Interpreting Graph Neural Networks with Myerson Values for Cheminformatics Approaches

Samuel K. R. Homberg
Institute of Medicinal and Pharmaceutical Chemistry
University of Münster

Janosch Menke
Institute of Medicinal and Pharmaceutical Chemistry
University of Münster

Garrett M. Morris
Department of Statistics
Oxford University

Oliver Koch
Institute of Medicinal and Pharmaceutical Chemistry
University of Münster
oliver.koch@uni-muenster.de

April 28, 2023

Abstract

Here we introduce a novel method to interpret the predictions of graph neural networks (GNNs) based on Myerson values from cooperative game theory. Myerson values are closely related to Shapley values and thus provide an interpretability approach similar to the SHAP values [1]. We developed the technique for applications in drug discovery, but it can be used with any graph. Using the GNN as a coalition game and the interpreted graph as the cooperation structure, the Myerson values determine the worth of each node of the graph. The worth of all nodes of the graph adds up to the predicted value of the model, allowing for a simple and intuitive interpretation of the prediction. To interpret predictions on molecular graphs we show visual explanations on molecular structures using two molecular datasets.

1 Introduction

Explainable artificial intelligence (XAI) has seen rapid development in recent years. The demand to interpret complicated machine learning methods has increased due to the increasing adoption of machine learning into public tools and services. Shapley additive explanations (SHAP) [1] is a popular XAI method that attributes an importance score to every feature. The method is based on the the Shapley value [2] from game theory to determine fair contributions and has been widely adopted to explain predictions of opaque machine learning models. The basic technique is model agnostic but has faster algorithms for the explanation of specific machine learning methods such as random forest models and deep neural networks. While Shapley or SHAP values are not intuitively interpretable, the method can be used to directly highlight regions of importance in images or important words in natural language processing, allowing for an easy interpretation of the explanations.

For the work with molecules, on the other hand, no direct visualizations have been implemented. However, SHAP has been investigated and used for the explanations of activity predictions [3]. But, because molecules are often represented using Morgan fingerprints [4], applying SHAP or other feature attribution methods result in a ranking of nondescript bits. The authors were able to map those bits back to the molecular structure and then could identify key-functional groups for their predictions. Morgan fingerprints are widely used molecular representations that have proven to be useful for virtual screening and as inputs for machine learning, as their fixed length makes mathematical operations on them trivial [5]. However, drawbacks to these fingerprints are the possibility of bit collisions, where multiple distinct structures activate the same bit in the fingerprint. It is further possible that a certain substructure occurs multiple times in a molecule, but will only be referred to by one bit being set, making it impossible to see whether these substructures are generally important or only in specific locations. The deconvolution necessary to obtain molecular substructures referring to the encoded bits lead thus to the more difficult interpretability of these fingerprints. Morgan fingerprints are generated from the molecular graph, and the rising popularity of graph neural networks (GNNs), that allow models to directly learn on graphs, obviate the need to generate these fingerprints. But, GNN input consists of a combination of multiple matrices, instead of a single vector or single matrix, as would be the case for other machine learning models [6]. This makes the explanation of GNN features challenging, even for model agnostic XAI methods, such as SHAP. Despite that, graphs offer the opportunity to directly correlate the structure of the graph with a prediction, a desirable property for drug discovery workflows.

In the last years, different methods to explain GNNs have been developed [7–17]. However, most of these methods often emphasize node classification over the, for chemists, more relevant graph classification. A notable exception to this is EdgeSHAPer [8]. EdgeSHAPer, like SHAP, makes use of the Shapley value from game theory but was developed with the explicit goal of drug discovery in mind, which is reflected in the author's proof-of-concept on activity predictions. The Shapley value has been used in the explanations of GNNs before [11], however, the EdgeSHAPer approach is novel in that the explanations focus on the edges instead of the nodes.

As others have pointed out [18], the Shapley value is not the only concept from cooperative game theory that could be used to explain machine learning predictions. An recent example is GStarX, which uses a novel, structure-aware solution concept to explain graph neural networks [17]. An older solution concept, less popular than the Shapley value (although acknowledged in [17]), is that of Myerson values [19]. Like Shapley values, the Myerson values attribute a "worth" to each player of a game, denoting their contribution to the final gain of the game. Myerson's idea was to restrict the players' possible cooperation depending on a series of bilateral agreements between players, i.e., a graph in which the players form the nodes and the edges show possible cooperation, resulting in a structure-aware solution concept. In these games, any value gained by cooperation is only accessible in coalitions, in which a link between the cooperating players exist.

