category of ML models, this project implemented a variation of a feed-forward neural network based on the architecture of the perceptron. This initial structure was compounded with the addition of the 'Adam' gradient descent algorithm. Neural networks as a model type were second place in both median and peak performance relative to the other ML methods, indicating the model type produces consistently performing models.

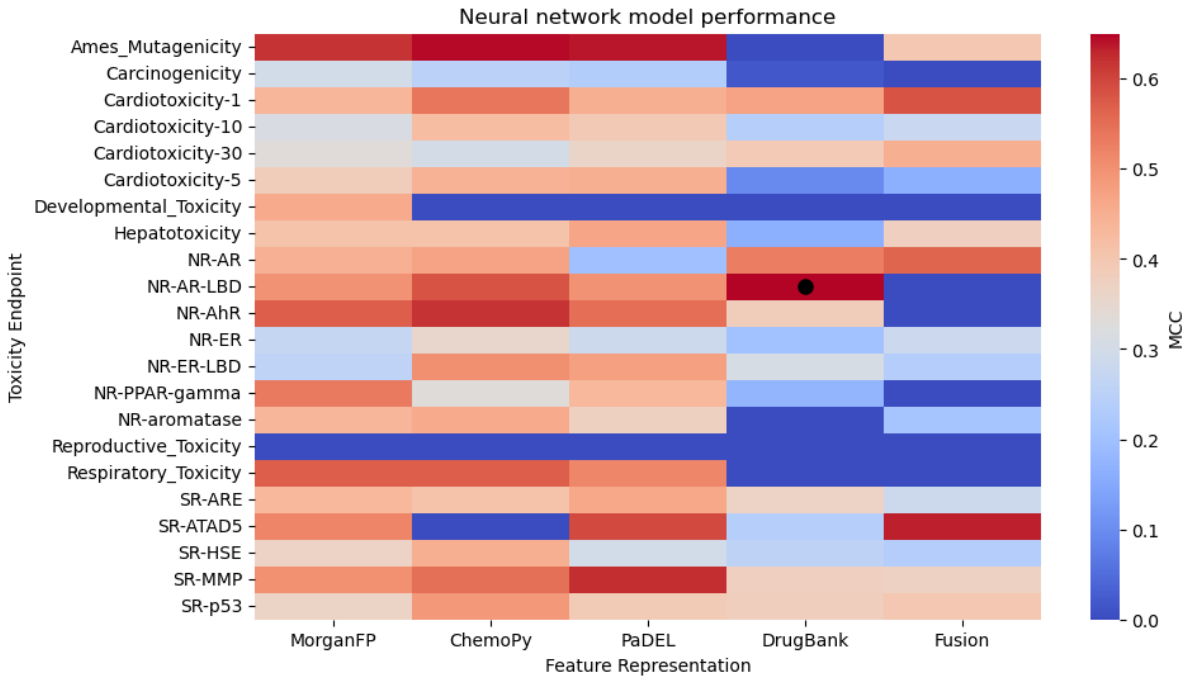

Figure 26: Neural network performance heatmap

The architecture shows strong success across most feature representations but noticeably lacks in performance in the DrugBank and Fusion-represented datasets. Most of the developed models were above the 0.33 quality threshold, with one above the 0.65 MCC threshold. Overall, the developed neural network architecture is generally applicable to most data representations and datasets.

3.8 Toxicity endpoint datasets

For predictive models to excel the data they are trained on must be well procured and high quality. The Toxric datasets have been aggregated from various sources, both new and old (Supplementary Table 1). By evaluating a range of datasets, the effectiveness of each model type and representation method can be assessed on strong and weak data.

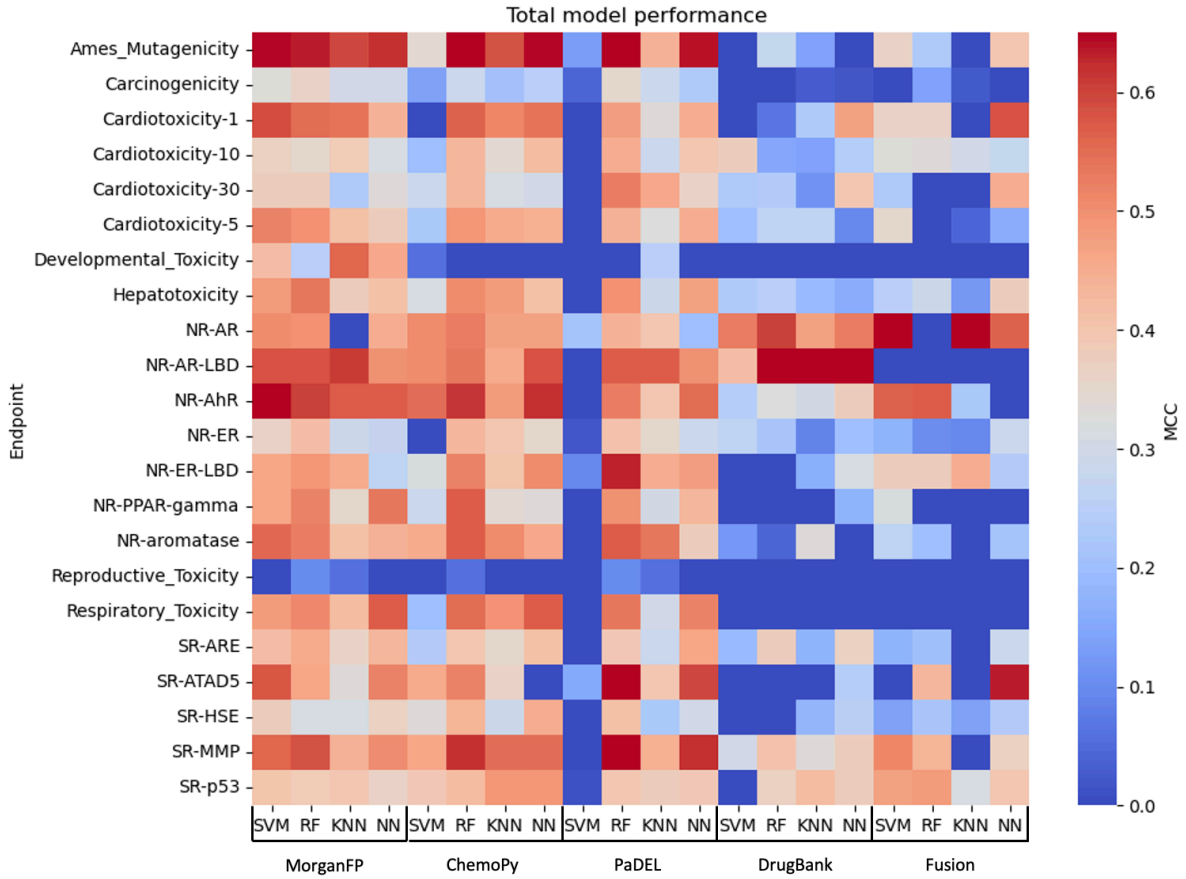

Figure 27: Heatmap of performance across all models

With respect to the data quality of each endpoint, the 'Reproductive_Toxicity' and 'Developmental_Toxicity' toxicological endpoints are the worst performing across all feature representations and model types. Contrasting this, 'NR-AR-LBD' and 'Ames_Mutagenicity' are toxicity endpoints that perform very well in most feature representations and datasets. The remaining endpoints were generally consistent in MCC value, performing better in structural representations rather than interaction or fusion sets.

