

IN-SILICO PREDICTION OF K_{el} AND K_{abs} RATE CONSTANTS USING RANDOM FOREST APPROACH

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Science degree in Bioinformatics and Data Science

by

ANGARA SREE THRIVENI SAMHITHA (40738001)



**DEPARTMENT OF BIOINFORMATICS
SCHOOL OF BIOCHEMICAL AND ENGINEERING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

MAY - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC | 12B Status by UGC | Approved by AICTE

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF BIOINFORMATICS

BONAFIDE CERTIFICATE

This is to certify that the Project Report submitted entitled '**INSILICO PREDICTION OF K_{el} AND K_{abs} RATE CONSTANTS USING RANDOM FOREST APPROACH**' is a bonafide work done by **ANGARA SREE THRIVENI SAMHITHA (40738001)** under my supervision at Theevanam Additives & Nutraceuts Pvt Ltd, IITM Bioincubator, BT 218, Department of Biotechnology, IIT Madras, Chennai- 600036 from December 2022 to April 2023.

Internal Guide
DR. SWETHA SUNKAR

External Guide (if Applicable)
Dr. MUKESH DOBLE

Head of the Department
DR. JEMMYCHRISTY, MTECH., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner



M/s Theevanam Additives & Nutraceuticals Pvt Ltd
Registered Office: IITM Bioincubator, BT 218,
Department of Biotechnology, IIT Madras, Chennai- 600036, India
Tel: 044-22574103; Mob: +919884609107
Email: theevanamiit@gmail.com; Web: www.theevanam.com;

Dr. Mukesh Doble
Director, Theevanam Additives & Nutraceuticals p ltd
Professor (Retd), IIT Madras, Chennai-600036, India
mukeshd@iitm.ac.in; mukesh.doble0@gmail.com

23/04/2023

TO WHOMSOEVER IT MAY CONCERN

This is to inform **Ms. Angara Sree Thriveni Samhitha**, a 3rd year B.Sc. Bioinformatics and Data Science student from Sathyabama Institute of Science and Technology (Deemed to be University), that she has done her project at Theevanam additives and nutraceuticals p ltd, under my supervision. The project entitled '**INSILICO PREDICTION OF K_{el} AND K_{abs} RATE CONSTANTS USING RANDOM FOREST APPROACH**' is a work done by her from December 2022 to April 2023.

Sincerely yours

Director / Authorised Signatory

Mukesh Doble



DECLARATION

I **ANGARA SREE THRIVENI SAMHITHA (40738001)** hereby declare that the Project Report entitled **“INSILICO PREDICTION OF K_{el} AND K_{abs} RATE CONSTANTS BY RANDOM FOREST APPROACH”** done by me under the guidance of Dr. SWETHA SUNKAR and Dr. MUKESH DOBLE (External) at Theevanam Additives & Nutraceuticals Pvt Ltd, IITM Bioincubator, BT 218, Department of Biotechnology, IIT Madras, Chennai- 600036 is submitted in partial fulfillment of the requirements for the award of Bachelor of Science degree in BIOINFORMATICS AND DATA SCIENCE

DATE: 9-05-23

PLACE:Chennai

Sambhita A

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. REKHA CHAKRAVARTHI M.E.,Ph.D., Dean, School of BIO AND CHEMICAL ENGINEERING and Dr. JEMMY CHRISTY M.TECH.,Ph.D., Head of the Department of Bioinformatics for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr. SWETHA SUNKAR (Internal) and Dr. MUKESH DOBLE (External) for their valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Bioinformatics who were helpful in many ways for the completion of the project.

ABSTRACT

Prediction of human pharmacokinetic parameters using various machine learning algorithms has been widely used to predict dosage regimens and toxicity of various drugs way ahead of the drug development phase. Previous investigations have implemented many computational machine-learning approaches that utilize physiochemical properties calculated from the atomic structures of drugs that enabled the virtual predictions of the desired pharmacokinetic parameter. In this study, the k_{el} and k_a rate constants that play a major role in the absorption and elimination of a drug were predicted using 2D and 3D dimensional descriptors calculated from the most commonly used software platforms and packages. Random forest regression, a well-understood machine algorithm was used to design the models. These models were further evaluated using evaluation metrics to address the predictive performance and the generalization ability. However, the models require refinement in predictivity and further analysis with a different set of descriptors to better understand the biological phenomenon from a statistical point of view.

TABLE OF CONTENTS

CHAP NO	TITLE	PAGE NO
1	INTRODUCTION	1
	1.1 Background	1
	1.2 Essential Elements of Pharmacokinetics	1
	1.2.1 Bioavailability	1
	1.2.2 Volume of Distribution	2
	1.2.3 Clearance	2
	1.1.4 Elimination Half-Life	2
	1.3 In Vivo And In Vitro Pharmacokinetics	3
	1.4 Introduction to In Silico Modelling	3
2	LITERATURE SURVERY	4
	2.1 An Era of Machine Learning in Pharmacokinetics	4
	2.1.1 Volume of Distribution	4
	2.1.2 Oral Bioavailability	4
	2.1.3 Clearance	5
	2.1.4 Half-Life	6
	2.2 DISCUSSION AND FUTURE RESEARCH	7
3	AIM AND OBJECTIVE	8
	3.1 Objectives	8
4	MATERIALS AND METHODS	9
	4.1 Data Collection and Preparation	9
	4.1.1 For k Elimination Rate Constant	9
	4.1.2 For k Absorption Rate Constant	10
	4.2 Feature Selection	10
	4.2.1 Low Variance Filter	11
	4.2.2 Filter Based Method	11
	4.2.3 Embedded Method	11
	4.2.4 Principal Component Analysis (PCA)	12
	4.3 Model Building	12
	4.4 Model Evaluation	13
5	RESULTS AND DISCUSSION	14

	5.1 Discussion on Feature Selection	28
	5.2 Performance Analysis	28
6	CONCLUSION	30
7	REFERENCES	31
8	APPENDIX	34
	(a) Pearson correlation	34
	(b) Dropping highly correlated featured from the training set	36
	(c) Removing multicollinearity using VIF	36
	(d) Only random forest approach	37
	(e) PCA and random forest	38
	(f) Generating the correlation matrix and circle graph	41
	(g) Generating a tree from the forest	41

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
5.1	A heatmap representing the correlation between k elimination rate constant and the drug descriptors	14
5.2	A heatmap representing the correlation between the molecular descriptors after applying the high correlation filter.	15
5.3	Feature selection using random forest importance for k-elimination rate constant.	16
5.4	Cumulative explained variance for k elimination rate constant	17
5.5	Visual representation of the training dataset (k-elimination rate constant) after dimensionality reduction	17
5.6	A circle graph representing the correlation between Mordred descriptors and the principal components	18
5.7	A snippet of the correlation matrix representing the Pearson correlation coefficients between Mordred descriptors and the principal components.	18
5.8	A tree from the forest (from the k elimination rate constant prediction model)	19
5.9	A heatmap representing the multicollinearity between PaDEL descriptors (training data)	20
5.10	Feature selection using random forest importance for k absorption rate constant.	21
5.11	Cumulative explained variance for k absorption rate constant	21
5.12	Visual representation of the training dataset (k absorption rate constant) after dimensionality reduction	22
5.13	A tree from the forest (from the k absorption rate constant prediction model)	23

LIST OF TABLES

TABLE NO	TITLE	PAGE NO
5.1	Pearson correlation between k elimination rate constant and Mordred Descriptors	24
5.2	VIF results for multicollinearity	26

LIST OF ABBREVIATIONS

1. k_{el} = k elimination rate constant
2. k_a = k absorption rate constant
3. PCA - Principal Component Analysis

INTRODUCTION:

1.1 BACKGROUND

Pharmacokinetics has been the heart and soul of drug design and development. The body recognizes and treats each medicine differently depending on its distinct features from the minute it enters the body, and therefore pharmacokinetics is considered to be one of the main fields of pharmacology that focuses on how the body responds when it is exposed to drugs. (Grogan S & Preuss CV, 2022)

The four processes namely liberation, absorption, distribution, metabolism, and excretion (LADME) make up the majority of pharmacokinetics. These are used to explain the diverse bodily effects due to various prescription drugs. By having a strong understanding of these processes, clinicians can more easily prescribe and administer pills that will have the most beneficial effect at a minimal risk given the differing physiology and lifestyles of patients. They may additionally make adjustments as needed.

1.2 ESSENTIAL ELEMENTS OF PHARMACOKINETICS

1.2.1 Bioavailability

The percentage of the administered dose of a drug that enters the bloodstreams known as the drug's bioavailability. Once the drug is administered the pace at which it is absorbed is not taken into account by bioavailability. However, the elements that affect absorption also have an impact on bioavailability. By comparing the associated regions under the plasma concentration curves (AUC) following oral and intravenous delivery, it is possible to establish a drug's absolute availability. (Sylvain Chemtob, 2011)

The absolute bioavailability of a drug can be calculated as:

$$\text{Absolute Bioavailability} = \frac{\text{AUC}_{\text{po}} / \text{dose}_{\text{po}}}{\text{AUC}_{\text{iv}} / \text{dose}_{\text{iv}}}$$

1.2.2 Volume of Distribution

Another crucial PK parameter is the apparent volume of distribution (V_d), often known as the volume of distribution. V_d is a projection of the drug's extracellular fluid distribution. (Chatterjee et al., 2021)

The volume of distribution can be calculated as :

$$V_d = \frac{\text{Amount of drug in body}}{\text{Concentration of drug in the blood}}$$

1.2.3 Clearance

The amount of drug cleared from the plasma over a predetermined period is known as drug clearance.

Kidneys and the liver are the primary organs via which the drug present in the plasma can be eliminated from the body. The terms renal clearance and hepatic clearance are processes that account for clearance from the above-mentioned organs. (Horde GW & Gupta V, 2022)

Clearance of a drug can be calculated as:

$$Cl = \frac{\text{rate at which the drug is removed from plasma (mg/min)}}{\text{concentration of the drug in the plasma}}$$

1.2.4 Elimination Half-Life

Elimination Half-life refers to the amount of time needed for a substance's concentration in the body to drop to half of its initial dose (usually a medicine). (Hallare J & Gerriets V, 2022)

The elimination half-life can be calculated as:

$$t_{1/2} = \frac{(0.693 \times V_d)}{Cl}$$

Other elements like c_{max} , protein binding, a fraction of the available dose excreted unchanged, toxic concentrations, rate of availability, etc are also included in pharmacokinetic studies.

