Deciphering the Contents of Chemically-Trained Neural Networks into Physical Intuition

Eric Berquist

May 20, 2017

points from panel review:
- It did not become clear why identifying the inner workings of machine learning approaches, i.e., identifying whether an ML approach gets the right answer for the right reasons, has any practical relevance. - Could you explain more what is the payoff from identifying these inner workings? - Would this help with constructing more transferable machine learning approaches?

- Another point that did not become very clear was how you would construct the "novel" machine learning approach in aim #3. Can you explain what would be novel about your approach?

-> The point is not to apply a novel ML architecture to QC for improved prediction. The point is to apply a hypothetically better ML architecture to QC to see if better or worse prediction performance correlates with changes in input feature relevance.

1 Overview and Objectives

{The explosive growth of computing power over the past TODO years**REF** has led to the use of machine learning (ML) models for the accurate calculation of chemical properties.} Current ML models are capable predicting energetic properties of small, neutral organic molecules to almost 1 kcal/mol\{ REF}. However, it is unclear *what* the quality of a prediction will be when a ML model is given a molecule outside its training set, and *why* the model is giving that prediction. Each ML model is a set of unknown parameters that constitute a black box,[1] where it is not known or understood *how* the model functions, *why* the model gives the predictions it does, and whether or not correct predictions are being made for the right reasons. A critical step for the continued application of ML models to quantum chemistry will be to make predictions beyond the relatively simple example systems that are currently seeing widspread use.[2] Some of the most important, interesting, and unexplored chemistry in terms of reactivity and intrinsic molecular properties involves species that are either charged or have unpaired electrons. If ML models are only coincidentally forming correct predictions for simple species, there can be no guarantee or expectation for accurate predictions on these more realistic species.

The overall objective is to quantify what quantum chemistry-trained machine learning models are actually learning. The central hypothesis is that the application of machine learn-

ing to quantum chemistry is physically motivated, *i.e.*, ML models are learning parameters that connect the molecular representation (input features) to predictions in a manner that is qualitatively identical to how a trained chemist would apply their intuition. An implicit connection has already been made[3] between how a molecule is represented as input to ML models and the accuracy of prediction for certain kinds of properties, but this connection has not been explored. The rationale is that by understanding the connection between the molecular representation given to ML models and their predictions, ML models themselves can be used for rationalizing molecular structure-property relationships. The overall objective will be achieved through the following specific aims:

1. **Reproduce existing machine learning models from the literature.** I will replicate two neural networks (NNs) from [3] to serve as both the necessary architecture that analysis tools developed later in this proposal will be built on, and as the baseline for the quality of the results. NNs are chosen since they overall exhibit the most quantitatively accurate predictions compared to other ML architectures when trained on quantum chemical data. The properties that I will reproduce from the literature are the isotropic static polarizability $\bar{\alpha}$ [4] and the zero-point vibrational energy E_{ZPVE} [5], which represent molecular properties that can be derived from spectroscopic observables and are difficult to predict with current NN approaches.

2. **Characterize the parameters learned by existing machine learning models from the literature.** I will use relevance propagation[6] to quantify the relative importance of each input feature for each predicted property of the NNs in aim #1. Relevance propagation gives insight into a model's learned coefficients in a fashion that can be used directly in statistical analysis and visualization.

3. **Train multi-stage neural networks on both existing and new properties.** I will train combined unsupervised and supervised NNs to predict vibrational and nonlinear optical properties of the molecules used in aim #1. These properties are $\bar{\alpha}$, E_{ZPVE}, the static parallel first hyperpolarizability $\beta_{\parallel}$, and the set of frequencies $\{\tilde{\nu}\}$ that compose stick vibrational spectra. The first two properties will validate the performance of these new NNs against existing literature predictions, and the second two properties are a natural extension in complexity of the first two properties. Unsupervised NNs are capable of discovering intrinsic properties of their inputs\{ REF}, and adding them to the ML pipeline increases the likelihood that the models will learn "chemical" parameters.

4. **Characterize the parameters learned by multi-stage neural networks.** I will use the relevance propagation techniques from aim #2 to show the underlying causes for the performance of the NNs from aim #3 on both the existing and novel property predictions.

The expected outcomes are

- Clear connections between the input molecular representations and predicted outputs that can be used to build further quantitative structure-function relationships.

- Adding unsupervised learning stages improves the prediction accuracy of ML models in a form that appears analogous to chemical intuition.
- Well-defined and publicly-available protocols for applying more complex ML models to chemical properties, regarding all stages of the prediction pipeline: generation of the QM data from calculations, preparation of the input, training of the model, and analysis of the results.
- ML models capable of predicting complex molecular properties, with the first proof-of-concept predictions of higher-order nonlinear optical properties and vibrational spectra.

