Error-controlled interaction discovery in deep neural networks

Winston Chen[*]
University of Michigan
chenwt@umich.edu

Yifan Jiang[*]
University of Waterloo
yifan.jiang@uwaterloo.ca

William Stafford Noble
University of Washington
william-noble@uw.edu

Yang Young Lu[†]
University of Waterloo
yanglu@uwaterloo.ca

Abstract

The complexity of deep neural networks (DNNs) makes them powerful but also makes them challenging to interpret, hindering their applicability in error-intolerant domains. Existing methods attempt to reason about the internal mechanism of DNNs by identifying feature interactions that influence prediction outcomes. However, such methods typically lack a systematic strategy to prioritize interactions while controlling confidence levels, making them difficult to apply in practice for scientific discovery and hypothesis validation. In this paper, we introduce a method, Diamond, to address this limitation using knockoffs, which are dummy variables that are designed to mimic the dependence structure of a given set of features while being conditionally independent of the response. Together with a novel DNN architecture involving a pairwise-coupling layer, Diamond jointly controls the false discovery rate (FDR) and maximizes statistical power. In addition, we identify a challenge in correctly controlling FDR using off-the-shelf feature interaction importance measures. Diamond overcomes this challenge by proposing a calibration procedure applicable to any existing interaction importance measures to maintain FDR control at the target level. Finally, we validate the effectiveness of Diamond through extensive experiments on simulated and real datasets.

1 Introduction

Deep neural networks (DNNs) have emerged as a critical tool in many application domains, largely due to their ability to detect subtle relationships and patterns within complex data [1]. While DNNs' complexity contributes to their power, it also makes them challenging to interpret, leaving users with few clues about the underlying mechanisms. Consequently, this "black box" nature of DNNs has hindered their applicability in error-intolerant domains like healthcare and finance. Stakeholders, such as clinicians, need to understand why and how the models make predictions before making important decisions, such as disease diagnosis [2]. Importantly, without understanding the internal mechanisms, DNNs cannot be effectively used for making data-driven scientific discoveries, which are crucial for gaining human-understandable insights and driving successful innovation [3].

To enhance the interpretability of DNNs for better data-driven scientific discoveries, many methods have been developed to elucidate the internal mechanisms of these models [4]. These methods help to elucidate how individual features influence prediction outcomes by assigning an importance score to each feature so that higher scores indicate greater relevance to the prediction [5, 6, 7, 8, 9].

[*] Equal contribution.

[†] Corresponding author.

Interpretable AI: Past, Present and Future Workshop at NeurIPS 2024

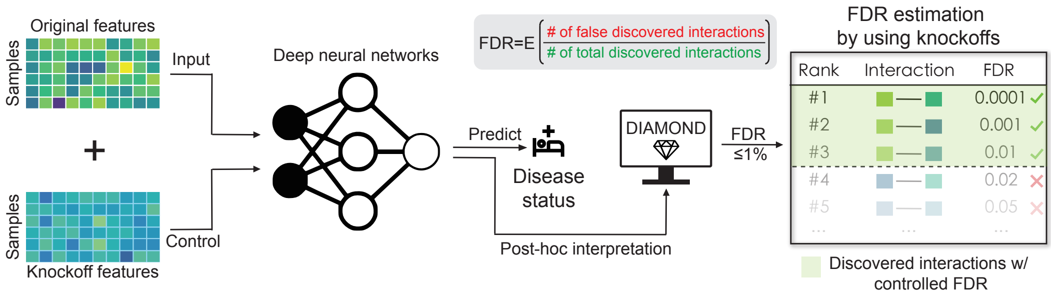

Figure 1: **Overview of Diamond.** Diamond trains DNNs using both the original features and their knockoff counterparts as inputs. Diamond quantifies feature interactions from trained DNNs and produces a ranked list of these interactions with estimated FDR, allowing users to confidently determine a cutoff threshold based on their desired confidence level.

However, these univariate interpretations overlook DNNs' primary advantage: their ability to model complex interactions between features in a data-driven way. In fact, input features usually do not work individually within a DNN but cooperate with other features to make inferences jointly [10]. For example, it is well established in biology that genes do not operate in isolation but work together in co-regulated pathways with additive, cooperative, or competitive interactions [11]. Additionally, gene-gene, gene-disease, gene-drug, and gene-environment interactions are critical in explaining genetic mechanisms, diseases, and drug effects [12].

Recognizing the limitations of univariate interpretations, efforts have been made to extend these interpretations to discover feature interactions. Briefly, these methods attribute the prediction influence to feature pairs and rank candidate feature pairs from a trained DNN, with highly ranked pairs indicating higher importance [10, 13, 14, 15, 16, 17, 18, 19, 20]. However, it is important to note that these approaches characterize feature pairs where both features are simultaneously important for a model's prediction rather than capturing the synergistic or interactive effects between the two features [21]. Furthermore, the induced ranked list of feature pairs must be cut off at a certain confidence level for use in scientific discovery and hypothesis validation [10]. However, selecting this threshold is typically under user control, subject to arbitrary choices, and without scientific rigor. Worse still, existing methods are sensitive to perturbations, in the sense that even unperceivable, random perturbations of the input data may lead to dramatic changes in the importance ranking [22, 23, 24].