3.9 Analysis of variance between tested model attributes

To determine if there is a significant difference in model performance between the utilised endpoint, feature representation or model type, a Kruskal-Wallis H test can be utilised[71]. This test involves determining if there is a statistical significance between the means of groups within a category. In this case, the categories are either the model types (SVM, RF, KNN, NN) or feature representations (MorganFP, ChemoPy, PaDEL, DrugBank, Fusion) or endpoints. Contrasting to other statistical tests like t-tests or analysis of variance tests, Kruskal-Wallis H tests do not assume data falls along a normal distribution or if the data has an equal variance. As a result, this test type is viable for the data of this project's model performance values which do not conform to such a data shape.

In these cases, the null hypothesis would be if feature selection/model type had no impact on the performance values (MCC) of each model. The p-value threshold for correlation is 0.05 and was calculated using the 'scipy' package in Python[72].

Table 9: Kruskal-Wallis H test of feature representations and model types

Across both feature representation and model type scenarios, the p-value was below the 0.05 threshold for significance, indicating that there is a difference in means in at least one of the constituent groups. In the case of feature representations, the p-value was over ten orders of magnitude lower than the 0.05 threshold which shows that is an implication that feature representation has a large impact on model performance. This is also true for model types and toxicological endpoints, but to a lesser extent.

Group	p-value
Toxicological endpoint	6.886041455389232e-14
Feature representations	2.6304049309889784e-19
Model types	2.7007892884026123e-05

4 DISCUSSION

Excessive toxicity is the cause of one-third of drug knockouts during the drug development pipeline[2]. Most of these candidate knockouts occur during the later pre-clinical and clinical testing phases which have steep resource requirements to run per drug candidate[3]. This project aims to develop predictive models capable of catching these toxic drugs early in this process, saving resources that would be spent on toxic molecules. These predictive models are based on data from the Toxric database, which contains datasets consisting of various molecules and their binary toxicity to a given endpoint. These compounds were then translated into various structural and interaction-based representations, before being fed into various model training methods for training and testing. In doing this, endpoint-representation-model type combinations can be assessed, and the impact of each of these three components on model performance can be assessed.

4.1 Assessed toxicity endpoints

From the Toxric database[21], 8 primary toxicity categories encompassing 22 endpoints were assessed (Supplementary Table 1). As indicated by the produced models and statistical tests (Figure 27) (Table 9), each dataset had wildly varying performances, such as how the 'Developmental_Toxicity' dataset produced poor models independent of the representation and model types utilised. This may stem from where the data for each set originates. Each Toxric dataset was compiled from various pieces of scientific literature and pre-existing databases such as LiverTox[73] and the Tox21 project[74]. By analysing and comparing the performance of each Toxric endpoint to existing literature, they efficacy of the database for toxicity prediction may be assessed.

4.1.1 Developmental & Reproductive toxicity

Both sourced from the Aggregated Computational Toxicity Resource (ACToR) database[23], the developmental and toxicity datasets showed the worst performance out of all datasets in this task. The ACToR database was created by the United States Environmental Protection Agency (EPA) in 2007 which aggregates data from biochemical assays. This database has seen success in QSAR tasks, such as in the aquatic toxicity prediction of chemicals for machine

crustaceans[75]. In this study, ACToR toxicity data in combination with fingerprint-based representations allowed for developed models with area under curve values ranging from 0.66 to 0.99, indicating medium to strong model performance[75]. This directly contrasts the ACToR sourced datasets in this project (developmental and reproductive toxicity) where, of the 40 models created for these datasets, 36 had an MCC lower than 0.33, indicating little to no correlation. The resulting discrepancy in model quality between these studies could indicative that the ACToR database is better suited for aquatic toxicity endpoints rather than human developmental and reproductive toxicity profiling.

4.1.2 Endocrine Disruption

Concerning 12 total datasets (NR-AR, NR-AhR, NR-aromatase, NR-AR-LBD, NR-PPAR-gamma, NR-ER, NR-ER-LBD, SR-ARE, SR-ATAD5, SR-HSE, SR-MMP, SR-p53), the endocrine disruption endpoints were aggregated by Toxric from the Tox21 project[74]. Another EPA program, Tox21 began in 2008 and acts to collate toxicological information via biochemical assays submitted under a set procedure for analysis and compilation into a database.

To emphasise the efficacy of this database for QSAR modelling, the National Institute of Health hosted the 'Tox21 Data Challenge' in 2014 which was an open call for researchers to create predictive models using their endocrine disruption datasets[13]. Resulting from this challenge were models with AUC values ranging from 0.7-0.9, indicating strong model production. The overall performance of the Tox21 datasets is consistent with the models created in this project, but individual dataset performance differs between the challenge models and those implemented in this project. This discrepancy in model quality can be attributed to the range of feature representation and model types utilised by challenge participants that were not present in this project such as inductive descriptors and elastic net algorithms. Overall, the endocrine disruption endpoints present in Toxric resulted in productive models ready for use within the drug developmental pipeline.

4.1.3 Hepatotoxicity

Concerning toxicity to the liver, and amalgamated from three primary databases, Toxric's hepatotoxicity dataset is among the most diversified of the sets utilised for this project. These

databases include LiverTox[73], the drug induced liver injury ranking (DILIRank[76]) and the liver toxicity knowledge base (LTKB[77]). Due to the diversity of the datasets utilised, it can be difficult to gauge the effectiveness of this project's hepatotoxicity predicting models compared to existing methods. In the case of the constituent datasets, predictive models using DILIRank and LTKB data resulted in balanced accuracy scores of 0.75[78], on the other hand, LiverTox data produced peak accuracies of 0.9[79]. The better hepatoxicity models produced in this project lie within this range, and thus are comparable. However, the discrepancies in performance potentially due to the presence of data from differing datasets. Due to these factors, the hepatoxicity endpoint from Toxric presented middling results in this project, but likely can be optimised with better representation and model type selection as seen in existing literature.

4.1.4 Cardiotoxicity

The four cardiotoxicity datasets ('Cardiotoxicity_1/5/10/30') were taken by Toxric from a scientific literature piece on extracting toxicity information from congregated data by Wu et. al. (2021)[22]. By tracing the cardiotoxicity data from this study back to its source, it ultimately originates from ChEMBL[80] with the number suffixing each of these datasets indicating the IC50 threshold utilised. The developed models using these datasets saw middling performance values with no strong (MCC > 0.65) models created. In existing literature, ChEMBL-derived datasets have seen general success in structure-based toxicity and QSAR prediction. One of these studies utilised ChEMBL cardiotoxicity data to produce peak accuracy values ranging from 0.6 to 0.85 using an SVM methodology based on bioactivity data[11]. This is consistent with model data generated from this project (Figure 27) with the discrepancy in peak results potentially stemming from the differing feature representation utilised. As a result, it is evident that these toxicological endpoints are

4.1.5 Respiratory toxicity

Also stemming from the paper by Wu et. al. (2012)[22], Toxric's respiratory toxicity dataset can be tracked to the now defunct (as of 2019) ToxNet database run by the National Library of Medicine[81]. In one study, this database, among multiple others was used to predict genotoxicity data with models peaking at a 0.8 accuracy[82]. While obfuscated with other datasets this still demonstrates that ToxNet data can produce valid models which is supported with the models

created in this project. Here, respiratory toxicity-based models using data originating from Toxric produced models with a peak MCC of 0.57 which equates to an accuracy of 0.75 (Figure 13). With middling performance, and a lack of strong models in literature, the respiratory endpoint in Toxric may be unsuited for toxicity prediction in drug development. However, with better optimization in model and representation choice, perhaps superior models may be created with viability for research use.

