CONTENTS

1. Machine learning projects

Assessment of bioconcentration and acute toxicity of chemicals in fish

A major demand of the modern era is the development of new, modern tools for designing pharmaceutical products and manufacturing processes with a minimal environmental footprint. The purpose of this project is derived from the fact that pharmaceutical compounds (and their derivatives) end up in the aquatic environment through wastewater, due to their use or disposal. This poses a significant threat to aquatic organisms. The bioaccumulation of chemical compounds has been primarily studied in fish, and we know it can cause both acute and chronic toxicity. However, in vivo studies on bioaccumulation (which relates to chronic toxicity) and acute toxicity are associated with high costs and ethical concerns due to the use and sacrifice of animals. For this reason, machine learning models have been developed for predicting the bioaccumulation and the acute ecotoxicity (bioactivity) of chemical compounds, eventually helping to establish guidelines or filters in the synthesis of green chemical compounds. The previous computational studies and models do not provide chemical insights that could be applied for green-by-design initiatives and are not accurate enough to reliably predict the environmental impact of compounds or classify chemicals based on their bioaccumulation or acute ecotoxicity.

The resulted models were trained using experimental data to quantitatively determine the relationship between chemical structure and acute toxicity or bioaccumulation in fish. These models identify bioaccumulative and ecotoxic compounds, strictly defining the physicochemical properties (molecular descriptors as numerical features) and chemical groups (Morgan and MACCS fingerprints as binary features) that render the compounds as ecotoxic. BCF (ratio of the compound's concentration in fish at steady state to its concentration in water) and 96-hour LC50 (compound’s lethal concentration to 50% of fish species in 96-hours tests) were utilized as metrics for the bioconcentration and acute toxicity of compounds in fish respectively.

The steps of the in-silico process are as follows:

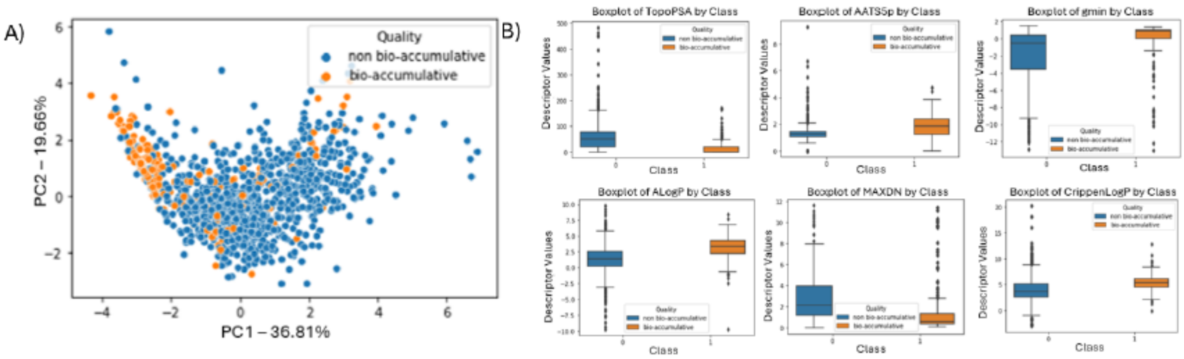
1. Collection of compounds with known BCF and 96-hour LC50 values from publicly available databases and curation of the generated datasets.
2. Normalization of SMILES (text characters representing the structure of chemical molecules) to a specific format (canonical SMILES after the removal of salts, compounds with ambiguous 3D structure) under pH 7.4.
3. Generation of features (e.g. molecular fingerprints via RDkit) for each SMILES structure.
4. Classification of compounds into bioaccumulative or non-bioaccumulative, based on the known BCF (3.3 log units as threshold based on EU REACH regulation) and LC50 (1 mg/L as threshold based on CLP rules) value.
5. Data preprocessing, including the reduction of features to the most important ones using the *feature importance* method (in trained datasets of classification models) and to the most “informative” ones using the *mutual information* method (in trained datasets of regression models).
6. Balancing the two classes in the classification random training datasets and investigation of target variable’s most optimal distribution in the regression random training datasets. These practises were followed by identifying the best model through 10 epochs of each model training on those random training datasets and evaluating each model on a separate random sample, independent of the training data.
7. Optimization of model parameters (Bayesian-type hyperparameter tuning) and statistical analysis of the features used in the classification model training.
8. External validation of final models in other publicly available databases.

