Using Machine Learning Modeling to Computationally Predict NMR Coupling Constants

Teresa Datta
Department of Chemistry and Chemical Biology,
*Cornell University, Ithaca, New York 14853-1301, USA**
(Dated: May 10, 2020)

This thesis describes the approach, development, and accuracy of a model designed to predict scalar coupling constants between atoms in molecules. Large scale datasets provided by CHAMPS, the Chemistry and Mathematics in Phase Space Project, relating to the structure, dipole moments, magnetic shielding, and potential energy for 130,789 distinct molecules were analyzed. Although the scalar coupling constant is known for all atom pairs in the listed molecules, the prediction power of supervised regression learning models were tested by dividing the dataset into Training and Testing Sets, fitting the models using the training data, and measuring predictive power. Among 10 implemented machine learning algorithms, the highest prediction accuracy with a mean absolute error of 0.364 Hz was achieved by LightGBM. The derived feature scores show the importance of spacial and Mulliken charge information in the model's decision-making process. These findings help establish the importance of further development in the intersection of data science and chemistry.

I. INTRODUCTION

Accurate analysis and prediction of NMR spectra have a vital role in structure validation and discovery [1]. Along with resonance and chemical shift, coupling constants are integral in forecasting spectral information as they measure the interaction between pairs of atoms and determine spacing between peaks on spectra. Coupling constants are also utilized in determining regiochemistry of molecular substituents [2]. Computational predictions of NMR spectra can also be used to train chemometric models. Gathered insights can then be implemented in laboratory experiments [3].

Although current quantum mechanical methodologies using density functionals and basis sets to determine scalar coupling constants are highly accurate, they are computationally expensive and time-consuming [4]. In Bally et al, protocol B3LYP/cc-pVTZ performed the best on a study of 165 coupling constants with an root mean square (RMS) deviation of less than 0.4 Hz, but required significant CPU time [5]. Thus, there exists high demand for cost-effective and efficient models [6].

Conversely, many NMR prediction software currently on the market emphasize coverage, speed, and utility, but only by sacrificing overall accuracy [7]. These NMR predictors include ChemNMR (in ChemDraw Ultra, developed by Upstream Solutions [8]), ACD/NMR (developed by Advanced Chemistry Development [9]), and the predictor in the MestReNova software (Modgraph NMRPredict [10]). In Lehtivarjo et al, ACD/NMR showed the highest coverage and accuracy of these software with a RMS error of 2.12 Hz on a test set of 99 molecules containing 255 coupling constants [5] [7]. Most distinctly, these web-based applications require the user to input a hypothesized chemical structure. The coupling constants are generated by breaking the structure down into substructures to generate base values, and then applying additivity rules [8].

Recently, with the rise of Big Data technologies, machine learning techniques have been applied to predict a variety of spectral properties. Web-interfaces have been developed to predict both ^{1}H and ^{13}C chemical shifts [11]. ^{1}H NMR scalar coupling constants have previously been predicted though associative neural networks, however, these models were trained using chemical shift information [1].

Here, I report the results of applying multiple methods to estimate proton NMR coupling constants by working with a dataset of over 4 million coupling constants. Using LightGBM, a gradient boosting framework that builds off of decision tree learning algorithms [12], yielded the highest predictive accuracy. A mean absolute error of 0.364 Hz was calculated. By rephrasing the problem of coupling constant estimations as a supervised learning regression problem- allowing the model to be trained on previously verified coupling constants in order to predict a continuous decimal value- a variety of computer science techniques were able to be applied.

This strategy is unique because of the limited information required for model generation. Unlike previously mentioned prediction software, visual structural information (i.e. what types of bonds are present between atoms) is not utilized in forecasting. Rather, this model was trained solely using atomic properties, 3-D atomic position coordinates, and chemical property information. This machine learning approach also results in a strategy that is less computationally expensive than previous quantum mechanical techniques.