2 Significance

This is the first attempt at understanding the parameters of ML models used to predict microscopic and macroscopic molecular properties, rather than treating ML models as black boxes that cannot be understood. If ML models are shown to be learning physically-motivated or chemically-intuitive parameters, ML can become more than just a path for the accurate prediction of molecular properties, but be transformed into a tool itself that can give deep insight into molecular structure-property relationships. Training ML models is expensive in terms of both human and CPU time, it would be good to quantify this. If this proposal shows that ML models are not learning properly, then the scientific research community can avoid the wasteful use of resources on model training and shift their focus to developing better model architectures than the current state-of-the-art.

3 Background

3.1 Machine Learning

Machine learning is the ability for computers to "learn" without being given explicit instructions. Rather than providing exact instructions though traditional programming, computers are fed sets of input data and are usually expected to return a certain result. By training itself to reproduce results, a learned ML model would ideally be able to predict outputs for new, unknown inputs. Common applications of ML are in email spam filtering, search engine prediction, image and voice recognition, and self-driving cars.

Some definitions and terms that are used throughout this proposal are

- *Architecture*: the formal structure of the network or ML model itself, encompassing the region from equations and diagrams to the implementation (code).
- *Model*: here, an implemented architecture (in code) with learned parameters.
- *Pipeline*: multiple steps and components chained together, such as the preparation of data for input into an architecture, the architecture itself, and any steps required to transform the architecture output into something else useful, such as visualizations or statistics.

There are two categories of learning discussed throughout this proposal:

- *Supervised learning*: Train a machine learning model using data where the correct output prediction is known and given for each input sample, and the goal of the model is to predict similar types of outputs for new inputs\{ REF}.
- *Unsupervised learning*: Train a machine learning model using data where the correct output prediction is not given, and the goal of the model is to learn intrinsic properties of the inputs by recreating the input as output\{ REF}.

3.2 Neural Networks

Short background on (artificial) neural networks; goal, general structure (w/ figure) and terminology, how do they learn (backpropagation)

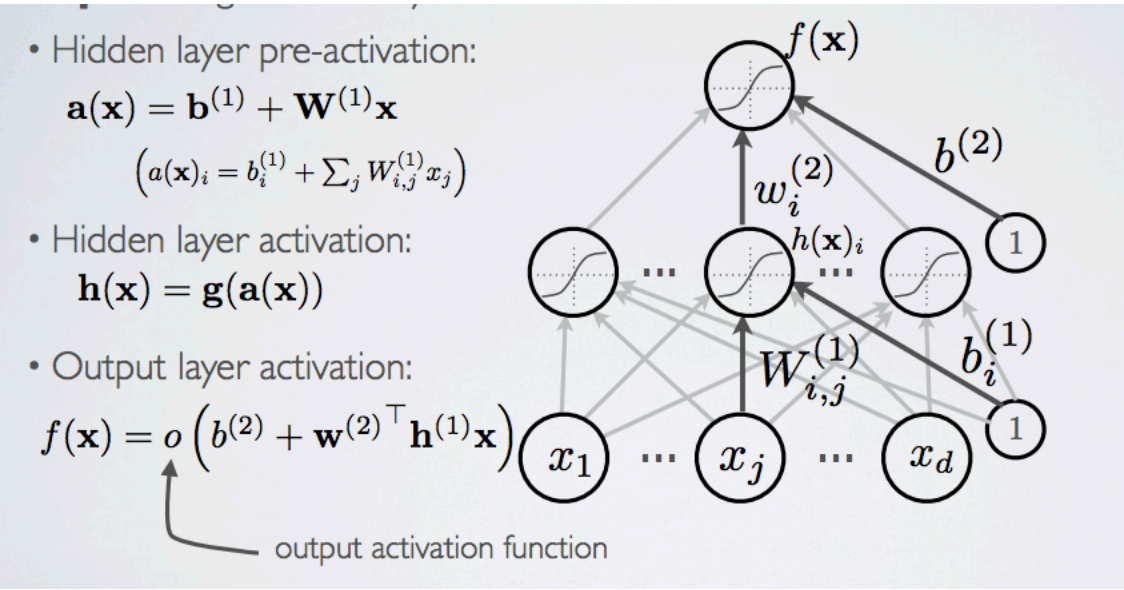

Figure 1: Cartoon represtation of a feedforward neural network, with the input layer on the bottom, hidden layer in the middle, and output layer on the top. Taken from [7]. Something like this but cleaner.

The neural network architectures implied by the two types of learning above lead to the capability of chaining them together in a pipeline, where an unsupervised NN is trained for some amount of time in a "pretraining" step, and the bottom layers (closest to the input) in an unsupervised NN are replaced with the contents of the unsupervised NN, rather than being initialized with random weights.

