Machine learning in predictive toxicology: recent applications and future directions for classification models

Marcus W. H. Wang†, Jonathan M. Goodman†*, Timothy E. H. Allen†‡

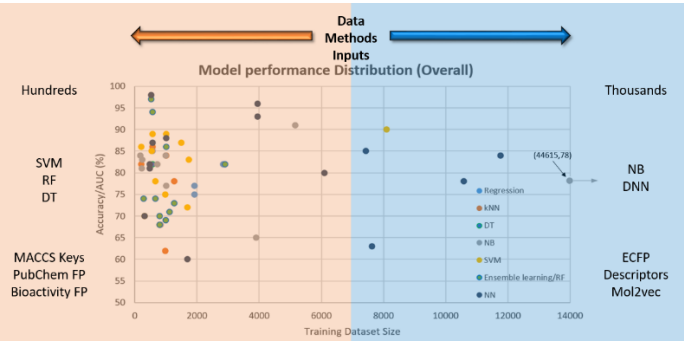
† Centre for Molecular Informatics, Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge, CB2 1EW, United Kingdom

‡ MRC Toxicology Unit, University of Cambridge, Hodgkin Building, Lancaster Road, Leicester, LE1 7HB, United Kingdom

Machine Learning, Artificial Intelligence, Predictive Toxicology, Computational Toxicology

ABSTRACT: In recent times, machine learning has become increasingly prominent in predictive toxicology as it has shifted from *in vivo* studies towards *in silico* studies. Currently, *in vitro* methods together with other computational methods such as QSAR modelling and ADME calculations are being used. An overview of machine learning and its applications in predictive toxicology is presented here, including support vector machines (SVMs), Random forest (RF) and decision trees (DTs), neural networks, regression models, Naive Bayes, k-nearest neighbours (kNNs), and ensemble learning. The recent successes of these machine learning methods in predictive toxicology are summarized and a comparison of some models used in predictive toxicology is presented. In predictive toxicology SVMs, RF and DTs are the dominant machine learning methods due to the characteristics of the data available. Lastly, this review describes the current challenges facing the use of machine learning in predictive toxicology and offers insights into the possible areas of improvement in the field.

For Table of Contents Only



INTRODUCTION

Machine learning is a recent field that has advanced computational chemistry with numerous applications such as drug discovery, cheminformatics, and predictive toxicology.[1-21] Machine learning generally involves building a model, training the model, performing validation, repeating training and validation until a suitable model is obtained, and finally testing the model on data not previously exposed to the model (Figure 1). The goal of a machine learning model is to pick out patterns from the input data, or generalize these data, and apply the results to unknown test data.[3–5,15–19,21]

These models can be used for predictive toxicology, which is a field that revolves around *in silico* predictions of *in vivo* toxicological effects, such as for drug candidates or drugs,[5,9,10,15,22–26] consumer products,[27] agrochemicals,[28] and foods.[29,30] Historically, the toxicity of new chemicals was determined through *in vivo* studies, pre-clinical trials, and clinical trials.[22–26,31,32] The toxicity of a drug or drug candidate should be determined before it reaches the market or before any clinical trials are performed, where there is a risk of causing severe adverse effects to humans.[22,23,25,26,31–33] However, determining the toxicity of new compounds is challenging due to a wide variety of potential metabolic products,[33,34] idiosyncratic effects which only occur in small parts of the population,[23,26,32,33] and the general complexity of

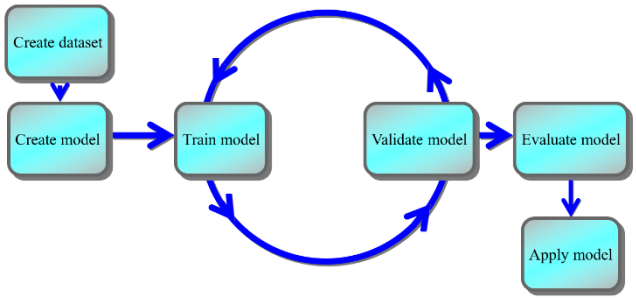


Figure 1. Outline of machine learning.

in vivo systems. These difficulties in determining toxicity have resulted in drug withdrawals or drug candidates being terminated at the pre-clinical or clinical phase.[22–26,31–36] Generally, the drug attrition rate is reported to be abozut 90% to 95% for Phase I trials.[30,37] Such a high failure rate only contributes to the high cost of drug development, which totalled $2.59 billion in the USA in 2014, and is expected to continue increasing in the future.[38] For these reasons, and due to ethical concerns and regulatory advances,[39–41] there has been a shift towards the use of *in silico* methods for predicting the toxicity of compounds.[30,42–46]

In silico methods for predictive toxicology include algorithms such as machine learning, QSAR, expert-based systems, and read-across.[15,16,19,47–49] We will focus on machine learning in predictive toxicology, and cover a variety of toxicity endpoints, of which the main types are hepatotoxicity, carcinogenicity/genotoxicity/mutagenicity, and cardiotoxicity. We also concentrate on classification algorithms, which treat toxicity as an active/inactive question, rather than on regression models, which attempt to make quantitative predictions of toxicity. To predict toxicity for drugs and drug candidates, several software packages have been developed. including Derek and the OECD Toolbox.[50–53]. Machine learning models are generally treated as black boxes, and the decisions made by them are sometimes unclear, or at least unexplained. In contrast, some mechanistic QSAR models and expert systems can be understood and interpreted, although this is not true for all of them. Machine learning also has the advantage that it is often able to handle complex problems more effectively and scales well to many different tasks, as evidenced by the successes so far in predictive toxicology.[5,16–18,20,21,54–56]

Some people consider machine learning algorithms as a type of QSAR and some consider it to be different. In general, QSAR modelling refers to using a structure-activity relationship to model a quantitative prediction of a label. On the other hand, machine learning refers to using a statistical technique to generalize the data, and obtaining predictions based on the model. In machine learning, structure-activity relationships can be used to model the data, which might give rise to confusion between the two types of modelling techniques. It is also noted that with machine learning, there are other features that can be used to model the data, which need not be the structure. Similarly, the activity of the compound need not be the predicted label and other labels such as toxicity endpoints can be used.

In predictive toxicology, common databases used to build datasets for machine learning models include ChEMBL, ToxCast, and PubChem.[57–64] These databases contain data for different groups of compounds. For example, ChEMBL has data for drug-like compounds while ToxCast focuses more on industrial chemicals.[59–63] Other sources of data include results provided by pharmaceutical companies, and data that can be extracted from published papers or the public domain.[65–71] The data are subsequently processed for suitability as model input. For example, missing, invalid, or unnecessary data would usually be removed. Another part of data processing is related to working with imbalanced data which is common in machine learning.[70-81] Several methods have been employed to balance the classes in the dataset to train good quality models.[70-81] These methods, which usually involve oversampling, undersampling, or a combination of both are described in the literature.[70-81] The checking and processing of the raw data, as well as checking the quality of end-point data, are essential steps that are often overlooked, which can result in poor models being developed.

Once the dataset has been prepared, the next step is to build a machine learning model. Many papers have been published on the models used in machine learning and its use in predictive toxicology.[3,10,13,15-21,47,54,82–85] In general, these models can be classified into three categories: supervised learning, unsupervised learning and semi-supervised learning (Figure 2).

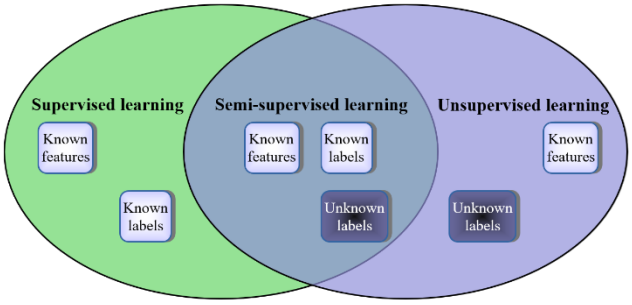


Figure 2. Different categories of machine learning models.

Supervised learning refers to models that are trained on a dataset containing known inputs (features) and outputs (labels).[86–90] The model predicts the labels while the accuracy is defined as the difference between the predicted labels and the experimental labels. These models are usually used for classification purposes. Unsupervised learning refers to models that are trained on a dataset containing known features but unknown labels, and this is commonly used for clustering or pattern recognition.[8,88,91–93] Lastly, semi-supervised learning aims to improve model performance compared to the former two model categories by making use of both labelled and unlabelled data.[90,94–101]

In predictive toxicology, supervised learning is commonly used as it can classify the input data into different classes (binary or multi-class classification) or labels (multi-label classification).[19,46,54,90,102,103] For example, a supervised learning model can be used to predict reproductive toxicity given the fingerprint of a compound.[104] Supervised learning can also be used for regression-based tasks, such as for predicting the quantitative value associated with compound toxicities.[105] In contrast, unsupervised learning and semi-supervised learning are less commonly used in predictive toxicology.

Here we will focus on supervised machine learning in predictive toxicology, with several model types that have been used in predictive toxicology being analyzed. These model types will be introduced in order of increasing complexity:

1. Regression models
2. k-nearest neighbours (kNNs)
3. Decision trees (DTs)
4. Naive Bayes (NB)
5. Support vector machines (SVMs)
6. Random forest (RF) and ensemble learning
7. Neural networks (Artificial neural networks (ANNs), deep neural networks (DNNs), Convolutional neural networks (CNNs))

Unlike related reviews on machine learning models in predictive toxicology,[12,18,56] this review aims to give a new perspective on predictive toxicology by giving an overall picture of the situation, as well as a more focused analysis on some toxicity endpoints.

In predictive toxicology, chemical structures are usually represented by features that can be processed by machine learning methods.[54] This can either take the form of molecular descriptors, molecular fingerprints, or both.[54] Molecular descriptors include features such as atom count, logP, solubility etc., and are commonly obtained using cheminformatics toolkits.[54] Molecular fingerprints represent the molecule as a binary string, with each bit in the string corresponding to the fragments or substructures in the molecule.[54] More detailed explanations of this subject are available.[17,20,54,78,92,106–123] Commonly used molecular fingerprints include MACCS (Molecular ACCess System) and extended-connectivity fingerprints (ECFPs) such as Morgan fingerprints.[107,124–126]

Finally, to determine the reliability of the results and the quality of these models, validation methods such as hold-out validation, k-fold cross-validation, or leave-one-out cross-validation are used.[5,12,18,119-

[121,127–133] Using validation methods to test the models allows for the assessment of the models' robustness and the reliability of the results obtained. Performance measures and metrics have also been reviewed recently.[20,54,72,102,122,134,135]

VALIDATION METHODS

Hold-out validation

Hold-out validation refers to a method where the dataset is split into a training and test set (Figure 3A).[119,121] The data which has been partitioned into the training set is subsequently used for training the model and it is validated by calculating performance statistics using the test set. In the case where the test set is independent of the data used for building the model, this is known as external or independent validation. Generally, the test set contains around 20% of the total dataset. In some cases, the data are split into three partitions to prevent hyperparameter bias. The training set is used for training, the validation set for hyperparameter tuning and the test set for final performance review.

k-fold cross-validation

K-fold cross-validation refers to a method where the dataset is split into k groups (Figure 3B), with k being chosen based on the dataset.[120,121,127,130,133] Commonly used values of k include 5 or 10. One of the groups is set aside as the test set while the model is trained on the remaining groups. This process is repeated iteratively until all groups have been chosen to be the test set once. Model performance is taken to be the average of the test set performance over all groups.

Leave-one-out cross-validation

Leave-one-out (LOO) cross-validation is a special case of k-fold cross-validation, where the number of groups equals the number of data points (Figure 3C).[106,112,131,132]

Regarding the model training process and evaluation, LOO cross-validation follows the process as described earlier in k-fold cross-validation. Model performance using this validation method is similarly taken to be the average across all runs. Generally, LOO cross-validation is the most computationally expensive due to the large number of training cycles required, while the model also tends to overestimate performance. LOO cross-validation is best used for small data sets to offset these disadvantages.

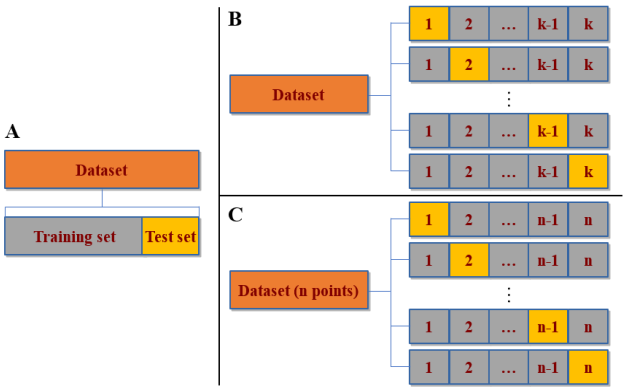


Figure 3: Common validation methods used in machine learning. (A) Hold-out validation, (B) k-fold cross-validation, (C) Leave-one-out cross-validation.

Leave-many-out cross-validation

Leave-many-out cross-validation (LMO-CV), also known as Monte Carlo resampling, or bootstrapping, is another cross-validation technique that is used in the field of machine learning.[136,137] It involves leaving out all possible subsets of m examples of the data.[137] In a similar fashion to LOO cross-validation, in LMO-CV the training data are split into two subsets: a subset of m examples which is used for validation and a subset of (n-m) examples for training the model.[137] In total, there are C_n^m splits that can be carried out on the training examples.[137] LMO-CV carries out this procedure on all possible C_n^m cases, resulting in an exhaustive procedure that is computationally expensive.[137]

Monte Carlo cross-validation

Another validation technique used in machine learning is Monte Carlo cross-validation (MCCV). This method randomly split the samples into two parts $S_c(i)$ (of size n_c) and $S_v(i)$ (of size n_v), and this procedure is repeated N times (i = 1, 2, …, N), where n_c and n_v are the number of samples in the calibration set and validation set respectively.[138] This is defined in equation (1):[138]

$$MCCV_{nv}(k) = 1Nn_v \sum i = 1N\left\|y_{S_v(i)} - \hat{y}_{S_v(i)}\right\|^2 \quad (1)$$

MCCV reduces the computational complexity drastically because of the reduction in the number of samples.[138] In general, $N = n^2$ is sufficient for $MCCV_{nv}$ to perform as well as CV_{nv} (cross-validation).[138] As compared to LOO cross-validation, MCCV has a larger probability to choose the correct number of components in a model.[138] For the dataset, in order to obtain a larger probability using MCCV, the lesser the number of samples is required for validation.[138]

SEARCH PROTOCOL

All references for the machine learning models (Table 3) were obtained by searching the first 50 pages of results in Google Scholar using the model type, the keywords "machine learning" and "toxicity", and the default settings, with all results being sorted by relevance. Each hit was manually checked to ensure relevance to the topic. The year range was restricted to 2010-2020 to obtain the most recent developments. QSARs, expert-based systems, read-across methods, and preprints were excluded from the results. The most recent search was carried out in March 2020.

A similar search was performed with the Web of Knowledge (Web of Science Core Collection) using the same keywords and the default settings. The results from the Web of Knowledge were refined with the same criteria used with Google Scholar. It was found that there were fewer hits from the Web of Knowledge compared to the search performed using Google Scholar. with the hits from Web of Knowledge generally being included in the search from Google Scholar (Table 1). Some of the hits from the Web of Knowledge are not included in the hits from the first 50 pages Google Scholar, perhaps due to the references lacking the keywords specified during the search, or the ordering of the hits in Google Scholar which caused some of the hits to be outside the first 50 pages. Additionally, the references found using Google Scholar were also searched in the Web of Knowledge using the title as the search criteria and the results are recorded in Table 1. In order to give a more complete picture of the developments in the field, the hits from the Web of Knowledge are also included in the review. This search protocol generated a total of 43 references which are listed in Table 3.[9,17,46,139–178]

Machine learning method	Total number of papers from both databases*	Number of papers found during Google Scholar search	Number of papers found during Web of Knowledge search	Number of identical papers found in both databases' search	Number of papers from Google Scholar found in Web of Knowledge
Regression models	2	2	1	1	2
kNNs	8	6	4	2	6
Decision trees	5	2	4	1	1
NB	14	9	5	0	9
SVMs	15	10	8	3	10
Ensemble learning and RF	13	9	9	5	9
Neural networks	13	10	5	2	9
Total	70	48	36	14	46

*Papers with multiple machine learning methods are counted as belonging to the groups they appear in when they are searched. This means that the papers can be counted multiple times and thus the overall total in the table includes this result.

Table 1. Comparison of literature searches in Google Scholar and Web of Knowledge for the year range 2010-2020.

In order to find out if there has been a shift in the situation since 2000-2009, papers were searched using the same criteria as Table 1 while restricting the year range to 2000-2009 with the results being presented in Table 2. However, to speed up the search, only the titles and the abstracts of the papers were considered during the searching process unlike the results in Table 1 where the results were verified manually. This is in line with our intention to estimate the situation for the year range 2000-2009.

Machine learning method	Total number of papers from both databases*	Number of papers found during Google Scholar search	Number of papers found during Web of Knowledge search	Number of identical papers found in both databases' search	Number of papers from Google Scholar found in Web of Knowledge
Regression models	6	5	1	0	5
kNNs	3	2	1	0	2
Decision trees	3	3	1	1	3
NB	3	2	1	0	2
SVMs	9	5	4	0	4
Ensemble learning and RF	3	2	1	0	2
Neural networks	8	6	3	1	5
Total	35	25	12	2	23

*Papers with multiple machine learning methods are counted as belonging to the groups they appear in when they are searched. This means that the papers can be counted multiple times and thus the overall total in the table includes this result.

Table 2. Comparison of literature searches in Google Scholar and Web of Knowledge for the year range 2000-2009.

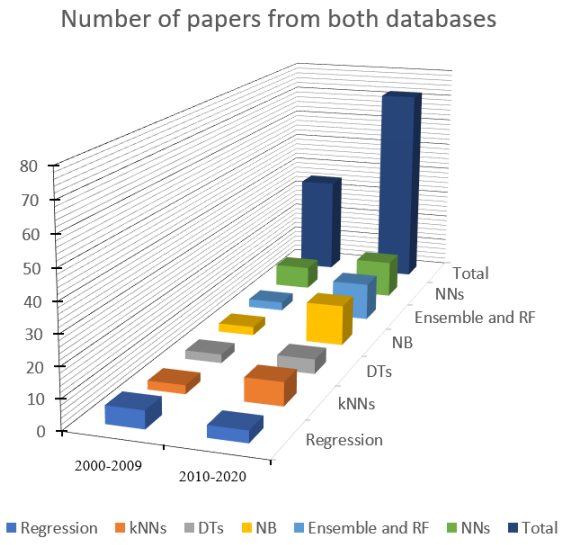


Figure 4. Comparison of results obtained using the search protocol over the year ranges specified.

It was also generally observed that the total number of results using Google Scholar for 2000-2009 is significantly less than 2010-2020 (eg. ca. <1000 compared to >10000 for SVM, 1610 vs. 15600 for neural networks). Therefore, it can be concluded that there has been a general increase in the number of papers published for the field of predictive toxicology for the year range 2010-2020. This can also be seen in Figure 4 which shows the changes in the number of papers over the year ranges used for the search criteria.

However, compiling an exhaustive list of relevant articles on this topic when using this search criteria has limitations. This is because articles can list the names of machine learning methods used without explicitly indicating that they represent machine learning and vice versa. For example, it was observed during the search process that titles of papers generally do not include the machine learning method used ie. They just specify "machine learning" while the machine learning method used is usually specified in the abstract of the paper. A similar situation would also apply for the toxicity endpoints and terms. Such limitations would result in papers being missed or papers being counted wrongly if they are not checked manually for relevance, which is time-consuming. Therefore, to supplement the results of the search criteria obtained in Table 1 and Table 2, we have chosen to highlight several papers obtained using a manual search which have a significant impact in the field of predictive toxicology.

