Conjugated quantitative structure-property relationship models: Prediction of kinetic characteristics linked by the Arrhenius equation

Dmitry Zankov, Timur Madzhidov, Igor Baskin, Alexandre Varnek

Received: 14 December 2022 | Revised: 8 July 2023 | Accepted: 24 July 2023

DOI: 10.1002/minf.202200275

RESEARCH ARTICLE

molecular informatics

Conjugated quantitative structure-property relationship models: Prediction of kinetic characteristics linked by the Arrhenius equation

Dmitry Zankov[1] | **Timur Madzhidov**[2] | **Igor Baskin**[3] | **Alexandre Varnek**[1]

[1]Laboratory of Chemoinformatics, University of Strasbourg, France

[2]Chemistry Solutions, Elsevier Ltd, Oxford, United Kingdom

[3]Department of Materials Science and Engineering, Technion - Israel Institute of Technology, Israel

Correspondence
Alexandre Varnek, Laboratory of Chemoinformatics, University of Strasbourg, France.
Email: varnek@unistra.fr

Timur Madzhidov, Chemistry Solutions, Elsevier Ltd, Oxford OX5 1GB, United Kingdom.
Email: t.madzhidov@elsevier.com

Igor Baskin, Department of Materials Science and Engineering, Technion - Israel Institute of Technology, Israel.
Email: igorb@technion.ac.il

Abstract

Conjugated QSPR models for reactions integrate fundamental chemical laws expressed by mathematical equations with machine learning algorithms. Herein we present a methodology for building conjugated QSPR models integrated with the Arrhenius equation. Conjugated QSPR models were used to predict kinetic characteristics of cycloaddition reactions related by the Arrhenius equation: rate constant log$k$, pre-exponential factor log$A$, and activation energy E_a. They were benchmarked against single-task (individual and equation-based models) and multi-task models. In individual models, all characteristics were modeled separately, while in multi-task models log$k$, log$A$ and E_a were treated cooperatively. An equation-based model assessed log$k$ using the Arrhenius equation and log$A$ and E_a values predicted by individual models. It has been demonstrated that the conjugated QSPR models can accurately predict the reaction rate constants at extreme temperatures, at which reaction rate constants hardly can be measured experimentally. Also, in the case of small training sets conjugated models are more robust than related single-task approaches.

KEYWORDS
Arrhenius equation, conjugated models, QSPR

1 | INTRODUCTION

A chemical reaction can be quantitatively described by such kinetic characteristics as the rate constant (log$k$), the pre-exponential factor (log$A$), and activation energy (E_a). Their knowledge is of particular importance because the distribution of reactant and product concentration at any moment can be calculated based on known kinetics. QSPR modeling of chemical reactions has made significant progress in recent years [1–4]. QSPR methodology employs machine learning algorithms to the data on reaction characteristics measured in the experiment to predict them for new reactions. Many approaches were proposed for reaction rate prediction. Usually, quantum chemistry approaches are used for the search for elementary reaction mechanisms and estimate reaction barriers and rates [5–7]. Computationally efficient machine learning potentials were shown to be a valuable alternative to quantum chemistry in the estimation of local minima and transition states energy [8]. Machine learning is currently widely used to predict reaction rate constants based on structural features of reactants and

products represented by a set of chemical descriptors [9]. This approach can be dated back to early studies based on the Linear Free Energy Principle [10] and the application of substituent constants as descriptors [11]. It has also been shown that quantum chemical descriptors are a good alternative to structural descriptors [12].

In our previous publications, we reported predictive models for the rate constants of S_N2 [13, 14] and E2 [15, 16] reactions. There are also examples of machine learning applications for predicting the activation energies of reactions. Singh *et al.* applied popular machine learning algorithms to predict the activation barriers of hydrogenation/dehydrogenation reactions [17]. Gambow and coworkers developed a deep graph convolutional neural network trained on the activation barriers of gas-phase reactions obtained with quantum-chemical calculations [5, 18]. Jorner *et al.* proposed an approach that combines traditional DFT transition state modeling and machine learning [19] and trained the model using different machine learning algorithms to accurately predict the reaction barriers of the nucleophilic aromatic substitution reaction (S_NAr). Chin and coworkers reported kinetic and thermodynamic analysis of the thermal degradation of plastic wastes including application of artificial neural networks and global optimization algorithms [20–23].

Previously, the temperature dependence of the reaction rate was mostly modeled by adding the temperature to the set of structural descriptors [16]. However, the dependence of the rate constant (log$k$) on the temperature is known to be expressed by the Arrhenius equation (1) that relates reaction rate with the temperature and two other parameters that are assumed to be temperature independent: the pre-exponential factor (A), and activation energy (E_a).

In our previous study [24] we reported SVR (Support Vector Regression) and GTM (Generative Topographic Mapping) modeling of log$k$, log$A$ and E_A of cycloaddition reactions. Two scenarios for log$k$ assessment were examined. In the first scenario, the SVR algorithm learns to predict log$k$ directly from reaction descriptors. In the second scenario, two independent individual models are built: (i) for predicting the log$A$ and (ii) for predicting the E_A, which were used to calculate log$k$ using the Arrhenius equation:

$$\log k = \log A - \frac{E_A}{2.303RT} \quad (1)$$

We observed that the predicted values of log$k$ calculated using the Arrhenius equation (*Arrhenius-based* model) were less accurate in comparison to the *individual* model built directly from experimental values of log$k$.

The models embedding thermodynamic laws (*conjugated QSPR models*) were described in our recent study [25]. We proposed a machine learning model that combines ridge regression and a neural network with an equation that relates tautomer acidities with their equilibrium constants. We have demonstrated that the predictive performance of such conjugated models was as good as for the individual ones, while the former had some additional benefits like a good prediction of acidities for minor tautomers.

Here, the conjugated modeling approach involving ridge regression and neural network algorithms is applied to the kinetic parameters of chemical reactions linked by the Arrhenius equation: rate constant log$k$, pre-exponential factor log$A$, and activation energy E_a.