4.1.6 Ames Mutagenicity & Carcinogenicity

Defined by a method known as the 'Ames test' which is used to determine the mutagenicity of a chemical[83], Toxric has sourced 'Ames_Mutagencity' data from the same paper as the respiratory and cardiotoxicity datasets[22]. At its source, the Ames Mutagenicity data can be followed to ToxNet, along with the Chemical Carcinogenesis Research Information System (CCRIS) and the Carcinogenic Potency Database (CPDB)[84]. (Supplementary Table 1). The CPDB has seen use in QSAR prediction of mutagenicity with a peak accuracy of 0.83[68]. This correlates with this project's performance with the 'Ames_Mutagencity' dataset which produced multiple models of strong (MCC > 0.65) quality. As a result, the models produced by this endpoint within this project are robust and can be readily applied in drug development.

The Toxric 'Carcinogenicity' dataset is also sourced directly from the CPDB, but only concerns long-term animal-based data. On the other hand, the 'Carcinogenicity' dataset lacked in performance despite originating from similar data. This may be because of the lack of molecules present in the 'Carcinogenicity' dataset (1021) compared to the 'Ames_Mutagenicity' (7485) set (Supplementary Table 1). The difference in sample count is relevant to the performance of a model and as a result, the anomaly in MCC values may be attributed to this discrepancy. Overall, the poor performance of the 'Carcinogenicity' endpoint in this project means that greater research into the choices of model type and feature representation may be required to produce robust models in this context.

4.2 Tested data representations

Various feature representations were included in this project to determine which characteristics of a drug yield the most informative data for predictive modelling. Structural information, like a molecule's functional groups or charge were covered by Morgan fingerprinting, ChemoPy

features, and PaDEL descriptors, while drug-target interactions were identified with DrugBank data. Furthermore, by combining the MorganFP and DrugBank data, both structural and interaction-based data can be assessed in tandem. Overall, the selection of feature representation has a profound impact on model quality as demonstrated by the significant difference in means found with the Kruskall-Wallis H test (Table 9). This highlights the importance of applying multiple forms of feature selection for toxicity prediction to encompass as many drug characteristics as possible for the creation of robust models.

4.2.1 Morgan fingerprinting

Consisting of 2048 one-hot encoded atomic groups and implemented with RDKit[51], MorganFP was the most consistent feature representation in this project (Figure 17). Scoring the highest median amongst other feature extraction methods, Morgan fingerprinting can be seen as malleable and valid amongst most endpoints. However, the lowest peak MCC value out of its peers also indicates a lack of depth in feature quality that prevents strong (MCC > 0.65) models from being formed.

The middling quality feature extraction can be attributed to the simplicity of each individual category as they simply indicate the presence or absence of a given atomic group[28]. As a result, while this provides a general overview of the structural properties of a molecule, Morgan fingerprinting may lack the specificity to differentiate toxicity in edge cases, preventing the creation of robust models. This is consistent with existing literature that describes 2D molecular fingerprints as generally applicable[28], but underperforming compared to more specialised descriptors[85]. From these properties, MorganFP can be seen as a dowsing feature representation in the context of this project, that can aid in determining areas of possible toxicity before more rigorous analysis is conducted.

4.2.2 ChemoPy

The ChemoPy package extracted created 201 binary, numerical and continuous features from a 2D representation of each molecule. In terms of performance, generally the representation showed middling results across model types, but particularly suffered when it came to SVM models. This may be because of the feature complexity present in ChemoPy representations which match poorly with SVM models (See 4.3.1). The otherwise consistent performance of

ChemoPy data in differing model types is supported by existing literature which demonstrates that model choice has a minimal impact on ChemoPy model quality[54].

Compared to its peers, ChemoPy is almost entirely overshadowed by RF models with the representation struggling to find a niche relative to the other model types explored in this project. This may be a result of the limited number of features created by the package (201) which correlates to a relatively low data dimensionality, and thus a potentially poor overall representation of data[35]. With the inclusion of additional features, such as those generated via fingerprinting ChemoPy performance may be improved and better implemented in toxicity prediction.

4.2.3 PaDEL descriptors

Consisting of a large quantity of nonlinear features extracted from 3D molecule graphs, PaDEL descriptors saw model-dependant success across all toxicity endpoints (Figure 19). These features were derived from a 3D visualisation of each molecule, allowing for greater depth in feature intricacy[86]. Due to the complexity and lack of linearity between PaDEL features, SVM models suffered greatly in performance. Contrastingly RF models saw great success, producing three strong (MCC >0.65) models. This is supported by current research which sees the representation perform well with most ML models except for SVM, which produce poor models similarly to those seen in this project[87].

In terms of overall performance, the representation was middling among the implemented feature selection methods, being positioned third out of the five representations in both median and peak performance. Due to the strong synergy present with random forests, these models have potential to be applied in the drug screening process to confidently single out toxic drugs.

4.2.4 DrugBank polypeptide targets

Originating from DrugBank's website, this representation consisted of the overlap of DrugBank and Toxric data, causing large amounts of data loss. In terms of performance, the developed models produced poor results irrespective of model architecture (Figure 20). This may be attributed to data loss, which led to a reduced amount of training data available for each model to learn from[88]. Due to this drop in sample size and subsequent impact on model quality, it can be difficult to assess the impact of model choice in the DrugBank representation.

Contrasting the sample size discrepancy and overall weak models, DrugBank representations produced the highest scoring models among all feature extraction methods in conjunction with 'NR-AR-LBD' data, peaking at an MCC of 0.83. This performance implies that the represented features are of high quality, and the poor performance can be attributed to data loss. Though seeming like an outlier relative to the other DrugBank-represented datasets, the use of K-fold cross validation minimises likely ensures the reproducibility of these models.

In general, interaction-based representations like these are typically lower performing than structural feature extraction methods for QSAR tasks like toxicity prediction[31]. This is for the most part consistent with the produced models with the poor overall quality of this interaction-orientated representation. But in certain cases, DrugBank data can excel and outperform structural data. As a result of these properties, DrugBank polypeptide feature extraction is best utilised to identify toxicity in specific endpoints where the representation thrives.

4.2.5 MorganFP-DrugBank early fusion

Feature fusion is a feature representation methodology that involves combining feature sets at differing stages of model development[89]. In the case of this project early fusion was implemented where two datasets (MorganFP and DrugBank targets) were amalgamated before model development. As a result, ML techniques have access to a multifaceted representation of a molecule from different aspects of a molecule which may allow for patterns to be drawn across seemingly unrelated qualities. However, as the feature set was derived from the DrugBank representation, its large amount of data loss was carried over which may obfuscate the performance of the fusion features.

Comparatively to its constituents, the fusion set overall performed better than the DrugBank data but notably eschewed its excellent scoring on 'NR-AR-LBD' data in favour of a more rounded performance across other endpoints. On the other hand, compared to MorganFP it is far less generally applicable but has greater peaks in model quality such as in its strong 'NR-AR' models. As a result, fusing these representations together culminated in an average representation performance-wise between the DrugBank and MorganFP methods. This is consistent with existing literature on early fusion methods[59] which demonstrate how fusion at an early level typically averages out feature performance. Due to these attributes, the fusion

representation is best utilised similarly to DrugBank data; to indicate toxicity in niche endpoints.

