# Sum-of-Parts Models: Faithful Attributions for Groups of Features

**Weiqiu You**[◇] **Helen Qu**[★] **Marco Gatti**[★] **Bhuvnesh Jain**[★] **Eric Wong**[◇]
Department of Computer and Information Science[◇]
Department of Physics and Astronomy[★]
University of Pennsylvania
Philadelphia, PA 19104
{weiqiuy,exwong}@seas.upenn.edu
{helenqu,mgatti29,bjain}@sas.upenn.edu

## Abstract

An explanation of a machine learning model is considered *"faithful"* if it accurately reflects the model's decision-making process. However, explanations such as feature attributions for deep learning are not guaranteed to be faithful, and can produce potentially misleading interpretations. In this work, we develop *Sum-of-Parts (SOP)*, a class of models whose predictions come with grouped feature attributions that are faithful-by-construction. This model decomposes a prediction into an interpretable sum of scores, each of which is directly attributable to a sparse group of features. We evaluate SOP on benchmarks with standard interpretability metrics, and in a case study, we use the faithful explanations from SOP to help astrophysicists discover new knowledge about galaxy formation.

## 1 Introduction

In many high-stakes domains like medicine, law, and automation, important decisions must be backed by well-informed and well-reasoned arguments. However, many machine learning (ML) models are not able to give explanations for their behaviors. One type of explanations for ML models is *feature attribution*: the identification of input features that were relevant to the prediction [37].

For example, in medicine, ML models can assist physicians in diagnosing a variety of lung, heart, and other chest conditions from X-ray images [40, 10, 63, 55]. However, physicians only trust the decision of the model if an explanation identifies regions of the X-ray that make sense [41]. Such explanations are increasingly requested as new biases are discovered in these models [19].

The field has proposed a variety of feature attribution methods to explain ML models. One category consist of post-hoc attributions [42, 33, 39, 47, 53], which have the benefit of being able to apply to any model. Another category of approaches instead build feature attributions directly into the model [60, 23, 49, 46, 11], which promise more accurate attributions but require specially designed architectures or training procedures.

However, feature attributions do not always accurately represent the model's prediction process, a property known as *faithfulness*. An explanation is said to be faithful if it correctly represents the reasoning process of a model [34]. For a feature attribution method, this means that the highlighted features should actually influence the model's prediction. For instance, suppose a ML model for X-rays uses the presence of a bone fragment to predict a fracture while ignoring a jagged line. A faithful feature attribution should assign a positive score to the bone fragment while assigning a score of zero to the jagged line. On the other hand, an unfaithful feature attribution whould assign a positive

XAI in Action: Past, Present, and Future Applications @ NeurIPS 2023.

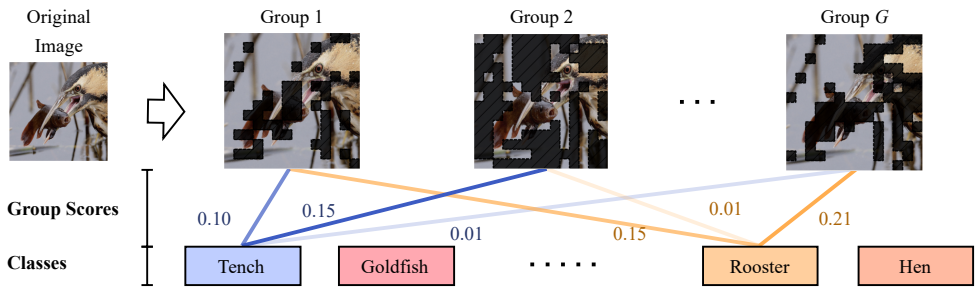



Figure 1: Visualization of grouped attributions. For a set of group attributions, scores are assigned to groups of features instead of individual features. The score for each group represents how much each group of features together contributes to the prediction. We can see that masks can be interpreted as objects kept and objects removed. In this example, group 2, which includes the fish and the predator, contributes 15% to predicting "tench", while group $G$, which has the fish and dark lines removed, contributes only 1% to predicting "tench", but 21% to predicting "Rooster".

score irrelevant regions. Unfortunately, studies have found that many post-hoc feature attributions do not satisfy basic sanity checks for faithfulness [34]. See Appendix E for related works.

In this paper, we first identify a fundamental barrier for feature attributions arising from the curse of dimensionality. Specifically, we prove that feature attributions incur exponentially large error in faithfulness tests for simple settings. These theoretical examples motivate a different type of attribution that scores *groups* of features to overcome this inherent obstacle. Motivated by these challenges, we develop Sum-of-Parts models (SOP), a class of models that attributes predictions to groups of features, which are illustrated in Figure 1. Our approach has three main advantages: SOP models (1) provide grouped attributions that overcome theoretical limitations of feature attributions; (2) are faithful by construction, avoiding pitfalls of post-hoc approaches; and (3) are compatible with any backbone architecture. Our contributions are as follows:

1. We prove that feature attributions incur at least exponentially large error in tests of faithfulness in simple settings. We show that grouped attributions can overcome this limitation.
2. We develop Sum-of-Parts (SOP), a class of models with group-sparse feature attributions that are faithful by construction and are compatible with any backbone architecture.
3. We evaluate our approach in standard image benchmarks with interpretability metrics.
4. In a case study, we use faithful attributions of SOP from weak lensing maps and uncover novel insights about galaxy formation meaningful to cosmologists.

## 2 Inherent Barriers for Feature Attributions

Feature attributions are one of the most common forms of ML models explanations. However, many studies find feature attributions failing basic sanity checks [1, 54] and interpretability tests [28, 8]. Perturbation tests are a widely-used technique for evaluating the faithfulness of an explanation [39, 58, 14]. These tests insert or delete various subsets of features from the input and check if the change in model prediction is in line with the scores from the feature attribution. We first formalize the error of an insertion-style test for a feature attribution on a subset of features.

**Definition 1.** *(Insertion error) The* insertion error *of an feature attribution $\alpha \in \mathbb{R}^d$ for a model $f : \mathbb{R}^d \to \mathbb{R}$ when inserting a subset of features $S$ from an input $x$ is*

$$\text{InsErr}(\alpha, S) = \left| f(x_S) - f(0_d) - \sum_{i \in S} \alpha_i \right| \quad \textit{where} \quad (x_S)_j = \begin{cases} x_j & \textit{if } j \in S \\ 0 & \textit{otherwise} \end{cases} \tag{1}$$

*The total insertion error is $\sum_{S \in \mathcal{P}} \text{InsErr}(\alpha, S)$ where $\mathcal{P}$ is the powerset of $\{1, \ldots, d\}$.*

The insertion error measures how well the total attribution from features in $S$ aligns with the change in model prediction when adding the same features to the $0_d$ vector. Intuitively, a faithful attribution score of the $i$th feature should reflect the change in model prediction after the $i$th feature is added and thus have low insertion error. A similar definition for deletion error is in Appendix A.

Note that if an explanation is faithful, then it achieves low deletion and insertion error. For example, a linear model $f(x) = \theta^T x$ is often described as an interpretable model because it admits a feature attribution $\alpha_i = \theta_i x_i$ that achieves zero deletion and insertion error. Common sanity checks for feature attributions often take the form of insertion and deletion on specific subsets of features [39].

### 2.1 Feature Attributions Incur a Minimum of Exponential Error

In this section, we provide a simple polynomial setting where any choice of feature attribution is guaranteed to incur at least exponential deletion and insertion error across all possible subsets. The key property of this multilinear monomials setting, or the product of $d$ Boolean inputs, is the presence of highly correlated features, which pose an insurmountable challenge for feature attributions.

**Theorem 1** (Insertion Error for Binomials)**.** *Let $p : \{0,1\}^d \to \{0,1,2\}$ be a multilinear binomial polynomial function of $d$ variables. Furthermore suppose that the features can be partitioned into $(S_1, S_2, S_3)$ of equal sizes where $p(x) = \prod_{i \in S_1 \cup S_2} x_i + \prod_{j \in S_2 \cup S_3} x_j$. Then, there exists an $x$ such that any feature attribution for $p$ at $x$ will incur exponential total insertion error.*

In other words, Theorem 1 states that the total insertion error of any feature attribution of a binomial will grow exponentially with respect to the dimension, and an example can be seen in Figure 2. For high-dimensional problems, this suggests that there does not exist a feature attribution that satisfies all possible insertion tests. We prove a similar result for deletion error: feature attributions cannot escape exponentially large insertion error for simple monomial functions. Proofs are deferred to Appendix A.