In particular, the Tox21 program, which was developed in accordance with the National Research Council's vision of testing toxicity in the 21st century.[179,180] The large amount of publicly accessible data generated due to the collaboration of many institutions have promoted the development and use of high-throughput screening assays.[179,180] In this review, Table 3 shows the results obtained by the search which generally involves predicting the toxicity of drug or drug-like compounds; in other words, pharmaceuticals. As compared to pharmaceuticals which is the focus in recent times (Table 3), Tox21/ToxCast is different because the focus is on environmental chemicals.[179,180] Several successful machine learning models have also been built using Tox21/ToxCast data, which have demonstrated the potential of these data in predictive toxicology.[140,181–185]

ToxPrint is a widely used set of structural features from molecules in toxicity databases, which can be represented as chemical fingerprints.[141,186] This was developed by Yang et al. in 2015 and is based on various toxicity prediction models and safety assessment guidelines by several institutions including the Food and Drug Administration.[141,186] ToxPrint has also begun to be used with machine learning models, including a recent study predicting estrogen binding.[187]

Another paper to be highlighted involves XGBoost. XGBoost is a scalable tree boosting method used in machine learning, and has received recognition in numerous data mining and machine learning challenges.[188] XGBoost is highly scalable in all scenarios which has contributed to its success in machine learning.[188] Further details can be found in the work by Chen et al. 2016.[188]

Table 3: A summary of the results of all machine learning methods.

No.	Machine learning method	Toxicity endpoint	Accuracy (%)	AUC (%)	SE (%)	SP (%)	Validation type	Dataset size	Reference Number
1	Regression (Linear)	Cardiotoxicity	-	75	-	-	10-fold	1917	[159]
2	Regression (Ridge)	Cardiotoxicity	-	77	-	-	10-fold	1917	[159]
3	Regression (Partial least square)	Molecular toxicity	82	-	-	-	Hold-out	2849	[170]
4	kNN	Aquatic toxicity	84	92	84	85	10-fold	1005	[174]
5	kNN	Carcinogenicity	-	-	84	84	External	661	[175]
6	kNN	Cardiotoxicity	82	78	82	57	External	206	[176]
7	kNN	Genotoxicity	86	93	80	90	5-fold	576	[177]
8	kNN	Hepatotoxicity	62	52	91	20	External	978	[178]
9	kNN	Hepatotoxicity	78	-	79	76	10-fold	1274	[139]
10	kNN	Hepatotoxicity	-	-	62 ± 20	92 ± 14	10-fold	288	[140]
11	kNN	Organ toxicity	-	-	92 ± 8	78 ± 6	5-fold	-	[141]
12	DT	Food-related toxicity	0.23[+]	-	-	-	10-fold	94	[172]
13	DT	Genotoxicity	82	81	75	87	5-fold	576	[177]
14	DT	Hepatotoxicity	89*	-	-	-	10-fold	575	[173]
15	DT	Hepatotoxicity	-	-	74 ± 16	94 ± 5	10-fold	288	[140]
16	DT	Organ toxicity	-	-	89 ± 12	76 ± 7	5-fold	-	[141]
17	NB	Aquatic toxicity	77	81	70	84	10-fold	1005	[174]
18	NB	Carcinogenicity	68 ± 2	-	60 ± 8	75 ± 10	5-fold	834	[142]
19	NB	Developmental toxicity	83	-	90	67	5-fold	232	[143]
20	NB	Genotoxicity	85	90	89	81	5-fold	576	[177]
21	NB	Hepatotoxicity	-	-	73	73	5-fold	336	[144]
22	NB	Hepatotoxicity	-	-	70 ± 15	85 ± 7	10-fold	288	[140]
23	NB	Immunotoxicity	-	78	73	70	Hold-out	44615	[145]
24	NB	Mitochondrial toxicity	81 ± 1	-	88 ± 4	77 ± 4	5-fold	226	[148]
25	NB	Mutagenicity	90.9 ± 0.3	-	39 ± 4	95 ± 1	5-fold	5159	[146]
26	NB	Mutagenicity	65	-	-	-	10-fold	3903	[147]
27	NB	Myelotoxicity	82 ± 3	-	76 ± 6	84 ± 3	External	727	[149]
28	NB	Nephrotoxicity	-	-	62	78	Stratified 3-fold	27	[150]
29	NB	Respiratory toxicity	84	-	84	85	5-fold	993	[151]
30	NB	Urinary tract toxicity	84	-	84	85	5-fold	173	[152]
31	SVM	Acute oral toxicity	90	-	-	-	External	8102	[46]
32	SVM	Acute toxicity	70	72	85	59	External	321	[153]
33	SVM	Aquatic toxicity	89	94	89	89	10-fold	1005	[174]
34	SVM	Carcinogenicity	-	-	73	79	External	499	[9]
35	SVM	Carcinogenicity	68 ± 3	73 ± 3	65 ± 5	72 ± 5	5-fold	802	[154]
36	SVM	Carcinogenicity	78	-	84	74	External	661	[175]
37	SVM	Cardiotoxicity	86	72	88	29	External	206	[176]
38	SVM	Cardiotoxicity	87	-	90	74	10-fold	1501	[155]
39	SVM	Genotoxicity	89	95	92	84	5-fold	576	[177]

40	SVM	Hepatotoxicity	-	-	58 ± 16	99 ± 6	10-fold	288	[140]
41	SVM	Hepatotoxicity	75	61	93	38	External	978	[178]
42	SVM	Hepatotoxicity	83	89	93	68	External	1731	[156]
43	SVM	Mutagenicity	72	-	69	74	10-fold	1696	[157]
44	SVM	Nephrotoxicity	-	-	79	84	Stratified 3-fold	27	[150]
45	SVM	Ototoxicity	85	-	82	92	External	536	[158]
46	Ensemble learning/RF	Aquatic toxicity	86	93	83	89	10-fold	1005	[174]
47	Ensemble learning/RF	Carcinogenicity	70 ± 3	77 ± 3	67 ± 5	73 ± 4	5-fold	802	[154]
48	Ensemble learning/RF	Carcinogenicity	74	-	65	80	Hold-out	661	[175]
49	Ensemble learning/RF	Carcinogenicity	68 ± 3	74 ± 3	64 ± 5	73 ± 4	5-fold	802	[154]
50	Ensemble learning/RF	Cardiotoxicity	82	94	65	98	10-fold	2901	[160]
51	Ensemble learning/RF	Cardiotoxicity	97	-	-	-	External	522	[161]
52	Ensemble learning/RF	Genotoxicity	94	96	95	93	External	576	[177]
53	Ensemble learning/RF	Hepatotoxicity	69 ± 3	75 ± 3	76 ± 3	62 ± 5	5-fold	993	[163]
54	Ensemble learning/RF	Hepatotoxicity	74	79	-	-	10-fold	281	[162]
55	Ensemble learning/RF	Hepatotoxicity	-	-	58 ± 15	97 ± 5	10-fold	288	[140]
56	Ensemble learning/RF	Hepatotoxicity	71 ± 3	76 ± 2	80 ± 4	60 ± 5	5-fold	1117	[163]
57	Ensemble learning/RF	Hepatotoxicity	73	-	77	66	10-fold	1274	[139]
58	Ensemble learning/RF	Nephrotoxicity	-	-	89	75	10-fold	30	[164]
59	NN (ANN)	Acute toxicity	-	70	100	53	External	321	[153]
60	NN (ANN)	Aquatic toxicity	88	94	87	89	10-fold	1005	[174]
61	NN (ANN)	Genotoxicity	87	94	91	82	5-fold	576	[177]
62	NN (ANN)	Hepatotoxicity	82	-	71	98	External	475	[166]
63	NN (ANN)	Mutagenicity	60	-	40	81	10-fold	1696	[157]
64	NN (ANN)	Mutagenicity	80	87	84	75	5-fold	6094	[167]
65	NN (DNN)	Cardiotoxicity	93	97	93	91	Hold-out	3954	[165]
66	NN (DNN)	Cardiotoxicity	98	-	-	-	External	522	[161]
67	NN (DNN)	General	-	84	-	-	External	11764	[17]
68	NN (DNN)	Hepatotoxicity	81	-	82	80	External	475	[166]
69	NN (Graph CNN)	Cardiotoxicity	-	96	-	-	-	3954	[165]
70	NN (CNN)	General	-	78	-	100	-	10588	[168]
71	NN (CNN)	General	-	85	-	-	-	7438	[169]
72	NN (CNN)	Hepatotoxicity	63	62	64	62	-	7630	[171]

*corrected classification rate (CCR) used instead of accuracy.

+Error between the predicted value vs. the actual value was used instead of accuracy.

MODEL TYPES

Regression models

A regression task involves predicting a numerical response variable using several predictor variables by learning a model that minimises the loss function.[189] First, we will distinguish the regression models from support vector regression (SVR), which is an application of support vector machines (SVM), and will be covered in a later section. Also, in the literature, kernel functions are sometimes referred to as regression models. In this review, we will make a distinction between the two: the term kernel functions will be reserved for the SVMs while regression models will be discussed in this section.

Regression models, which are statistical models, can be broadly classified as linear regression and non-linear regression. These include linear, multivariate linear, polynomial, stepwise, ridge, and least absolute shrinkage and selection operator (LASSO).[190-205] These regression models are used for quantitative predictions unlike the

standard classification models. Examples of these quantitative characteristics of toxicity include LD_{50}, LC_{50}, IC_{50}, and EC_{50}.

While linear regression models have low computational cost, their linear nature limits their ability to model complex problems, unlike non-linear regression models.[84,206,207] However, using non-linear regression models will increase the computational cost.

In predictive toxicology, regression functions have been employed in several works. These models, as well as all machine learning methods in this review, will be measured by performance metrics which include accuracy (Q), sensitivity (SE), specificity (SE), and area under the receiver operating characteristic curve (AUC). Even though the accuracy was chosen as the performance metric for regression model types, it is acknowledged that R^2 and root-mean-square error (RMSE) are more adequate performance metrics to gauge the quality of the model. However, in line with our intention to give an estimate of the performance of regression models as compared to the other machine learning methods, accuracy which is a common performance metric used in machine learning was used.

Table 3, entries 1 – 3, summarizes the recent performance of several regression models across different toxicity endpoints.

k-nearest neighbours (kNNs)

kNN is a non-parametric classifier where the test sample is assigned a class label based on the most frequently occurring class label among the k-nearest neighbours.[208–210] A proximity measure, such as Euclidean distance or Manhattan distance, is used to define the k-nearest neighbours to each test sample.[208–210] All samples are represented by points in an n-dimensional feature space, while the neighbours are taken from a set of objects for which the correct classification or value is known.[209]

More formally, the k-nearest neighbours algorithm is defined as follows: Given a collection of incomplete/unlabelled test data $\{(x_i, y_i), i = n + 1, \dots, n + m\}$, the problem amounts to predicting the class labels for $y^* = \{y_{n+1}, \dots, y_{n+m}\}$ with corresponding feature vectors $x^* = \{x_{n+1}, \dots, x_{n+m}\}$.[208,211] Thus, the k-nearest neighbours algorithm amounts to classifying an unlabelled y_{n+1} as the most common class among the k-nearest neighbours of x_{n+1} in the training set $\{(x_i, y_i), i = 1, \dots, n\}$.[208,209,211] In the algorithm, the value of k is typically a positive integer, usually small (such as $k = 1$), or is chosen based on leave-one-out cross-validation.[208,209,211,212] If the value of k chosen is too small, it might result in overfitting while if the value of k is too large, it might result in misclassification.[209] While kNN is easy to implement and often gives good performance, it is heavily dependent on the classification accuracy of the test class labels as well as the value of k.[211] Samsudin et al. has also outlined several improvements to kNN in the literature.[210] Table 3, entries 4 – 11, summarizes some of the results achieved by kNN models.

Decision trees (DT)

A decision tree is a tree-strcutured classifier consisting of a root, nodes and leaves where each node has only one unique path from the root. The decision that the classifier makes at each node is based on decision rules, which depends on the features of the data used. Several papers have explained DTs in great detail and will not be reproduced here.[213–215] Furthermore, in a later section, Random Forest which is an ensemble of decision trees will also introduce the basics of decision trees.

DTs have the advantage of model interpretability because the decision rules can be retrieved from the model for each result, unlike complex models like neural networks where each node is based on all of the nodes in the previous layer. However, due to the design of the tree, it is easier for errors to accumulate at each level, and thus a compromise on accuracy and efficiency has to be reached.

In predictive toxicology, decision trees are not commonly used as evidenced by the data from Table 1 and Table 2. This could be because the toxicity endpoints are complex and thus a simple model cannot generalize all the patterns in the data.

Table 3, entries 12 – 16, summarizes some of the results achieved by DT models.

Naive Bayes (NB)

In Bayesian classification, the given data are hypothesized to belong to a particular class.[216] The probability that the hypothesis is true is then calculated.[216] Another way to describe the Bayesian classifier is shown in equation (2), where the Bayesian classifier is defined as obtaining the posterior probability $P(C_i \mid A_1, \dots, A_n)$ of each class C_i, using Bayes rule.[217,218]

$$P(C_i \mid A_1, \dots, A_n) = P(C_i)P(A_1 \mid C_i) \dots P(A_n \mid C_i)/P(A) \quad (2)$$

This equation makes the simplifying assumption that given the class, the attributes, A, are independent and thus the likelihood can be obtained by the product of the individual conditional probabilities of each attribute.[217,218] This is called a naive Bayes (NB) classifier.

Naive Bayes models are efficient, generally robust and have high accuracy.[217,218] However, the NB model classification accuracy decreases when the attributes are not independent.[217] Also, NB models cannot deal with non-parametric continuous attributes.[217] Some improvements such as feature selection have been carried out to tackle these issues.[219] Additional details about these improvements can be found in the literature.[217,219–222] Table 3, entries 17 – 30, shows the results obtained for NB models in predictive toxicology.

Support vector machines (SVM)

The next model type to be introduced is SVM, which was introduced by Vapnik et al..[223] It is based on the structural risk minimization principle which was developed from statistical learning theory.[223–227] Vapnik et al. state that the basis of the principle is to control the Vapnik–Chervonenkis dimension (VC-dimension) to minimise the guaranteed risk, which is the sum of the empirical risk and the confidence interval.[224] This, however, involves a trade-off as the minimum empirical risk decreases while the confidence interval increases as the VC-dimension increases.[224] Yan et al. describes this in another way, which is that SVMs, which are maximum margin classifiers, simultaneously minimize the empirical classification error and maximize the geometric margin.[225]

According to Vapnik et al., an SVM maps the input vectors into some high dimensional feature space Z through some non-linear mapping chosen *a priori*.[223] This allows a hyperplane to be constructed between the data points, where there is a margin of separation between the two classes.[223] If the training data cannot be separated with error, the algorithm would separate the training set with a minimal number of errors, which results in a soft margin SVM.[223]

SVMs usually results in a feature space where the data are linearly separable, even when the initial feature space is non-linear. In the case of a linear classifier, the feature space is separated by a hyperplane with equation (3), where w is the weight vector, x is the input vector, and b is the bias (Figure 5).[20,223,228–230]

$$w^T \cdot x + b = 0 \quad (3)$$

The geometric margin is thus represented by the constraints shown in (4) and this is shown in Figure 5 as the boundary lines running parallel to the hyperplane.[20,223,228–230]

$$w^T \cdot x + b \begin{cases} \geq 1 \; when \; y_i = +1 \\ \leq -1 \; when \; y_i = -1 \end{cases} \quad (4)$$

To construct a linear SVM classifier for a non-linear feature space, kernels such as a Gaussian, radial basis function, or polynomial types are used.[9,157,226,231–236] These kernels are functions that work by mapping the input data which is linearly non-separable into a higher dimensional space where the data are linearly separable.[9,231] A hyperplane can thus be constructed that separates the two classes, resulting in a situation similar to a linear classifier.

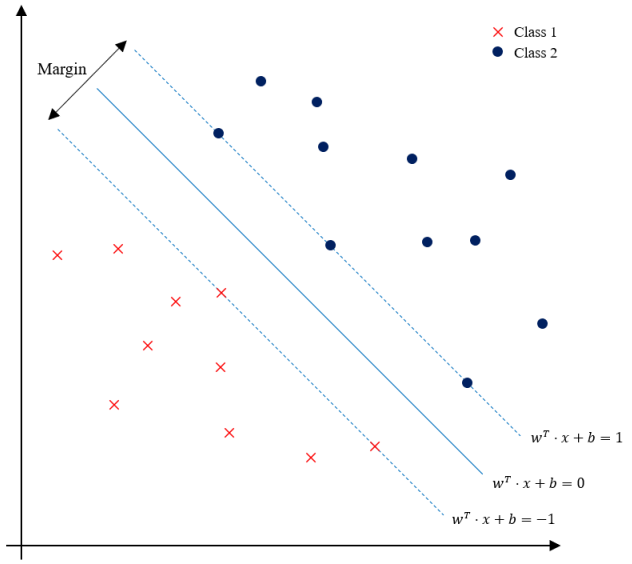


Figure 5. A representation of an SVM with linearly separable classes.

It is known that SVMs are good at pattern recognition, can generalize well, and can handle high-dimensional data.[223,226,228,237] However, the drawbacks of SVMs include difficulty in choosing an appropriate kernel and being time-consuming for large datasets.[231,233,237] Improvements have been made to counter these drawbacks. For example, SVMs such as the generalized eigenvalue proximal support vector machine (GEPSVM) and twin support vector machine (TWSVM) have been developed to reduce the time consumed.[238–240] These SVMs work by constructing two non-parallel hyperplanes instead of a single hyperplane, effectively reducing the quadratic programming problem (QPP) required to generate the hyperplane(s) from a single large QPP to two smaller QPPs.[238–240] Also, multiple kernel functions have been developed which can handle complex classification problems better by adapting better to the characteristics of the data.[241–244]

In recent years, several SVMs have been used in predictive toxicology. These results are tabulated in Table 3, entries 31 – 45.

Random forest (RF) and ensemble learning

A recent review by Sagi et al. provides a comprehensive survey of ensemble learning.[245] In this review, some of the ideas by Sagi et al. will be introduced in this section as a general introduction to this machine learning method. There are also other reviews on ensemble learning.[246–252] In this review, a general overview on ensemble learning will be given before focusing on RF as it a popular machine learning method used in predictive toxicology.

Ensemble learning refers to methods that combine multiple inducers to make a decision, typically in supervised machine learning tasks.[245] An inducer, or base-learner, is an algorithm that takes a set of labelled examples as input and produces a model that generalizes these examples.[245] By combining multiple models, the error of a single inducer will likely be compensated by other inducers, which leads to better overall performance as compared to a single inducer.[245] In other words, the predictive performance of an ensemble learning model cannot be lower than the predictive performance of each model making up the ensemble.[245]

More formally, the ensemble learning method can be represented as follows: Given a dataset of n examples and m features, D = {(x_i , y_i)}(| D | = n, $x_i \in R^m$, $y_i \in R$), an ensemble learning model φ uses an aggregation function G that aggregates K inducers, { f_1 , f_2 , … , f_k } towards predicting a single output.[245] This is represented by equation (5).[245]

$$\hat{y}_i = \varphi(x_i) = G(f_1, f_2, \ldots, f_k) \quad (5)$$

In an ensemble method, there exist two main types of frameworks.[245] The dependent framework is one where the inducer's output affects the construction of the next inducer.[245] In contrast, the independent framework each inducer is built independently from other inducers.[245] For instance, popular ensemble methods such as AdaBoost, bagging, and random forest (RF) are examples of dependent, dependent, and independent frameworks respectively.[245]

Ensemble learning methods are generally able to handle class imbalance, concept drift, and the curse of dimensionality better as compared to their machine learning counterparts.[245] Also, these methods tend to avoid overfitting as different hypotheses are averaged to give the final result.[245] Moreover, ensemble learning methods decrease the risk of finding a local minimum while also giving a better representation of the data.[245]

However, even with these numerous advantages, there are some considerations when building an ensemble learning model. This includes the individual method's suitability towards the data, difficulty in interpreting the output of the ensemble learning model, the software availability, and the usability, and, sometimes, computational cost.[245] The key considerations would be the first two points. Elaborating slightly further on the first point, it is known that each machine learning method is more suitable to handle certain types of data or settings. This will be explained in more detail in the section which covers the overall analysis. Next, as ensemble learning methods are built up of multiple models, it follows that the total computational cost will at least be equivalent to running each model separately. It is thus likely that significant computational resources need to be allocated to train an ensemble learning method for results to be obtained within a reasonable time frame. Improvements to ensemble learning methods are also covered in the review by Sagi et al..[245]

Even though ensemble learning uses a general approach eg. Bagging, we intend to give an estimate of the performance of models that use these general approaches as compared to concrete machine learning methods like RF. Ensemble learning can be interpreted as a group of algorithms eg. Bagging and a machine learning method, or as a group of machine learning models eg. NB and neural networks. Thus, care should be taken when comparing performance metrics of ensemble learning methods to other machine learning methods. It should also be noted that RF is a special case because it is made up of a ensemble of decision trees, and is treated as a concrete machine learning method.

