Insights into Pharmacokinetic Properties for Exposure Chemicals: Predictive Modelling of Human Plasma Fraction Unbound (fu) and Hepatocyte Intrinsic Clearance (Clint) Data Using Machine Learning

Souvik Pore, Kunal Roy*

Drug Theoretics and Cheminformatics Laboratory

Department of Pharmaceutical Technology, Jadavpur University

188 Raja S C Mullick Road, 700032, Kolkata, India

Supplementary Information-2 (SI-2)

*Corresponding Author

Prof. Kunal Roy, Phone: +919831594140; Fax: +91-33-2837-1078

Email: kunalroy_in@yahoo.com; kunal.roy@jadavpuruniversity.in

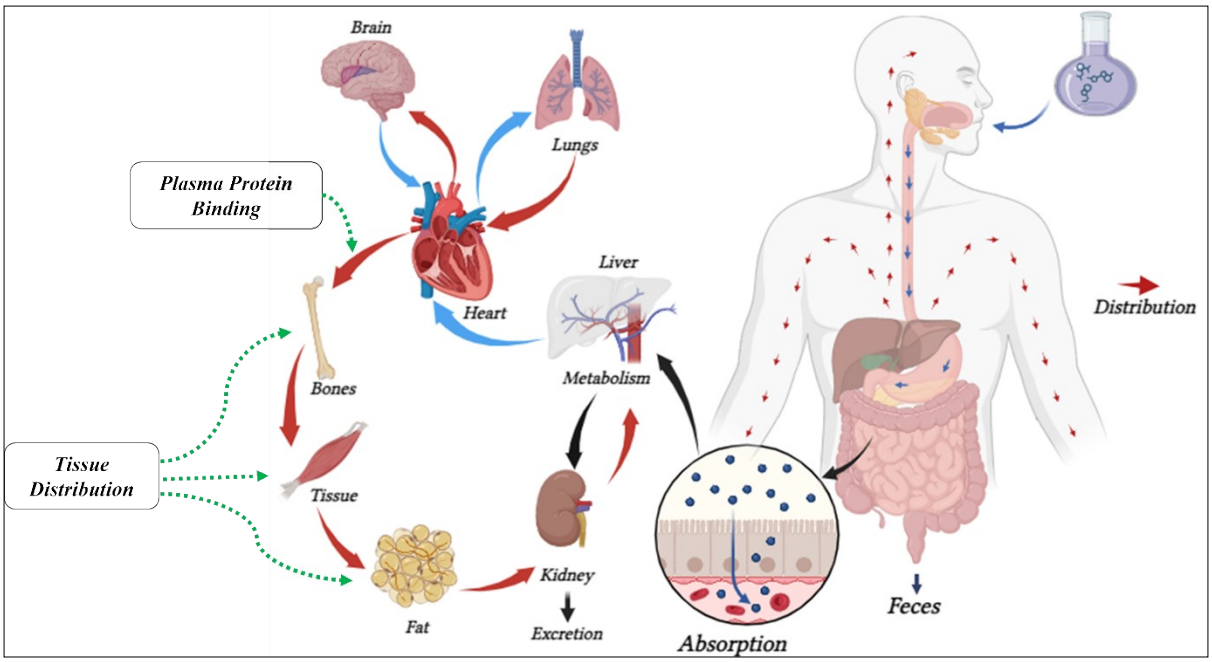

Figure S1: Pharmacokinetic distribution of an ingested chemical substance.

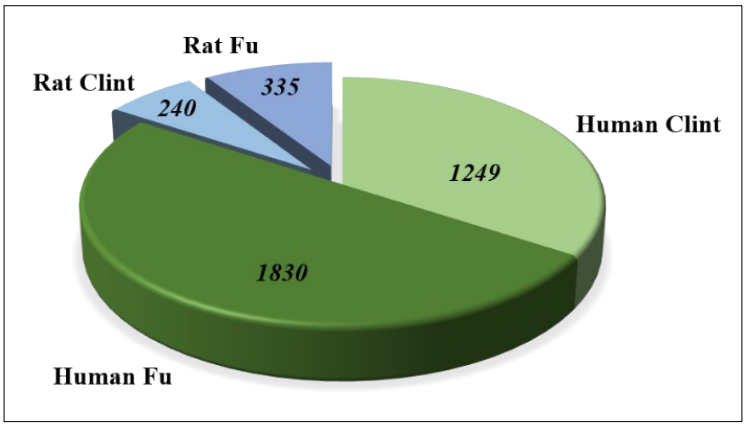

Figure S2: Number of data points in the fu and Clint datasets.

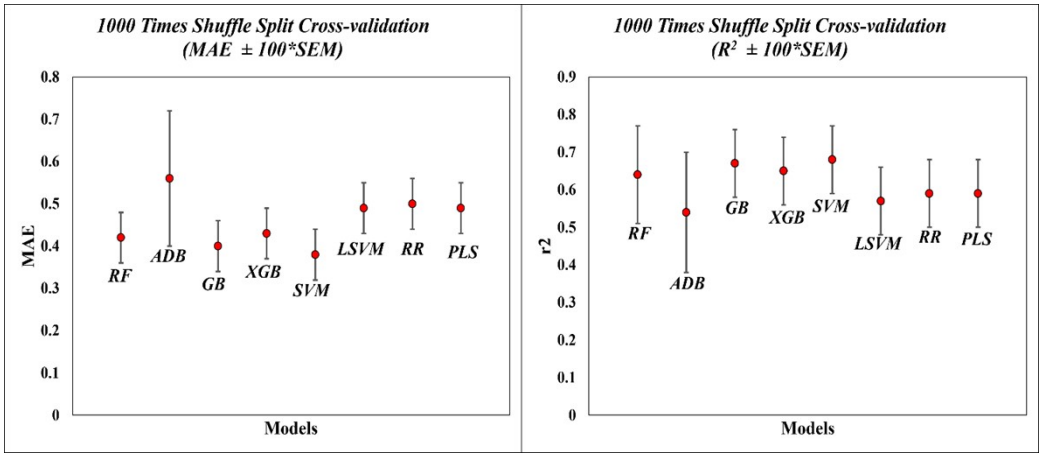

Figure S3: 1000 times shuffle split Cross-validation statistics (mean ± 100 SEM) (fraction unbound data).