We used the data set on cycloaddition reactions from our previous study [24] to build *individual* (*single-task*), *equation-based* (*Arrhenius-based*), *multi-task*, and *conjugated* models for predicting log$k$, log$A$ and E_A. *Individual* models were built independently for each kinetic characteristic (Figure 1, I). The *Arrhenius-based* model uses the Arrhenius equation to calculate the log$k$ with log$A$ and E_A predicted by individual models (Figure 1, I). The multi-task approach (Figure 1, II) uses all available data across the different reaction characteristics and models them cooperatively in contrast to single-task learning. Conjugated learning (Figure 1, III) uses all available data on multiple tasks, but, in contrast to the multi-task approach, explicitly embeds a mathematical equation (in this study it is the Arrhenius equation) relating the tasks to the machine learning algorithm. This approach ensures that the predicted reaction characteristics satisfy the fundamental chemical laws and empowers the conjugated QSPR models with new capabilities.

2 | DESIGN OF CONJUGATED LEARNING ALGORITHMS

2.1 | Ridge regression individual models

Ridge regression (RR) is a popular machine learning algorithm that was extensively used in practice [26]. In ridge regression, the prediction of reaction characteristic y^{pred} is performed by multiplying the reaction descriptors X by the vector of regression weights w:

$$y^{pred} = Xw \quad (2)$$

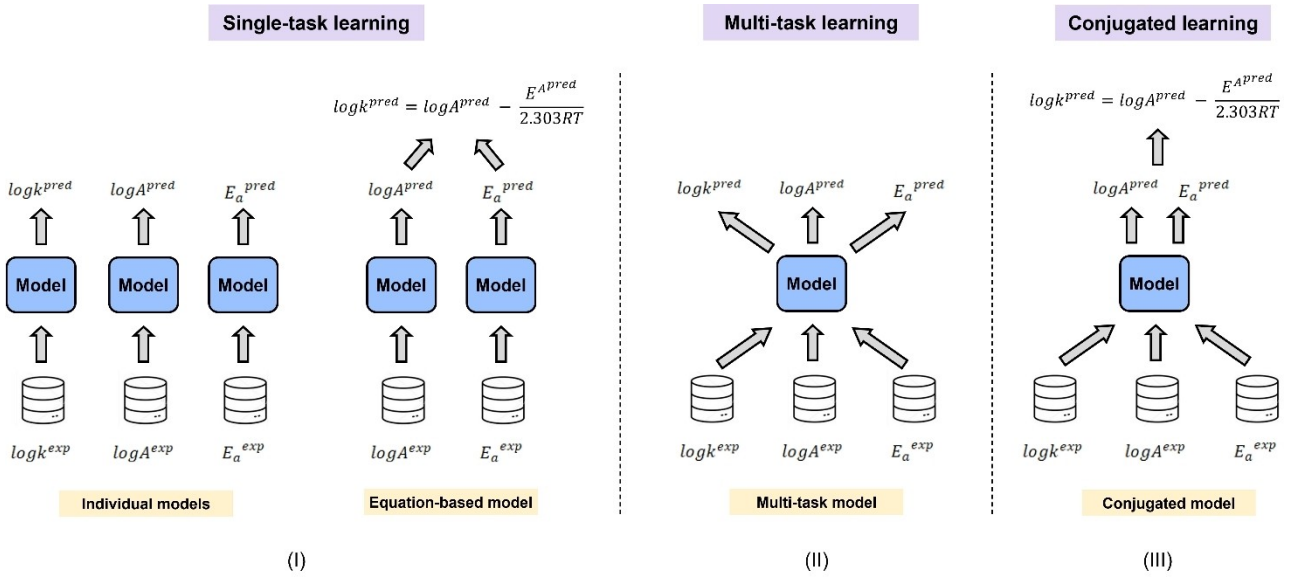

FIGURE 1 Approaches to modeling kinetic characteristics related by Arrhenius equation. In ordinary single-task learning (I) each characteristic is modeled independently. Multi-task learning (II) performs cooperative modeling of all three characteristics, whereas conjugated learning (III) embeds the strict mathematical relationship relating the kinetics characteristics (Arrhenius equation) into the machine learning algorithm.

The regression weights w can be calculated using the following expression:

$$w = (X^T X + \lambda I)^{-1} X^T y^{exp} \tag{3}$$

where X is the descriptor matrix of training reactions associated with experimental values y^{exp} of the target characteristic. Hyperparameter λ is a regularization coefficient controlling the complexity of the model. We used ridge regression to independently build three *individual* models for predicting the log$k$, log$A$ and E_A of cycloaddition reactions. The regularization coefficient was adjusted using the grid search technique.

2.2 | Ridge regression conjugated models

In conjugated models, fundamental chemical laws are integrated with machine learning algorithms. In this study, we consider the Arrhenius equation, which can be embedded into the ridge regression algorithm. In the *equation-based* (*Arrhenius-based*) model the rate constant y_K^{pred} is calculated using the Arrhenius equation applied to the values of log$A$ and E_A predicted by *individual* QSPR models from reaction descriptors X_K:

$$\log k = \log A - \frac{E_A}{2.303RT} \Rightarrow y_K^{pred} = X_K w_A - T X_K w_E \tag{4}$$

where T is the diagonal matrix with the elements that are calculated as:

$$\frac{1}{2.303RT_i} \tag{5}$$

and T_i is the temperature of the i-th reaction. On the other hand, if experimental data on log$k$ are available, the Arrhenius equation can be integrated with ridge regression using a special loss function:

$$E_K(w_A,\ w_E) = \\ \|y_K^{exp} - y_K^{pred}\|^2 = \|y_K^{exp} - X_K w_A + T X_K w_E\|^2 \tag{6}$$