From a practitioner's perspective, a given set of discovered feature interactions is scientifically valuable only if a systematic strategy exists to prioritize and select relevant interactions in a robust and error-controlled manner, even in the presence of noise. Although many methods have been developed for feature interaction discovery, we are unaware of any previous attempts to conduct discovery while explicitly estimating and controlling the discovery error. Without this, accurate and reliable findings cannot be achieved. In this study, we introduce a pioneering error-controlled interaction discovery method named Diamond (Discovering InterActions in Machine learning mOdels with a coNtrolleD error rate). Here, the error is quantified by the false discovery rate (FDR) [25], which informally represents the expected proportion of falsely discovered interactions among all discovered interactions. A false discovery is a feature interaction that is discovered but not truly relevant.

The key novelty of Diamond lies in two aspects (Fig. 4). Firstly, Diamond achieves false discovery rate (FDR) control by leveraging the model-X knockoffs framework [26, 27]. The core idea of this framework is to generate dummy features that perfectly mimic the empirical dependence structure among the original features while being conditionally independent of the response given the original features. Secondly, we discover that naively using off-the-shelf feature interaction importance measures cannot correctly control the FDR. To address this issue, we propose a calibration procedure to distill non-additive or interactive effects from the reported interaction importance measures from existing methods, thereby maintaining FDR control at the target level. Additionally, we have applied Diamond to simulated and real datasets to demonstrate its empirical utility. Practically speaking, Diamond paves the way for the broader deployment of DNNs in scientific discovery and hypothesis generation, potentially leading to significant breakthroughs.

2 Background

2.1 Problem setup

Consider a supervised learning task where we have n independent and identically distributed (i.i.d.) samples $\mathbf{X} = \{x_i\}_{i=1}^n \in \mathbb{R}^{n \times p}$ and $\mathbf{Y} = \{y_i\}_{i=1}^n \in \mathbb{R}^{n \times 1}$, denoting the p-dimensional feature matrix and the corresponding response, respectively. The task is modeled by a black-box function $f : \mathbb{R}^p \mapsto \mathbb{R}$, parameterized by a DNN that maps from the input $x \in \mathbb{R}^p$ to the response $y \in \mathbb{R}$. When modeling the task, the function f learns non-additive feature interactions from the data, of which each interaction $\mathcal{I} \subset \{1, \cdots, p\}$ is a subset of interacting features. This work focuses on pairwise interactions, i.e., $|\mathcal{I}| = 2$. We say that $\mathcal{I}$ is a non-additive interaction of function f if and only if f cannot be decomposed into an addition of $|\mathcal{I}|$ subfunctions f_i, each of which excludes a corresponding interaction feature [28, 10], *i.e.,* $f(x) \neq \sum_{i \in \mathcal{I}} f_i\left(x_{\{1,\cdots,p\}\setminus i}\right)$. For example, the multiplication between two features x_i and x_j is a non-additive interaction because it cannot be decomposed into a sum of univariate functions, *i.e.,* $x_i x_j \neq f_i(x_j) + f_j(x_i)$. Assume that there exists a set of interactions $\mathcal{S} = \{\mathcal{I}_1, \mathcal{I}_2, \cdots\}$ such that conditional on interactions $\mathcal{S}$, the response $\mathbf{Y}$ is independent of interactions in the complement $\mathcal{S}^c = \{1, \cdots, p\} \times \{1, \cdots, p\} \setminus \mathcal{S}$. We aim to discover feature interactions in $\mathcal{S}$ without erroneously reporting too many incorrect ones in $\mathcal{S}^c$.

2.2 FDR control with knockoffs

Diamond achieves FDR control by leveraging the model-X knockoffs framework [26, 27], which was proposed in the setting of error-controlled feature selection. The core idea is to generate dummy features that perfectly mimic the empirical dependence structure among the original features but are conditionally independent of the response given the original features. Briefly speaking, the knockoff filter achieves FDR control in two steps: (1) construction of knockoff features and (2) filtering using knockoff statistics. We review details on these two steps in Appendix A.

2.3 DNN feature interaction measurement

Diamond is compatible with any model-agnostic feature interaction interpretation methods, which provides a ranked order of candidate interactions without assuming any specific model architecture([14, 15, 29, 16, 17, 18, 19, 20]). In our experiments, we use Expected Hessian [16], a state-of-the-art method, as the interaction interpretation method Diamond leverages.

3 Approach

3.1 Knockoff-tailored DNN

Diamond integrates the idea of knockoff filter with DNNs to enable interaction detection while maintaining controlled FDR. Our method builds prior work, DeepPINK [30], by leveraging a knockoff-tailored DNN architecture that combines any off-the-shelf DNN with a plugin pairwise-coupling input layer. A detailed description of the DeepPINK architecture can be found in Appendix B.