* Email:td334@cornell.edu

II. EXPERIMENTAL

A. Data

The datasets utilized were provided by CHAMPS, the Chemistry and Mathematics in Phase Space Project, and utilized under a CC BY-SA license.

Information for 85,012 distinct organic molecules is provided in the training data. Data for a wide range of molecules is included. The average number of atomic pairs for which coupling constants were provided per molecule is 54.80, for a total of 4,659,076 data points. A total of 8 coupling types are explored: ${}^1J_{HC}$,${}^2J_{HH}$,${}^1J_{HN}$,${}^2J_{HN}$,${}^2J_{HC}$,${}^3J_{HH}$,${}^3J_{HC}$,${}^3J_{HN}$. The distribution of these coupling types can be seen in Figure 2. ${}^3J_{HC}$ has the greatest number of data points. The molecules in the dataset are composed of 5 unique elements: carbon, hydrogen, nitrogen, oxygen, and fluorine, although not all elements are featured in every molecule. Figure 1 shows the distribution of coupling constants present in the data via a 1000 bin histogram. Coupling constants span from -44.761 Hz to 207.709 Hz, with a large proportion of data present in the -5 Hz to 12 Hz region.

Information on dipole moment vectors and potential energies of each molecule as well as magnetic shielding tensors, Mulliken charges, and 3-D coordinates for each atom in each molecule were also provided and utilized as additional features for model training.

B. Machine Learning Models

Ten different machine learning algorithms were utilized: DummyRegressor (DM) [13], LightGBM (LGBM) [14], Linear Regression (LR) [15], k-Nearest Neighbors (KNN)[16], Support Vector Machine (SVM) [17], Decision Tree (DT) [18], Random Forest (RF) [19], Adaptive Boosting (ADA) [20], Gradient Boosting (GB) [21], and Multilayer Perceptron (MLP) [22]. The Dummy Regressor represents the most generic estimate: the mean coupling constant per type, and is meant to act as a baseline to compare the results of the other models to. All models were programmed using Python with the scikit-learn package [23]. These 10 models were chosen because they represent a diverse array of learning algorithms that are popularly used in supervised regression problems.

All models were trained to predict coupling constants by randomly dividing the data into training and validation sets such that 20% of molecules were placed in the validation set. The models were then trained using a 5-fold cross-validation strategy. The 80% of molecules in the training data were randomly divided into 5 groups, with 4 of these sets used for training and 1 for testing. This technique allowed for a defense against overfitting in order to produce more accurate predictions. Error was calculated by using mean absolute error of validation and predicted values.

III. APPROACH

The process used to create the analyzed models involved a series of steps. First, data sources were compiled and cleaned. Because these wrangled large scale datasets have over 4 million rows, exploratory data analysis was performed in order to better understand the presented information available. The highlights of these findings are included in the Data Subsection. Further evaluation of accessible tools and literature reviews stimulated the ideation process for how to approach this problem.

One major factor influencing the value of the coupling constant is the coupling type. Eight coupling types are present in the data. Thus, the training data was grouped by coupling type so that each could be treated as a separate learning problem for the model.

The scalar coupling constant, J, is a measure of the interaction between pairs of atoms. As such, it is largely controlled by the geometry of the orbitals involved between coupling nuclei [24]. One major role of NMR spectra is the elucidation of chemical structure [1]. Thus, neither bond structure, atomic composition, nor bonds present in each molecule are included in the training data. However, the 3-D atomic coordinates and element information are. These are not reliable or comprehensive enough to be used to model the atomic structure, but by using common distance formulas, the absolute distances between atoms can be computed to form a distance matrix, and these values can be used as features.

The produced distance matrix would yield n^2 new features, where n is the number of atoms present in each molecule. Adding this many new features to the training data not only increases the time and space complexity quadratically, but it also increases the possibility of the model overfitting to the training data. Overfitting occurs when the model relies too heavily on the training data and produces an overly complex model that does not perform as well when exposed to new data. As such, by following Occam's Razor via encouraging simplicity [25] and in order to meet the limited computing power of my personal technological devices, these distance matrices had to be truncated and condensed.

