



Characterization of Quantum Derived Electronic Properties of Molecules: A Computational Intelligence Approach

Julio J. Valdés^(✉)  and Alain B. Tchagang 

Digital Technologies Research Centre, National Research Council Canada,
M50, 1200 Montreal Rd., Ottawa K1A0R6, Canada
{julio.valdes,alain.tchagang}@nrc-cnrc.gc.ca

Abstract. The availability of BIG molecular databases derived from quantum mechanics computations represent an opportunity for computational intelligence practitioners to develop new tools with same accuracy but much lower computational complexity compared to the costly Schrödinger equation. In this study, unsupervised and supervised learning methods are applied to investigate the internal structure of the data and to learn the mapping between the atomic coordinates of molecules and their properties. Low dimensional spaces revealed a well defined clustering structure as defined by the measures used for comparing molecules based their atom distributions and chemical composition. Supervised learning techniques were applied on the original predictor variables, as well as on a subset of selected variables found using evolutionary algorithms guided by residual variance analysis (Gamma Test). Black and white box modeling approaches were used (random forests, neural networks and model trees and adaptive regression respectively). All of them delivered good performance, error and correlation-wise, with neural networks producing the best results. In particular white box techniques obtained explicit functional dependencies, some of them achieving considerably reduction of the feature set and expressed as simple models.

Keywords: Computational intelligence · Quantum mechanics · Molecules · 3D visualization · Random forests · Neural networks · Model trees · Multivariate adaptive regression · Black box models · White box models

1 Introduction

Computational quantum mechanics derived from first principle has traditionally been used for the discovery and design of de-novo molecules and for the study of their structures and electronic properties [1]. More recently, the availability of huge molecular databases derived from quantum mechanics computations has given rise to new methods based machine learning [2–4]. These quantum mechanics machine learning models have shown great promises, approaching the same

© Crown 2019

I. V. Tetko et al. (Eds.): ICANN 2019, LNCS 11731, pp. 771–782, 2019.

https://doi.org/10.1007/978-3-030-30493-5_72

accuracy as first principle quantum mechanics computations at a much lower computational cost.

There are two main models in machine learning: discriminative (forward) and generative (inverse). In the context of quantum mechanics and discriminative learning which is the aim of this study, the goal is to learn a mapping from molecule x to a property y . In general, given a dataset [$x_i \rightarrow y_i$, with $i = 1$ to N] that consists of N molecules (x_i) with their associated properties (y_i), the discriminative model will learn a mapping from $x_i \rightarrow y_i$ and use that mapping to predict new molecules. Several approaches have been explored in the literature to tackle these problems [4].

In these approaches, observations are described in terms of collections of variables/attributes, having several kinds of mutual dependencies, redundancies and noise. However, such a description may affect performance statistical and machine learning procedures because of the curse of dimensionality. Often the data concentrate in low dimension nonlinear manifolds, embedded within the high dimensional space in which the data is represented, either using an instance-attribute (the present case) or a dissimilarity representation. The implication is that in fact the data is not often really high dimensional. The dimension of those manifolds is considered to be the intrinsic dimension and usually it is much smaller than that of the original data representation space. Learning and uncovering these manifolds is important and useful for understanding the internal structure of the data, as well as for improving the performance of data analytic methods like clustering, classification and regression. In this study, we explored unsupervised and supervised learning approaches to map the atomic coordinates of the molecules to their electronics properties as follows. (a) From the atomic coordinate of each molecule, its Coulomb matrix is computed [2]. (b) Rows and columns of these matrices are sorted in decreasing order according to their column norm. This gives rise to an $N \times M$ feature matrix, where N is the total number of molecules and M the number of atoms that make the largest molecule in the set. (c) The Gamma test is performed to estimate the level of noise in the data. (d) M5 Model trees, Random Forests, adaptive regression and Neural networks are used to learn a mapping from the feature matrix to the electronics properties of the molecules with correlation coefficient close to 0.996.