The following table and figures summarize the efficiency (*Table 1*) and the interpretability of the final models. For summarization purposes, only the analysis of BCF classification models is displayed. Based on this analysis, the primary physicochemical properties of bioaccumulative compounds (*Figure 1.C*) and the chemical substructures (*Figure 2.A* and *Figure 3.A*) predominantly present in bioaccumulative compounds are extracted.

Table 1. External validation metrics of the most accurate models on publicly available datasets (ADORE ‘s t_F2F and KOWall3 datasets for LC50 and BCF models respectively).

Target variable	**Classifier**	**Training features**	**Accuracy (%)**	**ROC AUC (%)**
BCF	Gradient Boosting	Molecular Descriptors	88.43	90.95
LC50	ExtraTrees	Molecular Descriptors	90.44	96.06

Target variable	**Regressor**	**Training features**	**R2**	**RMSE**
BCF	ExtraTrees	Molecular Descriptors	0.77	0.62
LC50	ExtraTrees	Molecular Descriptors	0.81	0.57



C)

Descriptors	Descriptor type	P-value	Weight in PC1 (36.81% of variance)	Weight in PC2 (19.66% of variance)	SHAP value (mean)
TopoPSA	Topological polar surface area	2.72e-40	0.440	- 0.243	0.424
AATS5p	Autocorrelation	2.53e-22	- 0.393	- 0.424	0.192
gmin	Atom type E- state	5.86e-27	- 0.551	0.582	0.185
ALogP	ALOGP	7.59e-37	0.105	0.489	0.306
MAXDN	Atom type E- state	1.42e-28	0.001	- 0.230	0.228
CrippenLogP	Crippen logP	1.54e-25	- 0.007	- 0.002	0.513

Figure 1. Analysis of most accurate BCF classification model. **A)** 2D PCA plot using BCF classification model training features. PC1 (x΄x) and PC2 (y΄y) account for 36.81% and 19,66 of the variance, respectively. **B)** Boxplots distribution of the most impactful features across the two classes in the BCF training dataset. **C)** The most impactful features for the distinction of bio-accumulative compounds in multiple fish species, along with P-values from the Mann-Whitney test and their coefficients in PC1 and PC2. Moreover, the relevant types of the molecular descriptors are given, as well as the mean SHAP values of features, representing their impact on the model's performance. Each feature represents a specific physicochemical property.