The distance features were calculated in order to efficiently maximize information about the atomic environment of the coupled pair in question. Each distance value is labeled d_n_m where n and m correspond to atomic indices within the molecule. Indices a and b correspond to the atomic pair for which the coupling constant is being predicted for. Then, using the computed center of the atomic pair, numerical indices correspond to atoms closest to the pair center in increasing order, i.e. index 1 corresponds to the atom closest to the coupled pair center, 2 corresponds to the second closest atom, etc.

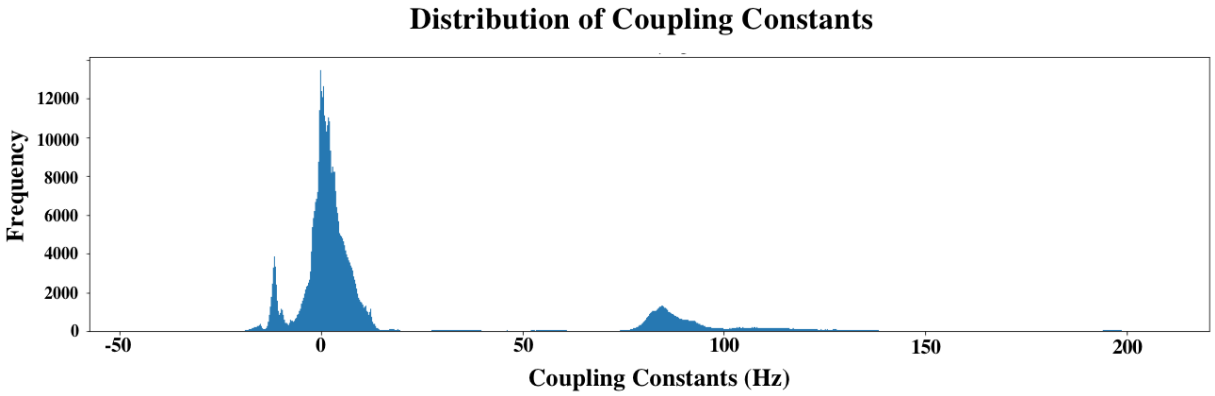

FIG. 1. Histogram distribution of scalar coupling constants present in training data separated into 1000 bins. Height of bar corresponds to frequency. Coupling constants spanning from -44.761 to 207.709 Hz were present.

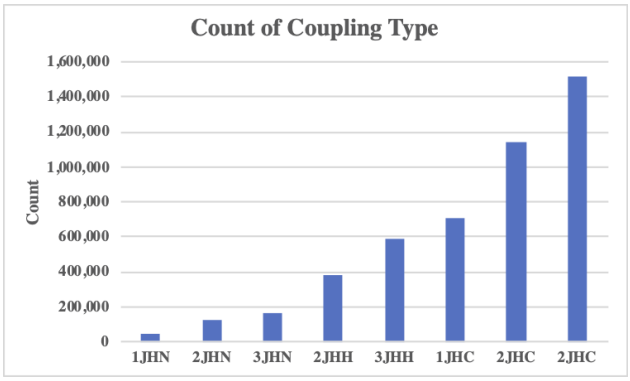

FIG. 2. Bar chart distribution of the eight types of coupling constants present in dataset.

ML Model	MAE
Dummy Model	9.717
LightGBM	0.512
Linear Regression	1.612
k-Nearest Neighbor	0.883
Support Vector Machine	2.444
Decision Tree	0.633
Random Forest	1.65
Adaptive Boosting	1.99
Gradient Boosting	1.021
Multilayer Perceptron	1.141

FIG. 3. Test accuracy of initial iterations of 10 machine learning models as measured by MAE: mean absolute error.

Atomic properties, atomic number and electronegativity, for featured indices are also included as descriptors in the training data [26].