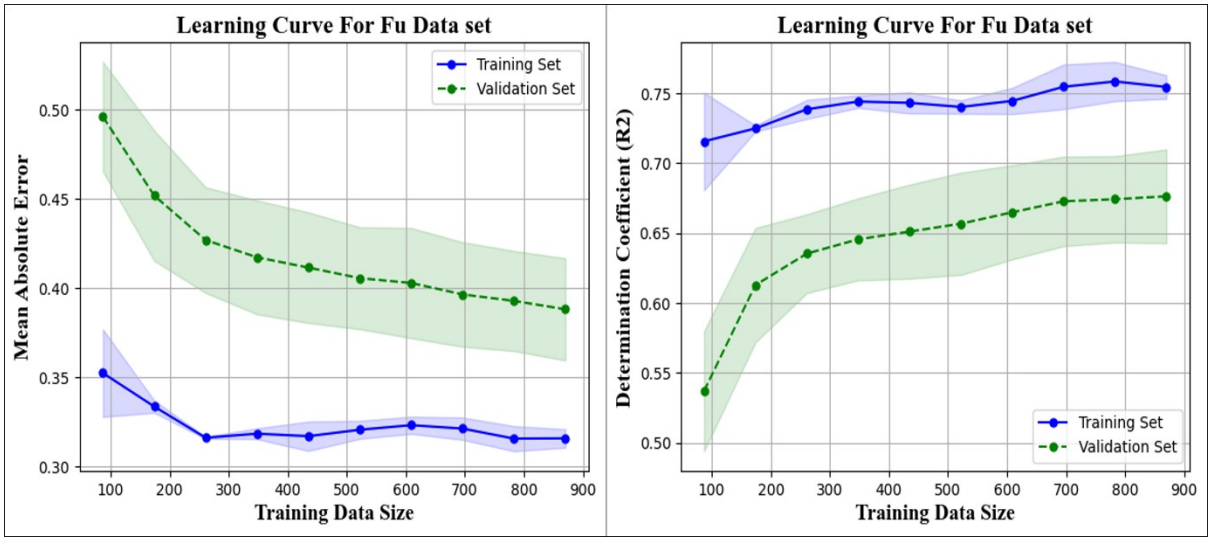

Figure S4. Learning curves for the SVM model by considering Mean Absolute Error (MAE) and R^2 as the objective functions (fraction unbound data)

