Supervised Learning of the Magnetic Moment of Metal Phthalocyanine Molecules

Summary

Metal Phthalocyanines (MPcs) form an important class of organic semiconductors. Recent interest has been motivated by the possibility of using MPcs as spin-filters in molecular spintronics. However, calculating the magnetic moment of a given MPc, a necessary step in selecting a spin filter, is an involved process that requires computationally expensive density functional theory calculations for each molecule. This work seeks to examine the possibility of using machine learning models, specifically multiple linear regression (MLR) and random forest (RF) regression, to predict the magnetic moment of MPcs molecules and its metal atom based on other results. We find the MLR model weakly predicts the magnetic moments while the RF model accurately predicts the magnetic moment of the resulting molecule/metal atom, but possibly overfits the model due to the minimal DFT+U contribution in the dataset.

Scientific Background

Metal Phthalocyanines (MPcs) are a promising material in technological fields as the molecules are generally chemically and physically stable while the various electrical, optical, and magnetic properties can be tuned by the choice of metal and chemical functionalization. They have found renewed interest as a molecular junction in spintronic applications as the metal atom can impart a magnetic moment and act as a spin-filter while the electronic structure and transport properties can be tuned by the choice of functional groups. Recently, Fadlallah and coauthors [1] have performed density functional theory (DFT) calculations on MPcs. They systematically varied the central metal across the entire first row of the d-block and exchanged the hydrogenated peripheral (MPc) for a fluorinated version (F16MPc) (Figure 1), to study the relationship with the magnetic moment (MM) of the Pc and the metal atom in its center. However, these DFT calculations are computationally expensive and need to be performed for each metal and functional group combination. Additionally, the correlated electrons of the d-block metal atom necessitate the need for calculations in the DFT+U framework[1], which adds another dimension of computational cost.

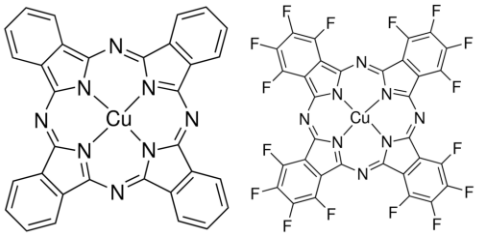

Figure 1: The chemical structure of regular copper Pc, CuPc, (left) and the fluorinated analog, F16CuPc (right)

Motivation

This work uses regression techniques to predict the MM of the total MPc molecule and the respective metal atom. The goal of this project is to predict the magnetic moment of a molecule or its metal atom given a limited data set. Ideally, this eventually would lead to a list of molecules with strong magnetic moments for further investigation/synthesis without the need for further DFT calculations.

---

[1] DFT typically operates in the single-particle approximation, meaning one electron is calculated while the remaining system is approximated as non-interacting. This fails in the case where multiple electrons need to interact to describe *strongly correlated* phenomena (e.g. magnetism, superconductivity) in d- and f-block systems. The U correction is an attempt to articulate the electron-electron interactions. The U term is a scalar parameter.

Data Collection

Computational data was drawn from Fadlallah and coauthors [1] yielding 132 total data points. Tables were extracted from the online version of the paper using the BeautifulSoup Python library. The predictor was taken as the computed MM. Estimators were taken to be the class of molecule (H16MPc or F16MPc), the class of computational system (metal atom or overall molecule), the U level of the computation (0, 4, and 8) and the metal atom identity (see next section).

Feature Engineering

Rather than leave the metal identity as a class feature, it was split into two features: the electron occupancy of the d-orbital and the d-spin state, both with the approximation of a +2 oxidation state. The decision to replace the metal name with the occupancy was chosen to replicate the fundamental pattern of the periodic table. The decision to add the d-spin state was based on the domain knowledge that the electron occupancy of the d-orbitals are generally responsible for the magnetic properties. The molecular orbitals were filled and spin states generated according to the model presented in [2].

Methodology

All programming was done in Python with the aid of the Pandas and scikit-learn (sklearn) packages. All class variables (molecule vs. metal calculation and molecule type) were split via the one hot encoding method, where a binary value was assigned (1 if belonging to the class, 0 if not). To compute the MM, three models were chosen. First a dummy model that simply predicted the mean of the data set was chosen as a preliminary baseline model purely for comparison. A multiple linear regression (MLR) was chosen as a second model due to its simplicity in implementing and interpreting. Finally, a random forest regression was chosen due to its general ability to avoid overfitting and for its overall ease of use.

All data was split using the train, test pack in sklearn with a training set of 99 samples and a testing set of 33 samples (75%/25% of the total sample, respectively). In all cases, a leave-one-out cross validation (loocv) was used to determine if the model was under or overfitting the data. The loocv was chosen in concert with the small data set to maximize the number of training samples in model generation. R^2 values are reported for the trained model on test data for unbiased comparison. Mean squared error (MSE) is the scoring function in all methods. The random forest model was first optimized over a range of hyperparameters using the grid search cross validation. Tested parameters included the number of trees, the maximum depth of the trees, the maximum number of features test, the minimum samples split, minimum samples per leave. Optimal hyperparameters can be found in the source code. All magnetic moments are reported in units of Bohr magneton (μB).