4.3 Evaluated model types

Four different machine learning methods were assessed in the context of this project. These included random forests, support vector machines, K-nearest neighbour models, and neural networks. Each chosen for their unique mechanisms of action, the differing model types showed a significant difference in means (Table 9) (Figure 27). Therefore, it is likely that the chosen model creation method has a large impact on the quality of the final produced model, and, by implementing a wide swathe of ML methods the best possible model for each endpoint-representation combination can be developed.

4.3.1 Support vector machines

Involving the creation of a hyperplane to separate data plotted in Nth dimensional space, SVM models in this project have shown an overall weak performance compared to its peers (Figure 21). Here, the algorithm produced models with the lowest peak MCC and the second-lowest mean metrics. In particular, the ML method suffered with PaDEL-represented data relative to other model types (Figure 27). This poor performance may be due to the high complexity of the PaDEL data compared to other structural representations. Conventionally, SVM models have been known to struggle with classifying non-homogenous data which can be seen in the wide range of non-linear, not necessarily interrelated structural features found in PaDEL. This, to a lesser extent, may also be the case for ChemoPy data which contains a lower count of complex continuous features. On the other hand, SVM functions excel with high dimensional (high feature count), linear features like those found in MorganFP data which contained a large quantity of simple and correlated features in the form of 2048 binarily encoded atomic groups. Finally, in the case of DrugBank and fusion data, SVM models were less effective than its peers which may potentially be due to how these features are nonlinear categories; some of the polypeptide targets may be wholly unrelated from one another. By taking these properties into account the developed SVM can be seen as a specialised set of ML models for prediction using specific endpoints with large amounts of linear data such as Morgan fingerprinting.

4.3.2 Random forests

An ML method that consists of multiple decision trees, RF models showed strong success across all feature representation types relative to its peers. Conventionally random forest models show strong success in nonlinear problems for classification tasks. This is reflected in its strong success in PaDEL-represented data where the model type produced 3 strong (MCC >0.65) models. Its success can be attributed to RF's capability to assess multiple features at each node which allows for the analysis of multiple feature categories simultaneously. As a result of screening multiple features in tandem, a more complex analysis of data may be achieved which is reflected in its success with complex data types. On the other hand, while still strong, among the structural representations, RF models slightly underperformed with MorganFP, DrugBank and Fusion data which may be due to the simplicity of the feature sets. Due to these characteristics, the created RF models can be generally used for toxicity assessment across multiple endpoints.

4.3.3 K-nearest neighbours

A non-iterative algorithm involving the classification of new data with the identity of known neighbouring data, KNN models performed moderately on most data. The algorithm was also responsible for the two best-scoring models in the 'NR-AR' Fusion and 'NR-AR-LBD' DrugBank datasets respectively. These are datasets in which other models also performed well with 3 of the four model types scoring an MCC above 0.65 in the 'NR-AR-LBD' data, implying that this dataset is rich in relevant patterns towards toxicity prediction. KNN's compounded success on strong dataset-representation combinations may be due to its simple classification method which works effectively with clustered, and thus information-rich data[47]. On the other hand, because of its simplicity, KNN models tend to struggle in more complex datasets and representations where patterns are difficult to locate[48] such as PaDEL or ChemoPy. Due to these properties, the developed KNN models can be used to quickly and certainly gauge specific endocrine disruption endpoints for a given molecule.

4.3.4 Neural networks

The most consistent among the model types across all datasets and feature representations, the implemented feed-forward neural network saw moderate to strong performance across almost

all its developed models. This consistent, but moderate performance can be attributed to the architecture utilised, with the implemented optimiser, layer size and loss function, among other hyperparameters being designed for general use[64]. As a result, the developed models lack the precision that optimised neural network models typically allow for. While overshadowed by other model types for each dataset-representation combination, with proper optimization and hyperparameter selection for each use case, the neural network framework has the potential to surpass its peers in performance[35].

4.4 Statistical tests

Do directly identify if adjusting each model constituent affected model performance a series of Kruskall-Wallis H tests were performed (Table 9). The p-values stemming from these tests indicate that all the adjusted variables (endpoint, feature choice, model architecture) had a statistically significant difference in means. Of these variables the feature representation method had the lowest p-value, followed by toxicological endpoint and then model type choice. In this context, a lower p-value suggests a higher probability that the factor directly affects model performance. As a result, it is evident that all tested variables have a large impact on how well a model performs, meaning that at each level of model development it is integral to carefully choose which method to apply based on the context.

4.5 Limitations & Future directions

Outside of the individual limitations outlined for each independent endpoint, feature representation method and ML model type, there were multiple unresolved issues across this project. These encountered roadblocks either fell outside the scope of the project or required time and resource access unavailable at the time of creation.

4.5.1 Class imbalance

Addressed with an undersampling methodology, all imbalanced datasets were adjusted to a 5:6 class ratio. This, however, induces a large loss of data in each affected dataset (Supplementary Table 2, 3), which results in less data for each model to be trained and tested on. As having an ample sample size is crucial for training predictive models, datasets that have incurred large data loss due to class imbalance are innately disadvantaged due to the sample size disparity.

There are two alternative solutions to this issue rather than undersampling: oversampling and procuring additional data[32]. The former involves duplicating samples of a less populous class to reach a desired ratio, alleviating the issue of too small a sample size occurring for model training. However, this may also result in overfitting as multiple instances of each datapoint in the less frequent class are utilised for data analysis. On the other hand, by sourcing additional data for endpoints with limited samples the class ratio can be equalised. In this case, the classification for whether a compound is toxic to an endpoint will need to be determined to standardise data across differing datasets and databases but would allow for a roughly equal class ratio for prediction. In future work, by implementing these techniques the presence of class discrepancy can be minimised across all endpoints.

4.5.2 Feature representation expansion

Across the course of this project, five differing feature representations were explored. In particular, the DrugBank polypeptide feature representation can be further explored by analysing qualities attributed to each drug target and by correlating this information to its original compound. By finding intricacies of each drug's targets such as their Gene Ontologies, further interaction-based information can be gleaned regarding each drug. This technique has seen usage in toxicological profiling using neural networks, achieving a peak accuracy of 0.88 for acute toxicity cases[90]. By implementing more extensive feature extraction methods like this, potentially better model efficacy can be achieved.

4.5.3 Package creation

By compiling each of the most robust (MCC > 0.65) models into a Python package, these tools can readily be applied and compounded upon by outside collaborators. Currently, to apply and utilise the created models in this project, a degree of coding knowledge is required to rerun and compile the developed scripts. By moving strong models into a package, these models can be utilised by a broader audience, potentially allowing for greater accessibility for individuals attempting to assess drug toxicity endpoints.

5 CONCLUSION

Excess drug toxicity can be attributed to a large volume of candidate knockouts in the preclinical and clinical trial phases of drug development. Toxicity testing at this stage is highly costly and time intensive. To alleviate this problem, a series of predictive models were developed that detect a drug's potential toxicity from its core features with a peak MCC of 0.83, verified with K-fold cross validation. In doing this, toxic compounds can be identified early in drug development.