The rest of this paper is organized as follows. In Sect. 2 we described the molecular dataset used in this study. Section 3 presents the machine learning techniques. Section 4 presents the experimental settings and the results obtained while we conclude in Sect. 5.

2 Molecules

The QM7 dataset used in this study is a subset of the GDB-13 dataset and was downloaded from [2]. This set consists of 7102 small organic molecules and their associated atomization energy. Initial Cartesian coordinates were generated and subsequently relaxed using the Universal Force Field106 as implemented in OpenBabel107 (version 2.3.2). Structures were further relaxed and self-consistent

field energies calculated at the density functional level of theory (DFT) using the Perdew-Burke-Ernzerhof (PBE0) functional with def2-TZVP basis set as implemented in Gaussian (version 09 rev. D.01). Atomization energies were then obtained by subtracting free atom energies computed in the same fashion. More information relative to this dataset can be obtained at [2].

2.1 Coulomb Matrix

The inputs to our machine learning models are the same descriptors that also enter the Schrodinger equations i.e. the nuclear charges Z_i and the atomic positions R_i . Our machine learning model, instead of finding the wavefunction which maps the system’s Hamiltonian to its energy, it directly learns a mapping from the system to energy based on examples given for training. The Coulomb matrix used in this case is directly obtained from Z_i and R_i .

$$Z_{ij} = \begin{cases} 0.5Z_i^{2.4} & i = j \\ \frac{Z_i Z_j}{|R_i - R_j|} & i \neq j \end{cases} \quad (1)$$

Z_i is the atomic number or nuclear charge of atom i , and R_i is its position in atomic units. The Coulomb matrix M is symmetric and has as many rows and columns as the number of atoms in the molecule.

While the Coulomb matrix is invariant to translation and rotation of the molecule, it is not invariant to re-indexing of its atoms. One remedy is to sort the columns and rows of the Coulomb matrices by descending order relative to their norm 2 [2]. That is, for each molecule in the dataset, compute its Coulomb matrix. Pad each matrix to the right and bottom with zeros so they all have the same size that is 23×23 , which is the maximum number of atoms per molecule in the QM7 dataset. Compute the norm-2 of each molecules and sort rows and columns in descending order. Given that the Coulomb matrix is symmetrical, only the lower triangular part is kept. Finally they are unfolded into 1D vector representation of the molecule. For the 7102 QM7 molecules the representation has a matrix of 7102×276 feature, where each row represents the signature of a molecule. This matrix was extended by including five extra features given by the chemical composition of the molecule with respect to the number of atoms of Carbon, Hydrogen, Nitrogen, and Sulfur. Both matrices were converted to z-scores (column-wise) by subtracting the mean and dividing by the standard deviation. The final data matrix was composed of 7102×282 features (281 predictors and the target property: Atomization energy).

3 Machine Learning Techniques

3.1 Unsupervised Analysis and Data Exploration

Low Dimensional Spaces for Data Exploration. It is possible to create spaces for data exploration and visualization by computing low dimensional spaces that preserve chosen properties of the original dissimilarity matrix

describing the objects [17]. Many techniques have been developed based on different principles. Among them are the Sammon nonlinear mapping and the t-distributed Stochastic Neighbor Embedding (t-SNE).

Sammon’s nonlinear mapping [15] transforms vectors of two spaces of different dimensions ($D > m$) by means of a transformation like $\varphi : \mathbb{R}^D \rightarrow \mathbb{R}^m$ which maps vectors $\mathbf{x} \in \mathbb{R}^D$ to vectors $\mathbf{y} \in \mathbb{R}^m$, $\mathbf{y} = \varphi(\mathbf{x})$. Sammon error = $\frac{1}{\sum_{i < j} \delta_{ij}} \sum_{i < j} \frac{(\delta_{ij} - d(\mathbf{y}_i, \mathbf{y}_j))^2}{\delta_{ij}}$, where typically d is an Euclidean distance in \mathbb{R}^m . The weight term δ_{ij}^{-1} gives more importance to the preservation of smaller distances rather than larger ones and is determined by the dissimilarity distribution in the data space. Moreover, they are fixed, which is referred to as lack of plasticity.