Next, RF will be covered in more detail while details on the other two methods can be found in the literature.[252–257]

Breiman introduced RF in 2001 as a classifier that is made up of multiple tree-structured classifiers (decision trees) {h(x, θ_k), k = 1, ...} where the {θ_k} are independent identically distributed random vectors, and each tree casts a unit vote for the most popular class at input x.[111,258] Each RF consists of a root, nodes and leaves, with each split representing two branches at each node. (Figure 6). Each node in an RF represents the decision made by the model which is based on a subset of the available features while the leaves represent the outputs of the RF.[111,258–260] These outputs, which are predictions of each tree, are combined by taking the most common prediction across all trees.[109,258,259]

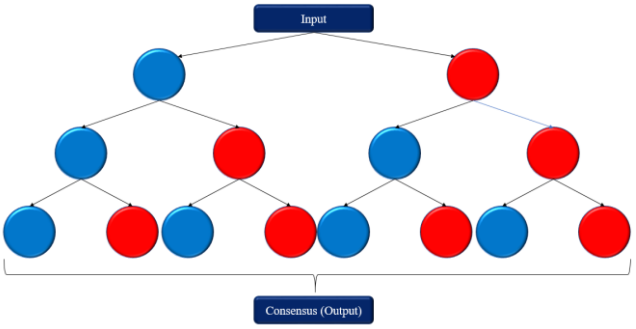


Figure 6. A Random Forest with two decision trees. The two colours represent the positive or negative decision made at each split.

To build an RF for machine learning, an algorithm has to be chosen. One common algorithm used in literature was developed by Breiman and uses the Bagging principle together with random feature selection.[111,258] Each tree in the RF is grown by randomly drawing N samples from the original training set with replacement, also known as bootstrapping which is then used to build the tree.[111,258] For each node of the tree, a portion of features from the original feature space is randomly drawn without replacement, among which the best split is selected.[111,258] Throughout the process, no pruning of the tree is performed.[111,258]

Building an RF also requires that one considers pruning as well as the number of trees. Pruning refers to removing nodes in the RF-based on a criterion which simplifies the RF.[260,261] On the other hand, the number of trees in an RF affects the generalization of the model.[111,258] However, it was found that above 10000 trees using Adaboost, or 100 trees for a well-defined value of the number of random features pre-selected in the splitting process using the Forest-RI method, adding more trees does not improve performance.[111,258]

RFs have several advantages such as they can minimise the issue of overfitting, are resistant to noise with some algorithms, can handle high-dimensional data well, having the ability to ignore irrelevant descriptors, and being interpretable in terms of the decision rules made.[109,262,263] RFs are also known to keep the benefits afforded by DTs while achieving better results most of the time.[263] On the contrary, the disadvantages of RF include being susceptible to bias when there are dominant features, as well as placing more emphasis on the correlation among smaller groups of features.[264]

Recently, a review by Fawagreh et al. in 2014 has covered some of the recent advancements in RF.[265] They mention in the review that the performance of the RF can be improved through the use of different voting methods, or by implementing a weighting scheme for the features or for discarding trees.[265] Another review by Rokach in 2016 on decision forests (including RFs) introduces the models, the methods to construct them, and surveys the state-of-the-art methods in the field.[246]

Next, Table 3, entries 46 – 58, summarizes the results of RFs and ensemble learning used in predictive toxicology.

Neural networks (Artificial neural networks (ANNs), deep neural networks (DNNs), Convolutional neural networks (CNNs))

Neural networks (NNs) can generally be divided into three groups, namely artifical neural networks (ANNs), deep neural networks (DNNs), and convolutional neural networks (CNNs). In this review. ANNs and DNNs will be elaborated in more detail first as they as similar, following which CNNs will be explained in more detail.

In recent times, artificial neural networks (ANNs) which excel at pattern recognition and classification have been successfully applied in multiple fields such as novelty detection, renewable energy systems, and image processing.[266–268] The most widely used types of ANNs include feedforward and recurrent neural networks, of which feedforward neural networks are generally enough for most binary classification tasks.[269] In this review, the focus will be on feedforward neural networks. Recurrent neural networks such as long short-term memory (LSTM) are not covered in this review and can be found in the literature.[270–274] In a typical feedforward neural network, the network is made up of layers, namely the input layer, hidden layers and the output layer. Each layer is in turn made up of independent nodes or neurons, with each node connected to all nodes in the subsequent layer. This is illustrated in Figure 7. In the literature, several strategies for determining the number of nodes as well as hidden layers when building a neural network have been described.[275–277]

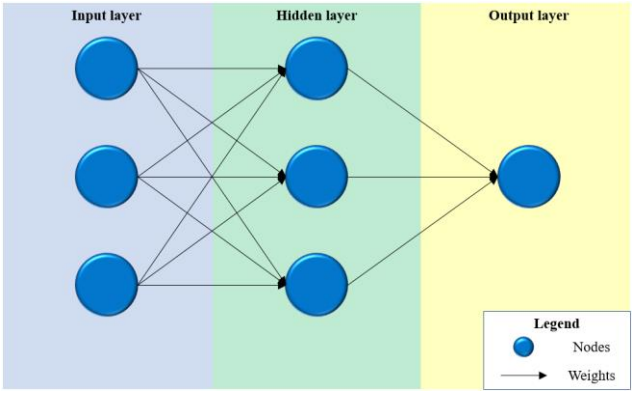


Figure 7. Graphical representation of a feedforward ANN.

In addition to the network architecture, one must choose an activation function which is a function that transforms the activation level of a unit (neuron) into an output signal.[278] Examples of activation functions include linear, sigmoid, Gaussian, and Elliot.[278] A popular and recent activation function called rectified linear unit (ReLU) has been shown to converge quickly for neural networks.[279–281] The ReLU activation function is represented by equation (6).

$$f(u) = max(0, u) \quad (6)$$

Overall, the output in a feedforward neural network can be described by equation (7) where o_i represents the output, σ_i is the output function, b_i is the bias input of the ith hidden node, w_{ij} is the weight of the connection from the jth input, u_j, to the ith hidden node, and M is the total number of inputs.[282]

$$o_i = \sigma_i(b_i + \sum_{j=1}^{M}(w_{ij} * u_j)) \quad (7)$$

The starting weights are usually randomised, with the final weights commonly determined by backpropagation. Backpropagation is a method to calculate the gradient of the error with respect to weights and several algorithms have been developed which include first order and second-order algorithms.[283,284] One of the commonly used algorithm is the Levenberg-Marquardt (LM) algorithm which is a second-order algorithm.[285] Wilamowski et al. describes these backpropagation algorithms in more detail.[285]

Recently, deep neural networks (DNNs) have gained interest due to deep learning, of which there have been several reviews published in the literature.[1,84,91,269,284,286–290] These DNNs are neural networks with a deep network architecture, where the number of hidden layers is more than one.[291,292] By increasing the number of hidden layers, DNNs can handle more complex problems, of which there are successful examples such as skin cancer classification, image classification, and syntheses route planning.[291–293]

Despite the growing popularity of DNNs, one must acknowledge the several limitations of DNNs. These include long network training times due to the increase in the number of hidden layers and parameters. Improvements have been made to rectify these issues. For example, graphical processing units (GPUs) have been employed to train neural networks, where the higher processing capabilities of GPUs help reduce training times.[294–297]

On the other hand, ANNs have their own set of limitations which include their expressivity, which shows that they are unable to express certain functions that DNNs are capable of, as well as having lower approximation capability as compared to DNNs.[298] Unfortunately, as these are issues intrinsic to the model type, there are no good solutions for them.

In the field of predictive toxicology, DNNs have been shown to be successful in numerous examples (Table 3). Several papers have also covered on the topic of neural networks or DNNs in the field of predictive toxicology,[10,16,102] in particular, the review by Tang et al. in 2018.[5] Hence, this review will not focus on the recent advances of DNNs, but rather the model performance of DNNs in predictive toxicology. Given the increased performance of DNNs as compared to their ANN counterparts, it is predicted that the use of DNNs in predictive toxicology will become more prevalent. The complexity provided by DNN could also give these models an edge when predicting complex toxicity endpoints, for which simpler models might not be able to model well.

Convolutional neural networks are similar to ANNs and DNNs in the sense that they also consist of layers and are feedforward networks. However, unlike their counterparts, a CNN typically consists of convolutional and pooling layers stacked on top of each other.[299–301] The fully connected layers that follow these layers interpret the feature representations and perform the function of high-level reasoning, such as classification.[300]

Convolutional layers serve as feature extractors, where the neurons are arranged into feature maps.[302,303] Each neuron has a receptive field which is connected to the neurons in the previous layer via a set of trainable weights while all neurons within a feature map have weights that are constrained to be equal.[302,303] Equation (8) shows the representation of the kth output feature map Y_k, where x represents the input image, W_k the convolutional filter, the multiplication sign as the 2D convolutional operator used to calculate the inner product of the filer model at each location of the input image, and $f(\cdot)$ the non-linear activation function.[300]

$$Y_k = f(W_k * x) \tag{8}$$

In contrast, the pooling layers reduces the spatial resolution of the feature maps which reduces the number of parameters (controls overfitting), achieving spatial invariance to input distortions and translations.[300] These pooling operations play a role in producing downstream representations that are more robust to the effects of variations in data while still preserving important motifs.[300,302] Lee et al. and Rawat et al. describes these pooling operations in more detail.[300,302] In toxicology, molecules are represented as images, grids, or graphs which are then fed into the CNN for training.[165,168,169,171]

Similar to ANNs and DNNs, CNNs also uses the backpropagation algorithm during training.[300] A typical CNN with both convolutional and pooling layers is shown in Figure 8.

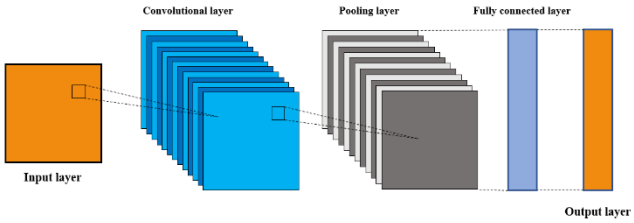


Figure 8. A CNN with convolutional and pooling layers.

CNNs have several advantages over traditional ANNs such as requiring fewer free parameters and being able to deal with the variability of 2D shapes.[300] However, CNNs require a large amount of training data which increases computational cost and lengthens training time.[300] In recent years, new CNNs have been developed such as deep CNNs and graph CNNs which outperforms traditional CNNs. A recent review by Rawat et al. covers the topic of deep CNNs more extensively.[300] On the other hand, graph CNNs are the most recent CNNs to be developed which consist of embedding nodes in a graph in Euclidean space.[303] More information about graphs and graph CNNs can be found in the literature.[303–307]

Although CNNs have made advances in recent times, in the field of predictive toxicology, CNNs are uncommonly used. This is because CNNs use images as inputs while in predictive toxicology, molecular structures, fingerprints, or descriptors are more common as inputs. An example of this is the work by Jimenez-Carretero et al.[308] In their work, a CNN using cell-based images as features was developed. This is a situation where machine learning is used to generalize the biological properties, rather than the chemical properties which the main focus of predictive toxicology focuses on. In this review, we have chosen to focus on molecular structure in predictive toxicology as thus far, CNNs have not shown themselves to be useful in this field.

Table 3, entries 59 – 72, summarizes some of the results achieved by NN models.

OVERALL ANALYSIS

The popularity of SVMs and RFs could be attributed to their advantages, such as being efficient while easy to use, as well as being able to generalize the data well. Furthermore, numerous successful models have been built for these two model types, thus contributing to their well-established reputation. On the other hand, recently developed machine learning models, such as DNNs or ensemble learning, are less popular possibly due to their high computational cost or their complexity, or because they are not as well-established as SVMs and RFs. Machine learning methods perform differently on different datasets and these differences arise from the diverse characteristics of the data, such as the dataset size, class distribution, and the distribution of the data in the feature space.

It is also observed that the simplest machine learning method, regression models are not the most popular model of choice in predictive toxicology. The performance of regression models also falls behind their classification counterparts. However, caution should be taken when comparing the performance of regression models with classification models as the two are inherently different. This illustrates the complexity of the problem of predicting the toxicity of chemicals, where usually, a more complex machine learning model is required to model the problem or the data more effectively. A more complex model might also produce results that one would otherwise miss if a simpler model was used instead, simply because the model does not over-generalize the data. Another possible reason is perhaps the familiarity or ease of use of such machine learning methods, where one is inclined to use a method that is more familiar to raise the chances of building a successful model. Since SVM has been well established in the literature, it continues to be a popular machine learning method for predictive toxicology.

The statistical performance results of all machine learning methods in predictive toxicology covered, including the values for sensitivity (SE), specificity (SP), accuracy, AUC, and the validation type are summarized in Table 3. It is common to see different studies use different performance metrics to measure their model's performance. While the best performance metric to use is still up for debate, such diversity in the performance metrics makes it difficult to compare across different models. Moreover, as the models generally do not use a benchmark dataset, it is once again difficult to compare the performance of different models.

Generally, the machine learning methods in Table 3 have an accuracy or AUC of 75% or above. Those models which reported a lower performance than expected could have experienced issues with the dataset, such as the dataset not being able to generalize well to another test set. Most of the time, it is more likely for there to be a problem with the data or the deployment of the algorithm, rather than the machine learning method of choice which are generally well established.

Based on the data and the search criteria outlined in Table 3, hepatotoxicity, carcinogenicity/mutagenicity/genotoxicity, and

cardiotoxicity are the most common types of toxicity that have been investigated. Hepatotoxicity is important in predictive toxicology because most toxicity originates from the liver which is the main site of metabolism for drugs; that is hepatotoxic compounds would have adverse side effects *in vivo* and thus are unsuitable to be drugs.[139,156] Hence, determining the hepatotoxicity potential of a drug candidate would allow for the quick screening of potential drug candidates from all of the compounds. Cardiotoxicity is important because side effects such as cardiac arrest are highly undesirable. Lastly, tests for carcinogenicity and mutagenicity, during drug screening are common as cancer is known to be a leading cause of death in the developed world.[225] The Ames test is used as part of a battery of *in vitro* methods which covers different mechanisms that may lead to carcinogenicity.[309,310] *In silico* predictions of the Ames test mutagenicity have also been investigated by Xu et al. and Hillebrecht et al., which highlights the need for faster and more accurate predictions of a key test in toxicology.[167,311] Hillebrecht et al. has demonstrated that the expert-based system Derek performs the best for predicting Ames test mutagenicity, though they believe that the fusion of the expert-based system and QSAR techniques would lead to improvements in the predictive power of the *in silico* models.[311] Therefore, by screening for these major types of toxicity, the number of potential drug candidates can be reduced to a smaller number from a larger range of drug-like compounds.

The distribution of all models in Table 3 is shown in Figure 9 (A). Entries without accuracy/AUC values, or missing data were omitted. Also, for entries that report both values, accuracy was chosen to represent the model performance as it ais a common performance metric. Larger dataset sizes do not correspond to higher model performance, but some model types do appear to have been more prominently reported for some dataset sizes. Ensemble learning/RF, nearest neighbour algorithms, SVMs, and naïve Bayes models are more common where training datasets are under 1000 datapoints. Neural networks of all types see more use in datasets over 2000 datapoints.

When models with accuracy or AUC greater than or equal to 90% are considered, Ensemble learning/RF are more prominent among smaller datasets while neural networks, support vector machines and naïve Bayes are more common among larger datasets. Neural networks are the most represented algorithm type in this category, with three models scoring above 90%.

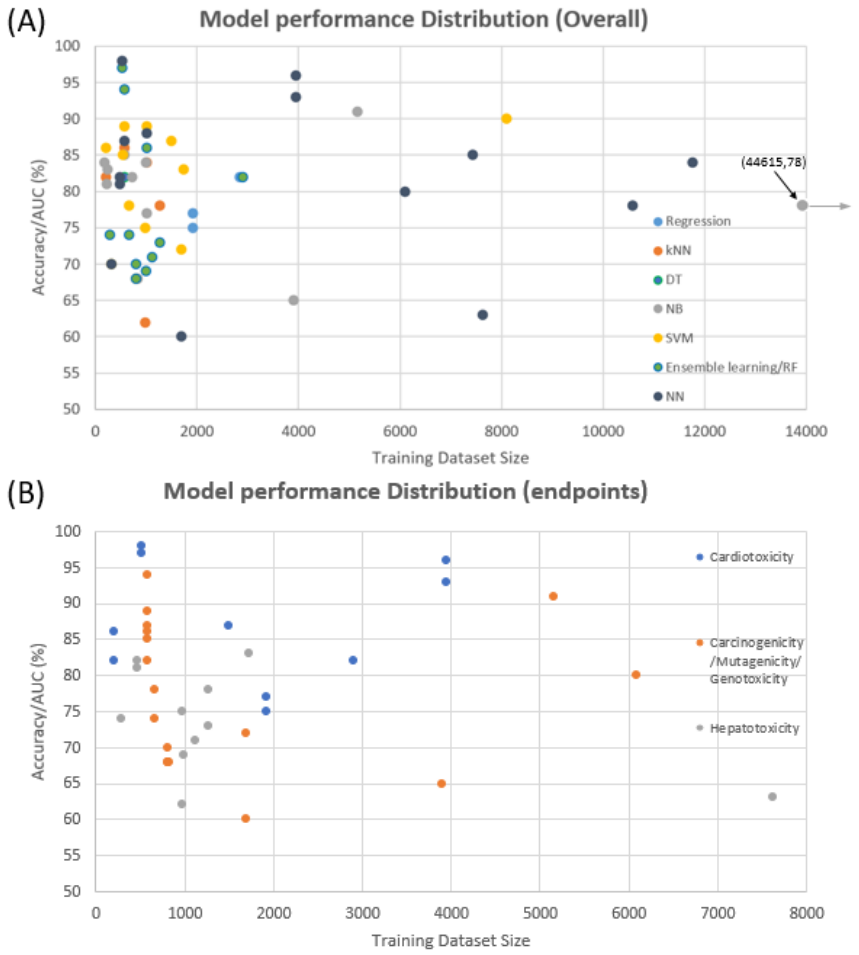


Figure 9. A graph showing model performance against training dataset size across (A) different model types (B) different endpoints.

Looking at the highest performing models for each of the toxicity endpoints most represented in Table 3, some suggestions can be provided for those new to machine learning model construction. The highest performing models for carcinogenicity, genotoxicity, and mutagenicity are random forests and support vector machines using MACCS keys and PubChem fingerprints as inputs and relatively small datasets.[177] High performing models on larger datasets use molecular descriptors and ECFPs in a naïve Bayes model.[146] For cardiotoxicity, the highest performing model is a deep neural network using fingerprints on a small dataset.[161] A support vector machine using MACCS fingerprints on a small dataset provides the highest performing model in hepatotoxicity.[156] These are summarized in Figure 10 and serve as suggestions to be considered when constructing new models as the

data type, distribution, and modelability also affect which models will perform best and how high model performance statistics will be.