To develop these models, binary toxicity data concerning 17,773 molecules across 22 toxicological endpoints were sourced from the 2023 Toxric database. For each of these datasets, structural and interaction-based molecular feature representations were extracted. These feature extraction methods were applied with a multitude of machine learning architectures to culminate in 440 models, each with a unique endpoint-representation-model type combination. In doing this, synergies within these permutations can be uncovered and applied towards toxicity prediction.

In analysing these combinations, this project aimed to determine the impact of endpoint, feature extraction and model architecture choice onto the prediction of toxicity. Through the application of Kruskall-Wallis H-tests and its corresponding p-values, it was readily evident that each of these choices has a profound impact on model quality. Of these three factors, the feature extraction technique seemed to have the highest difference in means, followed by the toxicological endpoint, and finally model architecture method. This underscores the importance of each step within the model development pipeline in the creation of robust predictors.

Amongst the developed models, 10 scored above the quality threshold of 0.65 MCC. These models are robust and relative in performance to the strong models found in existing literature. By applying these models as a diagnosis tool in the early stages of drug development, potentially toxic compounds can be identified quickly and either confirmed with assay work, or outright removed. In doing this, time and resources can be saved that would have otherwise been invested into a toxic compound

6 REFERENCES

1. Hughes, J.P., Rees, S., Kalindjian, S.B., and Philpott, K.L. (2011). Principles of early drug discovery. Br J Pharmacol *162*, 1239-1249. 10.1111/j.1476-5381.2010.01127.x.
2. Guengerich, F.P. (2011). Mechanisms of drug toxicity and relevance to pharmaceutical development. Drug Metab Pharmacokinet *26*, 3-14. 10.2133/dmpk.dmpk-10-rv-062.
3. Sun, D., Gao, W., Hu, H., and Zhou, S. (2022). Why 90% of clinical drug development fails and how to improve it? Acta Pharm Sin B *12*, 3049-3062. 10.1016/j.apsb.2022.02.002.
4. Ashenden, S.K. (2021). Chapter 6 - Lead optimization. In The Era of Artificial Intelligence, Machine Learning, and Data Science in the Pharmaceutical Industry, S.K. Ashenden, ed. (Academic Press), pp. 103-117. https://doi.org/10.1016/B978-0-12-820045-2.00007-6.
5. Huang, W., Percie du Sert, N., Vollert, J., and Rice, A.S.C. (2020). General Principles of Preclinical Study Design. Handb Exp Pharmacol *257*, 55-69. 10.1007/164_2019_277.
6. Patton, E.E., Zon, L.I., and Langenau, D.M. (2021). Zebrafish disease models in drug discovery: from preclinical modelling to clinical trials. Nat Rev Drug Discov *20*, 611-628. 10.1038/s41573-021-00210-8.
7. Piantadosi, S. (2024). Clinical trials: a methodologic perspective (John Wiley & Sons).
8. Parasuraman, S. (2011). Toxicological screening. J Pharmacol Pharmacother *2*, 74-79. 10.4103/0976-500x.81895.
9. Vearrier., M.E.M.A.A.R.J.C.D. (2024). Ibuprofen Toxicity. Treasure island. StatPearls.
10. Hasan, M.R., Alsaiari, A.A., Fakhurji, B.Z., Molla, M.H.R., Asseri, A.H., Sumon, M.A.A., Park, M.N., Ahammad, F., and Kim, B. (2022). Application of Mathematical Modeling and Computational Tools in the Modern Drug Design and Development Process. Molecules *27*. 10.3390/molecules27134169.
11. Creanza, T.M., Delre, P., Ancona, N., Lentini, G., Saviano, M., and Mangiatordi, G.F. (2021). Structure-Based Prediction of hERG-Related Cardiotoxicity: A Benchmark Study. J Chem Inf Model *61*, 4758-4770. 10.1021/acs.jcim.1c00744.
12. Srisongkram, T. (2024). DeepRA: A novel deep learning-read-across framework and its application in non-sugar sweeteners mutagenicity prediction. Computers in Biology and Medicine *178*, 108731. https://doi.org/10.1016/j.compbiomed.2024.108731.
13. Huang, R., Xia, M., Nguyen, D.-T., Zhao, T., Sakamuru, S., Zhao, J., Shahane, S., Rossoshek, A., and Simeonov, A. (2016). Tox21Challenge to Build Predictive Models of Nuclear Receptor and Stress Response Pathways as Mediated by Exposure to Environmental Chemicals and Drugs. Frontiers in Environmental Science *3*. 10.3389/fenvs.2015.00085.
14. Gupta, R., Srivastava, D., Sahu, M., Tiwari, S., Ambasta, R.K., and Kumar, P. (2021). Artificial intelligence to deep learning: machine intelligence approach for drug discovery. Mol Divers *25*, 1315-1360. 10.1007/s11030-021-10217-3.
15. Fradkov, A.L. (2020). Early History of Machine Learning. IFAC-PapersOnLine *53*, 1385-1390. https://doi.org/10.1016/j.ifacol.2020.12.1888.

16. Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review *65*, 386.
17. Chavan, S., Friedman, R., and Nicholls, I.A. (2015). Acute Toxicity-Supported Chronic Toxicity Prediction: A k-Nearest Neighbor Coupled Read-Across Strategy. Int J Mol Sci *16*, 11659-11677. 10.3390/ijms160511659.
18. Lysenko, A., Sharma, A., Boroevich, K.A., and Tsunoda, T. (2018). An integrative machine learning approach for prediction of toxicity-related drug safety. Life Sci Alliance *1*, e201800098. 10.26508/lsa.201800098.
19. Masoudi-Sobhanzadeh, Y., Omidi, Y., Amanlou, M., and Masoudi-Nejad, A. (2020). Drug databases and their contributions to drug repurposing. Genomics *112*, 1087-1095. https://doi.org/10.1016/j.ygeno.2019.06.021.
20. Kaufman, L., Gore, K., and Zandee, J.C. (2017). Data Standardization, Pharmaceutical Drug Development, and the 3Rs. ILAR Journal *57*, 109-119. 10.1093/ilar/ilw030.
21. Wu, L., Yan, B., Han, J., Li, R., Xiao, J., He, S., and Bo, X. (2023). TOXRIC: a comprehensive database of toxicological data and benchmarks. Nucleic Acids Res *51*, D1432-D1445. 10.1093/nar/gkac1074.
22. Wu, Z., Jiang, D., Wang, J., Hsieh, C.-Y., Cao, D., and Hou, T. (2021). Mining Toxicity Information from Large Amounts of Toxicity Data. Journal of Medicinal Chemistry *64*, 6924-6936. 10.1021/acs.jmedchem.1c00421.
23. Judson, R., Richard, A., Dix, D., Houck, K., Elloumi, F., Martin, M., Cathey, T., Transue, T.R., Spencer, R., and Wolf, M. (2008). ACToR--Aggregated Computational Toxicology Resource. Toxicol Appl Pharmacol *233*, 7-13. 10.1016/j.taap.2007.12.037.
24. Hu, Y., Ren, Q., Liu, X., Gao, L., Xiao, L., and Yu, W. (2023). In Silico Prediction of Human Organ Toxicity via Artificial Intelligence Methods. Chemical Research in Toxicology *36*, 1044-1054. 10.1021/acs.chemrestox.2c00411.
25. Gini, G. (2016). QSAR Methods. In In Silico Methods for Predicting Drug Toxicity, E. Benfenati, ed. (Springer New York), pp. 1-20. 10.1007/978-1-4939-3609-0_1.
26. Wu, Z., Jiang, D., Hsieh, C.-Y., Chen, G., Liao, B., Cao, D., and Hou, T. (2021). Hyperbolic relational graph convolution networks plus: a simple but highly efficient QSAR-modeling method. Briefings in Bioinformatics *22*. 10.1093/bib/bbab112.
27. Harnik, Y., and Milo, A. (2024). A focus on molecular representation learning for the prediction of chemical properties. Chem Sci *15*, 5052-5055. 10.1039/d4sc90043j.
28. Capecchi, A., Probst, D., and Reymond, J.-L. (2020). One molecular fingerprint to rule them all: drugs, biomolecules, and the metabolome. Journal of Cheminformatics *12*, 43. 10.1186/s13321-020-00445-4.
29. Fuzi, B., Mathai, N., Kirchmair, J., and Ecker, G.F. (2023). Toxicity prediction using target, interactome, and pathway profiles as descriptors. Toxicol Lett *381*, 20-26. 10.1016/j.toxlet.2023.04.005.
30. Dang, L.H., Dung, N.T., Quang, L.X., Hung, L.Q., Le, N.H., Le, N.T.N., Diem, N.T., Nga, N.T.T., Hung, S.H., and Le, N.Q.K. (2021). Machine Learning-Based Prediction of Drug-Drug Interactions for Histamine Antagonist Using Hybrid Chemical Features. Cells *10*. 10.3390/cells10113092.

