

Article

Hepatotoxicity Modeling Using Counter-Propagation Artificial Neural Networks: Handling an Imbalanced Classification Problem

Benjamin Bajželj^{1,2} and Viktor Drgan^{1,*}¹ National Institute of Chemistry, Hajdrihova 19, 1001 Ljubljana, Slovenia; benjamin.bajzelj@ki.si² Biotechnical Faculty, University of Ljubljana, Jamnikarjeva 101, 1000 Ljubljana, Slovenia

* Correspondence: viktor.drgan@ki.si; Tel.: +386-1476-0388

Academic Editor: Marjan Vračko

Received: 31 December 2019; Accepted: 21 January 2020; Published: 23 January 2020



Abstract: Drug-induced liver injury is a major concern in the drug development process. Expensive and time-consuming *in vitro* and *in vivo* studies do not reflect the complexity of the phenomenon. Complementary to wet lab methods are *in silico* approaches, which present a cost-efficient method for toxicity prediction. The aim of our study was to explore the capabilities of counter-propagation artificial neural networks (CPANNs) for the classification of an imbalanced dataset related to idiosyncratic drug-induced liver injury and to develop a model for prediction of the hepatotoxic potential of drugs. Genetic algorithm optimization of CPANN models was used to build models for the classification of drugs into hepatotoxic and non-hepatotoxic class using molecular descriptors. For the classification of an imbalanced dataset, we modified the classical CPANN training algorithm by integrating random subsampling into the training procedure of CPANN to improve the classification ability of CPANN. According to the number of models accepted by internal validation and according to the prediction statistics on the external set, we concluded that using an imbalanced set with balanced subsampling in each learning epoch is a better approach compared to using a fixed balanced set in the case of the counter-propagation artificial neural network learning methodology.

Keywords: hepatotoxicity; counter-propagation artificial neural networks; imbalanced dataset; genetic algorithm; QSAR

1. Introduction

The liver is the primary site of xenobiotic metabolism, and as such, is prone to suffering from toxic effects. Moreover, it is well known that drug-induced liver injury (DILI) poses one of the main challenges in the drug development process. Aside from efficacy, toxicity is the main reason for the termination of drug development [1], while hepatotoxicity is one of the biggest threats in the drug development process. Hepatotoxicity presents the main reason for the withdrawal of drugs from the market [2,3]. Therefore, it is important to address the problem in the early stages of drug development. Preclinical stages consist of obligatory animal testing. However, a study on 3290 approved drugs and formulations [4] suggests that the absence of hepatotoxic effects in animals does not generally predict safety in humans. The study also implies medium predictive power of a positive result on animals in the case of liver disorders. Apart from moderate predictivity, other reasons aimed at reducing animal testing are ethical concerns and implementation of the reduction, refinement, and replacement (“3R”) strategy in research [5]. Human primary hepatocytes (hPH) are considered the gold standard for studying *in vitro* hepatotoxicity, but their major limitation is that they rapidly de-differentiate and liver-specific functions, such as albumin production and cytochrome P450 expression, decline quickly over the first 24–48 h of culture [6]. The use of immortalized primary hepatocytes, which suffer from

hindered biotransformation capabilities, is frequent. Alternatives to *in vitro* toxicity include induced pluripotent stem cells [7] (especially in high throughput screening), while more complex models are being developed, such as organoids or liver-on-a-chip [8,9]. *In vitro* methods evaluating endpoints related to DILI can achieve a sensitivity of up to 70% [10]. However, while they substitute animal testing, *in vitro* methods are not truly equal to the *in vivo* system and may also be time consuming and expensive. In this regard, *in silico* methods offer a time- and cost-efficient method to address the issue of toxicity early in the drug development process [11]. In recent years, efforts have been made to predict hepatotoxicity using various techniques of quantitative structure–activity relationship (QSAR) modeling. QSAR studies of DILI are based mainly on small datasets. One such study was conducted by Cruz-Monteagudo [12]. They achieved good results using 3D molecular descriptors and a training dataset with 33 compounds in the hepatotoxic class and 41 compounds in the non-hepatotoxic class. In their study, based on a structurally and pharmacologically diverse training dataset, they developed classification models using linear discriminant analysis (LDA), artificial neural networks using radial basis function (RBF) architecture, and the OneR classification algorithm. In the external validation experiment, they achieved sensitivity of 100% and specificity of 67% using the LDA model, sensitivity of 67% and specificity of 67% using the RBF model, and sensitivity of 67% and specificity of 100% using the OneR classifier. A study with a larger number of compounds was conducted by Ekins et al. [13]. In the study, they used extended connectivity fingerprint counts along a few additional interpretable 2D descriptors for the Bayesian model. Sensitivity and specificity on the external set of 237 compounds were 56% and 67%, respectively. On a smaller set of 28 structurally similar compounds with different DILI potential, the performance was similar. Fourches et al. [14] developed QSAR models with support-vector machines (SVMs) for hepatotoxicity, with a dataset of 531 compounds (248 hepatotoxic, 283 non-hepatotoxic) acquired with text mining, and achieved accuracy from 55.7% to 72.6% with external cross-validation. As discussed in Kotsampasakou et al. [15], data curation is of high importance and text mining is error-prone and presents the reason for a lower quality of dataset. More accurate predictions can be made using ensemble modeling, as was shown in the study of Liew et al. [10]. The ensemble model, constructed by the stacking of random forest, k-nearest neighbor, and naive Bayes models with naive Bayes, was based on 2D descriptors. While the average sensitivity and specificity of 617 base classifiers on the external set of 120 compounds were 62.4% and 61.8%, respectively, the ensemble model had 84.5% sensitivity and 65.1% specificity for 101 objects of the external set that were inside the applicability domain. It is worth noting, however, that the ensemble model could not separate non-toxic compounds from structurally very similar toxic compounds, the problem also encountered in the study of Ekins et al. [13]. The study also suggested that weakly hepatotoxic compounds have a big influence on decision boundary and their removal greatly affects prediction metrics. In the recent study by Wang et al. [16], which used 450 molecules, an ensemble model was built mainly on different molecular fingerprints using multiple machine learning algorithms, and achieved an accuracy of 81.67%, sensitivity of 64.55%, specificity of 96.15%, and the area under the receiver operating characteristic curve (AUC) of 80.35% on the external validation set. A recursive random forest approach on a set containing 122 DILI-positive and 932 DILI-negative compounds with adjusting decision threshold was used in the study by Zhu et al. [17] to achieve good sensitivity and specificity (both around 80%) for models each using different types of structure presentation (CDK, MACCS, and Mold2 descriptors). In that way, the authors significantly reduced the number of meaningful descriptors, as they ended up with 8, 26, and 20 descriptors. In this study, ensemble modeling also significantly improved performance.

In our study, we employ counter-propagation artificial neural networks (CPANNs) in an attempt to capture the nonlinear relationship between molecular structure and hepatotoxicity of drugs. CPANNs combine supervised and unsupervised learning strategies and have been successfully applied to different QSAR problems [18–20]. Due to the learning strategies used, they are especially suitable for classification problems. To our knowledge, the method has never been used for modeling hepatotoxicity before. The majority of the drugs in the data set used do not exhibit hepatotoxicity. To deal with

the imbalanced dataset, we apply two approaches. One approach presents manual balancing of the training dataset prior to model building with CPANNs. In the second approach, we modify the classical training algorithm of CPANNs by integrating random subsampling into the training procedure to obtain balanced representation of two toxicity classes during the training of a CPANN. Molecular descriptors calculated from 2D representation of compounds are used in optimizations of CPANN models by a genetic algorithm. With 148 models passing internal validation criteria, we construct two consensus models, one using 24 models and the other one using 124 models. The first consensus model, based on a manually-constructed balanced training dataset, yields sensitivity and specificity of 0.5 and 0.79, respectively. The second consensus model, obtained using models developed by the modified training algorithm, has sensitivity and specificity of 0.65 and 0.85, respectively. A simple metric is constructed to identify descriptors important for the prediction of hepatotoxicity.

2. Results and Discussion

2.1. Dataset

After data curation, our dataset contained 524 compounds, which were gathered using five literature sources. As can be seen in Figure 1, there were no compounds that were selected from just one source; an exception is the LiverTox database.

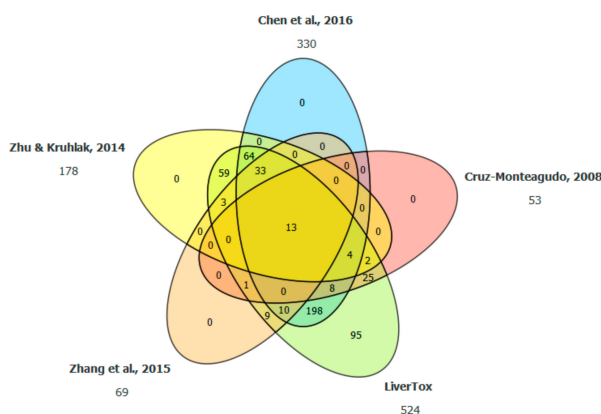


Figure 1. Venn diagram of selected compounds. Compounds in our dataset that we classified either as DILI-positive or DILI-negative were extracted from different literature sources. Only LiverTox contained compounds that were not present in any of the other sources.

Different approaches that were used to deal with dataset imbalance influenced the selection of compounds into training set, test set (TE1), internal validation set (TE2), and external validation set. The underlying reason was the difference in the size of training sets for the two approaches. In the first approach, a balanced number of hepatotoxic and non-hepatotoxic compounds was used in the training set, which consisted of 216 compounds. For the second approach, where the modified CPANN training algorithm was applied, an imbalanced training set of 404 compounds, with 26.7% compounds in the hepatotoxic class, was used. Thus, the test set, internal validation set, and external validation set used in the first approach contained a larger number of compounds from the non-hepatotoxic class. The class distribution of compounds among the sets is given in Table 1. The selection of compounds into the validation set was made based on the distribution of compounds on the Kohonen top-map shown in Figures 2 and 3 for the first and the second approach, respectively. The red color in Figures 2 and 3 indicates compounds selected for the validation set. Principal component analysis (PCA) was performed to verify if it could separate compounds belonging to the hepatotoxic and non-hepatotoxic classes. It was observed that the first few principal components could not separate the two classes of compounds. More details are given in the Supplementary Materials (Supplementary file: Supplementary1.pdf).

Table 1. Distribution of compounds into classes for the datasets used.