1.3 INVIVO AND INVITRO PHARMACOKINETICS

There are several in vitro techniques like metabolic stability tests, reactive metabolite tests, enzyme inhibitor tests, enzyme induction tests, protein bonding tests, membrane permeability tests, transporter testing, hepatic clearance prediction, and chemical property assessments to understand the nature of a drug. Along with the preliminary in-vitro techniques, several in-vivo assessments like the exploratory pharmacokinetic tests and quantitative analysis of endogenous compounds are also adapted to measure the concentration in blood plasma or tissue of various animals like mice, rats, etc using the LC-MSMS method.

Because in vitro studies help cut down expenses, immediately analyze product performance, and provide advantages in terms of ethical considerations in vitro studies may function as the best method when compared with in vivo pharmacokinetic studies.(Polli, 2008) Despite its advantages and importance, in-vitro techniques are still time-consuming processes. To overcome this disadvantage the concept of mathematical and in-silico predictions has been introduced.

1.4 INTRODUCTION TO IN-SILICO MODELLING

In silico models investigate the quantitative connection (QSAR, QSPR) that provides useful information for developing and prescribing new drugs. To understand and predict a drug's pharmacokinetic qualities based on its structure, in-silico evaluation of absorption, distribution, metabolism, and excretion (ADME) properties is necessary and therefore uses effective techniques like mathematical and computer modelling.

The mathematical technique that predicts chemical information embedded within descriptors (chemical properties that determine the observed response) is the foundation of the quantitative structure-activity relationship (QSAR) and quantitative structure-property relationship (QSPR) methodologies. These investigations are extremely valuable and include data preparation, data processing, and data interpretation for a group of chemicals. (Hamidović et al., 2021)

A few advantages of using in-silico pharmacokinetics models would be understanding the parameters like its rate of absorption, toxicity, and availability of novel drug designs in a faster and more efficient manner.

LITERATURE SURVEY

2.1 AN ERA OF MACHINE LEARNING IN PHARMACOKINETICS

2.1.1 Volume of Distribution

In 2012, about 132 acidic drugs were used to create QSAR models by Zhivkova and Doytchinova. Models used a stepwise linear regression and GA to develop regression models with about 178 descriptors (four sets of 2D-QSAR and two sets of 3D-QSAR descriptors) to predict the volume of distribution. The predicted values for acidic drugs were shown to be strongly affected by the presence of sulfur and sulfuric groups. Cross-validation was performed with external test sets from the literature resulting in 80% accuracy with a three-fold error. (Zhivkova & Doytchinova, 2012)

Alexa A Freitas *et al.* predicted the volume of distribution at steady state using regression-based decision trees with a total of 604 compounds. Furthermore, they also proved that the use of predicted Kt:p (tissue:plasma partition coefficient) values as descriptors may be advantageous for the accurate prediction of the volume of distribution using decision trees if prior feature selection is applied. This was demonstrated by comparing models that only used molecular descriptors, in particular, the Bagging decision tree (mean fold error of 2.33), with those employing predicted Kt:p values in addition to the molecular descriptors. Some of the most relevant descriptors involved in the prediction of the volume of distribution include Kt:p, Lop P, TPSA, PEOE_VSA +0, Vsurf_ID6, and C_ratio. (Freitas et al., 2015)

2.1.2 Oral Bioavailability

Drugs that are orally administered cross several barriers before reaching the target site and this usually tends to reduce the amount of drug available at the target site. Therefore, to know the amount of drug that would be available at the target site is a necessity to prescribe the required initial dosage, prevent human toxicity and industrial failure. Xue Xu *et al.* developed QSAR-based models with about 805 structurally diverse drugs that help in the prediction of oral bioavailability by integrating the information of the ATP-dependent efflux protein P-gp and the cytochrome P4503A4 and P4502D6 metabolizing enzymes. Machine learning algorithms like support-vector machine regression, partial least squares regression,

and multiple linear regression were compared against each other for their predictivity with five-fold cross-validation. In conclusion, the support vector regressor that used Gaussian radial basis function ($R^2=80$, $SEE= 0.31$) was considered a slightly better algorithm based on model evaluation metrics. (Xu et al., 2012)

Along with various regression-based models, researchers have also developed numerous classifier models that help to classify drugs as high oral bioavailable or low oral bioavailable drugs. It is because about 50% of candidate drugs fail in the real drug development process due to poor oral availability. As a result, one of the important elements affecting the success or failure of clinical studies of novel drugs is the level of oral availability. Min Wei *et al.* developed a computational model that accurately predicts human oral bioavailability (HOB) using the random forest approach. A total of five random forest models were constructed and the consensus model of all the predictions has been shown to represent consistency with its excellent prediction accuracies. Prediction models were trained on 1157 molecules and were later tested on test set 1 and test set 2 with 290 and 141 molecules respectively. SsOH, ATS5i, and TopoPSA (NO) were the key descriptors that increased the predictive power of the classification model. (Wei et al., 2022)

2.1.3 Clearance

In 2013, Gombar *et al.* developed QSAR-based models to predict clearance using support vector machine regression and multiple linear regression. Models were trained on 525 molecules with 77 fingerprints and were tested on 56 unique compounds obtained from the literature. A Q^2 of 0.32 was the best average achieved from the 50 cross-validation models that used the support vector machine regression algorithm. In addition, a distinct set of 653 two-atom fragments were created for the multiple linear regression model from the training set of 525 compounds, and their E-state values were employed as structure descriptors. Their final model after refinement was an 89 descriptors model built using 510 molecules. The prediction model showed a good performance with an R^2 of 0.70. However, their models represent limited applicability due to their narrow focus on a restricted subset of drugs. (Gombar & Hall, 2013)

Berellini *et al.* developed a prediction model for clearance using the partial least squares algorithm with a total of 754 compounds after excluding the artesunate, carboplatin, and gadoversetamine because of failure in the fragment generation.

Out of the 754 compounds about 220 were neutral and 534 were ionized compounds (184 anionic, 285 cationic, 65 zwitterionic) at a neutral pH. Based on ten therapeutic/structural analog classes, the partial least squares model was validated using the "ELASTICO" approach (Enhanced Leave Analog-Structural, Therapeutic, Ionisation Class Out). The model was shown to predict 59% of compounds within a 2-fold error and 80% within a 3-fold error, with a geometric mean fold error of 2.1, and therefore demonstrated very good performance with ionized compounds at a neutral pH. (Berellini et al., 2012)

2.1.4 Half-Life

Lu *et al.* developed high-quality prediction models for elimination half-life using multiple machine learning approaches like gradient boosting machine (GBM), support vector regressions (Radial basis function-SVR and Linear-SVR), local lazy regression (LLR), SA, SR, and GP. To improve prediction accuracy, two consensus model-building procedures were investigated. The prediction models were trained on 1105 organic compounds collected from the literature. The gradient boosting machine performed the best of the seven unique models ($R^2 = 0.820$ and RMSE = 0.555 for the test set), whereas Linear-SVR provided the least accurate predictions ($R^2 = 0.738$ and RMSE = 0.672). The scope of the QSAR models built was effectively determined by the application of distance-based applicability domains. However, the prediction performance was not enhanced by the consensus models constructed by integrating the individual models. LogP, LogD, LogS, and MW are some of the relevant descriptors that demonstrated significant correlation with the elimination half-life. (Lu et al., 2016)

In 2019, Wang *et al.* developed machine learning models to predict human intravenous pharmacokinetic parameters (elimination half-life, fraction unbound in plasma, volume of distribution and clearance) using multiple regression algorithms like support vector machine, random forest, gradient boosting machine, and XGboost. A total of 17,392 descriptors and fingerprints were calculated using MOE, PaDEL, and Mordred for about 1253 chemical compounds. However, after standard

data processing and feature selection using the Boruta algorithm only about 162 variables were used to train and test the model. With an R^2 value of 0.832, random forest performed better than the statistical findings of other machine learning algorithms. The significance of descriptors like autocorrelations, physical characteristics, or lipophilicity was highly emphasized in this study. Additionally, it was observed that the partial charge descriptors (PEOE_VSA +3, PEOE_VSA +6, PEOE_VSA +7, PEOE_VSA 0, PEOE_VSA 1, PEOE_VSA 2, PEOE_VSA_NEG, PEOE_VSA_PNEG, PEOE_VSA_POL, Q_VSA_PNEG, Xc 5d, Xch 7d) and the $\log t_{1/2}$ values showed a strong positive association. (Wang et al., 2019)

2.2 DISCUSSION AND FUTURE RESEARCH

Many authors investigated the fairly common pharmacokinetic parameters like oral bioavailability, clearance, half-life, volume of distribution, fraction unbound in plasma, and toxicity. However, it was mentioned earlier that the other crucial parameters which affect the four major biological processes namely absorption, distribution, metabolism, and excretion need equal attention for a better understanding of different drug formulations. Three such parameters are the k release rate constant, the k absorption rate constant (k_{abs}), and the k elimination rate constant (k_{el}) of a drug. All three parameters determine the rate at which a drug is being released, absorbed, and eliminated into/from the body which thereby largely affects the availability, time taken to reach the highest concentration of a drug in the blood, and the toxicity studies if incorporated. For instance, k_{el} can be used to determine when a drug concentration becomes no longer harmful in case of a drug overdose. Knowing k_{el} may also be used to predict the time it will take for the plasma concentration to reach a therapeutic level during the first treatment, letting doctors and patients know when they can anticipate a positive response to a medication. The elimination rate constant is also associated with other pharmacokinetic parameters like the volume of distribution, clearance, and elimination half-life. Therefore, investigators must give importance to in silico estimations of these parameters in pharmacokinetic studies.

AIM AND OBJECTIVE

Prediction of human pharmacokinetic parameters using machine learning has been studied for more than a decade often with preclinical and in-vitro metabolism data in combination with the molecular descriptors to obtain models with almost 80-90% accuracy. However, the association between the k_{el} and k_{abs} rate constants with the atomic structure of a drug was never statistically understood. k_{el} is defined as a value used in pharmacokinetics to describe the rate at which a drug is removed from the body while k_{abs} is defined as a value used in pharmacokinetics to describe the rate at which the drug enters the body. Both these parameters tend to differ for every drug based on several factors and depend on the various properties of a drug. These rate constants are extremely important to understand as they most certainly affect the kinetics of a drug.