31. Chen, L., Lu, J., Luo, X., and Feng, K.Y. (2014). Prediction of drug target groups based on chemical-chemical similarities and chemical-chemical/protein connections. Biochim Biophys Acta *1844*, 207-213. 10.1016/j.bbapap.2013.05.021.
32. Datasets: Imbalanced datasets. (2024). https://developers.google.com/machine-learning/crash-course/overfitting/imbalanced-datasets.
33. Mansouri, K., Cariello, N.F., Korotcov, A., Tkachenko, V., Grulke, C.M., Sprankle, C.S., Allen, D., Casey, W.M., Kleinstreuer, N.C., and Williams, A.J. (2019). Open-source QSAR models for pKa prediction using multiple machine learning approaches. J Cheminform *11*, 60. 10.1186/s13321-019-0384-1.
34. Wu, Z., Xian, Z., Ma, W., Liu, Q., Huang, X., Xiong, B., He, S., and Zhang, W. (2021). Artificial neural network approach for predicting blood brain barrier permeability based on a group contribution method. Computer Methods and Programs in Biomedicine *200*, 105943. https://doi.org/10.1016/j.cmpb.2021.105943.
35. Courville, I.G.a.Y.B.a.A. (2016). Deep Learning (MIT Press).
36. Iqbal, S., A, N.Q., Li, J., and Mahmood, T. (2023). On the Analyses of Medical Images Using Traditional Machine Learning Techniques and Convolutional Neural Networks. Arch Comput Methods Eng *30*, 3173-3233. 10.1007/s11831-023-09899-9.
37. Valkenborg, D., Rousseau, A.-J., Geubbelmans, M., and Burzykowski, T. (2023). Support vector machines. American Journal of Orthodontics and Dentofacial Orthopedics *164*, 754-757. https://doi.org/10.1016/j.ajodo.2023.08.003.
38. Cortes, C. (1995). Support-Vector Networks. Machine Learning.
39. Kumar, B., and Gupta, D. (2021). Universum based Lagrangian twin bounded support vector machine to classify EEG signals. Computer Methods and Programs in Biomedicine *208*, 106244. https://doi.org/10.1016/j.cmpb.2021.106244.
40. Sun, H., Shahane, S., Xia, M., Austin, C.P., and Huang, R. (2012). Structure Based Model for the Prediction of Phospholipidosis Induction Potential of Small Molecules. Journal of Chemical Information and Modeling *52*, 1798-1805. 10.1021/ci3001875.
41. Ho, T.K. (1995). Random decision forests. (IEEE), pp. 278-282.
42. Shi, G., Liu, G., Gao, Q., Zhang, S., Wang, Q., Wu, L., He, P., and Yu, Q. (2023). A random forest algorithm-based prediction model for moderate to severe acute postoperative pain after orthopedic surgery under general anesthesia. BMC Anesthesiol *23*, 361. 10.1186/s12871-023-02328-1.
43. Trinh, T.X., Seo, M., Yoon, T.H., and Kim, J. (2022). Developing random forest based QSAR models for predicting the mixture toxicity of TiO2 based nano-mixtures to Daphnia magna. NanoImpact *25*, 100383. https://doi.org/10.1016/j.impact.2022.100383.
44. Toth, R., Schiffmann, H., Hube-Magg, C., Büscheck, F., Höflmayer, D., Weidemann, S., Lebok, P., Fraune, C., Minner, S., Schlomm, T., et al. (2019). Random forest-based modelling to detect biomarkers for prostate cancer progression. Clin Epigenetics *11*, 148. 10.1186/s13148-019-0736-8.
45. Silverman, B.W., and Jones, M.C. (1989). E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951). International Statistical Review / Revue Internationale de Statistique *57*, 233-238. 10.2307/1403796.

46. Peterson, L.E. (2009). K-nearest neighbor. Scholarpedia *4*, 1883.
47. Laaksonen, J., and Oja, E. (1996). Classification with learning k-nearest neighbors. 3-6 June 1996. pp. 1480-1483 vol.1483.
48. DiFranzo, A., Sheridan, R.P., Liaw, A., and Tudor, M. (2020). Nearest Neighbor Gaussian Process for Quantitative Structure–Activity Relationships. Journal of Chemical Information and Modeling *60*, 4653-4663. 10.1021/acs.jcim.0c00678.
49. Tao, L., Chen, G., and Li, Y. (2021). Machine learning discovery of high-temperature polymers. Patterns (N Y) *2*, 100225. 10.1016/j.patter.2021.100225.
50. Liang, Y., Huangfu, X., Huang, R., Han, Z., Wu, S., Wang, J., Long, X., Ma, J., and He, Q. (2024). Machine learning for predicting halogen radical reactivity toward aqueous organic chemicals. J Hazard Mater *472*, 134501. 10.1016/j.jhazmat.2024.134501.
51. RDKit: Open-source cheminformatics (2024).
52. Cao, D.S., Xu, Q.S., Hu, Q.N., and Liang, Y.Z. (2013). ChemoPy: freely available python package for computational biology and chemoinformatics. Bioinformatics *29*, 1092-1094. 10.1093/bioinformatics/btt105.
53. Dong, J., Yao, Z.J., Zhu, M.F., Wang, N.N., Lu, B., Chen, A.F., Lu, A.P., Miao, H., Zeng, W.B., and Cao, D.S. (2017). ChemSAR: an online pipelining platform for molecular SAR modeling. J Cheminform *9*, 27. 10.1186/s13321-017-0215-1.
54. Wang, Y.T., Yang, Z.X., Piao, Z.H., Xu, X.J., Yu, J.H., and Zhang, Y.H. (2021). Prediction of flavor and retention index for compounds in beer depending on molecular structure using a machine learning method. RSC Adv *11*, 36942-36950. 10.1039/d1ra06551c.
55. Yap, C.W. (2011). PaDEL-descriptor: an open source software to calculate molecular descriptors and fingerprints. J Comput Chem *32*, 1466-1474. 10.1002/jcc.21707.
56. Mswahili, M.E., Martin, G.L., Woo, J., Choi, G.J., and Jeong, Y.S. (2021). Antimalarial Drug Predictions Using Molecular Descriptors and Machine Learning against Plasmodium Falciparum. Biomolecules *11*. 10.3390/biom11121750.
57. Allinger, N.L. (1977). Conformational analysis. 130. MM2. A hydrocarbon force field utilizing V1 and V2 torsional terms. Journal of the American Chemical Society *99*, 8127-8134. 10.1021/ja00467a001.
58. Chambers, J., Davies, M., Gaulton, A., Hersey, A., Velankar, S., Petryszak, R., Hastings, J., Bellis, L., McGlinchey, S., and Overington, J.P. (2013). UniChem: a unified chemical structure cross-referencing and identifier tracking system. Journal of Cheminformatics *5*, 3. 10.1186/1758-2946-5-3.
59. Gadzicki, K., Khamsehashari, R., and Zetzsche, C. (2020). Early vs Late Fusion in Multimodal Convolutional Neural Networks. 6-9 July 2020. pp. 1-6.
60. Pedregosa, F.a.V., G. and Gramfort, A. and Michel, V., and Thirion, B.a.G., O. and Blondel, M. and Prettenhofer, P., and Weiss, R.a.D., V. and Vanderplas, J. and Passos, A. and, and Cournapeau, D.a.B., M. and Perrot, M. and Duchesnay, E. (2011). Scikit-learn: Machine Learning in {P}ython. Journal of Machine Learning Research *12*, 2825--2830.
61. Chollet, F.a.o. (2015). Keras.
62. Szaleniec, M. (2012). Prediction of enzyme activity with neural network models based on electronic and geometrical features of substrates. Pharmacol Rep *64*, 761-781. 10.1016/s1734-1140(12)70873-3.