t-SNE is an enhancement of SNE [10], where the mapping from higher dimensional space to lower dimensional space is based on the consideration of the similarity of conditional probabilities between datapoints. A conditional probability $p_j|i$ is the probability of datapoint x_i to have x_j as a neighbor based on a Gaussian distribution $p_j|i = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}$, where σ_i^2 is the variance of datapoint x_i and k is a perplexity parameter related to selected local neighbors size. For the lower dimensional space, SNE utilizes conditional probabilities $q_j|i$ of datapoints x_i based on another Gaussian distribution. The goal is to minimize the difference between the probability distributions of the two spaces, expressed as the sum of Kullback-Leibler divergences: $C = \sum_i \sum_j p_j|i \log \frac{p_j|i}{q_j|i}$. One drawback of SNE, is the low cost when representing widely separated points. t-SNE applies a symmetric cost function and uses the Student’s t-distribution in the target space, which has a heavier tail [18]. These modifications represent a notable improvement.

3.2 Supervised Analysis

Gamma Test. The Gamma test is a nonparametric technique aimed at estimating the variance of the noise present in a dataset [6, 12, 16], very useful in the construction of data-driven models. Noise is any source of variation in the target variable that cannot be explained by a smooth function (model) relating the target with the predictor variables. The gamma estimate indicates whether it is possible to explain the target variable by a smooth deterministic model based on the predictor variables. From this, an assessments can be made on (i) whether it is hopeful or hopeless to find a smooth model to the data, (ii) whether more explanatory variables should be incorporated to the data, (iii) how many observations are minimally required in order to build a model, (iv) appropriate thresholds in order to avoid overfitting during training and (v) what is the overall quality of the data. The most important assumptions of the procedure are (i) the model function f is continuous within the input space, (ii) the noise is independent of the input vector \overleftarrow{x} and (iii) the function f has bounded first and second partial derivatives.

Let \mathcal{S} be a system described in terms of a set of variables and with $y \in \mathbb{R}$ being a variable of interest, potentially related to a set of m variables $\overleftarrow{x} \in \mathbb{R}^m$

expressed as $y = f(\overleftarrow{x}) + r$, where f is a smooth unknown function representing the system, \overleftarrow{x} is a set of predictor variables and r is a random variable representing noise or unexplained variation. Let M be the number of observations and p is the number of nearest neighbors considered. If $\overleftarrow{x}_{N[i,k]}$ is the k -th nearest neighbor of object \overleftarrow{x}_i , for every $k \in [1, p]$, a sequence of estimates of $\mathbf{E}(\frac{1}{2}(y' - y)^2)$ based on sample means is computed as

$$\begin{aligned} \gamma_M(k) &= \frac{1}{2M} \sum_{i=1}^M |y_{N[i,k]} - y_i|^2 \\ \delta_M(k) &= \frac{1}{M} \sum_{i=1}^M |\overleftarrow{x}_{N[i,k]} - \overleftarrow{x}_i|^2 \end{aligned} \tag{2}$$

where \mathbf{E} denotes the mathematical expectation and $|\cdot|$ Euclidean distance. The relationship between $\gamma_M(k)$ and $\delta_M(k)$ is assumed linear as $\delta_M(k) \rightarrow 0$ and an estimate for the variance of the noise $\Gamma = Var(r)$ is obtained by linear regression of $\delta_M(k)$ vs. $\gamma_M(k)$

$$\gamma_M(k) = \Gamma + G \delta_M(k) \tag{3}$$

From Eq. 3 the vRatio (V_r) is defined as a normalized Γ value with respect to the variance of the target variable. Since $V_r \in [0, 1]$, it allows comparisons across different datasets:

$$V_r = \frac{\Gamma}{Var(y)} \tag{4}$$

Assessing the relevance of the predictor variables is approached by searching for subsets with good Γ -statistics. In real-world cases the search space is determined by the power set of the predictor variables and evolutionary computation methods provide an alternative to the prohibitive brute force. A genetic algorithms explores subsets of predictors represented as binary vectors $\overleftarrow{\vartheta} = \{0, 1\}^m \in \mathbb{R}^m$ (masks). Each represents a subset determined by the predictors present in the vector and the target y . The potential of each subset of variables is given by the Γ -statistics, which could be specified in different ways. A single-objective cost function can be formulated as a linear combination of partial fitness coming from (i) the MSE as associated to V_r (the I_f term), (ii) ‘model smoothness’ as associated to G (the G_f term) and (iii) ‘model complexity’ given by the relative number of predictors (the L_f term).

$$F(\overleftarrow{\vartheta}) = W_i * I_f(\overleftarrow{\vartheta}) + W_g * G_f(\overleftarrow{\vartheta}) + W_l * L_f(\overleftarrow{\vartheta}) \tag{5}$$

where $W_i = 0.8$, $W_g = 0.1$, $W_l = 0.1$ are the weights of the contributing fitness terms, the largest of which is given to I_f , directly related to the estimated MSE.

$$\begin{aligned}
I_f(\overleftarrow{\vartheta}) &= \begin{cases} 1 - (1 - 10 * V_r(\overleftarrow{\vartheta}))^{-1} & \text{if } V_r(\overleftarrow{\vartheta}) < 0 \\ 2 - 2(1 + V_r(\overleftarrow{\vartheta}))^{-1} & \text{otherwise} \end{cases} \\
G_f(\overleftarrow{\vartheta}) &= 1 - (1 + |G(\overleftarrow{\vartheta})|/range(y))^{-1} \\
L_f(\overleftarrow{\vartheta}) &= \sum \overleftarrow{\vartheta} / m
\end{aligned} \tag{6}$$

The choice of the weights $\{W_i, W_g, W_l\}$ is a compromise between the importance given to the partial fitness components coming from the subset’s V_r , the model complexity G and the model’s cardinality (the smaller, the simpler, since it contains fewer predictors). The practical form of $\{I_f, G_f, L_f\}$ in (6) is a heuristic emerging from many different applications. This use of *GammaTest* statistics has been very successful elsewhere [21–23].

3.3 Modeling Techniques

Several black and white box approaches have been used for learning predictive models for the Atomization Energy property. Namely, Neural networks (fully connected multilayer perceptrons) and Random Forests [5, 13] as black box representatives, with M5 model trees [11, 14, 19, 20] and Multivariate adaptive regression splines (MARS)¹ [7, 8] as white box instances.

4 Results

4.1 Unsupervised Analysis and Data Exploration

Since it is not possible to properly display 3D content on hard media, snapshots from fixed perspectives are presented. In order to simplify the representation, the original 7102 objects were pre-clustered using the leader algorithm [9] with an Euclidean distance threshold of 12.33, which produced 1003 clusters (leaders). They are shown as semi-transparent spheres with sizes proportional to the cluster sizes. For both methods, the Sammon and the t-SNE mappings, the 3D transformations clearly exhibit the presence of well defined structures composed of different clusters. In the case of Sammon mapping (Fig. 1, Top), there is a lower density structure (upper right, mostly composed of outlying elements), well differentiated from a left area of much higher density composed of a sequence of clusters which progressively become more sparse. Under t-SNE, the mapping exhibits outlying elements at the top and the right respectively (Fig. 1, Bottom). Several clusters are also well defined and they correspond to the major structures of the Sammon mapping.

This initial exploration of the data using unsupervised visualization techniques reveals the existence of well differentiated classes of molecules, determined by their Coulomb matrices and atomic composition. These structures would be exploited by supervised techniques aiming at predicting molecular properties.

¹ MARS is trademarked and licensed to Salford Systems.