		TR		TE1		TE2		VA	
Ninit		Hepat.	Non-H.	Hepat.	Non-H.	Hepat.	Non-H.	Hepat.	Non-H.
182	a	108	296	20	20	20	20	20	20
	b	108	108	20	83	20	83	20	82
98	a	108	296	20	20	20	20	20	20
	b	108	108	20	83	20	83	20	82
50	a	108	296	20	20	20	20	20	20
	b	108	108	20	83	20	83	20	82

TR—training set; TE1—the first test set; TE2—the second test set; VA—external validation set. Ninit—Number of initial descriptors in the dataset. Hepat—Number of compounds in hepatotoxic class. Non-H—Number of compounds in non-hepatotoxic class. a—Imbalanced training set used during optimization. b—Manually balanced training set used during optimization.

210	187 723 943	28 183 212 563			306 326 847 860 940	205 564 764 947	355			109 396 899 457	249 432 655 657	312 618 619 705	438 926 919 968	
115 451 760 936	106 821	203 461 555	181 674 695 876	33 87 209 714	33 87 209 714	253 293 640	118 382 933 945		602 951 130 229	130 229 839	839	145 186 344 626 734 975	680	
717	741 745 903	44 381 698 739	15 53 193 688 980	116 235 920	227 736 920	38 383 834 962	78 170 976		666 792 949	692 812 965	72 744 965	341 473 545 610 931	467 671 911	279 861
299 324 468 595	110 117	327 763	433 609 630 693	69 124 314 389 472	69 124 314 389 472	662 879 598 981	38 140 55 790 982			623	534	93 300	295	
356 443	978	297 371 411 494 636 882	301 628	134 191 372 643 735	26 497 505 673	386 514 548 631 696	22 66 318 815		607 685	42 291 456 633 697 854	431 501	710	135 157 384 568 889	754
45 937 960	560 579 601 759	73 901 909		195 197 889 969		922	336		67 408 526 529 580 740		509	604 684 515 538	12 111 516	
	608 773	319	373 941 957	256 656 701	226 426 522 616 948	65 204 64 645 649	90 183 166 167		768	788 798 959		274 603 851 974	189 289 241 423 437	
	593	142 899	85 703	397 963		7 315 842	161 162 577 675		880 921	18 201 286 424 650 896 984	491 492	13 155 651 858	146 444 581 715 843 845	
578	30 104 485 569 635 772 826	902	75 622	362 648	213 357 869 905	323 513 776 814 918	131 419 476 559 867		405 647 938		331 393 510 854		11 174 844	
452 637	711	454 561 774	285 704 757	24 377 622 742 872 985	277 547 556		518 640 646		276 455 583	862	595	152 584 526	47 316 654 767	1 304 929
37 554 805	486 553	874	584	511 775	101	369 557 793 827	251		434 863 864 865 866 886 924	179 758 870	52 114 136 831		338 490 669 825 950	8 729
322 566 574 596 725 771	172 765	524 771 787 898 915	217 900	151 575 634 679 731	194 678 726 886	379 398 684	64			732	46 722	20 125 852	442 465 466	562 751
878	105 720 833	653 904	417	207 354 600 897	99	664			225 286 422 456 458 615 681 683	71 273 531			532 755 483	138 281
565 587 605	153 233 276 292 311 672 730	658 769	54 506 597 721 835	296 677 728 988	585 590 691	224 302 462 668 830	449		586 621 624 625		263 572 588 676 682 895	709 894	567 818 756	737 756

Figure 2. Selection of validation set compounds for the first approach using the Kohonen top-map.

210	187 723 943	28 183 212 583			306 326 647 860 940	205 564 764 947	355		109 396 399 457	249 432 655 657	312 618 919	619 705 968	430 926
115 451 768 936	106 821 555	203 461 695 676	181 674 209 714	33 87 227 736	39 390 540	253 393 933 945	118 392 692 951		130 229 839		145 186 344 626 734 970		680
717	741 746 903	44 381 696 739	15 53 193 688 980	116 235 923	227 736 834 962	96 383 976	78 170 949	666 792 965	692 812 72 744		341 473 545 610 931	467 671 911	279 861
299 324 468 535	110 117 927 783	433 609 630 693	69 124 314 389 472		662 879 596 981	38 140 551 700 982	63 287 345		623 534		93 300		395
358 443	978 494 635 882	297 371 411 901 628	134 191 372 643 725	26 497 903 548 631 690 873	386 514 700 806	22 66 310 815		607 685 633 697 856	42 291 450 431 501		710 668 899	135 157 367 568 899	754
45 937 960	560 579 601 759	73 901 909		195 197 869 969			922 336	67 408 520 529 590 740		509	604 694 515 538		12 111 516
	608 773	319	373 941 967	356 656 701	226 426 527 516 948	65 204 641 645 649	90 163 166 187	768 959	788 798 851 974		274 603 241 423		189 289 437
	593	142 699	65 703 987	397 983		7 315 842	161 162 677 675	880 921 984	10 201 200 424 650 896	491 492 651 858	13 155 581 715		146 464 843 845
578	30 104 489 569 635 772 826	902	75 622	362 649 869 905	213 357 814 918	323 513 776 559 867	131 419 476 938	405 647 938		331 393 510 854			11 174 844
453 637	711	454 561 774	285 704 757	34 377 627 742 872 985	277 547 856	100 958 646	518 640 983	276 455 862		152 584 595	47 316 526	1 304 654 767	929
37 554 805	486 553	874	594	511 775 101	369 557 793 827		251	434 863 884 865 866 868 504	179 788 870	52 114 136 831		338 490 669 825	8 729 950
322 566 574 596 725 770	172 765 898 915	524 771 787 217 900	151 575 634 679 731	194 678 726 886	379 398 684	64			792	46 722 852	20 125 442 465		552 751 466
878	105 720 833	653 904	417	207 354 600 897	99	664		225 290 422 456 458 615 681 683	71 273 531			532 755 483	138 281
565 587 605	153 230 270 292 311 672 730	658 769	721 835	296 677 728 988	585 590 691	224 302 462 588 830	449	586 621 624 625		263 572 589 676 682 893	709 884	567 818	737 756 766

Figure 3. Selection of validation set compounds for the second approach using the Kohonen top-map.

2.2. Models

Optimizations of CPANNs were made using the manually balanced training set in the first approach and using the modified CPANN training algorithm in the second approach. The optimizations for the first approach resulted in 24 models that passed our criteria, i.e., with sensitivity and specificity on training, test set TE1, and test set TE2 of at least 0.7, where test set TE2 was used as the internal validation set. The second approach resulted in 124 models passing the same criteria. The results for individual models are given in Tables 2–4, where each table presents the results for one of the three optimization criteria used. The averages of sensitivity and specificity calculated based on the results of individual models for the first approach were 0.55 and 0.71, respectively, while for the second approach, the averages were 0.63 and 0.67.

Table 2. Average values of sensitivity and specificity obtained for models optimized using optimization criterion 1. Average values and standard deviations are given for 100 models built using selected descriptors and different permutations of objects in the training set.

OC1		TR		TE1		TE2		VA	
Nin.	model	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
182	1 ^a	0.81 ± 0.02	0.87 ± 0.02	0.82 ± 0.07	0.84 ± 0.03	0.74 ± 0.07	0.70 ± 0.04	0.56 ± 0.06	0.68 ± 0.04
	2 ^b	0.92 ± 0.02	0.79 ± 0.02	0.90 ± 0.06	0.92 ± 0.05	0.76 ± 0.08	0.78 ± 0.09	0.64 ± 0.08	0.72 ± 0.08
	3 ^b	0.91 ± 0.02	0.79 ± 0.02	0.86 ± 0.06	0.90 ± 0.06	0.72 ± 0.09	0.78 ± 0.09	0.63 ± 0.08	0.73 ± 0.09
	4 ^b	0.92 ± 0.02	0.78 ± 0.02	0.88 ± 0.07	0.90 ± 0.07	0.71 ± 0.07	0.77 ± 0.08	0.65 ± 0.07	0.75 ± 0.08
	5 ^b	0.86 ± 0.03	0.76 ± 0.03	0.85 ± 0.07	0.86 ± 0.07	0.72 ± 0.08	0.71 ± 0.08	0.57 ± 0.08	0.78 ± 0.08
	6 ^b	0.92 ± 0.02	0.81 ± 0.02	0.91 ± 0.06	0.92 ± 0.06	0.72 ± 0.07	0.74 ± 0.09	0.70 ± 0.07	0.66 ± 0.07
	7 ^b	0.88 ± 0.03	0.77 ± 0.03	0.77 ± 0.08	0.83 ± 0.08	0.71 ± 0.07	0.71 ± 0.07	0.73 ± 0.07	0.77 ± 0.09
	8 ^b	0.88 ± 0.02	0.77 ± 0.03	0.90 ± 0.04	0.88 ± 0.06	0.77 ± 0.08	0.70 ± 0.08	0.64 ± 0.07	0.68 ± 0.09
98	1 ^b	0.89 ± 0.02	0.76 ± 0.02	0.91 ± 0.06	0.90 ± 0.05	0.72 ± 0.08	0.70 ± 0.09	0.74 ± 0.08	0.48 ± 0.10
	2 ^b	0.89 ± 0.02	0.77 ± 0.02	0.87 ± 0.05	0.92 ± 0.04	0.77 ± 0.05	0.70 ± 0.06	0.60 ± 0.06	0.57 ± 0.08
	3 ^b	0.89 ± 0.02	0.77 ± 0.02	0.86 ± 0.05	0.91 ± 0.04	0.75 ± 0.06	0.71 ± 0.07	0.60 ± 0.06	0.58 ± 0.09
	4 ^b	0.90 ± 0.02	0.77 ± 0.02	0.86 ± 0.05	0.91 ± 0.06	0.76 ± 0.07	0.71 ± 0.06	0.68 ± 0.07	0.63 ± 0.08
	5 ^b	0.94 ± 0.02	0.82 ± 0.02	0.84 ± 0.07	0.92 ± 0.05	0.70 ± 0.06	0.71 ± 0.07	0.69 ± 0.06	0.65 ± 0.07
	6 ^b	0.85 ± 0.02	0.71 ± 0.02	0.88 ± 0.07	0.83 ± 0.07	0.73 ± 0.07	0.73 ± 0.07	0.69 ± 0.06	0.49 ± 0.09
	7 ^b	0.95 ± 0.02	0.80 ± 0.02	0.88 ± 0.05	0.89 ± 0.07	0.75 ± 0.07	0.77 ± 0.08	0.54 ± 0.08	0.68 ± 0.07
	8 ^b	0.96 ± 0.02	0.79 ± 0.02	0.87 ± 0.06	0.88 ± 0.06	0.75 ± 0.08	0.74 ± 0.08	0.53 ± 0.07	0.67 ± 0.07
	9 ^b	0.95 ± 0.02	0.81 ± 0.02	0.87 ± 0.05	0.90 ± 0.06	0.76 ± 0.07	0.79 ± 0.07	0.53 ± 0.08	0.70 ± 0.07
	10 ^b	0.78 ± 0.03	0.73 ± 0.03	0.81 ± 0.08	0.80 ± 0.06	0.71 ± 0.07	0.75 ± 0.06	0.63 ± 0.06	0.62 ± 0.07
	11 ^b	0.87 ± 0.03	0.71 ± 0.03	0.80 ± 0.08	0.84 ± 0.07	0.72 ± 0.09	0.76 ± 0.08	0.64 ± 0.08	0.58 ± 0.09
	12 ^b	0.93 ± 0.02	0.80 ± 0.02	0.93 ± 0.07	0.93 ± 0.05	0.71 ± 0.06	0.77 ± 0.08	0.66 ± 0.07	0.66 ± 0.08
	13 ^b	0.82 ± 0.03	0.75 ± 0.02	0.84 ± 0.06	0.86 ± 0.08	0.72 ± 0.08	0.74 ± 0.07	0.72 ± 0.08	0.64 ± 0.06
	14 ^b	0.85 ± 0.02	0.76 ± 0.02	0.86 ± 0.07	0.90 ± 0.07	0.70 ± 0.08	0.72 ± 0.08	0.70 ± 0.07	0.64 ± 0.07
	15 ^b	0.79 ± 0.03	0.73 ± 0.02	0.90 ± 0.05	0.92 ± 0.06	0.77 ± 0.05	0.72 ± 0.07	0.64 ± 0.05	0.77 ± 0.07
	16 ^b	0.79 ± 0.03	0.72 ± 0.02	0.89 ± 0.07	0.93 ± 0.05	0.80 ± 0.06	0.71 ± 0.06	0.64 ± 0.06	0.78 ± 0.07
50	1 ^a	0.77 ± 0.02	0.71 ± 0.02	0.88 ± 0.04	0.83 ± 0.04	0.77 ± 0.04	0.73 ± 0.04	0.64 ± 0.03	0.68 ± 0.04
	2 ^a	0.77 ± 0.02	0.72 ± 0.02	0.86 ± 0.04	0.84 ± 0.02	0.76 ± 0.04	0.75 ± 0.03	0.63 ± 0.04	0.71 ± 0.04
	3 ^a	0.77 ± 0.02	0.71 ± 0.02	0.89 ± 0.04	0.82 ± 0.04	0.77 ± 0.03	0.73 ± 0.05	0.64 ± 0.03	0.68 ± 0.04
	4 ^a	0.81 ± 0.02	0.79 ± 0.02	0.85 ± 0.08	0.83 ± 0.03	0.72 ± 0.06	0.71 ± 0.03	0.59 ± 0.07	0.73 ± 0.04
	5 ^a	0.86 ± 0.03	0.86 ± 0.02	0.84 ± 0.06	0.81 ± 0.03	0.73 ± 0.08	0.71 ± 0.04	0.58 ± 0.08	0.70 ± 0.04
	6 ^b	0.84 ± 0.03	0.72 ± 0.03	0.80 ± 0.07	0.84 ± 0.06	0.70 ± 0.07	0.71 ± 0.07	0.61 ± 0.08	0.64 ± 0.09