It has to be acknowledged that it is hard to compare the algorithm performance over different toxicity endpoints due to the difference in complexity and data available. Hence, a subplot of Figure 9 (A) was generated for the three common toxicity endpoints, namely, hepatotoxicity, carcinogenicity/mutagenicity/genotoxicity, and cardiotoxicity. This subplot is shown in Figure 9 (B). There are more models predicting cardiotoxicity with an accuracy/AUC of above 90%, followed by carcinogenicity/mutagenicity/genotoxicity, and lastly hepatotoxicity. The lower overall performance of models predicting hepatotoxicity could possibly arise due to a lack of data used for building the models. It is also observed that most of the models have 2000 or less training data points.

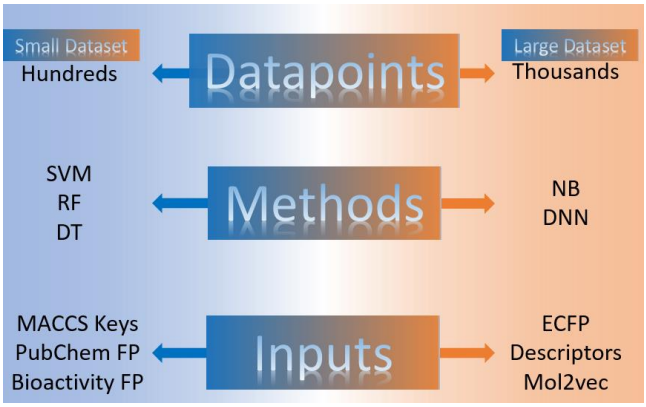


Figure 10. A summary of the highest performing methods and inputs found separated by dataset size.

To study model performance based on human vs animal and *in vivo* vs *in vitro* data, the studies in Table 3 were differentiated based on the data used where possible and plotted alongside each other (Supporting Information). No clear trend or pattern was observed for the generated plots. This indicates that the model performance is dependent on the quality and quantity of the data used rather than the type of data (eg. animal vs. human). Caution must also be taken when interpreting these results as a whole because these studies use different datasets. The lack of high-quality *in vivo* data could also have contributed to this outcome.

In 2016, Huang et al. built predictive models for 72 *in vivo* toxicity endpoints using *in vitro* data from the Tox21 10 K library.[181] It was found that the *in vitro* assay data-based models performed better when predicting *in vivo* toxicity end points for humans than using *in vivo* data from animals to predict *in vivo* end points for humans.[181] Whilst more high-quality data for human toxicity and *in vivo* studies are required to better assess the model performance of these *in vitro* data-based models in the prediction of *in vivo* toxicity, this result demonstrates that *in vitro* data from assays offers a promising alternative to expensive and low throughput methods of obtaining *in vivo* data, and that extrapolation from *in vitro* data to human *in vivo* end points is more reliable than extrapolation from animal to human *in vivo* data.[181]

In another study by Novotarskyi et al. in 2016, the top model for the ToxCast EPA (Environmental Protection Agency) challenge was reported.[312] The aim of the challenge was to develop a model to predict the lowest effect level concentration (*in vivo* toxicity) based on *in vitro* measurements and calculated *in silico* descriptors.[312] A recent study by Xu et al. in 2020 also investigated *in vivo* toxicity. In their work, predictive models for human organ toxicity based on *in vitro* bioactivity data and chemical structure were developed.[313] The models could be used to hazard screen large sets of chemicals for potential human toxicity, as well as to provide insights into toxicity mechanisms.[313]

In this review, we look at both *in vivo* and *in vitro* assay data, but we expect that the importance of *in vitro* data is likely to increase in future studies. A related subject is the extrapolation of in vitro assay data towards in vivo data (IVIVE).[314] This is an important area, for which machine learning has yet to be applied, and may be one to watch for the future.

The recent successes of machine learning in predictive toxicology have demonstrated that machine learning methods can generalize the data, as well as predict the toxicity potential of compounds accurately. While the results in Table 3 are generally of the single task classification method, multi-task classification/learning is also another method used in predictive toxicology. By learning tasks in parallel, multi-task learning has the potential to improve the generalization of the model, provided that sufficient data are available, longer training times and more complex architectures are practical, and that the distinct datasets are sufficiently closely linked for the models to be related.[315] The assignment of more than one label to each instance might improve model performance by increasing model complexity. Several papers have explained multi-task classification, and this could be an alternative approach to take instead of varying the machine learning method used.[19,315–317] For example, the work by Mayr et al. in 2016 used multi-task learning and found that it enhances the model performance for 10 out of 12 assays.[17] In another study, Wu et al. used multi-task learning on four different quantitative toxicity datasets.[10] Generally, it was observed that the multi-task models performed better than single task models when suitable data were available.[10] Hence, by using machine learning in predictive toxicology, the advantages of the various machine learning methods can be applied to the databases of drugs and drug-like compounds. This would lead to even more efficient and accurate predictions for drug toxicity.

However, there are some considerations when using machine learning to predict toxicity, and even for using machine learning in general. Firstly, care must be taken when processing the dataset for input into the model. This is because the results obtained by machine learning are highly dependent on the characteristics of the input data. For example, a dataset containing a significant majority of non-toxic compounds will likely result in a model that is skewed towards predicting non-toxicity. This would also affect the model's ability to predict toxicity on unseen data. This is part of the imbalanced data problem in predictive toxicology, as most of the available data are about toxic compounds, while the number of non-toxic compounds is significantly fewer. Other than the class distribution in the dataset, the size of the dataset also needs to be considered. Generally, machine learning methods perform better and can generalize better as the size of the dataset increases, provided that the quality of the data does not diminish.

Another issue to take note of for machine learning concerns overfitting when training the model on the dataset. Overfitting of a model means that the model learns the training data too well; that is the model has memorised the training data. This affects the results when predicting new, unseen test data, in which case the model normally performs badly or more poorly than expected. In contrast, the results for the training data would score very well across most common performance metrics. During model training, overfitting can be identified as the region after the point at which the loss function reaches the minimum and is represented by an increase in the loss function after the minimum point. Another indicator of overfitting is when there is a large difference between the training and test accuracy, or when the gap between these two metrics increases during model training.

Some methods to tackle overfitting include sampling techniques for imbalanced data, which have been mentioned in an earlier section. In contrast, regularization is commonly used during training to handle the overfitting problem. Regularization aims to minimise the loss function, subject to a regularization condition on the model parameter, where the regularization parameter is represented by λ.[318–320] The first regularization method is early stopping, which as its name suggests, is stopping the training of the model before the overfitting region.[321,322] However, this can only be carried out if there is a clear identification of the overfitting region. Another two common regularization methods are L_1 and L_2 regularization. L_1 regularization uses a penalty term which encourages the sum of the absolute values of the parameters to be small while L_2 regularization encourages the sum of the squares of the parameters to be small.[323] More details about regularization can be found in the literature.[318–327]

Therefore, overfitting is an issue of immense importance in the development of all computational models, and complex machine learning algorithms are often considered to have the most danger of overfitting. Modellers often use regularization or external validation to limit the effects of overfitting, but these do not establish how applicable a model is to an incoming novel chemical. For this an applicability domain is appropriate, and these domains are common practice in the QSAR field and have been identified as a key element in *in silico* toxicology modelling.[328] The use of applicability domains does not yet seem to be commonplace in the development of machine learning algorithms. For additional acceptance in the field of toxicology, particularly at a regulatory level, machine learning models should aim to meet these requirements.

With the vast quantity of data available, how should one then choose a machine learning model for their dataset? In our opinion, there is perhaps no best method that can generalize all datasets, but rather only the most suitable method. For data with a strong correlation between features, regression models, kNNs, NB models, or SVMs seem to be the most suitable due to their characteristics. DTs or RF can be considered for noisy data as they are generally more resistant to noise while being able to output results efficiently. To model complex problems which typical machine learning methods are unable to handle, deep learning (for example DNNs), as well as ensemble learning seem to be the most suitable machine learning method of choice due to their more complex model architecture. Images are handled by the specialised CNNs, while ANNs are a general machine learning method that can be used to model data. Therefore, understanding the data well is the first step to building a successful model in machine learning.

FUTURE OUTLOOK AND CONCLUSION

Thus far, machine learning has been discussed, while the recent results of machine learning in predictive toxicology have been summarized and analyzed. In the future, it is expected that more models will be developed to predict toxicity, especially with technological advances which help lower computational costs and the continual development of new data sources. However, much needs to be done to address the main bottleneck facing machine learning in predictive toxicology, which is the quality and quantity of the data that is available to create datasets. While collaborations with pharmaceutical companies help mitigate part of this issue, as well as there being publicly accessible databases online, there are some gene or protein targets, or even toxicological endpoints which cannot be reliably predicted due to the lack of data. However, if a complete computational model, or more likely, a collection of computational models encompassing all of human toxicology, is to be built, these gaps in data need to be addressed. Moreover, more must be done to solve the issue of imbalanced data in predictive toxicology, an example of which is to collect, and disseminate, negative experimental results for the compounds.

Nevertheless, there have been many machine learning models that have high-performance metrics for predicting drug toxicity, which demonstrates the applicability of machine learning in predictive toxicology. However, even the best performing model type has its own set of limitations that has been covered and needs to be addressed and improved on, for machine learning in predictive toxicology to make further advances. While several common types of machine learning methods have been discussed in this review, other machine learning methods are being developed and may become influential as their efficacy is established.

Recent research in predictive toxicology heavily focuses on hepatotoxicity, carcinogenicity, cardiotoxicity, and mutagenicity while other types of toxicity are relatively less explored. While detailed models of these toxicities have been developed, this leaves a large amount of human toxicology unexplored. By understanding more about all types of toxicity (and not just the main types of toxicity), the community as a whole can move closer to the dream of eliminating *in vivo* toxicity testing by replacing it with *in silico* means.

Mechanistic understanding is also key in the future of toxicology, hence, another topic on the rise in predictive toxicology deals with adverse outcome pathways (AOPs) and molecular initiating events (MIEs).[57,58,329–334] While structure-activity relationships have been well established and researched in predictive toxicology, AOPs and MIEs are relatively less researched. The lack of data on AOPs and MIEs prove to be detrimental when trying to predict toxicological endpoints from the compound structure. Although there have been some improvements in this aspect such as the establishment of a publicly accessible AOP database (AOPWiki: https://aopwiki.org/), much still has to be done if the accurate prediction of toxicity is to become a reality.

Nevertheless, *in silico* methods have been increasingly employed in predictive toxicology, particularly during the screening process of new chemicals for safety decision making. The use of *in silico* methods such as machine learning to complement *in vitro* methods is the current status quo of the industry. While examples tend to be commercially sensitive and this data are rarely shared openly, there are some recent papers that contain some information about the use of *in silico* methods in the industry.[335–337] As the amount of available data increases, machine learning methods will likely become more attractive over expert-based systems due to their scalability. In particular, if more data can be generated or made publicly available, machine learning is also expected to perform better even if there are no improvements in the current algorithms. Perhaps, this indicates a possible direction of development for future *in silico* methods, where the focus will be on generating new data.

Even as the need for new data is highlighted, regulatory acceptance of such *in silico* methods is also key to their widespread use and acceptance in the industry and by regulators. Regulations protect consumers, ensuring products have been through a rigorous safety protocol, allowing them to use these products while feeling at ease. Protocols to establish the quality and reliability of *in silico* methods have to be created, which should also ensure that reproducible results can be obtained. *In silico* and *in vitro* approaches must be demonstrated to be able to produce risk assessments as rigorous as those of traditional methods with clear mechanistic understanding throughout. In order to help bridge these gaps, machine learning algorithms should be combined with more traditional computational approaches such as read across and experimental *in vitro* studies as part of a weight of evidence approach. Case studies will need to be constructed and presented to regulators to gain confidence. More effort thus has to be put in to develop such studies and encourage their usage throughout industry and academia if there is to be even more progress in the use of *in silico* methods.

Currently, machine learning algorithms are significantly useful, and this is highlighted by the recent successes of such methods in predictive toxicology. However, these methods still have much potential to be unlocked, which can only be done if the issues of insufficient high-quality data and regulatory acceptance are resolved. Nevertheless, the future of machine learning applications in predictive toxicology is bright, and we envision that *in silico* methods, in particular machine learning algorithms, will be increasingly used in the industry and in academia to complement the use of *in vitro* methods.

AUTHOR INFORMATION

Corresponding Author

*E-mail: jmg11@cam.ac.uk. Phone: 01223 336434.

Author Contributions

The manuscript was written through the contributions of all authors. All authors have approved the final version of the manuscript.

Funding Sources

The authors acknowledge the financial support of Unilever.

ABBREVIATIONS

ADME, absorption, distribution, metabolism, and excretion; ANN, artificial neural network; AOP, adverse outcome pathway; AUC, area under the receiver operating characteristic curve; CCR, corrected classification rate; CNN, convolutional neural network; DNN, deep neural network; DT, decision tree; ECFP, extended connectivity fingerprint; EPA, Environmental Protection Agency; GEPSVM, generalized eigenvalue proximal support vector machine; GPU, graphical processing unit; IVIVE, *in vitro* to *in vivo* extrapolation; kNN, k nearest neighbor; LASSO, least absolute shrinkage and selection operator; LM, Levenberg-Marquardt; LMO-CV, leave-many-out cross-validation; LOO, leave one out; LSTM, long short-term memory; MACCS, Molecular ACCess System; MCCV, Monte Carlo cross-validation; MIE, molecular initiating event; NB, naïve Bayes; NN, neural network; QPP, quadratic programming problem; QSAR, quantitative structure-activity relationship; ReLU, rectified linear unit; RF, random forest; RMSE, root-mean-square error; SE, sensitivity; SP, specificity; SVM, support vector machine; SVR, support vector regression; TWSVM, twin support vector machine

SUPPORTING INFORMATION

Further data and discussion on *in vitro* and *in vivo* data, and between animal and human studies, is available in the supporting information.

REFERENCES

(1) Angermueller, C.; Pärnamaa, T.; Parts, L.; Stegle, O. Deep Learning for Computational Biology. *Mol. Syst. Biol.* **2016**, *12* (7), 878.

(2) Dana, D.; Gadhiya, S. V.; Surin, L. G. S.; Li, D.; Naaz, F.; Ali, Q.; Paka, L.; Yamin, M. A.; Narayan, M.; Goldberg, I. D.; Narayan, P. Deep Learning in Drug Discovery and Medicine; Scratching the Surface. *Molecules* **2018**, *23* (9), 2384.

(3) Chen, J.; Tang, Y. Y.; Fang, B.; Guo, C. In Silico Prediction of Toxic Action Mechanisms of Phenols for Imbalanced Data with Random Forest Learner. *J. Mol. Graph. Model.* **2012**, *35*, 21–27.

(4) Merlot, C. Computational Toxicology-a Tool for Early Safety Evaluation. *Drug Discovery Today* **2010**, *15* (1-2), 16–22.

(5) Tang, W.; Chen, J.; Wang, Z.; Xie, H.; Hong, H. Deep Learning for Predicting Toxicity of Chemicals: A Mini Review. *J. Environ. Sci. Heal. - Part C Environ. Carcinog. Ecotoxicol. Rev.* **2018**, *36* (4), 252–271.

(6) Koutsoukas, A.; St. Amand, J.; Mishra, M.; Huan, J. Predictive Toxicology: Modeling Chemical Induced Toxicological Response Combining Circular Fingerprints with Random Forest and Support Vector Machine. *Front. Environ. Sci.* **2016**, *4*, 11.

(7) Cao, D. S.; Zhao, J. C.; Yang, Y. N.; Zhao, C. X.; Yan, J.; Liu, S.; Hu, Q. N.; Xu, Q. S.; Liang, Y. Z. In Silico Toxicity Prediction by Support Vector Machine and SMILES Representation-Based String Kernel. *SAR QSAR Environ. Res.* **2012**, *23* (1–2), 141–153.

(8) Kourou, K.; Exarchos, T. P.; Exarchos, K. P.; Karamouzis, M. V.; Fotiadis, D. I. Machine Learning Applications in Cancer Prognosis and Prediction. *Computational and Structural Biotechnology Journal* **2015**, *13*, 8–17.

(9) Cao, D. S.; Dong, J.; Wang, N. N.; Wen, M.; Deng, B. C.; Zeng, W. Bin; Xu, Q. S.; Liang, Y. Z.; Lu, A. P.; Chen, A. F. In Silico Toxicity Prediction of Chemicals from EPA Toxicity Database by Kernel Fusion-Based Support Vector Machines. *Chemom. Intell. Lab. Syst.* **2015**, *146*, 494–502.

(10) Wu, K.; Wei, G. W. Quantitative Toxicity Prediction Using Topology Based Multitask Deep Neural Networks. *J. Chem. Inf. Model.* **2018**, *58* (2), 520–531.

(11) Lavecchia, A. Machine-Learning Approaches in Drug Discovery: Methods and Applications. *Drug Discovery Today* **2015**, *20* (3), 318–331.

(12) Mitchell, J. B. O. Machine Learning Methods in Chemoinformatics. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2014**, *4* (5), 468–481.

(13) Gao, M.; Igata, H.; Takeuchi, A.; Sato, K.; Ikegaya, Y. Machine Learning-Based Prediction of Adverse Drug Effects: An Example of Seizure-Inducing Compounds. *J. Pharmacol. Sci.* **2017**, *133* (2), 70–78.

(14) Lo, Y.-C.; Rensi, S. E.; Torng, W.; Altman, R. B. Machine Learning in Chemoinformatics and Drug Discovery. *Drug Discov. Today* **2018**, *23* (8), 1538-1546.

(15) Wu, Y.; Wang, G. Machine Learning Based Toxicity Prediction: From Chemical Structural Description to Transcriptome Analysis. *Int. J. Mol. Sci.* **2018**, *19* (8), 2358.

(16) Idakwo, G.; Luttrell, J.; Chen, M.; Hong, H.; Zhou, Z.; Gong, P.; Zhang, C. A Review on Machine Learning Methods for in Silico Toxicity Prediction. *J. Environ. Sci. Heal. - Part C Environ. Carcinog. Ecotoxicol. Rev.* **2018**, *36* (4), 169–191.

(17) Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S. DeepTox: Toxicity Prediction Using Deep Learning. *Front. Environ. Sci.* **2016**, *3* (FEB), 80.

(18) Zhang, L.; Zhang, H.; Ai, H.; Hu, H.; Li, S.; Zhao, J.; Liu, H. Applications of Machine Learning

Methods in Drug Toxicity Prediction. *Curr. Top. Med. Chem.* **2018**, *18* (12), 987–997.

(19) Raies, A. B.; Bajic, V. B. *In Silico* Toxicology: Comprehensive Benchmarking of Multi-Label Classification Methods Applied to Chemical Toxicity Data. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2018**, *8* (3), e1352.

(20) Pu, L.; Naderi, M.; Liu, T.; Wu, H. C.; Mukhopadhyay, S.; Brylinski, M. EToxPred: A Machine Learning-Based Approach to Estimate the Toxicity of Drug Candidates. *BMC Pharmacol. Toxicol.* **2019**, *20* (1), 2.

(21) Lysenko, A.; Sharma, A.; Boroevich, K. A.; Tsunoda, T. An Integrative Machine Learning Approach for Prediction of Toxicity-Related Drug Safety. *Life Sci. Alliance* **2018**, *1* (6).

(22) Wenlock, M. C.; Austin, R. P.; Barton, P.; Davis, A. M.; Leeson, P. D. A Comparison of Physiochemical Property Profiles of Development and Marketed Oral Drugs. *J. Med. Chem.* **2003**, *46* (7), 1250–1256.

(23) Frank, C.; Himmelstein, D. U.; Woolhandler, S.; Bor, D. H.; Wolfe, S. M.; Heymann, O.; Zallman, L.; Lasser, K. E. Era Of Faster FDA Drug Approval Has Also Seen Increased Black-Box Warnings And Market Withdrawals. *Health Aff.* **2014**, *33* (8), 1453–1459.