The final collection of features, or atomic descriptors, were selected in order to maximize both information gain by the model and running time efficiency. This was specifically accomplished through removal of descriptor variables with the least impact on error scores or least measured importance. These descriptors were systematically randomized amongst the training set, and the transformed data was utilized to train the model in order to draft new predictions. The feature corresponding to the new error that was least impacted or lowest was thus removed [1], and this process was repeated to yield the final collection of features.

Utilizing this initial set of features relating purely to structural distance data and atomic information, all chosen supervised machine learning regression models were trained using the same randomized train-test split. The prediction accuracy of these models were measured through mean absolute error(MAE).

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{1}$$

As seen in Equation 1, MAE computes the average residual or absolute difference between predicted and actual target values. This is a linear score, so it does not penalize outliers as greatly as other regression error formulas [27]. This is beneficial since our dataset is large and noisy with a diverse collection of molecules, and models should not be too heavily penalized for predicting a few skewed outliers.

Figure 3 shows the testing accuracy of the 10 trained models using purely structural and atomic property data. As previously stated, the Dummy Model represents the simplest regressor and acts as a baseline to compare the other 9 models to. This dummy baseline error score of 9.717 gives insight into how hard our learning problem was. All nine other models achieved mean absolute errors of less than 2, significantly lower than the results of the Dummy Model. Figure 4 graphically displays the mean absolute errors of the 9 trained models. From this chart, it is initially clear that LightGBM and Decision Tree

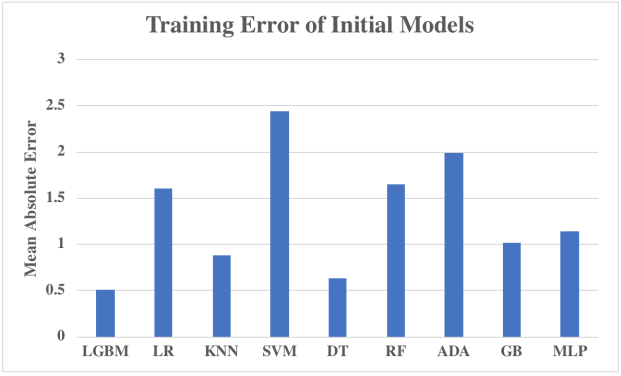

FIG. 4. Bar chart comparing accuracy of various initial iterations of machine learning models as measured by MAE: mean absolute error.

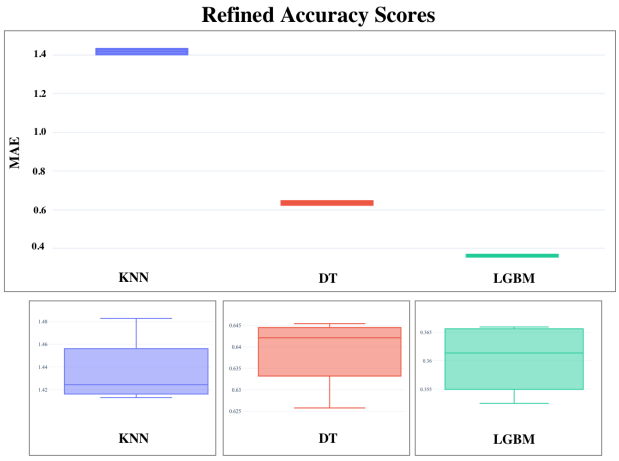

FIG. 5. Comparison of mean absolute errors of k-Nearest Neighbors (KNN, blue), Decision Tree (DT, red), and LightGBM(LGBM, green) models using updated features. Magnified boxplots of MAE for four prediction iterations of each model are featured below.

Regressors appear to be the most promising as they have the lowest MAE scores. Because LightGBM builds off of a decision tree framework [14], this might help explain the similarity in these scores. The k-Nearest Neighbor based model has the third lowest MAE score.

