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]

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 [Mol22].

For example, in medicine, ML models can assist physicians in diagnosing a variety of lung, heart, and other chest conditions from X-ray images [Raj+17; Cha+22; Zha+22; Ter+23]. However, physicians only trust the decision of the model if an explanation identifies regions of the X-ray that make sense [Rey+20]. Such explanations are increasingly requested as new biases are discovered in these models [Glo+22].

The field has proposed a variety of feature attribution methods to explain ML models. One category consist of post-hoc attributions [RSG16; LL17; PDS18; Sel+16; STY17a], which have the benefit of being able to apply to any model. Another category of approaches instead build feature attributions directly into the model [WP19; Jai+20; SVZ14a; SFH17; CBD15], 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 [LAC22]. 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 would assign a positive score irrelevant regions. Unfortunately, studies have found that many post-hoc feature attributions do not satisfy basic sanity checks for faithfulness [LAC22].

[1] Code is available at `https://github.com/DebugML/sop`

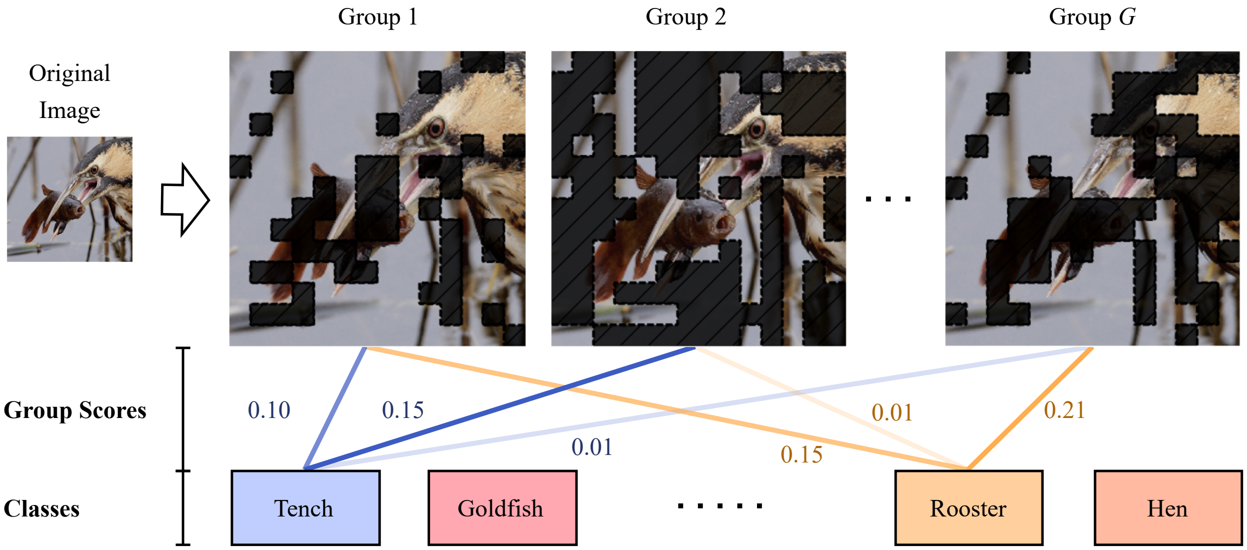


Figure 1: Visualization of grouped attributions. For a set of group attributions, scores are assigned to groups of features instead of individual features. Each group has a binary assignment in whether or not to include each feature. The score for each group represents how much each group of features 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".

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 must incur at least exponentially large error in tests of faithfulness for simple settings. We further 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 explanation for ML models. However, numerous studies have found that feature attributions fail basic sanity checks [Ade+18; STY17b] and interpretability tests [Kin+19; Bil+22].

Perturbation tests are a widely-used technique for evaluating the faithfulness of an explanation [PDS18; VL20a; DeY+20]. 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 a deletion-style test for a feature attribution on a subset of features.