3.3 Machine Learning in Chemistry

The use of machine learning to make chemical predictions is not new, with work dating back over 25 years for prediction of NMR spectra using small neural networks trained on experimental data.[8] The largest application of machine learning to chemical problems is

within cheminformatics, where it has seen wide use within industrial drug discovery with emphasis on predicting quantitative structure-activity relationships (QSAR).[9] The goal is to predict the activity of a given drug candidate based on experimental activities of many other molecules, with inputs being information about atom types, bond types, number of aromatic rings, atomic partial charges, and other pieces of structural information, all of which are related to the molecular graph or connectivity.[10]

In particular, there is a recent application of deep neural networks (DNNs) to QSAR datasets,[11] which contains a systematic study for determining the best model parameters. The machine learning community calls this "hyperparameter tuning", which is another term for parameter optimization. However, this is still an empirical black-box approach, where the input is carefully controlled and statistical analysis is performed on the output, but this does not provide enough insight into how or why the quality of a model changes. For example, whether or not a rectified linear unit (ReLU) or sigmoid unit is the best function to represent neuron activation says nothing about why one molecule may be more potent than another in a QSAR study. This brute-force type of parameter optimization *does* provide a good starting point for understanding the sensitivity of a ML model. Unfortunately, even parameter optimization has not been extensively performed on models trained using quantum chemical data. In that sense, cheminformatics is a step ahead of other sub-disciplines in chemistry regarding the *application* of machine learning models, but not in the *understanding* of their models.

Their parameter optimization study is especially relevant to this proposal because it examines the effect of placing an unsupervised NN before other NNs for unsupervised pre-training. Surprisingly, the authors found that an unsupervised pretraining step decreased the accuracy of their predictions, which is counter to the expected outcome of this proposal. However, the paper implies that their results are not even valid due to algorithmic restrictions in their software. Therefore it seems incorrect to draw any conclusions from this, such as "no unsupervised pretraining is needed". It would be interesting to see if the same conclusion is drawn for models trained on quantum chemical data using the proper algorithms, as will be done in this proposal.

Additionally, it is unclear why a DNN trained on the combination of all QSAR data sets (called a "joint DNN") performs better than separate DNNs for each data set when considering the lack of overlap in the training sets. The methods developed in this proposal, while being applied to models trained on quantum chemical data, should be applicable to any DNN (consider that relevance propagation is mostly developed in computer vision/image recognition). One goal of this proposal is to transfer the idea of relevance propagation from its original intended application field to other fields. If it is indeed transferable, then it may shed some light on why unsupervised learning resulted in decreased prediction performance and the improvement of join DNNs over separate DNNs.

3.4 Machine Learning in Quantum Chemistry

Here is where I cite Aspuru-Guzik, Parkhill, von Lilienfeld (others?), with focus on the Arxiv paper from 2017-02,[3] need to keep digesting paper from 2017-04-04.[12] The former is the paper I base most of my proposal argument on.

3.5 Relevance Propagation

Layer-wise relevance propagation is a method for identifying what a ML model has learned[13] in terms of the model's input features. Figure 2 is a concrete example of what the output from relevance propagation looks like when applied to image classification by a neural network. Here, we assume that the network correctly identified the subject of the image as a cat (rather than a dog or a potted plant), but relevance propagation shows which image pixels were most important for the network to determine the photo is of a cat. The pixel-wise importance is a single number for each pixel that can be interpreted as a contribution for that pixel to the final classification of the image. More generally, is it the relative importance of each input feature to the predicted output; here and in other image recognition examples, pixels are input features. Applications to image classification resulting in pixel importance naturally lends itself to visualizing the output as a heatmap on top of the original input.

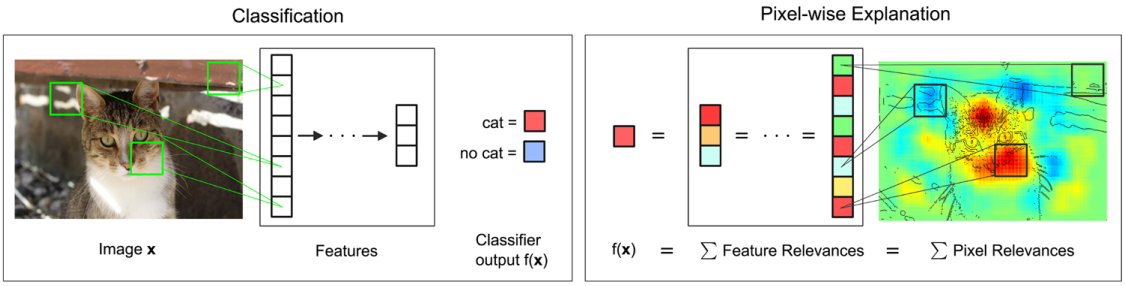

Figure 2: Example of output from relevance propagation showing which sections of an image the neural network considered important during classification. Taken from [13].

Other methods exist for assigning rules of how input features map to predictions.[14–16] Several of these are based on the idea of gradient perturbations, where repeated changes in prediction are measured as a result of small changes in the input. Performed enough times, this creates a map of the network's decision boundary.[17] A gradient perturbation-based method is unsatisfactory because it requires repeated forward passes through the network with a set value for the perturbation size, and relevance propagation requires only one backward pass with no free parameters. Additionally, most methods for assigning input relevance have only been used for image classification, where the input features are of uniform type (pixel data). The input featurization for representing molecular structures[3] is heterogeneous, and it is not clear how the perturbation parameter should be varied for each kind of molecular feature.