(24) Foufelle, F.; Fromenty, B. Role of Endoplasmic Reticulum Stress in Drug-induced Toxicity. *Pharmacol. Res. Perspect.* **2016**, *4* (1).

(25) DiMasi, J. A. Risks in New Drug Development: Approval Success Rates for Investigational Drugs. *Clin Pharmacol Ther* **2001**, *69*, 297–307.

(26) Segall, M. D.; Barber, C. Addressing Toxicity Risk When Designing and Selecting Compounds in Early Drug Discovery. *Drug Discovery Today* **2014**, *19* (5), 688–693.

(27) Fröhlich, E.; Roblegg, E. Models for Oral Uptake of Nanoparticles in Consumer Products. *Toxicology* **2012**, *291*, 10–17.

(28) Mantovani, A.; Maranghi, F.; La Rocca, C.; Tiboni, G. M.; Clementi, M. The Role of Toxicology to Characterize Biomarkers for Agrochemicals with Potential Endocrine Activities. *Reprod. Toxicol.* **2008**, *26* (1), 1–7.

(29) Smith, M.-C.; Madec, S.; Coton, E.; Hymery, N. Natural Co-Occurrence of Mycotoxins in Foods and Feeds and Their in Vitro Combined Toxicological Effects. *Toxins (Basel).* **2016**, *8* (4), 94.

(30) Rovida, C.; Asakura, S.; Daneshian, M.; Hofman-Huether, H.; Leist, M.; Meunier, L.; Reif, D.; Rossi, A.; Schmutz, M.; Valentin, J. P.; Zurlo, J.; Hartung, T. Food for Thought...: Toxicity Testing in the 21st Century beyond Environmental Chemicals. *ALTEX* **2015**, *32* (3), 171–181.

(31) Van Norman, G. A. Drugs, Devices, and the FDA: Part 1: An Overview of Approval Processes for Drugs. *JACC Basic to Transl. Sci.* **2016**, *1* (3), 170–179.

(32) Lee, K. H.; Baik, S. Y.; Lee, S. Y.; Park, C. H.; Park, P. J.; Kim, J. H. Genome Sequence Variability Predicts Drug Precautions and Withdrawals from the Market. *PLoS One* **2016**, *11* (9), e0162135.

(33) McNaughton, R.; Huet, G.; Shakir, S. An Investigation into Drug Products Withdrawn from the EU Market between 2002 and 2011 for Safety Reasons and the Evidence Used to Support the Decision-Making. *BMJ Open* **2014**, *4* (1).

(34) Park, B. K.; Boobis, A.; Clarke, S.; Goldring, C. E. P.; Jones, D.; Kenna, J. G.; Lambert, C.; Laverty, H. G.; Naisbitt, D. J.; Nelson, S.; Nicoll-Griffith, D. A.; Obach, R. S.; Routledge, P.; Smith, D. A.; Tweedie, D. J.; Vermeulen, N.; Williams, D. P.; Wilson, I. D.; Baillie, T. A. Managing the Challenge of Chemically Reactive Metabolites in Drug Development. *Nature Reviews Drug Discovery* **2011**, *10* (4), 292–306.

(35) O'Brien, P. J.; Siraki, A. G.; Shangari, N. Aldehyde Sources, Metabolism, Molecular Toxicity Mechanisms, and Possible Effects on Human Health. *Crit. Rev. Toxicol.* **2005**, *35* (7), 609–662.

(36) Mak, I. W. Y.; Evaniew, N.; Ghert, M. Lost in Translation: Animal Models and Clinical Trials in Cancer Treatment. *American Journal of Translational Research* **2014**, *6* (2), 114.

(37) Smietana, K.; Siatkowski, M.; Møller, M. Trends in Clinical Success Rates. *Nature reviews. Drug Discovery* **2016**, *15* (6), 379-380.

(38) Evens, R. P. Pharma Success in Product Development—Does Biotechnology Change the Paradigm in Product Development and Attrition. *AAPS J.* **2016**, *18* (1), 281–285.

(39) Festing, S.; Wilkinson, R. The Ethics of Animal Research. *EMBO Rep.* **2007**, *8* (6), 526–530.

(40) Varga, O. E.; Hansen, A. K.; Sandøe, P.; Olsson, I. A. S. Validating Animal Models for Preclinical Research: A Scientific and Ethical Discussion. *ATLA Alternatives to Laboratory Animals* **2010**, *38* (3), 245–248.

(41) Adler, S.; Basketter, D.; Creton, S.; Pelkonen, O.; Van Benthem, J.; Zuang, V.; Andersen, K. E.; Angers-Loustau, A.; Aptula, A.; Bal-Price, A.; Benfenati, E.; Bernauer, U.; Bessems, J.; Bois, F. Y.; Boobis, A.; Brandon, E.; Bremer, S.; Broschard, T.; Casati, S.; Coecke, S.; Corvi, R.; Cronin, M.; Daston, G.; Dekant, W.; Felter, S.; Grignard, E.; Gundert-Remy, U.; Heinonen, T.; Kimber, I.; Kleinjans, J.; Komulainen, H.; Kreiling, R.; Kreysa, J.; Leite, S. B.; Loizou, G.; Maxwell, G.; Mazzatorta, P.; Munn, S.; Pfuhler, S.; Phrakonkham, P.; Piersma, A.; Poth, A.; Prieto, P.; Repetto, G.; Rogiers, V.; Schoeters, G.; Schwarz, M.; Serafimova, R.; Tähti, H.; Testai, E.; Van Delft, J.; Van Loveren, H.; Vinken, M.; Worth, A.; Zaldivar, J. M. Alternative (Non-Animal) Methods for Cosmetics Testing: Current Status and Future Prospects-2010. *Archives of Toxicology* **2011**, *85* (5), 367–485.

(42) Burden, N.; Mahony, C.; Müller, B. P.; Terry, C.; Westmoreland, C.; Kimber, I. Aligning the 3Rs with New Paradigms in the Safety Assessment of Chemicals. *Toxicology* **2015**, *330*, 62–66.

(43) Sullivan, K. M.; Manuppello, J. R.; Willett, C. E. Building on a Solid Foundation: SAR and QSAR as a Fundamental Strategy to Reduce Animal Testing. *SAR QSAR Environ. Res.* **2014**, *25* (5), 357–365.

(44) Chapman, K. L.; Holzgrefe, H.; Black, L. E.; Brown, M.; Chellman, G.; Copeman, C.; Couch, J.; Creton, S.; Gehen, S.; Hoberman, A.; Kinter, L. B.; Madden, S.; Mattis, C.; Stemple, H. A.; Wilson, S. Pharmaceutical Toxicology: Designing Studies to Reduce Animal Use, While Maximizing Human Translation. *Regul. Toxicol. Pharmacol.* **2013**, *66* (1), 88–103.

(45) Knudsen, T. B.; Keller, D. A.; Sander, M.; Carney, E. W.; Doerrer, N. G.; Eaton, D. L.; Fitzpatrick, S. C.; Hastings, K. L.; Mendrick, D. L.; Tice, R. R.; Watkins, P. B.; Whelan, M. FutureTox II: In Vitro Data and In Silico Models for Predictive Toxicology. *Toxicol. Sci.* **2015**, *143* (2), 256–267.

(46) Li, X.; Chen, L.; Cheng, F.; Wu, Z.; Bian, H.; Xu, C.; Li, W.; Liu, G.; Shen, X.; Tang, Y. *In Silico* Prediction of Chemical Acute Oral Toxicity Using Multi-Classification Methods. *J. Chem. Inf. Model.* **2014**, *54* (4), 1061–1069.

(47) Goh, G. B.; Hodas, N. O.; Vishnu, A. Deep Learning for Computational Chemistry. *Journal of computational chemistry* **2017**, *38* (16), 1291–1307.

(48) Raies, A. B.; Bajic, V. B. In Silico Toxicology: Computational Methods for the Prediction of Chemical Toxicity. *Wiley Interdiscip. Rev. Comput. Mol. Sci.* **2016**, *6* (2), 147–172.

(49) Raunio, H. In Silico Toxicology – Non-Testing Methods. *Front. Pharmacol.* **2011**, *2*, 33.

(50) Greene, N.; Judson, P. N.; Langowski, J. J.; Marchant, C. A. Knowledge-Based Expert Systems for Toxicity and Metabolism Prediction: DEREK, StAR and METEOR. *SAR QSAR Environ. Res.* **1999**, *10* (2–3), 299–314.

(51) Marchant, C. A.; Briggs, K. A.; Long, A. In Silico Tools for Sharing Data and Knowledge on Toxicity and Metabolism: Derek for Windows, Meteor, and Vitic. *Toxicology Mechanisms and Methods* **2008**, *18* (2-3), 177–187.

(52) Mombelli, E.; Devillers, J. Evaluation of the OECD (Q)SAR Application Toolbox and Toxtree for Predicting and Profiling the Carcinogenic Potential of Chemicals. *SAR QSAR Environ. Res.* **2010**, *21* (7–8), 731–752.

(53) Dimitrov, S. D.; Diderich, R.; Sobanski, T.; Pavlov, T. S.; Chankov, G. V.; Chapkanov, A. S.; Karakolev, Y. H.; Temelkov, S. G.; Vasilev, R. A.; Gerova, K. D.; Kuseva, C. D.; Todorova, N. D.; Mehmed, A. M.; Rasenberg, M.; Mekenyan, O. G. QSAR Toolbox – Workflow and Major Functionalities. *SAR QSAR Environ. Res.* **2016**, *27* (3), 203–219.

(54) Yang, H.; Sun, L.; Li, W.; Liu, G.; Tang, Y. In Silico Prediction of Chemical Toxicity for Drug Design Using Machine Learning Methods and Structural Alerts. *Front. Chem.* **2018**, *6*, 30.

(55) Vamathevan, J.; Clark, D.; Czodrowski, P.; Dunham, I.; Ferran, E.; Lee, G.; Li, B.; Madabhushi, A.; Shah, P.; Spitzer, M.; Zhao, S. Applications of Machine Learning in Drug Discovery and Development. *Nature Reviews Drug Discovery* **2019**, *18* (6), 463–477.

(56) Baskin, I. I. Machine Learning Methods in Computational Toxicology. In *Computational Toxicology* **2018**; Humana Press, New York, NY. 119–139.

(57) Villeneuve, D. L.; Crump, D.; Garcia-Reyero, N.; Hecker, M.; Hutchinson, T. H.; LaLone, C. A.; Landesmann, B.; Lettieri, T.; Munn, S.; Nepelska, M.; Ottinger, M. A.; Vergauwen, L.; Whelan, M. Adverse Outcome Pathway (AOP) Development I: Strategies and Principles. *Toxicol. Sci.* **2014**, *142* (2), 312–320.

(58) Knapen, D.; Vergauwen, L.; Villeneuve, D. L.; Ankley, G. T. The Potential of AOP Networks for Reproductive and Developmental Toxicity Assay

Development. *Reprod. Toxicol.* **2015**, *56*, 52–55.

(59) Richard, A. M.; Judson, R. S.; Houck, K. A.; Grulke, C. M.; Volarath, P.; Thillainadarajah, I.; Yang, C.; Rathman, J.; Martin, M. T.; Wambaugh, J. F.; Knudsen, T. B.; Kancherla, J.; Mansouri, K.; Patlewicz, G.; Williams, A. J.; Little, S. B.; Crofton, K. M.; Thomas, R. S. ToxCast Chemical Landscape: Paving the Road to 21st Century Toxicology. *Chemical Research in Toxicology* **2016**, *29* (8) 1225–1251.

(60) Dix, D. J.; Houck, K. A.; Martin, M. T.; Richard, A. M.; Setzer, R. W.; Kavlock, R. J. The ToxCast Program for Prioritizing Toxicity Testing of Environmental Chemicals. *Toxicol. Sci.* **2007**, *95* (1), 5–12.

(61) Gaulton, A.; Hersey, A.; Nowotka, M.; Bento, A. P.; Chambers, J.; Mendez, D.; Mutowo, P.; Atkinson, F.; Bellis, L. J.; Cibrián-Uhalte, E.; Davies, M.; Dedman, N.; Karlsson, A.; Magariños, M. P.; Overington, J. P.; Papadatos, G.; Smit, I.; Leach, A. R. The ChEMBL Database in 2017. *Nucleic Acids Res.* **2017**, *45* (D1), D945–D954.

(62) Gaulton, A.; Bellis, L. J.; Bento, A. P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani, B.; Overington, J. P. ChEMBL: A Large-Scale Bioactivity Database for Drug Discovery. *Nucleic Acids Res.* **2012**, *40* (D1), D1100–D1107.

(63) Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P. The ChEMBL Bioactivity Database: An Update. *Nucleic Acids Res.* **2014**, *42* (D1), D1083–D1090.

(64) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; Zaslavsky, L.; Zhang, J.; Bolton, E. E. PubChem 2019 Update: Improved Access to Chemical Data. *Nucleic Acids Res.* **2019**, *47*.

(65) Bowes, J.; Brown, A. J.; Hamon, J.; Jarolimek, W.; Sridhar, A.; Waldron, G.; Whitebread, S. Reducing Safety-Related Drug Attrition: The Use of in Vitro Pharmacological Profiling. *Nature Reviews Drug Discovery* **2012**, *11* (12), 909–922.

(66) Schulz, M.; Schmoldt, A.; Schulz, M. Therapeutic and Toxic Blood Concentrations of More than 800 Drugs and Other Xenobiotics. *Die Pharmazie-An International Journal of Pharmaceutical Sciences* **2003**, *58* (7), 447-474.

(67) Sagar, S.; Kaur, M.; Radovanovic, A.; Bajic, V. B. Dragon Exploration System on Marine Sponge Compounds Interactions. *J. Cheminform.* **2013**, *5* (1), 11.

(68) Cases, M.; Pastor, M.; Sanz, F. The ETOX Library of Public Resources for in Silico Toxicity Prediction. *Molecular Informatics* **2013**, *32* (1), 24–35.

(69) Wishart, D. S.; Knox, C.; Guo, A. C.; Eisner, R.; Young, N.; Gautam, B.; Hau, D. D.; Psychogios, N.; Dong, E.; Bouatra, S.; Mandal, R.; Sinelnikov, I.; Xia, J.; Jia, L.; Cruz, J. A.; Lim, E.; Sobsey, C. A.; Shrivastava, S.; Huang, P.; Liu, P.; Fang, L.; Peng, J.; Fradette, R.; Cheng, D.; Tzur, D.; Clements, M.; Lewis, A.; De Souza, A.; Zuniga, A.; Dawe, M.; Xiong, Y.; Clive, D.; Greiner, R.; Nazyrova, A.; Shaykhutdinov, R.; Li, L.; Vogel, H. J.; Forsythe, I. HMDB: A Knowledgebase for the Human Metabolome. *Nucleic Acids Res.* **2009**, *37* (Database), D603–D610.

(70) Wishart, D. S.; Jewison, T.; Guo, A. C.; Wilson, M.; Knox, C.; Liu, Y.; Djoumbou, Y.; Mandal, R.; Aziat, F.; Dong, E.; Bouatra, S.; Sinelnikov, I.; Arndt, D.; Xia, J.; Liu, P.; Yallou, F.; Bjorndahl, T.; Perez-Pineiro, R.; Eisner, R.; Allen, F.; Neveu, V.; Greiner, R.; Scalbert, A. HMDB 3.0—The Human Metabolome Database in 2013. *Nucleic Acids Res.* **2012**, *41* (D1), D801–D807.

(71) Wishart, D. S.; Tzur, D.; Knox, C.; Eisner, R.; Guo, A. C.; Young, N.; Cheng, D.; Jewell, K.; Arndt, D.; Sawhney, S.; Fung, C.; Nikolai, L.; Lewis, M.; Coutouly, M.-A.; Forsythe, I.; Tang, P.; Shrivastava, S.; Jeroncic, K.; Stothard, P.; Amegbey, G.; Block, D.; Hau, D. D.; Wagner, J.; Miniaci, J.; Clements, M.; Gebremedhin, M.; Guo, N.; Zhang, Y.; Duggan, G. E.; MacInnis, G. D.; Weljie, A. M.; Dowlatabadi, R.; Bamforth, F.; Clive, D.; Greiner, R.; Li, L.; Marrie, T.; Sykes, B. D.; Vogel, H. J.; Querengesser, L. HMDB: The Human Metabolome Database. *Nucleic Acids Res.* **2007**, *35* (Database), D521–D526.

(72) Han, H.; Wang, W. Y.; Mao, B. H. Borderline-SMOTE: A New over-Sampling Method in Imbalanced Data Sets Learning. In *International conference on intelligent computing* **2005**, 878–887.

(73) Chen, J. J.; Tsai, C. A.; Young, J. F.; Kodell, R. L. Classification Ensembles for Unbalanced Class Sizes in Predictive Toxicology. *SAR QSAR Environ. Res.* **2005**, *16* (6), 517–529.

(74) Fernández, A.; García, S.; Herrera, F.; Chawla, N. V. SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-Year Anniversary. *Journal of Artificial Intelligence Research* **2018**, *61*, 863–905.

(75) Buda, M.; Maki, A.; Mazurowski, M. A. A Systematic Study of the Class Imbalance Problem in Convolutional Neural Networks. *Neural Networks* **2018**, *106*, 249–259.

(76) Lin, W. C.; Tsai, C. F.; Hu, Y. H.; Jhang, J. S. Clustering-Based Undersampling in Class-Imbalanced Data. *Inf. Sci. (Ny).* **2017**, *409–410*, 17–26.

(77) Boughorbel, S.; Jarray, F.; El-Anbari, M. Optimal Classifier for Imbalanced Data Using Matthews Correlation Coefficient Metric. *PLoS One* **2017**, *12* (6), e0177678.

(78) Yan, Y.; Chen, M.; Shyu, M. L.; Chen, S. C. Deep Learning for Imbalanced Multimedia Data Classification. In *IEEE International Symposium on Multimedia* **2015**, 483–488.

(79) Akbani, R.; Kwek, S.; Japkowicz, N. Applying Support Vector Machines to Imbalanced Datasets. In *European conference on machine learning* **2004**, 39–50.

(80) Batista, G. E. A. P. A.; Prati, R. C.; Monard, M. C. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *ACM SIGKDD Explor. Newsl.* **2004**, *6* (1), 20-29.

(81) Lemaˆıtre, G.; Nogueira, F.; Aridas, C. K. Imbalanced-Learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning; *The Journal of Machine Learning Research* **2017**, *18* (1), 559-563.

(82) Erickson, B. J.; Korfiatis, P.; Akkus, Z.; Kline, T. L. Machine Learning for Medical Imaging. *Radiographics* **2017**, *37* (2), 505–515.

(83) Beam, A. L.; Kohane, I. S. Big Data and Machine Learning in Health Care. *JAMA* **2018**, *319* (13), 1317–1318.

(84) Gawehn, E.; Hiss, J. A.; Schneider, G. Deep Learning in Drug Discovery. *Molecular informatics* **2016**, *35* (1), 3-14.

(85) Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T. The Rise of Deep Learning in Drug Discovery. *Drug Discovery Today* **2018**, *23* (6), 1241–1250.

(86) Nichols, J. A.; Herbert Chan, H. W.; Baker, M. A. B. Machine Learning: Applications of Artificial Intelligence to Imaging and Diagnosis. *Biophysical Reviews* **2019**, *11* (1), 111–118.

(87) Wang, X.; Wang, X.; Wilkes, D. M. Supervised Learning for Data Classification Based Object Recognition. In *Machine Learning-based Natural Scene Recognition for Mobile Robot Localization in An Unknown Environment* **2020**, 179–194.

(88) Jordan, M. I.; Mitchell, T. M. Machine Learning: Trends, Perspectives, and Prospects. *Science* **2015**, 349 (6245), 255–260.

(89) Schrider, D. R.; Kern, A. D. Supervised Machine Learning for Population Genetics: A New Paradigm. *Trends in Genetics* **2018**, *34* (4), 301–312.