3.2 Measuring non-additive interactive effect

As a key precursor to FDR estimation, Diamond quantifies feature interactions from trained DNNs and produces a ranked list of these interactions, with higher-ranked interactions indicating greater importance. For notational simplicity, we use indices for both original features and knockoffs as $\{1, 2, \cdots, 2p\}$, with $\{1, \cdots, p\}$ and $\{p+1, \cdots, 2p\}$ corresponding to the original features and their respective knockoff counterparts. Here, we define $\mathbf{E}^{\text{2D}} = [e_{ij}]_{i,j=1}^{2p} \in \mathbb{R}^{2p \times 2p}$ as a reported interaction importance measure from existing methods. There are many feature interaction importance measures available for $\mathbf{E}^{\text{2D}}$, each attributing the prediction influence to feature pairs in different ways. However, it is important to note that such measures favor pairs where both features are simultaneously important for a model's prediction, rather than capturing the true non-additive or interactive effects between the two features [21]. Further supported by simulation studies (Fig. 2**b**), we observed off-the-shelf feature interaction importance measure tend to assign higher interaction scores to two marginally important but non-interacting features compared to two random ones, even though neither pair has a real interaction, leading to the failure of FDR control.

The direct reason the interaction importance measure failed to control FDR is that it violated the knockoff filter's assumption. Specifically, the knockoff filter requires that the importance scores of knockoff-involving interactions and false interactions have a similar distribution. To resolve this issue, we introduce a calibration procedure to be applied on top of existing interaction importance measures. Specifically, we consider that a reported interaction importance measure from existing methods comprises a mixture of several factors: prediction-dependent marginal effects for individual features, prediction-independent feature biases, independent random noise, and potential non-additive interactive effects between feature pairs. Thus, the reported interaction between features i and j is represented as:

$$e_{ij} = s_{ij} + g_i(e_i) + g_j(e_j) + b(I_{ij}) + \varepsilon_{ij} \tag{1}$$

Where $\varepsilon_{ij} \in \mathbb{R}$ is random noise independent of both features and predictions, $e_{ij} \in \mathbb{R}$ and $e_i, e_j \in \mathbb{R}$ are reported pairwise and univariate feature importance measures that are dependent on the model's predictions, respectively. The functions $g_i, g_j : \mathbb{R} \mapsto \mathbb{R}$ adapt univariate feature importance to be compatible with feature interaction importance. The function $b : \mathbb{R}^{2p} \mapsto \mathbb{R}$ models the feature-specific biases that are independent of the model's predictions, where $I_{ij} \in \{0,1\}^{2p}$ indicates the presence of feature i and j. We aim to identify s_{ij}, the potential non-additive interactive effects between features i and j.

We formulate the identification of interactive effects s_{ij} as the residuals of a regression task:

$$\min_{b, g_1, g_2, \cdots} \sum_{i<j} w_{ij} \cdot \|e_{ij} - g_i(e_i) - g_j(e_j) - b(I_{ij})\|^2 \tag{2}$$

where $w_{ij} > 0$ is the conditional probability of being either original-only (*i.e.,* $i, j \leq p$) or knockoff-involving (*i.e.,* $i > p$ or $j > p$) feature pairs given two univariate feature importance measures e_i and e_j, estimated by a logistic regression model [31]. The rationale is based on the important observation that most feature pairs do not exhibit non-additive interactions, especially those involving knockoff features. Therefore, we want to focus more on potential non-additive interactions that have large univariate feature importance, as important interactions naturally consist of significant marginal features. In this study, we parameterize the functions $b : \mathbb{R}^{2p} \mapsto \mathbb{R}$ and $g_i, g_j : \mathbb{R} \mapsto \mathbb{R}$ using generalized additive models and optimize Eq. 2 using the pyGAM library [32].

3.3 FDR control for interactions

After calculating the non-additive interactive effects using Eq. 2, we denote the resultant set of interactive effects as $\Gamma = \{s_{ij} | i < j, i \neq j - p\}$. We arrange Γ in decreasing order and select interactions for which the interactive effect, Γ_j, exceeds some threshold, T. This selection ensures that the chosen interactions adhere to a desired FDR level $q \in (0, 1)$.

However, the heterogeneous interactions, which include original-only and knockoff-involving interactions, introduce a point of complexity. The latter further comprises original-knockoff, knockoff-original, and knockoff-knockoff interactions. Following the strategy outlined by [33], the threshold T is determined by:

$$T = \min \left\{ t \in \mathcal{T},\ \frac{|\{j : \Gamma_j \geq t, j \in \mathcal{D}\}| - 2 \cdot |\{j : \Gamma_j \geq t, j \in \mathcal{DD}\}|}{|\{j : \Gamma_j \geq t, j \notin \mathcal{D} \text{ and } j \notin \mathcal{DD}\}|} \leq q \right\} \tag{3}$$

where $\mathcal{D}$ and $\mathcal{DD}$ respectively denote the sets of interactions that include at least one knockoff feature and both knockoff features, while $\mathcal{T}$ refers to the set of unique nonzero values present in Γ.