Although no improvements will be made to the basic relevance propagation algorithm itself, there is novelty in two areas. To the best of the author's knowledge, this is the first time relevance propagation will be applied to a regression task rather than a classification task, and the first time relevance propagation will be applied outside of image classification or computer vision. A potential connection between the heatmap representation and hallucinations from generative adversarial networks (GANs), which have been applied to quantum chemistry\{ TODO, John Parkhill's paper} is an interesting future research area.

Mention the Kearnes 2016 Graph Convolutions paper; there is a figure that shows the "evolution of input features" over time, but it's completely unclear as to what the metric/units

are. A difference is that we are not necessarily interested in the time aspect, just the final explanation.

4 Research Plan

4.1 Specific Aim #1: Reproduction of Existing Literature Neural Networks

4.1.1 Introduction

- The objective is to reproduce trained neural networks from the literature in order to create the foundation of the machine learning pipeline to be developed within this proposal and to serve as a validation baseline for further predictions.
- The hypothesis is that the published neural networks are entirely reproducible by connecting free, open-source tools.
- To test this hypothesis, I will reproduce the ML pipeline and results from [3].
- The expected outcome is a fully-worked and documented reproduction of neural networks from the literature that can serve as the basis for not only this proposal's later aims, but for future pipelines within the wider chemistry and machine learning communities.

4.1.2 Research Design

Unfortunately, the learned models for the results presented in [3] are not available, only descriptions of the architectures. Recreating the literature models requires an implementation of the model architecture and input data in the proper format.

There are two neural network-based architectures described in [3]: Graph Convolutions[10] (GC) and Gated Graph Neural Networks[18] (GG). These NN architectures are used again as baselines in [12]. Since the original GC implementation referenced in [3] is openly available,[19] I will use the GC-based architecture with modifications described in section E5 of [3]. Is it safe to just reference this paper, or are more details necessary? Details for the GC architecture input, called the Molecular Graph representation, are shown in tables 1 and 2 of [3] and reproduced here.

The QM9 dataset consists of 134K molecules[2] containing up to 9 heavy atoms from the elements H, C, O, N, and F, with a maximum number of 29 atoms. The representation in tables 1 and 2 will result in an input size of $21\binom{x}{1}+13\binom{x}{2}$ for a given number of atoms x, leading to a total length of 5,887 for the maximum number of 29 atoms in QM9. Inputs are available as modified XYZ files from the Quantum Machine (`http://quantum-machine.org/`) website under the QM9 Dataset section,[2,20] which will be transformed into the Molecular Graph (MG) representation using RDKit[21] with Gasteiger partial charges as in [10].

- Modify the original Graph Convolutions architecture to the one described in Section E5 of [3].

Table 1: The Molecular Graph (MG) input representation: single atom features

Feature	Description	Size
Atom type	H, C, N, O, or F (one-hot)	5
Chirality	R or S (one-hot or null)	2
Formal charge	Integer electronic charge	1
Partial charge	Calculated partial charge	1
Ring sizes	For each ring size (3-8), the number of rings that include this atom	6
Hybridization	sp, sp^2, or sp^3 (one-hot or null)	3
Hydrogen bonding	Whether this atom is a hydrogen bond donor and/or acceptor (binary values)	2
Aromaticity	Whether this atom is part of an aromatic system	1
		21

Table 2: The Molecular Graph (MG) input representation: atom pair features

Feature	Description	Size
Bond type	Single, double, triple, or aromatic (one-hot or null)	4
Graph distance	For each distance (1-7), whether the shortest path between the atoms in the pair is less than or equal to that number of bonds (binary values)	7
Same ring	Whether the atoms in the pair are in the same ring	1
Spatial distance	The Euclidean distance between the two atoms	1
		13

- Using the model parameters described in that section, train two separate models, one for the isotropic static polarizability $\bar{\alpha}$, and another for the zero-point vibrational energy E_{ZPVE}.

For the reproduction of literature results, the only numerical values from [3] are in Table 3, which shows the mean absolute error (MAE) for each input representation-architecture combination. Because the sample size of QM9 is sufficiently large (134K molecules), the MAE is calculated using out-of-sample validation, where the ML models are trained using 90% of the available data and compared against the DFT (B3LYP/6-31(2df,p)) results for the remaining 10%. The 90% constitutes ~117K molecules after removing 3K from 134K due to failed SMILES consistency tests. This 90/10 (training + validation)/test set split allows for 10-fold cross-validation. It is not mentioned how the concrete splits are obtained or how the final MAE is calculated from the 10 models. For this proposal, I will perform an unbiased random shuffle of QM9 index codes and split them into 10 uniform bins. After training and model validation using the procedure described above, the final MAE will be calculated as the mean of the 10 individual MAEs. Is this sufficient? If so, are there more technical terms for these procedures? The literature models will be considered replicated if the two final models I train have MAEs within 95% of 0.227 a_0^3 for the polarizability and 0.00975 eV for the ZPVE, respectively. Is there a better error metric than 95% of a single number? Seems very unsatisfactory

There is some ambiguity to me here. If I wanted to perform a prediction of one of these molecular properties, would I then train an 11th model using all 100% of the available sample data? What is the "final" model?