Multiple Linear Regression (MLR) Results

Table 1: Coefficients returned for the multiple linear regression model of the training data

Variable	U Level	Total D Electrons	D Spin State	F16MPc	MPc	Metal Calc	Molecule Calc
Coefficient	0.075	-0.094	2.633	-0.271	0.271	-0.402	0.402

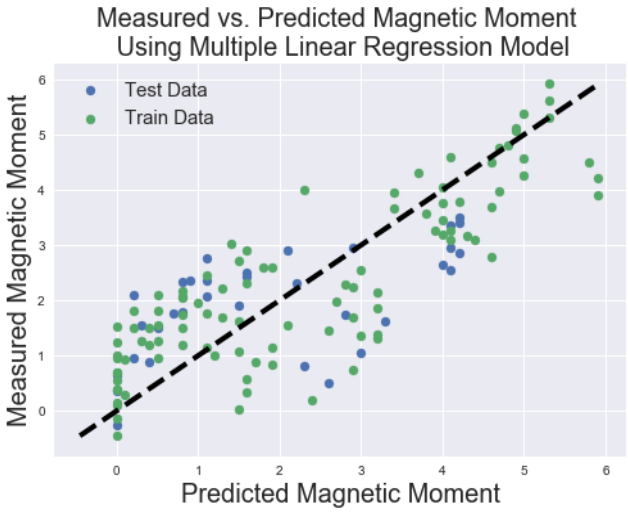

Figure 2: Measured vs. Predicted magnetic moment generated using the multiple linear regression model. Note the large cluster along the left side. These samples were predicted to have almost no magnetic moment, even though they do.

The MLR is a simple regression that takes multiple inputs and assumes a linear response to predict the outcome variable. Examining the table of coefficients generated by the model, a few things stand out. First, the d-spin state is by far the strongest estimator of magnetic moment; given what is known about magnetism, this should not come as any surprise. The MLR weighs the classes (both the molecule type and molecule vs. metal calculation) as negatives of each other as well, predicting a roughly constant shift between both sets of classes. Interestingly, the MLR model predicts the U level of the calculation has little correlation to the total magnetic moment of the resulting molecule or metal atom. This disagrees with the established physical connection between DFT and strongly correlated materials.. If one were to trust the MLR model, this would mean the various U calculations are generally superfluous. This main be a limitation of the linear model.

In table 3, the cross validation and R^2 scores are presented. The cross validation scores are both relatively high (> 1 μB), whereas the largest measured moment was 6 μB meaning an error on the order of ~15%. The mean being high likely indicates the linear model is biased in some fashion to accurately represent the interactions yielding magnetic moment (see further work). The high variance means the model is not generalizing well to new data and thus is consistently predicting values not in-line with experiments. The final R^2 value quantifies the weak relationship between the model's predictive capabilities and the known values.

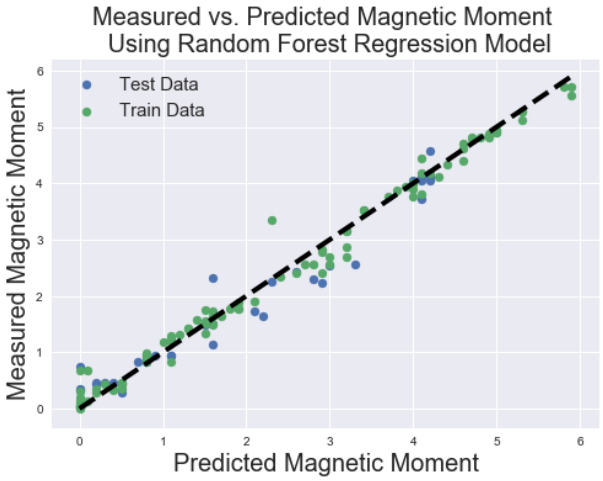

Figure 3: Measured vs. Predicted magnetic moment generated using the random forest regression model

In figure 2, we compare the measured and predicted MM. Beyond the small R^2, this figure exemplifies the need for caution in using this model to predict molecules for addition computational work or experimental synthesis. While it generally predicts the proper trend, it predicts several samples to have little to no MM when they have values ranging from zero to three μB. If we used this model to generate a candidate list of molecules, the list would neglect several candidates.

Random Forest Regression (RF) Results

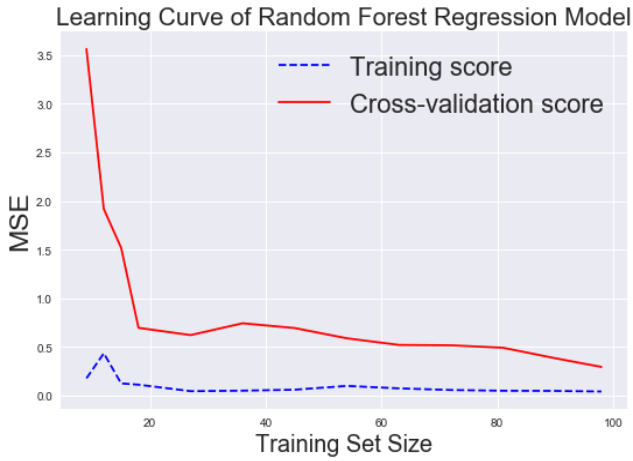

Figure 4: Learning Curve for the random forest regression Model. Scoring function is the mean squared error (y-axis). Note how the cross validation score quickly approaches the training score, indicating low variance.