(90) Libbrecht, M. W.; Noble, W. S. Machine Learning Applications in Genetics and Genomics. *Nature Reviews Genetics* **2015**, *16* (6), 321–332.

(91) Längkvist, M.; Karlsson, L.; Loutfi, A. A Review of Unsupervised Feature Learning and Deep Learning for Time-Series Modeling. *Pattern Recognit. Lett.* **2014**, *42* (1), 11–24.

(92) Le, Q. V. Building High-Level Features Using Large Scale Unsupervised Learning. In *IEEE International Conference on Acoustics, Speech and Signal Processing* **2013**, 8595–8598.

(93) Raina, R.; Madhavan, A.; Ng, A. Y. Large-Scale Deep Unsupervised Learning Using Graphics Processors. In *Proceedings of the 26th annual international conference on machine learning* **2009**, 873-880.

(94) van Engelen, J. E.; Hoos, H. H. A Survey on Semi-Supervised Learning. *Mach. Learn.* **2019**, *109* (2), 373-440.

(95) Guo, B.; Tao, H.; Hou, C.; Yi, D. Semi-Supervised Multi-Label Feature Learning via Label Enlarged Discriminant Analysis. *Knowl. Inf. Syst.* **2019**, 1-35.

(96) Kostopoulos, G.; Kotsiantis, S.; Fazakis, N.; Koutsonikos, G.; Pierrakeas, C. A Semi-Supervised Regression Algorithm for Grade Prediction of Students in Distance Learning Courses. *Int. J. Artif. Intell. Tools* **2019**, *28* (4).

(97) Zhou, L.; Liu, Z.; Tan, H.; Xie, X. Semisupervised Learning with Adversarial Training among Joint Distributions. *J. Electron. Imaging* **2019**, *28* (05), 1.

(98) Wang, J.; Zuo, R.; Xiong, Y. Mapping Mineral Prospectivity via Semi-Supervised Random Forest. *Nat. Resour. Res.* **2019**, *29* (1), 189-202.

(99) Huang, G.; Song, S.; Gupta, J. N. D.; Wu, C. Semi-Supervised and Unsupervised Extreme Learning Machines. *IEEE Trans. Cybern.* **2014**, *44* (12), 2405–2417.

(100) Tanha, J.; van Someren, M.; Afsarmanesh, H. Semi-Supervised Self-Training for Decision Tree Classifiers. *Int. J. Mach. Learn. Cybern.* **2017**, *8* (1),

355–370.

(101) Zhou, Z.-H. A Brief Introduction to Weakly Supervised Learning. *Natl. Sci. Rev.* **2018**, *5* (1), 44–53.

(102) Idakwo, G.; Thangapandian, S.; Luttrell, J.; Zhou, Z.; Zhang, C.; Gong, P. Deep Learning-Based Structure-Activity Relationship Modeling for Multi-Category Toxicity Classification: A Case Study of 10K Tox21 Chemicals With High-Throughput Cell-Based Androgen Receptor Bioassay Data. *Front. Physiol.* **2019**, *10*, 1044.

(103) Sun, L.; Yang, H.; Cai, Y.; Li, W.; Liu, G.; Tang, Y. *In Silico* Prediction of Endocrine Disrupting Chemicals Using Single-Label and Multilabel Models. *J. Chem. Inf. Model.* **2019**, *59* (3), 973–982.

(104) Jiang, C.; Yang, H.; Di, P.; Li, W.; Tang, Y.; Liu, G. In Silico Prediction of Chemical Reproductive Toxicity Using Machine Learning. *J. Appl. Toxicol.* **2019**, *39* (6), 844–854.

(105) Xu, Y.; Pei, J.; Lai, L. Deep Learning Based Regression and Multiclass Models for Acute Oral Toxicity Prediction with Automatic Chemical Feature Extraction. *J. Chem. Inf. Model.* **2017**, *57* (11), 2672–2685.

(106) Pereira, J. C.; Caffarena, E. R.; Dos Santos, C. N. Boosting Docking-Based Virtual Screening with Deep Learning. *J. Chem. Inf. Model.* **2016**, *56* (12), 2495–2506.

(107) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50* (5), 742–754.

(108) Cai, Y.; Yang, H.; Li, W.; Liu, G.; Lee, P. W.; Tang, Y. Computational Prediction of Site of Metabolism for UGT-Catalyzed Reactions. *J. Chem. Inf. Model.* **2019**, *59* (3), 1085–1095.

(109) Wang, Y.; Guo, Y.; Kuang, Q.; Pu, X.; Ji, Y.; Zhang, Z.; Li, M. A Comparative Study of Family-Specific Protein-Ligand Complex Affinity Prediction Based on Random Forest Approach. *J. Comput. Aided. Mol. Des.* **2015**, *29* (4), 349–360.

(110) Bartlett, M. S.; Littlewort, G.; Frank, M.; Lainscsek, C.; Fasel, I.; Movellan, J. Recognizing Facial Expression: Machine Learning and Application to Spontaneous Behavior. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2005**, *2*, 568–573.

(111) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45* (1), 5–32.

(112) Kumari, P.; Nath, A.; Chaube, R. Identification of Human Drug Targets Using Machine-Learning Algorithms. *Comput. Biol. Med.* **2015**, *56*, 175–181.

(113) Fooshee, D.; Mood, A.; Gutman, E.; Tavakoli, M.; Urban, G.; Liu, F.; Huynh, N.; Van Vranken, D.; Baldi, P. Deep Learning for Chemical Reaction Prediction. *Mol. Syst. Des. Eng.* **2018**, *3* (3), 442–452.

(114) Patil, S. R.; Suralkar, S. R. Neural Network Based Fingerprint Classification. *International Journal of Science and Research* **2013**, *2* (1).

(115) Jha, D.; Ward, L.; Paul, A.; Liao, W. keng; Choudhary, A.; Wolverton, C.; Agrawal, A. ElemNet: Deep Learning the Chemistry of Materials From Only Elemental Composition. *Sci. Rep.* **2018**, *8* (1).

(116) Chandrashekar, G.; Sahin, F. A Survey on Feature Selection Methods. *Comput. Electr. Eng.* **2014**, *40* (1), 16–28.

(117) Blum, A. L.; Langley, P. Selection of Relevant Features and Examples in Machine Learning. *Artif. Intell.* **1997**, *97* (1–2), 245–271.

(118) Korkmaz, S.; Zararsiz, G.; Goksuluk, D. Drug/Nondrug Classification Using Support Vector Machines with Various Feature Selection Strategies. *Comput. Methods Programs Biomed.* **2014**, *117* (2), 51–60.

(119) Zhang, P.; Wang, F.; Hu, J.; Sorrentino, R. Label Propagation Prediction of Drug-Drug Interactions Based on Clinical Side Effects. *Sci. Rep.* **2015**, *5*.

(120) Rao, R. B.; Fung, G.; Rosales, R. On the Dangers of Cross-Validation. An Experimental Evaluation. In *Proceedings of the 2008 SIAM international conference on data mining* **2008**, 588–596.

(121) Tan, N. X.; Li, P.; Rao, H. B.; Li, Z. R.; Li, X. Y. Prediction of the Acute Toxicity of Chemical Compounds to the Fathead Minnow by Machine Learning Approaches. *Chemom. Intell. Lab. Syst.* **2010**, *100* (1), 66–73.

(122) Wen, M.; Zhang, Z.; Niu, S.; Sha, H.; Yang, R.; Yun, Y.; Lu, H. Deep-Learning-Based Drug-Target Interaction Prediction. *J. Proteome Res.* **2017**, *16* (4), 1401–1409.

(123) Gayvert, K. M.; Madhukar, N. S.; Elemento, O. A Data-Driven Approach to Predicting Successes and Failures of Clinical Trials. *Cell Chem. Biol.* **2016**, *23* (10), 1294–1301.

(124) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving beyond Fingerprints. *J. Comput. Aided. Mol. Des.*

2016, *30* (8), 595–608.

(125) Cereto-Massagué, A.; Ojeda, M. J.; Valls, C.; Mulero, M.; Garcia-Vallvé, S.; Pujadas, G. Molecular Fingerprint Similarity Search in Virtual Screening. *Methods* **2015**, *71*, 58–63.

(126) Durant, J. L.; Leland, B. A.; Henry, D. R.; Nourse, J. G. Reoptimization of MDL Keys for Use in Drug Discovery. *J. Chem. Inf. Comput. Sci.* **2002**, *42* (6), 1273–1280.

(127) Schaffer, C. Selecting a Classification Method by Cross-Validation. *Mach. Learn.* **1993**, *13* (1), 135–143.

(128) Kerns, S. L.; Kundu, S.; Oh, J. H.; Singhal, S. K.; Janelsins, M.; Travis, L. B.; Deasy, J. O.; Janssens, A. C. J. E.; Ostrer, H.; Parliament, M.; Usmani, N.; Rosenstein, B. S. The Prediction of Radiotherapy Toxicity Using Single Nucleotide Polymorphism-Based Models: A Step Toward Prevention. In *Seminars in radiation oncology* **2015**, *25* (4), 281–291.

(129) Linden, A.; Yarnold, P. R.; Nallamothu, B. K. Using Machine Learning to Model Dose-Response Relationships. *J. Eval. Clin. Pract.* **2016**, *22* (6), 860–867.

(130) Zhong, E.; Fan, W.; Yang, Q.; Verscheure, O.; Ren, J. Cross Validation Framework to Choose amongst Models and Datasets for Transfer Learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* **2010**, 547–562.

(131) Cawley, G. C.; Talbot, N. L. C. Efficient Leave-One-out Cross-Validation of Kernel Fisher Discriminant Classifiers. *Pattern Recognit.* **2003**, *36* (11), 2585–2592.

(132) Cawley, G. C.; Talbot, N. L. C. Fast Exact Leave-One-out Cross-Validation of Sparse Least-Squares Support Vector Machines. *Neural Networks* **2004**, *17* (10), 1467–1475.

(133) Chen, T.; Cao, Y.; Zhang, Y.; Liu, J.; Bao, Y.; Wang, C.; Jia, W.; Zhao, A. Random Forest in Clinical Metabolomics for Phenotypic Discrimination and Biomarker Selection. *Evidence-based Complement. Altern. Med.* **2013**, *2013*.

(134) He, H.; Bai, Y.; Garcia, E. A.; Li, S. ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning. In *IEEE international joint conference on neural networks* **2008**, 1322–1328.

(135) Graser, J.; Kauwe, S. K.; Sparks, T. D. Machine Learning and Energy Minimization Approaches for Crystal Structure Predictions: A Review and New Horizons. *Chemistry of Materials* **2018**, *30* (11), 3601–3612.

(136) An, S.; Liu, W.; Venkatesh, S. Fast Cross-Validation Algorithms for Least Squares Support Vector Machine and Kernel Ridge Regression. *Pattern Recognit.* **2007**, *40* (8), 2154–2162.

(137) Rácz, A.; Bajusz, D.; Héberger, K. SAR and QSAR in Environmental Research Modelling Methods and Cross-Validation Variants in QSAR: A Multi-Level Analysis. *SAR QSAR Environ. Res.* **2018**, *29* (9), 661–674.

(138) Xu, Q. S.; Liang, Y. Z. Monte Carlo Cross Validation. *Chemom. Intell. Lab. Syst.* **2001**, *56* (1), 1–11.

(139) He, S.; Ye, T.; Wang, R.; Zhang, C.; Zhang, X.; Sun, G.; Sun, X. An In Silico Model for Predicting Drug-Induced Hepatotoxicity. *Int. J. Mol. Sci.* **2019**, *20* (8), 1897.

(140) Liu, J.; Mansouri, K.; Judson, R. S.; Martin, M. T.; Hong, H.; Chen, M.; Xu, X.; Thomas, R. S.; Shah, I. Predicting Hepatotoxicity Using ToxCast in Vitro Bioactivity and Chemical Structure. *Chem. Res. Toxicol.* **2015**, *28* (4), 738–751.

(141) Liu, J.; Patlewicz, G.; Williams, A. J.; Thomas, R. S.; Shah, I. Predicting Organ Toxicity Using in Vitro Bioactivity Data and Chemical Structure. *Chem. Res. Toxicol.* **2017**, *30* (11), 2046–2059.

(142) Zhang, H.; Cao, Z. X.; Li, M.; Li, Y. Z.; Peng, C. Novel Naïve Bayes Classification Models for Predicting the Carcinogenicity of Chemicals. *Food Chem. Toxicol.* **2016**, *97*, 141–149.

(143) Zhang, H.; Ren, J. X.; Kang, Y. L.; Bo, P.; Liang, J. Y.; Ding, L.; Kong, W. B.; Zhang, J. Development of Novel in Silico Model for Developmental Toxicity Assessment by Using Naïve Bayes Classifier Method. *Reprod. Toxicol.* **2017**, *71*, 8–15.

(144) Zhang, H.; Ding, L.; Zou, Y.; Hu, S. Q.; Huang, H. G.; Kong, W. B.; Zhang, J. Predicting Drug-Induced Liver Injury in Human with Naïve Bayes Classifier Approach. *J. Comput. Aided. Mol. Des.* **2016**, *30* (10), 889–898.

(145) Schrey, A. K.; Nickel-Seeber, J.; Drwal, M. N.; Zwicker, P.; Schultze, N.; Haertel, B.; Preissner, R. Computational Prediction of Immune Cell Cytotoxicity. *Food Chem. Toxicol.* **2017**, *107*, 150–166.

(146) Zhang, H.; Kang, Y. L.; Zhu, Y. Y.; Zhao, K. X.; Liang, J. Y.; Ding, L.; Zhang, T. G.; Zhang, J. Novel Naïve Bayes Classification Models for Predicting

the Chemical Ames Mutagenicity. *Toxicol. Vitr.* **2017**, *41*, 56–63.

(147) Seal, A.; Passi, A.; Abdul Jaleel, U. C.; Wild, D. J.; Consortium, O. S. D. D. In-Silico Predictive Mutagenicity Model Generation Using Supervised Learning Approaches. *J. Cheminform.* **2012**, *4* (1), 10.

(148) Zhang, H.; Yu, P.; Ren, J. X.; Li, X. B.; Wang, H. L.; Ding, L.; Kong, W. B. Development of Novel Prediction Model for Drug-Induced Mitochondrial Toxicity by Using Naïve Bayes Classifier Method. *Food Chem. Toxicol.* **2017**, *110*, 122–129.

(149) Zhang, H.; Yu, P.; Zhang, T. G.; Kang, Y. L.; Zhao, X.; Li, Y. Y.; He, J. H.; Zhang, J. In Silico Prediction of Drug-Induced Myelotoxicity by Using Naïve Bayes Method. *Mol. Divers.* **2015**, *19* (4), 945–953.

(150) Su, R.; Li, Y.; Zink, D.; Loo, L.-H. Supervised Prediction of Drug-Induced Nephrotoxicity Based on Interleukin-6 and -8 Expression Levels. *BMC Bioinformatics* **2014**, *15* (S16), S16.

(151) Zhang, H.; Ma, J. X.; Liu, C. T.; Ren, J. X.; Ding, L. Development and Evaluation of in Silico Prediction Model for Drug-Induced Respiratory Toxicity by Using Naïve Bayes Classifier Method. *Food Chem. Toxicol.* **2018**, *121*, 593–603.

(152) Zhang, H.; Ren, J. X.; Ma, J. X.; Ding, L. Development of an in Silico Prediction Model for Chemical-Induced Urinary Tract Toxicity by Using Naïve Bayes Classifier. *Mol. Divers.* **2019**, *23* (2), 381–392.

(153) Pella, A.; Cambria, R.; Riboldi, M.; Jereczek-Fossa, B. A.; Fodor, C.; Zerini, D.; Torshabi, A. E.; Cattani, F.; Garibaldi, C.; Pedroli, G.; Baroni, G.; Orecchia, R. Use of Machine Learning Methods for Prediction of Acute Toxicity in Organs at Risk Following Prostate Radiotherapy. *Med. Phys.* **2011**, *38* (6Part1), 2859–2867.

(154) Zhang, L.; Ai, H.; Chen, W.; Yin, Z.; Hu, H.; Zhu, J.; Zhao, J.; Zhao, Q.; Liu, H. CarcinoPred-EL: Novel Models for Predicting the Carcinogenicity of Chemicals Using Molecular Fingerprints and Ensemble Learning Methods. *Sci. Rep.* **2017**, *7* (1).

(155) Shen, M.; Su, B.-H.; Esposito, E. X.; Hopfinger, A. J.; Tseng, Y. J. A Comprehensive Support Vector Machine Binary HERG Classification Model Based on Extensive but Biased End Point HERG Data Sets. *Chem. Res. Toxicol.* **2011**, *24* (6), 934–949.

(156) Li, X.; Chen, Y.; Song, X.; Zhang, Y.; Li, H.; Zhao, Y. The Development and Application of: In Silico Models for Drug Induced Liver Injury. *RSC Adv.* **2018**, *8* (15), 8101–8111.

(157) Sharma, A.; Kumar, R.; Varadwaj, P. K.; Ahmad, A.; Ashraf, G. M. A Comparative Study of Support Vector Machine, Artificial Neural Network and Bayesian Classifier for Mutagenicity Prediction. *Interdiscip Sci Comput Life Sci* **2011**, *3* (3), 232–239.

(158) Zhou, S.; Li, G. B.; Huang, L. Y.; Xie, H. Z.; Zhao, Y. L.; Chen, Y. Z.; Li, L. L.; Yang, S. Y. A Prediction Model of Drug-Induced Ototoxicity Developed by an Optimal Support Vector Machine (SVM) Method. *Comput. Biol. Med.* **2014**, *51*, 122–127.

(159) Lee, H.-M.; Yu, M.-S.; Kazmi, S. R.; Oh, S. Y.; Rhee, K.-H.; Bae, M.-A.; Lee, B. H.; Shin, D.-S.; Oh, K.-S.; Ceong, H.; Lee, D.; Na, D. Computational Determination of HERG-Related Cardiotoxicity of Drug Candidates. *BMC Bioinformatics* **2019**, *20* (S10), 250.

(160) Siramshetty, V. B.; Chen, Q.; Devarakonda, P.; Preissner, R. The Catch-22 of Predicting HERG Blockade Using Publicly Accessible Bioactivity Data. *J. Chem. Inf. Model.* **2018**, *58* (6), 1224–1233.

(161) Zhang, Y.; Zhao, J.; Wang, Y.; Fan, Y.; Zhu, L.; Yang, Y.; Chen, X.; Lu, T.; Chen, Y.; Liu, H. Prediction of HERG K+ Channel Blockage Using Deep Neural Networks. *Chem. Biol. Drug Des.* **2019**, *94* (5), 1973–1985.

(162) Kim, E.; Nam, H. Prediction Models for Drug-Induced Hepatotoxicity by Using Weighted Molecular Fingerprints. *BMC Bioinformatics* **2017**, *18* (S7), 227.

(163) Ai, H.; Chen, W.; Zhang, L.; Huang, L.; Yin, Z.; Hu, H.; Zhao, Q.; Zhao, J.; Liu, H. Predicting Drug-Induced Liver Injury Using Ensemble Learning Methods and Molecular Fingerprints. *Toxicol. Sci.* **2018**, *165* (1), 100–107.

(164) Kandasamy, K.; Chuah, J. K. C.; Su, R.; Huang, P.; Eng, K. G.; Xiong, S.; Li, Y.; Chia, C. S.; Loo, L. H.; Zink, D. Prediction of Drug-Induced Nephrotoxicity and Injury Mechanisms with Human Induced Pluripotent Stem Cell-Derived Cells and Machine Learning Methods. *Sci. Rep.* **2015**, *5*, 12337.

(165) Cai, C.; Guo, P.; Zhou, Y.; Zhou, J.; Wang, Q.; Zhang, F.; Fang, J.; Cheng, F. Deep Learning-Based Prediction of Drug-Induced Cardiotoxicity. *J. Chem. Inf. Model.* **2019**, *59* (3), 1073–1084.