4.1.3 Expected Outcomes

The concrete products of this aim will be a set of Python scripts that transform the XYZ-like files into the Molecular Graph representation, implement the modified Graph Convolutions architecture, train MG/GC models for each molecular property ($\bar{\alpha}$ and E_{ZPVE}) using out-of-sample cross-validation, and calculate each molecular property from the trained models when given a normal XYZ molecular structure.

More details here, especially about the impact. Perhaps comment on how the literature is unclear without the code, so a particular emphasis will be placed on documentation during the replication with a desired clear connection to a literature-type discussion of the implementation. Literate programming (Jupyter Notebook) may be an option.

Also, what to do in the event that the lit model isn't reproducible? It may indicate that not enough information is given in the literature: publications are notorious for not providing enough information about code/model parameters for reproduction. Even in the event of supposed failure to fall within the error metric, the architecture is still a valuable building block. Is it safe to assume that the PI (me) will not make any mistakes in the implementation? Another possible point of failure may be due to the non-convex optimization problem posed by NNs. What is the likelihood of training a network into a local minimum?

4.2 Specific Aim #2: Characterization of Existing Literature Neural Networks

4.2.1 Introduction

- The objective is to quantify what already-published neural network-based ML models have learned.
- The hypothesis is that when predicting an output, the most important (relevant) parts of the input for that output align with our trained chemical intuition. Specifically, for strongly geometry-dependent properties, such as the ZPVE, more relevance will be placed on geometric input features such as bond lengths, angles, and dihedrals. For strongly wavefunction- or density-dependent properties, such as the isotropic polarizability or the HOMO-LUMO gap, more relevance will be placed on electronic input features such as atomic charges compared to other features.
- To test this hypothesis, I will develop the necessary ML pipeline for adding relevance propagation and analysis steps to the already-published ML models. This will involve connecting existing relevance propagation tools\{ TODO} to the end of the pipeline from aim #1.

4.2.2 Research Design

- Relevance propagation (http://heatmapping.org/)
  - available as a "toolbox" on top of TensorFlow, which is convenient considering that the original GC model is also on top of TensorFlow
  - which relprop model is appropriate? need to be one that conserves relevance
  - are there any other (free) parameters that I will need to control/adjust?
  - perform relevance propagation
  - derive form for analyzing contributions of input features to results, such as coefficients $\{c\}$ where $\sum_i^{\text{input features}} c_i^2 = 1$
  - analyze results from relevance propagation: graphs, histograms, etc.
    - how are the results represented straight out of the relprop algorithms? may need to do some transformations

4.2.3 Expected Outcomes

- Evidence for or against published ML models having learned chemically-intuitive parameters
- A model ML pipeline for applying relevance propagation to quantum chemistry models

4.3 Specific Aim #3: Construction and Training of Novel Neural Networks

4.3.1 Introduction

- The objective is to construct and train neural networks that can be analyzed for what they have learned.
- The already-trained properties will be the isotropic static polarizability $\bar{\alpha}$ and the ZPVE, and the not-before trained properties will be the static parallel first hyperpolarizability, $\beta_{\parallel}$, and full vibrational (stick) spectra, the set of frequencies $\{\tilde{\nu}\}$.
- The hypothesis is that because the multi-stage NN will be at least as sophisticated as the single-stage NN used previously in the literature, both E_{ZPVE} and $\bar{\alpha}$ should be predicted using the multi-stage NN with equal or less error than the single-stage NN. The more complex properties $\beta_{\parallel}$ and $\{\tilde{\nu}\}$ are expected to have larger relative errors, in particular the set of vibrational frequencies, as predictions of the highest fundamental frequency ω_1 alone already have large errors.[3]
- Applying new ML architectures to already well-studied properties is a safety check for the architeture's use; if it performs worse than current models for existing property predictions than it cannot be expected that it will perform well for new/more complex property predictions.

4.3.2 Research Design

- Results for $\bar{\alpha}$, E_{ZPVE}, and $\{\tilde{\nu}\}$ are already present in the labeled data that was used is inputs in aim #1 (that is, the QM9 data set[2]).
- I will use the DALTON quantum chemistry program package[22] for the hyperpolarizability calculations, as it is free for academic use and designed especially for the calculation of molecular response properties such as hyperpolarizabilities. These calculations will employ the B3LYP density functional in combination with the 6-31G(2df,p) basis set to maintain comparability with past calculations from the QM9 data set.[2]
- Start with the resulting (supervised) NN architectures/models from the literature that were used in aim #1.
- Build a "small" unsupervised NN architecture that can be connected to the front of the existing GC NN architecture (the "combined" architecture).

4.3.3 Expected Outcomes

- Models with unsupervised learning steps have improved prediction accuracy of chemical properties compared to those without. That is, the models developed and trained in this aim should show better prediction performance than the literature models from aim #1. This implies the models from this aim are of higher-quality and are more likely to have "learned correctly" in the sense that they learned "chemical intutition".