In combination, Theorems 1 and 2 imply that even for simple problems (Boolean binomials and monomials), the total insertion and deletion error grows exponentially with respect to the dimension.[1] This is precisely the curse of dimensionality, but for feature attributions. These results suggest that a fundamentally different attribution is necessary in order to satisfy deletion and insertion tests.

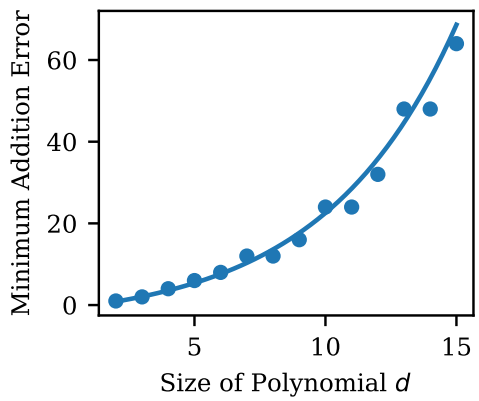



Figure 2: Minimum insertion error for binomials. Fitted function: $\text{InsErr}(d) = e^{\lambda_2 d + \lambda_1} + \lambda_0$ where $(\lambda_2, \lambda_1, \lambda_0) = (0.198, 1.332, 4.778)$.

### 2.2 Grouped Attributions Overcome Barriers for Feature Attributions

The inherent limitations of feature attributions stem from the highly correlated features. A standard feature attribution is limited to assigning one number to each feature. This design is fundamentally unable to accurately model interactions between multiple features, as seen in Theorems 2 and 1.

To explain these correlated effects, we explore a different type of attribution called *grouped attributions*. Grouped attributions assign scores to groups of features instead of individual features. In a grouped attribution, a group only contributes its score if all of its features are present. This concept is formalized in Definition 2.

**Definition 2.** *Let $x \in \mathbb{R}^d$ be an example, and let $S_1, \ldots, S_G \in \{0,1\}^d$ designate $G$ groups of features where $j \in S_i$ if feature $j$ is included in the $i$th group. Then, a grouped feature attribution is a collection $\beta = \{(S_i, c_i)\}_{i=1}^G$ where $c_i \in \mathbb{R}$ is the attributed score for the $i$th group of features $m_i$.*

Grouped attributions have three main characteristics. First, unlike standard feature attributions, a single feature can show up in multiple groups with different scores. Second, the standard feature attribution is a special case where $S_i$ is the singleton set $\{i\}$ for $i = 1, \ldots, G$ for $G = d$. Third, there

[1] The proof technique for Theorems 2 and 1 involves computing a verifiable certificate at each $d$. We were able to computationally verify the result up to $d \leq 20$, and hence the theorem statements are proven only for $d \leq 20$. We conjecture that a general result holds for $d > 20$ for both the insertion and deletion settings.

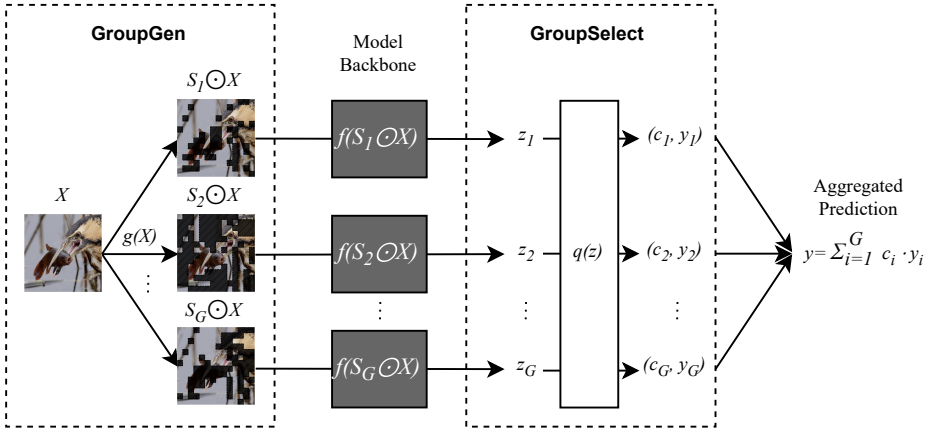



Figure 3: Structure of a Sum-of-Parts Model. A group generator $g$ first generates groups of features. Each group of features $S_i \odot X$ is then passed through the black-box model to obtain the group embedding $z_i$. A group selector $q$ then assigns a score $c_i$ to each group $i$'s representation. The partial logits are then aggregated with a weighted sum to get the predicted logit $y$ for a class.

exists grouped attributions that can succinctly describe the earlier settings from Theorems 2 and 1 with zero insertion and deletion error (Corollary 1 in Appendix A).

To summarize, grouped attributions are able to overcome exponentially growing insertion and deletion errors when the features interact with each other. In contrast, traditional feature attributions lack this property on even simple settings.

## 3 Sum-of-Parts Models

In this section, we develop the Sum-of-Parts (SOP) framework, a way to create faithful grouped attributions. Our proposed grouped attributions consist of two parts: the subsets of features called groups $(S_1, \ldots, S_G) \in [0,1]^d$ and the scores for each group $(c_1, \ldots, c_G)$. We divide our approach into two main modules: GroupGen which generates the groups $S_i$ of features from an input, and GroupSelect which assigns scores $c_i$ to select which groups to use for prediction, as in Figure 3.

**Group Generator.** The group generator $\mathsf{GroupGen} : \mathbb{R}^d \to [0,1]^{G \times d}$ takes in an input $X \in \mathbb{R}^d$ and outputs $G$ masks, each of which corresponds to a group $S_i \in [0,1]^d$. To generate these masks, we use a self-attention mechanism [59] to parameterize a probability distributions over features. The classic attention layer is $\mathsf{Attention}(X) = \mathsf{softmax}\left(\frac{W_q X (W_k X)^T}{\sqrt{d_k}}\right) W_v X$, where $W_k, W_q, W_v \in \mathbb{R}^d$ are learned parameters. However, the outputs of self attention are continuous and dense. Furthermore, we only need the attention weights to generate groups and can ignore the value. To make groups interpretable, we use a sparse variant using the sparsemax operator [35] without the value:

$$\mathsf{GroupGen}(X) = \mathsf{sparsemax}\left(\frac{W_q X (W_k X)^T}{\sqrt{d}}\right) \tag{2}$$

The SparseMax operator uses a simplex projection to make the attention weights sparse. In total, the generator computes sparse attention weights and recombines the input features into groups $S_i$.

**Group Selector.** After we acquire these groups, we use the backbone model $f : \mathbb{R}^d \to \mathbb{R}^h$ to obtain each group's encoding $z_i = f(S_i \odot X)$ with embedding dimension $h$, where $\odot$ is Hadamard product. The goal of the second module, GroupSelect, is to choose a sparse subset of the groups for prediction. Sparsity ensures that a human interpreting the result is not overloaded with too many scores.

The group selector GroupSelect takes in the output of the backbone from all the groups $z_1, \ldots, z_G \in \mathbb{R}^h$ and produces scores $(c_1, \ldots, c_G) \in [0,1]^G$ and logits $(y_1, \ldots, y_G) \in \mathbb{R}^G$ for all groups. To assign a score to each group, we again use a modified sparse attention

$$\mathsf{GroupSelect}(z_1, \ldots, z_G) = \mathsf{sparsemax}\left(\frac{W_{q'} C (W_{k'} z)^T}{\sqrt{h}}\right), C z^T \tag{3}$$

| | | LIME | SHAP | RISE | Grad-CAM | IntGrad | FRESH | SOP (ours) |
|---|---|---|---|---|---|---|---|---|
| ImageNet | Perf ↑ | *0.9160* | *0.9160* | *0.9160* | *0.9160* | *0.9160* | 0.8560 | **0.8880** |
| | Ins ↑ | 0.5121 | 0.6130 | 0.5816 | 0.4545 | 0.3232 | 0.5979 | **0.6149** |
| | $\text{Ins}_G$ ↑ | 0.6121 | 0.6254 | 0.6180 | 0.6303 | 0.4909 | 0.6195 | **0.6396** |
| | Del ↓ | 0.3798 | 0.3009 | 0.4066 | 0.4532 | *0.2357* | 0.4132 | 0.3929 |
| | $\text{Del}_G$ ↓ | 0.3254 | 0.3008 | 0.3135 | 0.3104 | 0.5612 | 0.3302 | **0.2836** |
| VOC 07 | Perf. ↑ | *0.9550* | *0.9550* | *0.9550* | *0.9550* | *0.9550* | **0.9300** | **0.9300** |
| | Ins ↑ | 0.2617 | 0.3137 | 0.2769 | 0.2789 | 0.0915 | 0.2231 | **0.3742** |
| | $\text{Ins}_G$ ↑ | 0.4022 | 0.4043 | 0.3841 | 0.4050 | 0.1870 | 0.3661 | **0.4071** |
| | Del ↓ | 0.0653 | 0.0377 | 0.0866 | 0.2280 | **0.0217** | 0.1590 | 0.0947 |
| | $\text{Del}_G$ ↓ | 0.0825 | 0.0794 | 0.0883 | 0.1037 | 0.2609 | 0.0978 | **0.0765** |