Thus, the focus of this project and the overall aim is to develop and evaluate machine learning models that could predict the k_{el} and k_{abs} rate constants for novel drug molecules.

3.1 OBJECTIVES

The specific objectives of this project were:

1. To assess the statistical association between the k_{el} and k_{abs} rate constants with the molecular descriptors of drug compounds.
2. To develop a machine learning model to predict the k_{el} rate constants of chemical compounds using the random forest approach.
3. To develop a prototype model to predict the k_{abs} rate constants of chemical compounds using the random forest approach.
4. To understand the predictive performance of these models using evaluation metrics

MATERIALS AND METHODS

4.1 DATA COLLECTION AND PREPARATION

4.1.1 For k elimination rate constant

The half-life data for about 502 structurally diverse drugs were collected from a well-known database called Drug Bank Online. The k_{el} rate constants of these drugs are then calculated through the proven mathematical relationship between half-life and the k elimination rate constant.

$$k_{el} = \frac{0.693}{t_{1/2}}$$

The obtained k_{el} rate constants were then log-transformed because of two main reasons: first, to minimize the exclusion of leverage points due to the high cost of human data, and second, to avoid prediction of negative values for compounds with very low k_{el} values as they could be more difficult to infer, without log transformation. (Gombar & Hall, 2013)

Only about 80 molecular descriptors that have demonstrated a significant correlation with half-life from the previously published works were collected from Mordred. The collected 2-D descriptors were composed of structural information like physiochemical properties, subdivided surface areas, atom counts and bond counts, adjacency and distance matrix descriptors, pharmacophore features, and partial charges of a drug. The final dataset used in the prediction model was composed of 464 drugs with 80 unique molecular descriptors after data pre-processing by removing duplicates, null values, and other inconsistencies that may hinder the performance of the model.

4.1.2 For k absorption rate constant

The k absorption rate constant data on the 100 most popular drugs in the market was collected from various literature studies to develop a prototype using random forest. In order to obtain a better insight into the structural features, 1877 (2D and 3D) descriptors were calculated using the PaDEL software. However, after thorough data cleaning by removing the features with an increased number of null values the final dataset was made up of 100 drugs with 1446 molecular descriptors. Finally, the obtained k_{abs} values were log-transformed in order to normalize the skewed data before it was used to train the model.

4.2 FEATURE SELECTION

Developing reliable QSAR models requires the use of molecular descriptors, which are quantitative representations of the physical, chemical, or topological properties of small molecules. (Mercader et al., 2008)

However, the descriptor data generated could be enormous containing negligible data concerning the target variable. Therefore, it makes feature selection an important step before the development of a machine learning model.

Feature selection is defined as a process where the raw or previously processed data is reduced into a smaller set of features that best capture the vital properties of the data. To enhance the performance and produce quicker, more affordable models, feature selection prevents a machine-learning model from overfitting. The original semantics of the features and their representation are both retained during feature selection. Additionally, by focusing on a few features, researchers can learn more about the underlying mechanisms that cause the data variance. (Suresh et al., 2022)

In this study, the feature selection was employed by three different methods as explained below.

4.2.1 Low variance filter

Generally, features with zero variance or features demonstrating a small variance that is below the threshold set by the investigator state that they do not present any predictive ability in a model.

For this reason, molecular descriptors which represent a variance less than 0.2 were eliminated from the dataset using the sci-kit learn's VarianceThreshold method.

This successfully removed a handful of features from the ka dataset. However, it was it did not show any impact on the k_{el} dataset as the descriptors chosen were carefully handpicked.

4.2.2 Filter-based method

For any regression-based model, it is important to understand how each independent variable is associated with the target variable. The Pearson correlation coefficient determines how strongly two variables are correlated linearly. The values of the coefficient range between -1 to 1, with -1 denoting a perfect negative linear correlation, 0 denoting no correlation, and 1 denoting a perfect positive linear correlation. (Williams et al., 2020)

Once the correlation was programmed the selected features were further filtered by a high correlation filter. A high correlation between two independent variables indicates the presence of multi-collinearity in the data set which could drastically reduce the performance of the machine learning model. For this reason, the high correlation filter was used to exclude one of the two highly correlated descriptors for which the correlation coefficient was higher the 0.90.

The results of the high correlation filter were also supported by the variance inflation factor.

4.2.3 Embedded method

Embedded methods are feature-selection techniques that have the qualities of both filter and wrapper methods. Embedded methods are usually integrated as a part of the learning algorithm, which in this study would be the random forest approach. A random forest is simply a collection of several decision trees where the results

provided by all the trees in the forest are aggregated into a final result. Each node in a decision tree specifies the conditions under which values in a single feature are to be split, ensuring that similar values of the dependent variable are placed in the same set following the split. The criterion is based on impurity, which is the variance for regression trees. Therefore, it calculates the percentage of each feature that contributes to reducing the weighted impurity when training a tree. The `feature_importances_` function from Scikit-Learn was used to retrieve the importance score for each descriptor in both k_{el} and k_{abs} datasets.

4.2.4 Principal Component Analysis (PCA)

The principal component analysis is a dimensionality reduction technique that is mostly used to visualize high-dimensional data by simplifying the complexity present in the data. In other words, PCA generally tries to project high-dimensional data onto a lower-dimensional surface. This also means that the transformed features also referred to as the principal components are a linear combination of two or more features. This was further validated by calculating the correlation matrix between the molecular descriptors and the obtained principal components using the `mlxtend` package.

4.3 MODEL BUILDING

Random Forest Regression is a supervised learning algorithm that uses an ensemble learning method to solve regression problems. A Random Forest Regressor operates by constructing several decision trees while training the model time and outputs the mean of the classes as the prediction of all the trees. The chosen algorithm is a powerful technique that is highly effective, robust, and performs great with non-linear relationships. It is also one of the popular algorithms in the field of pharmacokinetics for demonstrating good performance in predicting several pharmacokinetic parameters. Therefore, several random forest models were built using the set of features obtained from each feature selection algorithm explored above. Both k_{el} and k_{abs} datasets used 80% of their data points to train all the models and 20% of their data points to test the predictivity of those models. It

was also made certain that there was no overlap of data points between the training and test sets.

4.4 MODEL EVALUATION

There are several evaluation metrics like the mean absolute error (MAE), mean absolute percentage error (MAPE), mean squared error (MSE), etc. However, two statistical measures that gauge the goodness of fit were used in this study to evaluate the performance of the regression models: the coefficient of determination (R^2 , the square of Pearson's correlation coefficient, based on the line of best fit) and the root-mean-square error (RMSE). They are calculated using the following equations:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y})^2}{n}}$$

where y_i , $\hat{y}$, and $\bar{y}$ are the experimental, predicted, and mean value of chemicals, respectively. Finally, the generated machine learning models from the training set were externally validated using a test set to understand their predictive and generalization abilities. (Tedesco et al., 2004)

RESULTS AND DISCUSSION

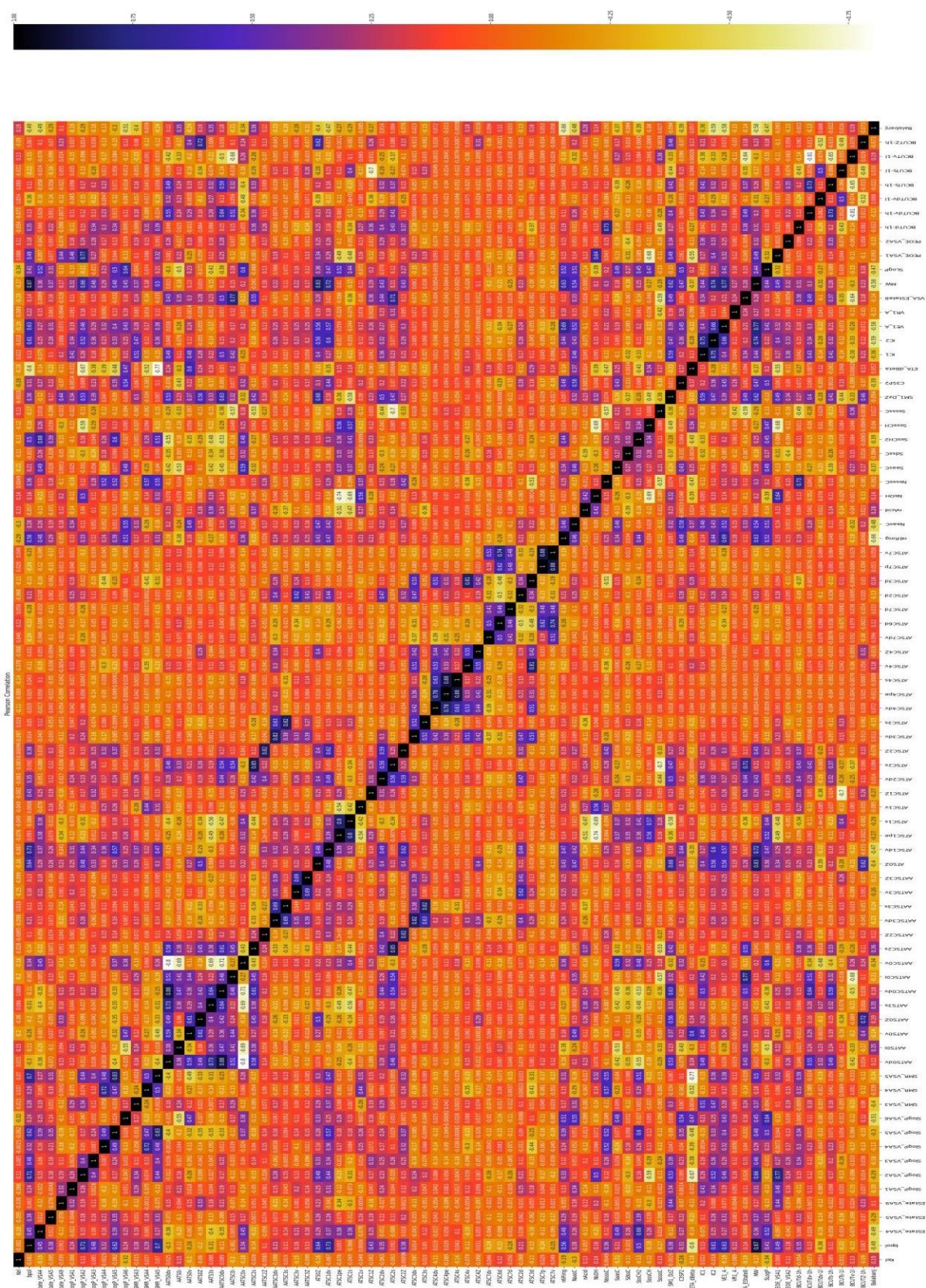


Figure 5.1: A heatmap representing the correlation between k elimination rate constant and the drug descriptors