4.4 Specific Aim #4: Characterization of Novel Neural Networks

4.4.1 Introduction

- The objective is to determine the relative importance of each component in the molecular representation to predictions of complex molecular properties. This will be done by applying the analysis techniques developed in aim #1 to the neural networks trained in aim #2.
- The hypothesis is that the most important input features for $\beta_{\parallel}$ and $\{\tilde{\nu}\}$ are similar to those for $\bar{\alpha}$ and E_{ZPVE}, respectively.

4.4.2 Research Design

This specific aim, as currently planned, is just the application of the analysis from aim #1 to the models developed and trained in aim #2. Although the goal of each specific aim sounds logical, the actual division of work between each of the specific aims seems very uneven.

4.4.3 Expected Outcomes

- The parameters learned by ML models, and therefore their predictions, will show a strong dependence on the input features in chemically-intuitive manner.
- Neural network-based ML architectures are a valid path forward for predicting novel and more complex chemical properties.

5 Broader Impacts

A crucial reason for the growth in cross-disciplinary applications of machine learning is the openness and extensiveness of introductory tools, specifically tutorials and examples. Historically, chemistry lags behind other sciences in terms of openness of procedures and results. The current infrastructure surrounding the combination of machine learning and quantum chemistry is very poor: disorganized work, disorganized results, and not all components are available for reuse. The development of these machine learning pipelines will constitute the development of open-source, freely available infrastructure that will be easily extendable. I will provide openly **all** components of the machine learning pipeline developed in this proposed work, including the fully-trained models, meaning the implementations using open-source software and the learned parameters for each model. All components will be placed on GitHub, the premier location for the open hosting of machine learning tools. Making these tools available will enable the verification of future, more advanced machine learning models that has not been possible to date. The tools will also serve as a pedagogical example for how machine learning can be applied to quantum chemical problems.

As the application of machine learning within quantum chemistry is relatively new and fast-moving, still being in the "discovery" phase, there have not been attempts at replicating machine learning pipelines, peer-reviewed or otherwise. Additionally, in traditional quantum chemistry there are a plethora of well-known program packages for performing

electronic structure calculations[22–25] that are self-contained, *e.g.* a single program can calculate optimized geometries, vibrational spectra, NMR chemical shifts, reaction energies, etc. This infrastructure exists to some degree for machine learning, with base packages such as scikit-learn[26] and TensorFlow[27] themselves being self-contained with excellent tutorials and examples, however this infrastructure does not exist for quantum chemistry-derived machine learning models. Introductions to machine learning are numerous and extensive using the standard "fruit fly" of NNs, the MNIST database of handwritten digits,[28] and similar fully-worked introductions should exist for quantum chemistry as well. Releasing this pipeline from this proposal allows it to serve as the "fruit fly" for quantum chemistry in machine learning.

References

(1) https://en.wikipedia.org/wiki/Black_box (accessed 03/08/2017).
(2) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. *Scientific Data* **2014**, *1*, DOI: 10.1038/sdata.2014.22.
(3) Faber, F. A.; Hutchison, L.; Huang, B.; Gilmer, J.; Schoenholz, S. S.; Dahl, G. E.; Vinyals, O.; Kearnes, S.; Riley, P. F.; Anatole von Lilienfeld, O. *ArXiv e-prints* **2017**.
(4) Soscún, H.; Alvarado, Y.; Hernández, J.; Hernández, P.; Atencio, R.; Hinchliffe, A. *Journal of Physical Organic Chemistry* **2001**, *14*, 709–715, DOI: 10.1002/poc.407.
(5) Irikura, K. K. *Journal of Physical and Chemical Reference Data* **2007**, *36*, 389–397, DOI: 10.1063/1.2436891.
(6) Lapuschkin, S.; Binder, A.; Montavon, G.; Müller, K.-R.; Samek, W. *Journal of Machine Learning Research* **2016**, *17*, 1–5.
(7) https://dmm613.wordpress.com/2014/12/16/demystifying-artificial-neural-networks-part-2/ (accessed 05/15/2017).
(8) Thomsen, J.; Meyer, B. *Journal of Magnetic Resonance (1969)* **1989**, *84*, 212–217, DOI: 10.1016/0022-2364(89)90021-8.
(9) Baskin, I. I.; Winkler, D.; Tetko, I. V. *Expert Opinion on Drug Discovery* **2016**, *11*, PMID: 27295548, 785–795, DOI: 10.1080/17460441.2016.1201262.
(10) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. *Journal of Computer-Aided Molecular Design* **2016**, *30*, 595–608, DOI: 10.1007/s10822-016-9938-8.
(11) Ma, J.; Sheridan, R. P.; Liaw, A.; Dahl, G. E.; Svetnik, V. *Journal of Chemical Information and Modeling* **2015**, *55*, PMID: 25635324, 263–274, DOI: 10.1021/ci500747n.
(12) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. *ArXiv e-prints* **2017**.
(13) Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; Samek, W. *PLOS ONE* **2015**, *10*, 1–46, DOI: 10.1371/journal.pone.0130140.
(14) Finnegan, A. I.; Song, J. S. *bioRxiv* **2017**, DOI: 10.1101/105957.
(15) Slavin Ross, A.; Hughes, M. C.; Doshi-Velez, F. *ArXiv e-prints* **2017**.
(16) Lundberg, S.; Lee, S.-I. *ArXiv e-prints* **2016**.
(17) https://en.wikipedia.org/wiki/Decision_boundary (accessed 05/20/2017).
(18) Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. *ArXiv e-prints* **2015**.
(19) Kipf, T. N.; Welling, M. *arXiv preprint arXiv:1609.02907* **2016**.
(20) Ruddigkeit, L.; van Deursen, R.; Blum, L. C.; Reymond, J.-L. *Journal of Chemical Information and Modeling* **2012**, *52*, PMID: 23088335, 2864–2875, DOI: 10.1021/ci300415d.
(21) RDKit: Open-source cheminformatics., http://www.rdkit.org (accessed 05/07/2017).
(22) Aidas, K.; Angeli, C.; Bak, K. L.; Bakken, V.; Bast, R.; Boman, L.; Christiansen, O.; Cimiraglia, R.; Coriani, S.; Dahle, P.; Dalskov, E. K.; Ekström, U.; Enevoldsen, T.; Eriksen, J. J.; Ettenhuber, P.; Fernández, B.; Ferrighi, L.; Fliegl, H.; Frediani, L.; Hald, K.; Halkier, A.; Hättig, C.; Heiberg, H.; Helgaker, T.; Hennum, A. C.; Hettema, H.; Hjertenæs, E.; Høst, S.; Høyvik, I.-M.; Iozzi, M. F.; Jansík, B.; Aa. Jensen, H. J. Aa.; Jonsson, D.; Jørgensen, P.; Kauczor, J.; Kirpekar, S.; Kjærgaard, T.;