A game restricted by the graph $G(V, E)$ with the nodes (vertices) V and the edges E is thus defined as:

$$(v/G)\,(S) = \sum_{T \in S/G} v(T). \tag{1}$$

Here, the coalition function $v(S)$ assigns a worth to a coalition of players S. But because of the graph-restriction of the game, only the worth of the *individual* connected subgraphs are allowed to form coalitions of possible players, the sum of these individual coalitions T is then taken as the worth of the coalition S, where S is restricted by the graph G. The set of graph restricted coalitions S/G (read as "S divided by G") is defined as:

$$S/G = \big\{\{i \mid i \text{ and } j \text{ are connected in } S \text{ by } G\} | j \in S\big\}. \tag{2}$$

To calculate the Myerson value is then a matter of simply taking the Shapley value of the newly constructed, graph-restricted game:

$$\text{My}_i(v, G) = \text{Sh}_i(v/G). \tag{3}$$

The Shapley value itself is calculated using the formula:

$$\text{Sh}_i\,(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(|N| - |S| - 1)!}{|N|!} \big(v\,(S \cup \{i\}) - v\,(S)\big). \tag{4}$$

where S refers to the coalition and N to the grand coalition of all players.

2 Results and Discussion

This process is straightforward with the exception of the null graph. The null graph, which has no nodes, is the graph of the empty coalition with no players. It is, however, programatically impossible to pass *nothing* to a GNN. For the correct calculation of the Myerson values, the worth of the null graph therefore had to be set manually. While it is possible to chosen any arbitrary value, to obey all axiom of the Myerson value, the worth of the null graph and, in extension, the worth of the empty set of players has to be set to zero.

Code to calculate the Myerson values as explanations for any GNN will be made available at `https://github.com/kochgroup` in the near future.

2.1 Proof of Concept

As an initial proof of concept, and inspired by Rasmussen et. al. [20], the calculated logarithmic octanol-water coefficient (clogP) was chosen to be predicted by graph neural networks, and these explanations were then to be

Table 1: Training results for two different GNNs used to predict the clogP of a molecule. Different metrics are reported: the mean absolute error (MAE), mean squared error (MSE), coefficient of determination (r^2) and Kendall's tau (τ).