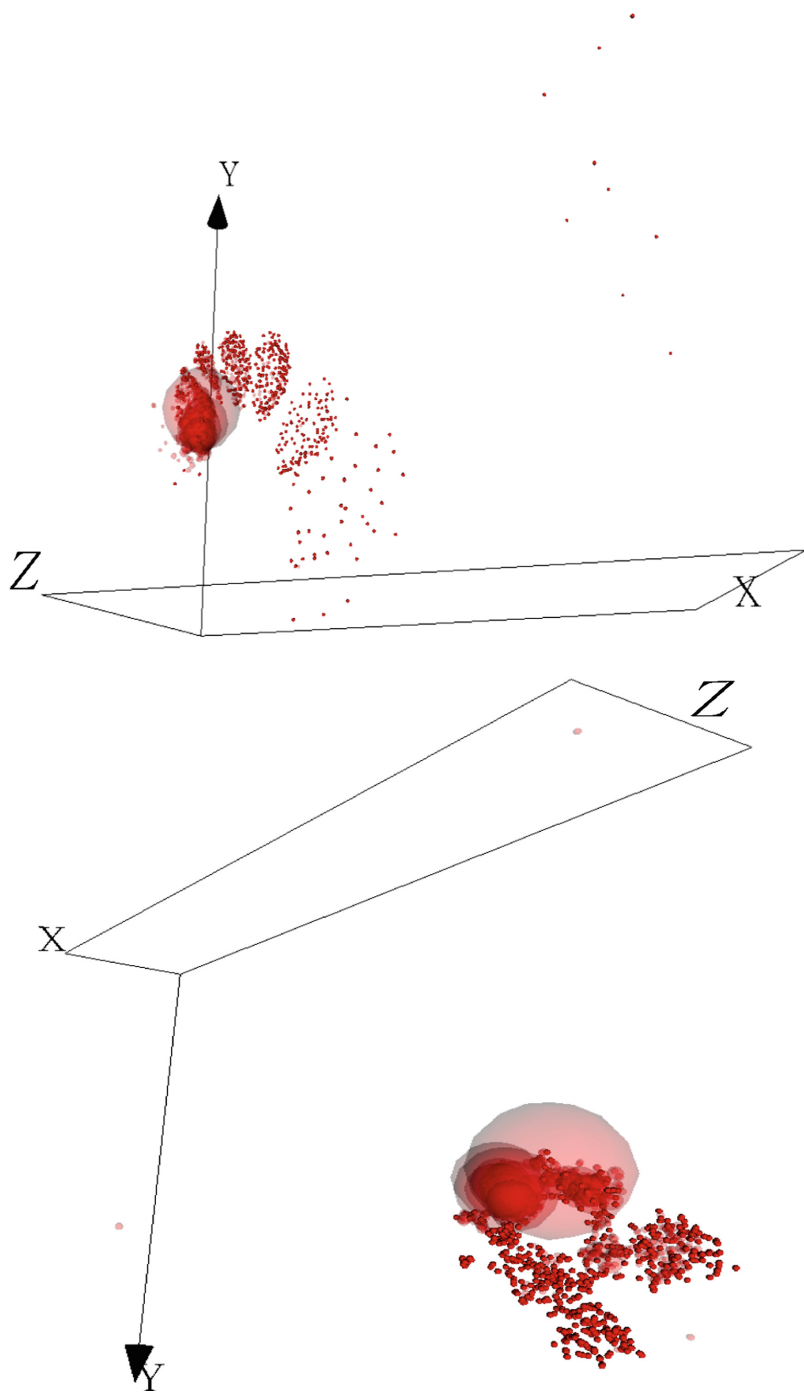


Fig. 1. Low dimensional spaces. Top: Sammon mapping. Bottom: t-SNE mappings.