63. Stathakis, D. (2009). How many hidden layers and nodes? International Journal of Remote Sensing *30*, 2133-2147.
64. Huang, G.-B. (2003). Learning capability and storage capacity of two-hidden-layer feedforward networks. IEEE transactions on neural networks *14*, 274-281. 10.1109/tnn.2003.809401.
65. Diederik, P.K. (2014). Adam: A method for stochastic optimization. (No Title).
66. Chicco, D., and Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics *21*, 6. 10.1186/s12864-019-6413-7.
67. Chicco, D., and Jurman, G. (2023). The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification. BioData Min *16*, 4. 10.1186/s13040-023-00322-4.
68. Yang, X., Zhang, Z., Li, Q., and Cai, Y. (2021). Quantitative structure-activity relationship models for genotoxicity prediction based on combination evaluation strategies for toxicological alternative experiments. Sci Rep *11*, 8030. 10.1038/s41598-021-87035-y.
69. Poldrack, R.A., Huckins, G., and Varoquaux, G. (2020). Establishment of Best Practices for Evidence for Prediction: A Review. JAMA Psychiatry *77*, 534-540. 10.1001/jamapsychiatry.2019.3671.
70. Hunter, J.D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering *9*, 90-95. 10.1109/MCSE.2007.55.
71. MacFarland, T.W., and Yates, J.M. (2016). Kruskal–Wallis H-Test for Oneway Analysis of Variance (ANOVA) by Ranks. In Introduction to Nonparametric Statistics for the Biological Sciences Using R, T.W. MacFarland, and J.M. Yates, eds. (Springer International Publishing), pp. 177-211. 10.1007/978-3-319-30634-6_6.
72. Pauli Virtanen, R.G., Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E.A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods *17(3)*, 261-272. 10.1038/s41592-019-0686-2.
73. Hoofnagle, J.H. (2013). Chapter 40 - LiverTox: A Website on Drug-Induced Liver Injury. In Drug-Induced Liver Disease (Third Edition), N. Kaplowitz, and L.D. DeLeve, eds. (Academic Press), pp. 725-732. https://doi.org/10.1016/B978-0-12-387817-5.00040-6.
74. Richard, A.M., Huang, R., Waidyanatha, S., Shinn, P., Collins, B.J., Thillainadarajah, I., Grulke, C.M., Williams, A.J., Lougee, R.R., Judson, R.S., et al. (2021). The Tox21 10K Compound Library: Collaborative Chemistry Advancing Toxicology. Chemical Research in Toxicology *34*, 189-216. 10.1021/acs.chemrestox.0c00264.

75. Liu, L., Yang, H., Cai, Y., Cao, Q., Sun, L., Wang, Z., Li, W., Liu, G., Lee, P.W., and Tang, Y. (2019). In silico prediction of chemical aquatic toxicity for marine crustaceans via machine learning. Toxicol Res (Camb) *8*, 341-352. 10.1039/c8tx00331a.
76. Chen, M., Suzuki, A., Thakkar, S., Yu, K., Hu, C., and Tong, W. (2016). DILIrank: the largest reference drug list ranked by the risk for developing drug-induced liver injury in humans. Drug Discov Today *21*, 648-653. 10.1016/j.drudis.2016.02.015.
77. Thakkar, S., chen, M., Fang, H., Liu, Z., Roberts, R., and Tong, W. (2018). The Liver Toxicity Knowledge Base (LKTB) and drug-induced liver injury (DILI) classification for assessment of human liver injury. Expert Review of Gastroenterology & Hepatology *12*, 31-38. 10.1080/17474124.2018.1383154.
78. Hong, H., Thakkar, S., Chen, M., and Tong, W. (2017). Development of Decision Forest Models for Prediction of Drug-Induced Liver Injury in Humans Using A Large Set of FDA-approved Drugs. Sci Rep *7*, 17311. 10.1038/s41598-017-17701-7.
79. Kelleci Çelik, F., and Karaduman, G. (2023). Machine Learning-Based Prediction of Drug-Induced Hepatotoxicity: An OvA-QSTR Approach. Journal of Chemical Information and Modeling *63*, 4602-4614. 10.1021/acs.jcim.3c00687.
80. Gaulton, A., Bellis, L.J., Bento, A.P., Chambers, J., Davies, M., Hersey, A., Light, Y., McGlinchey, S., Michalovich, D., Al-Lazikani, B., and Overington, J.P. (2012). ChEMBL: a large-scale bioactivity database for drug discovery. Nucleic Acids Res *40*, D1100-1107. 10.1093/nar/gkr777.
81. Wexler, P. (2001). TOXNET: An evolving web resource for toxicology and environmental health information. Toxicology *157*, 3-10. https://doi.org/10.1016/S0300-483X(00)00337-1.
82. Pradeep, P., Judson, R., DeMarini, D.M., Keshava, N., Martin, T.M., Dean, J., Gibbons, C.F., Simha, A., Warren, S.H., Gwinn, M.R., and Patlewicz, G. (2021). Evaluation of Existing QSAR Models and Structural Alerts and Development of New Ensemble Models for Genotoxicity Using a Newly Compiled Experimental Dataset. Comput Toxicol *18*. 10.1016/j.comtox.2021.100167.
83. Zeiger, E. (2019). The test that changed the world: The Ames test and the regulation of chemicals. Mutation Research/Genetic Toxicology and Environmental Mutagenesis *841*, 43-48. https://doi.org/10.1016/j.mrgentox.2019.05.007.
84. Fitzpatrick, R.B. (2008). CPDB: Carcinogenic Potency Database. Med Ref Serv Q *27*, 303-311. 10.1080/02763860802198895.
85. Zagidullin, B., Wang, Z., Guan, Y., Pitkänen, E., and Tang, J. (2021). Comparative analysis of molecular fingerprints in prediction of drug combination effects. Briefings in Bioinformatics *22*. 10.1093/bib/bbab291.
86. Gao, K., Nguyen, D.D., Sresht, V., Mathiowetz, A.M., Tu, M., and Wei, G.W. (2020). Are 2D fingerprints still valuable for drug discovery? Phys Chem Chem Phys *22*, 8373-8390. 10.1039/d0cp00305k.
87. Cao, H., Peng, J., Zhou, Z., Sun, Y., Wang, Y., and Liang, Y. (2022). Insight into the defluorination ability of per- and polyfluoroalkyl substances based on machine learning and quantum chemical computations. Science of The Total Environment *807*, 151018. https://doi.org/10.1016/j.scitotenv.2021.151018.