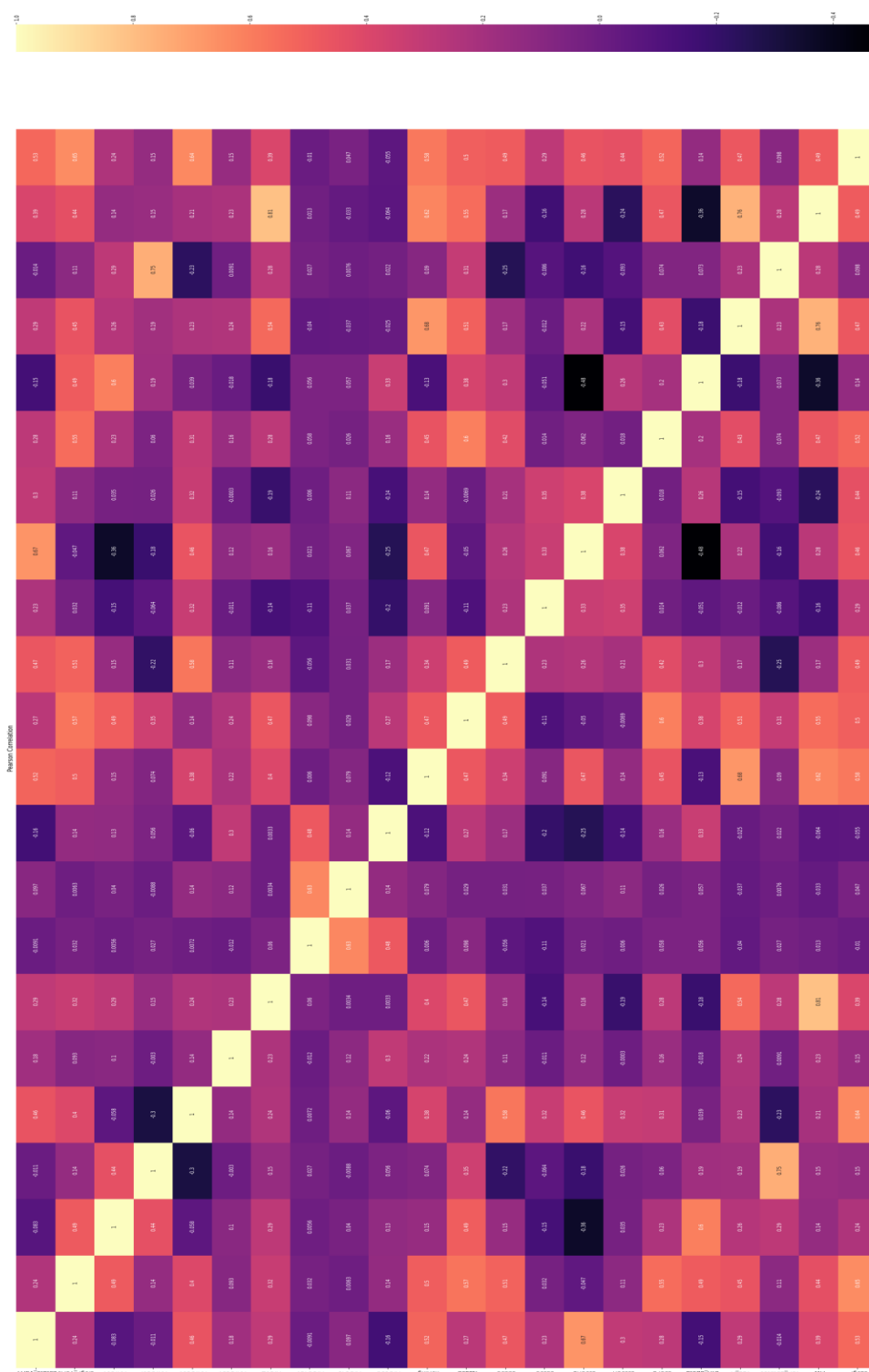


Figure 5.2: A heatmap representing the correlation between the molecular descriptors after applying the high correlation filter.

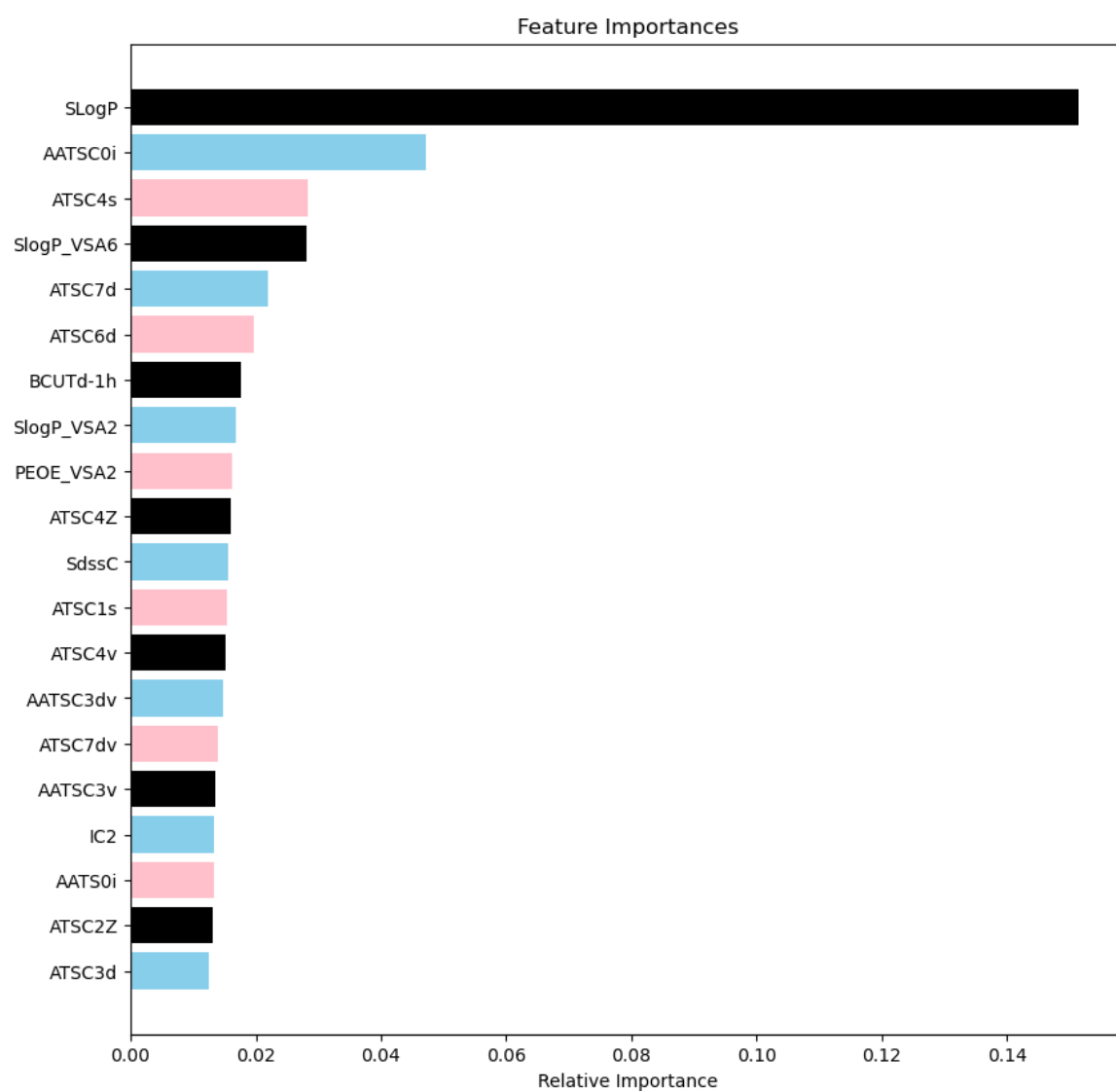


Figure 5.3: Feature selection using random forest importance for k -elimination rate constant.