Table 1: Results on ImageNet and VOC 07 on all baselines and SOP on accuracy, insertion, grouped insertion, deletion, and grouped deletion. If a metric has ↑, it means higher numbers in the metric is better, and vice versa. For accuracy, post-hoc methods show the accuracy of the original model.

where $W_{q'}, W_{k'}, C \in \mathbb{R}^h$. We use a projected class weight $W_{q'}C$ to query projected group encodings $W_{k'}z$. In practice, we can initialize the value weight $C$ to the linear classifier of a pretrained model. GroupSelect then simultaneously produces the scores assigned to all groups $(c_1, \ldots, c_G)$ and each group's partial prediction $(y_1, \ldots, y_G)$.

The final prediction is then made by $y = \sum_{i=1}^{G} c_i y_i$, and the corresponding group attribution is $(c_1, S_1), \ldots, (c_G, S_G)$. Since we use a sparsemax operator, in practice there can be significantly fewer than $G$ groups active in the final prediction. This group attribution is faithful to the model since the prediction uses exactly these groups $S_i$, each of which is weighted precisely by the score $c_i$.

## 4 Evaluating SOP Grouped Attributions

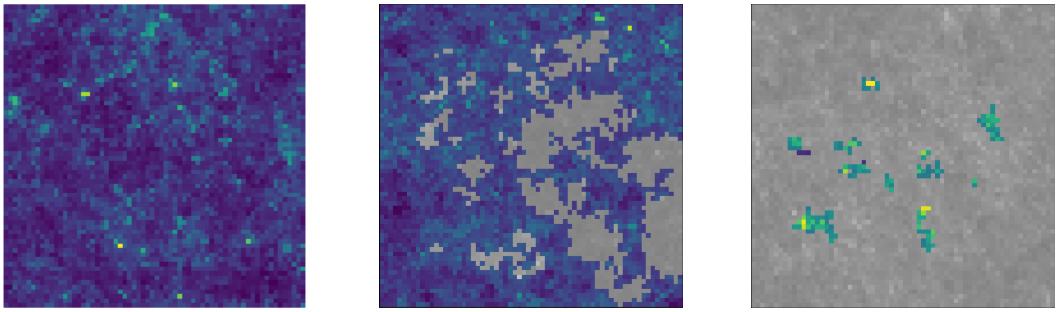
In this section, we perform a standard suite of evaluation metrics on ImageNet [45] and PASCAL VOC 07 [16], using Vision Transformer [15] as our backbone. We compare with post-hoc [42, 33, 39, 47, 53] and built-in [23] attribution methods, and evaluate on accuracy, insertion and deletion, and our proposed grouped insertion and deletion. A complete description of training and evaluation can be found in Appendix C.2 and C.3. Here we highlight results on grouped insertion and deletion.

**Grouped Insertion and Deletion** A standard feature attribution method scores individual pixels, and therefore classic tests check whether inserting and deleting pixels one at a time aligns with the scores. In contrast, grouped attributions assign scores for groups of features, and thus a grouped insertion and deletion test assesses whether deleting groups of features at a time aligns with the scores. Table 1 shows that SOP outperforms all other baselines in both grouped insertion and grouped deletion. This shows that SOP finds grouped attribution that are better at determining which groups of features contribute more to the prediction. This is to be expected as it is faithful-by-construction.

## 5 Case Study: Cosmology

To validate the usability of our approach in real settings, we collaborated with domain experts and used SOP to discover new cosmological knowledge about the expansion of the universe and the growth of cosmic structure.

Weak lensing maps in cosmology calculate the spatial distribution of matter density in the universe using precise measurements of the shapes of ∼100 million galaxies [18]. Cosmologists hope to use weak lensing maps to predict two key parameters related to the initial state of the universe: $\Omega_m$ and $\sigma_8$. $\Omega_m$ captures the average energy density of all matter in the universe (such as radiation and dark energy), while $\sigma_8$ describes the fluctuation of this density. From these parameters, a cosmologist can simulate how cosmological structures, such as galaxies, superclusters and voids, develop throughout cosmic history. However, $\Omega_m$ and $\sigma_8$ are not directly measurable, and the inverse relation from cosmological structures in the weak lensing map to $\Omega_m$ and $\sigma_8$ is unknown.



Weak Lensing Map Void Cluster

Figure 4: A weak lensing map (left) contains large dark areas which are voids, and hot pixels which are clusters. Example voids are shown in the middle, and clusters on the right. We find that voids are used more in prediction in general. Clusters are used comparatively more for $\sigma_8$.

One approach to inferring $\Omega_m$ and $\sigma_8$ from weak lensing maps, as demonstrated for example by Ribli et al. [44], Matilla et al. [36], Fluri et al. [17], is to apply deep learning models that can compare measurements to simulated weak lensing maps. Even though these models have high performance, we do not fully understand how these models predict $\Omega_m$ and $\sigma_8$. As a result, the following remains an open question in cosmology:

*What structures from weak lensing maps can we use to infer the cosmology parameters $\Omega_m$ and $\sigma_8$?*

In collaboration with expert cosmologists, we use convolutional networks trained on CosmoGridV1 [25] to predict $\Omega_m$ and $\sigma_8$ as the backbone of an SOP model to get accurate predictions with faithful group attributions. Crucially, the guarantee of faithfulness in SOP provides confidence that the attributions reflect how the model makes its prediction, opposing to possibly being a red herring.

**Cosmological discoveries** Our initial discoveries come from grouped attributions that correspond to two known structures in the weak lensing maps (as identified by cosmologists): voids and clusters. Figure 4 shows examples of voids (dark regions) and an example of clusters (bright hot pixels), both of which are automatically learned as groups in the SOP model without supervision. We use standard deviation $\sigma$ away from the mean mass intensity for each map to define voids and clusters, where voids are groups with mean intensity of $\leq 0$ and clusters $\geq +3\sigma$. Precise definitions are in Appendix D.

We summarize the discoveries that we made with cosmologists on how clusters and voids influence the prediction of $\Omega_m$ and $\sigma_8$ as follows:

1. A new finding of our work that was surprising to cosmologists relates to the distinction between the two qualitatively different parameters, $\Omega_m$ and $\sigma_8$. We find that voids have especially higher weights for predicting $\Omega_m$. Clusters, especially high-significance ones, have higher weights for predicting $\sigma_8$. More details can be found in Appendix D.3.
2. Using a higher threshold of $+2$ or $+3\sigma$ gives the clusters higher weight especially for $\sigma_8$ than with a lower threshold of $+1\sigma$. This aligns with the cosmology concept that rarer clusters with high standard deviation are more sensitive to $\sigma_8$, the parameter for fluctuation.
3. In general, the voids have higher weights for prediction than the clusters. This is consistent with previous work [36] that voids are the most important feature in prediction.

## 6 Conclusion

In this paper, we identify a fundamental barrier for feature attributions in satisfying faithfulness tests, which can be overcome by grouped instead of individual features. We develop the SOP models to generate group attributions and improve upon standard feature attributions on the majority of insertion and deletion-based interpretability metrics. Most importantly, we used the faithful grouped attributions from SOP to discover cosmological knowledge about expansion of the universe. Our groups are semantically meaningful to cosmologists and revealed new properties in cosmological

structures such as voids and clusters. We hope this work paves the way for further scientific discoveries from faithful explanations of ML models that capture complex and unknown patterns.

## References


[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018.

