Impact of Dataset Composition on the Performance of Machine Learning Models in Virtual Screening

Anastasiia Krokhina (Bar-Ilan University, University of Strasbourg)
Advisor: Hanoch Senderowitz (Bar-Ilan University)

Introduction

Common practices suggest that predictive Machine Learning (ML) models should be developed using well-balanced datasets. This study explores the influence of dataset class composition on the binary classification performances of machine learning models in the context of virtual screening. Specifically, we evaluated the performance of k-nearest neighbors (KNN)[1] and extreme gradient boosting (XGBoost)[2] models trained on datasets with balanced and imbalanced class distributions on selected examples taken from two widely used virtual screening datasets: LIT-PCBA[3] and DUD-E[4]. Furthermore, during model training, hyperparameter tuning was performed via grid search cross-validation (GSCV) using two virtual screening-aware metrics, namely, area under the receiver operating characteristic curve (ROC AUC) and Matthew's correlation coefficient (MCC).

Methodology

Data: The DUD-E database was selected for its extensive collection of active compounds and decoys. At the same time, the LIT-PCBA dataset provides robust experimental data for model validation and was designed to mitigate the biases presented in DUD-E datasets. We took three datasets from each database to conduct the experiment.

Descriptors: The Canvas program calculated 1D and 2D molecular descriptors (Physicochemical, Topological, and LigFilter) for all compounds from all datasets, resulting in 754 descriptors for each molecule.

Machine Learning: Both datasets underwent preprocessing to ensure compatibility with the machine learning workflows, including normalization and splitting into balanced and imbalanced training and test sets (Table 1).

ML models were developed and evaluated using Python3.9, leveraging Scikit-learn 1.2.2 as the primary library for data preprocessing, model training, and evaluation. Hyperparameter optimization was performed through GSCV, ensuring optimal model performance.

This study employed two ML algorithms, KNN and XGBoost 2.1.1, implemented as a standalone library. These models were chosen for their distinct methodologies and ability to complement each other in analyzing the datasets.

Metrics: Hyperparameters were optimized, and final models were evaluated using two metrics: ROC AUC, a threshold-independent metric, and the MCC, a threshold-dependent metric. Both metrics were computed using Scikit-learn's built-in functionalities. MCC score was normalized for easier comparison, as described in [5].

The full ML pipeline is presented in he Figure 1.