OC1—optimization criterion 1; TR—Training set; TE1—The first test set; TE2—The second test set; VA—External validation set. Nin.—Number of initial descriptors in the dataset; model—Model number; Sens.—Sensitivity; Spec.—Specificity. ^a Imbalanced training set used during optimization. ^b Manually balanced training set used during optimization.

Table 3. The average values of sensitivity and specificity obtained for models optimized using optimization criterion 2. Average values and standard deviations are given for 100 models built using selected descriptors and different permutations of objects in training set.

OC2		TR		TE1		TE2		VA	
Nin.	model	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
182	1 ^a	0.92 ± 0.02	0.92 ± 0.02	0.93 ± 0.05	0.85 ± 0.03	0.76 ± 0.07	0.72 ± 0.04	0.39 ± 0.09	0.69 ± 0.04
	2 ^a	0.93 ± 0.02	0.92 ± 0.02	0.94 ± 0.05	0.85 ± 0.04	0.76 ± 0.07	0.72 ± 0.04	0.39 ± 0.08	0.67 ± 0.04
	3 ^a	0.90 ± 0.03	0.90 ± 0.03	0.83 ± 0.07	0.86 ± 0.04	0.72 ± 0.08	0.71 ± 0.04	0.50 ± 0.08	0.76 ± 0.04
	4 ^a	0.89 ± 0.03	0.89 ± 0.03	0.84 ± 0.07	0.85 ± 0.04	0.74 ± 0.07	0.73 ± 0.04	0.50 ± 0.10	0.76 ± 0.04
	5 ^a	0.77 ± 0.02	0.79 ± 0.03	0.82 ± 0.06	0.86 ± 0.04	0.70 ± 0.09	0.74 ± 0.04	0.61 ± 0.08	0.77 ± 0.04
	6 ^a	0.80 ± 0.02	0.84 ± 0.02	0.91 ± 0.05	0.90 ± 0.04	0.73 ± 0.08	0.72 ± 0.04	0.58 ± 0.06	0.73 ± 0.03
	7 ^a	0.81 ± 0.03	0.84 ± 0.02	0.93 ± 0.04	0.88 ± 0.04	0.71 ± 0.06	0.74 ± 0.04	0.56 ± 0.07	0.72 ± 0.03
	8 ^a	0.82 ± 0.03	0.86 ± 0.02	0.84 ± 0.08	0.79 ± 0.04	0.73 ± 0.08	0.72 ± 0.04	0.57 ± 0.08	0.73 ± 0.04
	9 ^a	0.82 ± 0.03	0.87 ± 0.02	0.86 ± 0.08	0.78 ± 0.04	0.74 ± 0.09	0.71 ± 0.04	0.59 ± 0.09	0.70 ± 0.04
	10 ^b	0.92 ± 0.02	0.79 ± 0.02	0.85 ± 0.06	0.95 ± 0.05	0.72 ± 0.08	0.72 ± 0.08	0.65 ± 0.07	0.76 ± 0.08
	11 ^b	0.92 ± 0.02	0.79 ± 0.02	0.86 ± 0.06	0.94 ± 0.05	0.71 ± 0.07	0.76 ± 0.07	0.68 ± 0.06	0.69 ± 0.10
	12 ^b	0.92 ± 0.02	0.79 ± 0.02	0.85 ± 0.06	0.95 ± 0.05	0.72 ± 0.07	0.76 ± 0.07	0.67 ± 0.07	0.68 ± 0.07
	13 ^b	0.91 ± 0.02	0.78 ± 0.02	0.83 ± 0.08	0.91 ± 0.06	0.76 ± 0.07	0.75 ± 0.08	0.63 ± 0.07	0.69 ± 0.08
	14 ^b	0.92 ± 0.02	0.77 ± 0.02	0.84 ± 0.08	0.89 ± 0.06	0.73 ± 0.07	0.76 ± 0.09	0.62 ± 0.07	0.71 ± 0.08
	15 ^b	0.95 ± 0.02	0.85 ± 0.02	0.91 ± 0.05	0.95 ± 0.05	0.70 ± 0.09	0.74 ± 0.08	0.67 ± 0.06	0.81 ± 0.07
	16 ^b	0.95 ± 0.02	0.84 ± 0.02	0.92 ± 0.04	0.95 ± 0.05	0.72 ± 0.07	0.73 ± 0.07	0.69 ± 0.05	0.81 ± 0.07
	17 ^b	0.95 ± 0.02	0.84 ± 0.02	0.92 ± 0.04	0.95 ± 0.05	0.73 ± 0.09	0.75 ± 0.08	0.68 ± 0.06	0.80 ± 0.07
	18 ^b	0.88 ± 0.02	0.75 ± 0.02	0.85 ± 0.07	0.92 ± 0.06	0.72 ± 0.07	0.72 ± 0.07	0.61 ± 0.10	0.66 ± 0.08
	19 ^b	0.90 ± 0.02	0.78 ± 0.02	0.84 ± 0.07	0.82 ± 0.07	0.74 ± 0.08	0.71 ± 0.08	0.59 ± 0.08	0.63 ± 0.09
	20 ^b	0.90 ± 0.02	0.77 ± 0.02	0.80 ± 0.09	0.81 ± 0.09	0.75 ± 0.06	0.72 ± 0.09	0.71 ± 0.09	0.64 ± 0.09
	21 ^b	0.91 ± 0.02	0.79 ± 0.02	0.79 ± 0.08	0.81 ± 0.07	0.73 ± 0.05	0.72 ± 0.07	0.60 ± 0.06	0.60 ± 0.08

Table 3. Cont.