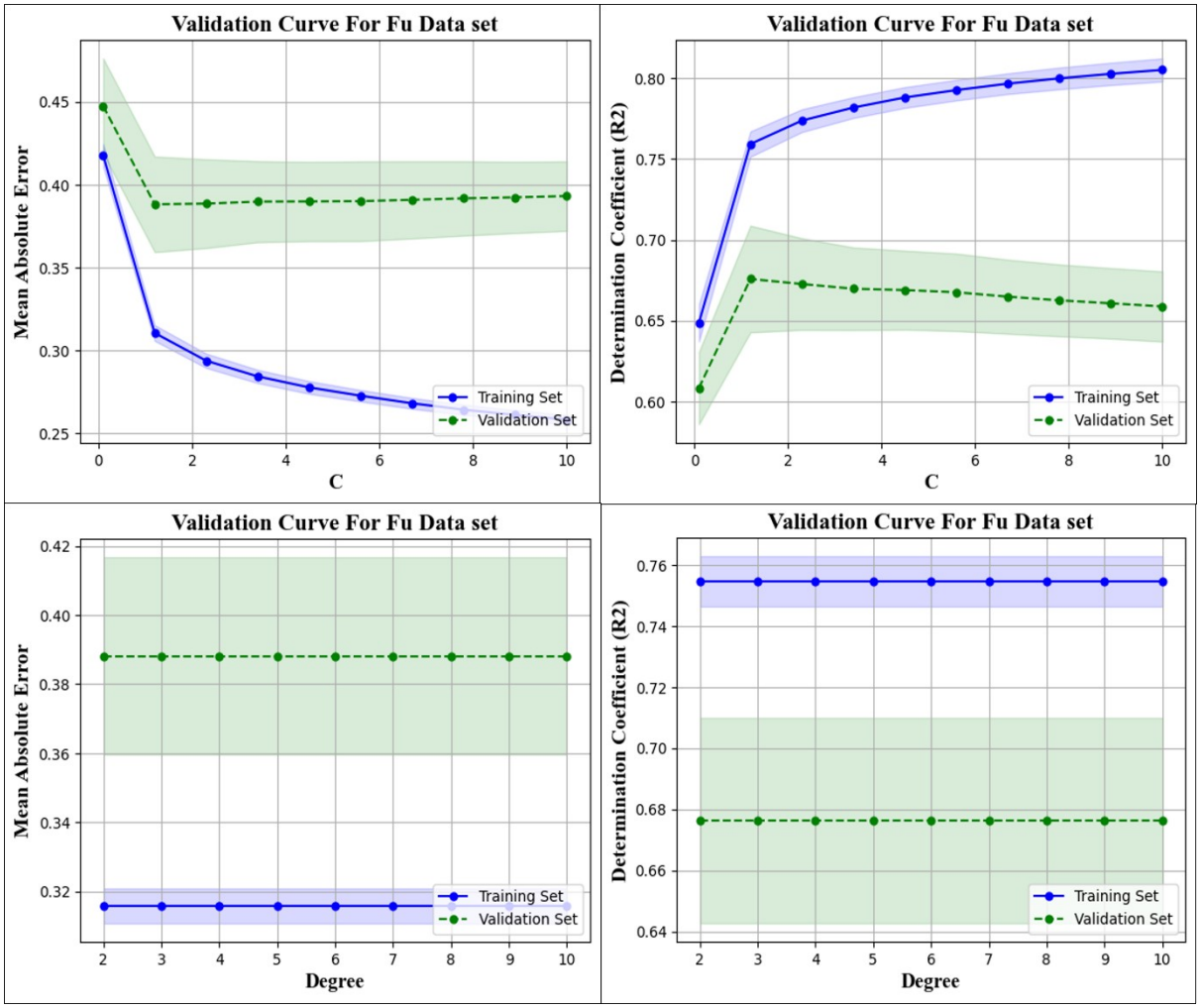

Figure S5. Validation curve for SVM model by considering Mean Absolute Error (MAE) and R^2 as the objective functions, for parameters C and degree (fraction unbound data).

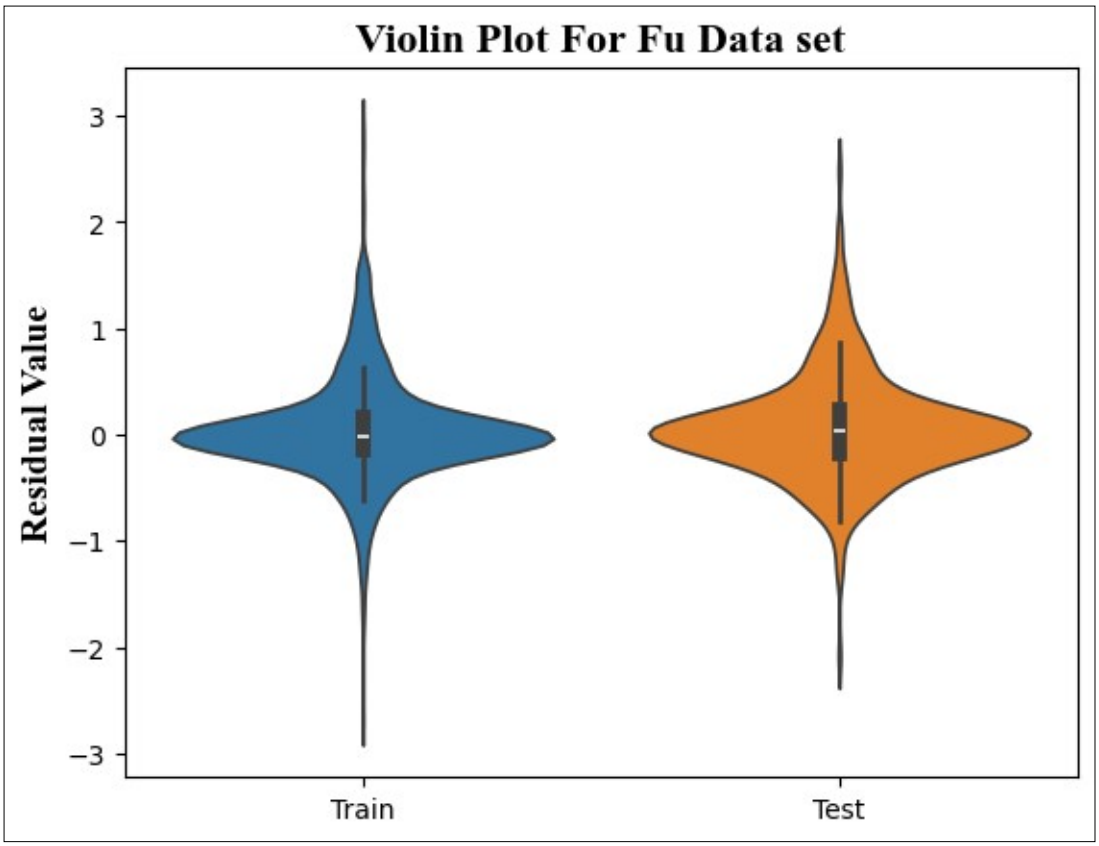

Figure S6. Violin plot for the SVM model (fraction unbound data)