4 Results

4.1 Simulated Data Analysis

We started by evaluating the performance of Diamond on simulated datasets, assessing its ability to identify important non-additive interactions while controlling the FDR. We benchmarked Diamond on a test suite of 10 simulated datasets generated by different simulation functions proposed by [10]. These datasets contain a mixture of univariate functions and multivariate interactions, exhibiting varied order, strength, and nonlinearity (Appendix C). Since our goal is to detect pairwise interactions, high-order interaction functions (*e.g.,* $F(x_1, x_2, x_3) = x_1 x_2 x_3$) are decomposed into pairwise interactions (*e.g.,* (x_1, x_2), (x_1, x_3), and (x_2, x_3)) to serve as the ground truth.

4.1.1 Experimetnal Setup

Following the settings used in [10], we employed a sample size of $n = 20,000$, equally divided into training and test sets. In addition, the number of features is set at $p = 30$, and all features are sampled randomly from a continuous uniform distribution, $U(0, 1)$. Only the first 10 out of 30 features

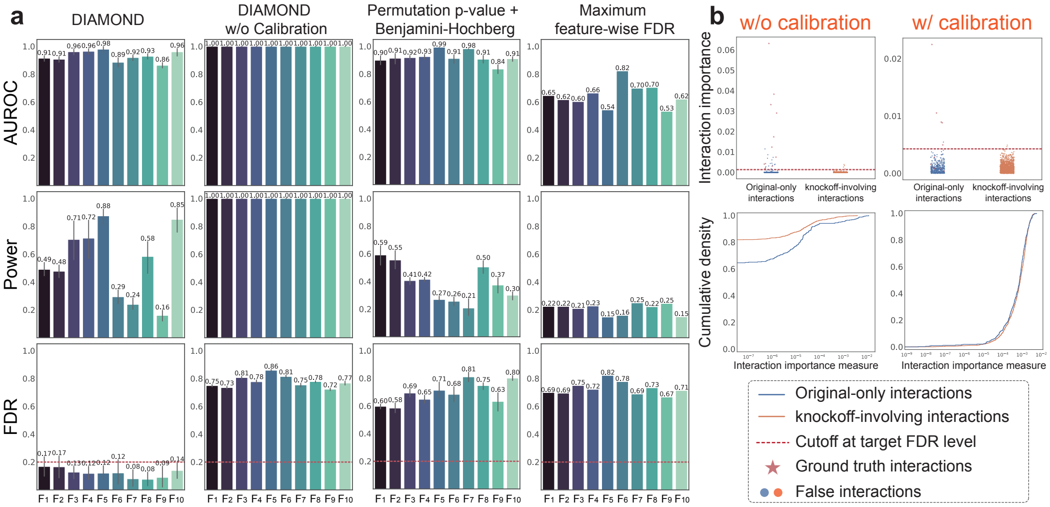

Figure 2: Evaluating Diamond and baseline methods on a test suite of 10 simulated datasets, in terms of AUROC, FDR, and power. Error bars correspond to the 95% confidence interval estimated across 20 repetitions. The figure comprises five columns, each presenting the evaluation results for one interaction selection method. (A) Diamond identified important non-additive interactions with controlled FDR across all 10 simulation functions. The calibration procedure is critical; without it, the FDR cannot be controlled. Baseline methods fail to control the FDR correctly, rendering the reported high power and AUROC invalid. (B) The reported importance of interaction from existing methods in simulation function F_1 reveals a clear distribution disparity between original-only interactions and those involving knockoffs.

contribute to the corresponding response, while the remaining features serve as noise, increasing the task's complexity. For robustness, we repeated the experiment 20 times for each simulated dataset using different random seeds. Each repetition involved data generation, knockoff generation using KnockoffsDiagnostics [34], DNN training, and interaction-wise FDR estimation. We reported the mean performance with 95% confidence intervals for all simulation settings, fixing the target FDR level at $q = 0.2$.

4.1.2 Simulation Data Results

Our analysis shows that Diamond consistently identifies important non-additive interactions with controlled FDR across all simulation functions (Fig. 2**a**). We discovered that the proposed calibration procedure (Sec 3.2) is critical; without it, the FDR cannot be controlled by naively using reported interaction importance values from existing methods.

Additionally, we verified whether alternative baseline methods can accurately identify important non-additive interactions with controlled FDR. We compared two baseline methods for FDR estimation: one based on permutation-based interaction-wise p-values coupled with the Benjamini-Hochberg procedure, and the other representing interaction-wise FDR as the aggregation of feature-wise FDR (See details in Appendix D). Our analysis shows that neither method correctly controls the FDR (Fig. 2**a**). This greatly reduces the utility of these methods, despite their reported high power and AUROC.