In table 2, the feature importance for all the estimators is presented. Similar to the MLR, the RF weighs the d-spin state most heavily in predicting the MM. In contrast to the MLR, the RF weighs the total d-occupancy comparatively to the d-spin state, tin effect deriving a connection the structure of periodic table. Unlike the MLR, the U level is predicted to have a feature importance greater than the class of the molecule or the type of calculation.

Figure 3 compares the measured vs. the predicted MM using the RF model. Qualitatively, the model seems to very accurately predict the MM. With an R^2 of 0.93, the model accurately predicts the MM.

In table 3 are the cross validation and R^2 scores for the RF model. Immediately, the high R^2 on the test data raises concerns about overfitting. How should could the model fit such sparse data? This model should be interpreted with caution. Additionally, figure 4 presents the learning curve of the RF model. The training score hovering just above 0 MSE, even at with a small training set size, also supports the overfitting hypothesis.

However, the cross validation data does not support the idea of overfitting. The mean being relatively low indicates the model is not overly biased, but, more importantly, the variance being low means the model consistently is able to generalize to the unseen data and predict the value somewhat well. The learning curve also shows the cross validation MSE approaching the training error after ~20 samples. Lastly, random forest models are likely to overfit estimator data when using an excessive number of trees. In this case, the lower trees tend to fit superfluous features or noise. The solution to this would be to remove, or *prune,* the random forest. However, the grid search cross validation specifically explored the number of trees (50, 100, and 150) and found that the smallest number of trees (50) generated the best cross validation score and, by virtue of the smallest number of trees, least prone to overfitting.

Table 2: Feature Importance value returned for the random forest model of the training data

Variable	U Level	Total D Electrons	D Spin State	F16MPc	MPc	Metal Calc	Molecule Calc
Coefficient	0.036	0.32	0.545	0.0196	0.0171	0.0279	0.0337

Table 3: Cross variance (CV) and R^2 scores for the various model in this report

Model	**CV Mean**	**CV Variance**	**R-Squared**
Dummy	3.364	8.578	-0.0739
MLR	1.116	1.446	0.362
RF	0.296	0.49	0.942

Feature Engineering Discussion

Before discussing the confidence in the model, it is worth visiting the underlying assumptions in the model. The engineered feature is the +2 oxidation state and the subsequent filling of the d-orbitals. The assumption of the +2 oxidation state is generally true for MPcs with at least a half-filled d-orbital, but tends to break down for metals with less than half-filled. In several cases, multiple oxidation states have been reported for the same molecule. For example, MnPc and TiPc have been reported to have an oxidation state of +4 and +2 [3]. Additionally, the model assumed each Pc had the same fundamental electronic structure and thus filled the orbitals in the same order, as presented in [2], which may not be true.

Conclusions and Future Work

There are multiple avenues to expand the predictive powers of the model. In terms of modifying the existing model (either MLR or RF in this case), the largest worry is the poor correlation of the U parameter and the model ignoring it. Regarding the MLR, this would mean the model is essentially sees the same data point three time and is thus memorizing the answers from the training data. However the RF model returns a feature importance on the same order as the input classes. *This aspect necessitates further investigation before establishing full confidence in the RF model for predictive uses.* Regarding the MLR, fitting the total number of d-electrons to a higher order polynomial might allow the model to recognize that metals at both ends of the d-block would be relatively paramagnetic (nonmagnetic) while metals in the middle would likely predict higher magnetic moments. Given the small number of features, I do not believe any regularization technique (ridge or LASSO) would be unnecessary.

In terms of expanding the capabilities of the current model, the next step would be estimating the confidence interval of both models. This would allow the end user to know which candidate molecules the model has strong or weak confidence in. The random forest package in sklearn does not currently support this feature, but a confidence interval package has recently been developed and could be implemented [4].

In terms of drawing inference, the connections between the class of molecules is the most interesting. A general rule for predicting the change in MM from a non-fluorinated to fluorinated (or partially fluorinated) molecule would be provide novel insight to the larger material science community [5].

References

[1] M. M. Fadlallah, U. Eckern, A. H. Romero and U. Schwingenschlögl, New J. Phys., 2016, 18, 013003
[2] Bartolomé, J.; Luis, F.; Fernandez, J. F. Molecular Magnets; Springer: Heidelberg, Germany, 2014
[3] Jianzhuang Jiang, Functional Phthalocyanine Molecular Materials; Springer-Verlag Berlin Heidelberg, 2010
[4] S. Wager, T. Hastie, B. Efron, Journal of Machine Learning Research 2014 15, 1625-1651
[5] B. Milián-Medina, J.J. Gierschner, Phys Chem Liq, 8 (2017), pp. 91-101