Klopper, W.; Knecht, S.; Kobayashi, R.; Koch, H.; Kongsted, J.; Krapp, A.; Kristensen, K.; Ligabue, A.; Lutnæs, O. B.; Melo, J. I.; Mikkelsen, K. V.; Myhre, R. H.; Neiss, C.; Nielsen, C. B.; Norman, P.; Olsen, J.; Olsen, J. M. H.; Osted, A.; Packer, M. J.; Pawlowski, F.; Pedersen, T. B.; Provasi, P. F.; Reine, S.; Rinkevicius, Z.; Ruden, T. A.; Ruud, K.; Rybkin, V. V.; Saek, P.; Samson, C. C. M.; de Merás, A. S.; Saue, T.; Sauer, S. P. A.; Schimmelpfennig, B.; Sneskov, K.; Steindal, A. H.; Sylvester-Hvid, K. O.; Taylor, P. R.; Teale, A. M.; Tellgren, E. I.; Tew, D. P.; Thorvaldsen, A. J.; Thøgersen, L.; Vahtras, O.; Watson, M. A.; Wilson, D. J. D.; Ziolkowski, M.; Ågren, H. *WIREs Comput. Mol. Sci.* **2014**, *4*, 269–284, DOI: 10.1002/wcms.1172.

(23) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Petersson, G. A.; Nakatsuji, H.; Li, X.; Caricato, M.; Marenich, A. V.; Bloino, J.; Janesko, B. G.; Gomperts, R.; Mennucci, B.; Hratchian, H. P.; Ortiz, J. V.; Izmaylov, A. F.; Sonnenberg, J. L.; Williams-Young, D.; Ding, F.; Lipparini, F.; Egidi, F.; Goings, J.; Peng, B.; Petrone, A.; Henderson, T.; Ranasinghe, D.; Zakrzewski, V. G.; Gao, J.; Rega, N.; Zheng, G.; Liang, W.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Throssell, K.; Montgomery Jr., J. A.; Peralta, J. E.; Ogliaro, F.; Bearpark, M. J.; Heyd, J. J.; Brothers, E. N.; Kudin, K. N.; Staroverov, V. N.; Keith, T. A.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A. P.; Burant, J. C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Millam, J. M.; Klene, M.; Adamo, C.; Cammi, R.; Ochterski, J. W.; Martin, R. L.; Morokuma, K.; Farkas, O.; Foresman, J. B.; Fox, D. J. Gaussian16 Revision A.03., Gaussian Inc. Wallingford CT, 2016.