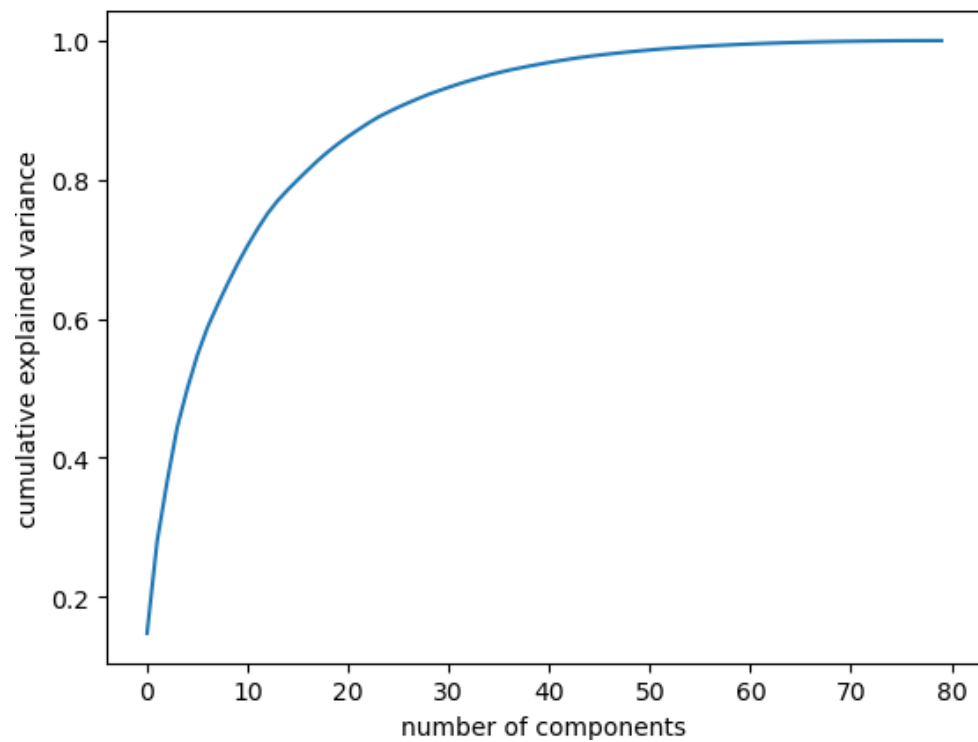


Figure 5.4: Cumulative explained variance for k elimination rate constant

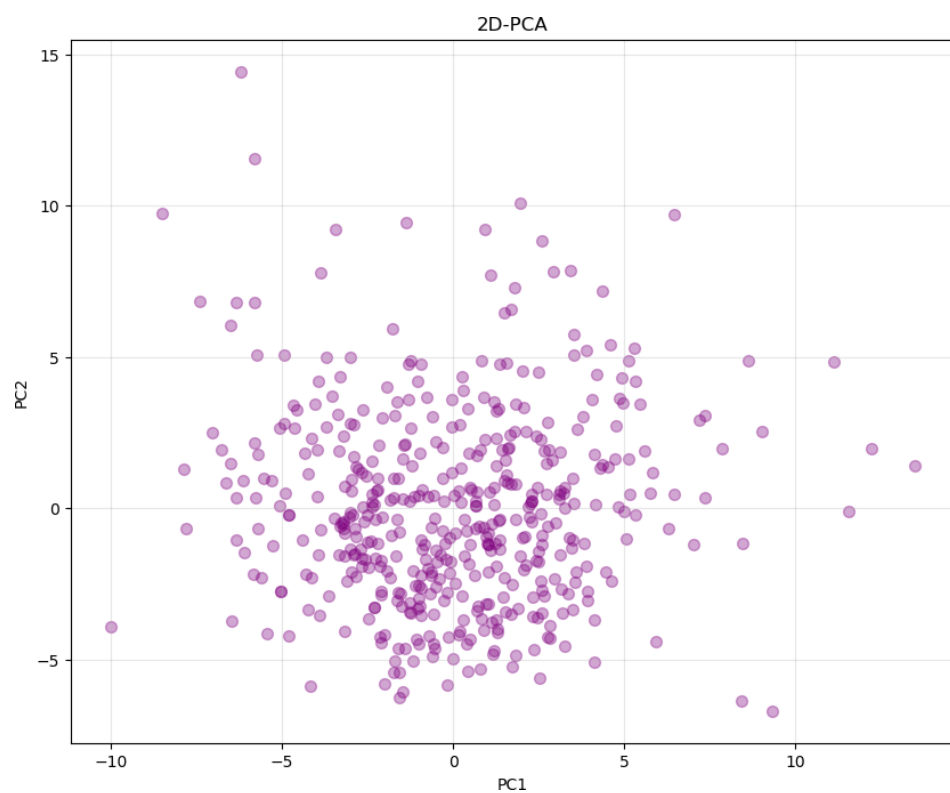


Figure 5.5: Visual representation of the training dataset (k -elimination rate constant) after dimensionality reduction

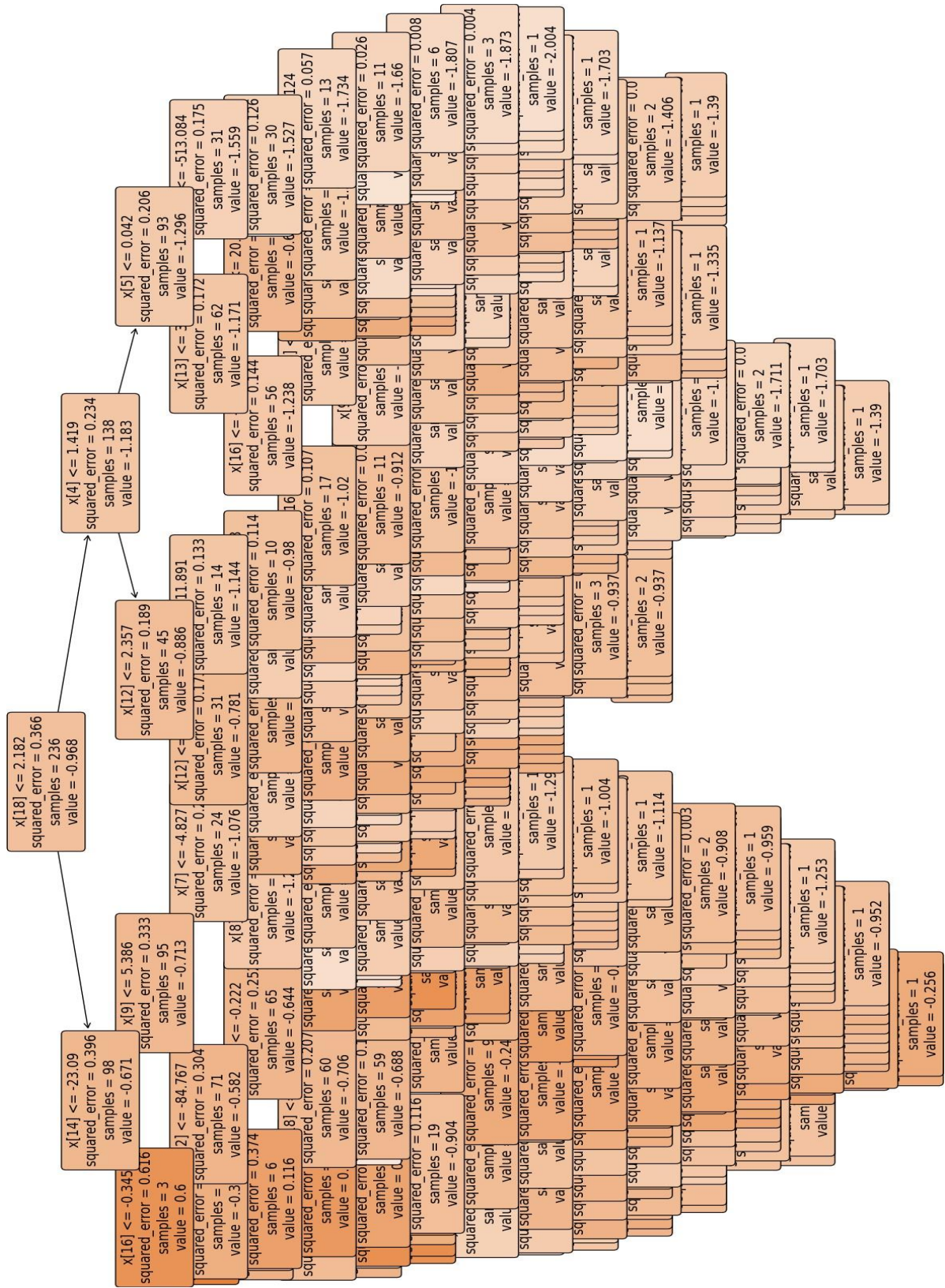
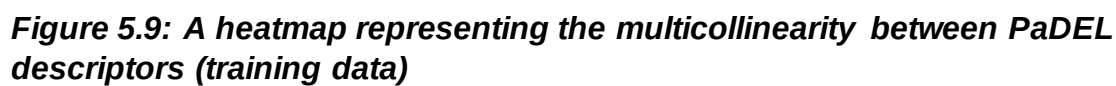


Figure 5.8: A tree from the forest (from the k elimination rate constant prediction model)



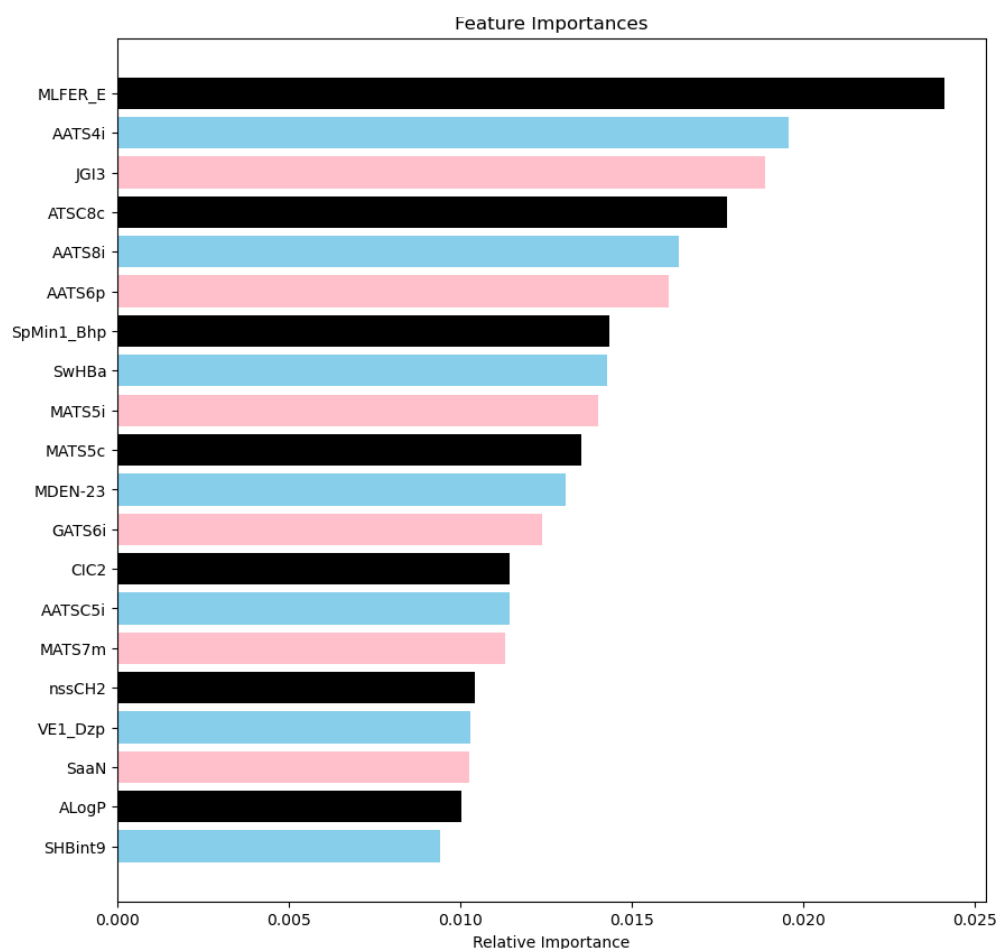


Figure 5.10: Feature selection using random forest importance for k absorption rate constant.