Definition 2.1. (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

$$\mathrm{DelErr}(\alpha, S) = \left| f(x) - f(x_{\neg S}) - \sum_{i \in S} \alpha_i \right| \quad \text{where} \;\; (x_{\neg S})_j = \begin{cases} x_j & \text{if } j \notin S \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

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. Intuitively, a faithful attribution score of the ith feature should reflect the change in model prediction after the ith feature is removed and thus have low deletion error. We can formalize an analogous error for insertion-style tests as follows:

Definition 2.2. (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

$$\mathrm{InsErr}(\alpha, S) = \left| f(x_S) - f(0_d) - \sum_{i \in S} \alpha_i \right| \quad \text{where} \;\; (x_S)_j = \begin{cases} x_j & \text{if } j \in S \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The total insertion error is $\sum_{S \in \mathcal{P}} \mathrm{InsErr}(\alpha, S)$ 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. 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 [PDS18].

2.1 Feature Attributions Incur a Minimum of Exponential Error

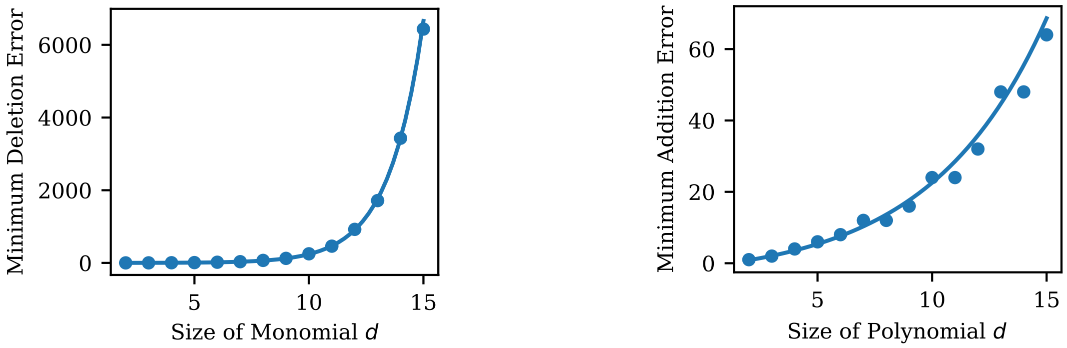
In this section, we provide two simple polynomial settings where any choice of feature attribution is guaranteed to incur at least exponential deletion and insertion error across all possible subsets. The key property in these examples is the presence of highly correlated features, which pose an insurmountable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

Theorem 2.3 (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 $(\gamma_1, \gamma_0) = (0.664, -1.159)$.*

In other words, Theorem 2.3 states that the total deletion error of any feature attribution of a monomial will grow exponentially with respect to the dimension, as visualized in Figure 2a. 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 2.4.

Lemma 2.4 (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, as shown in Figure 2b. The two terms in the binomial must have some overlapping features or else the problem reduces to a monomial.



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

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

Figure 2: The minimum (a) deletion error of monimials of size d and (b) insertion errors of binomials 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.

Theorem 2.5 (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 an approximate lower bound of* $\exp(\lambda_2 d + \lambda_1) + \lambda_0$ *error in insertion-based faithfulness tests, where* $(\lambda_2, \lambda_1, \lambda_0) = (0.198, 1.332, 4.778)$ *and* $d \leq 20$.

In combination, Theorems 2.3 and 2.5 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.

2.2 Grouped Attributions Overcome Barriers for Feature Attributions

The inherent limitations of feature attributions stems 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 2.3 and 2.5.

To explain these correlated effects, we explore a different type of attributions 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 2.6.

Definition 2.6. 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 ith 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 ith group of features S_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 exists grouped attributions that can succinctly describe the earlier settings from Theorems 2.3 and 2.5 with zero insertion and deletion error (Corollary A.4 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.

[2] The proof technique for Theorems 2.3 and 2.5 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.