To gain insight into the FDR control failure in the absence of calibration, we conducted a qualitative comparison assessing interaction importance before and after calibration using the simulation function F_1 (Fig. 2**b**). The primary cause of the FDR control failure appears to lie in the distributional disparity between interactions only involving original features (original-only) and interactions involving knockoffs (knockoff-involving). This observation suggests violating the knockoff filter's assumption in controlling the FDR (See discussion in Sec 3.2). The proposed calibration procedure mitigates the disparity by extracting non-additive interactive effects from the reported interaction importance measures, thereby enhancing the utility of knockoff-involving interactions as a negative control for FDR estimation.

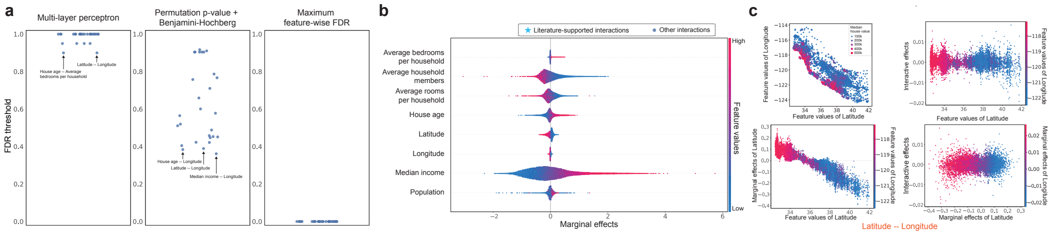

Figure 3: Evaluating Diamond on the California housing dataset. (a) Diamondis compared against two baseline methods. Each possible interaction is measured by the minimum FDR threshold cutoff at which it is selected, with the top interaction annotated. (b) Each feature contributes differently to predicting housing prices as measured by the Expected Gradient scores in the MLP model. It is worth mentioning that the most marginally important features do not necessarily result in important feature interactions, as anticipated in Diamond's design. (c) The top interaction between latitude and longitude is qualitatively evaluated from four aspects: the contribution of feature values to response prediction, the marginal importance measure, the interaction importance measure, and the contribution of the marginal important measures to the interaction importance measure.

4.2 Real data Analysis

We then evaluated the performance of Diamond on the California housing dataset [35], assessing its ability to identify important interactions in influencing California housing prices in 1990. This dataset contains $n = 20640$ samples, each characterized by eight standardized features, including average bedrooms per household, average household members, average rooms per household, house age, latitude, longitude, median income, and population (Fig. 3(b)). The task is constructing a DNN to predict the housing price given these eight features. We applied Diamond and the two baseline methods to predict housing prices and discover important interactions influencing housing prices. For robustness, we repeated each experiment 20 times using different random seeds. Each repetition involved knockoff generation, DNN training, and interaction-wise FDR estimation.

4.2.1 Real Data Results

Our analysis shows that Diamond identifies the important interaction between latitude and longitude (Fig. 3(b)). This interaction is supported by the well-known fact that housing prices are strongly dependent on geographical location, which is jointly determined by latitude and longitude.

Our qualitative evaluation further supports this interaction (Fig. 3(c)). Specifically, we examined the interaction between latitude and longitude in four ways: the contribution of feature values to response prediction, the marginal importance measure, the interaction importance measure, and the contribution of the marginal importance measures to the interaction importance measure. This analysis showed that latitudes and longitudes corresponding to the coastal area of California tend to have higher housing prices than those corresponding to the inland area. Meanwhile, higher interactive effects can be found in areas with lower latitude and higher longitude, which corresponds to the increase in housing prices around the Bay Area in northern California. Higher interactive effects can also be seen when latitude and longitude are lower, corresponding to the LA area in southern California, another region with higher housing prices than the surrounding area.

Finally, we find that Diamond does not report a lower FDR for interactions comprising two marginally important features. Specifically, latitude and longitude are both individually not important for predicting housing prices compared to other features; however, jointly, they have a strong interactive effect as estimated by Diamond.

5 Conclusion

In conclusion, Diamond enables error-controlled detection of feature interactions in any DNNs. The flexibility of Diamond makes it widely applicable in high-stakes and error-intolerant domains where interpretability and statistical rigor are needed. We believe that this powerful tool will facilitate the broader deployment of DNNs in scientific discovery and hypothesis validation.

References

[1] Z. Obermeyer and E. J. Emanuel. Predicting the future–big data, machine learning, and clinical medicine. *The New England Journal of Medicine*, 375(13):1216, 2016.

[2] Z. C. Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[3] A. Agrawal, J. McHale, and A. Oettl. Artificial intelligence and scientific discovery: A model of prioritized search. *Research Policy*, 53(5):104989, 2024.

[4] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278, 2021.

[5] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[6] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, 2017.

[7] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 2017.

[8] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, 2017.

[9] Y. Y. Lu, W. Guo, X. Xing, and W. S. Noble. DANCE: Enhancing saliency maps using decoys. In *International Conference on Machine Learning*, 2021.

[10] M. Tsang, D. Cheng, and Y. Liu. Detecting statistical interactions from neural network weights. *International Conference on Learning Representations*, 2018.

[11] Y. Y. Lu and W. S. Noble. A wider field of view to predict expression. *Nature Methods*, 18(10):1155–1156, 2021.