Different machine learning models use different methodologies and algorithms to establish prediction criteria. Understanding these algorithms is thus useful in order to understand why prediction results differ and why certain models might be more appropriate for solving this learning problem [6]. Conversely, seeing the error scores for all models gives more concrete evidence favoring certain models over others. The Random Forest (RF) algorithm also takes advantage of a multiple-decision-tree framework, but did not perform as well. Further fine tuning of hyperparameters, which requires substantial experience and ability, for each of these models can increase prediction accuracy. For this first iteration, default parameters were utilized. The 3 top-performing models- LightGBM, Decision Tree, and k-Nearest Neighbor, were further iterated and compared in detail.

The feature space was expanded to merge the other available data points. Specifically, the current training data was mapped to Mulliken charges for the specific atoms involved in the coupled atomic pair, along with dipole vector coordinates and the potential energies of the molecules. The 3 previously chosen models were then retrained using adjusted hyperparameters and this expanded feature space. Four iterations of randomized train-test splits were used, and the resulting box plot of mean absolute errors for these 3 models is shown in Figure 5.

As seen in Figure 5, LightGBM performs the best, followed by Decision Tree, followed by k-Nearest Neighbor. Interestingly, the accuracy of the KNN model decreased from the initial iteration, while the accuracy of the other two models increased. Because the resulting error scores for each training split have very low variance, the box plots are not easily interpreted when placed in unison on the same scale. The zoomed in box plots for these models is thus shown at the bottom of Figure 5. The tight range of these error scores provides evidence that the model is not overfitting the training data, and that the high performance results are consistent.

IV. DISCUSSION

Machine learning models are often synonymous with "black boxes" of computation, meaning that data is inputted into a model, and predictions are outputted, but explanations as to how these predictions were made are difficult to determine [28]. However, because the LightGBM is based on a decision tree framework, this box becomes more translucent as we are able to access both the decision path and feature importance values.

In order to begin the analysis of the effectiveness and predicting power of our LightGBM model, the measured target values compared to the predicted target values for the subset of the 34,944 testing data points corresponding to the ${}^1J_{HN}$ coupling type are shown in Figure 6. The line of perfect prediction is also traced on this figure, and most points are clustered around this line. There are, however, a few noticeable outliers that are visibly separated from the rest of the plot points. In order to gain more information as to the nature of these outliers, the residuals, or the vertical distance between the measured and predicted values, were plotted in a violin plot seen in Figure 7. It shows that the majority of residuals have less than an absolute value of 2. Further analysis determined that the median of the residuals was 0.003124 with a standard deviation of 0.7877. This standard deviation metric is also known as the root mean square error. There are comparatively few outliers, but the residual values go

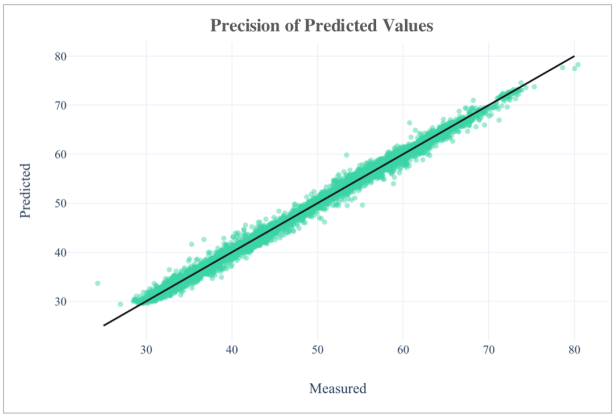

FIG. 6. Scatter plot comparing predicted target values with actual target values of $^1J_{HN}$ subset of training set. Black line represents perfect predictions.