GNN	MAE	MSE	r^2	τ
GCN	0.1895	0.0646	0.9530	0.8812
GAT	0.1592	0.0550	0.9464	0.8790

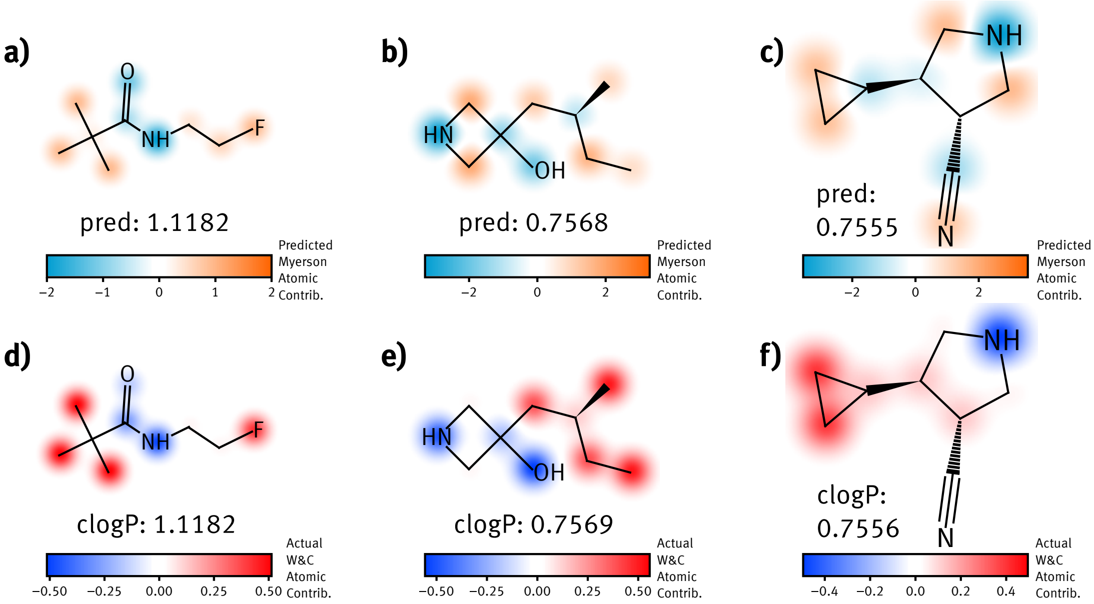

Figure 1: Three exemplary explanations (**a** - **c**) of some of the best logP predictions the GAT made contrasted with the ground truths (**d** - **f**). Atoms highlighted orange/light blue have a positive/negative contribution towards the logP prediction and atoms highlighted in red/blue have a positive/negative atomic logP Wildman and Crippen (W&C) contribution.

explained. Besides being a useful descriptor in drug discovery, as is evident by its inclusion in Lipinski's rule of five [21], the method to calculate logP developed by Wildman and Crippen [22] uses individual atom contributions adding up to the molecules logP. These atom contributions can be compared to the Myerson explanations as a ground truth to quickly show whether the explanations have merit.

The clogP was calculated for a dataset of small and light molecules curated from a subset of the ZINC database [23]. Two different message passing algorithms were used to build GNNs. The older, but still used graph convolutional network (GCN) [24] and the more recent graph attention network (GAT) [25]. Both networks showed good performance, the training results can be seen in Table 1.

To visualize the results, the Myerson values for each atom were mapped onto the molecular structure. For comparison, the ground truth atom contributions to clogP were also mapped to the molecular structure. Three examples of this are shown in Figure 1.

The Myerson values show a good overlap with the ground truth, even though the scale of Myerson values to ground truths is different. For some atoms, the Myerson value does not overlap with the ground truth atom contribution. For molecule **a**/**d**, the two methylene carbons next to the fluorine atom show a positive Myerson value, while the ground truth contribution is close to zero. The same is apparent for the methylene carbon atoms of the ring in the **b**/**e** molecule. Additionally, one other carbon atom has a negative Myerson value, in contrast to the positive ground truth contribution of the same carbon. Molecule **c**/**f** also has some of these flipped contributions and the atoms of the nitrile group show Myerson values seemingly canceling each other out instead of both atom Myerson values being close to zero, like the ground truth contributions. Overall, this initial investigation of the Myerson values is promising. However, when analyzing interpretations for individual predictions, it is difficult to distinguish whether the interpretation or the underlying model is responsible for apparent discrepancies.

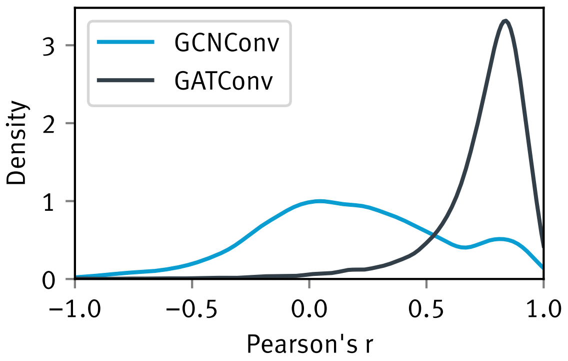

Figure 2: Density plot of the correlation coefficient between the Myerson explanations and the ground truth for the clogP predictions of the GCN and the GAT. Despite the similar predictive performance, the GAT shows better explanations.

Thus, a large number of molecules was analyzed by means of the correlation coefficient (Pearson's r) between the Myerson values and the ground truths. The GCN had a mean correlation coefficient 0.18 vs 0.73 for the GAT. Thus, despite their similar predictive performance (Table 1), the quality of the explanations is independent from the performance of the GNN. The distribution of the correlation coefficients over the entire test set can be seen in Figure 2.

We see two possible reasons for this phenomenon. First, it is possible that the GCN learned to predict clogP not based on the underlying atom contributions, but arrived at the prediction through different reasoning, which would be a version of the clever Hans effect [26]. Second, the generated explanations could be wrong. This can occur, because the subgraphs used in the calculation of the Myerson values can be unknown to the trained GNN and can also be invalid molecules. If the GNN generates nonsensical outputs for these subgraphs, this would throw the Myerson values off balance. While these subgraphs can be outside the trainigs domain of the GNN, so far no game theoretic solution concept is able to address the molecular validity of subsets. Figure 2 shows that the Myerson explanations can successfully distinguish between different learning methodologies. Wrong explanations because of invalid molecular fragments therefore play at most a minor role in the explanations and the main reason for unexpected explanations is the explained GNN. The Myerson explanations can therefore be seen as correct explanation.