(166) Xu, Y.; Dai, Z.; Chen, F.; Gao, S.; Pei, J.; Lai, L. Deep Learning for Drug-Induced Liver Injury. *J.*

Chem. Inf. Model. **2015**, *55* (10), 2085–2093.

(167) Xu, C.; Cheng, F.; Chen, L.; Du, Z.; Li, W.; Liu, G.; Lee, P. W.; Tang, Y. In Silico Prediction of Chemical Ames Mutagenicity. *J. Chem. Inf. Model.* **2012**, *52* (11), 2840–2847.

(168) Fernandez, M.; Ban, F.; Woo, G.; Hsing, M.; Yamazaki, T.; LeBlanc, E.; Rennie, P. S.; Welch, W. J.; Cherkasov, A. Toxic Colors: The Use of Deep Learning for Predicting Toxicity of Compounds Merely from Their Graphic Images. *J. Chem. Inf. Model.* **2018**, *58* (8), 1533–1543.

(169) Yuan, Q.; Wei, Z.; Guan, X.; Jiang, M.; Wang, S.; Zhang, S.; Li, Z. Toxicity Prediction Method Based on Multi-Channel Convolutional Neural Network. *Molecules* **2019**, *24* (18), 3383.

(170) Sharma, A. K.; Srivastava, G. N.; Roy, A.; Sharma, V. K. ToxiM: A Toxicity Prediction Tool for Small Molecules Developed Using Machine Learning and Chemoinformatics Approaches. *Front. Pharmacol.* **2017**, *8*, 880.

(171) Su, R.; Wu, H.; Liu, X.; Wei, L. Predicting Drug-Induced Hepatotoxicity Based on Biological Feature Maps and Diverse Classification Strategies. *Brief. Bioinform.* **2019**.

(172) Yajima, D.; Ohkawa, T.; Muroi, K.; Imaishi, H. Predicting Toxicity of Food-Related Compounds Using Fuzzy Decision Trees. *Int. J. Biosci. Biochem. Bioinforma.* **2014**, *4* (1), 33.

(173) Hammann, F.; Schöning, V.; Drewe, J. Prediction of Clinically Relevant Drug-Induced Liver Injury from Structure Using Machine Learning. *J. Appl. Toxicol.* **2019**, *39* (3), 412–419.

(174) Li, F.; Fan, D.; Wang, H.; Yang, H.; Li, W.; Tang, Y.; Liu, G. In Silico Prediction of Pesticide Aquatic Toxicity with Chemical Category Approaches. *Toxicol. Res. (Camb).* **2017**, *6* (6), 831–842.

(175) Li, X.; Du, Z.; Wang, J.; Wu, Z.; Li, W.; Liu, G.; Shen, X.; Tang, Y. In Silico Estimation of Chemical Carcinogenicity with Binary and Ternary Classification Methods. *Mol. Inform.* **2015**, *34* (4), 228–235.

(176) Zhang, C.; Zhou, Y.; Gu, S.; Wu, Z.; Wu, W.; Liu, C.; Wang, K.; Liu, G.; Li, W.; Lee, P. W.; Tang, Y. In Silico Prediction of HERG Potassium Channel Blockage by Chemical Category Approaches. *Toxicol. Res. (Camb).* **2016**, *5* (2), 570–582.

(177) Fan, D.; Yang, H.; Li, F.; Sun, L.; Di, P.; Li, W.; Tang, Y.; Liu, G. In Silico Prediction of Chemical Genotoxicity Using Machine Learning Methods and Structural Alerts. *Toxicol. Res. (Camb).* **2018**, *7* (2), 211–220.

(178) Zhang, C.; Cheng, F.; Li, W.; Liu, G.; Lee, P. W.; Tang, Y. *In Silico* Prediction of Drug Induced Liver Toxicity Using Substructure Pattern Recognition Method. *Mol. Inform.* **2016**, *35* (3–4), 136–144.

(179) 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.; Houck, K. A.; Shobair, M.; Yang, C.; Rathman, J. F.; Yasgar, A.; Fitzpatrick, S. C.; Simeonov, A.; Thomas, R. S.; Crofton, K. M.; Paules, R. S.; Bucher, J. R.; Austin, C. P.; Kavlock, R. J.; Tice, R. R. The Tox21 10K Compound Library: Collaborative Chemistry Advancing Toxicology. *Chem. Res. Toxicol.* **2020**.

(180) Hsieh, J.-H.; Smith-Roe, S. L.; Huang, R.; Sedykh, A.; Shockley, K. R.; Auerbach, S. S.; Merrick, B. A.; Xia, M.; Tice, R. R.; Witt, K. L. Identifying Compounds with Genotoxicity Potential Using Tox21 High-Throughput Screening Assays. *ACS Publ.* **2019**, *32* (7), 1384–1401.

(181) Huang, R.; Xia, M.; Sakamuru, S.; Zhao, J.; Shahane, S. A.; Attene-Ramos, M.; Zhao, T.; Austin, C. P.; Simeonov, A. Modelling the Tox21 10 K Chemical Profiles for in Vivo Toxicity Prediction and Mechanism Characterization. *Nat. Commun.* **2016**, *7* (1), 1–10.

(182) Capuzzi, S. J.; Politi, R.; Isayev, O.; Farag, S.; Tropsha, A. QSAR Modeling of Tox21 Challenge Stress Response and Nuclear Receptor Signaling Toxicity Assays. *Front. Environ. Sci.* **2016**, *4* (FEB), 3.

(183) Judson, R.; Houck, K.; Martin, M.; Knudsen, T.; Thomas, R. S.; Sipes, N.; Shah, I.; Wambaugh, J.; Crofton, K. *In Vitro* and Modelling Approaches to Risk Assessment from the U.S. Environmental Protection Agency ToxCast Programme. *Basic Clin. Pharmacol. Toxicol.* **2014**, *115* (1), 69–76.

(184) Norinder, U.; Boyer, S. Conformal Prediction Classification of a Large Data Set of EnRvironmental Chemicals from ToxCast and Tox21 Estrogen Receptor Assays. *Chem. Res. Toxicol.* **2016**, *29* (6), 1003–1010.

(185) Sipes, N. S.; Martin, M. T.; Reif, D. M.; Kleinstreuer, N. C.; Judson, R. S.; Singh, A. V.; Chandler, K. J.; Dix, D. J.; Kavlock, R. J.; Knudsen, T. B. Predictive Models of Prenatal Developmental Toxicity from Toxcast High-Throughput Screening Data. *Toxicol. Sci.* **2011**, *124* (1), 109–127.

(186) Yang, C.; Tarkhov, A.; Rg Marusczyk, J.; Bienfait,

B.; Gasteiger, J.; Kleinoeder, T.; Magdziarz, T.; Sacher, O.; Schwab, C. H.; Schwoebel, J.; Terfloth, L.; Arvidson, K.; Richard, A.; Worth, A.; Rathman, J. New Publicly Available Chemical Query Language, CSRML, To Support Chemotype Representations for Application to Data Mining and Modeling. *J. Chem. Inf. Model* **2015**, *55* (3), 25.

(187) Russo, D. P.; Zorn, K. M.; Clark, A. M.; Zhu, H.; Ekins, S. Comparing Multiple Machine Learning Algorithms and Metrics for Estrogen Receptor Binding Prediction. *Mol. Pharm.* **2018**, *15* (10), 4361–4370.

(188) Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. *Proc. 22nd acm sigkdd Int. Conf. Knowl. Discov. data Min.* **2016**, 785–794.

(189) Jagielski, M.; Oprea, A.; Biggio, B.; Liu, C.; Nita-Rotaru, C.; Li, B. Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning. In *IEEE Symposium on Security and Privacy* **2018**, 19–35.

(190) Stulp, F.; Sigaud, O. Many Regression Algorithms, One Unified Model: A Review. *Neural Networks* **2015**, *69*, 60–79.

(191) De Menezes, L. M. B.; Volpato, M. C.; Rosalen, P. L.; Cury, J. A. Bone as a Biomarker of Acute Fluoride Toxicity. *Forensic Sci. Int.* **2003**, *137* (2–3), 209–214.

(192) Nandi, S.; Vracko, M.; Bagchi, M. C. Anticancer Activity of Selected Phenolic Compounds: QSAR Studies Using Ridge Regression and Neural Networks. *Chem. Biol. Drug Des.* **2007**, *70* (5), 424–436.

(193) Das, R. N.; Roy, K. Development of Classification and Regression Models for Vibrio Fischeri Toxicity of Ionic Liquids: Green Solvents for the Future. *Toxicol. Res. (Camb).* **2012**, *1* (3), 186–195.

(194) Borchert, D. M.; Walgenbach, J. F.; Kennedy, G. G.; Long, J. W. Toxicity and Residual Activity of Methoxyfenozide and Tebufenozide to Codling Moth (Lepidoptera: Tortricidae) and Oriental Fruit Moth (Lepidoptera: Tortricidae). *Journal of economic entomology* **2004**, *97* (4), 1342-1352.

(195) Tibshirani, R. Regression Shrinkage and Selection via the Lasso: A Retrospective. *J. R. Stat. Soc. Ser. B (Statistical Methodol.)* **2011**, *73* (3), 273–282.

(196) Micevska, T.; Warne, M. S. J.; Pablo, F.; Patra, R. Variation in, and Causes of, Toxicity of Cigarette Butts to a Cladoceran and Microtox. *Arch. Environ. Contam. Toxicol.* **2006**, *50* (2), 205–212.

(197) Osborne, M. R.; Presnell, B.; Turlach, B. A. On the LASSO and Its Dual. *J. Comput. Graph. Stat.* **2000**, *9* (2), 319–337.

(198) Isbister, G. K.; O'Regan, L.; Sibbritt, D.; Whyte, I. M. Alprazolam Is Relatively More Toxic than Other Benzodiazepines in Overdose. *Br. J. Clin. Pharmacol.* **2004**, *58* (1), 88–95.

(199) Hawkins, D. M.; Basak, S. C.; Mills, D. QSARs for Chemical Mutagens from Structure: Ridge Regression Fitting and Diagnostics. *Environ. Toxicol. Pharmacol.* **2004**, *16* (1–2), 37–44.

(200) Loganayagam, A.; Arenas Hernandez, M.; Corrigan, A.; Fairbanks, L.; Lewis, C. M.; Harper, P.; Maisey, N.; Ross, P.; Sanderson, J. D.; Marinaki, A. M. Pharmacogenetic Variants in the DPYD, TYMS, CDA and MTHFR Genes Are Clinically Significant Predictors of Fluoropyrimidine Toxicity. *Br. J. Cancer* **2013**, *108* (12), 2505–2515.

(201) Cawley, G. C.; Talbot, N. L. C. Reduced Rank Kernel Ridge Regression. *Neural Process. Lett.* **2002**, *16* (3), 293–302.

(202) Roy, K.; Ghosh, G. QSTR with Extended Topochemical Atom (ETA) Indices. 9. Comparative QSAR for the Toxicity of Diverse Functional Organic Compounds to Chlorella Vulgaris Using Chemometric Tools. *Chemosphere* **2007**, *70* (1), 1–12.

(203) Agarwal, V.; Gribok, A. V.; Koschan, A.; Abidi, M. A. Estimating Illumination Chromaticity via Kernel Regression. In *International Conference on Image Processing* **2006**, 981–984.

(204) Kar, S.; Roy, K. QSAR Modeling of Toxicity of Diverse Organic Chemicals to Daphnia Magna Using 2D and 3D Descriptors. *J. Hazard. Mater.* **2010**, *177* (1–3), 344–351.

(205) Duchowicz, P. Linear Regression QSAR Models for Polo-Like Kinase-1 Inhibitors. *Cells* **2018**, *7* (2), 13.

(206) Pan, Y.; Jiang, J.; Wang, R.; Cao, H. Advantages of Support Vector Machine in QSPR Studies for Predicting Auto-Ignition Temperatures of Organic Compounds. *Chemom. Intell. Lab. Syst.* **2008**, *92* (2), 169–178.

(207) Razi, M. A.; Athappilly, K. A Comparative Predictive Analysis of Neural Networks (NNs), Nonlinear Regression and Classification and Regression Tree (CART) Models. *Expert Syst. Appl.* **2005**, *29* (1), 65–74.

(208) Bermejo, S.; Cabestany, J. Learning with Nearest Neighbour Classifiers. *Neural Process. Lett.* **2001**,

13 (2), 159–181.

(209) Neelamegam, S. and Ramaraj, E. Classification Algorithm in Data Mining: An Overview. *Int. J. P2P Netw. Trends Technol.* **2013**, *4* (8), 369–374.

(210) Samsudin, N. A.; Bradley, A. P. Nearest Neighbour Group-Based Classification. *Pattern Recognit.* **2010**, *43* (10), 3458–3467.

(211) Friel, N.; Pettitt, A. N. Classification Using Distance Nearest Neighbours. *Stat. Comput.* **2011**, *21* (3), 431–437.

(212) Lakshmi, S. V.; Prabakaran, T. E. Application of K-Nearest Neighbour Classification Method for Intrusion Detection in Network Data. *International Journal of Computer Applications* **2014**, *97* (7).

(213) Safavian, S.R.; Landgrebe, D. A Survey of Decision Tree Classifier Methodology. *IEEE Trans. Syst. Man. Cybern.* **1991**, *21* (3), 660–674.

(214) Song, Y. Y.; Lu, Y. Decision Tree Methods: Applications for Classification and Prediction. *Shanghai Arch. Psychiatry* **2015**, *27* (2), 130–135.

(215) Dey, A. Machine Learning Algorithms: A Review. *Int. J. Comput. Sci. Inf. Technol.* **2016**, *7* (3), 1174–1179.

(216) Panda, M.; Patra, M. R. Network Intrusion Detection Using Naive Bayes. *International journal of computer science and network security* **2007**, *7* (12), 258-263.

(217) Martfnez-Arroyo, M.; Sucar, L. E. Learning an Optimal Naive Bayes Classifier. In *18th International Conference on Pattern Recognition* **2006**, *3*, 1236–1239.

(218) Taheri, S.; Mammadov, M. Learning the naive Bayes classifier with optimization models. *Int. J. Appl. Math. Comput. Sci* **2013**, *23* (4), 787–795.

(219) Wei, W.; Visweswaran, S.; Cooper, G. F. The Application of Naive Bayes Model Averaging to Predict Alzheimer's Disease from Genome-Wide Data. *J. Am. Med. Informatics Assoc.* **2011**, *18* (4), 370–375.

(220) Frank, E.; Hall, M.; Pfahringer, B. Locally Weighted Naive Bayes. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence* **2002**, 249–256.

(221) Klon, A. E.; Lowrie, J. F.; Diller, D. J. Improved Naïve Bayesian Modeling of Numerical Data for Absorption, Distribution, Metabolism and Excretion (ADME) Property Prediction. *J. Chem. Inf. Model.* **2006**, *46* (5), 1945–1956.

(222) Jiang, L.; Zhang, H.; Cai, Z. A Novel Bayes Model: Hidden Naive Bayes. *IEEE Trans. Knowl. Data Eng.* **2009**, *21* (10), 1361–1371.

(223) Cortes, C.; Vapnik, V.; Saitta, L. Support-Vector Networks. *Machine learning* **1995**, *20* (3), 273-297.

(224) Vapnik, V. Principles of Risk Minimization for Learning Theory. In *Advances in neural information processing systems* **1992**, 831–838.

(225) Zhong, M.; Nie, X.; Yan, A.; Yuan, Q. Carcinogenicity Prediction of Noncongeneric Chemicals by a Support Vector Machine. *Chem. Res. Toxicol.* **2013**, *26* (5), 741–749.

(226) Burges, C. J. C. A Tutorial on Support Vector Machines for Pattern Recognition. *Data mining and knowledge discovery* **1998**, *2* (2), 121-167.

(227) Vapnik, V. N. An Overview of Statistical Learning Theory. *IEEE transactions on neural networks* **1999**, *10* (5), 988-999.

(228) Ben-Hur, A.; Weston, J. A User's Guide to Support Vector Machines. In Data mining techniques for the life sciences **2010**, 223-239.

(229) Noble, W. S. What Is a Support Vector Machine? *Nature Biotechnology* **2006**, *24* (12), 1565–1567.

(230) Rebentrost, P.; Mohseni, M.; Lloyd, S. Quantum Support Vector Machine for Big Data Classification. *Phys. Rev. Lett.* **2014**, *113* (3).

(231) Amari, S.; Wu, S. Improving Support Vector Machine Classifiers by Modifying Kernel Functions. *Neural Networks* **1999**, *12* (6), 783–789.

(232) Varewyck, M.; Martens, J. P. A Practical Approach to Model Selection for Support Vector Machines with a Gaussian Kernel. *IEEE Trans. Syst. Man, Cybern. Part B Cybern.* **2011**, *41* (2), 330–340.

(233) Hussain, M.; Wajid, S. K.; Elzaart, A.; Berbar, M. A Comparison of SVM Kernel Functions for Breast Cancer Detection. In *8th International Conference on Computer Graphics, Imaging and Visualization* **2011**, 145–150.

(234) Kuo, B. C.; Ho, H. H.; Li, C. H.; Hung, C. C.; Taur, J. S. A Kernel-Based Feature Selection Method for SVM with RBF Kernel for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7* (1), 317–326.

(235) Schölkopf, B.; Sung, K. K.; Burges, C. J. C.; Girosi, F.; Niyogi, P.; Poggio, T.; Vapnik, V. Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers. *IEEE Trans. Signal Process.* **1997**, *45* (11), 2758–2765.

(236) Xiao, Y.; Wang, H.; Zhang, L.; Xu, W. Two Methods of Selecting Gaussian Kernel Parameters for One-Class SVM and Their Application to Fault Detection. *Knowledge-Based Syst.* **2014**, *59*, 75–84.

(237) Kim, H. C.; Pang, S.; Je, H. M.; Kim, D.; Bang, S. Y. Constructing Support Vector Machine Ensemble. *Pattern Recognit.* **2003**, *36* (12), 2757–2767.

(238) Mangasarian, O. L.; Wild, E. W. Multisurface Proximal Support Vector Machine Classification via Generalized Eigenvalues. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28* (1), 69–74.

(239) Khemchandani, R.; Chandra, S. Twin Support Vector Machines for Pattern Classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29* (5), 905–910.

(240) Shao, Y. H.; Zhang, C. H.; Wang, X. B.; Deng, N. Y. Improvements on Twin Support Vector Machines. *IEEE Trans. Neural Networks* **2011**, *22* (6), 962–968.

(241) Dioşan, L.; Rogozan, A; Pecuchet, J. P. Improving Classification Performance of Support Vector Machine by Genetically Optimising Kernel Shape and Hyper-Parameters. *Appl Intell* **2012**, *36*, 280–294.

(242) Lanckriet, G. R.; Bartlett, P.; El Ghaoui, L.; Jordan, M. I.; Cristianini, N.; Jordan Lanckriet, M. I.; Ghaoui, E. Learning the Kernel Matrix with Semidefinite Programming. *Journal of Machine learning research* **2004**, *5* (Jan), 27-72.

(243) Sonnenburg, S., Rätsch, G., Schäfer, C. and Schölkopf, B. Large scale multiple kernel learning. *Journal of Machine Learning Research* **2006**, *7* (Jul), 1531-1565.

(244) Bucak, S. S.; Jin, R.; Jain, A. K. Multiple Kernel Learning for Visual Object Recognition: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2013**, *36* (7), 1354–1369.

(245) Sagi, O.; Rokach, L. Ensemble Learning: A Survey. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8* (4), e1249.

(246) Rokach, L. Decision Forest: Twenty Years of Research. *Inf. Fusion* **2016**, *27*, 111–125.

(247) Gomes, H. M.; Barddal, J. P.; Enembreck, A. F.; Bifet, A. A Survey on Ensemble Learning for Data Stream Classification. *ACM Computing Surveys* **2017**, *50* (2), 1-36.