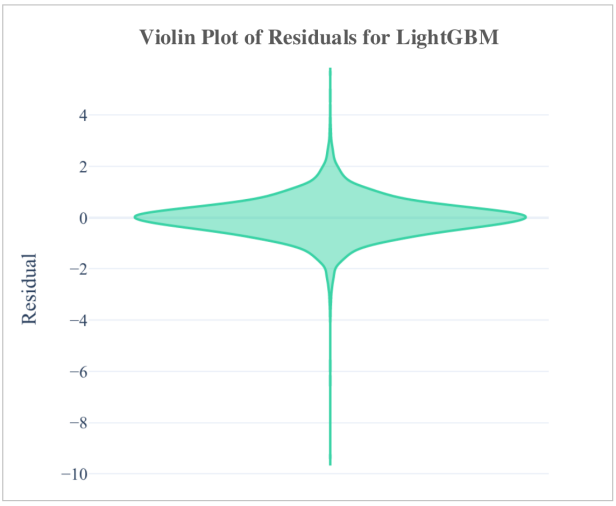

FIG. 7. Violin plot of differences in predicted target values versus actual target values of $^1J_{HN}$ subset of training set.

up to an absolute value of 9.325. This implies that there are a small percentage of training points for which the LightGBM model is predicting dramatically incorrectly.

The scores displaying the relative importance (as determined via information gain) of the features for the LightGBM model are shown in Figure 8. It shows that distance features and Mulliken charges are the dominant factors used to predict scalar coupling constants. Distance descriptors are labeled in the format of *d_n_m* such that n and m correspond to atomic indices within the molecule based on distance from the examined pair. The indices *a* and *b* correspond to the coupled pair. Thus, *d_a_b*, the most important feature corresponds to the distance between the examined pair. It makes sense as to why this feature was relied on the most by the model since this distance could be a good indicator or heuristic for the types and arrangement of bonds present

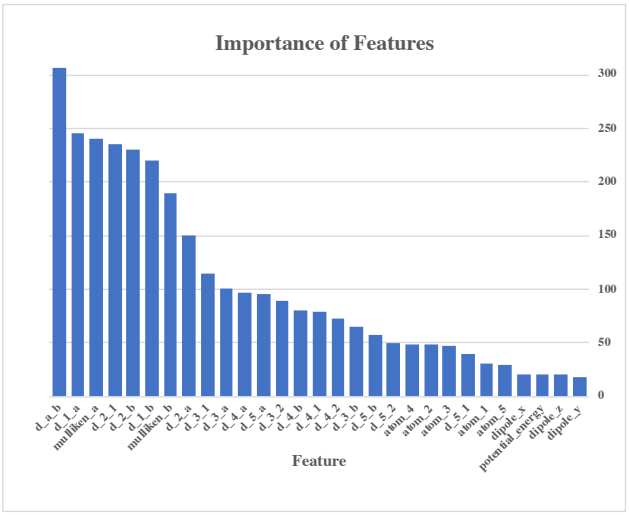

FIG. 8. Importance scores of features derived from LightGBM model.

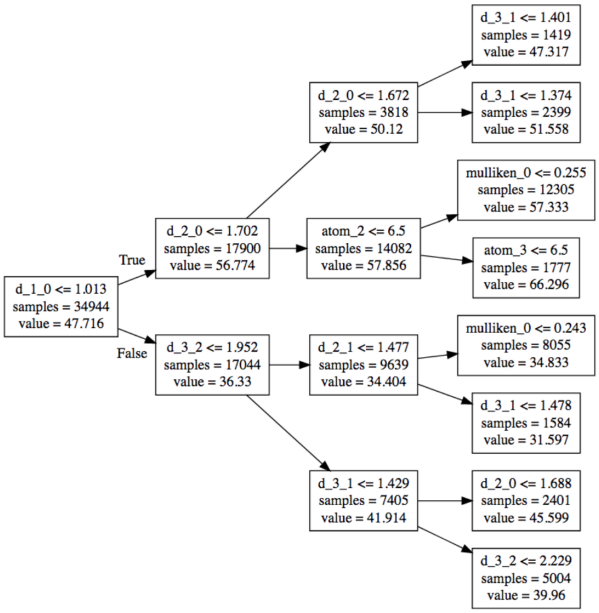

FIG. 9. Visualization of fragment of decision tree from LightGBM model showing decision nodes and fragment sizes.

between a pair of atoms. Numerical atomic indices correspond to atoms closest to the pair's center in increasing order. Features labeled *atom_n* represent the atomic information- atomic number and electronegativity for the corresponding index as previously described [26]. Interestingly, the information about the surrounding atomic environment did not score as highly in importance as the distance factors. The dipole moment information and potential energy values were also not as heavily relied on for predictive decisions.