[12] D. S. Watson. Interpretable machine learning for genomics. *Human Genetics*, 141(9):1499–1513, 2022.

[13] M. Tsang, H. Liu, S. Purushotham, P. Murali, and Y. Liu. Neural interaction transparency (NIT): Disentangling learned interactions for improved interpretability. *Advances in Neural Information Processing Systems*, 31, 2018.

[14] T. Cui, P. Marttinen, and S. Kaski. Recovering pairwise interactions using neural networks. *arXiv preprint arXiv:1901.08361*, 2019.

[15] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, 2020.

[16] J. D. Janizek, P. Sturmfels, and S.-I. Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *Journal of Machine Learning Research*, 22:104:1–104:54, 2021.

[17] M. Sundararajan, K. Dhamdhere, and A. Agarwal. The shapley taylor interaction index. In *International Conference on Machine Learning*, pages 9259–9268. PMLR, 2020.

[18] C. Chang, R. Caruana, and A. Goldenberg. NODE-GAM: Neural generalized additive model for interpretable deep learning. *International Conference on Learning Representations*, 2022.

[19] S. Lerman, C. Venuto, H. Kautz, and C. Xu. Explaining local, global, and higher-order interactions in deep learning. In *International Conference on Computer Vision*, pages 1224–1233, 2021.

[20] H. Zhang, Y. Xie, L. Zheng, D. Zhang, and Q. Zhang. Interpreting multivariate shapley interactions in DNNs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10877–10886, 2021.

[21] M. Tsang, J. Enouen, and Y. Liu. Interpretable artificial intelligence through the lens of feature interaction. *arXiv preprint arXiv:2103.03103*, 2021.

[22] A. Ghorbani, A. Abid, and J. Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3681–3688, 2019.

[23] P.-J. Kindermans, S. Hooker, J. Adebayo, M. Alber, K. T. Schütt, S. Dähne, D. Erhan, and B. Kim. The (un)reliability of saliency methods. *Explainable AI: Interpreting, explaining and visualizing deep learning*, pages 267–280, 2019.

[24] Y. Y. Lu, T. C. Yu, G. Bonora, and W. S. Noble. ACE: Explaining cluster from an adversarial perspective. In *International Conference on Machine Learning*, 2021.

[25] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B*, 57:289–300, 1995.

[26] R. F. Barber and E. J. Candès. Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43(5):2055–2085, 2015.

[27] E. J. Candès, Y. Fan, L. Janson, and J. Lv. Panning for gold: Model-X knockoffs for high-dimensional controlled variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(3):551–577, 2018.

[28] D. Sorokina, R. Caruana, M. Riedewald, and D. Fink. Detecting statistical interactions with additive groves of trees. In *International Conference on Machine learning*, pages 1000–1007, 2008.

[29] M. Tsang, S. Rambhatla, and Y. Liu. How does this interaction affect me? interpretable attribution for feature interactions. *Advances in Neural Information Processing Systems*, 33:6147–6159, 2020.

[30] Y. Y. Lu, Y. Fan, J. Lv, and W. S. Noble. DeepPINK: reproducible feature selection in deep neural networks. In *Advances in Neural Information Processing Systems*, 2018.

[31] D. A. Freedman and R. A. Berk. Weighting regressions by propensity scores. *Evaluation Review*, 32(4):392–409, 2008.

[32] D. Serven and C. Brummitt. pyGAM: Generalized additive models in python, 03 2018.

[33] T. Walzthoeni, M. Claassen, A. Leitner, F. Herzog, S. Bohn, F. Förster, M. Beck, and R. Aebersold. False discovery rate estimation for cross-linked peptides identified by mass spectrometry. *Nature Methods*, 9(9):901–903, 2012.

[34] A. Blain, B. Thirion, J. Linhart, and P. Neuvial. When knockoffs fail: diagnosing and fixing non-exchangeability of knockoffs. *arXiv preprint arXiv:2407.06892*, 2024.

[35] R Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297, 1997.

[36] Tianyu Cui, Khaoula El Mekkaoui, Jaakko Reinvall, Aki S Havulinna, Pekka Marttinen, and Samuel Kaski. Gene–gene interaction detection with deep learning. *Communications Biology*, 5(1):1238, 2022.

Appendices

A Review on error-controlled feature selection with the knockoff filter

Knockoff filter achieves FDR control in two steps: (1) constructing knockoff features and (2) filtering using knockoff statistics.