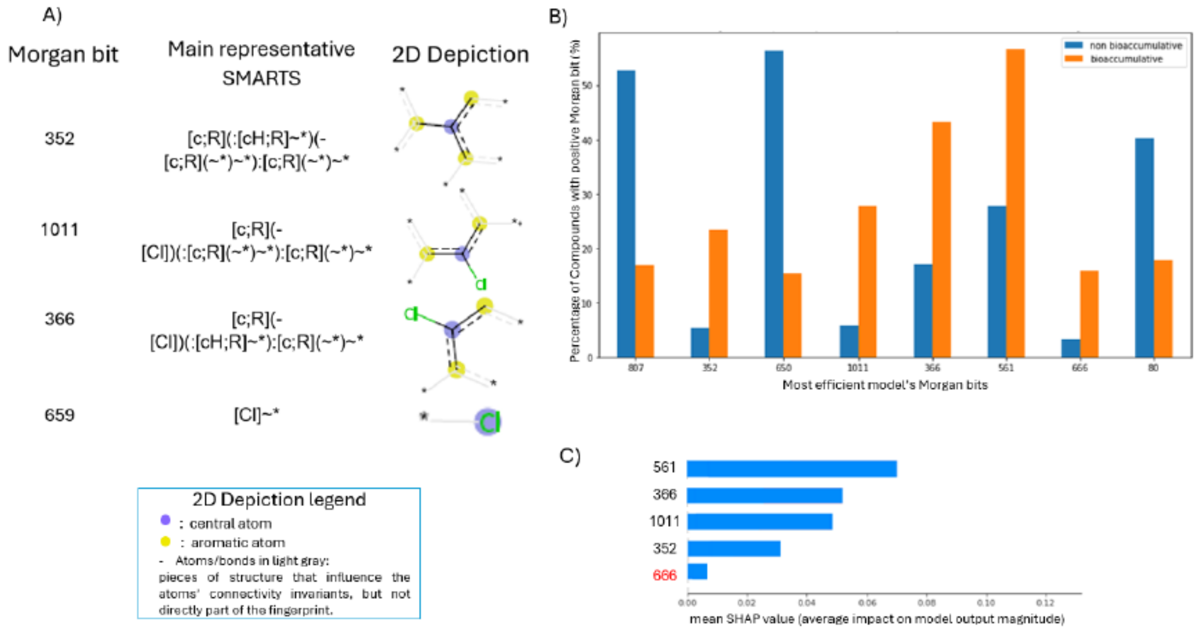


Figure 2. Analysis of BCF model utilizing Morgan fingerprints as training features. **A)** Morgan bits with significant influence (high SHAP value) on model's decision-making process and with a value of "1" primarily in class 1 (bioaccumulative) compounds. The main representative SMARTS (text characters representing chemical substructures) patterns and corresponding substructures are also presented. **B)** Percentage distribution of compounds per class, where each model's training bit is assigned a value of "1" (Histogram). **C)** SHAP analysis of model's training Morgan bits with value "1" primarily in bioaccumulative compounds. The red-colored bit was not considered as significant feature, because it was assigned with small mean SHAP value (Morgan bit 666).

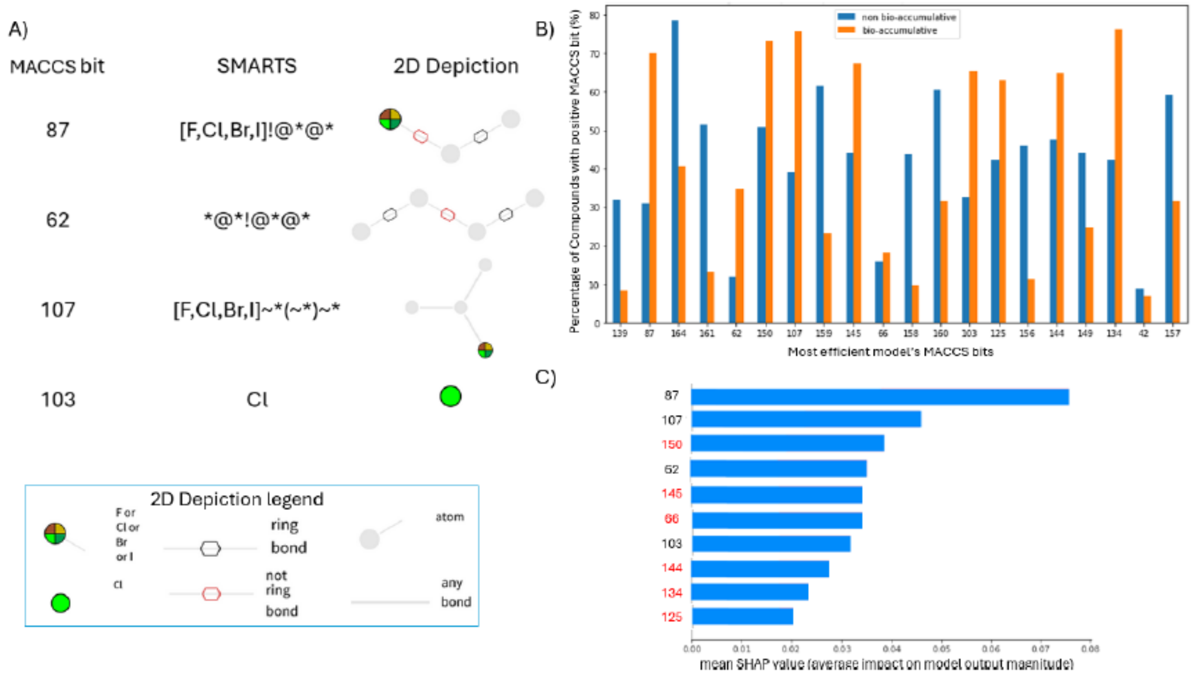


Figure 3. Analysis of BCF model utilizing MACCS fingerprints as training features. **A)** MACCS bits with significant influence (high SHAP value) on model's decision-making process and with a value of "1" primarily in class 1 (bioaccumulative) compounds. The representative SMARTS patterns and corresponding substructures are also

presented. **B)** Percentage distribution of compounds per class, where each model's training bit is assigned a value of "1" (Histogram). **C)** SHAP analysis of model's training MACCS bits with value "1" primarily in bioaccumulative compounds. The red-colored bits were not considered as significant features, because they either were positive in a relatively large percentage of non-bioaccumulative compounds (MACCS bits 150, 145, 66, 144, 125) or were assigned with small mean SHAP value (MACCS bits 134, 125).