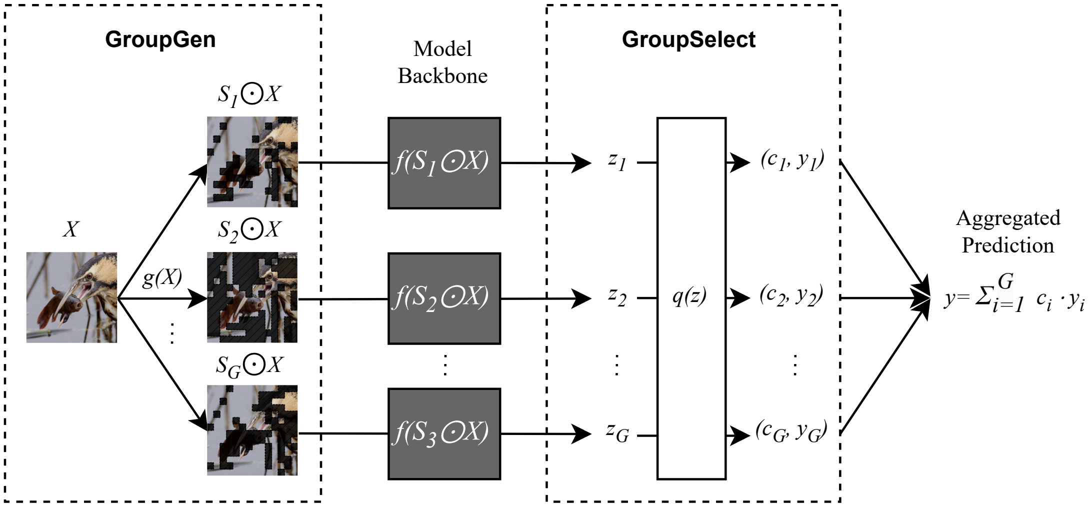


Figure 3: Structure of a Sum-of-Parts Model. A group generator g first generates groups of features. Each group of features $S_i \odot X$ is then passed through the black-box model to obtain the group embedding z_i. A group selector q then assigns a score c_i to each group i's representation. The partial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

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.

These groups and their corresponding scores form the grouped attribution of SOP. To make a prediction our approach linearly aggregates the prediction of each group according to the score to produce a final prediction. Since the prediction for a group solely relies on the features within the group, the grouped attribution is faithful-by-construction to the prediction.

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 [Vas+17] 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_q, W_k, W_v 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 [MA16] without the value:

$$\mathsf{GroupGen}(X) = \mathsf{sparsemax}\left(\frac{W_q X (W_k X)^T}{\sqrt{d}}\right) \tag{3}$$

where $W_q, W_k \in \mathbb{R}^d$. 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 now choose a sparse subset of these groups to use 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

$$\mathsf{GroupSelect}(z_1, \ldots, z_G) = \mathsf{sparsemax}\left(\frac{W_{q'}C(W_{k'}z)^T}{\sqrt{h}}\right), Cz^T \tag{4}$$

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 that are 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 scores c_i. As we are "summing" weighted "parts" of inputs, we call this a Sum-of-Parts model, the complete algorithm of which can be found in Algorithm 1.

4 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.

4.1 Experimental Setups

We evaluate SOP on ImageNet [Rus+15] for single-label and PASCAL VOC 07 [Eve+10] for multi-label classification. We use Vision Transformer [Dos+21] as our backbone. More information about training and datasets are in Appendix C.1.

We compare against different types of baselines:

1. *Surrogate-model-based*: LIME [RSG16], SHAP [LL17]

2. *Perturbation-based*: RISE [PDS18]

3. *Gradient-based*: GradCAM [Sel+16], IntGrad [STY17a]

4. *Built-in explanation*: FRESH [Jai+20]

To evaluate our approach, we use interpretability metrics that are standard practice in the literature for feature attributions [PDS18; VL20b; Jai+20]. We summarize these metrics as follows and provide precise descriptions in Appendix C.2:

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 insertion and deletion tests that are standard for feature attributions [PDS18]. 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.

		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**
	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
	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**
	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
	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.

4.2 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 multiple groups are potentially the advantage of SOP over single-group attributions from FRESH to model interactions between different groups of features.

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, Das, and Saenko [PDS18] 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 [Jai+22], we simply replace features entirely with zeros which correspond to gray pixels. Also, to accommodate the tests designed for individual-feature attributions, we first perform weighted sum on the groups of features from SOP and then do insertion and deletion tests on the aggregated attributions.

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.

5 Case Study: Cosmology

While outperforming other methods on standard metrics shows the advantage of our grouped attributions, the ultimate goal of interpretability methods is for domain experts to use these tools and be able to use the explanations in real settings. To validate the usability of our approach, 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. We find that the groups generated with SOP contain semantically meaningful structures to cosmologists. The resulting scores of these groups led to findings linking certain cosmological structures to the initial state of the universe, some of which were surprising and previously not known.

Weak lensing maps in cosmology calculate the spatial distribution of matter density in the universe using precise measurements of the shapes of $\sim$100 million galaxies [Gat+21]. The shape of each galaxy is distorted (sheared and magnified) due to the curvature of spacetime induced by mass inhomogenities as light travels towards us. Cosmologists have techniques that can infer the distribution of mass in the universe from these distortions, resulting in a weak lensing map [Jef+21a].