3 Conclusion and Outlook

To conclude, using Myerson values in combination with a heatmap visualization is a promising new method to explain GNN predictions. In the investigated proof-of-concept, the explanations of calculated logP predictions had a high correlation with the underlying ground truths.

However, as of now the method is not fully developed. The exponential computational cost restricts the Myerson values to small graphs/molecules. Related methods addressed this by sampling subsets of coalitions. Additionally, a thorough comparison with other GNN explanation methods based on available benchmark datasets [27, 28].

4 Materials and Methods

4.1 Data

All molecules lighter than 200 Da were downloaded from the ZINC database, and subsequently all molecules with more than ten atoms were excluded from the selection, resulting in a dataset of 112 583 molecules. The molecules were transformed into graphs, with the bonds being transformed into edges and the heavy atoms into nodes. The dataset, as well as the code to generate it are available in the GitHub repository. Each node was characterized by a feature vector with the atom information of atomic number, degree, formal charge, hybridization and aromaticity. The graphs were labeled with clogP and split 80:20 into training and test set.

4.2 Models and Training

Two GNNs, with the same architecture but different graph convolutional layers (GCN and GAT) were built. The GNNs have three graph convolutional layers (size 256) alternated with ReLU functions, followed by a mean pooling layer and a final linear layer (size 256). Both models were trained for 150 epochs with a learning rate of 0.001 and a batch size of 64. Hyperparameters were not optimized. PyTorch [29] was used together with the PyTorch Geometric library [30] to create and train the models. The code is available in the GitHub repository.

4.3 Visualization

For the visualization of the explanations and ground truths, RDKit's [31] molecule drawing functions were modified.

References

[1] Lundberg, S. M.; Lee, S.-I. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc., 2017; pp 4765–4774.

[2] Shapley, L. S. 17. A Value for n-Person Games. In *Contributions to the Theory of Games (AM-28), Volume II*; Kuhn, H. W., Tucker, A. W., Eds.; Annals of Mathematics Studies; Princeton University Press: Princeton, NJ, 1953; Vol. Volume II; pp 307–318.

[3] Rodríguez-Pérez, R.; Bajorath, J. Interpretation of Compound Activity Predictions from Complex Machine Learning Models Using Local Approximations and Shapley Values. *J. Med. Chem.* **2020**, *63*, 8761–8777.

[4] Rogers, D.; Hahn, M. Extended-connectivity fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742–754.

[5] Cereto-Massagué, A.; Ojeda, M. J.; Valls, C.; Mulero, M.; Garcia-Vallvé, S.; Pujadas, G. Molecular fingerprint similarity search in virtual screening. *Methods* **2015**, *71*, 58–63.

[6] Veličković, P. Everything is Connected: Graph Neural Networks. *arXiv* **2023**, `https://arxiv.org/abs/2301.08210`.

[7] Lucic, A.; Ter Hoeve, M. A.; Tolomei, G.; De Rijke, M.; Silvestri, F. CF-GNNExplainer: Counterfactual Explanations for Graph Neural Networks. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*; Camps-Valls, G., Ruiz, F. J. R., Valera, I., Eds.; Proceedings of Machine Learning Research; PMLR, 2022; Vol. 151; pp 4499–4511.

[8] Mastropietro, A.; Pasculli, G.; Feldmann, C.; Rodríguez-Pérez, R.; Bajorath, J. EdgeSHAPer: Bond-centric Shapley value-based explanation method for graph neural networks. *iScience* **2022**, *25*, 105043.

[9] Ying, Z.; Bourgeois, D.; You, J.; Zitnik, M.; Leskovec, J. GNNExplainer: Generating Explanations for Graph Neural Networks. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc., 2019; Vol. 32.

[10] Huang, Q.; Yamada, M.; Tian, Y.; Singh, D.; Yin, D.; Chang, Y. GraphLIME: Local Interpretable Model Explanations for Graph Neural Networks. *arXiv* **2020**, `https://arxiv.org/abs/2001.06216`.