[2] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey Hinton. Neural additive models: Interpretable machine learning with neural nets. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL `https://openreview.net/forum?id=wHkKTW2wrmm`.

[3] Afra Amini, Tiago Pimentel, Clara Meister, and Ryan Cotterell. Naturalistic causal probing for morpho-syntax. *Transactions of the Association for Computational Linguistics*, 11:384–403, 2022. URL `https://api.semanticscholar.org/CorpusID:248811730`.

[4] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10, 2015. URL `https://api.semanticscholar.org/CorpusID:9327892`.

[5] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Mueller. How to explain individual classification decisions, 2009.

[6] Jasmijn Bastings and Katja Filippova. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, 2020. doi: 10.18653/v1/2020.blackboxnlp-1.14. URL `http://dx.doi.org/10.18653/V1/2020.BLACKBOXNLP-1.14`.

[7] Serge Beucher. Image segmentation and mathematical morphology, 2023. URL `https://people.cmm.minesparis.psl.eu/users/beucher/wtshed.html`. Accessed: September 29, 2023.

[8] Blair Bilodeau, Natasha Jaques, Pang Wei Koh, and Been Kim. Impossibility theorems for feature attribution. *ArXiv*, abs/2212.11870, 2022. URL `https://api.semanticscholar.org/CorpusID:254974246`.

[9] Nitay Calderon, Eyal Ben-David, Amir Feder, and Roi Reichart. DoCoGen: Domain counterfactual generation for low resource domain adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7727–7746, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.533. URL `https://aclanthology.org/2022.acl-long.533`.

[10] Pierre Chambon, Christian Bluethgen, Jean-Benoit Delbrouck, Rogier Van der Sluijs, Małgorzata Połacin, Juan Manuel Zambrano Chaves, Tanishq Mathew Abraham, Shivanshu Purohit, Curtis P Langlotz, and Akshay Chaudhari. Roentgen: vision-language foundation model for chest x-ray generation. *arXiv preprint arXiv:2211.12737*, 2022.

[11] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. In *NIPS*, 2015. URL `https://api.semanticscholar.org/CorpusID:1518846`.

[12] Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. How do decisions emerge across layers in neural models? interpretation with differentiable masking. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020. doi: 10.18653/v1/2020.emnlp-main.262. URL `http://dx.doi.org/10.18653/v1/2020.emnlp-main.262`.

[13] Misha Denil, Alban Demiraj, and Nando de Freitas. Extraction of salient sentences from labelled documents. *ArXiv*, abs/1412.6815, 2014. URL https://api.semanticscholar.org/CorpusID:9121062.

[14] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4443–4458, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.408. URL https://aclanthology.org/2020.acl-main.408.

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

[16] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88: 303–338, 2010.

[17] Janis Fluri, Tomasz Kacprzak, Aurelien Lucchi, Aurel Schneider, Alexandre Refregier, and Thomas Hofmann. Full $w$CDM analysis of KiDS-1000 weak lensing maps using deep learning. *Physical Review D*, 105(8), April 2022. doi: 10.1103/physrevd.105.083518. URL https://doi.org/10.1103%2Fphysrevd.105.083518.

[18] M. Gatti, E. Sheldon, A. Amon, M. Becker, M. Troxel, A. Choi, C. Doux, N. MacCrann, A. Navarro-Alsina, I. Harrison, D. Gruen, G. Bernstein, M. Jarvis, L. F. Secco, A. Ferté, T. Shin, J. McCullough, R. P. Rollins, R. Chen, C. Chang, S. Pandey, I. Tutusaus, J. Prat, J. Elvin-Poole, C. Sanchez, A. A. Plazas, A. Roodman, J. Zuntz, T. M. C. Abbott, M. Aguena, S. Allam, J. Annis, S. Avila, D. Bacon, E. Bertin, S. Bhargava, D. Brooks, D. L. Burke, A. Carnero Rosell, M. Carrasco Kind, J. Carretero, F. J. Castander, C. Conselice, M. Costanzi, M. Crocce, L. N. da Costa, T. M. Davis, J. De Vicente, S. Desai, H. T. Diehl, J. P. Dietrich, P. Doel, A. Drlica-Wagner, K. Eckert, S. Everett, I. Ferrero, J. Frieman, J. García-Bellido, D. W. Gerdes, T. Giannantonio, R. A. Gruendl, J. Gschwend, G. Gutierrez, W. G. Hartley, S. R. Hinton, D. L. Hollowood, K. Honscheid, B. Hoyle, E. M. Huff, D. Huterer, B. Jain, D. J. James, T. Jeltema, E. Krause, R. Kron, N. Kuropatkin, M. Lima, M. A. G. Maia, J. L. Marshall, R. Miquel, R. Morgan, J. Myles, A. Palmese, F. Paz-Chinchón, E. S. Rykoff, S. Samuroff, E. Sanchez, V. Scarpine, M. Schubnell, S. Serrano, I. Sevilla-Noarbe, M. Smith, M. Soares-Santos, E. Suchyta, M. E. C. Swanson, G. Tarle, D. Thomas, C. To, D. L. Tucker, T. N. Varga, R. H. Wechsler, J. Weller, W. Wester, and R. D. Wilkinson. Dark energy survey year 3 results: weak lensing shape catalogue. *MNRAS*, 504(3):4312–4336, July 2021. doi: 10.1093/mnras/stab918.

[19] Ben Glocker, Charles Jones, Melanie Bernhardt, and Stefan Winzeck. Risk of bias in chest x-ray foundation models. *arXiv preprint arXiv:2209.02965*, 2022.

[20] Max Glockner, Ivan Habernal, and Iryna Gurevych. Why do you think that? exploring faithful sentence-level rationales without supervision. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1080–1095, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.97. URL https://aclanthology.org/2020.findings-emnlp.97.

[21] L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006. doi: 10.1109/TPAMI.2006.233.

[22] Saachi Jain, Hadi Salman, Eric Wong, Pengchuan Zhang, Vibhav Vineet, Sai Vemprala, and Aleksander Madry. Missingness bias in model debugging. *arXiv preprint arXiv:2204.08945*, 2022.

[23] Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C. Wallace. Learning to faithfully rationalize by construction. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

[24] N. Jeffrey, M. Gatti, C. Chang, L. Whiteway, U. Demirbozan, A. Kovacs, G. Pollina, D. Bacon, N. Hamaus, T. Kacprzak, O. Lahav, F. Lanusse, B. Mawdsley, S. Nadathur, J. L. Starck, P. Vielzeuf, D. Zeurcher, A. Alarcon, A. Amon, K. Bechtol, G. M. Bernstein, A. Campos, A. Carnero Rosell, M. Carrasco Kind, R. Cawthon, R. Chen, A. Choi, J. Cordero, C. Davis, J. DeRose, C. Doux, A. Drlica-Wagner, K. Eckert, F. Elsner, J. Elvin-Poole, S. Everett, A. Ferté, G. Giannini, D. Gruen, R. A. Gruendl, I. Harrison, W. G. Hartley, K. Herner, E. M. Huff, D. Huterer, N. Kuropatkin, M. Jarvis, P. F. Leget, N. MacCrann, J. McCullough, J. Muir, J. Myles, A. Navarro-Alsina, S. Pandey, J. Prat, M. Raveri, R. P. Rollins, A. J. Ross, E. S. Rykoff, C. Sánchez, L. F. Secco, I. Sevilla-Noarbe, E. Sheldon, T. Shin, M. A. Troxel, I. Tutusaus, T. N. Varga, B. Yanny, B. Yin, Y. Zhang, J. Zuntz, T. M. C. Abbott, M. Aguena, S. Allam, F. Andrade-Oliveira, M. R. Becker, E. Bertin, S. Bhargava, D. Brooks, D. L. Burke, J. Carretero, F. J. Castander, C. Conselice, M. Costanzi, M. Crocce, L. N. da Costa, M. E. S. Pereira, J. De Vicente, S. Desai, H. T. Diehl, J. P. Dietrich, P. Doel, I. Ferrero, B. Flaugher, P. Fosalba, J. García-Bellido, E. Gaztanaga, D. W. Gerdes, T. Giannantonio, J. Gschwend, G. Gutierrez, S. R. Hinton, D. L. Hollowood, B. Hoyle, B. Jain, D. J. James, M. Lima, M. A. G. Maia, M. March, J. L. Marshall, P. Melchior, F. Menanteau, R. Miquel, J. J. Mohr, R. Morgan, R. L. C. Ogando, A. Palmese, F. Paz-Chinchón, A. A. Plazas, M. Rodriguez-Monroy, A. Roodman, E. Sanchez, V. Scarpine, S. Serrano, M. Smith, M. Soares-Santos, E. Suchyta, G. Tarle, D. Thomas, C. To, J. Weller, and DES Collaboration. Dark Energy Survey Year 3 results: Curved-sky weak lensing mass map reconstruction. *MNRAS*, 505(3):4626–4645, August 2021. doi: 10.1093/mnras/stab1495.