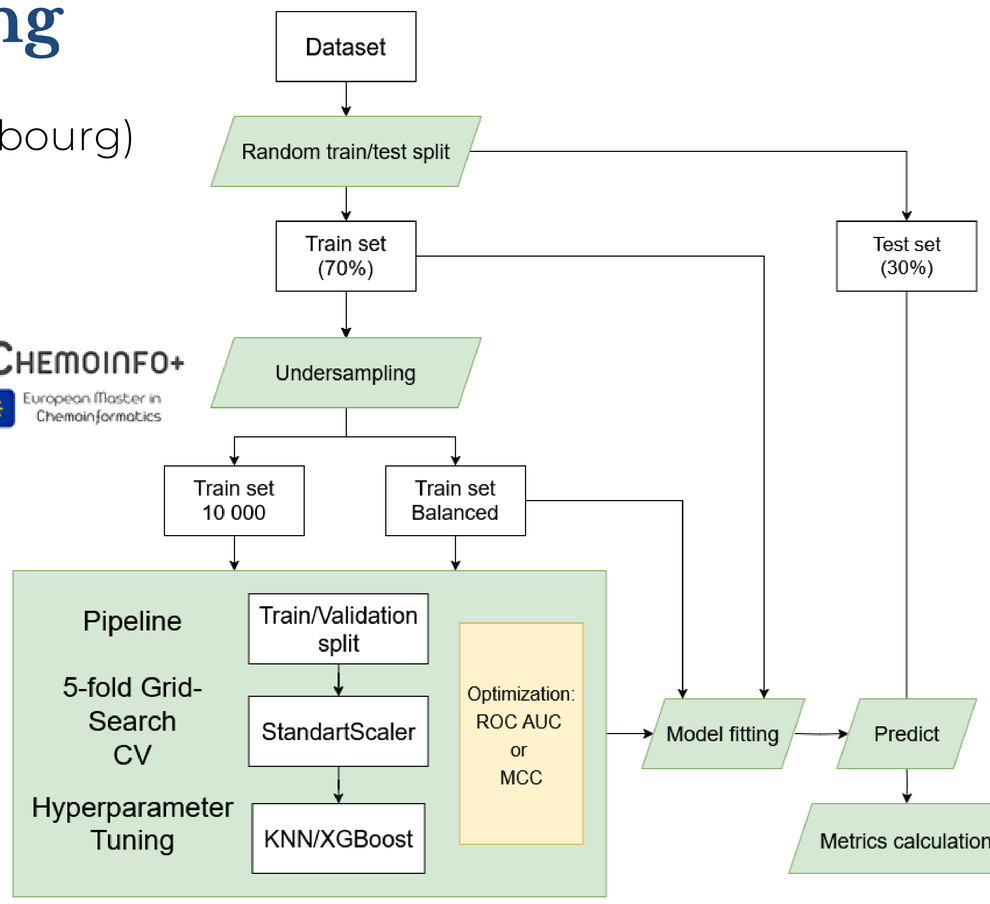

Figure 1. Machine Learning Pipeline.

Results

Our analysis reveals significant differences in model performance depending on the dataset composition and optimization metric (Table 2). Models optimized using the MCC during GSCV consistently demonstrated superior performance when trained on imbalanced datasets compared to those trained on balanced datasets. This superiority highlights the practical utility of imbalanced data sets in scenarios where real-world data are inherently skewed.

Despite this, models optimized for MCC often exhibited lower ROC AUC scores. However, they achieved higher precision for the active molecule class, evidenced by an increased true positive rate among all predictions labeled as positive. Additionally, these models demonstrated improved recall for the decoy class, reflected in a higher true negative rate. These results underscore the trade-offs introduced by MCC optimization.

Furthermore, models trained on imbalanced data and optimized through GSCV-MCC consistently displayed a lower false positive rate across all experiments than models trained on balanced data. This reduction in false positives emphasizes the effectiveness of imbalanced datasets and MCC optimization in prioritizing correct classifications over potentially misleading results.

The XGBoost models optimized using ROC AUC and imbalanced data yielded results similar to those optimized with balanced data. However, for the KNN model, the imbalanced data optimization led to a notable improvement in performance, outperforming its balanced data counterpart based on Precision values. This suggests that the imbalanced data allowed the KNN model to better capture the underlying patterns, contributing to a more robust predictive capability.

Table 1. Data sets composition.

Dataset	aldh1			fen1			vdr			aces			mk14			thrb		
	Total size	Actives	Inactives	Total size	Actives	Inactives	Total size	Actives	Inactives	Total size	Actives	Inactives	Total size	Actives	Inactives	Total size	Actives	Inactives
Imbalanced	179867	7284	172583	381415	433	380982	408185	960	407225	26704	452	26252	36321	577	35744	27396	460	26936
Imbalanced Train	125906	5098	120808	266990	303	266687	285729	672	285057	18692	316	18376	25423	403	25020	19177	322	18855
Imbalanced Test	53961	2186	51775	114425	130	114295	122456	288	122168	8012	136	7876	10898	174	10724	8219	138	8081
Imbalanced Train GSCV	9999	405	9594	10000	11	9989	10000	24	9976	9747	165	9582	10000	159	9841	14392	242	14150
Balanced Train	10196	5098	5098	606	303	303	1344	672	672	632	316	316	806	403	403	644	322	322

Table 2. Calculated metrics for XGBoost and KNN models for the test set.