In the case of $E_K(w_A,\ w_E)$, there are two sets of regression weights, w_A (for predicting log$A$) and w_E (for predicting E_A), which can be optimized to predict the log$k$. To enable correct prediction of log$A$ and the E_A, loss function $E_K(w_A,\ w_E)$ can be combined with individual loss functions for the log$A$ and E_A and regularization terms:

$$E_A(w_A) = \|y_A^{exp} - y_A^{pred}\|^2 = \|y_A^{exp} - X_A w_A\|^2 + \lambda_A w_A^T w_A \tag{7}$$

$$E_E(w_E) = \|y_E^{exp} - y_E^{pred}\|^2 = \|y_E^{exp} - X_E w_E\|^2 + \lambda_E w_E^T w_E \tag{8}$$

resulting in a conjugated model loss function:

$$E(w_A, w_E) = c_K E_K(w_A, w_E) + c_A E_A(w_A) + c_E E_E(w_E) + \lambda_A w_A^T w_A + \lambda_E w_E^T w_E \quad (9)$$

where c_K, c_A, c_E are trade-off coefficients that control the contribution of each type of the loss function to conjugated loss $E(w_A, w_E)$, λ_A and λ_E are regularization coefficients. After differentiation of the loss function $E(w_A, w_E)$, the optimal regression weights w_A and w_E can be calculated using the following analytical expressions:

$$w_A = (I - BD)^{-1}(A + BC) \quad (10)$$

$$w_E = (I - DB)^{-1}(C + DA) \quad (11)$$

where matrices A, B, C, D are obtained as follows:

$$\begin{aligned} A &= \left(c_K X_K^T X_K + c_A X_A^T X_A + \lambda_A I\right)^{-1}\left(c_K X_K^T y_K + c_A X_A^T y_A\right) \\ B &= \left(c_K X_K^T X_K + c_A X_A^T X_A + \lambda_A I\right)^{-1}\left(c_K X_K^T T X_K\right) \\ C &= \\ &\left(c_K X_K^T T^T T X_K + c_E X_E^T X_E + \lambda_E I\right)^{-1}\left(c_E X_E^T y_E - c_K X_K^T T y_K\right) \\ D &= \left(c_K X_K^T T^T T X_K + c_E X_E^T X_E + \lambda_E I\right)^{-1}\left(c_K X_K^T T X_K\right) \end{aligned} \quad (12)$$

As a result, regression weights w_A and w_E in the *conjugated* model are estimated using the training sets of log$k$ (X_K), log$A$ (X_A) and E_A (X_E) data.

2.3 | Neural network individual, multi-task, and conjugated models

Individual, *multi-task*, and *conjugated* models can be built using neural networks (NN). In *individual* models, each characteristic is modeled independently using a standard multilayer neural network with one or more hidden layers and one output neuron (Figure 2a). *Multi-task* models can be built using a neural network with three output neurons, each predicting one of the kinetic characteristics (Figure 2b). This neural network can be trained using the multi-task loss:

$$Multitask\ loss = c_K\left(logk^{exp} - logk^{pred}\right)^2 + c_A\left(logA^{exp} - logA^{pred}\right)^2 + c_E\left(E_a^{exp} - E_a^{pred}\right)^2 \quad (13)$$

where c_K, c_A, c_E are coefficients that control the contribution of each type of error to the multi-task loss.

The *conjugated* models can be built using the neural networks shown in Figure 2c. This neural network has two output neurons. The first output neuron predicts log$A$ and the second one predicts E_A (Figure 2c). The predicted values of log$A$ and E_A are then used to calculate the prediction of log$k$ using the Arrhenius equation. Finally, the obtained predicted values of log$k$, log$A$ and E_A are used to calculate the conjugated loss:

$$Conjugated\ loss = c_K\left(logk^{exp} - \left(logA^{pred} - \frac{E_a^{pred}}{2.303RT}\right)\right)^2 + c_A\left(logA^{exp} - logA^{pred}\right)^2 + c_E\left(E_a^{exp} - E_a^{pred}\right)^2 \quad (14)$$

Individual, *multi-task*, and *conjugated* NN models discussed hereafter had one hidden layer with 256 neurons. Neural network weights were optimized using a gradient descent algorithm at a learning rate of 0.001. The

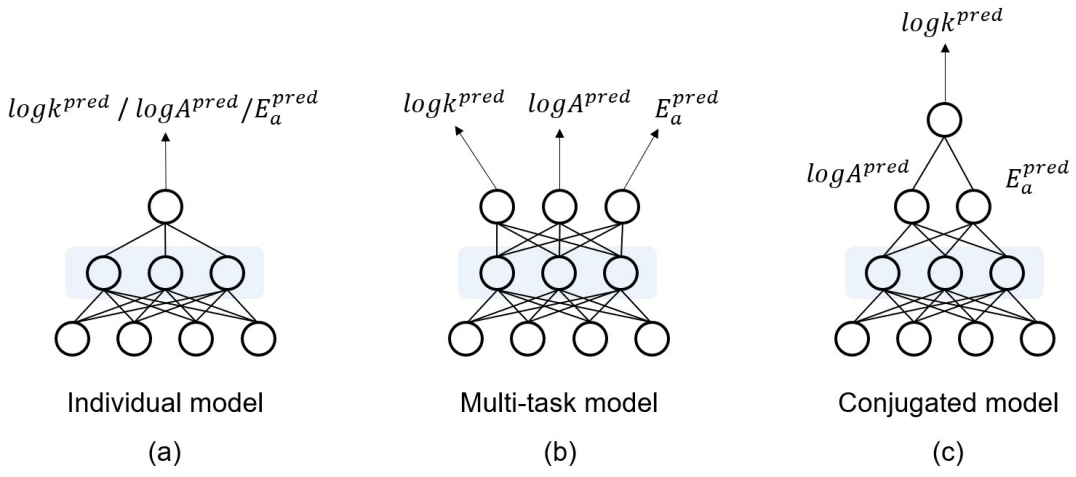

FIGURE 2 Neural network architectures for building an *individual* (a), *multi-task* (b), and *conjugated* (c) model for prediction of the reaction kinetic characteristics related by the Arrhenius equation.

complexity of the *individual* and *conjugated* NN models were controlled by the weight decay parameter (L2 regularization), which took values from 10^{-3} to 10^{1}. Neural networks were implemented using the PyTorch package [27].

3 | COMPUTATIONAL DETAILS

3.1 | Data

The reaction data set for model building were taken from our previous paper [24]. The data set includes 1849 cycloaddition reactions with 1849 experimental values of log*k*, 1236 experimental values of log*A*, and 1350 experimental values of E_a (kJ/mol). The rate constants log*k* were measured in different solvents and at different temperatures *T*. The data set contains Diels-Alder (4+2) cycloaddition, (3+2) dipolar cyclization, and (2+2) cycloadditions. Among the 1849 reactions, there were 763 unique structural transformations (Table 1).