[25] Tomasz Kacprzak, Janis Fluri, Aurel Schneider, Alexandre Refregier, and Joachim Stadel. CosmoGridV1: a simulated LambdaCDM theory prediction for map-level cosmological inference. *JCAP*, 2023(2):050, February 2023. doi: 10.1088/1475-7516/2023/02/050.

[26] Ákos Kádár, Grzegorz Chrupała, and Afra Alishahi. Representation of linguistic form and function in recurrent neural networks. *Computational Linguistics*, 43(4):761–780, December 2017. doi: 10.1162/COLI_a_00300. URL `https://aclanthology.org/J17-4003`.

[27] Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. Learning the difference that makes a difference with counterfactually-augmented data, 2020.

[28] Pieter-Jan Kindermans, Sara Hooker, Julius Adebayo, Maximilian Alber, Kristof T. Schütt, Sven Dähne, Dumitru Erhan, and Been Kim. The (un)reliability of saliency methods. *Lecture Notes in Computer Science*, pp. 267–280, 2019. ISSN 1611-3349. doi: 10.1007/978-3-030-28954-6_14. URL `http://dx.doi.org/10.1007/978-3-030-28954-6_14`.

[29] Thibault Laugel, Xavier Renard, Marie-Jeanne Lesot, Christophe Marsala, and Marcin Detyniecki. Defining locality for surrogates in post-hoc interpretablity, 2018.

[30] Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure, 2017.

[31] Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pp. 150–158, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450314626. doi: 10.1145/2339530.2339556. URL `https://doi.org/10.1145/2339530.2339556`.

[32] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pp. 623–631, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450321747. doi: 10.1145/2487575.2487579. URL `https://doi.org/10.1145/2487575.2487579`.

[33] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[34] Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. Towards faithful model explanation in nlp: A survey, 2022.

[35] André F. T. Martins and Ramón Fernández Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. *ArXiv*, abs/1602.02068, 2016.

[36] José Manuel Zorrilla Matilla, Manasi Sharma, Daniel Hsu, and Zoltán Haiman. Interpreting deep learning models for weak lensing. *Physical Review D*, 102(12), December 2020. ISSN 2470-0029. doi: 10.1103/physrevd.102.123506. URL `http://dx.doi.org/10.1103/physrevd.102.123506`.

[37] Christoph Molnar. *Interpretable Machine Learning*. Independently published, 2 edition, 2022. URL `https://christophm.github.io/interpretable-ml-book`.

[38] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2016.11.008. URL `https://www.sciencedirect.com/science/article/pii/S0031320316303582`.

[39] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. In *British Machine Vision Conference*, 2018.

[40] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

[41] Mauricio Reyes, Raphael Meier, Sérgio Pereira, Carlos A Silva, Fried-Michael Dahlweid, Hendrik von Tengg-Kobligk, Ronald M Summers, and Roland Wiest. On the interpretability of artificial intelligence in radiology: Challenges and opportunities. *Radiology: Artificial Intelligence*, 2(3):e190043, 2020. doi: 10.1148/ryai.2020190043.

[42] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[43] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018. doi: 10.1609/aaai.v32i1.11491. URL `https://ojs.aaai.org/index.php/AAAI/article/view/11491`.

[44] Dezső Ribli, Bálint Ármin Pataki, José Manuel Zorrilla Matilla, Daniel Hsu, Zoltán Haiman, and István Csabai. Weak lensing cosmology with convolutional neural networks on noisy data. *Monthly Notices of the Royal Astronomical Society*, 490(2):1843–1860, 09 2019. ISSN 0035-8711. doi: 10.1093/mnras/stz2610. URL `https://doi.org/10.1093/mnras/stz2610`.

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[46] Sara Sabour, Nicholas Frosst, and Geoffrey E. Hinton. Dynamic routing between capsules. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 3859–3869, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[47] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128:336–359, 2016. URL `https://api.semanticscholar.org/CorpusID:15019293`.

[48] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 3145–3153. JMLR.org, 2017.

[49] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop at International Conference on Learning Representations*, 2014.

[50] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.

[51] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise, 2017.

[52] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2014. URL `https://api.semanticscholar.org/CorpusID:12998557`.

[53] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, 2017. URL `https://api.semanticscholar.org/CorpusID:16747630`.

[54] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks, 2017.

[55] Chakkrit Termritthikun, Ayaz Umer, Suwichaya Suwanwimolkul, Feng Xia, and Ivan Lee. Explainable knowledge distillation for on-device chest x-ray classification. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2023.

[56] Michael Tsang, Sirisha Rambhatla, and Yan Liu. How does this interaction affect me? interpretable attribution for feature interactions. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6147–6159. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/443dec3062d0286986e21dc0631734c9-Paper.pdf`.

[57] Bhavan Vasu and Chengjiang Long. Iterative and adaptive sampling with spatial attention for black-box model explanations. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[58] Bhavan Vasu and Chengjiang Long. Iterative and adaptive sampling with spatial attention for black-box model explanations. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020. doi: 10.1109/wacv45572.2020.9093576. URL `http://dx.doi.org/10.1109/WACV45572.2020.9093576`.

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

[60] Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 11–20, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1002. URL `https://aclanthology.org/D19-1002`.

[61] Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel Weld. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6707–6723, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.523. URL `https://aclanthology.org/2021.acl-long.523`.

[62] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.

[63] Ran Zhang, Xin Tie, John W Garrett, Dalton Griner, Zhihua Qi, Nicholas B Bevins, Scott B Reeder, and Guang-Hong Chen. A generalizable artificial intelligence model for covid-19 classification task using chest x-ray radiographs: Evaluated over four clinical datasets with 15,097 patients. *arXiv preprint arXiv:2210.02189*, 2022.

[64] Ran Zmigrod, Sabrina J. Mielke, Hanna Wallach, and Ryan Cotterell. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1651–1661, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1161. URL `https://aclanthology.org/P19-1161`.

# A Theorem Proofs for Section 2

## A.1 Definition for Deletion Error

We can formalize an analogous error for deletion-style tests as follows:

**Definition 3.** *(Deletion error) The* deletion error *of an feature attribution $\alpha \in \mathbb{R}^d$ for a model $f : \mathbb{R}^d \to \mathbb{R}$ when removing a subset of features $S$ from an input $x$ is*

$$\mathrm{DelErr}(\alpha, S) = \left| f(x) - f(x_{\neg S}) - \sum_{i \in S} \alpha_i \right| \quad \textit{where} \;\; (x_{\neg S})_j = \begin{cases} x_j & \textit{if } j \notin S \\ 0 & \textit{otherwise} \end{cases} \tag{4}$$

*The total deletion error is $\sum_{S \in \mathcal{P}} \mathrm{DelErr}(\alpha, S)$ where $\mathcal{P}$ is the powerset of $\{1, \ldots, d\}$.*

The deletion error measures how well the total attribution from features in $S$ aligns with the change in model prediction when removing the same features from $x$.

## A.2 Feature Attributions Incur a Minimum of Exponential Error

In this section, we provide two simple polynomial settings where any choice of feature attribution is guaranteed to incur at least exponential deletion and insertion error across all possible subsets. The key property in these examples is the presence of highly correlated features, which pose an insurmountable challenge for feature attributions. We begin with the first setting: multilinear monomials, or the product of $d$ Boolean inputs.