OC2		TR		TE1		TE2		VA	
Nin.	model	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
98	1 ^b	0.82 ± 0.03	0.75 ± 0.02	0.86 ± 0.08	0.82 ± 0.08	0.73 ± 0.07	0.77 ± 0.08	0.70 ± 0.07	0.64 ± 0.09
	2 ^b	0.91 ± 0.02	0.78 ± 0.02	0.78 ± 0.08	0.91 ± 0.06	0.71 ± 0.08	0.71 ± 0.08	0.61 ± 0.08	0.74 ± 0.08
	3 ^b	0.89 ± 0.02	0.76 ± 0.02	0.86 ± 0.08	0.91 ± 0.06	0.70 ± 0.07	0.74 ± 0.06	0.64 ± 0.07	0.59 ± 0.08
	4 ^b	0.89 ± 0.03	0.73 ± 0.02	0.89 ± 0.07	0.92 ± 0.05	0.73 ± 0.07	0.72 ± 0.08	0.67 ± 0.08	0.56 ± 0.09
	5 ^b	0.85 ± 0.02	0.78 ± 0.02	0.84 ± 0.06	0.93 ± 0.06	0.77 ± 0.06	0.79 ± 0.10	0.63 ± 0.08	0.74 ± 0.06
	6 ^b	0.88 ± 0.03	0.77 ± 0.03	0.78 ± 0.08	0.81 ± 0.09	0.75 ± 0.07	0.79 ± 0.07	0.70 ± 0.09	0.71 ± 0.06
	7 ^b	0.87 ± 0.02	0.78 ± 0.02	0.78 ± 0.07	0.90 ± 0.08	0.76 ± 0.06	0.84 ± 0.07	0.62 ± 0.07	0.73 ± 0.07
	8 ^b	0.93 ± 0.02	0.76 ± 0.02	0.86 ± 0.07	0.93 ± 0.05	0.81 ± 0.07	0.72 ± 0.08	0.68 ± 0.06	0.66 ± 0.06
	9 ^b	0.91 ± 0.02	0.76 ± 0.02	0.83 ± 0.08	0.91 ± 0.06	0.79 ± 0.06	0.75 ± 0.07	0.60 ± 0.06	0.68 ± 0.08
	10 ^b	0.91 ± 0.02	0.76 ± 0.02	0.80 ± 0.08	0.90 ± 0.06	0.78 ± 0.07	0.75 ± 0.07	0.60 ± 0.08	0.68 ± 0.07
	11 ^b	0.91 ± 0.02	0.75 ± 0.02	0.81 ± 0.08	0.91 ± 0.06	0.77 ± 0.07	0.75 ± 0.08	0.60 ± 0.08	0.67 ± 0.09
	12 ^b	0.82 ± 0.03	0.70 ± 0.03	0.84 ± 0.08	0.85 ± 0.07	0.75 ± 0.08	0.76 ± 0.08	0.68 ± 0.09	0.59 ± 0.07
	13 ^b	0.86 ± 0.03	0.74 ± 0.03	0.83 ± 0.08	0.88 ± 0.06	0.70 ± 0.07	0.71 ± 0.07	0.55 ± 0.08	0.71 ± 0.08
	14 ^b	0.91 ± 0.02	0.78 ± 0.02	0.80 ± 0.08	0.90 ± 0.05	0.70 ± 0.08	0.82 ± 0.06	0.67 ± 0.07	0.55 ± 0.08
	15 ^b	0.91 ± 0.02	0.77 ± 0.02	0.81 ± 0.08	0.89 ± 0.06	0.71 ± 0.06	0.81 ± 0.06	0.68 ± 0.08	0.57 ± 0.08
	16 ^b	0.86 ± 0.02	0.78 ± 0.02	0.88 ± 0.07	0.93 ± 0.06	0.73 ± 0.08	0.75 ± 0.06	0.64 ± 0.06	0.73 ± 0.07
	17 ^b	0.85 ± 0.03	0.79 ± 0.02	0.85 ± 0.08	0.92 ± 0.06	0.72 ± 0.08	0.79 ± 0.05	0.64 ± 0.06	0.74 ± 0.07
	18 ^b	0.81 ± 0.02	0.79 ± 0.02	0.87 ± 0.06	0.92 ± 0.05	0.76 ± 0.06	0.81 ± 0.06	0.65 ± 0.06	0.75 ± 0.07
	19 ^b	0.91 ± 0.02	0.77 ± 0.02	0.89 ± 0.07	0.88 ± 0.07	0.73 ± 0.09	0.71 ± 0.09	0.61 ± 0.08	0.51 ± 0.09
	20 ^b	0.87 ± 0.02	0.73 ± 0.02	0.78 ± 0.08	0.83 ± 0.06	0.73 ± 0.08	0.73 ± 0.07	0.64 ± 0.07	0.56 ± 0.08
	21 ^b	0.90 ± 0.02	0.72 ± 0.02	0.81 ± 0.07	0.88 ± 0.06	0.71 ± 0.07	0.72 ± 0.08	0.53 ± 0.08	0.63 ± 0.09
50	1 ^a	0.76 ± 0.03	0.74 ± 0.03	0.85 ± 0.07	0.78 ± 0.04	0.71 ± 0.08	0.72 ± 0.05	0.61 ± 0.08	0.70 ± 0.04
	2 ^a	0.77 ± 0.03	0.74 ± 0.04	0.86 ± 0.07	0.78 ± 0.05	0.72 ± 0.07	0.71 ± 0.04	0.61 ± 0.07	0.69 ± 0.04
	3 ^a	0.77 ± 0.04	0.74 ± 0.04	0.85 ± 0.08	0.78 ± 0.04	0.71 ± 0.08	0.71 ± 0.05	0.60 ± 0.07	0.69 ± 0.04
	4 ^b	0.97 ± 0.01	0.83 ± 0.02	0.88 ± 0.06	0.85 ± 0.08	0.71 ± 0.08	0.72 ± 0.08	0.55 ± 0.07	0.79 ± 0.08
	5 ^b	0.96 ± 0.02	0.83 ± 0.02	0.87 ± 0.07	0.85 ± 0.06	0.71 ± 0.08	0.70 ± 0.08	0.59 ± 0.07	0.75 ± 0.08

OC2—optimization criterion 2; TR—Training set; TE1—The first test set; TE2—The second test set; VA—External validation set. Nin.—Number of initial descriptors in the dataset; model—Model number; Sens.—Sensitivity; Spec.—Specificity. ^a Imbalanced training set used during optimization. ^b Manually balanced training set used during optimization.

Table 4. The average values of sensitivity and specificity obtained for models optimized using optimization criterion 3. Average values and standard deviations are given for 100 models built using selected descriptors and different permutations of objects in training set.

OC3		TR		TE1		TE2		VA	
Nin.	model	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
182	1 ^a	0.91 ± 0.03	0.91 ± 0.03	0.88 ± 0.06	0.81 ± 0.04	0.71 ± 0.08	0.70 ± 0.04	0.47 ± 0.08	0.74 ± 0.04
	2 ^a	0.90 ± 0.03	0.90 ± 0.03	0.83 ± 0.06	0.86 ± 0.04	0.70 ± 0.09	0.71 ± 0.04	0.52 ± 0.09	0.74 ± 0.04
	3 ^a	0.86 ± 0.03	0.85 ± 0.03	0.86 ± 0.06	0.83 ± 0.05	0.77 ± 0.08	0.70 ± 0.04	0.50 ± 0.10	0.73 ± 0.04
	4 ^b	0.94 ± 0.02	0.82 ± 0.02	0.93 ± 0.06	0.89 ± 0.06	0.74 ± 0.07	0.72 ± 0.08	0.63 ± 0.08	0.65 ± 0.07
	5 ^b	0.94 ± 0.02	0.81 ± 0.02	0.94 ± 0.05	0.91 ± 0.06	0.73 ± 0.08	0.71 ± 0.08	0.65 ± 0.07	0.64 ± 0.07
	6 ^b	0.93 ± 0.02	0.84 ± 0.02	0.90 ± 0.04	0.96 ± 0.05	0.72 ± 0.07	0.71 ± 0.09	0.69 ± 0.06	0.66 ± 0.09
	7 ^b	0.93 ± 0.02	0.83 ± 0.02	0.91 ± 0.04	0.96 ± 0.05	0.73 ± 0.07	0.71 ± 0.08	0.66 ± 0.07	0.66 ± 0.08
	8 ^b	0.93 ± 0.02	0.82 ± 0.02	0.90 ± 0.05	0.92 ± 0.06	0.73 ± 0.08	0.71 ± 0.07	0.61 ± 0.06	0.74 ± 0.07
	9 ^b	0.93 ± 0.02	0.82 ± 0.02	0.89 ± 0.05	0.92 ± 0.06	0.76 ± 0.07	0.73 ± 0.07	0.63 ± 0.07	0.75 ± 0.07
	10 ^b	0.89 ± 0.03	0.77 ± 0.03	0.84 ± 0.06	0.90 ± 0.07	0.75 ± 0.08	0.72 ± 0.09	0.73 ± 0.07	0.64 ± 0.07
	11 ^b	0.88 ± 0.02	0.78 ± 0.02	0.89 ± 0.07	0.91 ± 0.06	0.70 ± 0.08	0.72 ± 0.09	0.64 ± 0.08	0.68 ± 0.08
	12 ^b	0.86 ± 0.02	0.81 ± 0.03	0.88 ± 0.08	0.96 ± 0.06	0.75 ± 0.07	0.73 ± 0.08	0.65 ± 0.07	0.72 ± 0.07
	13 ^b	0.87 ± 0.02	0.81 ± 0.02	0.86 ± 0.07	0.95 ± 0.05	0.73 ± 0.07	0.72 ± 0.09	0.68 ± 0.06	0.72 ± 0.08
	14 ^b	0.87 ± 0.02	0.80 ± 0.02	0.87 ± 0.07	0.95 ± 0.05	0.75 ± 0.08	0.72 ± 0.08	0.67 ± 0.07	0.73 ± 0.07
	15 ^b	0.92 ± 0.02	0.83 ± 0.02	0.84 ± 0.06	0.92 ± 0.06	0.72 ± 0.07	0.71 ± 0.08	0.63 ± 0.06	0.73 ± 0.06
	16 ^b	0.93 ± 0.02	0.83 ± 0.02	0.85 ± 0.07	0.91 ± 0.06	0.72 ± 0.07	0.71 ± 0.09	0.64 ± 0.06	0.78 ± 0.08
	17 ^b	0.94 ± 0.02	0.81 ± 0.02	0.93 ± 0.06	0.91 ± 0.06	0.75 ± 0.07	0.77 ± 0.08	0.66 ± 0.06	0.72 ± 0.08
	18 ^b	0.94 ± 0.02	0.81 ± 0.02	0.92 ± 0.06	0.91 ± 0.07	0.74 ± 0.08	0.72 ± 0.08	0.65 ± 0.05	0.72 ± 0.08
	19 ^b	0.96 ± 0.01	0.86 ± 0.02	0.92 ± 0.06	0.95 ± 0.05	0.70 ± 0.07	0.70 ± 0.06	0.58 ± 0.06	0.72 ± 0.06
98	1 ^a	0.86 ± 0.03	0.89 ± 0.02	0.91 ± 0.06	0.76 ± 0.04	0.71 ± 0.08	0.71 ± 0.05	0.52 ± 0.10	0.65 ± 0.04
	2 ^b	0.95 ± 0.02	0.82 ± 0.02	0.79 ± 0.07	0.93 ± 0.06	0.70 ± 0.06	0.77 ± 0.06	0.57 ± 0.07	0.68 ± 0.06
	3 ^b	0.91 ± 0.02	0.78 ± 0.02	0.82 ± 0.07	0.88 ± 0.06	0.70 ± 0.08	0.71 ± 0.07	0.68 ± 0.08	0.61 ± 0.08
	4 ^b	0.91 ± 0.02	0.79 ± 0.02	0.80 ± 0.07	0.90 ± 0.05	0.73 ± 0.07	0.71 ± 0.06	0.69 ± 0.07	0.64 ± 0.07
	5 ^b	0.92 ± 0.02	0.80 ± 0.02	0.85 ± 0.07	0.95 ± 0.05	0.71 ± 0.09	0.74 ± 0.07	0.63 ± 0.07	0.65 ± 0.07
	6 ^b	0.87 ± 0.02	0.77 ± 0.02	0.78 ± 0.08	0.87 ± 0.07	0.73 ± 0.06	0.72 ± 0.07	0.63 ± 0.07	0.63 ± 0.07
	7 ^b	0.86 ± 0.02	0.75 ± 0.02	0.79 ± 0.07	0.87 ± 0.07	0.73 ± 0.07	0.71 ± 0.07	0.62 ± 0.07	0.63 ± 0.08
	8 ^b	0.87 ± 0.02	0.76 ± 0.02	0.79 ± 0.08	0.85 ± 0.08	0.73 ± 0.06	0.73 ± 0.08	0.60 ± 0.08	0.61 ± 0.09

Table 4. Cont.