The data set was divided into training and test sets (in the proportion of 90/10) so that the test set contained structural transformations which did not occur in the training set (Table 1). As a result, the test set contained 73 unique structural transformations that were not represented in the training set, which consisted of 690 unique structural transformations (Table 1). The training set was used to build the *individual*, *Arrhenius-based*, *multi-task*, and *conjugated* models, while the test set was used to evaluate the predictive performance of the models.

3.2 | Descriptors

Each cycloaddition reaction was transformed into the corresponding Condensed Graph of Reaction (CGR) (Figure 3) [28] generated using the CGRtools package [29]. A CGR is derived from the superposition of products and reactants and contains both conventional chemical bonds (single, double, triple, aromatic, etc.) and so-called "dynamic" bonds describing chemical transformations, i.e., breaking or forming a bond or changing bond order.

All generated CGRs were processed using the ISIDA tool [30, 31] to calculate fragment descriptors counting the occurrence of particular subgraphs (structural fragments) of different topologies and sizes. We tested different types of fragment descriptors and selected atom-centered descriptors with a radius from 2 to 5. The total number of fragment descriptors was 3733. The vector of fragment descriptors for each reaction was concatenated with the vector of solvent descriptors, which included 14 descriptors describing such properties of solvent as polarity, polarizability, Catalan constants SPP, SA, SB, Kamlet-Taft constants α, β, π*, dielectric constants, the function of the refractive index. These descriptors were successfully applied in our previous publications [24, 25, 32, 33].

To build *individual* and *multi-task* models, the fragment/solvent descriptor matrices were concatenated with the temperature descriptor. In *Arrhenius-based* and *conjugated* models, only fragment and solvent descriptors were used as reaction descriptors, while reaction temperatures were introduced by the Arrhenius equation. The calculated descriptors constituted three

TABLE 1 Description of the training and test set on cycloaddition reactions.

	# Reactions	# Unique structural transformations	# Kinetic characteristics		
			log*k*	log*A*	E_A
Training set	1478	690	1478	1008	1120
Test set	371	73	371	228	230

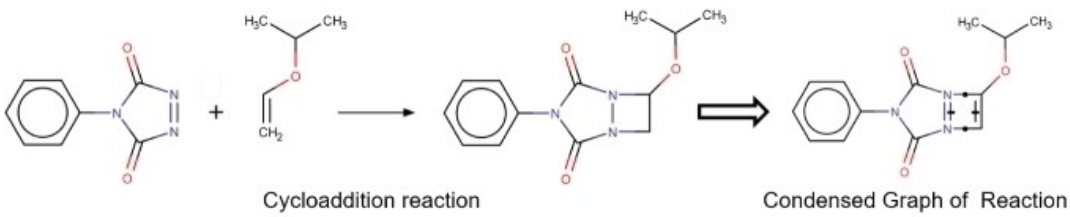

FIGURE 3 A cycloaddition reaction from the data set and the corresponding CGR describing the structural transformation. The formed bonds are denoted with a circle, while the broken ones are crossed.

descriptor matrices: X_K, X_A and X_E, where the number of rows in each matrix corresponds to the number of experimental values of log$k$, log$A$ and E_A for cycloaddition reactions (Table 1).

3.3 | Model building

The best models were selected with the coefficient of determination (R^2) calculated using the 5-fold transformation-out cross-validation procedure [34] implemented in the in-house CIMtools package (https://github.com/cimm-kzn/CIMtools). Transformation-out cross-validation prepares test folds that include structural transformations that are not presented in training folds. This cross-validation strategy provides an unbiased estimation of the predictive performance of the models for novel types of structural transformations.

Building ridge regression models. *Individual* and *conjugated* RR models were implemented using *PyTorch* tensors [27], which enabled the training of RR models on both CPU and GPU. *Individual* RR models have hyperparameter λ, the regularization coefficient, which controls the model complexity. For *individual* models, we tested discrete values of λ between 10^{-10} to 10^5 and found the optimal value using the grid search technique.

Conjugated RR models have hyperparameters c_K, c_A and c_E that balance the prediction error of the log$k$, log$A$ and E_A characteristics. The other two hyperparameters of the *conjugated* model are the regularization coefficients λ_A and λ_E (Figure 4). To optimize the hyperparameters of the RR *conjugated* models, we used the *hyperopt* package [35], which applies advanced optimization algorithms to navigate in the hyperparameters space. The values of coefficients c_K, c_A and c_E were sampled from a continuous space defined between 0 to 1, while the regularization coefficients λ_A and λ_E took discrete values between 10^{-10} to 10^5 (Figure 4). The *hyperopt* algorithm adjusts the hyperparameters by maximizing the value of the objective function which was calculated as an average prediction accuracy of all characteristics: $[R^2(\log k) + R^2(\log A) + R^2(E_A)]/3$. The *hyperopt* algorithm takes the average accuracy and proposes the next combination of possible optimal hyperparameters (Figure 4).

Building neural network models. *Individual*, *multi-task*, and *conjugated* NN models were built with the architectures depicted in Figure 2. In NN *multi-task* and *conjugated* models, the coefficients c_K, c_A, and c_E were automatically adjusted together with neural network weights using the gradient descent algorithm. This means that the trade-off coefficients are optimized dynamically during network training, rather than being fixed as hyperparameters before model training as in RR *conjugated* models. This approach to optimization of the trade-off coefficients in the NN *multi-task* and *conjugated* models significantly reduces the computational resources required for model training and hyperparameters optimization.