Drug-Protein binding affinity prediction using Graph Neural Networks

In this ongoing project, I am developing and training (heterogenous) Graph Neural Network (GNN) models to predict drug-protein binding affinities represented by experimental pKi values. The project leverages PyTorch Geometric and integrates domain knowledge on chemical interactions (interaction fingerprints), physicochemical properties (descriptors) and structural components (fingerprints) of the reactants with GNN techniques. Key metrics (RMSE, MAE, R2) are tracked and averaged across epochs and runs.

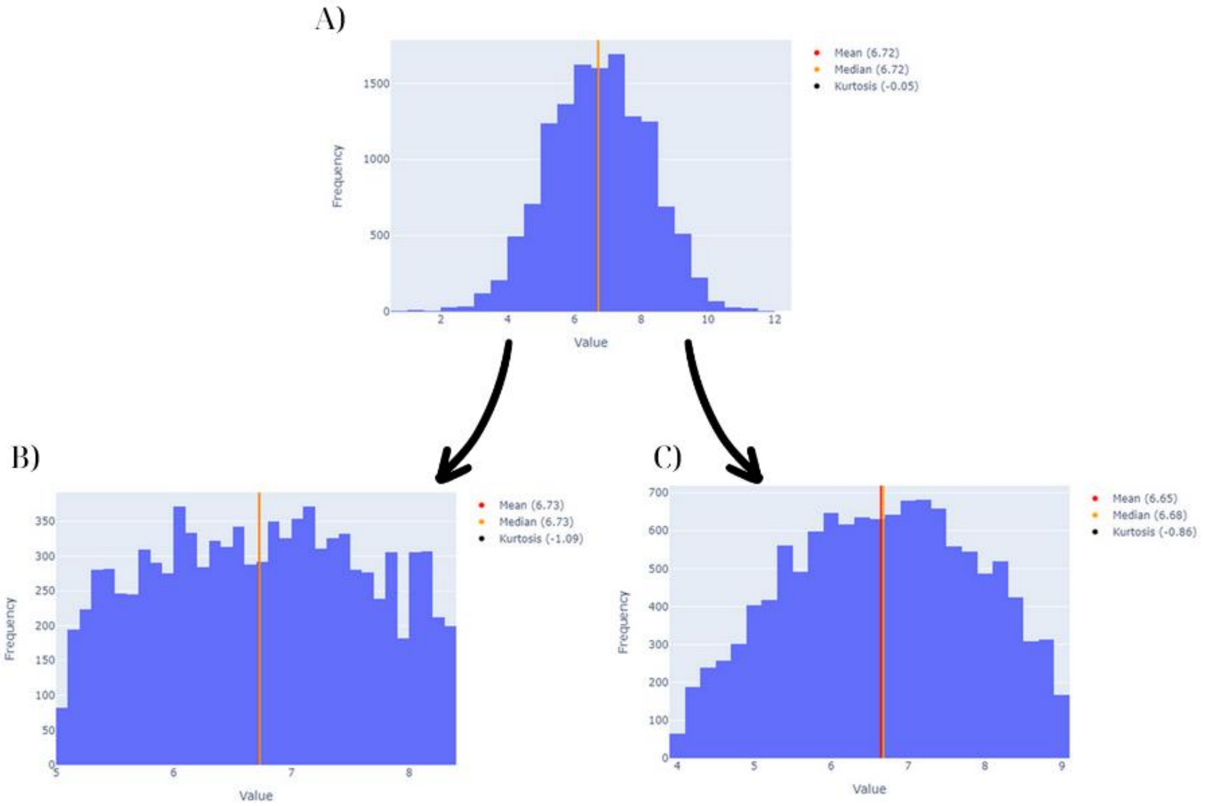


Figure 4. Investigation for the optimal distribution of target variable pki for model training. **A)** Distribution of pki values in the initial training dataset. **B)** Distribution following outlier removal using the Interquartile Range (IQR) method, with lower and upper fences set at 5.05 and 8.40, respectively. **C)** Distribution after trimming the extreme 5% of the data, excluding observations with pki < 4 or pki > 9.

Large virtual screening and ML model distinguishing false positive results

In this study, we performed large-scale molecular docking using GNINA and AutoDock Vina, generating multiple poses for thousands of small molecules across diverse protein targets. While docking is an essential step in virtual screening pipelines, it often produces a high proportion of false positive poses with low docking score that complicate downstream analyses. To address this, we implemented a machine learning–based classification framework trained on interaction Protein–Ligand Extended Connectivity (PLEC) fingerprints of protein–ligand complexes, enabling systematic discrimination between correct and incorrect docking results. The final model was trained on docking poses generated from the refined set of the PDBbind database, ensuring high-quality structural input. By integrating conformal prediction into the workflow, our model not only improved the reliability of pose selection but also quantified prediction uncertainty, offering a robust methodological advance for filtering docking outputs in structure-based drug discovery.