OC3		TR		TE1		TE2		VA	
Nin.	model	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.	Sens.	Spec.
98	9 ^b	0.86 ± 0.03	0.75 ± 0.03	0.73 ± 0.09	0.92 ± 0.06	0.72 ± 0.07	0.72 ± 0.08	0.67 ± 0.09	0.64 ± 0.08
	10 ^b	0.87 ± 0.02	0.77 ± 0.02	0.74 ± 0.09	0.93 ± 0.06	0.75 ± 0.06	0.75 ± 0.06	0.70 ± 0.08	0.58 ± 0.09
	11 ^b	0.92 ± 0.02	0.80 ± 0.02	0.84 ± 0.07	0.91 ± 0.06	0.76 ± 0.08	0.80 ± 0.06	0.71 ± 0.07	0.68 ± 0.07
	12 ^b	0.92 ± 0.02	0.80 ± 0.02	0.84 ± 0.06	0.90 ± 0.06	0.71 ± 0.07	0.80 ± 0.08	0.66 ± 0.08	0.68 ± 0.08
	13 ^b	0.92 ± 0.02	0.79 ± 0.02	0.86 ± 0.07	0.93 ± 0.05	0.73 ± 0.08	0.80 ± 0.08	0.68 ± 0.07	0.68 ± 0.07
	14 ^b	0.86 ± 0.02	0.79 ± 0.03	0.74 ± 0.06	0.92 ± 0.06	0.71 ± 0.08	0.78 ± 0.08	0.58 ± 0.07	0.73 ± 0.09
	15 ^b	0.86 ± 0.03	0.79 ± 0.03	0.72 ± 0.07	0.91 ± 0.05	0.72 ± 0.08	0.78 ± 0.08	0.53 ± 0.07	0.68 ± 0.09
	16 ^b	0.92 ± 0.02	0.79 ± 0.02	0.88 ± 0.06	0.90 ± 0.05	0.75 ± 0.05	0.72 ± 0.07	0.64 ± 0.06	0.64 ± 0.08
	17 ^b	0.93 ± 0.02	0.80 ± 0.02	0.73 ± 0.09	0.96 ± 0.04	0.70 ± 0.07	0.71 ± 0.08	0.64 ± 0.07	0.70 ± 0.08
	18 ^b	0.93 ± 0.02	0.80 ± 0.02	0.79 ± 0.08	0.93 ± 0.05	0.72 ± 0.07	0.77 ± 0.08	0.63 ± 0.07	0.68 ± 0.08
	19 ^b	0.92 ± 0.02	0.81 ± 0.02	0.74 ± 0.08	0.87 ± 0.06	0.73 ± 0.08	0.79 ± 0.06	0.67 ± 0.09	0.65 ± 0.08
	20 ^b	0.91 ± 0.02	0.81 ± 0.02	0.83 ± 0.07	0.91 ± 0.05	0.74 ± 0.07	0.76 ± 0.07	0.56 ± 0.08	0.68 ± 0.08
	21 ^b	0.91 ± 0.02	0.81 ± 0.02	0.83 ± 0.08	0.92 ± 0.06	0.74 ± 0.07	0.75 ± 0.08	0.57 ± 0.09	0.66 ± 0.08
	22 ^b	0.92 ± 0.02	0.81 ± 0.02	0.85 ± 0.07	0.95 ± 0.05	0.76 ± 0.07	0.81 ± 0.07	0.60 ± 0.08	0.68 ± 0.09
	23 ^b	0.91 ± 0.02	0.80 ± 0.02	0.77 ± 0.08	0.87 ± 0.07	0.72 ± 0.08	0.70 ± 0.07	0.63 ± 0.07	0.65 ± 0.08
	24 ^b	0.91 ± 0.02	0.80 ± 0.03	0.78 ± 0.09	0.89 ± 0.05	0.72 ± 0.08	0.73 ± 0.07	0.59 ± 0.07	0.64 ± 0.07
	25 ^b	0.91 ± 0.02	0.81 ± 0.02	0.84 ± 0.08	0.96 ± 0.04	0.71 ± 0.07	0.74 ± 0.07	0.58 ± 0.07	0.69 ± 0.09
	26 ^b	0.94 ± 0.02	0.84 ± 0.02	0.78 ± 0.08	0.92 ± 0.06	0.83 ± 0.06	0.72 ± 0.08	0.62 ± 0.08	0.66 ± 0.07
	27 ^b	0.94 ± 0.02	0.84 ± 0.02	0.81 ± 0.09	0.93 ± 0.06	0.83 ± 0.06	0.74 ± 0.07	0.59 ± 0.08	0.66 ± 0.06
	28 ^b	0.94 ± 0.02	0.84 ± 0.02	0.81 ± 0.07	0.92 ± 0.06	0.83 ± 0.05	0.72 ± 0.09	0.61 ± 0.08	0.63 ± 0.06
	29 ^b	0.96 ± 0.02	0.85 ± 0.02	0.83 ± 0.08	0.95 ± 0.05	0.70 ± 0.09	0.74 ± 0.08	0.63 ± 0.10	0.63 ± 0.09
	30 ^b	0.96 ± 0.01	0.85 ± 0.02	0.83 ± 0.07	0.92 ± 0.06	0.72 ± 0.07	0.70 ± 0.07	0.54 ± 0.07	0.66 ± 0.08
	31 ^b	0.97 ± 0.02	0.86 ± 0.02	0.82 ± 0.07	0.94 ± 0.04	0.72 ± 0.07	0.72 ± 0.07	0.54 ± 0.07	0.67 ± 0.07
	32 ^b	0.96 ± 0.02	0.85 ± 0.02	0.85 ± 0.07	0.95 ± 0.04	0.71 ± 0.08	0.71 ± 0.06	0.55 ± 0.07	0.62 ± 0.07
	33 ^b	0.88 ± 0.03	0.77 ± 0.03	0.82 ± 0.08	0.91 ± 0.07	0.78 ± 0.07	0.73 ± 0.08	0.67 ± 0.08	0.68 ± 0.08
	34 ^b	0.88 ± 0.03	0.77 ± 0.03	0.80 ± 0.08	0.91 ± 0.06	0.77 ± 0.07	0.73 ± 0.07	0.66 ± 0.08	0.70 ± 0.08
	35 ^b	0.88 ± 0.03	0.77 ± 0.03	0.80 ± 0.07	0.91 ± 0.06	0.75 ± 0.08	0.72 ± 0.06	0.66 ± 0.09	0.68 ± 0.08
	36 ^b	0.96 ± 0.02	0.83 ± 0.02	0.74 ± 0.07	0.93 ± 0.06	0.79 ± 0.06	0.71 ± 0.06	0.65 ± 0.07	0.66 ± 0.07
	37 ^b	0.95 ± 0.02	0.82 ± 0.02	0.73 ± 0.07	0.94 ± 0.05	0.79 ± 0.06	0.71 ± 0.06	0.65 ± 0.06	0.65 ± 0.07
	38 ^b	0.96 ± 0.02	0.82 ± 0.02	0.76 ± 0.07	0.90 ± 0.06	0.78 ± 0.06	0.70 ± 0.07	0.65 ± 0.08	0.65 ± 0.06
	39 ^b	0.86 ± 0.03	0.71 ± 0.03	0.85 ± 0.08	0.83 ± 0.06	0.70 ± 0.09	0.72 ± 0.08	0.68 ± 0.07	0.63 ± 0.09
	40 ^b	0.87 ± 0.02	0.71 ± 0.03	0.82 ± 0.08	0.82 ± 0.06	0.74 ± 0.08	0.71 ± 0.08	0.66 ± 0.06	0.61 ± 0.08
	41 ^b	0.83 ± 0.02	0.77 ± 0.02	0.80 ± 0.08	0.90 ± 0.06	0.71 ± 0.07	0.78 ± 0.06	0.54 ± 0.07	0.59 ± 0.08
	42 ^b	0.82 ± 0.02	0.76 ± 0.03	0.80 ± 0.07	0.90 ± 0.06	0.76 ± 0.08	0.73 ± 0.07	0.53 ± 0.07	0.54 ± 0.06
	43 ^b	0.83 ± 0.03	0.74 ± 0.02	0.81 ± 0.08	0.91 ± 0.05	0.76 ± 0.08	0.73 ± 0.06	0.56 ± 0.07	0.54 ± 0.07
44 ^b	0.95 ± 0.02	0.79 ± 0.02	0.71 ± 0.09	0.90 ± 0.04	0.74 ± 0.07	0.71 ± 0.08	0.64 ± 0.07	0.69 ± 0.07	
45 ^b	0.94 ± 0.02	0.81 ± 0.02	0.84 ± 0.06	0.90 ± 0.06	0.71 ± 0.07	0.79 ± 0.06	0.76 ± 0.07	0.65 ± 0.07	
50	1 ^a	0.92 ± 0.02	0.92 ± 0.02	0.82 ± 0.05	0.79 ± 0.04	0.72 ± 0.07	0.70 ± 0.03	0.53 ± 0.06	0.69 ± 0.04
	2 ^a	0.91 ± 0.02	0.93 ± 0.02	0.83 ± 0.07	0.83 ± 0.04	0.72 ± 0.07	0.70 ± 0.04	0.60 ± 0.07	0.66 ± 0.04
	3 ^b	0.92 ± 0.02	0.78 ± 0.02	0.82 ± 0.07	0.83 ± 0.07	0.71 ± 0.09	0.71 ± 0.06	0.61 ± 0.07	0.77 ± 0.08
	4 ^b	0.87 ± 0.03	0.76 ± 0.02	0.84 ± 0.09	0.85 ± 0.07	0.73 ± 0.08	0.71 ± 0.07	0.66 ± 0.07	0.66 ± 0.08
	5 ^b	0.87 ± 0.03	0.75 ± 0.02	0.85 ± 0.07	0.83 ± 0.09	0.72 ± 0.08	0.72 ± 0.08	0.66 ± 0.07	0.66 ± 0.07
	6 ^b	0.95 ± 0.02	0.83 ± 0.02	0.80 ± 0.08	0.93 ± 0.06	0.72 ± 0.07	0.73 ± 0.07	0.55 ± 0.08	0.62 ± 0.07
	7 ^b	0.96 ± 0.02	0.83 ± 0.02	0.81 ± 0.07	0.93 ± 0.06	0.71 ± 0.08	0.74 ± 0.07	0.58 ± 0.07	0.63 ± 0.07

OC3—optimization criterion 3; TR—Training set; TE1—The first test set; TE2—The second test set; VA—External validation set. Nin.—Number of initial descriptors in the dataset; model—Model number; Sens.—Sensitivity; Spec.—Specificity. ^a Imbalanced training set used during optimization. ^b Manually balanced training set used during optimization.