[11] Duval, A.; Malliaros, F., D., GraphSVX: Shapley Value Explanations for Graph Neural Networks. *arXiv* **2021**, `https://arxiv.org/abs/2104.10482`.

[12] Schlichtkrull, M. S.; Cao, N. D.; Titov, I. Interpreting Graph Neural Networks for NLP With Differentiable Edge Masking. In *International Conference on Learning Representations*; 2021.

[13] Yuan, H.; Yu, H.; Wang, J.; Li, K.; Ji, S. On Explainability of Graph Neural Networks via Subgraph Explorations. In *Proceedings of the 38th International Conference on Machine Learning*; Meila, M., Zhang, T., Eds.; Proceedings of Machine Learning Research; PMLR, 2021; Vol. 139; pp 12241–12252.

[14] Luo, D.; Cheng, W.; Xu, D.; Yu, W.; Zong, B.; Chen, H.; Zhang, X. Parameterized Explainer for Graph Neural Network. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*; NIPS'20; Curran Associates Inc.: Red Hook, NY, USA, 2020.

[15] Vu, M.; Thai, M. T. PGM-Explainer: Probabilistic Graphical Model Explanations for Graph Neural Networks. In *Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H., Eds.; Curran Associates, Inc., 2020; Vol. 33; pp 12225–12235.

[16] Yuan, H.; Tang, J.; Hu, X.; Ji, S. XGNN: Towards Model-Level Explanations of Graph Neural Networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*; KDD '20; Association for Computing Machinery: New York, NY, USA, 2020; p 430–438.

[17] Zhang, S.; Liu, Y.; Shah, N.; Sun, Y. GStarX: Explaining Graph Neural Networks with Structure-Aware Cooperative Games. In *Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc., 2022; Vol. 35; pp 19810–19823.

[18] Rozemberczki, B.; Watson, L.; Bayer, P.; Yang, H.-T.; Kiss, O.; Nilsson, S.; Sarkar, R. The Shapley Value in Machine Learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*; Raedt, L. D., Ed.; International Joint Conferences on Artificial Intelligence Organization, 2022; pp 5572–5579, Survey Track.

[19] Myerson, R. B. Graphs and Cooperation in Games. *Mathematics of Operations Research* **1977**, *2*, 225–229.

[20] Rasmussen, M. H.; Christensen, D. S.; Jensen, J. H. Do machines dream of atoms? A quantitative molecular benchmark for explainable AI heatmaps. *ChemRxiv* **2022**, `https://doi.org/10.26434/chemrxiv-2022-gnq3w-v2`.

[21] Lipinski, C. A.; Lombardo, F.; Dominy, B. W.; Feeney, P. J. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Adv. Drug Deliv. Rev.* **2001**, *46*, 3–26.

[22] Wildman, S. A.; Crippen, G. M. Prediction of Physicochemical Parameters by Atomic Contributions. *J. Chem. Inf. and Comput. Sci.* **1999**, *39*, 868–873.

[23] Irwin, J. J.; Shoichet, B. K. ZINC - a free database of commercially available compounds for virtual screening. *J. Chem. Inf. Model.* **2005**, *45*, 177–182.

[24] Kipf, T. N.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations*; 2017.

[25] Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph Attention Networks. In *International Conference on Learning Representations*; 2018.

[26] Lapuschkin, S.; Wäldchen, S.; Binder, A.; Montavon, G.; Samek, W.; Müller, K.-R. Unmasking Clever Hans predictors and assessing what machines really learn. *Nat. Commun.* **2019**, *10*, 1096.

[27] Matveieva, M.; Polishchuk, P. Benchmarks for interpretation of QSAR models. *J. Cheminform.* **2021**, *13*, 41.

[28] Jiménez-Luna, J.; Skalic, M.; Weskamp, N. Benchmarking Molecular Feature Attribution Methods with Activity Cliffs. *J. Chem. Inf. Model* **2022**, *62*, 274–283.

[29] Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc., 2019; Vol. 32.

[30] Fey, M.; Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric. In *ICLR 2019 Workshop on Representation Learning on Graphs and Manifolds*; 2019.

[31] Landrum, G. RDKit: Open-Source Cheminformatics Software. `http://www.rdkit.org/`.