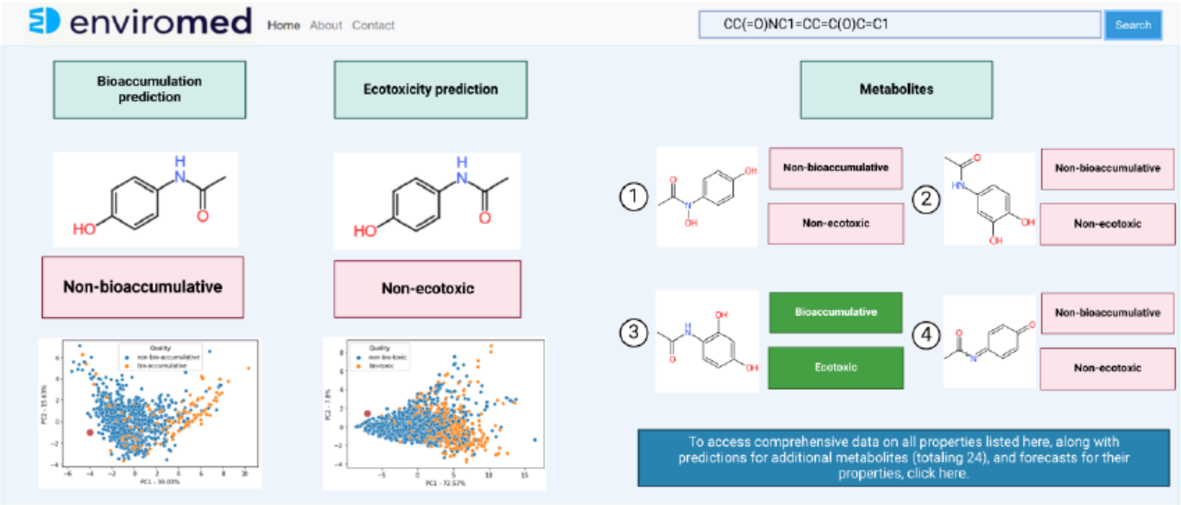
Table 2. Validation metrics obtained from CASF-2016 dataset utilized as external dataset.

Classifier	**Accuracy**	**Specificity**	**Sensitivity**	**ROC AUC**
Random Forest	88.80	96.51	61.84	79.18
XGBoost	85.61	90.01	72.08	81.04
SVM	88.19	91.51	77.69	84.60
MLP	84.47	86.13	79.41	82.77

2. Full-stack developing project

G.AI.A: a next generation toolbox for greener pharmaceutical design

The designed platform leverages the relevant ML models to classify chemical molecules and their predicted primary metabolites (integration of SyGMA, a software for *in silico* prediction of metabolites in human body) based on bioaccumulation and acute toxicity in fish. It also provides structural and physicochemical insights to substantiate the predicted label for each parent compound. Built on the Django framework, the platform integrates a Python-based backend for classifying compounds with molecular weight greater than 150 Da. It also generates 2D PCA plots and boxplot graphs of key variables, allowing users to compare the compound's physicochemical characteristics to other compounds within the same class. Moreover, potential chemical groups of the compound mainly present in class 1 compounds are highlighted. The front-end, developed using HTML and CSS, offers a user-friendly interface for seamless interaction with the system, enabling users to input chemical compounds in SMILES format and explore detailed chemical classifications. This combination ensures robust functionality while maintaining accessibility for non-technical users.



To access comprehensive data on all properties listed here, along with predictions for additional metabolites (totaling 24), and forecasts for their properties, click here.

Figure 5. Part of G.AI.A front-end with paracetamol as input.

3. NPL Tasks

Integration of LLMs in the C-Natural platform

C-Natural is a platform designed to aggregate knowledge about natural products from publicly available databases. The scope of this project was to retrieve and summarize information associated with patents and scientific literature related to each natural product. To achieve this, several NLP models were evaluated for processing scientific texts. The final implemented NLP model (BART architecture augmented with SciBERT model embeddings) was capable of summarizing input (scientific) text to less than half the original word count while maintaining a high ROUGE score, demonstrating its efficiency and accuracy in text abstraction and compression. This result was made possible through targeted web scraping and text selection from every relevant API.