**Theorem 2** (Deletion Error for Monomials)**.** *Let $p : \{0,1\}^d \to \{0,1\}$ be a multilinear monomial function of $d \leq 20$ variables, $p(x) = \prod_{i=1}^d x_i$. Then, there exists an $x$ such that any feature attribution for $p$ at $x$ will incur an approximate lower bound of $e^{\gamma_1 d + \gamma_0}$ total deletion error, where $(\gamma_1, \gamma_0) = (0.664, -1.159)$.*

*Proof.* Let $p : \{0,1\}^d \to \{0,1\}$ be a multilinear monomial function of $d \leq 20$ variables, $p(x) = \prod_{i=1}^d x_i$. Then, there exists an $x$ such that any feature attribution for $p$ at $x$ will incur an approximate lower bound of $e^{\gamma_1 d + \gamma_0}$ total deletion error, where $(\gamma_1, \gamma_0) = (0.664, -1.159)$. □

In other words, Theorem 2 states that the total deletion error of any feature attribution of a monomial will grow exponentially with respect to the dimension. For high-dimensional problems, this suggests that there does not exist a feature attribution that satisfies all possible deletion tests. On the other hand, monomials can easily achieve low insertion error, as formalized in Lemma 1.

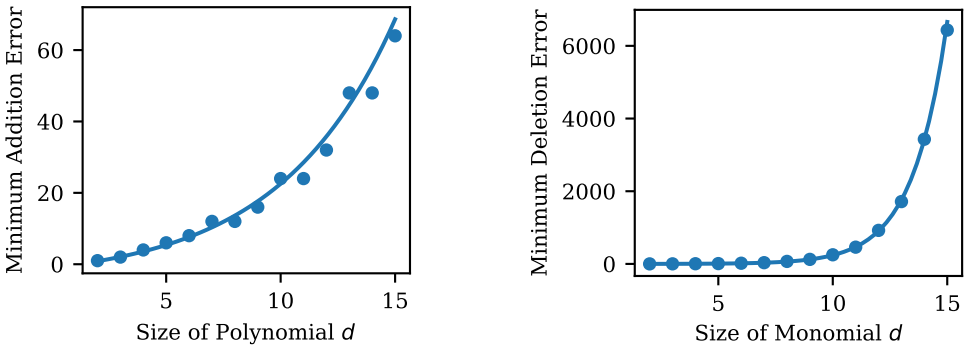
**Lemma 1** (Insertion Error for Monomials)**.** *Let $p : \{0,1\}^d \to \{0,1\}$ be a multilinear monomial function of $d$ variables, $p(x) = \prod_{i=1}^d x_i$. Then, for all $x$, there exists a feature attribution for $p$ at $x$ that incurs at most $1$ total insertion error.*

However, once we slightly increase the function complexity to binomials, we find that the total insertion error of any feature attribution will grow exponentially with respect to $d$. The two terms in the binomial must have some overlapping features or else the problem reduces to a monomial.

**Theorem 1** (Insertion Error for Binomials)**.** *Let $p : \{0,1\}^d \to \{0,1,2\}$ be a multilinear binomial polynomial function of $d$ variables. Furthermore suppose that the features can be partitioned into $(S_1, S_2, S_3)$ of equal sizes where $p(x) = \prod_{i \in S_1 \cup S_2} x_i + \prod_{j \in S_2 \cup S_3} x_j$. Then, there exists an $x$ such that any feature attribution for $p$ at $x$ will incur exponential total insertion error.*

In combination, Theorems 2 and 1 imply that even for simple problems (Boolean monomials and binomials), the total deletion and insertion error grows exponentially with respect to the dimension.[2] This is precisely the curse of dimensionality, but for feature attributions. These results suggest that a fundamentally different attribution is necessary in order to satisfy deletion and insertion tests.

[2]The proof technique for Theorems 2 and 1 involves computing a verifiable certificate at each $d$. We were able to computationally verify the result up to $d \leq 20$, and hence the theorem statements are proven only for $d \leq 20$. We conjecture that a general result holds for $d > 20$ for both the insertion and deletion settings.




(a) Minimum insertion error for binomials. Fitted function: $\text{InsErr}(d) = e^{\lambda_2 d + \lambda_1} + \lambda_0$ where $(\lambda_2, \lambda_1, \lambda_0) = (0.198, 1.332, 4.778)$.

(b) Minimum deletion error for monomials. Fitted function: $\text{DelErr}(d) = e^{\gamma_1 d + \gamma_0}$ where $(\gamma_1, \gamma_0) = (0.664, -1.159)$.

Figure 5: The minimum (a) insertion errors of binomials of size $d$ and (b) deletion error of monimials of size $d$, where the minimum is over all possible feature attributions. These lower bounds suggest an inherent fundamental limitation of feature attributions in faithfully explaining correlated features.

### A.3 Proof for Feature Attributions Incur a Minimum of Exponential Error

**Theorem 2** (Deletion Error for Monomials)**.** *Let $p : \{0,1\}^d \to \{0,1\}$ be a multilinear monomial function of $d \leq 20$ variables, $p(x) = \prod_{i=1}^d x_i$. Then, there exists an $x$ such that any feature attribution for $p$ at $x$ will incur an approximate lower bound of $e^{\gamma_1 d + \gamma_0}$ total deletion error, where $(\gamma_1, \gamma_0) = (0.664, -1.159)$.*

*Proof.* Let $x = \mathbf{1}_d$, and let $\alpha \in \mathbb{R}^d$ be any feature attribution. Consider the set of all possible perturbations to the input, or the power set of all features $\mathcal{P}$, We can write the error of the attribution under a given perturbation $S \in \mathcal{P}$ as

$$\text{error}(\alpha, S) = \left| 1 - \sum_{i \in S} \alpha_i \right| \tag{5}$$

This captures the faithfulness notion that $\alpha_i$ is faithful if it reflects a contribution of $\alpha_i$ to the prediction. Then, the feature attribution $\alpha^*$ that achieves the lowest possible faithfulness error over all possible subsets is

$$\alpha^* = \arg\min_{\alpha} \sum_{S \in \mathcal{P}} \text{error}(\alpha, S) \tag{6}$$

This can be more compactly written as

$$\alpha^* = \arg\min_{\alpha} \mathbf{1}^\top |\mathbf{1} - M\alpha| \tag{7}$$

where $M_{ij} = \begin{cases} 1 & \text{if } j \in S_i \\ 0 & \text{otherwise} \end{cases}$ for an enumeration of all elements $S_i \in \mathcal{P}$. This is a convex program that can be solved with linear programming solvers such as CVXPY. We solve for $\alpha^*$ using ECOS in the `cvxpy` library for $d \in \{2, \ldots, 20\}$. To fit the exponential function, we fit a linear model to the log transform of the output which has high degree of fit (with a relative absolute error of 0.008), with the resulting exponential function shown in Figure 5b. □

**Lemma 1** (Insertion Error for Monomials)**.** *Let $p : \{0,1\}^d \to \{0,1\}$ be a multilinear monomial function of $d$ variables, $p(x) = \prod_{i=1}^d x_i$. Then, for all $x$, there exists a feature attribution for $p$ at $x$ that incurs at most $1$ total insertion error.*

*Proof.* Consider $\alpha = 0_d$. If $x \neq \mathbf{1}_d$ then this achieves 0 insertion error. Otherwise, suppose $x = \mathbf{1}_d$. Then, for all subsets $S \neq [d]$, $p(x_S) = 0 = \sum_{i \in S} \alpha_i$ so $\alpha$ incurs no insertion error for all but one

subset. For the last subset $S = [d]$, the insertion error is 1. Therefore, the total insertion error is at most 1 for $\alpha = 0_d$. □

**Theorem 1** (Insertion Error for Binomials)**.** *Let $p : \{0,1\}^d \to \{0,1,2\}$ be a multilinear binomial polynomial function of $d$ variables. Furthermore suppose that the features can be partitioned into $(S_1, S_2, S_3)$ of equal sizes where $p(x) = \prod_{i \in S_1 \cup S_2} x_i + \prod_{j \in S_2 \cup S_3} x_j$. Then, there exists an $x$ such that any feature attribution for $p$ at $x$ will incur exponential total insertion error.*

*Proof.* Consider $x = \mathbf{1}_d$. The addition error for a binomial function can be written as

$$\text{error}(\alpha, S) = \left| \sum_{i \in S} \alpha_i - 1[S_1 \cup S_2 \subseteq S] - 1[S_2 \cup S_2 \subseteq S] \right| = |M_S^\top \alpha - c_S| \tag{8}$$

where $(M_S, c_S)$ are defined as $(M_S)_i = \begin{cases} 1 & \text{if } i \in S \\ 0 & \text{otherwise,} \end{cases}$ and $c_S$ contains the remaining constant terms. Then, the least possible insertion error that any attribution can achieve is

$$\alpha^* = \arg\min_\alpha \sum_{S \in \mathcal{P}} \text{error}(\alpha, S) = \arg\min_\alpha \mathbf{1}^\top |c - M\alpha| \tag{9}$$

where $(M, c)$ are constructed by stacking $(M_S, c_S)$ for some enumeration of $S \in \mathcal{P}$. This is a convex program that can be solved with linear programming solvers such as CVXPY. We solve for $\alpha^*$ using ECOS in the `cvxpy` library for $d \in \{2, \ldots, 20\}$. To get the exponential function, we fit a linear model to the log transform of the output, doing a grid search over the auxiliary bias term. The resulting function has a high degree of fit (with a relative absolute error of 0.106), with the resulting exponential function shown in Figure 2. □

**Insertion and Deletion Error for Grouped Attribution.** We can define analogous notions of insertion and deletion error when given a grouped attribution. It is similar to the original definition, however a group only contributes its score to the attribution if all members of the group are present.