4.2 Supervised Analysis

An orientative estimation of the predictive information contained in the data was obtained by computing the vRatio index (Eq. 4). The value obtained was quite low ($V_r = -0.0056$) indicating that the information contained within the 281 descriptor variables has a high predictive power for the Atomization Energy target variable (in z-score form). This result provides a quantitative assessment of what the exploratory methods of Sect. 3.1. Several supervised machine learning approaches were used to construct models for the Atomic Energy property: random forests, model trees, neural networks (multilayer perceptrons) and multivariate adaptive regression splines. In all cases, a 10-fold crossvalidation scheme was used, with standard performance measures: (1) Mean absolute error ($MAE = (1/n) \sum_{i=1}^n |o_i - p_i|$), (2) Root mean squared error ($RMSE = \sqrt{(1/n) \sum_{i=1}^n (o_i - p_i)^2}$) and (3) Pearson Correlation coefficient (R), where n is the number of observations, and p_i, o_i the predicted and observed values respectively (z-scores) of the Atomization Energy. Two of the modeling techniques used are considered as ‘black box’ (neural networks and random forests), while the model trees and adaptive regression are ‘white box’, transparent ones.

The neural networks models used were multilayer perceptrons with one and two fully connected hidden layers and one output layer in $20 \times 15 \times 1$, $30 \times 15 \times 1$ architectures respectively. All activation functions were of the relu type, with MAE as the loss function, optimized by the ‘Adam’ procedure (using a python-keras implementation). The number of epochs was set to 200, with 10% of the training set in the fold used for validation. Other parameters were: learning rate = 0.001, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and no AMSGrad. The random forests model had 10 trees in the forest, with a number of features equal to $\log(N_i) + 1$, where N_i is the number of inputs. M5 model trees were generated with a minimum of 4 instances/leaf node and pruned rules. In a second application, bagging of this type of models was used with 10 iterations of the bagging procedure, and 100% of the training set as the size of each bag. The Adaptive Regression models were applied with a maximum number of terms allowed set to 100 in two variants and with a maximum interaction degree of $K = 5$.

Two rounds of modeling experiments were conducted. In the first round the entire set of 281 predictor variables were used. For the second round, the power set of predictors were explored with a genetic algorithm using binary vectors as individuals, as described in Sect. 3.2. The objective function and the balancing weights W were those of Eq. 5, using the partial fitness from Eq. 6. The genetic algorithm settings were: population size = 100 individuals, one-point crossover operator with rate = 0.5 and bit mutation operator with rate = 0.05. At the end of the evolution process, the best individual had 156 selected predictors (55.5% of the originals) with a $V_r = 5.759 \cdot 10^{-8}$, indicating both a high degree of predictability of the target variable and a high degree of redundancy in the original set of predictors. The modeling results for the two round of experiments are shown in Table 1.

All modeling techniques produced predicted outputs which are highly correlated with the observed Atomization Energies. The minimum correlation coeffi-

Table 1. Modeling results with different machine learning techniques (10-fold Cross-validation)

ROUND 1 (281 original predictors)	MAE	RMSE	R
NN ($20 \times 15 \times 1$)	0.0220	0.0340	0.9995
NN ($30 \times 15 \times 1$)	0.0230	0.0330	0.9995
Random forest	0.0639	0.0857	0.9964
M5 model trees	0.0612	0.0915	0.9958
M5 model trees (bagging)	0.0515	0.0742	0.9973
Adaptive regression	0.0658	0.0876	0.9961
ROUND 2 (156 selected predictors)	MAE	RMSE	R
NN ($20 \times 15 \times 1$)	0.034	0.048	0.9990
NN ($30 \times 15 \times 1$)	0.035	0.051	0.9985
Random forest	0.0659	0.0874	0.9962
M5 model trees	0.0631	0.0982	0.9952
M5 model trees (bagging)	0.0538	0.0771	0.9971
Adaptive regression	0.0689	0.0926	0.9957

cient was 0.9952, corresponding to M5 model trees using the reduced set of variables. The highest correlation was obtained with the neural network (0.9995), closely followed by the bagged M5 model trees (0.9973). Overall, using only 55.5% of the predictors (Round 2) barely affected the correlation results. From the point of view of the error measures, the best models were the neural networks, in particular the $20 \times 15 \times 1$ architecture, with a cross-validation MAE of 0.0220 on the z-scores of Atomization Energy. More complex layouts ($30 \times 15 \times 1$) did not differ significantly in performance. This kind of behavior has been observed elsewhere, when working with neural networks. It is noteworthy that the bagged M5 model trees performed consistently better than Random Forest for all of the measured considered. Moreover, the later was matched by the Adaptive Regression model, which is an explicit, deterministic representation of the functional dependencies. While random forests are notoriously opaque, M5 and Adaptive Regression models are totally transparent. The single M5 model tree is composed of 100 rules when using 281 predictors and 92 when using the 156 selected by the evolutionary algorithm. Altogether, the model composition indicates a high level of irrelevancies in the set of descriptor variables, which could be removed without losing predictive performance.