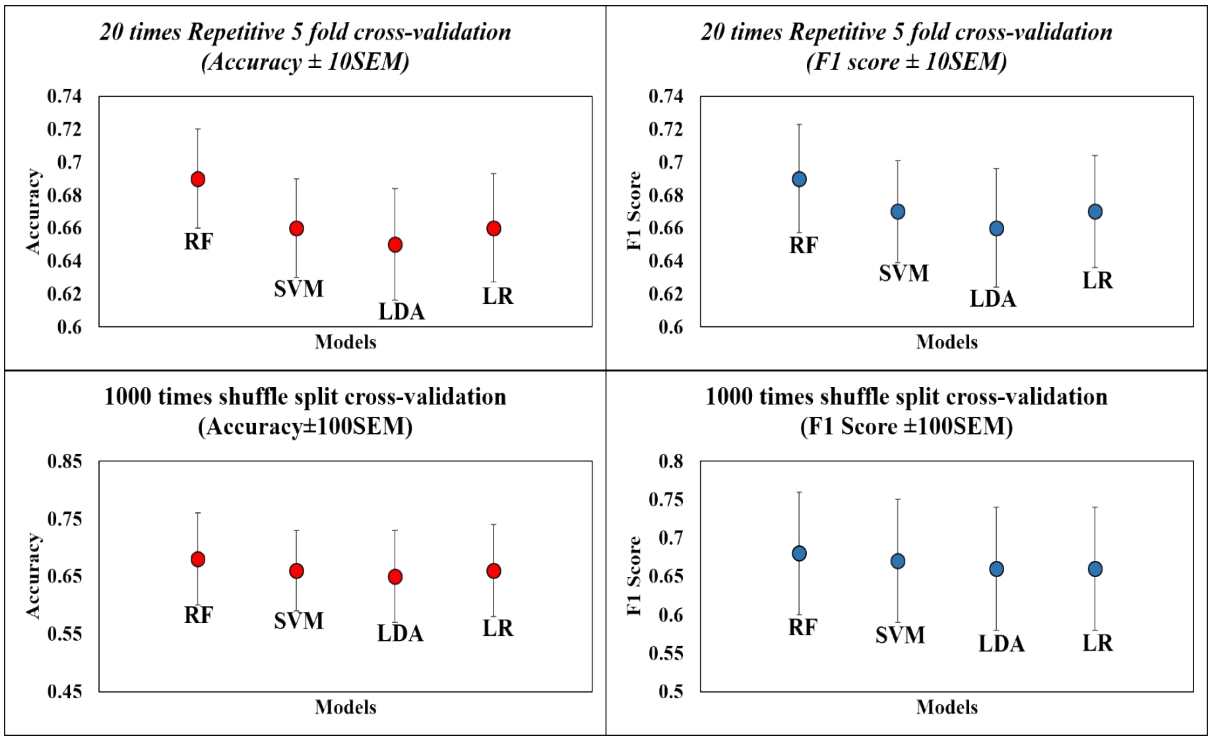

Figure S7. 20 times repetitive and 1000 times shuffle split cross-validation analysis by considering accuracy and F1 score as the objective functions (hepatocyte intrinsic clearance data).

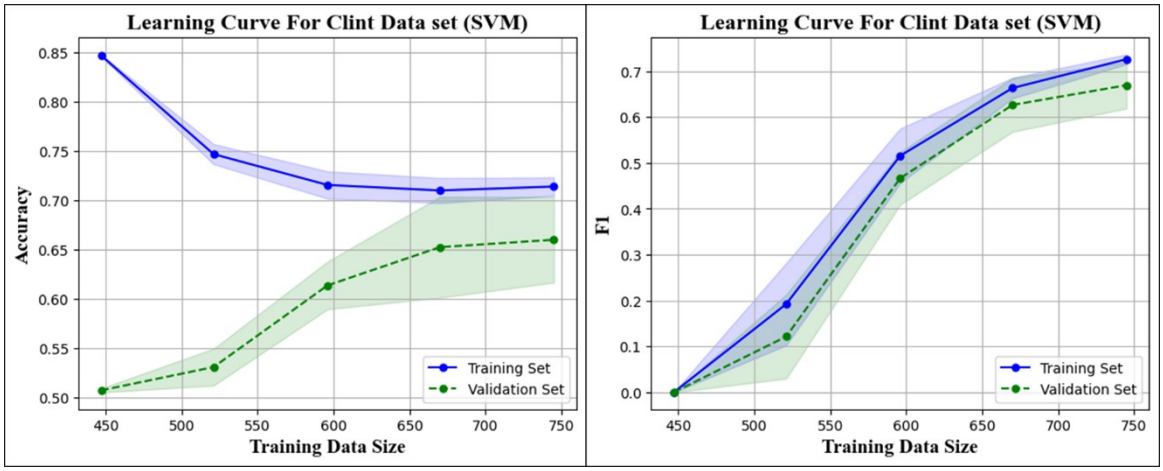

Figure S8. Learning curves for the SVM method by considering accuracy and f1 score as objective functions (hepatocyte intrinsic clearance data).

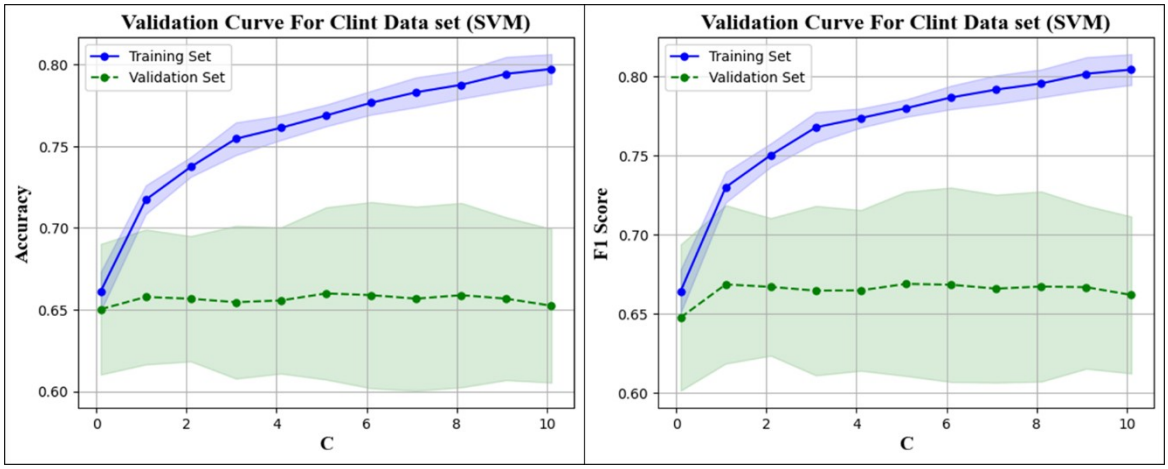

Figure S9. Validation curves for the SVM model at different values of parameter C (Hepatocyte intrinsic clearance data).