4 | RESULTS AND DISCUSSION

4.1 | Performance comparison of single-task, multi-task, and conjugated models

This section reports the results of the performance comparison of *individual*, *Arrhenius-based*, *multi-task*, and *conjugated* models. The prediction accuracy (R^2) of the

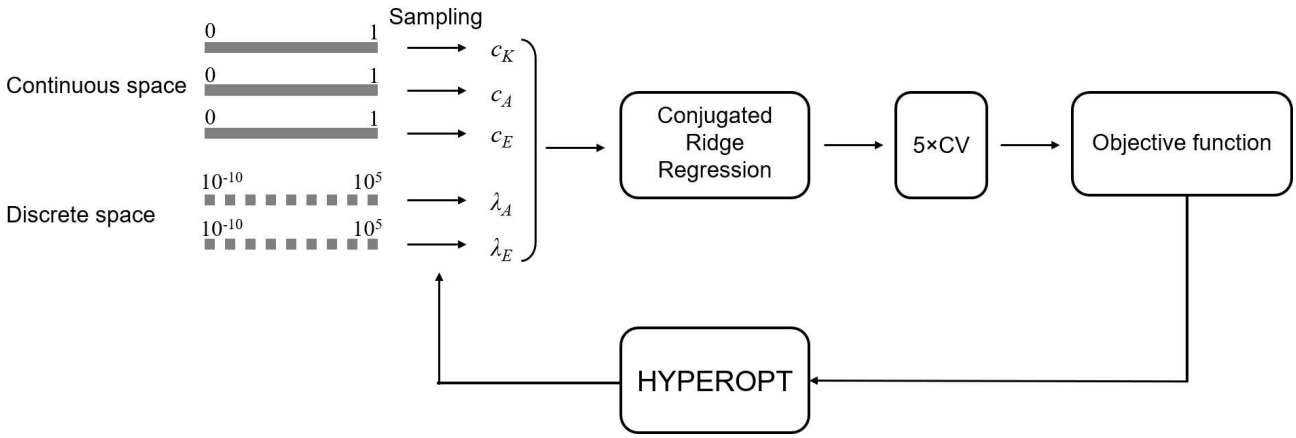

FIGURE 4 The workflow for optimization of hyperparameters of ridge regression *conjugated* models using *hyperopt* package. The trade-off coefficients were sampled from continuous space defined between 0 to 1. The regularization coefficients λ_A and λ_E took values from discrete 10^{-10} to 10^5. *Conjugated* models were built with sampled hyperparameters and evaluated using internal 5-fold cross-validation.

TABLE 2 Predictive performance of individual, Arrhenius-based, multi-task, and conjugated models. RR – Ridge Regression models and NN – Neural Network models.

			R^2 (Test set)		
Model	**Training set**	**Method**	log*k*	log*A*	E_a
Individual model	log*k*	RR	**0.78**	–	–
		NN	0.76	–	–
Individual model	log*A*	RR	–	0.46	–
		NN	–	0.56	–
Individual model	E_a	RR	–	–	**0.91**
		NN	–	–	0.90
Arrhenius-based model	log*A*, E_a	RR	0.27	–	–
		NN	0.35	–	–
Multi-task model	log*k*, log*A*, E_a	NN	0.76	0.48	0.83
Conjugated model	log*k*, log*A*, E_a	RR	0.75	**0.57**	0.90
		NN	0.71	0.56	0.84

models on the external test set is presented in Table 2. For clarity, we discuss NN models only, whereas the results obtained for RR models are available in Table 2 and show similar trends. We tested two single-task approaches for the prediction of log*k*: (1) direct modeling of log*k*, when the *individual* model was built on experimental data on log*k* and (2) *Arrhenius-based model* when first individual models for predicting the log*A* and E_A were built and then used to calculate the prediction of log*k* with the Arrhenius equation. The results demonstrate (Table 2) that the direct predictions of log*k* by the *individual model* are more accurate ($R^2_{Test}=0.76$) than those calculated with the Arrhenius equation in the *Arrhenius-based* model ($R^2_{Test}=0.35$). The prediction accuracy of the *conjugated* model ($R^2_{Test}=0.71$) is close to the *individual* ($R^2_{Test}=0.76$) and *multi-task* model ($R^2_{Test}=0.76$). *Individual* and *Arrhenius-based* models often disagree and provide significantly different predictions of log*k* for the same reaction (Figure S1 in SI). The *conjugated* model predicts log*k*, log*A* and E_A with similar accuracy to the *individual* models, but the predictions exactly agree with the Arrhenius equation (Figure S1 in SI)

Table 2 demonstrates that the RR and NN conjugated models perform similarly. Ridge regression models are easy to build since the optimal regression weights are calculated using analytical expressions. However, more sophisticated optimization of the hyperparameters (trade-off and regularization coefficients) may require a lot of time. On the other hand, the single NN model trains slower than the RR model, but the trade-off coefficients (c_K, c_A and c_E) in the NN model are optimized automatically during model training, which reduces the number of optimized hyperparameters. In addition, the current implementation of RR conjugated models requires a lot of computational resources in the case of large training sets (large sizes of descriptor matrices), while NN models can be trained on large data sets divided into smaller training batches.

4.2 | Building models with limited data

As follows from Table 2, *individual*, *multi-task*, and *conjugated* models perform similarly if the training set is representative. We hypothesized that in *multi-task* and *conjugated* models, abundant data for one modeled characteristic (e.g., *logk*) can compensate for the lack of training data for another characteristic (e.g. *logA* or E_A). In contrast to the standard case, we simulated a scenario in which the training sets for the *logA* or E_A characteristics were significantly reduced and tested the performance of the models under these conditions. We used the same test set of 371 reactions for the model evaluation (Table 2) but varied the size of the training set for *logA* or E_A. For the sake of clarity, only results for NN models are reported (RR models show similar trends).

The initial training set contained 1480 experimental values of *logk*, 1008 values of *logA* and 1120 values of E_A. We gradually reduced the number of log*A* and E_A training data and evaluated the resulting models on the test set. For this purpose, we randomly selected and removed *N*% of training reactions with associated log*A* and E_A from the initial training set and used reduced training sets to build *individual* $F_{Ind}(logA^{reduced})$ and $F_{Ind}(E_A{}^{reduced})$ models. The same reduced training sets on log*A* and E_A, as well as all available training data for log*k*, were used to build the

multi-task $F_{MT}(logk,\ logA^{reduced},\ E_A^{reduced})$ and *conjugated* $F_{Conj}(logk,\ logA^{reduced},\ E_A^{reduced})$ model. The models built on the reduced training sets were then used to predict the log$A$ and E_A for reactions from the test set.