Using the models developed by the first and second approach, consensus predictions were made for external validation set compounds. For the first approach, where the manually balanced training set was used, the obtained sensitivity and specificity for consensus predictions were 0.5 and 0.79, respectively. In the second approach, where models were developed by modified CPANN training algorithm, consensus predictions for external validation set resulted in sensitivity of 0.65 and specificity of 0.85. Comparing the consensus predictions with averages calculated from individual models, the consensus predictions based on the models from the second approach resulted in slightly increased sensitivity and significantly improved specificity. The consensus predictions were made considering and without considering the applicability domain of the models according to the method described by Minovski et al. [18]. In both cases, the consensus predictions were the same for all objects.

2.3. Descriptors

Since the composition of datasets and descriptors differed in the two approaches used for considering the imbalanced dataset, the importance of the selected descriptors was also assessed separately. In the first approach, where 24 models were selected, 143 unique descriptors were selected at least once, while in the second approach, 218 unique descriptors were used at least once. In order to assess the importance of selected descriptors, the following Equation (1) was used:

$$I_d = \sum_1^{nm} \frac{s \times a}{nm} \quad (1)$$

In Equation (1), nm is the number of models where a descriptor could be potentially selected, s is 1 if descriptor is selected or 0 if it is not, n is the number of all of selected descriptors in the model, and a is the number of all descriptors that could be selected by a model (either 50, 98, or 181).

We removed descriptors that appeared less than four times in the accepted models for both approaches to lower the chances of a descriptor being randomly selected. Descriptors reaching I_d of at least 2 are given in Tables 5 and 6 for the first and the second approach, respectively. To focus on the second modeling approach, which we assumed was superior, among the most important descriptors were two JGI descriptors (Table 6), which represented the total charge transfer (at topological distances 6 and 4). Interestingly, descriptor H%, which represented the percentage of H atoms in molecule, was placed highly in both tables. Another interpretable descriptor scoring high for the second approach was the Uc descriptor, which was correlated to the number of unsaturated bonds in a molecule.

It was difficult to draw conclusions of the role of the listed descriptors in hepatotoxicity. However, we plan to shed light on the effect of relevant descriptors in the future. To do so, we plan to improve the counter-propagation neural networks methodology to better discriminate compounds found in activity cliffs, which was a frequent event in the dataset used. Hopefully, future work will lead to a better understanding of the mechanisms of drug-induced liver injury.

Table 5. Most important descriptors for models using the first modeling approach.

Descriptor	Description	I_d
J_D/Dt	Balaban-like index from distance/detour matrix	8.945987
GATS5v	Geary autocorrelation of lag 5 weighted by van der Waals volume	8.042931
H%	Percentage of H atoms	7.579833
SpMin1_Bh(s)	Smallest eigenvalue n. 1 of Burden matrix weighted by I-state	6.506972
CATS2D_02_AA	CATS2D Acceptor-Acceptor at lag 02	5.406386
IC2	Information content index (neighborhood symmetry of 2-order)	5.3672
GATS1v	Geary autocorrelation of lag 1 weighted by van der Waals volume	4.810587
GATS2v	Geary autocorrelation of lag 2 weighted by van der Waals volume	4.727913
BAC	Balaban centric index	4.58365
SpPosA_X	Normalized spectral positive sum from chi matrix	4.303807
P_VSA_LogP_6	P_VSA-like on LogP, bin 6	3.877732
C-006	CH2RX	3.640303
P_VSA_e_3	P_VSA-like on Sanderson electronegativity, bin 3	3.475949
P_VSA_MR_2	P_VSA-like on Molar Refractivity, bin 2	3.236547
MATS8m	Moran autocorrelation of lag 8 weighted by mass	3.138591
nCsp3	Number of sp3 hybridized carbon atoms	2.675997
PDI	Packing density index	2.585321
P_VSA_m_4	P_VSA-like on mass, bin 4	2.511289
SpAD_EA(dm)	Spectral absolute deviation from edge adjacency mat. weighted by dipole moment	2.35969
CATS2D_04_AA	CATS2D Acceptor-Acceptor at lag 04	2.332174
X5Av	Average valence connectivity index of order 5	2.196837
X5A	Average connectivity index of order 5	2.100552

Table 6. Most important descriptors for models using the second modeling approach.

Descriptor	Description	I _d
JGI6	Mean topological charge index of order 6	3.502671
JGI4	Mean topological charge index of order 4	3.398279
SdssC	Sum of dssC E-states	3.372717
H%	Percentage of H atoms	3.287295
Uc	Unsaturation count	3.071672
P_VSA_LogP_6	P_VSA-like on LogP, bin 6	2.985918
H-052	H attached to C0(sp3) with 1X attached to next C	2.805648
MAXDN	Maximal electrotopological negative variation	2.620215
Chi1_EA(dm)	Connectivity-like index of order 1 from edge adjacency mat. Weighted by dipole moment	2.576782
SpMax_B(m)	Leading eigenvalue from Burden matrix weighted by mass	2.540987
GATS5m	Geary autocorrelation of lag 5 weighted by mass	2.480224
SpAD_EA(dm)	Spectral absolute deviation from edge adjacency mat. Weighted by Dipole moment	2.416726
GATS1i	Geary autocorrelation of lag 1 weighted by ionization potential	2.365205
SsssN	Sum of ssssN E-states	2.352358
SpMAD_EA(bo)	Spectral mean absolute deviation from edge adjacency mat. Weighted by bond order	2.344266
ChiA_B(s)	Average Randic-like index from Burden matrix weighted by I-State	2.338868
NssO	Number of atoms of type ssO	2.223777
VE2sign_A	Average coefficient of the last eigenvector from adjacency matrix	2.169277
MATS2p	Moran autocorrelation of lag 2 weighted by polarizability	2.169277
MATS1p	Moran autocorrelation of lag 1 weighted by polarizability	2.103962
SpMin1_Bh(v)	Smallest eigenvalue n. 1 of Burden matrix weighted by van der Waals volume	2.089443
ChiA_B(v)	Average Randic-like index from Burden matrix weighted by van der Waals volume	2.043667
Rbrid	Ring bridge count	2.040469
nCsp3	Number of sp3 hybridized Carbon atoms	2.038696
C-040	R-C(=X)-X / R-C#X / X=C=X	2.022

3. Materials and Methods

3.1. Dataset

The dataset used in the study was compiled using data from five literature sources related to drug-induced liver injury (DILI) [12,21–24]. Names of the compounds in the literature sources were used to extract SMILES from PubChem [25]. From the collection of acquired data, chemical entities, such as proteins, larger peptides, inorganic compounds, compounds containing elements, which were very uncommon in the dataset, and mixtures of compounds were removed. We also removed compounds that are found in drugs where the only route of administration is topical according to DrugBank [26]. Compounds were, where applicable, protonated or deprotonated in Pipeline Pilot [27] to achieve canonical SMILES representation of molecules. Finally, duplicates with the same active pharmaceutical ingredient were removed, while their information for toxicity classes was retained. Three sources [21,23,24] tackled the DILI problem with more than two classes. We assigned compounds to two classes, DILI-positive and DILI-negative, according to the following rules. From source [21], most-DILI-concern and less-DILI-concern were considered DILI-positive and no-DILI-concern were considered DILI-negative. The toxicity classes “severe DILI” and “non-severe DILI” from source [23] were considered DILI-positive, while the non-DILI class was considered DILI-negative. Compounds found in [24] were considered DILI-positive if they belonged to class A or B and DILI-negative if they belonged to class E, C, D, and E* classes from that source were not assigned to either the DILI-positive or DILI-negative class, and remained as such since they represent compounds with a lower likelihood of being toxic or nontoxic. In a way, such compounds represent unreliable data. Next, we checked if there were any conflicts in toxicity classes in the data from different sources. For example, if a compound was present in two out of five sources and classified as DILI-positive, we assigned the DILI-positive class to it. Similarly, if compound was present in three sources and classified as DILI-negative in two of them and as D in Livertox, the compound was not considered for further use (possible conflict of classes). From source [22], we extracted only compounds that had been manually checked for DILI (named calibration set in the source). In this way, we obtained 524 compounds that were used in our study.

3.2. Selection of Descriptors and Datasets

Descriptors used to represent compounds were 0-, 1-, and 2-dimensional descriptors calculated using Dragon [28] software. The selection of descriptors for modeling was made in two steps. The first step involved the removal of correlated and constant descriptors, and in the second step, the descriptors were reduced using Kohonen mapping. In the paper by Topliss and Edwards [29], where multiple regression models were studied, a limit of 0.8 for the pairwise correlation coefficient was suggested. However, we did not want to remove too many descriptors in the first step, thus we used a higher limit. The descriptors with pairwise correlation coefficient below 0.95 were kept. Descriptors with more than 70% constant values in the entire set were discarded. In that way, we obtained 580 descriptors, encoding a dataset of 524 compounds. Descriptors were normalized using z-scores. Compounds were mapped on 2-dimensional space using the Kohonen artificial neural network to select subsets used for the modeling. First, the objects for the external validation set were selected from the Kohonen top-map. We tried to maximize covered chemical space, while also attempting that validation set objects would reflect the density of all objects on the map, reflecting important aspects of diversity and closeness of sets [30]. Since we used two different approaches for dealing with an imbalanced dataset, each approach requiring a different size training set, we also used the validation sets of different sizes. After the exclusion of the external validation set, descriptors were again checked for constant values. Then, the number of descriptors was reduced. Kohonen maps of different sizes (5×5 , 7×7 , 10×10) were used to map the objects of the transposed data matrix. Kohonen top-maps were obtained, from which the closest and farthest objects (descriptor), considering the Euclidean distance to the neuron, were selected from each neuron. This resulted in sets containing 50, 98, and 181 descriptors used for modeling. The remaining compounds (excluding the validation set) were again mapped on the Kohonen top-map, this time using 50, 98, and 181 descriptors. The training set and two test sets (represented by TE1 and TE2) were selected from each top-map. Again, we tried to cover chemical space for each of the three sets, again taking the density of objects on the map into account to some degree. However, if a neuron was excited by up to two objects, they were favourably put in the training set. Descriptor values for all sets were normalized with z-score, with mean and standard deviation corresponding to the training set. Test set TE1 was used for model optimization; test set TE2 was used to estimate the performance of the models and select models, which were then externally validated using the external validation set.