For the first step, the knockoff features are defined as follows:

Definition 1 (Model-X knockoff [27]). *The model-X knockoff features for the family of random features $\mathbf{X} = (X_1, \dots, X_p)$ are a new family of random features $\tilde{\mathbf{X}} = (\tilde{X}_1, \dots, \tilde{X}_p)$ that satisfy two properties:*

1. $(\mathbf{X}, \tilde{\mathbf{X}})_{swap(\mathcal{S})} \overset{d}{=} (\mathbf{X}, \tilde{\mathbf{X}})$ *for any subset* $\mathcal{S} \subset \{1, \dots, p\}$*, where swap*$(\mathcal{S})$ *means swapping* X_j *and* $\tilde{X}_j$ *for each* $j \in \mathcal{S}$ *and* $\overset{d}{=}$ *denotes equal in distribution, and*
2. $\tilde{\mathbf{X}} \perp\!\!\!\perp \mathbf{Y} | \mathbf{X}$*, i.e.,* $\tilde{\mathbf{X}}$ *is independent of response* $\mathbf{Y}$ *given feature* $\mathbf{X}$.

According to Definition 1, the construction of the knockoffs must be independent of the response $\mathbf{Y}$. Thus, if we can construct a set $\tilde{X}$ of model-X knockoff features properly, then by comparing the original features with these control features, FDR can be controlled at target level q. In the Gaussian setting, *i.e.,* $\mathbf{X} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$ with covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{p \times p}$, the model-X knockoff features can be constructed easily:

$$\tilde{\mathbf{X}}|\mathbf{X} \sim N\big(\mathbf{X} - \mathrm{diag}\{\mathbf{s}\}\boldsymbol{\Sigma}^{-1}\mathbf{X}, 2\mathrm{diag}\{\mathbf{s}\} - \mathrm{diag}\{\mathbf{s}\}\boldsymbol{\Sigma}^{-1}\mathrm{diag}\{\mathbf{s}\}\big) \tag{4}$$

where $\mathrm{diag}\{\mathbf{s}\}$ is a diagonal matrix with all components of $\mathbf{s}$ being positive such that the conditional covariance matrix in Equation 4 is positive definite. As a result, the original features and the model-X knockoff features constructed by Equation 4 have the following joint distribution:

$$(\mathbf{X}, \tilde{\mathbf{X}}) \sim \mathcal{N}\left(\begin{pmatrix}\mathbf{0}\\ \mathbf{0}\end{pmatrix}, \begin{pmatrix}\boldsymbol{\Sigma} & \boldsymbol{\Sigma} - \mathrm{diag}\{\mathbf{s}\}\\ \boldsymbol{\Sigma} - \mathrm{diag}\{\mathbf{s}\} & \boldsymbol{\Sigma}\end{pmatrix}\right) \tag{5}$$

With the constructed knockoff $\tilde{\mathbf{X}}$, feature importances are quantified by computing the knockoff statistics $W_j = g_j(Z_j, \tilde{Z}_j)$ for $1 \le j \le p$, where Z_j and $\tilde{Z}_j$ represent feature importance measures for the j-th feature X_j and its knockoff counterpart $\tilde{X}_j$, respectively, and $g_j(\cdot, \cdot)$ is an antisymmetric function satisfying $g_j(Z_j, \tilde{Z}_j) = -g_j(\tilde{Z}_j, Z_j)$. The knockoff statistics W_j should satisfy a coin-flip property such that swapping an arbitrary pair X_j and its knockoff counterpart $\widetilde{X}_j$ only changes the sign of W_j but keeps the signs of other W_k ($k \neq j$) unchanged [27]. A desirable property for knockoff statistics W_j's is that important features are expected to have large absolute values, whereas unimportant ones should have small symmetric values around 0.

Finally, the absolute values of the knockoff statistics $|W_j|$'s are sorted in decreasing order, and FDR-controlled features are selected whose W_j's exceed some threshold T. In particular, the choice of threshold T follows $T = \min\left\{t \in \mathcal{W}, \frac{1+|\{j: W_j \le -t\}|}{|\{j: W_j \ge t\}|} \le q\right\}$ where $\mathcal{W} = \{|W_j| : 1 \le j \le p\} \setminus \{0\}$ is the set of unique nonzero values from $|W_j|$'s and $q \in (0, 1)$ is the desired FDR level specified by the user.

B Review on Knockoff-tailored DNN architecture

DeepPINK is a prior work that integrates the idea of knockoff filter with DNNs to enable feature selection while maintaining controlled FDR, as outlined in Fig. 4. DeepPINK begins by generating knockoffs $\tilde{\mathbf{X}} \in \mathbb{R}^{n \times p}$ from input data $\mathbf{X} \in \mathbb{R}^{n \times p}$. This is achieved by following the procedure described in Section A. After generating knockoffs, an augmented data matrix $(\mathbf{X}, \tilde{\mathbf{X}}) \in \mathbb{R}^{n \times 2p}$ is constructed and supplied to an off-the-shelf DNN through a plugin pairwise-coupling layer composed of p filters, encapsulated by $\mathbf{F} = (F_1, \cdots, F_p) \in \mathbb{R}^p$, where each j-th filter connects feature X_j and its knockoff counterpart $\tilde{X}_j$.