In order to gain more insight into the decision making process, a portion of the decision tree was derived from

the LightGBM model. This is shown in Figure 9. The full decision tree framework is too large and complex to be computed or to be shown here, but this snippet again relates to the $^1J_{HN}$ coupling type previously explored.

It is important to note that just because the high-performing LightGBM model placed importance and decision-making power on these distance and mulliken charge features, this correlation does not necessarily imply a direct causation between such features and scalar coupling constants. Rather, from these decision trees and feature analysis studies, one can extract hypothesis for the factors affecting scalar coupling constants in molecules.

V. CONCLUSION

In summary, this work highlights a LightGBM based machine learning model used to predict scalar coupling constants. This model had the lowest MAE score of ten examined machine learning algorithms with a mean absolute error of 0.364 Hz. The derived feature scores show the importance of spacial and Mulliken charge information in the model's decision-making process. With additional resources, being able to directly compare this model's results with web-based prediction software and quantum mechanical methods by utilizing the same validation set of molecules and error calculation techniques as found in Bally et al. [5] would have been insightful. These findings help establish the importance of further development in the intersection of data science and chemistry. The extension of these models to predicting physical and computation properties of chemical materials would potentially accelerate and reduce the cost of research and development.

[1] Y. Binev, M. M. B. Marques, and J. A. de Sousa, Prediction of 1h NMR coupling constants with associative neural networks trained for chemical shifts, *Journal of Chemical Information and Modeling*, **2007**, *47*, 2089–2097, [doi:10.1021/ci700172n].

[2] R. J. Deeth, A. Smith, and J. M. Brown, Electronic control of the regiochemistry in palladium-phosphine catalyzed intermolecular heck reactions, *Journal of the American Chemical Society*, **2004**, *126*, 7144–7151, [doi:10.1021/ja0315098].

[3] D. A. R. S. Latino and J. A. de Sousa, Linking databases of chemical reactions to NMR data: an exploration of1h NMR-based reaction classification, *Analytical Chemistry*, **2007**, *79*, 854–862, [doi:10.1021/ac060979s].

[4] B. Wang, X. He, and K. M. Merz, Quantum mechanical study of vicinal j spin–spin coupling constants for the protein backbone, *Journal of Chemical Theory and Computation*, **2013**, *9*, 4653–4659, [doi:10.1021/ct400631b].

[5] T. Bally and P. R. Rablen, Quantum-chemical simulation of1h NMR spectra. 2.† comparison of DFT-based procedures for computing proton–proton coupling constants in organic molecules, *The Journal of Organic Chemistry*, **2011**, *76*, 4818–4830, [doi:10.1021/jo200513q].

[6] Y. Xie, C. Zhang, X. Hu, C. Zhang, S. P. Kelley, J. L. Atwood, and J. Lin, Machine learning assisted synthesis of metal–organic nanocapsules, *Journal of the American Chemical Society*, **2019**, *142*, 1475–1481, [doi:10.1021/jacs.9b11569].

[7] J. Lehtivarjo, M. Niemitz, and S.-P. Korhonen, Universal j-coupling prediction, *Journal of Chemical Information and Modeling*, **2014**, *54*, 810–817, [doi:10.1021/ci500057f].

[8] ChemNMR http://www.upstream.ch/products/chemnmr.html.

[9] ACD/Labs ACD/NMR Predictors http://www.acdlabs.com/products/adh/nmr/nmr$_p$red/.

[10] Mestrelab Mnova NMR Predict Desktop http://mestrelab.com/software/mnova-nmrpredict-desktop/.