Information about the compounds and descriptors used are available in Supplementary Files “distribution_of_compounds_into_sets.xlsx” and “compound_descriptors.xlsx”. The workflow of the modeling approach used is schematically presented in Figure 4.

3.3. Models

When developing classification models, we used three different sets of descriptors, as explained in the previous Section 3.2. Due to the imbalanced nature of our dataset, having more non-toxic compounds than toxic, we also considered training with balanced training sets and an imbalanced training set. In that way, three different combinations of training and test sets were obtained for each of the two validation sets. The balanced training set contained 108 objects from each class, while the imbalanced training set was made up of 108 DILI-positive and 296 DILI-negative compounds. The imbalanced training set was further used in two ways. The first approach was to use all objects in the training set for each epoch of training. Secondly, due to the imbalanced nature of the training set, having 108 DILI-positive compounds and 296 DILI-negative compounds in the set, we used a balanced subset of 216 compounds in each epoch of training, which consisted of all 108 DILI-positive compounds and randomly selected 108 DILI-negative compounds for each epoch. Training was performed using the counter-propagation artificial neural network. The first approach of dealing with the imbalanced training set (i.e., not to deal with it) yielded no satisfactory results, so the first approach further in the text refers to the fixed balanced training set and the second approach refers to the imbalanced training set that was used as balanced in each epoch of training.

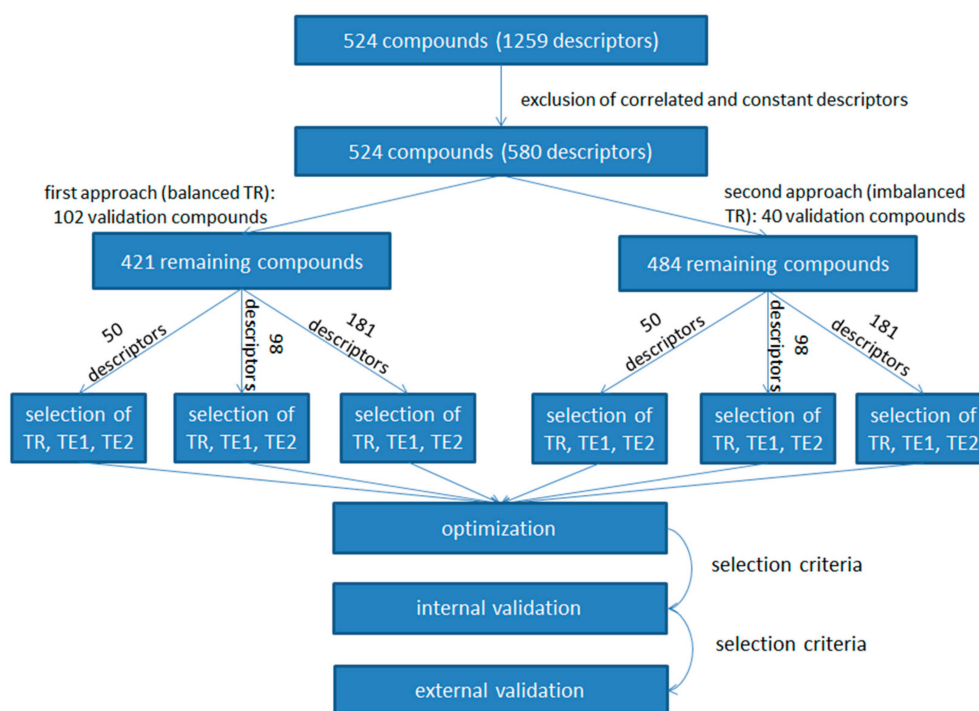


Figure 4. Modeling workflow. A total of 1259 Dragon descriptors were calculated for our dataset. The number of descriptors was reduced based on pairwise correlation and the frequency of the most common value. For the first modeling approach, where selection of the training set resulted in balanced classes, 102 validation objects were removed. For the second approach, where selection of the training set resulted in imbalanced classes, 40 objects were removed. For both approaches, the number of descriptors was further reduced: 50, 98, and 181 descriptors were selected. Based on the selected descriptors, objects for the training set (TR), test set 1 (TE1), and test set 2 (TE2) were selected. We applied the genetic algorithm for optimization of the models. Models passing the criteria were selected for internal validation with TE2. Lastly, models passing internal validation were externally validated with the corresponding validation set.

3.4. Optimization of Models Using Genetic Algorithm

We applied the genetic algorithm for the optimization of models. Optimizations were performed on test set TE1 or both the training and test set TE1 together, depending on the selected optimization criterion. We performed 7020 optimizations in total, each of them having a unique combination of an optimization criterion, a parameter describing the effect of number of selected descriptors on the cost function, number of neurons in the network, and the parameters used for genetic algorithm, such as the number of chromosomes for crossover. We selected three optimization criteria to optimize models. The selected optimization criteria (OC1, OC2, OC3) are written in Equations (2)–(4). The optimization criteria were based on the Matthew correlation coefficient (MCC) of test set TE1, the minimum of sensitivity and specificity of test set TE1, and the product of MCC of training and MCC of test set TE1. The effect of the number of selected descriptors ($N_{selected}$) on the optimization criterion was taken into account using Equation (5), where p is the parameter defined prior to each optimization and $N_{descriptors}$ is the total number of descriptors used in the training set. Matthew correlation coefficient, sensitivity, and specificity were calculated according to Equations (6)–(8), respectively. In Equations (6)–(8), TP , TN , FP , and FN indicate the number of true positive, true negative, false positive, and false negative predictions, respectively.

From all optimizations, we selected models where the sensitivity and specificity for all three sets (training, test TE1, and test TE2) were at least 0.7, and all three MCC were at least 0.5. Additionally, if any of the best three chromosomes during optimization had on average sensitivity and specificity

of at least 0.7 for all three sets in the last 20 generations of optimization, all three models were also selected. Afterwards, we trained each such selected model 100 times using a random order of training objects. Models with average sensitivity and average specificity of 0.7 or higher for all three sets were finally selected as acceptable models.

$$\text{OC1} = \min(\text{sensitivity}(\text{test set TE1}), \text{specificity}(\text{test set TE1})) \cdot f, \quad (2)$$

$$\text{OC2} = \text{MCC}(\text{test set TE1}) \cdot f, \quad (3)$$

$$\text{OC3} = \text{MCC}(\text{training set}) \cdot \text{MCC}(\text{test set TE1}) \cdot f, \quad (4)$$

$$f = 1 - p \cdot (N_{\text{selected}} - 1) / N_{\text{descriptors}}, \quad (5)$$

$$\text{MCC} = (TP \cdot TN - FP \cdot FN) / (((TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN))^{1/2}), \quad (6)$$

$$\text{sensitivity} = TP / (TP + FN), \quad (7)$$

$$\text{specificity} = TN / (TN + FP). \quad (8)$$

3.5. Kohonen and Counter-Propagation Artificial Neural Networks

Classification models were developed using counter-propagation artificial neural networks (CPANNs). Detailed descriptions of CPANNs can be found in the literature [19,20]. Here, a brief explanation of the training algorithm is given with a description of modifications made to the algorithm that was used when the number of objects in the two target classes of the training dataset was considerably different.

Counter-propagation artificial neural network models can be described as a 3D matrix of weights divided into two layers. A single column of weights in each layer represents one neuron. The upper layer of neurons in CPANNs is known as the Kohonen layer, and the layer beneath is the output layer, also known as the Grossberg layer. The weights in the Kohonen layer correspond to independent variables of objects (e.g., molecular descriptors), and the weights in the output layer correspond to dependent variables (e.g., target toxicity classes) of the input objects.

During the training, competitive learning is used in the Kohonen layer. The independent variables of an input object are compared to all the neurons in the Kohonen layer and the neuron that is the most similar to the object is selected as the central neuron, often called the “winning neuron”. The similarity between an object and a neuron is determined by calculating the Euclidean distance between the neuron weights and independent variables of the object. The neuron with the shortest Euclidean distance is selected as the central neuron. After the central neuron is determined, the corrections of the weights can be made. The corrections depend on a neighborhood function, learning rate, and differences between the current neuron weights and the corresponding variable of the input object. The neighborhood function defines the amount of correction made at a given topological distance from the central neuron. In our study, a triangular neighborhood function was used, where the correction was maximal on the central neuron and decreased with topological distance from the central neuron. The neighborhood function also decreased during the training so that at the end of the training only the weights corresponding to the central neuron were corrected. Learning rate, $\eta(t)$, linearly decreased during the training according to Equation 9, with the largest value (v_{max}) in the first iteration ($t = 1$) of the training and the smallest value (v_{min}) at the end of the training ($t = t_{\text{max}}$):

$$\eta(t) = v_{\text{min}} + (v_{\text{max}} - v_{\text{min}}) \cdot (t_{\text{max}} - t) / (t_{\text{max}} - 1). \quad (9)$$

The weights are corrected so that the weights in the Kohonen layer become more similar to the independent variables, and the weights in the output layer become more similar to the target values of

the input object. The weights are updated according to Equation 10, where the second term in the sum represents the correction made at given iteration t :

$$w_{i,j}^{\text{new}} = w_{i,j}^{\text{old}} + \eta(t) \cdot a(c,j,t) \cdot (x_i - w_{i,j}^{\text{old}}). \quad (10)$$

In Equation (10), $w_{i,j}^{\text{new}}$ and $w_{i,j}^{\text{old}}$ represent the new and previous values of the weight on neuron j , corresponding to variable i of the input object; x_i represents the actual value of the variable i of the input object, and $a(c,j,t)$ is the value of neighborhood function for neuron j and central neuron c .

The training lasts for a predefined number of epochs, where one epoch means that each object from the training set is used exactly once for the training. Here, we introduced a modification that we made to the training procedure when the number of objects in one class was considerably larger than in the other class, i.e., when the imbalanced dataset was used for training. Normally, all the input objects are used in one epoch of training. Due to imbalanced data, we made a random subsampling of input objects without repetition prior to each epoch of training, so that the subsample contained a balanced number of objects from each class. Therefore, one epoch in such a case was equal to the number of training iterations when each object from the subsample was used exactly once. Practically, that meant that the neural network was trained with the number of objects equal to the number of objects in the subsample, and in each epoch a different training set was used. In this way, we still used all data from the training set, however the objects from the majority class were less frequently used during the training than those from the minority class.

3.6. Genetic Algorithm

The genetic algorithm was used for the optimization of the counter-propagation artificial neural network models. The genetic algorithm was employed for the selection of descriptors used in the model and for the optimization of learning rate parameters. A description of genetic algorithms is given in the literature [31]. Genetic algorithm optimization starts with the creation of an initial random population of chromosomes. Chromosomes may define the features to be used in a model and other parameters that may be involved in the development of the model. Using the encoded information from the chromosomes, models are built for each chromosome and ranked according to preselected optimization criterion. A new population of chromosomes is then created by mating chromosomes with good optimization criteria. Mutation operators are applied to the new population of chromosomes to introduce random genetic alterations in the chromosomes. The created population of chromosomes is again used for the creation of new models, and all the steps repeat until a predefined number of populations is reached or when no improvement in optimization criterion is observed.