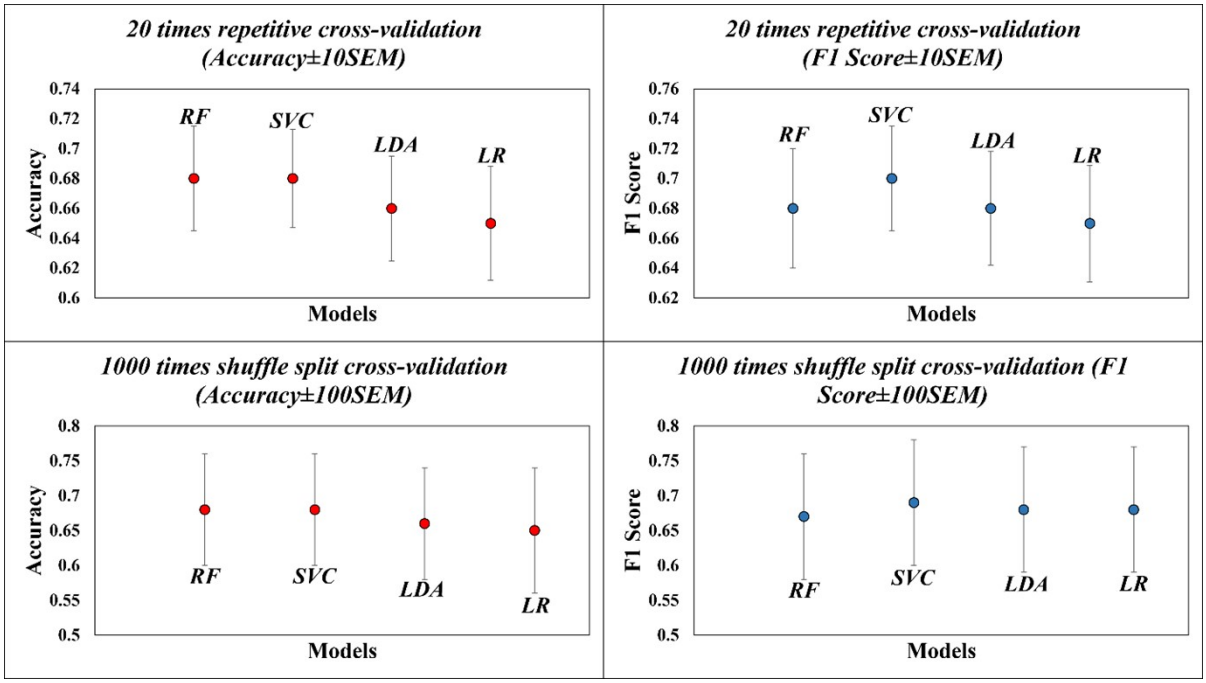

Figure S10. 20 times repetitive and 1000 times shuffle split cross-validation analysis by considering accuracy and F1 score as an objective function (for modeling Cl_{int} by taking f_u as a descriptor).

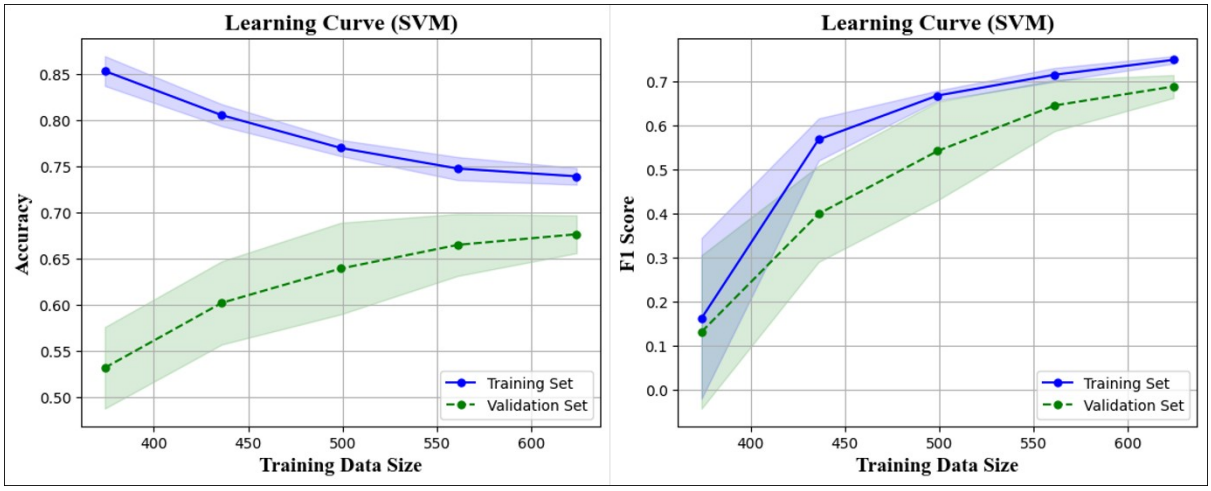

Figure S11. Learning curves for the SVM model (for modeling Cl_{int} by taking f_u as a descriptor).