XGBoost Test

Data name:	ALDH1 LIT-PCBA				FEN1 LIT-PCBA				VDR LIT-PCBA				ACES DUD-E				MK14 DUD-E				THRB DUD-E			
Proportion:	Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced	
Optimized metric:	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC
ROC AUC	0.71	0.53	0.71	0.71	0.84	0.50	0.84	0.85	0.74	0.52	0.74	0.76	0.97	0.90	0.96	0.96	0.95	0.84	0.96	0.96	0.96	0.93	0.96	0.95
MCC	0.17	0.16	0.17	0.17	0.06	0.00	0.07	0.07	0.05	0.14	0.05	0.06	0.48	0.86	0.47	0.45	0.49	0.78	0.50	0.50	0.54	0.92	0.51	0.49
Precision(0)	0.98	0.96	0.98	0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00
Precision(1)	0.09	0.48	0.09	0.09	0.01	0.00	0.01	0.01	0.01	0.41	0.01	0.01	0.24	0.92	0.24	0.21	0.27	0.92	0.27	0.27	0.32	0.99	0.28	0.26
Recall(0)	0.67	1.00	0.69	0.69	0.85	1.00	0.86	0.88	0.75	1.00	0.75	0.78	0.95	1.00	0.95	0.94	0.96	1.00	0.96	0.96	0.97	1.00	0.96	0.95
Recall(1)	0.75	0.06	0.72	0.72	0.82	0.00	0.82	0.82	0.72	0.05	0.72	0.74	0.99	0.80	0.97	0.99	0.95	0.67	0.96	0.96	0.96	0.86	0.96	0.95
F1(0)	0.80	0.98	0.81	0.81	0.92	1.00	0.93	0.93	0.86	1.00	0.86	0.87	0.97	1.00	0.97	0.97	0.98	1.00	0.98	0.98	0.98	1.00	0.98	0.98
F1(1)	0.16	0.11	0.16	0.16	0.01	0.00	0.01	0.02	0.01	0.09	0.01	0.02	0.39	0.86	0.38	0.35	0.42	0.78	0.42	0.42	0.48	0.92	0.44	0.41

KNN Test

Data name:	ALDH1 LIT-PCBA				FEN1 LIT-PCBA				VDR LIT-PCBA				ACES DUD-E				MK14 DUD-E				THRB DUD-E			
Proportion:	Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced		Imbalanced		Balanced	
Optimized metric:	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC	ROC	MCC
ROC AUC	0.71	0.57	0.71	0.71	0.57	0.70	0.83	0.82	0.53	0.64	0.74	0.76	0.92	0.95	0.89	0.93	0.86	0.90	0.83	0.92	0.95	0.96	0.89	0.95
MCC	0.18	0.27	0.18	0.18	0.22	0.44	0.07	0.06	0.09	0.32	0.05	0.06	0.86	0.89	0.24	0.41	0.77	0.80	0.17	0.35	0.92	0.91	0.25	0.40
Precision(0)	0.98	0.96	0.98	0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Precision(1)	0.09	0.60	0.09	0.09	0.37	0.49	0.01	0.01	0.11	0.37	0.01	0.01	0.88	0.88	0.08	0.19	0.82	0.82	0.05	0.15	0.95	0.90	0.08	0.18
Recall(0)	0.71	1.00	0.71	0.71	1.00	1.00	0.88	0.87	1.00	1.00	0.76	0.75	1.00	1.00	0.79	0.93	1.00	1.00	0.67	0.91	1.00	1.00	0.80	0.93
Recall(1)	0.71	0.13	0.72	0.72	0.14	0.40	0.78	0.77	0.07	0.28	0.73	0.77	0.85	0.90	0.99	0.93	0.73	0.79	0.99	0.94	0.89	0.93	0.99	0.96
F1(0)	0.82	0.98	0.82	0.82	1.00	1.00	0.94	0.93	1.00	1.00	0.86	0.86	1.00	1.00	0.88	0.96	1.00	1.00	0.80	0.95	1.00	1.00	0.89	0.96
F1(1)	0.16	0.22	0.17	0.17	0.20	0.44	0.01	0.01	0.09	0.32	0.01	0.01	0.86	0.89	0.14	0.32	0.77	0.80	0.09	0.25	0.92	0.91	0.14	0.31

Conclusion

This study highlights the effectiveness of using imbalanced datasets and MCC optimization in improving model performance for classification tasks, especially in real-world scenarios where data imbalances are common. While optimizing for MCC may reduce ROC AUC scores, it enhances precision for active molecules and recall for decoys, offering a practical balance between precision and false positive rate. These findings underscore the importance of tailoring data set construction and optimization strategies to the challenges of imbalanced datasets.

Literature

1. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011).

2. Chen, T. & Guestrin, C. XGBoost: A Scalable Tree Boosting System. in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 785–794 (2016). doi:10.1145/2939672.2939785.

3. Tran-Nguyen, V.-K., Jacquemard, C. & Rognan, D. LIT-PCBA: An Unbiased Data Set for Machine Learning and Virtual Screening. *J. Chem. Inf. Model.* **60**, 4263–4273 (2020).

4. Chen, L. *et al.* Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening. *PLOS ONE* **14**, e0220113 (2019).

5. Chicco, D. & Jurman, G. The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification. *BioData Min* **16**, 4 (2023).