To alleviate the effect of random reduction of the training sets, the above procedure was repeated 20 times, followed by the averaging of obtained R^2 values. Figure 5 reports the average R^2 on the test set at different sizes of the training set on $logA^{reduced}$ and $E_A^{reduced}$. For log$A$ models built on large training sets, *conjugated* learning has no advantages over *single* and *multi-task learning* (Figure 5a). The performance of all models gradually decreases as the log$A$ and E_A training sets were reduced until the models lose their predictive power at extremely small training sets $<6\%$ (<70 training reactions). But in general, conjugated model demonstrates more stable behavior towards extremely small training sets (Figure 5a). Similar behavior is observed in modeling E_A on reduced training sets. When the size of the training set is large (e.g. 1120 training reactions with known E_A, Figure 5b), the *individual* $F_{Ind}(E_A^{reduced})$ ($R^2_{Test}=0.90$) and *multi-task* model $F_{MT}(logk,\ logA^{reduced},\ E_A^{reduced})$ ($R^2_{Test}=0.83$) demonstrate the accuracy comparable with the *conjugated* model $F_{Conj}(logk,\ logA^{reduced},\ E_A^{reduced})$ ($R^2_{Test}=0.84$). However, for significantly reduced E_A training set (11 training reactions corresponding to 1% of the initial set), the *conjugated* models were still predictive ($R^2_{Test}=0.33$), whereas the *individual* ($R^2_{Test}=-0.60$) and *multi-task* ($R^2_{Test}=-0.30$) models failed.

Thus, *conjugated* models can correctly predict a target characteristic of reactions even for a few training instances if data on another characteristic related to the target characteristic by a strict mathematical relationship is available.

4.3 | Modeling the temperature dependence of the reaction rate constant

The dependence of the reaction rate constant on temperature is described by the Arrhenius equation. We were interested in how closely the rate constants predicted by the *individual* and *conjugated* models reproduce this dependence. In building *individual* models, the reaction temperature was a descriptor concatenated with fragment and solvent descriptors. Therefore, the *individual* and *multi-task* model can only capture the statistical relationship between log$k$ and temperature. In this context, we were interested to examine the performance of the models as a function of reaction temperature. For this purpose, we prepared a new temperature test set. The initial test set (Table 1) contained 1 reaction in 1,4-dioxane, 3 reactions in chlorobenzene, 4 reactions in benzene, and 53 reactions in toluene (in total 61 reactions) for which log$A$ and E_A were experimentally determined. We used the experimental log$A$ and E_A values of these 61 reactions to calculate new log$k$ using the Arrhenius equation at "extreme" temperatures, which significantly deviates from the temperature range of the training set. For example, for each cycloaddition reaction in toluene, the log$k$ was calculated for a list of temperatures that starts with the freezing temperature of toluene, changes in increments of 5 K, and ends with the boiling temperature of toluene. Thus, for each cycloaddition reaction in toluene, log$k$ were calculated at 42 new temperatures, including "extreme" temperatures close to the freezing and boiling point of toluene. The same procedure was repeated for reactions in 1,4-dioxane (18 temperatures), chlorobenzene (36 temperatures), and benzene (15 temperatures). As a result, the temperature test set consisted of 61 reactions associated with

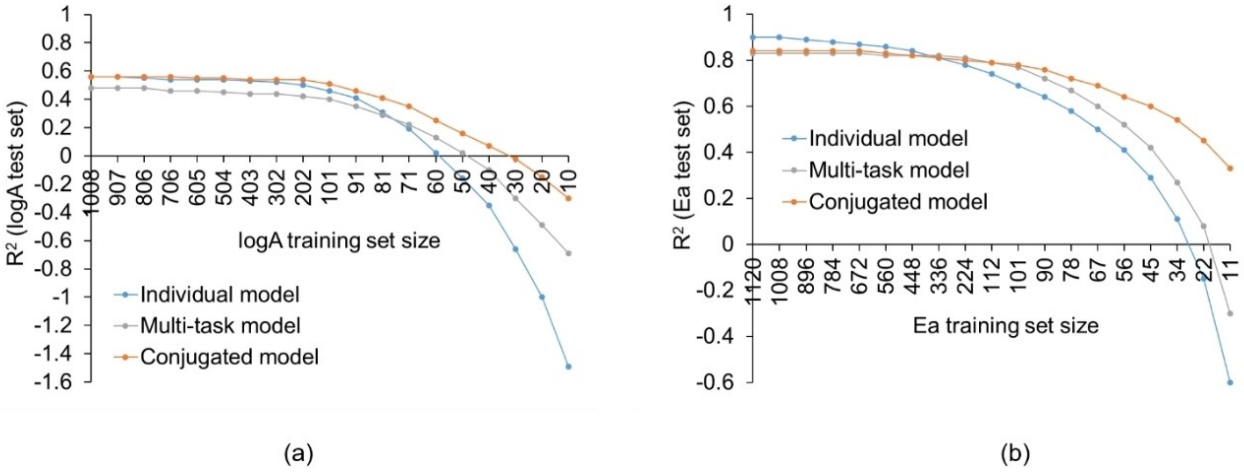

FIGURE 5 Predictive performance of the individual, multi-task, and conjugated neural network model on test set reactions at different sizes log$A$ (a) and E_A (b) training sets.