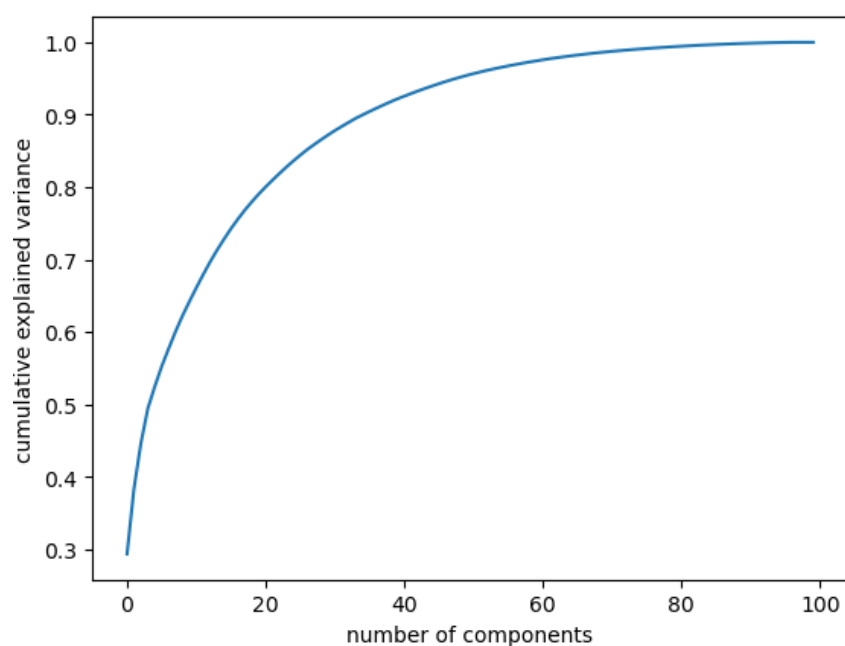


Figure 5.11: Cumulative explained variance for k absorption rate constant

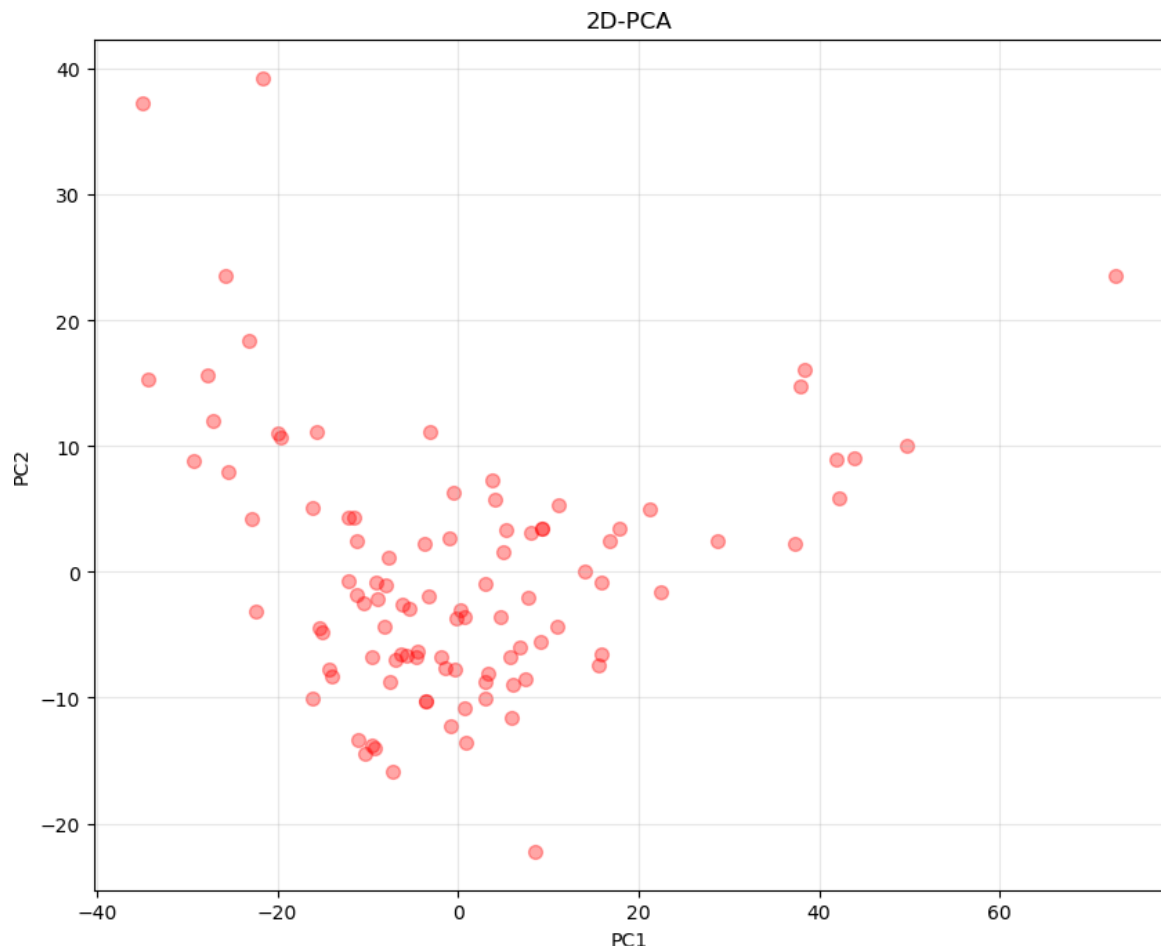


Figure 5.12: Visual representation of the training dataset (k absorption rate constant) after dimensionality reduction

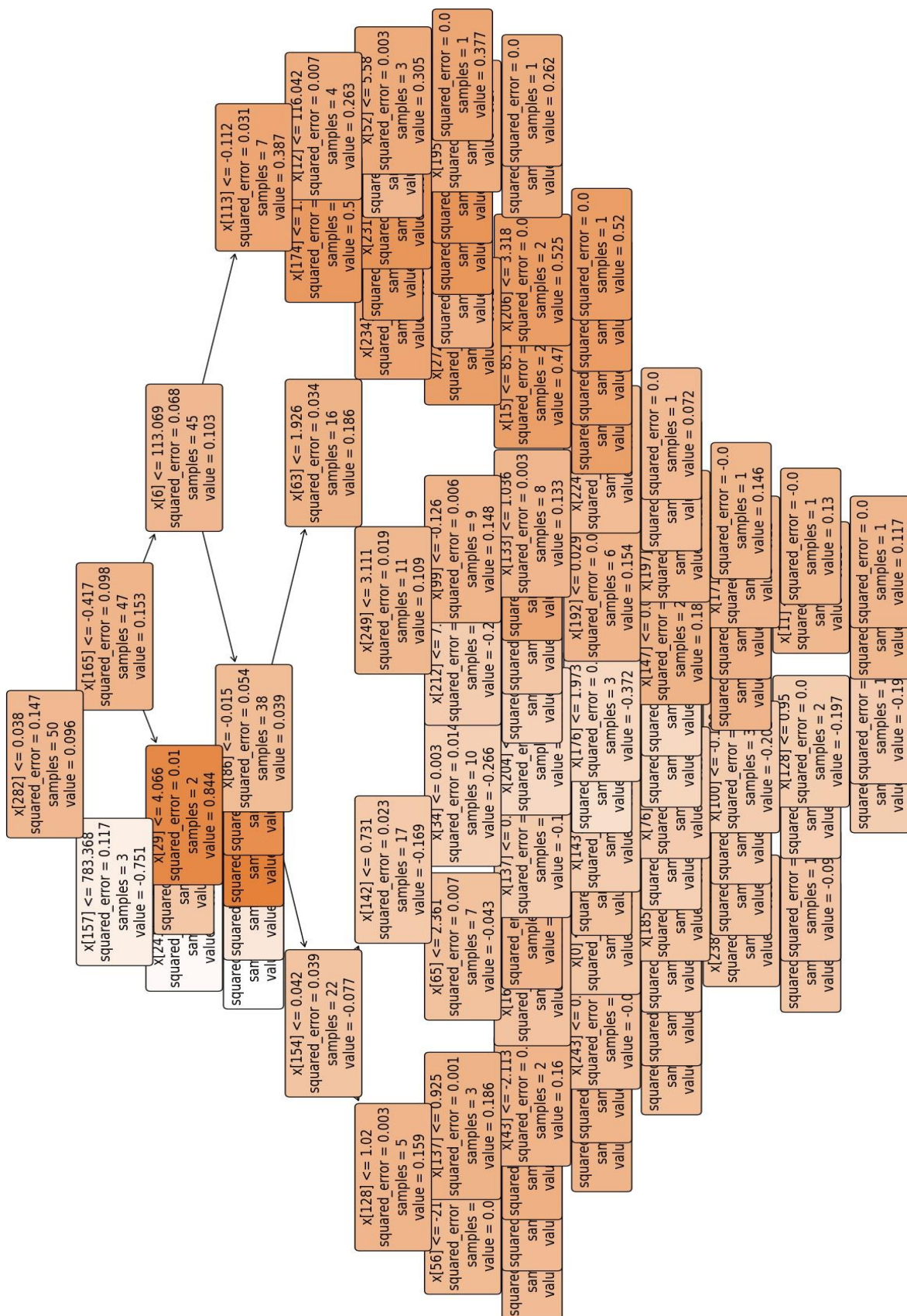


Figure 5.13: A tree from the forest (from the k absorption rate constant prediction model)

Table 5.1 Pearson correlation between k elimination rate constant and Mordred Descriptors

Descriptors	K _{el}
K _{el}	1
bpol	-0.08246
EState_VSA4	-0.18734
EState_VSA5	-0.09459
EState_VSA9	-0.03903
SlogP_VSA1	-0.0659
SlogP_VSA2	0.108425
SlogP_VSA3	0.085089
SlogP_VSA4	-0.00246
SlogP_VSA5	-0.06211
SlogP_VSA6	-0.31778
SMR_VSA3	-0.01469
SMR_VSA4	0.036752
SMR_VSA5	0.06985
AATS0dv	0.024918
AATS0i	0.149709
AATS0v	-0.2039
AATS0Z	-0.06046
AATS3s	0.085577
AATSC0dv	0.088606
AATSC0i	-0.20065
AATSC0v	-0.17888
AATSC2s	-0.02832
AATSC2Z	-0.00649
AATSC3dv	-0.09836
AATSC3s	0.019336
AATSC3v	-0.15145
AATSC3Z	-0.09942
ATS0Z	-0.1593
ATSC1dv	-0.06651
ATSC1pe	-0.07621
ATSC1s	-0.04894
ATSC1v	-0.04919
ATSC1Z	-0.06165
ATSC2dv	-0.06349
ATSC2s	-0.06278
ATSC2Z	-0.00096
ATSC3dv	-0.09685
ATSC3s	0.018622
ATSC4dv	-0.1145