(24) Shao, Y.; Gan, Z.; Epifanovsky, E.; Gilbert, A. T. B.; Wormit, M.; Kussmann, J.; Lange, A. W.; Behn, A.; Deng, J.; Feng, X.; Ghosh, D.; Goldey, M.; Horn, P. R.; Jacobson, L. D.; Kaliman, I.; Khaliullin, R. Z.; Kús, T.; Landau, A.; Liu, J.; Proynov, E. I.; Rhee, Y. M.; Richard, R. M.; Rohrdanz, M. A.; Steele, R. P.; Sundstrom, E. J.; Woodcock III, H. L.; Zimmerman, P. M.; Zuev, D.; Albrecht, B.; Alguire, E.; Austin, B.; Beran, G. J. O.; Bernard, Y. A.; Berquist, E.; Brandhorst, K.; Bravaya, K. B.; Brown, S. T.; Casanova, D.; Chang, C.-M.; Chen, Y.; Chien, S. H.; Closser, K. D.; Crittenden, D. L.; Diedenhofen, M.; DiStasio Jr., R. A.; Dop, H.; Dutoi, A. D.; Edgar, R. G.; Fatehi, S.; Fusti-Molnar, L.; Ghysels, A.; Golubeva-Zadorozhnaya, A.; Gomes, J.; Hanson-Heine, M. W. D.; Harbach, P. H. P.; Hauser, A. W.; Hohenstein, E. G.; Holden, Z. C.; Jagau, T.-C.; Ji, H.; Kaduk, B.; Khistyaev, K.; Kim, J.; Kim, J.; King, R. A.; Klunzinger, P.; Kosenkov, D.; Kowalczyk, T.; Krauter, C. M.; Lao, K. U.; Laurent, A.; Lawler, K. V.; Levchenko, S. V.; Lin, C. Y.; Liu, F.; Livshits, E.; Lochan, R. C.; Luenser, A.; Manohar, P.; Manzer, S. F.; Mao, S.-P.; Mardirossian, N.; Marenich, A. V.; Maurer, S. A.; Mayhall, N. J.; Oana, C. M.; Olivares-Amaya, R.; O'Neill, D. P.; Parkhill, J. A.; Perrine, T. M.; Peverati, R.; Pieniazek, P. A.; Prociuk, A.; Rehn, D. R.; Rosta, E.; Russ, N. J.; Sergueev, N.; Sharada, S. M.; Sharmaa, S.; Small, D. W.; Sodt, A.; Stein, T.; Stück, D.; Su, Y.-C.; Thom, A. J. W.; Tsuchimochi, T.; Vogt, L.; Vydrov, O.; Wang, T.; Watson, M. A.; Wenzel, J.; White, A.; Williams, C. F.; Vanovschi, V.; Yeganeh, S.; Yost, S. R.; You, Z.-Q.; Zhang, I. Y.; Zhang, X.; Zhou, Y.; Brooks, B. R.; Chan, G. K. L.; Chipman, D. M.; Cramer, C. J.; Goddard III, W. A.; Gordon, M. S.; Hehre, W. J.; Klamt, A.; Schaefer III, H. F.; Schmidt, M. W.; Sherrill, C. D.; Truhlar, D. G.; Warshel, A.; Xua, X.; Aspuru-Guzik, A.; Baer, R.; Bell, A. T.; Besley, N. A.; Chai, J.-D.; Dreuw, A.; Dunietz, B. D.; Furlani, T. R.; Gwaltney, S. R.; Hsu, C.-P.; Jung, Y.; Kong, J.; Lambrecht, D. S.; Liang, W.; Ochsenfeld, C.; Rassolov, V. A.; Slipchenko, L. V.; Subotnik, J. E.; Van Voorhis, T.; Herbert, J. M.; Krylov, A. I.; Gill, P. M. W.; Head-Gordon, M. *Mol. Phys.* **2015**, *113*, 184–215.

(25) Turney, J. M.; Simmonett, A. C.; Parrish, R. M.; Hohenstein, E. G.; Evangelista, F. A.; Fermann, J. T.; Mintz, B. J.; Burns, L. A.; Wilke, J. J.; Abrams, M. L.; Russ, N. J.; Leininger, M. L.; Janssen, C. L.; Seidl, E. T.; Allen, W. D.; Schaefer, H. F.; King, R. A.; Valeev, E. F.; Sherrill, C. D.; Crawford, T. D. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2012**, *2*, 556–565, DOI: 10.1002/wcms.93.

(26) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. *Journal of Machine Learning Research* **2011**, *12*, 2825–2830.

(27) Martn Abadi; Ashish Agarwal; Paul Barham; Eugene Brevdo; Zhifeng Chen; Craig Citro; Greg S. Corrado; Andy Davis; Jeffrey Dean; Matthieu Devin; Sanjay Ghemawat; Ian Goodfellow; Andrew Harp; Geoffrey Irving; Michael Isard; Jia, Y.; Rafal Jozefowicz; Lukasz Kaiser; Manjunath Kudlur; Josh Lev-

enberg; Dan Mané; Rajat Monga; Sherry Moore; Derek Murray; Chris Olah; Mike Schuster; Jonathon Shlens; Benoit Steiner; Ilya Sutskever; Kunal Talwar; Paul Tucker; Vincent Vanhoucke; Vijay Vasudevan; Fernanda Viégas; Oriol Vinyals; Pete Warden; Martin Wattenberg; Martin Wicke; Yuan Yu; Xiaoqiang Zheng TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems., Software available from tensorflow.org, 2015.

(28) LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. In *Intelligent Signal Processing*, IEEE Press: 2001, pp 306–351.