(248) Rokach, L. Ensemble-Based Classifiers. *Artif. Intell. Rev.* **2010**, *33* (1–2), 1–39.

(249) Ren, Y.; Zhang, L.; Suganthan, P. N. Ensemble Classification and Regression-Recent Developments, Applications and Future Directions. *IEEE Computational Intelligence Magazine* **2016**, *11* (1), 41–53.

(250) Woźniak, M.; Graña, M.; Corchado, E. A Survey of Multiple Classifier Systems as Hybrid Systems. *Inf. Fusion* **2014**, *16* (1), 3–17.

(251) Krawczyk, B.; Minku, L. L.; Gama, J.; Stefanowski, J.; Woźniak, M. Ensemble Learning for Data Stream Analysis: A Survey. *Inf. Fusion* **2017**, *37*, 132–156.

(252) Galar, M.; Fernandez, A.; Barrenechea, E.; Bustince, H.; Herrera, F. A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* **2011**, *42* (4), 463–484.

(253) Solomatine, D. P.; Shrestha, D. L. AdaBoost.RT: A Boosting Algorithm for Regression Problems. In *IEEE International Joint Conference on Neural Networks* **2004**, *2*, 1163–1168.

(254) Schapire, R. E. Explaining Adaboost. In *Empirical Inference* **2013**, 37–52.

(255) Breiman, L. Bagging Predictors. *Mach. Learn.* **1996**, *24* (2), 123–140.

(256) Schapire, R.E.; Freund, Y. A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting. *Journal of computer and system sciences* **1997**, *55* (1), 119-139.

(257) Dietterich, T. G. Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. *Mach. Learn.* **2000**, *40* (2), 139–157.

(258) Bernard, S.; Heutte, L.; Adam, S. On the Selection of Decision Trees in Random Forests. In *International Joint Conference on Neural Networks* **2009**, 302–307.

(259) Kingsford, C.; Salzberg, S. L. What Are Decision Trees? *Nature Biotechnology* **2008**, *26* (9), 1011–1012.

(260) Jiang, X.; Wu, C. A.; Guo, H. Forest Pruning Based on Branch Importance. *Comput. Intell. Neurosci.* **2017**, *2017*.

(261) Yang, F.; Lu, W. hang; Luo, L. kai; Li, T. Margin Optimization Based Pruning for Random Forest. *Neurocomputing* **2012**, *94*, 54–63.

(262) Svetnik, V.; Liaw, A.; Tong, C.; Christopher Culberson, J.; Sheridan, R. P.; Feuston, B. P.

Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (6), 1947–1958.

(263) Ali, J.; Khan, R.; Ahmad, N.; Maqsood, I. Random Forests and Decision Trees. *Int. J. Comput. Sci. Issues* **2012**, *9* (5), 272–278.

(264) Prajwala, T. R. A Comparative Study on Decision Tree and Random Forest Using R Tool. *Int. J. Adv. Res. Comput. Commun. Eng.* **2015**, *4* (1).

(265) Fawagreh, K.; Gaber, M. M.; Elyan, E. Random Forests: From Early Developments to Recent Advancements. *Syst. Sci. Control Eng.* **2014**, *2* (1), 602–609.

(266) Kalogirou, S. A. Artificial Neural Networks in Renewable Energy Systems Applications: A Review. *Renewable and Sustainable Energy Reviews* **2001**, *5* (4), 373–401.

(267) Markou, M.; Singh, S. Novelty Detection: A Review - Part 2:: Neural Network Based Approaches. *Signal Processing* **2003**, *83* (12), 2499–2521.

(268) Egmont-Petersen, M.; De Ridder, D.; Handels, H. Image Processing with Neural Networks- A Review. *Pattern Recognition* **2002**, *35* (10), 2279–2301.

(269) Schmidhuber, J. Deep Learning in Neural Networks: An Overview. *Neural Networks* **2015**, *61*, 85–117.

(270) Greff, K.; Srivastava, R. K.; Koutnik, J.; Steunebrink, B. R.; Schmidhuber, J. LSTM: A Search Space Odyssey. *IEEE Trans. Neural Networks Learn. Syst.* **2017**, *28* (10), 2222–2232.

(271) Graves, A.; Mohamed, A. R.; Hinton, G. Speech Recognition with Deep Recurrent Neural Networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing* **2013**, 6645–6649.

(272) Che, Z.; Purushotham, S.; Cho, K.; Sontag, D.; Liu, Y. Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Sci. Rep.* **2018**, *8* (1).

(273) Saha, S.; Raghava, G. P. S. Prediction of Continuous B-Cell Epitopes in an Antigen Using Recurrent Neural Network. *Proteins Struct. Funct. Bioinforma.* **2006**, *65* (1), 40–48.

(274) Zhang, H.; Wang, Z.; Liu, D. A Comprehensive Review of Stability Analysis of Continuous-Time Recurrent Neural Networks. *IEEE Trans. Neural Networks Learn. Syst.* **2014**, *25* (7), 1229–1262.

(275) Stathakis, D. How Many Hidden Layers and Nodes? *Int. J. Remote Sens.* **2009**, *30* (8), 2133–2147.

(276) Teoh, E. J.; Tan, K. C.; Xiang, C. Estimating the Number of Hidden Neurons in a Feedforward Network Using the Singular Value Decomposition. *IEEE Trans. Neural Networks* **2006**, *17* (6), 1623–1629.

(277) Verikas, A.; Bacauskiene, M. Feature Selection with Neural Networks. *Pattern Recognit. Lett.* **2002**, *23* (11), 1323–1335.

(278) Sibi, P.; Jones, S. A.; Siddarth, P. Analysis of Different Activation Functions Using Back Propagation Neural Networks. *J. Theor. Appl. Inf. Technol.* **2013**, *47* (3), 1264–1268.

(279) Nair, V.; Hinton, G. E. Rectified Linear Units Improve Restricted Boltzmann Machines. In *ICML* **2010**, 807–814.

(280) Zeiler, M. D.; Ranzato, M.; Monga, R.; Mao, M.; Yang, K.; Le, Q. V.; Nguyen, P.; Senior, A.; Vanhoucke, V.; Dean, J.; Hinton, G. E. On Rectified Linear Units for Speech Processing. In *IEEE International Conference on Acoustics, Speech and Signal Processing* **2013**, 3517–3521.

(281) Hara, K.; Saito, D.; Shouno, H. Analysis of Function of Rectified Linear Unit Used in Deep Learning. In *International Joint Conference on Neural Networks* **2015**, 1-8.

(282) Wray, J.; Green, G. G. R. Neural Networks, Approximation Theory, and Finite Precision Computation. *Neural networks* **1995**, *8* (1), 31-37.

(283) Siddique, M. N. H.; Tokhi, M. O. Training Neural Networks: Backpropagation vs Genetic Algorithms. In *International Joint Conference on Neural Networks* **2001**, *4*, 2673–2678.

(284) Ismail Fawaz, H.; Forestier, G.; Weber, J.; Idoumghar, L.; Muller, P. A. Deep Learning for Time Series Classification: A Review. *Data Min. Knowl. Discov.* **2019**, *33* (4), 917–963.

(285) Wilamowski, B. M.; Yu, H. Neural Network Learning without Backpropagation. *IEEE Trans. Neural Networks* **2010**, *21* (11), 1793–1803.

(286) Zhu, X. X.; Tuia, D.; Mou, L.; Xia, G. S.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geoscience and Remote Sensing Magazine* **2017**, *5* (4), 8–36.

(287) Miotto, R.; Wang, F.; Wang, S.; Jiang, X.; Dudley, J. T. Deep Learning for Healthcare: Review, Opportunities and Challenges. *Brief. Bioinform.* **2018**, *19* (6), 1236–1246.

(288) Guo, Y.; Liu, Y.; Oerlemans, A.; Lao, S.; Wu, S.;

Lew, M. S. Deep Learning for Visual Understanding: A Review. *Neurocomputing* **2016**, *187*, 27–48.

(289) Lecun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, 521 (7553), 436–444.

(290) Kriegeskorte, N. Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing. *Annu. Rev. Vis. Sci.* **2015**, *1* (1), 417–446.

(291) Segler, M. H. S.; Preuss, M.; Waller, M. P. Planning Chemical Syntheses with Deep Neural Networks and Symbolic AI. *Nature* **2018**, *555* (7698), 604–610.

(292) Esteva, A.; Kuprel, B.; Novoa, R. A.; Ko, J.; Swetter, S. M.; Blau, H. M.; Thrun, S. Dermatologist-Level Classification of Skin Cancer with Deep Neural Networks. *Nature* **2017**, *542* (7639), 115–118.

(293) Sladojevic, S.; Arsenovic, M.; Anderla, A.; Culibrk, D.; Stefanovic, D. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Comput. Intell. Neurosci.* **2016**, *2016*.

(294) Cireşan, D. C.; Meier, U.; Gambardella, L. M.; Schmidhuber, J. Deep, Big, Simple Neural Nets for Handwritten Digit Recognition. *Neural Computation* **2010**, *22* (12), 3207–3220.

(295) Pallipuram, V. K.; Bhuiyan, M.; Smith, M. C. A Comparative Study of GPU Programming Models and Architectures Using Neural Networks. *J. Supercomput.* **2012**, *61* (3), 673–718.

(296) Huqqani, A. A.; Schikuta, E.; Ye, S.; Chen, P. Multicore and GPU Parallelization of Neural Networks for Face Recognition. *Procedia Comput. Sci.* **2013**, *18*, 349–358.

(297) Oh, K. S.; Jung, K. GPU Implementation of Neural Networks. *Pattern Recognit.* **2004**, *37* (6), 1311–1314.

(298) Lin, S. B. Limitations of Shallow Nets Approximation. *Neural Networks* **2017**, *94*, 96–102.

(299) Lawrence, S.; Lee Giles, C.; Member, S.; Chung Tsoi, A.; Back, A. D. Face Recognition: A Convolutional Neural-Network Approach. *IEEE transactions on neural networks* **1997**, *8* (1), 98-113.

(300) Rawat, W.; Wang, Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation* **2017**, *29* (9), 2352–2449.

(301) Wiatowski, T.; Bolcskei, H. A Mathematical Theory of Deep Convolutional Neural Networks for Feature Extraction. *IEEE Transactions on Information Theory* **2017**, *64* (3), 1845–1866.

(302) Lee, C. Y.; Gallagher, P. W.; Tu, Z. Generalizing Pooling Functions in Convolutional Neural Networks: Mixed, Gated, and Tree. In *Artificial intelligence and statistics* **2016**, 464–472.

(303) Chami, I.; Ying, R.; Ré, C.; Leskovec, J. Hyperbolic Graph Convolutional Neural Networks. In *Advances in Neural Information Processing Systems* **2019**, 4868–4879.

(304) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: A Benchmark for Molecular Machine Learning. *Chem. Sci.* **2018**, *9* (2), 513–530.

(305) Niepert, M.; Ahmad, M.; Kutzkov, K. Learning Convolutional Neural Networks for Graphs. In *International Conference on Machine Learning* **2016**, 2014-2023.

(306) Coley, C. W.; Jin, W.; Rogers, L.; Jamison, T. F.; Jaakkola, T. S.; Green, W. H.; Barzilay, R.; Jensen, K. F. A Graph-Convolutional Neural Network Model for the Prediction of Chemical Reactivity. *Chem. Sci.* **2019**, *10* (2), 370–377.

(307) Ying, R.; He, R.; Chen, K.; Eksombatchai, P.; Hamilton, W.L.; Leskovec, J. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* **2018**, 974–983.

(308) Jimenez-Carretero, D.; Abrishami, V.; Fernández-de-Manuel, L.; Palacios, I.; Quílez-Álvarez, A.; Díez-Sánchez, A.; del Pozo, M. A.; Montoya, M. C. Tox_(R)CNN: Deep Learning-Based Nuclei Profiling Tool for Drug Toxicity Screening. *PLoS Comput. Biol.* **2018**, *14* (11), e1006238.

(309) Press, E. B.; Maron, D. M.; Ames, B. N. Revised Methods for the Salmonella Mutagenicity Test. *Mutation Research/Environmental Mutagenesis and Related Subjects* **1983**, *113* (3-4), 173-215.

(310) Ames, B. N.; Durston, W. E.; Yamasaki, E.; Lee, F. D. Carcinogens Are Mutagens: A Simple Test Combining Liver Homogenates for Activation and Bacteria for Detection. *Proc. Natl. Acad. Sci. U. S. A.* **1973**, *70* (8), 2281–2285.

(311) Hillebrecht, A.; Muster, W.; Brigo, A.; Kansy, M.; Weiser, T.; Singer, T. Comparative Evaluation of in Silico Systems for Ames Test Mutagenicity Prediction: Scope and Limitations. *Chem. Res.*

Toxicol. **2011**, *24* (6), 843–854.

(312) Novotarskyi, S.; Abdelaziz, A.; Sushko, Y.; Kö, R.; Vogt, J.; Tetko, I. V. ToxCast EPA in Vitro to in Vivo Challenge: Insight into the Rank-I Model. *Chem. Res. Toxicol* **2016**, *29* (5), 49.

(313) Xu, T.; Ngan, D. K.; Ye, L.; Xia, M.; Xie, H. Q.; Zhao, B.; Simeonov, A.; Huang, R. Predictive Models for Human Organ Toxicity Based on In Vitro Bioactivity Data and Chemical Structure. *Chem. Res. Toxicol* **2020**, *2020* (3), 731–741.

(314) Rostami-Hodjegan, A. Physiologically Based Pharmacokinetics Joined with in Vitro-in Vivo Extrapolation of ADME: A Marriage under the Arch of Systems Pharmacology. *Clin. Pharmacol. Ther.* **2012**, *92* (1), 50–61.

(315) Caruana, R. Multitask Learning. *Mach. Learn.* **1997**, *28* (1), 41–75.

(316) Gibaja, E.; Ventura, S. Multi-Label Learning: A Review of the State of the Art and Ongoing Research. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2014**, *4* (6), 411–444.

(317) Sosnin, S.; Vashurina, M.; Withnall, M.; Karpov, P.; Fedorov, M.; Tetko, I. V. A Survey of Multi-Task Learning Methods in Chemoinformatics. *Molecular Informatics* **2019**, *38 (4)*, 1800108.

(318) Zhang, T. Analysis of Multi-Stage Convex Relaxation for Sparse Regularization. *J. Mach. Learn. Res.* **2010**, *11* (Mar), 1081–1107.

(319) Weigend, A. S.; Rumelhart, D. E.; Huberman, B. A. Generalization by Weight-Elimination with Application to Forecasting. In *Advances in neural information processing systems* **1991**, 875–882.

(320) Hastie, T.; Rosset, S.; Tibshirani, R.; Zhu, J. The Entire Regularization Path for the Support Vector Machine. *J. Mach. Learn. Res.* **2004**, *5* (Oct), 1391–1415.

(321) Yao, Y.; Rosasco, L.; Caponnetto, A. On Early Stopping in Gradient Descent Learning. *Constr. Approx.* **2007**, *26* (2), 289–315.

(322) Hagiwara, K. Regularization Learning, Early Stopping and Biased Estimator. *Neurocomputing* **2002**, *48* (1-4), 937–955.

(323) Ng, A. Y. Feature Selection, L1 vs. L2 Regularization, and Rotational Invariance. In *Proceedings of the twenty-first international conference on Machine learning* **2004**, 78.

(324) Girosi, F.; Jones, M.; Poggio, T. Regularization Theory and Neural Networks Architectures. *Neural Comput.* **1995**, *7* (2), 219–269.

(325) Williams, P. M. Bayesian Regularization and Pruning Using a Laplace Prior. *Neural Comput.* **1995**, *7* (1), 117–143.

(326) Shi, G.; Zhang, J.; Li, H.; Wang, C. Enhance the Performance of Deep Neural Networks via L2 Regularization on the Input of Activations. *Neural Process. Lett.* **2019**, *50* (1), 57–75.

(327) Lee, S.-I.; Ganapathi, V.; Koller, D. Efficient Structure Learning of Markov Networks Using L 1-Regularization. In *Advances in neural Information processing systems* **2007**, 817–824.

(328) Cronin, M.T.; Richarz, A.N.; Schultz, T. W. Identification and Description of the Uncertainty, Variability, Bias and Influence in Quantitative Structure-Activity Relationships (QSARs) for Toxicity Prediction. *Regul. Toxicol. Pharmacol.* **2019**, *106*, 90–104.

(329) Pittman, M. E.; Edwards, S. W.; Ives, C.; Mortensen, H. M. AOP-DB: A Database Resource for the Exploration of Adverse Outcome Pathways through Integrated Association Networks. *Toxicol. Appl. Pharmacol.* **2018**, *343*, 71–83.

(330) Leist, M.; Ghallab, A.; Graepel, R.; Marchan, R.; Hassan, R.; Bennekou, S. H.; Limonciel, A.; Vinken, M.; Schildknecht, S.; Waldmann, T.; Danen, E.; van Ravenzwaay, B.; Kamp, H.; Gardner, I.; Godoy, P.; Bois, F. Y.; Braeuning, A.; Reif, R.; Oesch, F.; Drasdo, D.; Höhme, S.; Schwarz, M.; Hartung, T.; Braunbeck, T.; Beltman, J.; Vrieling, H.; Sanz, F.; Forsby, A.; Gadaleta, D.; Fisher, C.; Kelm, J.; Fluri, D.; Ecker, G.; Zdrazil, B.; Terron, A.; Jennings, P.; van der Burg, B.; Dooley, S.; Meijer, A. H.; Willighagen, E.; Martens, M.; Evelo, C.; Mombelli, E.; Taboureau, O.; Mantovani, A.; Hardy, B.; Koch, B.; Escher, S.; van Thriel, C.; Cadenas, C.; Kroese, D.; van de Water, B.; Hengstler, J. G. Adverse Outcome Pathways: Opportunities, Limitations and Open Questions. *Arch. Toxicol.* **2017**, *91* (11), 3477–3505.

(331) Vinken, M. The Adverse Outcome Pathway Concept: A Pragmatic Tool in Toxicology. *Toxicology* **2013**, *312*, 158–165.

(332) Allen, T. E. H.; Goodman, J. M.; Gutsell, S.; Russell, P. J. Defining Molecular Initiating Events in the Adverse Outcome Pathway Framework for Risk Assessment. *Chem. Res. Toxicol.* **2014**, *27* (12), 2100–2112.

(333) Ellison, C. M.; Enoch, S. J.; Cronin, M. T. D. A Review of the Use of in Silico Methods to Predict the Chemistry of Molecular Initiating Events

Related to Drug Toxicity. *Expert Opin. Drug Metab. Toxicol.* **2011**, *7* (12), 1481–1495.

(334) Allen, T. E. H.; Goodman, J. M.; Gutsell, S.; Russell, P. J. A History of the Molecular Initiating Event. *Chemical Research in Toxicology* **2016**, *29* (12), 2060–2070.

(335) Slikker Jr, W.; de Souza Lima, T.A.; Archella, D.; de Silva Junior, J.B.; Barton-Maclaren, T.; Bo, L.; Buvinich, D.; Chaudhry, Q.; Chuan, P.; Deluyker, H.; Domselaar, G. Emerging Technologies for Food and Drug Safety. *Regul. Toxicol. Pharmacol.* **2018**, *98*, 115–128.

(336) Zaunbrecher, V.; Beryt, E.; Parodi, D.; Telesca, D.; Doherty, J.; Malloy, T.; Allard, P. Has Toxicity Testing Moved into the 21st Century? A Survey and Analysis of Perceptions in the Field of Toxicology. *Environ. Health Perspect.* **2017**, *125* (8), 087024.

(337) Chesnut, M.; Yamada, T.; Adams, T.; Knight, D., Kleinstreuer, N.; Kass, G.; Luechtefeld, T.; Hartung, T.; Maertens, A. Regulatory Acceptance of Read-Across. *ALTEX-Alternatives to Anim. Exp.* **2018**, *35* (3), 413–419.