Descriptors	Kel
ATSC4pe	-0.08853
ATSC4s	-0.14035
ATSC4v	-0.07836
ATSC4Z	-0.06803
ATSC7dv	0.046395
ATSC6d	0.12273
ATSC7d	0.055811
ATSC2d	-0.06816
ATSC3d	-0.11492
ATSC7p	0.028649
ATSC7v	0.049192
n6Ring	-0.29412
NaasC	-0.30226
nAcid	0.125134
NsOH	0.141103
NssssC	0.004887
SaasC	-0.20684
SdssC	-0.1086
SssCH2	-0.13043
SsssCH	-0.11546
SssssC	0.126144
SM1_DzZ	-0.00263
C3SP2	-0.26238
ETA_dBeta	-0.18028
IC1	0.021948
IC2	-0.01813
VE1_A	-0.18638
VR1_A	-0.08943
VSA_EState8	-0.19165
MW	-0.16634
SLogP	-0.33522
PEOE_VSA1	0.088004
PEOE_VSA2	0.107947
BCUTd-1h	-0.05375
BCUTdv-1h	-0.07305
BCUTdv-1l	0.053703
BCUTs-1h	0.089257
BCUTs-1l	-0.03118
BCUTv-1l	0.131071
BCUTZ-1h	-0.0899
BalabanJ	0.190422

Table 5.2 VIF results for multicollinearity

feature	VIF
bpol	1718.895
EState_VSA4	11.70058
EState_VSA5	7.166046
EState_VSA9	12.78435
SlogP_VSA1	57.1528
SlogP_VSA2	501.1003
SlogP_VSA3	32.50285
SlogP_VSA4	19.9326
SlogP_VSA5	83.73195
SlogP_VSA6	150.7025
SMR_VSA3	16.3501
SMR_VSA4	11.77761
SMR_VSA5	70.56227
AATS0dv	1150.483
AATS0i	7715.406
AATS0v	4039.512
AATS0Z	297.3467
AATS3s	97.49723
AATSC0dv	837.3695
AATSC0i	219.6799
AATSC0v	4345.758
AATSC2s	27.221
AATSC2Z	11.15641
AATSC3dv	21.40764
AATSC3s	16.2327
AATSC3v	9.075571
AATSC3Z	4.483596
ATS0Z	552.2217
ATSC1dv	235.576
ATSC1pe	22.89574
ATSC1s	91.08892
ATSC1v	23.76443
ATSC1Z	18.00953
ATSC2dv	44.28622
ATSC2s	39.58189
ATSC2Z	21.90144
ATSC3dv	38.06899
ATSC3s	15.60809
ATSC4pe	18.05426
ATSC4s	9.051718
ATSC4v	13.94599
ATSC4Z	8.679376

ATSC7dv	4.77406
ATSC6d	6.620823
ATSC7d	3.023629
ATSC2d	6.332882
ATSC3d	22.20868
ATSC7p	13.99998
ATSC7v	18.39237
n6Ring	21.15242
NaasC	53.83085
nAcid	5.91172
NsOH	33.5137
NssssC	14.93902
SaasC	18.22518
SdssC	4.268738
SssCH2	28.13398
SsssCH	11.31286
SssssC	12.51603
SM1_DzZ	386.0746
C3SP2	16.83455
ETA_dBeta	56.08339
IC1	979.1685
IC2	1108.484
VE1_A	366.2222
VR1_A	2.077173
VSA_EState8	31.5833
MW	6972.192
SLogP	198.7389
PEOE_VSA1	73.68525
PEOE_VSA2	12.30498
BCUTd-1h	560.7318
BCUTdv-1h	893.587
BCUTdv-1l	20.68057
BCUTs-1h	100.9337
BCUTs-1l	129.3589
BCUTv-1l	6502.058
BCUTZ-1h	56.73356
BalabanJ	105.1315

5.1 DISCUSSION ON FEATURE SELECTION

The association between the k_{el} rate constants and the molecular descriptors of the drugs was plotted and thoroughly studied using filter and embedded methods. (Figures 5.1, 5.3 & Table 5.1) Although the k_{el} values showed a high correlation with the $t_{1/2}$, it was observed that the correlation between the $\log k_{el}$ values and 2-D molecular descriptors was significantly low.

The application of PCA to retrieve principal components that are a linear combination of the molecular descriptors was carried out successfully. About 20 components were required to demonstrate 90% of the variance present in the data (Figure 5.4). The obtained principal components were again correlated with the descriptors to understand the linear combination of features. (Figure 5.6)

In Figure 5.7, all the highlighted cells show significant positive (yellow highlights) or negative (blue highlights) correlation which validates the presence of linear combination. For example, the significant negative correlations highlighted between PC4 and the descriptors $bpol$ and $SlogP_VSA2$ suggest that more than one descriptor contributes to identifying each component.

On the other hand, k_a values presented coefficients with very poor to almost no correlation. (Figure 5.10) Similar analyses like PCA, and high correlation filter were performed to determine the most informative features of the pruned data to avoid regression error. In the case of k_a rate constant data, 1446 descriptors were transformed into 20 principal components indicating that over 75% of the variance from the original dataset was captured in the data.

5.2 PERFORMANCE ANALYSIS

Random forest regression models with about 100 decision trees for each set of features obtained through different feature selection algorithms were developed. (Figures 5.8 & 5.13) The training scores (R^2 train) obtained for both k_{el} and k_{abs} prediction models ranged between 0.80-0.90. However, the predictive performance (R^2 test) of the models was non-significant.

The highest predictive ability observed was 0.40 by the k_{el} dataset of 464 drugs. This performance is not validated as the performance of the model fluctuated

between 0.25-0.33 for most of the training and test set combinations that were randomly generated. This proved that the generalization ability of the model must be improved further. One of the possible causes for this behaviour in both k_{el} and k_a prediction models is due to the vast diversity in the structures of drugs collected.

Therefore, implementing regression after classifying the drugs into different groups based on how high/low the elimination rate constant is with a larger dataset might lead to better results.

In the case of k_{abs} absorption prediction models, the data collected from the literature resulted in poor predictivity due to the multicollinearity and various factors affecting the rate of absorption. These factors include different groups of populations, ages, genders, various formulations, different routes of administration, different pH conditions in the stomach, variations in absorption in healthy volunteers vs. diseased patients, and so on. Other normalization techniques that can keep the occurrence of an error to a minimum and a different set of descriptor data could improve the correlation and predictivity of the model.

Some of the descriptors that might improve the performance of both k_{el} and k_{abs} prediction models include physiochemical properties like solubility, partition coefficients, dissociation constants, and so on.

CONCLUSION

The in-silico prediction of k_{el} and k_a rate constants is an essential concept of pharmacokinetics that requires attention to predict the dosing regimen appropriately. This current work has demonstrated an approach to use the random forest algorithm for both k_{el} and k_{abs} regression models. Prediction models developed in this study are based entirely on structural descriptors and not interspecies extrapolation. Thus, the predictive performance of the models needs further improvement by increasing the associativity between the rate constants and the molecular descriptors. Some of the possible ways to improve the model would be to use a larger yet chemically classified dataset to carefully train the prediction model on how certain chemical descriptors affect the target variable in the study. In conclusion, using genetic algorithms or advanced machine learning methods with other normalization techniques that capture non-linear relationships could refine the predicted values of k_{el} and k_{abs} rate constants.

REFERENCES

1. Berellini, G., Waters, N. J., & Lombardo, F. (2012). In silico Prediction of Total Human Plasma Clearance. *Journal of Chemical Information and Modeling*, 52(8), 2069-2078. doi:10.1021/ci300155y
2. Chatterjee, B., Sengupta, P., & Tekade, R. K. (2021). Pharmacokinetic characterization of drugs and new product development. In *Biopharmaceutics and Pharmacokinetics Considerations* (pp. 195-277). doi:10.1016/b978-0-12-814425-1.00010-3
3. Freitas, A. A., Limbu, K., & Ghafourian, T. (2015). Predicting volume of distribution with decision tree-based regression methods using predicted tissue:plasma partition coefficients. *Journal of Cheminformatics*, 7(1), 6. doi:10.1186/s13321-015-0054-x
4. Gombar, V. K., & Hall, S. D. (2013). Quantitative structure-activity relationship models of clinical pharmacokinetics: clearance and volume of distribution. *Journal of Chemical Information and Modeling*, 53(4), 948-957. doi:10.1021/ci400001u
5. Grogan S, Preuss CV. Pharmacokinetics. [Updated 2022 Jun 23]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2023 Jan-
6. Hallare J, Gerriets V. Half Life. [Updated 2022 Jun 23]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2023 Jan
7. Hamidović, A., Hasković, E., Muhić, S., Planinić, M., Omerović, N., & Škrbo, S. (2021). Application of in silico methods in pharmacokinetic studies during drug development. In *IFMBE Proceedings*. IFMBE Proceedings (pp. 499-510). doi:10.1007/978-3-030-73909-6_58
8. Horde GW, Gupta V. Drug Clearance. [Updated 2022 Jun 23]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2023 Jan-
9. Lu, J., Lu, D., Zhang, X., Bi, Y., Cheng, K., Zheng, M., & Luo, X. (2016). Estimation of elimination half-lives of organic chemicals in humans using gradient boosting machine. *Biochimica et Biophysica Acta (BBA) - General Subjects*, 1860(11), 2664-2671. doi:10.1016/j.bbagen.2016.05.019
10. Mercader, A. G., Duchowicz, P. R., Fernández, F. M., & Castro, E. A. (2008). Modified and enhanced replacement method for the selection of molecular descriptors in QSAR and QSPR theories. *Chemometrics and*