In the case of Adaptive Regression models, they did not achieve a competitive error-wise performance with respect to neural networks. However, they ranked similarly with Random Forest with respect to both error and correlation measures. The later is specially important when looking at the model structure. Both Adaptive Regression models for data with 281 and 156 predictors respectively, used only 5 variables (1.78% and 3.2% respectively). Considering

the original high dimensionality of the descriptor space, this represents a considerable reduction from a feature selection point of view. An important aspect is that this reduction is performed simultaneously with learning the underlying functional dependencies. The explicit models when using data with 281 and 156 predictors respectively are

$$\begin{aligned} zscores(Atom.Energy) = & -k_1 * x_{276} - k_2 * x_{277} + k_3 * Max(0, -x_1 - k_4) + \\ & k_5 * Max(0, x_1 + k_6) + k_7 * Max(0, -x_{19} - k_8) * Max(0, -x_{26} - k_9) - \\ & k_{10} * Max(0, x_{19} + k_{11}) * Max(0, -x_{26} - k_{12}) + k_{13} * Max(0, -x_{26} - k_{14}) * \\ & Max(0, -x_{26} - k_{16}) + k_{17} * Max(0, -x_{26} - k_{18}) * Max(0, x_{26} + k_{19}) + k_{20} * \\ & Max(0, -x_{26} - k_{21}) - k_{22} * Max(0, x_{26} + k_{23}) - k_{25} \end{aligned}$$

and

$$\begin{aligned} zscores(Atom.Energy) = & -K_1 * x_{152} - K_2 * x_{153} + K_3 * Max(0, -x_1 - K_4) + \\ & K_5 * Max(0, x_1 + K_6) + K_7 * Max(0, -x_{11} - K_8) - K_9 * Max(0, x_{11} + K_{10}) + \\ & K_{11} * Max(0, -x_{24} - K_{12}) - K_{13} * Max(0, x_{24} + K_{14}) - K_{15}, \end{aligned}$$

where $x, k_i, K_i \in \mathbb{R}, i \in \mathbb{N}^+$ are constants found during the learning process and $Max(p, q)$ is the maximum between p and q . In models of this kind, it is possible not only to explicit the important predictor variables, but also the way in which they interact, which is transparently shown by the multiplicative terms involving the basis functions (e.g. $max(0, -x_1 + k_{11})max(0, x_1 - k_{12})$). As discussed above, this model has a performance that matches the one from a random forest, which is a widely used, well established machine learning technique.

From the point of view of performance, neural networks outperformed all other models. However, explicit, transparent models were capable of providing good results, at the level of other well established (black box) techniques, while working with significantly fewer number of predictors, with the advantage of exposing the nature of their interrelations and producing simple models.

5 Conclusions

The visualization of low-dimensional mappings from Coulomb matrices and atomic composition provided understanding of the structure of the data. They revealed the existence of well defined clusters from the point of view of both local distance preservation and consistency of conditional probability distributions between the original and the target spaces. The results obtained with different machine learning techniques aiming at modeling Atomization Energy (random forests, neural networks, model trees and adaptive regression), proved effective at capturing the functional dependencies between molecular structure and composition, and molecular properties, like Atomization Energy. Black and white models were produced that combine performance with transparency and explanation, identifying small subsets of relevant variables. Performance-wise, neural network models were superior, but adaptive regression in particular, produced relatively compact and transparent models, with accuracies comparable to

those provided by well established techniques like random forests. Future work will expand the studies to other molecular properties, as well as to mechanisms for deriving molecular structure from desired properties.