[11] A. Loss, R. Stenutz, E. Schwarzer, and C.-W. von der Lieth, GlyNest and CASPER: two independent approaches to estimate 1h and 13c NMR shifts of glycans available through a common web-interface, *Nucleic Acids Research*, **2006**, *34*, W733–W737, [doi:10.1093/nar/gkl265].

[12] LightGBM documentation https://lightgbm.readthedocs.io/en/latest/.

[13] P. S. L. Yip and E. W. K. Tsang, Interpreting dummy variables and their interaction effects in strategy research, *Strategic Organization*, **2007**, *5*, 13–30, [doi:10.1177/1476127006073512].

[14] G. Ke, Q. Meng, T. Finely, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, *NIP*, **2017**.

[15] H. Huang, Z. Xu, X. Shao, D. Wismeijer, P. Sun, J. Wang, and G. Wu, Multivariate linear regression analysis to identify general factors for quantitative predictions of implant stability quotient values, *PLOS ONE*, **2017**, *12*, e0187010, [doi:10.1371/journal.pone.0187010].

[16] T. Cover and P. Hart, Nearest neighbor pattern classification, *IEEE Transactions on Information Theory*, **1967**, *13*, 21–27, [doi:10.1109/tit.1967.1053964].

[17] C. J. Burges, *Data Mining and Knowledge Discovery*, **1998**, *2*, 121–167, [doi:10.1023/a:1009715923555].

[18] Y. Koch, T. Wolf, P. K. Sorger, R. Eils, and B. Brors, Decision-tree based model analysis for efficient identification of parameter relations leading to different signaling states, *PLoS ONE*, **2013**, *8*, e82593, [doi:10.1371/journal.pone.0082593].

[19] T. Hengl, M. Nussbaum, M. N. Wright, G. B. Heuvelink, and B. Gräler, Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables, *PeerJ*, **2018**, *6*, e5518, [doi:10.7717/peerj.5518].

[20] R. Wang, AdaBoost for feature selection, classification and its relation with SVM, a review, *Physics Procedia*, **2012**, *25*, 800–807, [doi:10.1016/j.phpro.2012.03.160].

[21] C. Zhou, H. Yu, Y. Ding, F. Guo, and X.-J. Gong, Multi-scale encoding of amino acid sequences for predicting protein interactions using gradient boosting decision tree, *PLOS ONE*, **2017**, *12*, e0181426, [doi:10.1371/journal.pone.0181426].

[22] W. Castro, J. Oblitas, R. Santa-Cruz, and H. Avila-George, Multilayer perceptron architecture optimization using parallel computing techniques, *PLOS ONE*, **2017**, *12*, e0189369, [doi:10.1371/journal.pone.0189369].

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Édouard Duchesnay, Scikit-learn: Machine learning in python, *Journal of Machine Learning Research*, **2011**, *12*, 2825–2830, [url].

[24] J. Yan, A. D. Kline, H. Mo, M. J. Shapiro, and E. R. Zartler, The absolute sign ofJ coupling constants determined using the order matrix calculation, *Magnetic Resonance in Chemistry*, **2004**, *42*, 962–967, [doi:10.1002/mrc.1418].

[25] M. Pacer and T. Lombrozo, Ockham's razor cuts to the root: Simplicity in causal explanation., *Journal of Experimental Psychology: General*, **2017**, *146*, 1761–1780, [doi:10.1037/xge0000318].

[26] Data Table simple features https://www.kaggle.com/titericz/giba-r-data-table-simple-features-1-17-lb, https://www.kaggle.com/criskiev/distance-is-all-you-need-lb-1-481.

[27] A new typology design of performance metrics to measure errors in machine learning regression algorithms, *Interdisciplinary Journal of Information, Knowledge, and Management*, **2019**, *14*, 045–076, [doi:10.28945/4184].

[28] V. Buhrmester, D. Muench, and M. Arens, Analysis of explainers of black box deep neural networks for computer vision: A survey, **2019**.