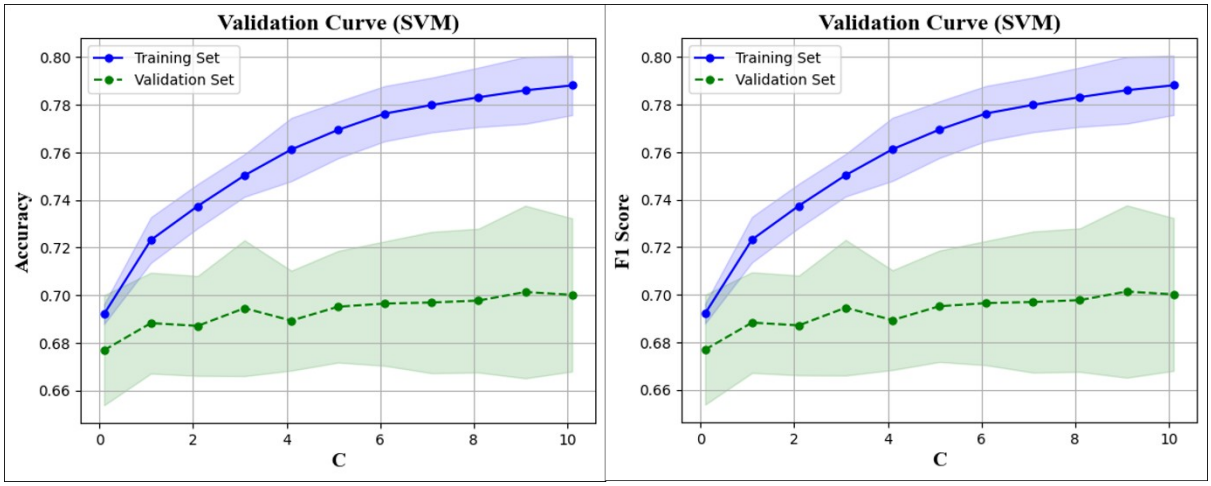

Figure S12. Validation curve for the SVM model (for modeling Cl_{int} by taking f_u as a descriptor).

Brief Introduction about Clearance

Clearance can be divided into three groups: renal clearance (Cl_R), hepatic clearance (Cl_H), and extrahepatic clearance (Cl_{EH}), depending on the organ involved in the elimination process. Hepatic clearance can be further divided into two groups - hepatic metabolic clearance (Cl_{HM}) and biliary clearance (Cl_B). The total body clearance (TBC) or systemic clearance (Cl_s) is the sum of all individual clearances involved in the overall elimination of a chemical compound.[1] It can be calculated using the following equation:

$$Cl_s = Cl_R + Cl_{HM} + Cl_B + Cl_{EH} \quad \text{(S1)}$$

The clearance of an organ can be defined as the ratio of the elimination rate of a substance to its concentration before passing through that organ,[2] represented by the equation:

$$Cl_{organ} = \frac{Q(C_A - C_V)}{C_A} = Q \times E \quad \text{(S2)}$$

where Q is the amount of blood flow, C_A and C_B concentration of the compound before and after entering the organ respectively, and E is the extraction ratio of the organ. An essential parameter to consider is intrinsic clearance (Cl_{int}), which can be defined as the hepatic clearance of a chemical substance without any restriction posed by the blood flow. The intrinsic clearance provides an accurate interpretation of the liver's metabolizing capability.[3,4] The mathematical relationship between Cl_{HM} and Cl_{int} is represented below:

$$Cl_{HM} = (Q_H)\frac{Cl_{int}}{Cl_{int} + Q_H} = (Q_H)\frac{f_u Cl'_{int}}{f_u Cl'_{int} + Q_H} = Q_H \times E_{int} \quad \text{(S3)}$$

here, Q_H is the hepatic blood flow and Cl'_{int} ($=Cl_{int}/f_u$) is the intrinsic unbound clearance. From the above equations, we can find out that the clearance parameter is also influenced by the protein binding. These two parameters also can directly affect the elimination half-life of a chemical within the body,[5] which can be mathematically represented as follows:

$$t_{1/2} = \frac{o.693(V)}{Cl_R + (Q_H)\dfrac{f_u Cl'_{int}}{Q_H + f_u Cl'_{int}}} \quad \text{(S4)}$$

where V ($=7 + 8f_u + V_T\frac{f_u}{f_{ut}}$, V_T=tissue volume, f_{ut}=tissue fraction unbound) indicates the apparent volume of distribution.

Feature Selection through the GA method

The main principle behind the genetic algorithm (GA) is adopted from the biological evaluation[6] that in a certain environmental condition species with the highest fitness score will be selected for the subsequent generation. The GA method of feature selection is first used by Roger and Hopfinger[7] in the development of the QSAR modeling. The GA method consists of 4 basic steps that are:

a) Initialization - The whole descriptor matrix represents the initial population of size 'n' descriptors (chromosome). The descriptors are assigned to the bit string 1 (descriptor is selected) or 0 (descriptor is not selected).
b) Selection – In this stage descriptors are randomly selected for mating and given their first generation. The next descriptor combination for mating is selected based on the fitness score of the descriptors. In this process, a part of the population is transferred from the previous generation to the next generation.
c) Genetic operation – This step helps in the modification of the descriptor combination by three processes – reproduction (making a copy of a chromosome and transferring to the next generation), crossover (two chromosomes selected for mating and share character with each other) and mutation (alteration of the part of the chromosome).

d) Termination – This process is terminated until a certain level of fitness score is reached.