- Intelligent Laboratory Systems, 92(2), 138-144. doi:10.1016/j.chemolab.2008.02.005
11. Polli, J. E. (2008). In vitro studies are sometimes better than conventional human pharmacokinetic in vivo studies in assessing bioequivalence of immediate-release solid oral dosage forms. *The AAPS Journal*, 10(2), 289-299. doi:10.1208/s12248-008-9027-6
 12. Suresh, S., Newton, D. T., Everett, T. H., 4th, Lin, G., & Duerstock, B. S. (2022). Feature selection techniques for a machine learning model to detect autonomic Dysreflexia. *Frontiers in Neuroinformatics*, 16, 901428. doi:10.3389/fninf.2022.901428
 13. Sylvain Chemtob, 20 - Basic Pharmacologic Principles, Editor(s): Richard A. Polin, William W. Fox, Steven H. Abman, Fetal and Neonatal Physiology (Fourth Edition), W.B. Saunders, 2011, Pages 211-223, ISBN 9781416034797
 14. Tedesco, M., Pulliainen, J., Takala, M., Hallikainen, M., & Pampaloni, P. (2004). Artificial neural network-based techniques for the retrieval of SWE and snow depth from SSM/I data. *Remote Sensing of Environment*, 90(1), 76-85. doi:10.1016/j.rse.2003.12.002
 15. Wang, Y., Liu, H., Fan, Y., Chen, X., Yang, Y., Zhu, L., ... Zhang, Y. (2019). In Silico Prediction of Human Intravenous Pharmacokinetic Parameters with Improved Accuracy. *Journal of Chemical Information and Modeling*. doi:10.1021/acs.jcim.9b00300
 16. Wei, M., Zhang, X., Pan, X. et al. HobPre: accurate prediction of human oral bioavailability for small molecules. *J Cheminform* **14**, 1 (2022). <https://doi.org/10.1186/s13321-021-00580-6>
 17. Williams, B., Halloin, C., Löbel, W., Finklea, F., Lipke, E., Zweigerdt, R., & Cremaschi, S. (2020). Data-Driven Model Development for Cardiomyocyte Production Experimental Failure Prediction. 30th European Symposium on Computer Aided Process Engineering, 1639-1644. doi:10.1016/b978-0-12-823377-1.50274-3
 18. Xu, X., Zhang, W., Huang, C., Li, Y., Yu, H., Wang, Y., ... Ling, Y. (2012). A Novel Chemometric Method for the Prediction of Human Oral Bioavailability. *International Journal of Molecular Sciences*, 13(6), 6964-6982. doi:10.3390/ijms13066964

19. Zhivkova, Z., & Doytchinova, I. (2012). Prediction of Steady-State Volume of Distribution of Acidic Drugs by Quantitative Structure-Pharmacokinetics Relationships. *Journal of Pharmaceutical Sciences*, 101(3), 1253-1266. doi:10.1002/jps.22819

APPENDIX

(a) PEARSON CORRELATION

```
#importing the required packages

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

from sklearn.ensemble import RandomForestRegressor

from sklearn.preprocessing import StandardScaler

from sklearn.model_selection import train_test_split

from sklearn import metrics

from sklearn.metrics import r2_score

import seaborn as sns

from sklearn.feature_selection import VarianceThreshold

from sklearn.decomposition import PCA

##.....

#Reading the data

data =
pd.read_excel('C:/Users/SESHU.A/Documents/keldata.xlsx')

print(data.shape)

#Dropping null values

data=data.dropna(axis=0)

data=data.reset_index(drop=True)

data=data.iloc[:,0:]

x=data.iloc[:,2:]

print(x)
```

```

y = data.iloc[:,1]

print(y)

features=x.columns

# Pearson correlation - Feature selection

target_corr= data.corr(method='pearson')["kel"]

print("Correlation with the Target ", target_corr, "\n")

target_corr=target_corr.dropna()

target_corr.to_excel("C:/Users/SESHU.A/Documents/correlation.
xlsx")

posrelevant_features = target_corr.index[target_corr > 0.2]

print(posrelevant_features)

negrelevant_features = target_corr.index[target_corr < -0.1]

print(negrelevant_features)

finaldata0=data[posrelevant_features]

finaldata1=data[negrelevant_features]

finaldata0=finaldata0.drop(["kel"], axis=1)

#Creating a filtered dataset with the correlated features

all_features=pd.concat([finaldata0,finaldata1], axis=1)

all_features.to_csv("C:/Users/SESHU.A/Documents/c.csv")

finaldata= pd.read_csv("C:/Users/SESHU.A/Documents/c.csv")

finaldata= finaldata.iloc[:,1:]

X_train, X_test, y_train, y_test = train_test_split(finaldata,
y, test_size=0.2)

##.....

```

(b) DROPPING HIGHLY CORRELATED FEATURES FROM THE TRAINING SET

```
def highlycorr(data,threshold):

    col_correlation=set()
    corr_mat=data.corr()
    for i in range(len(corr_mat.columns)):
        for j in range(i):
            if corr_mat.iloc[i,j] > threshold:
                colname=corr_mat.columns[i]
                col_correlation.add(colname)

    return col_correlation


corr_features=highlycorr(X_train,0.9)

print(corr_features)

print('\n')

train= X_train.drop(corr_features,axis=1)

Xtest= X_test.drop(corr_features,axis=1)

print(Xtrain.shape)

print(Xtest.shape)

###.....
```

(c) REMOVING MULTICOLLINEARITY USING VIF

```
from statsmodels.stats.outliers_influence import
variance_inflation_factor

z=data.iloc[:,2:]

vif_data = pd.DataFrame()

vif_data["feature"] = z.columns
```

```

# calculating VIF for each feature

vif_data["VIF"] = [variance_inflation_factor(z.values, i)
                   for i in range(len(z.columns))]

print (vif_data)

vif_data.to_excel("C:/Users/SESHU.A/Documents/multico.xlsx")

##.....

```

(d) ONLY RANDOM FOREST APPROACH

```

x=data.iloc[:,2:]

print(x)

y = data.iloc[:,1]

print(y)

rf=RandomForestRegressor(n_estimators=100,random_state=0)

rf.fit(x,y)

selector=SelectFromModel(rf)

mask1=selector.get_support()

new=features[mask1]

x_fil=data[new]

X_train,X_test,y_train,y_test =
train_test_split(x_fil,y,test_size=0.2)

# PLOTTING GRAPHS

features = z.columns

importances = rf.feature_importances_

indices = np.argsort(importances)

# customized number

num_features = 20

```

```

plt.figure(figsize=(10,10))

plt.title('Feature Importances')

# only plot the customized number of features

plt.barh(range(num_features),          importances[indices[-
num_features:]],          color=['skyblue','black','pink'],
align='center')

plt.yticks(range(num_features),    [features[i]    for    i    in
indices[-num_features:]])

plt.xlabel('Relative Importance')

plt.style.use("seaborn-pastel")

plt.show()

#MODEL TRAINING AND PREDICTION

rf.fit(X_train,y_train)

y_pred=rf.predict(X_test)

#EVALUATION

print("R2 score:",r2_score(y_test,y_pred))

print("\n")

print("training score:",rf.score(X_train,y_train))

```

```

##.....

```

(e) PCA AND RANDOM FOREST

```

x=data.iloc[:,2:]

print(x)

y = data.iloc[:,1]

print(y)

#REMOVING FEATURES WITH LOW VARIANCE

```

```

features= z.columns

v_threshold= VarianceThreshold(threshold=0)

v_threshold.fit(z)

mask=v_threshold.get_support()

new0=features[mask]

x=data[new0]

sc_X = StandardScaler()

sc_YT = StandardScaler()

x_std=sc_X.fit_transform(x)

pca=PCA(15)

pca.fit(x_std)

x_dimen=pca.transform(x_std)

#TRAIN AND TEST SPLIT

X_train,X_test,y_train,y_test =
train_test_split(x_dimen,y,test_size=0.2)


#X_test=X_test.values

sc_Y = StandardScaler()

y_std=sc_Y.fit_transform(X_test)

pca.fit(y_std)

y_dimen=pca.transform(y_std)

#MODEL BUILDING

reg = RandomForestRegressor(n_estimators=100, random_state=0)

reg.fit(X_train, y_train)

y_pred = reg.predict(y_dimen)

```

```

print(y_pred)

#PLOTING GRAPHS

print("PCA          feature          explained
variance",pca.explained_variance_)

plt.bar(range(1,len(pca.explained_variance_)+1),
pca.explained_variance_)

plt.xlabel('PCA Feature')

plt.ylabel('Explained variance')

plt.title('Feature Explained Variance')

plt.show()

print("      \n      \n      Explained      variance      ratio",
pca.explained_variance_ratio_)

pcaplot=PCA().fit(x_std)

plt.plot(np.cumsum(pcaplot.explained_variance_ratio_))

plt.xlabel('number of components')

plt.ylabel('cumulative explained variance');

print("R2 score:",r2_score(y_test,y_pred))

fig = plt.figure(figsize=(10,8))

fig.add_subplot(projection='3d')

plt.scatter(x_dimen[:,0],x_dimen[:,1],          s=50,
c='purple',alpha=0.35)

plt.xlabel('PC1')

plt.ylabel('PC2')

#plt.zlabel('PC3')

plt.style.use("default")

plt.title("2D-PCA")

```

```

plt.grid(which='both')

plt.grid(which='minor', alpha=0.5)

plt.grid(which='major', alpha=0.3)

plt.show()

# MODEL EVALUATION

print('Root          Mean          Squared          Error:',
      np.sqrt(metrics.mean_squared_error(y_test, y_pred)))

print("R2 score:", r2_score(y_test, y_pred))

##.....

```

(f) GENERATING THE CORRELATION MATRIX AND CIRCLE GRAPH

```

import statistics as s

feature_names=x.columns

X_norm= x-x.std(axis=0)

#X_norm.dropna(inplace=True)

from mlxtend.plotting import plot_pca_correlation_graph

figure,          correlation_matrix          =
plot_pca_correlation_graph(X_norm,

feature_names,

dimensions=(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15),

figure_axis_size=6)

##.....

```

(g) GENERATING A TREE FROM THE FOREST

```

import pydot

```

```
# Pull out one tree from the forest

Tree = rf.estimators_[5]

# Export the image to a dot file

from sklearn import tree

plt.figure(figsize=(25,15))

tree.plot_tree(Tree,filled=True,

               rounded=True,

               fontsize=14);
```