88. Ramezan, C.A., Warner, T.A., Maxwell, A.E., and Price, B.S. (2021). Effects of Training Set Size on Supervised Machine-Learning Land-Cover Classification of Large-Area High-Resolution Remotely Sensed Data. Remote Sensing *13*, 368.
89. Cheng, G., and Han, J. (2016). A survey on object detection in optical remote sensing images. ISPRS Journal of Photogrammetry and Remote Sensing *117*, 11-28. https://doi.org/10.1016/j.isprsjprs.2016.03.014.
90. Jain, A., and Kihara, D. (2019). NNTox: Gene Ontology-Based Protein Toxicity Prediction Using Neural Network. Sci Rep *9*, 17923. 10.1038/s41598-019-54405-6.

7 APPENDIX

7.1 Supplementary Tables

Supplementary Table 1: Toxric endpoints, datasets & sourcing

Data taken from Toxric consisted of eight primary toxicity endpoints, two of which contained multiple datasets. The 'Cardiotoxicity' endpoints represent the molecule's IC50 threshold (1, 5, 10, 30) in µM. On the other hand, the 'Endocrine disruption' datasets are differing targets that when disrupted negatively impact the endocrine system.

Toxicity Type	Dataset endpoint	Molecule Count	Source(s)
Mutagenicity	Ames mutagenicity	7485	Literature
Respiratory Toxicity		1348	Literature
Hepatotoxicity		2889	Liver Toxicity Knowledge Base (LTKB), Drug Induced Liver Injury Rank (DILIRank), Livertox, Literature
Carcinogenicity		1021	(Carcinogenicity Potency Database) CPDB
Developmental Toxicity Reproductive Toxicity		218 156	Aggregated Computational Toxicity Resource (ACToR)

Cardiotoxicity	Cardiotoxicity-1	1547	Literature
	Cardiotoxicity-5		
	Cardiotoxicity-10		
	Cardiotoxicity-30		
Endocrine Disruption	Nuclear receptor – androgen receptor (NR-AR)	7118	Toxicology in the 21st Century (Tox21)
	Nuclear receptor – androgen receptor – ligand-binding domain (NR-AR-LBD)	6629	
	Nuclear receptor – aryl hydrocarbon acceptor (NR-AhR)	6410	
	Nuclear receptor - aromatase (NR-aromatase)	5707	
	Nuclear receptor – estrogen receptor (NR-ER)	6068	
	Nuclear receptor – estrogen receptor – ligand-binding domain (NR-ER-LBD)	6820	
	Nuclear receptor – peroxisome proliferator-activated receptor gamma (NR-PPAR-gamma)	6322	
	stress response - ATPase family AAA Domain Containing 5	6931	

	(SR-ATAD5)		
	Stress response – heat shock sequence (SR-HSE)	6338	
	Stress response – mitochondrial membrane potential (SR-MMP)	5700	
	Stress response – p53 pathway (SR-p53)	6636	
	Stress response – antioxidant response element (SR-ARE)	5712	

Supplementary Table 2: Toxric datasets before and after accounting for class imbalance

Each of the Toxric endpoint datasets that contained a class imbalance ratio greater than 1:4 had undersampling conducted until a 5:6 ratio was reached.

	Before balancing		After balancing	
Dataset	Sample count	Toxic/Non-toxic molecules	Sample count	Toxic/Non-toxic molecules
Ames Mutagenicity	7485	4196/3289	-	-
Carcinogenicity	1021	504/517	-	-
Cardiotoxicity-1	1547	349/1198	-	-
Cardiotixicity-5	1547	715/832	-	-
Cardiotoxicity-10	1547	949/598	-	-
Cardiotoxicity-30	1547	1284/263	688	425/263
Developmental toxicity	218	189/29	69	40/29
Hepatotoxicity	2889	1465/1424	-	-
NR-AhR	6410	738/5672	1674	738/936
NR-AR	7118	303/6815	625	303/322
NR-AR-LBD	6629	234/6295	480	234/246
NR-aromatase	5707	291/5416	604	291/313
NR-ER	6068	779/5289	1836	779/1057

NR-ER-LBD	6820	347/6473	726	347/379
NR-PPAR-gamma	6322	179/6143	365	169/186
Reproductive toxicity	156	138/18	41	23/18
Respiratory toxicity	1348	819/529	-	-
SR-ARE	5712	904/4808	2119	904/1215
SR-ATAD5	6931	253/6678	516	253/263
SR-HSE	6338	364/5794	760	364/396
SR-MMP	5700	895/4805	2106	895/1211
SR-p53	6636	406/6230	847	406/441

Supplementary table 3: DrugBank overlap datasets before and after class imbalance removal

Due to the method of DrugBank target extraction, many molecules found in the Toxric datasets were not found in the resulting feature representation. As a result, all datasets lost most of their compounds which was further reduced with the implementation of class imbalance. In particular, the 'Respiratory toxicity', 'Developmental toxicity' and 'Reproductive toxicity' datasets did not have enough samples for k-fold validation and as a result were removed with all their resulting models being set to 0 MCC.

	Before balancing		**After balancing**	
Dataset	Sample count	Toxic/Non-toxic molecules	Sample count	Toxic/Non-toxic molecules
Ames Mutagenicity	390	91/299	390	91/299
Carcinogenicity	192	58/134	192	58/134
Cardiotoxicity-1	266	39/227	97	39/58
Cardiotixicity-5	266	103/163	266	103/163
Cardiotoxicity-10	266	103/163	266	103/163
Cardiotoxicity-30	266	219/47	120	73/47
Developmental toxicity	47	43/4	9	5/4
Hepatotoxicity	820	457/363	820	457/363
NR-AhR	1117	110/1007	228	110/118
NR-AR	1258	117/1141	254	117/137

NR-AR-LBD	1154	85/1069	182	85/97
NR-aromatase	1016	76/940	164	76/88
NR-ER	1079	180/899	488	180/308
NR-ER-LBD	1207	90/1117	191	90/101
NR-PPAR-gamma	1073	32/1041	66	32/34
Reproductive toxicity	25	20/5	25	20/5
Respiratory toxicity	69	27/42	69	27/42
SR-ARE	967	184/783	450	184/266
SR-ATAD5	1211	49/1162	104	49/53
SR-HSE	1116	60/1056	120	60/68
SR-MMP	988	158/830	374	158/216
SR-p53	1162	94/1068	208	94/114