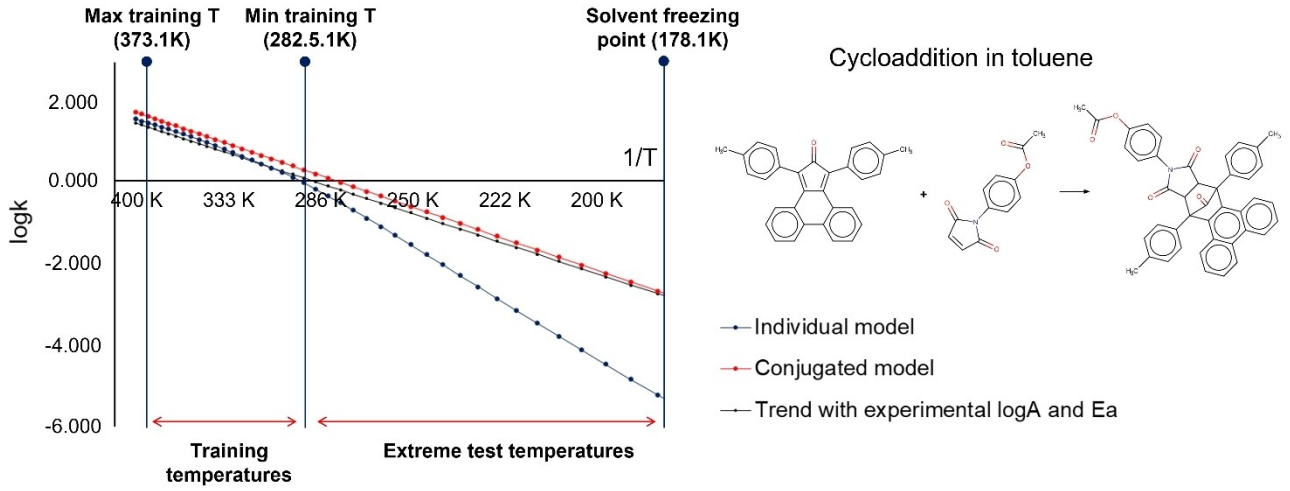

FIGURE 6 Calculated with experimental Arrhenius equation and predicted log*k* with individual and conjugated models for the cycloaddition reaction at different temperatures in toluene.

2412 log*k* values calculated from the Arrhenius equation for new temperatures including "extreme" temperatures; all remaining reactions with experimental temperatures were included in the training set. The lists of new temperatures were used in the log*k* predictions by the NN models. In the *conjugated* model, these temperatures were directly used in predicting the log*k*, while in the *individual* model, they were used as descriptors. Then, predicted with each model log*k* values were compared with calculated with the experimental Arrhenius equation log*k* values.

Each reaction (and associated set of calculated log*k* at new temperatures) of the 61 reactions in the test set was considered as a separate test subset for which R^2 and RMSE were calculated. For six reactions the individual or conjugated (or both) models failed to predict log*k* ($R^2 < 0$), and the reasons for such failure still need to be investigated. The erroneously predicted reactions were removed and analysis was carried out on the remaining 55 reactions, for which obtained R^2 and RMSE values were averaged (R^2 and RMSE for all reactions are reported in Table S1 in Supporting Information). As a result, the *conjugated* model ($R^2 = 0.96$ and RMSE = 0.35) demonstrated higher accuracy of log*k* predictions at new temperatures including extreme temperatures than the *individual* model ($R^2 = 0.88$ and RMSE = 0.62).

To take a closer look at the reasons for this behavior of the models, we selected one of the test cycloaddition reactions in toluene with experimentally measured log$A = 6.62$ and $E_A = 53$ kJ/mol, for which the log*k* values at "extreme" temperatures ranging far away from training temperatures were calculated. The log*k* predicted at "extreme" temperatures by the *individual* and the *conjugated* models were also plotted (Figure 6). We can see (Figure 6) that both models perfectly predict the rate constant at temperatures inside the training temperature range. However, in the range beyond the training temperatures, the log*k* predicted by the individual model significantly deviates from the experimental trend, while the *conjugated* model predicts the log*k* accurately, even at extremely low temperatures close to the freezing point of the solvent. This can be explained by the fact that the *individual* model accounts for only the statistical relationship between the reaction rate constant and the temperature descriptor, whereas the *conjugated* model includes the true temperature dependence in the form of the Arrhenius equation.

5 | CONCLUSION

In this study, the concept of conjugated learning was applied to model kinetic characteristics related by the Arrhenius equation: rate constant log*k*, pre-exponential factor log*A*, and activation energy E_A of cycloaddition reactions. In conjugated QSPR models, the Arrhenius equation was embedded into ridge regression and neural network machine learning algorithms. The conjugated models were compared with individual (single-task) models that were trained independently for each characteristic and multi-task model, where the kinetic characteristics were modeled cooperatively. An equation-based (Arrhenius-based) model was also considered in which the rate constant log*k* is calculated using the Arrhenius equation and predicted by individual models log*A* and E_A.

It was observed that the individual model, which predicts the log*k* directly from reaction descriptors, is more accurate than the Arrhenius-based model, which calculates log*k* using the Arrhenius equation. The predictions

of the log*k* of individual and Arrhenius-based models often disagree, which demonstrates that the standard QSPR models do not always obey the fundamental chemical laws. However, the conjugated model predicts log*k*, log*A* and E_A with similar accuracy to the individual models, but the predicted characteristics exactly comply the Arrhenius equation. Furthermore, the conjugated models are more accurate in predicting log*k* at the wide range of reaction temperatures. In the individual model, the temperature is treated as a descriptor, whereas in the conjugated models, the exact relationship between the rate constant and the temperature is embedded into the model in the form of the Arrhenius equation. To validate the models in new scenarios, a new temperature test set was generated which included log*k* values associated with "extreme" temperatures significantly deviating from the temperature range of the training set. It was demonstrated that the individual model cannot correctly predict the values of log*k* at temperatures that are significantly different from the training data, while the conjugated model correctly predicts log*k* even for the temperatures close to the freezing and boiling points of the reaction solvent.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available in Molecular Informatics at https://doi.org/10.1002/minf.201800077. These data were derived from the following resources available in the public domain: - Molecular Informatics, https://doi.org/10.1002/minf.201800077

ORCID

Timur Madzhidov http://orcid.org/0000-0002-3834-6985
Alexandre Varnek http://orcid.org/0000-0003-1886-925X