4. Conclusions

Drug-induced liver injury (DILI) presents one of the major concerns in the drug development process. Using *in silico* approaches, one may efficiently evaluate the possible hepatotoxic effect of drugs. Imbalance of the dataset and activity cliffs present major obstacles to developing useful QSAR models. In this study, we encountered both difficulties when performing the modeling of DILI using counter-propagation artificial neural networks (CPANNs). The modification of the classical training algorithm with the integration of random subsampling within the training procedure of CPANN presented in this work was intended to solve the problem of dataset imbalance. A number of optimizations of CPANNs were performed using a genetic algorithm to find acceptable models. Based on the comparison of the results, we conclude that using an imbalanced set with balanced subsampling in each learning epoch is a better approach compared to using a fixed balanced set in the case of the counter-propagation artificial neural network learning methodology. The models obtained using the modified CPANN training algorithm were applied for consensus predictions and showed sensitivity of 0.65 and specificity of 0.85 for compounds in the external validation set.

Supplementary Materials: The following are available online, Supplementary1.pdf: Results of Principal Component Analysis, compound_descriptors.xlsx, distribution_of_compounds_into_sets.xlsx.

Author Contributions: Conceptualization, B.B. and V.D.; methodology, B.B.; formal analysis, B.B.; investigation, B.B.; data curation, B.B.; writing—original draft preparation, B.B. and V.D.; visualization, B.B.; supervision, V.D. All authors have read and agreed to the published version of the manuscript.

Funding: The work was funded by Javna Agencija za Raziskovalno Dejavnost Republike Slovenije (ARRS; Slovenian Research Agency) through PhD study grant for young researchers (B.B.) and through research program P1-0017 (V.D.).

Acknowledgments: The authors thank Prof. Dr. Marjana Novič for discussions regarding QSAR modeling approaches.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Waring, M.J.; Arrowsmith, J.; Leach, A.R.; Leeson, P.D.; Mandrell, S.; Owen, R.M.; Pairaudeau, G.; Pennie, W.D.; Pickett, S.D.; Wang, J.; et al. An analysis of the attrition of drug candidates from four major pharmaceutical companies. *Nat. Rev. Drug Discov.* **2015**, *14*, 475–486. [[CrossRef](#)] [[PubMed](#)]
2. Watkins, P.B. Drug Safety Sciences and the Bottleneck in Drug Development. *Clin. Pharmacol. Ther.* **2011**, *89*, 788–790. [[CrossRef](#)] [[PubMed](#)]
3. Onakpoya, I.J.; Heneghan, C.J.; Aronson, J.K. Post-marketing withdrawal of 462 medicinal products because of adverse drug reactions: A systematic review of the world literature. *BMC Med.* **2016**, *14*, 10. [[CrossRef](#)] [[PubMed](#)]
4. Clark, M.; Steger-Hartmann, T. A big data approach to the concordance of the toxicity of pharmaceuticals in animals and humans. *Regul. Toxicol. Pharmacol.* **2018**, *96*, 94–105. [[CrossRef](#)]
5. Fenwick, N.; Griffin, G.; Gauthier, C. The welfare of animals used in science: How the “Three Rs” ethic guides improvements. *Can. Vet. J.* **2009**, *50*, 523–530.
6. Soldatow, V.Y.; Lecluyse, E.L.; Griffith, L.G.; Rusyn, I. In vitro models for liver toxicity testing. *Toxicol. Res. (Camb)*. **2013**, *2*, 23–39. [[CrossRef](#)]
7. Grimm, F.A.; Iwata, Y.; Sirenko, O.; Bittner, M.; Rusyn, I. High-content assay multiplexing for toxicity screening in induced pluripotent stem cell-derived cardiomyocytes and hepatocytes. *Assay Drug Dev. Technol.* **2015**, *13*, 529–546. [[CrossRef](#)]
8. Pettinato, G.; Lehoux, S.; Ramanathan, R.; Salem, M.M.; He, L.X.; Muse, O.; Flaumenhaft, R.; Thompson, M.T.; Rouse, E.A.; Cummings, R.D.; et al. Generation of fully functional hepatocyte-like organoids from human induced pluripotent stem cells mixed with Endothelial Cells. *Sci. Rep.* **2019**, *9*, 1–21. [[CrossRef](#)]
9. Verneti, L.A.; Senutovitch, N.; Boltz, R.; DeBiasio, R.; Ying Shun, T.; Gough, A.; Taylor, D.L. A human liver microphysiology platform for investigating physiology, drug safety, and disease models. *Exp. Biol. Med.* **2016**, *241*, 101–114. [[CrossRef](#)]
10. Liew, C.Y.; Lim, Y.C.; Yap, C.W. Mixed learning algorithms and features ensemble in hepatotoxicity prediction. *J. Comput. Aided. Mol. Des.* **2011**, *25*, 855–871. [[CrossRef](#)]
11. Przybylak, K.R.; Cronin, M.T.D. In silico models for drug-induced liver injury - Current status. *Expert Opin. Drug Metab. Toxicol.* **2012**, *8*, 201–217. [[CrossRef](#)] [[PubMed](#)]
12. Cruz-Monteagudo, M.; Cordeiro, M.N.D.S.; Borges, F. Computational chemistry approach for the early detection of drug-induced idiosyncratic liver toxicity. *J. Comput. Chem.* **2008**, *29*, 533–549. [[CrossRef](#)] [[PubMed](#)]
13. Ekins, S.; Williams, A.J.; Xu, J.J. A predictive ligand-based Bayesian model for human drug-induced liver injury. *Drug Metab. Dispos.* **2010**, *38*, 2302–2308. [[CrossRef](#)] [[PubMed](#)]
14. Fourches, D.; Barnes, J.C.; Day, N.C.; Bradley, P.; Reed, J.Z.; Tropsha, A. Cheminformatics analysis of assertions mined from literature that describe drug-induced liver injury in different species. *Chem. Res. Toxicol.* **2010**, *23*, 171–183. [[CrossRef](#)]
15. Kotsampasakou, E.; Montanari, F.; Ecker, G.F. Predicting drug-induced liver injury: The importance of data curation. *Toxicology* **2017**, *389*, 139–145. [[CrossRef](#)]

16. Wang, Y.; Xiao, Q.; Chen, P.; Wang, B. In silico prediction of drug-induced liver injury based on ensemble classifier method. *Int. J. Mol. Sci.* **2019**, *20*, 4106. [[CrossRef](#)] [[PubMed](#)]
17. Zhu, X.-W.; Li, S.-J. In Silico Prediction of Drug-Induced Liver Injury Based on Adverse Drug Reaction Reports. *Toxicol. Sci.* **2017**, *158*, 391–400. [[CrossRef](#)]
18. Minovski, N.; Župerl, Š.; Drgan, V.; Novič, M. Assessment of applicability domain for multivariate counter-propagation artificial neural network predictive models by minimum Euclidean distance space analysis: A case study. *Anal. Chim. Acta* **2013**, *759*, 28–42. [[CrossRef](#)]
19. Zupan, J.; Novič, M.; Gasteiger, J. Neural networks with counter-propagation learning strategy used for modelling. *Chemom. Intell. Lab. Syst.* **1995**, *27*, 175–187. [[CrossRef](#)]
20. Zupan, J.; Novič, M.; Ruisánchez, I. Kohonen and counterpropagation artificial neural networks in analytical chemistry. *Chemom. Intell. Lab. Syst.* **1997**, *38*, 1–23. [[CrossRef](#)]
21. Chen, M.; Suzuki, A.; Thakkar, S.; Yu, K.; Hu, C.; Tong, W. DILrank: The largest reference drug list ranked by the risk for developing drug-induced liver injury in humans. *Drug Discov. Today* **2016**, *21*, 648–653. [[CrossRef](#)] [[PubMed](#)]
22. Zhu, X.; Kruhlak, N.L. Construction and analysis of a human hepatotoxicity database suitable for QSAR modeling using post-market safety data. *Toxicology* **2014**, *321*, 62–72. [[CrossRef](#)]
23. Zhang, J.; Doshi, U.; Suzuki, A.; Chang, C.W.; Borlak, J.; Li, A.P.; Tong, W. Evaluation of multiple mechanism-based toxicity endpoints in primary cultured human hepatocytes for the identification of drugs with clinical hepatotoxicity: Results from 152 marketed drugs with known liver injury profiles. *Chem. Biol. Interact.* **2016**, *255*, 3–11. [[CrossRef](#)] [[PubMed](#)]
24. Hoofnagle, J.H.; Serrano, J.; Knoblen, J.E.; Navarro, V.J. LiverTox: A website on drug-induced liver injury. *Hepatology* **2013**, *57*, 873–874. [[CrossRef](#)] [[PubMed](#)]
25. Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B.A.; Thiessen, P.A.; Yu, B.; et al. PubChem 2019 update: Improved access to chemical data. *Nucleic Acids Res.* **2019**, *47*, D1102–D1109. [[CrossRef](#)]
26. Wishart, D.S.; Feunang, Y.D.; Guo, A.C.; Lo, E.J.; Marcu, A.; Grant, J.R.; Sajed, T.; Johnson, D.; Li, C.; Sayeeda, Z.; et al. DrugBank 5.0: A major update to the DrugBank database for 2018. *Nucleic Acids Res.* **2018**, *46*, D1074–D1082. [[CrossRef](#)]
27. *BIOVIA Pipeline Pilot (Release 2014)*; Dassault Systèmes: San Diego, CA, USA, 2016.
28. Mauri, A.; Consonni, V.; Pavan, M.; Todeschini, R. DRAGON software: An easy approach to molecular descriptor calculations. *MATCH Commun. Math. Comput. Chem.* **2006**, *56*, 237–248.
29. Topliss, J.G.; Edward, R.P. Chance Factors in Studies of Quantitative Structure-Activity Relationship. *J. Med. Chem.* **1979**, *22*, 1238–1244. [[CrossRef](#)]
30. Golbraikh, A.; Tropsha, A. Predictive QSAR modeling based on diversity sampling of experimental datasets for the training and test set selection. *J. Comput. Aided. Mol. Des.* **2002**, *16*, 357–369. [[CrossRef](#)]
31. Leardi, R. *Nature-Inspired Methods in Chemometrics: Genetic Algorithms and Artificial Neural Networks*; Elsevier: Amsterdam, The Netherlands, 2003; ISBN 9780444513502.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).