**Definition 4.** *(Grouped deletion error) The* grouped deletion error *of an grouped attribution $\beta = \{(S_i, c_i)\}_{i=1}^G$ for a model $f : \mathbb{R}^d \to \mathbb{R}$ when deleting a subset of features $S$ from an input $x$ is*

$$\text{GroupDelErr}(\alpha, S) = \left| f(x) - f(x_{\neg S}) - \sum_{i : S \subseteq S_i} c_i \right| \tag{10}$$

**Definition 5.** *(Grouped insertion error) The* grouped insertion error *of an feature attribution $\beta = \{(S_i, c_i)\}_{i=1}^G$ for a model $f : \mathbb{R}^d \to \mathbb{R}$ when inserting a subset of features $S$ from an input $x$ is*

$$\text{GroupInsErr}(\alpha, S) = \left| f(x_S) - f(0_d) - \sum_{i : S \subseteq S_i} c_i \right| \tag{11}$$

**Corollary 1.** *Consider $p_1$ and $p_2$, the polynomials from Theorem 2 and Theorem 1. Then, there exists a grouped attribution with zero deletion and insertion error for both polynomials.*

*Proof.* Let $[d]$ denote $\{1, \ldots, d\}$. First let $p_1(x) = \prod_i x_i$ and consider a grouped attribution with one group, $\beta = \{([d], 1)\}$. Then, no matter what subset $S$ is being tested, $S \subset [d]$ is always true, thus:

$$\text{GroupDelErr}(\beta, S) = \left| f(x) - f(x_{\neg S}) - \sum_{i : S \subseteq m_i} s_i \right| = |1 - 0 - 1| = 0$$

Next let $p_2(x) = \prod_{i \in S_1 \cup S_2} x_i + \prod_{j \in S_2 \cup S_3} x_j$ and consider a grouped attribution with two groups, $\beta = \{(S_1 \cup S_2, 1), (S_2 \cup S_3, 1)\}$. If $S = [d]$, then

$$\text{GroupInsErr}(\beta, S) = \left| f(x_S) - f(0) - \sum_{i : S \subseteq S_i} c_i \right| = 2 - 0 - (1 + 1) = 0$$

**Algorithm 1** Sum-of-Parts Models

**Require:** Group Generator GroupGen, Group Selector GroupSelect
**Require:** Input Features $X$, Prediction Model $f$

$S_1, S_2 \ldots, S_G \leftarrow \texttt{GroupGen}(X)$ ▷ Group Generation

**for** $j = 1 \ldots G$ **do**
  $z_i \leftarrow f(S_i \odot X)$ ▷ Embedding Grouped Input Features
**end for**

$(c_1, y_1), \ldots, (c_G, y_G) \leftarrow \texttt{GroupSelect}(z_1, \ldots, z_G)$ ▷ Group Evaluation

$y \leftarrow \sum_i^G c_i \cdot y_i$ ▷ Sum-of-Parts

If $S$ empty, then the insertion error is trivially 0. Otherwise suppose $S$ is missing an element from one of $S_1$ or $S_3$. WLOG suppose it is from $S_1$ but not $S_2$ or $S_3$. Then,

$$\text{GroupInsErr}(\beta, S) = \left| f(x_S) - f(0) - \sum_{i:S \subseteq S_i} c_i \right| = 2 - 1 - (1) = 0$$

Otherwise, suppose we are missing elements from both $S_1$ and $S_3$. Then,

$$\text{GroupInsErr}(\beta, S) = \left| f(x_S) - f(0) - \sum_{i:S \subseteq S_i} c_i \right| = 2 - 0 - (1 + 1) = 0$$

Lastly, suppose we are missing elements from $S_2$. Then,

$$\text{GroupInsErr}(\beta, S) = \left| f(x_S) - f(0) - \sum_{i:S \subseteq S_i} c_i \right| = 0 - 0 = 0$$

Thus by exhaustly checking all cases, $p_2$ has zero grouped insertion error. □

# B Algorithm Details

Our algorithm is formalized in Table 1. For GroupGen, we are using weights from all the queries to keys with multiple heads without reduction. Therefore we have $d$ number of group per attention head, and a total of $d \times d_a$ groups for $d_a$ attention heads.

# C Experiment Details for Evaluating SOP Grouped Attributions

In this section, we perform a standard evaluation with commonly-used metrics for measuring the quality of a feature attribution. These metrics align with the insertion and deletion error analyzed in Section 2. We find that our grouped attributions can improve upon the majority of metrics over standard feature attributions, which is consistent with our theoretical results.

## C.1 Experimental Setups

We evaluate SOP on ImageNet [45] for single-label and PASCAL VOC 07 [16] for multi-label classification. We use Vision Transformer [15] as our backbone. More information about training and datasets are in Appendix C.2.

We compare against different types of baselines:

1. *Surrogate-model-based*: LIME [42], SHAP [33]
2. *Perturbation-based*: RISE [39]
3. *Gradient-based*: GradCAM [47], IntGrad [53]

4. *Built-in explanation*: FRESH [23]

To evaluate our approach, we use interpretability metrics that are standard practice in the literature for feature attributions [39, 57, 23]. We summarize these metrics as follows and provide precise descriptions in Appendix C.3:

1. **Accuracy:** We measure the standard accuracy of the model. For methods that build explanations into the model such as SOP, it is desirable to maintain good performance.
2. **Insertion and Deletion:** We measure faithfulness of attributions on predictions with deletion and insertion tests that are standard for feature attributions [39]. These tests insert and delete features pixel by pixel.
3. **Grouped Insertion and Deletion:** Insertion and deletion tests were originally made for standard feature attributions, which assign at most one score per feature. Grouped attributions can have multiple scores per feature if a feature shows up in multiple groups. We therefore generalize these tests to their natural group analogue, which inserts and deletes features in groups.

### C.2 Training and Datasets

**ImageNet** ImageNet [45] contains 1000 classes for common objects. We use a subset of the first 10 classes for our evaluation. We use a finetuned vision transformer model from HuggingFace [3] for ImageNet, and our finetune of a pretrained model [4] for VOC.

**PASCAL VOC 07** PASCAL VOC 07 [16] is an object detection dataset with 20 classes. We train with multilabel classification training, but predict with single binary labels for target classes [62].

For both ImageNet and VOC datasets, we use a simple patch segmenter with patch size of $16 \times 16$ to segment the images, matching the size of ViT-base. For insertion and deletion evaluation, we evaluate on a subset of 50 examples for ImageNet and 400 for VOC due to time constraint.

### C.3 Metrics Details

#### C.3.1 Accuracy

We evaluate on accuracy to measure if the wrapped model has comparable performance with the original model, following Jain et al. [23]. For post-hoc explanations, the performance shown will be the performance of the original model, since they are not modifying the model.

#### C.3.2 Deletion and Insertion

Petsiuk et al. [39] proposes insertion and deletion for evaluating feature attributions for images.

**Deletion** Deletion deletes groups of pixels from the complete image at a time, also starting from the most salient pixels from the attribution. If the top attribution scores relfect the most attributed features, then the prediction consistency should drop down from the stsart and result in a lower deletion score.

**Insertion** Insertion adds groups of pixels to a blank or blurred image at a time, starting with the pixels deemed most important by the attribution, and computes the AUC of probability difference between predictions from the perturbed input and original input. If the top attribution scores faithfully reflect the most attributed features, then the prediction consistency should go up from the start and result in a higher insertion score.