REFERENCES

1. W. A. Warr, *Mol. Inform* **2014**, *33*, 469–476.
2. I. I. Baskin, T. I. Madzhidov, I. S. Antipin, A. A. Varnek, *Russ. Chem. Rev.* **2017**, *86*, 1127–1156.
3. T. C. Ho, *Catal. Rev. - Sci. Eng.* **2008**, *50*, 287–378.
4. A. Fernández-Ramos, J. A. Miller, S. J. Klippenstein, D. G. Truhlar, *Chem. Rev.* **2006**, *106*, 4518–4584.
5. C. A. Grambow, L. Pattanaik, W. H. Green, *Sci. Data.* **2020**, *7*, 137.
6. Y. Zhao, D. G. Truhlar, Acc. *Chem. Res.* **2008**, *41*, 157–167.
7. R. A. Friesner, *Proc. Natl. Acad. Sci.* **2005**, *102*, 6648–6653.
8. P.-L. Kang, Z.-P. Liu, *IScience.* **2021**, *24*, 102013.
9. T. I. Madzhidov, A. Rakhimbekova, V. A. Afonina, T. R. Gimadiev, R. N. Mukhametgaleev, R. I. Nugmanov, I. I. Baskin, A. Varnek, *Mendeleev Commun.* **2021**, *31*, 769–780.
10. P. R. Wells, *Chem. Rev.* **1963**, *63*, 171–219.
11. C. Hansch, A. Leo, R. W. Taft, *Chem. Rev.* **1991**, *91*,165–195.
12. K. Jorner, T. Brinck, P.-O. Norrby, D. Buttar, *Chem. Sci.* **2021**, *12*, 1163–1175.
13. R. I. Nugmanov, T. I. Madzhidov, G. R. Khaliullina, I. I. Baskin, I. S. Antipin, A. A. Varnek, J. *Struct. Chem.* **2014**, *55*, 1026–1032.
14. T. I. Madzhidov, P. G. Polishchuk, R. I. Nugmanov, A. V. Bodrov, A. I. Lin, I. I. Baskin, A. A. Varnek, I. S. Antipin, *Russ. J. Org. Chem.* **2014**, *50*, 459–463.
15. P. Polishchuk, T. Madzhidov, T. Gimadiev, A. Bodrov, R. Nugmanov, A. Varnek, *J. Comput. Aided. Mol. Des.* **2017**, *31*, 829–839.
16. T. I. Madzhidov, A. V. Bodrov, T. R. Gimadiev, R. I. Nugmanov, I. S. Antipin, A. A. Varnek, *J. Struct. Chem.* **2015**, *56*, 1227–1234.
17. A. R. Singh, B. A. Rohr, J. A. Gauthier, J. K. Nørskov, *Catal. Letters.* **2019**, *149*, 234–2354.
18. C. A. Grambow, L. Pattanaik, W. H. Green, J. Phys. *Chem. Lett.* **2020**, *11*, 2992–2997.
19. K. Jorner, T. Brinck, P.-O. O. Norrby, D. Buttar, *Chem. Sci.* **2021**, *12*, 1163–1175.
20. A. L. K. Chee, B. L. F. Chin, S. M. X. Goh, Y. H. Chai, A. C. M. Loy, K. W. Cheah, C. L. Yiin, S. S. M. Lock, *J. Energy Inst.* **2023**, *107*, 101194.
21. T. L. Yap, A. C. M. Loy, B. L. F. Chin, J. Y. Lim, H. Alhamzi, Y. H. Chai, C. L. Yiin, K. W. Cheah, M. X. J. Wee, M. K. Lam, others, *J. Environ. Chem. Eng.* **2022**, *10*, 107391.
22. M. Majid, B. L. F. Chin, Z. A. Jawad, Y. H. Chai, M. K. Lam, S. Yusup, K. W. Cheah, *Bioresour. Technol.* **2021**, *329*, 124874.
23. B. L. F. Chin, S. Yusup, A. Al Shoaibi, P. Kannan, C. Srinivasakannan, S. A. Sulaiman, *J. Clean. Prod.* **2014**, *70*, 303–314.
24. M. Glavatskikh, T. Madzhidov, D. Horvath, R. Nugmanov, T. Gimadiev, D. Malakhova, G. Marcou, A. Varnek, *Mol. Inform.* **2019**, *38*, 1800077.
25. D. V. Zankov, T. I. Madzhidov, A. Rakhimbekova, T. R. Gimadiev, R. I. Nugmanov, M. A. Kazymova, I. I. Baskin, A. Varnek, *J. Chem. Inf. Model.* **2019**, *59*, 4569–4576..
26. M. H. J. Gruber, *Routledge* **2017**, 1-632.
27. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, *Adv. Neural Inf. Process. Syst.* **2019**, *32*.

28. A. Varnek, D. Fourches, F. Hoonakker, V. P. Solov'ev, *J. Comput. Aided. Mol. Des.* **2005**, *19*, 693–703..
29. R. I. Nugmanov, R. N. Mukhametgaleev, T. Akhmetshin, T. R. Gimadiev, V. A. Afonina, T. I. Madzhidov, A. Varnek, *J. Chem. Inf. Model.* **2019**, *59*, 2516–2521..
30. G. Marcou, V. P. Solov'ev, D. Horvath, A. Varnek, **2017**, http://infochim.u-strasbg.fr/recherche/Download/.
31. A. Varnek, D. Fourches, D. Horvath, O. Klimchuk, C. Gaudin, P. Vayer, V. Solov'ev, F. Hoonakker, I. Tetko, G. Marcou, *Curr. Comput. Aided-Drug Des.* **2008**, *4*, 191–198..
32. T. Gimadiev, T. Madzhidov, I. Tetko, R. Nugmanov, I. Casciuc, O. Klimchuk, A. Bodrov, P. Polishchuk, I. Antipin, A. Varnek, *Mol. Inform.* **2019**, *38*, 1800104..
33. T. I. Madzhidov, T. R. Gimadiev, D. A. Malakhova, R. I. Nugmanov, I. I. Baskin, I. S. Antipin, A. A. Varnek, *J. Struct. Chem.* **2017**, *58*, 650–656..
34. A. Rakhimbekova, T. N. Akhmetshin, G. I. Minibaeva, R. I. Nugmanov, T. R. Gimadiev, T. I. Madzhidov, I. I. Baskin, A. Varnek, *SAR QSAR Environ. Res.* **2021**, *32*, 207–219..
35. J. Bergstra, D. Yamins, D. D. Cox, *30th Int. Conf. Mach. Learn. ICML* **2013**, 11–123..

How to cite this article: D. Zankov, T. Madzhidov, I. Baskin, A. Varnek, *Molecular Informatics* **2023**, *42*, e202200275. https://doi.org/10.1002/minf.202200275