References

1. Baldi, P., Müller, K.R., Schneider, G.: Charting chemical space: challenges and opportunities for artificial intelligence and machine learning. *Mol. Inf.* **30**(9), 751 (2011)
2. Rupp, M.: Machine learning for quantum mechanics in a nutshell. *Int. J. Quantum Chem.* **115**, 1058–1073 (2015)
3. Montavon, G., et al.: Machine learning of molecular electronic properties in chemical compound space. *New J. Phys.* **15**(9), 095003 (2013)
4. Belisle, E., Huang, Z., Le Digabel, S., Gheribi, A.E.: Evaluation of machine learning interpolation techniques for prediction of physical properties. *Comput. Mater. Sci.* **98**, 170–177 (2015)
5. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
6. Evans, D., Jones, A.J.: A proof of the gamma test. *Proc. Roy. Soc. Lond. A* **458**, 1–41 (2002)
7. Friedman, J.: Multivariate adaptive regression splines. *Ann. Stat.* **19**(1), 1–67 (1991)
8. Friedman, J.: Fast mars. Technical Report 110, Stanford University, Department of Statistics (1993)
9. Hartigan, J.A.: *Clustering Algorithms*. Wiley, New York (1975)
10. Hinton, G.E., Roweis, S.T.: Stochastic neighbor embedding. In: *Advances in Neural Information Processing Systems*, vol. 15, pp. 833–840 (2002)
11. Holmes, G., Hall, M., Prank, E.: Generating rule sets from model trees. In: Foo, N. (ed.) *AI 1999. LNCS (LNAI)*, vol. 1747, pp. 1–12. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-46695-9_1
12. Jones, A.J., Evans, D., Margetts, S., Durrant, P.: The Gamma Test. In: Sarker, R., Abbass, H., Newton, S. (eds.) *Heuristic and Optimization for Knowledge Discovery*. Idea group Publishing (2002)
13. Kuncheva, L.I.: *Combining Pattern Classifiers, Methods and Algorithms*. Wiley, New York (2005)
14. Quinlan, J.R.: Learning with continuous classes. In: *5th Australian Joint Conference on Artificial Intelligence*, Singapore, pp. 343–348 (1992)
15. Sammon, J.W.: A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.* **C-18**(5), 401–409 (1969)
16. Stefánsson, A., Končar, N., Jones, A.J.: A note on the gamma test. *Neural Comput. Appl.* **5**, 131–133 (1997)
17. Valdés, J.J.: Virtual reality representation of information systems and decision rules: an exploratory technique for understanding data and knowledge structure. In: Wang, G., Liu, Q., Yao, Y., Skowron, A. (eds.) *RSFDGrC 2003. LNCS (LNAI)*, vol. 2639, pp. 615–618. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-39205-X_101
18. Maaten, L.V.D., Hinton, G.: Visualizing high-dimensional data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008)
19. Wang, Y., Witten, I.H.: Induction of model trees for predicting continuous classes. In: *Proceedings European Conference on Machine Learning*, Prague, pp. 128–137 (1997)

20. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco (2005)
21. Valdés, J.J., Cheung, C., Wang, W.: Evolutionary computation methods for helicopter loads estimation. In: Proceedings of the 2011 IEEE Congress on Evolutionary Computation, New Orleans, USA, 5–11 June 2011 (2011)
22. Valdés, J.J., Cheung, C., Li, M.: Towards conservative helicopter loads prediction using computational intelligence techniques. In: Proceedings of the 2012 IEEE World Congress on Computational Intelligence, International Convention Centre, Brisbane, Australia, 10–15 June 2012, pp. 1853–1860 (2012)
23. Valdés, J.J., Cheung, C., Li, M.: Sensor dynamics in high dimensional phase spaces via nonlinear transformations: application to helicopter loads monitoring. In: 2014 IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2014), Orlando, 9–12 December 2014 (2014)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