Machine Learning Models

The RF method is a decision tree-based ensemble algorithm in which multiple decision trees are developed to make predictions. The RF algorithm works based on the bagging (bootstrap aggregation) mechanism where multiple bootstrap data sets are generated and with these data sets multiple decision trees are generated from which the final prediction is taken either by averaging or voting based on the target data type.[8] The ADB and GB also work based on a decision tree-based ensemble algorithm but the main difference is that these algorithms work by boosting mechanism where decision trees are generated sequentially.[9] The XGB algorithm works similarly to the GB algorithm but the only difference is that it takes into consideration the feature distribution across all the data points in a single leaf node. The XGB method generates multiple tree branches parallelly, so it requires less time compared with the GB method when a large number of features is present.[10] The SVM and LSVM are classification ML algorithms but these are also applicable in the regression problems. The main principle behind this is to draw a decision boundary between the observations to make predictions. In the LSVM algorithm, the data domain is mapped into the response domain to make decisions but in SVM the data is first transformed in the feature space (using the kernel function) before mapping with the response.[11] The RR algorithm is an important regularization technique that helps to reduce the multicollinearity problem of the multiple linear regression (MLR) models without removing the independent variable.[9] The PLS is a generalized version of the MLR model which is mainly applied to the collinear, correlated, and noisy data sets. The main principle behind the PLS method is to derive latent variables (LVs) or X-scores and the Y-score from the descriptor and response data respectively, and these are finally used to train the model and make predictions of the response.[12] LDA is a dimensionality reduction algorithm that aims to find linear combinations of features that best separate two or more classes. It tries to maximize the distance between the means of the different classes while minimizing the variance within each class. It computes the eigenvalues and eigenvectors of the covariance matrix to find the linear discriminants.[13] LR is a machine learning algorithm used for binary and multiclass classification. It models the probability of the instance belonging to the particular class by applying a sigmoid function to the linear combinations of the feature. It uses cross-entropy as a cost function and tries to minimize the difference between the predicted probabilities and the actual class.[14]

Statistical quality and validation metrics

Regression-based metrics:

$$R^2 = 1 - \frac{\sum (Y_{obs(training)} - Y_{cal(training)})^2}{\sum (Y_{obs(training)} - \bar{Y}_{training})^2} \quad \text{(S5)}$$

$$MAE_{training} = \frac{\sum |Y_{obs(training)} - Y_{cal(training)}|}{n_{training}} \quad \text{(S6)}$$

$$RMSEC = \sqrt{\frac{\sum (Y_{obs(training)} - Y_{cal(training)})^2}{n_{training}}} \quad \text{(S7)}$$

$$Q^2_{F1} = 1 - \frac{\sum (Y_{obs(test)} - Y_{cal(test)})^2}{\sum (Y_{obs(test)} - \bar{Y}_{training})^2} \quad \text{(S8)}$$

$$Q^2_{F2} = 1 - \frac{\sum (Y_{obs(test)} - Y_{cal(test)})^2}{\sum (Y_{obs(test)} - \bar{Y}_{test})^2} \quad \text{(S9)}$$

$$Q^2_{F3} = 1 - \frac{[\sum (Y_{obs(test)} - Y_{cal(test)})^2]/n_{test}}{\left[\sum (Y_{obs(training)} - \bar{Y}_{training})^2\right]/n_{training}} \quad \text{(S10)}$$

$$MAE_{test} = \frac{\sum |Y_{obs(test)} - Y_{cal(test)}|}{n_{test}} \quad \text{(S11)}$$

$$RMSEP = \sqrt{\frac{\sum (Y_{obs(test)} - Y_{cal(test)})^2}{n_{test}}} \quad \text{(S12)}$$

$$Q^2_{LOO} = 1 - \frac{\sum (Y_{obs(training)} - Y_{loo\ predicted(training)})^2}{\sum (Y_{obs(training)} - \bar{Y}_{training})^2} \quad \text{(S13)}$$

$$MAE_{LOO} = \frac{\sum |Y_{obs(training)} - Y_{loo\ predicted(training)}|}{n_{training}} \quad \text{(S14)}$$

$$MSE_{LOO} = \frac{\sum (Y_{obs(training)} - Y_{loo\ predicted(training)})^2}{n_{training}} \quad \text{(S15)}$$

Here, $Y_{obs(trainining)}$ = observed response value of the training set, $Y_{cal(training)}$ = calculated response value of the training set, $Y_{training}$ = mean observed response value of the training set, $Y_{loo\ predicted(training)}$ = leave-one-out predicted response value of the training set, $n_{training}$ = the number of observation in the training set, $Y_{obs(test)}$ = observed response value of the test set, $Y_{cal(test)}$ = calculated response value of the test set, Y_{test} = mean observed response value of the test set, n_{test} = number of observation in the test set.