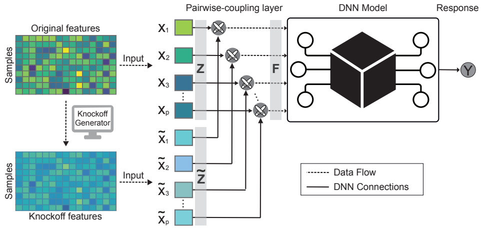

Figure 4: **Overview of DeepPINK.** DeepPINK is built upon an off-the-shelf DNN with a plugin pairwise-coupling layer containing p filters, one per input feature, where each filter connects the original feature and its knockoff counterpart. The filter weights Z_j and $\tilde{Z}_j$ for the j-th feature and its knockoff counterpart are initialized identically for fair competition. The outputs of the filters are fed into an off-the-shelf DNN model.

Table 1: A test suite of data-generating simulation functions by [10].

F_1	$\pi^{x_1 x_2}\sqrt{2x_3} - \sin^{-1}(x_4) + \log(x_3 + x_5) - \frac{x_9}{x_{10}}\sqrt{\frac{x_7}{x_8}} - x_2 x_7$
F_2	$\pi^{x_1 x_2}\sqrt{2\lvert x_3\rvert} - \sin^{-1}(0.5x_4) + \log(\lvert x_3 + x_5\rvert + 1) - \frac{x_9}{1+\lvert x_{10}\rvert}\sqrt{\frac{x_7}{1+\lvert x_8\rvert}} - x_2 x_7$
F_3	$\exp\lvert x_1 - x_2\rvert + \lvert x_2 x_3\rvert - x_3^{2\lvert x_4\rvert} + \log(x_4^2 + x_5^2 + x_7^2 + x_8^2) + x_9 + \frac{1}{1+x_{10}^2}$
F_4	$\exp\lvert x_1 - x_2\rvert + \lvert x_2 x_3\rvert - x_3^{2\lvert x_4\rvert} + (x_1 x_4)^2 + \log(x_4^2 + x_5^2 + x_7^2 + x_8^2) + x_9 + \frac{1}{1+x_{10}^2}$
F_5	$\frac{1}{1+x_1^2+x_2^2+x_3^2} + \sqrt{\exp(x_4 + x_5)} + \lvert x_6 + x_7\rvert + x_8 x_9 x_{10}$
F_6	$\exp(\lvert x_1 x_2\rvert + 1) - \exp(\lvert x_3 + x_4\rvert + 1) + \cos(x_5 + x_6 - x_8) + \sqrt{x_8^2 + x_9^2 + x_{10}^2}$
F_7	$(\arctan(x_1) + \arctan(x_2))^2 + \max(x_3 x_4 + x_6, 0) - \frac{1}{1+(x_4 x_5 x_6 x_7 x_8)^2} + (\frac{\lvert x_7\rvert}{1+\lvert x_9\rvert})^5 + \sum_{i=1}^{10} x_i$
F_8	$x_1 x_2 + 2^{x_3+x_5+x_6} + 2^{x_3+x_4+x_5+x_7} + \sin(x_7 \sin(x_8 + x_9)) + \arccos(0.9x_{10})$
F_9	$\tanh(x_1 x_2 + x_3 x_4)\sqrt{\lvert x_5\rvert} + \exp(x_5 + x_6) + \log((x_6 x_7 x_8)^2 + 1) + x_9 x_{10} + \frac{1}{1+\lvert x_{10}\rvert}$
F_{10}	$\sinh(x_1 + x_2) + \arccos(\tanh(x_3 + x_5 + x_7)) + \cos(x_4 + x_5) + \sec(x_7 x_9)$

The filter weights, $\mathbf{Z} \in \mathbb{R}^p$ and $\tilde{\mathbf{Z}} \in \mathbb{R}^p$ are initialized identically and engage in a competitive dynamic via pairwise connections during the DNN training. Additionally, we employ a linear activation function in the pairwise-coupling layer to stimulate competition between different features.

The outputs of the filters are subsequently channeled into a DNN model that learns to map to the response $\mathbf{Y}$. In this study, we choose an MLP with the exponential linear unit (ELU) activation and four hidden layers as our DNN model. While we use this specific model, DeepPINK's overall process is versatile and fully applicable to any off-the-shelf DNN architecture beyond the MLP.

C Simulation Dataset

Following [10], we use a simulation dataset to evaluate Diamond. 10 synthetic functions (described in Talble 1) are used to generate the simulation dataset.

D Alternative FDR estimation methods

We evaluate the performance of Diamond compared to other baseline methods. We employ a permutation-based approach for the first baseline method to calculate the interaction-wise FDR. Specifically, this involves using a previously described permutation procedure tailored for neural networks to assess the significance of interactions and calculate permutation p-values [36], followed by the Benjamini–Hochberg procedure [25] to estimate the FDR.

For the second baseline method, we consider an ensemble-based approach that represents interaction-wise FDR as the aggregation of feature-wise FDR. This approach follows the intuition that an important feature interaction comprises important univariate features. Specifically, we use a previously described knockoff-based procedure tailored for neural networks to estimate the feature-wise FDR of each univariate feature [30]. We then approximate the interaction-wise FDR as the maximum of the two comprising univariate feature-wise FDRs.