**Grouped Insertion and Deletion** For a standard attribution, it orders the features, each feature is a group, and thus we test by deleting or inserting in that order. For a grouped attribution, the natural generalization is then to delete or insert each group at each time. Besides the regular version of insertion and deletion, we also use a grouped version. For deletion, instead of removing a fixed number of pixels every step, we delete a group of features. If the features to remove overlaps with already deleted features, we only remove what has not been removed. The same is performed for grouped insertion when adding features. To get the groups, we use groups generated from SOP.

[3] https://huggingface.co/google/vit-base-patch16-224

[4] https://huggingface.co/google/vit-base-patch16-224-in21k

### C.3.3 Sparsity

Having sparse explanations helps with interpretability for humans. We evaluate the sparsity of our grouped attributions by count the number of input features in each group $i$, and then average the count for all groups with non-zero group score $c_i$.

$$\#\text{ group nonzeros} = \frac{\sum_i (|S_i| \mathbb{1}(c_i \geq 0))}{|X_i|}$$

The fewer number of nonzeros implies more sparsity, and thus better human interpretability. On both ImageNet and VOC, we get around 60% nonzeros. This shows that SOP produces groups that are sparse.

### C.4 Results and Discussions

**Accuracy.** To evaluate the performance of built-in explanation models have, we evaluate on accuracy. The intuition is that built-in attributions use a subset of features when they make the prediction. Therefore, it is possible that they do not have the same performance as the original models. A slight performance drop is an acceptable trade-off, while a large drop makes the model unusable.

We compare with FRESH which is also a model with built-in attributions that initially works for language but we adapt for vision. Table 1 shows that SOP retains the most accuracy on ImageNet and VOC and no less than FRESH. This shows that our built-in grouped attributions do not degrade model performance while adding faithful attributions. The grouped attributions are potentially the advantage of SOP over non-grouped attributions such as FRESH.

**Insertion and Deletion.** To evaluate how faithful the attributions are, we evaluate on insertion and deletion tests. The intuition behind insertion is that, if the attribution scores are faithful, then adding the highest scored features first from the blank image will give a higher AUC, and deleting them first from the full image will give a low AUC. While Petsiuk et al. [39] perturb an image by blurring to avoid adding spurious correlations to the classifier, this may not entirely remove a feature. Since modern backbones (such as the Vision Transformer that we use) are known to not be as biased as classic models when blacking out features [22], we simply replace features entirely with black pixels.

We compare against all the post-hoc and built-in baselines. Table 1 shows that SOP has the best insertion AUC among all methods for both ImageNet and VOC. Having higher insertion scores shows that the highest scored attributions from SOP are more sufficient than other methods in making the prediction. While the deletion scores are lower, SOP does not promise that the attributions it selects are comprehensive, and thus have the potential of lowering the deletion scores.

**Grouped Insertion and Deletion.** While we can still technically evaluate grouped attributions with pixel-wise insertion and deletion tests, it does not quite match the semantics of a grouped attribution, which score groups of features instead of individual features. A standard feature attribution method scores individual pixels, and therefore classic tests check whether inserting and deleting pixels one at a time aligns with the scores. In contrast, grouped attributions assign scores for groups of features, and thus a grouped insertion and deletion test assesses whether deleting groups of features at a time aligns with the scores.

Table 1 shows that SOP outperforms all other baselines in both grouped insertion and grouped deletion. This shows that SOP finds grouped attribution that are better at determining which groups of features contribute more to the prediction. This is to be expected as it is faithful-by-construction.

## D Case Study: Cosmology

In our collaboration with cosmologists, we identified two cosmological structures learned in our group attributions: voids and clusters. In this section, we describe how we extracted void and cluster labels from the group attributions.

Let $S$ be a group from SOP when making predictions for an input $x$. Previous work [36] defined a cluster as a region with a mean intensity of greater than $+3\sigma$, where $\sigma$ is the standard deviation of the intensity for each weak lensing map. This provides a natural threshold for our groups: we can identify

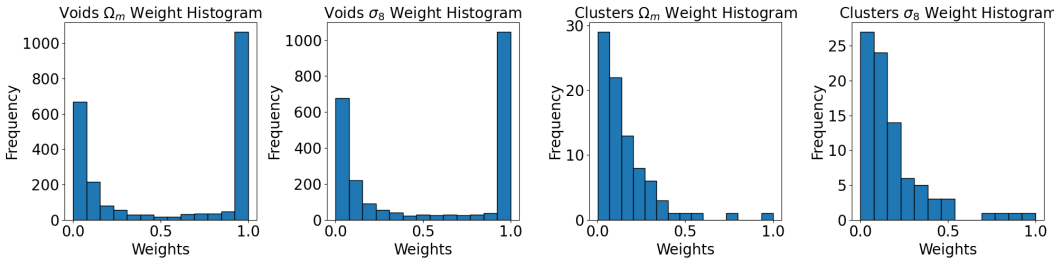



Figure 6: Voids (left two) are used more in prediction, weighing 100% in about half the cases. Clusters (right two) are used more in predicting $\sigma_8$ than $\Omega_m$.

groups containing clusters as those whose features have a mean intensity of $+3\sigma$. Specifically, we calculate

$$\text{Intensity}(x, S) = \frac{1}{|S|} \sum_{i:S_i>0} x_i$$

Then, a group $S$ is labeled as a cluster if Intensity$(x, S) \geq 3\sigma$. Similarly, Matilla et al. [36] define a void as a region with mean intensity less than 0. Then, a group $S$ is labeled as a cluster if Intensity$(x, S) < 0$.

### D.1 Cosmogrid Dataset

CosmoGridV1 is a suite of cosmological N-body simulations, spanning different cosmological parameters (including the parameters $\Omega_m$ and $\sigma_8$ considered in this work). They have been produced using a high performance N-body treecode for self-gravitating astrophysical simulations (PKDGRAV3). The output of the simulations are a series of snapshots representing the distribution of matter particles as a function of position on the sky; each snapshot represents the output of the simulation at a different cosmic time (and, therefore, represents a snapshot of the Universe at a different distance from the observer). The output of the simulations have been post-processed to produce weak lensing mass maps, which are weighted and projected maps of the mass distribution and that can be estimated from current weak lensing observations (e.g., [24]).

### D.2 Preprocessing

For input features used in CosmoGridV1, we segment the weak lensing maps using a contour-based segmentation method watershed [7] implemented in scikit-image. We use watershed instead of a patch segmenter because watershed is able to segment out potential input features that can constitue voids and clusters. In our preliminary experiments, we also experimented with patch, quickshift [21] for segmentation. Only the model finetuned on watershed segments is able to obtain comparable MSE loss as the original model.

### D.3 Voids and Clusters Serve Purpose in Predicting Different Parameters

We find that voids have especially higher weights for predicting $\Omega_m$, with average of $55.4\%$ weight for $\Omega_m$ over $54.0\%$ weight for $\sigma_8$. Clusters, especially high-significance ones, have higher weights for predicting $\sigma_8$, with average of $14.8\%$ weight for $\sigma_8$ over $8.8\%$ weight for $\Omega_m$. With relaxed thresholds of ($\geq +2\sigma$) for clusters ($\leq 0$) for voids, the whole distribution of weights can be seen from the histograms in Figure 6.

## E Related Works

**Post-hoc Attributions.** There have been a lot of previous work in attributing deep models post-hoc. One way is to use gradients of machine learning models, including using gradients themselves [47, 5, 50, 6], gradient $\times$ inputs [53, 13, 51] and through propagation methods [43, 52, 4, 48, 38].

Another type of attribution includes creating a surrogate model to approximate the original model [42, 33, 29]. Other works use input perturbation including erasing partial inputs [39, 58, 27, 30, 26,

43, 12] and counterfactual perturbation that can be manual [27] or automatic [9, 64, 3, 61]. While the above methods focus on individual features, Tsang et al. [56] investigates feature interactions. Multiple works have shown the failures of feature attributions [8, 54, 1, 28]

**Built-in Attributions.** For built-in feature attributions, one line of work first predict which input features to use, and then predict using only the selected features, including FRESH [23] and [20]. FRESH [23] has a similar structure as our model, with a rationale extractor that extracts partial input features, and another prediction model to predict only on the selected features. Another line of work that learns different modules when using different input features, including CAM [31], $GA^2M$ [32], and NAM [2]. The key difference of our work from the previous built-in attributions is that we use grouped attributions while previous works attribute to input features individually.