Classification-based metrics:

$$Sensitivity = \frac{TP}{TP + FN} \quad \text{(S16)}$$

$$Specificity = \frac{TN}{TN + FP} \quad \text{(S17)}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{(S18)}$$

$$Precision = \frac{TP}{TP + FP} \quad \text{(S19)}$$

$$F1\ score = \frac{2Sensitivity\ \times Precision}{Sensitivity + Precision} \quad \text{(S20)}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad \text{(S21)}$$

$$Cohen\ \kappa = \frac{P_r(a) - P_r(e)}{1 - P_r(e)} \quad \text{(S22)}$$

$$P_r(a) = \frac{TP + TN}{TP + FP + FN + TN}$$

$$P_r(e) = \frac{\{(TP + FP) \times (TP + FN)\} + \{(TN + FP) \times (TN + FN)\}}{(TP + FP + FN + TN)^2}$$

Here, TP = positive compounds that have been correctly classified as positives by the model

FP = negative compounds that have been incorrectly classified as positives by the model

TN = negative compounds that have been correctly classified as negatives by the model

FN = positive compounds that have been incorrectly classified as negatives by the model

Table S1. Read-across validation metrics for fraction unbound data.

Metrics	Euclidean Distance	Gaussian Kernel	Laplacian Kernel
Q^2_{F1}	0.66	0.67	**0.70**
Q^2_{F2}	0.66	0.67	**0.70**
RMSEP	0.06	0.06	**0.05**
MAE	0.04	0.04	**0.04**

Table S2. Results of the prediction for the external set (fraction unbound data).

Models	Accuracy	Precision	Recall	Specificity	F_1 Score	MCC	Cohen kappa
RF	**0.65**	**0.63**	**0.56**	**0.72**	**0.59**	**0.29**	**0.29**
ADB	**0.60**	**0.54**	**0.73**	**0.48**	**0.62**	**0.22**	**0.21**
GB	**0.64**	**0.62**	**0.56**	**0.72**	**0.59**	**0.28**	**0.28**
XGB	**0.65**	**0.62**	**0.59**	**0.70**	**0.61**	**0.29**	**0.29**
SVM	**0.64**	**0.61**	**0.59**	**0.68**	**0.60**	**0.27**	**0.27**
LSVM	**0.60**	**0.56**	**0.58**	**0.61**	**0.57**	**0.20**	**0.20**
RR	**0.62**	**0.57**	**0.62**	**0.61**	**0.60**	**0.23**	**0.23**
PLS	**0.60**	**0.56**	**0.61**	**0.60**	**0.58**	**0.21**	**0.21**

Table S3. Read-across prediction of the Cl_{int} data set.

Method	AUC-ROC	Sensitivity	Specificity	Accuracy	Precision	F1-Score	MCC	Cohen's k
ED	0.70	0.95	0.24	0.75	0.76	0.84	0.27	0.23
GK	0.70	0.91	0.3	0.75	0.77	0.84	0.30	0.27
LK	0.70	0.91	0.37	0.75	0.79	0.84	0.33	0.31

Table S4. Read-across prediction for the common compounds (for modeling Cl_{int} by taking f_u as a descriptor).

Method	AUC-ROC	Sensitivity	Specificity	Accuracy	Precision	F1-Score	MCC	Cohen's k

ED	0.76	0.96	0.20	0.68	0.70	0.79	0.26	0.19
GK	0.76	0.96	0.23	0.69	0.68	0.79	0.28	0.21
LK	0.75	0.94	0.28	0.69	0.69	0.79	0.30	0.25

REFERENCES

1. Lucas, A.J.; Sproston, J.L.; Barton, P.; Riley, R.J. Estimating human ADME properties, pharmacokinetic parameters, and likely clinical dose in drug discovery. *Expert Opin. Drug Discov.* **2019,** 14, 1313-1327.
2. Toutain, P.L.; Bousquet-mélou, A. Plasma clearance. *J. Vet. Pharmacol. Ther.* **2004**, 27, 415-425.
3. Roberts, J.A.; Pea, F.; Lipman, J. The clinical relevance of plasma protein binding changes. *Clin. pharmacokinet.* **2013**, 52, 1-8.
4. Martinez, M.N. Article II: Volume, clearance, and half-life. *J. Am. Vet. Med. Assoc.* **1998,** 213, 1122-1127.
5. Toutain, P.L.; Bousquet-mélou, A. Plasma terminal half-life. *J. Vet. Pharmacol. Ther.* **2004,** 27, 427-439.
6. Oh, I.S.; Lee, J.S.; Moon, B.R. Hybrid genetic algorithms for feature selection. IEEE Trans. Pattern. Anal. Mach. Intell. 2004, 26, 1424-1437.
7. Rogers, D.; Hopfinger, A.J. Application of genetic function approximation to quantitative structure-activity relationships and quantitative structure-property relationships. J. Chem. Inf. Comput. 1994, 34, 854-866.
8. Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J.C.; Sheridan, R.P.; Feuston, B.P. Random forest: a classification and regression tool for compound classification and QSAR modeling. J. Chem. Inf. Comput. 2003, 43, 1947-1958.
9. Géron, A. Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, Inc., USA, 2022.
10. 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. 2016, 785-794.
11. Suthaharan, S.; Suthaharan, S. Support vector machine. Machine learning models and algorithms for big data classification: thinking with examples for effective learning. Boston, MA, 2016, 207-235.

12. Roy, K.; Kar, S.; Das, R.N. Understanding the Basics of QSAR for Applications in Pharmaceutical Sciences and Risk Assessment, Academic Press, NY, 2015.
13. Xanthopoulos, P.; Pardalos, P.M.; Trafalis, T.B.; Xanthopoulos, P.; Pardalos, P.M.; Trafalis, T.B. Linear discriminant analysis. Robust data mining. New York, NY, 2013, 27-33.
14. Nick, T.G.; Campbell, K.M. Logistic regression. Topics in biostatistics. Humana Press, 2007, 273-301.