Cosmologists hope to use weak lensing maps to predict two key parameters related to the initial state of the universe: Ω_m and σ_8. Ω_m captures the average energy density of all matter in the universe (relative to the total energy density which includes radiation and dark energy), while σ_8 describes the fluctuation of matter distribution (see e.g. [Abb+22]). From these parameters, a cosmologist can simulate how cosmological structures, such as galaxies, superclusters and voids, develop throughout cosmic history. However, Ω_m and σ_8 are not directly measurable, and the inverse relation from cosmological structures in the weak lensing map to Ω_m and σ_8 is unknown.

One approach to inferring Ω_m and σ_8 from weak lensing maps, as demonstrated for example by Ribli et al. [Rib+19], Matilla et al. [Mat+20], and Fluri et al. [Flu+22], 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 they predict Ω_m and σ_8. As a result, the following remains an open question in cosmology:

What structures from weak lensing maps can we use to infer the cosmological parameters Ω_m and σ_8?

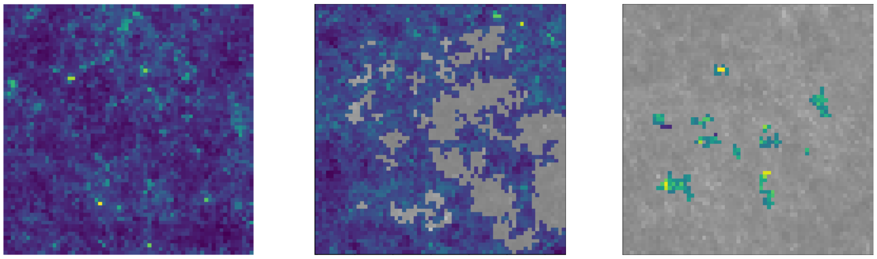
In collaboration with expert cosmologists, we use convolutional networks trained to predict Ω_m and σ_8 as the backbone of an SOP model to get accurate predictions with faithful group attributions. Crucially, the guarantee of faithfulness in SOP provides confidence that the attributions reflect how the model makes its prediction, as opposed to possibly being a red herring. We then interpret and analyze these attributions and understand how structures in weak lensing maps of CosmoGridV1 [Kac+23] influence Ω_m and σ_8.

Cosmological findings: Our initial findings come from grouped attributions that correspond to two known structures in the weak lensing maps (as identified by cosmologists): voids and clusters. Voids are large regions that are under-dense relative to the mean density and appear as dark regions in the weak lensing mass maps, whereas clusters are areas of concentrated high density and appear as bright dots. Figure 4 shows an example of voids (middle panel) and an example of clusters (right panel), both of which are automatically learned as groups in the SOP model without supervision. We use standard deviation σ away from the mean mass intensity for each map to define voids and clusters, where voids are groups that have mean density ≤ 0 and clusters are groups that have overdensity $\geq +3\sigma$. A precise definition of these structures is provided in Appendix D.

We summarize the discoveries that we made with cosmologists on how clusters and voids influence the prediction of Ω_m and σ_8 as follows:

1. A new finding of our work relates to the distinction between the two parameters, Ω_m and σ_8 (which are qualitatively different for cosmologists). We find that voids have especially higher weights for predicting Ω_m, with average of 55.4% weight for Ω_m over 54.0% weight for σ_8. Clusters, especially high-significance ones, have higher weights for predicting σ_8, with average of 14.8% weight for σ_8 over 8.8% weight for Ω_m. With relaxed thresholds of ($\geq +2\sigma$) for clusters (≤ 0) for voids, the whole distribution of weights can be seen from the histograms in Figure 5.

2. Using a higher threshold of $+2$ or $+3\sigma$ gives the clusters higher weight especially for σ_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 σ_8, the parameter for fluctuations.



Weak Lensing Map Void Cluster

Figure 4: A weak lensing mass map (left) contains large dark areas which are voids, and hot pixels which are clusters. Voids (middle) are darker and larger areas in weak lensing maps. Clusters (right) are small groups of hot pixels. We find that voids are used more in predicting both Ω_m and σ_8 from such noiseless maps. Clusters are used less in general, but comparatively more for σ_8.

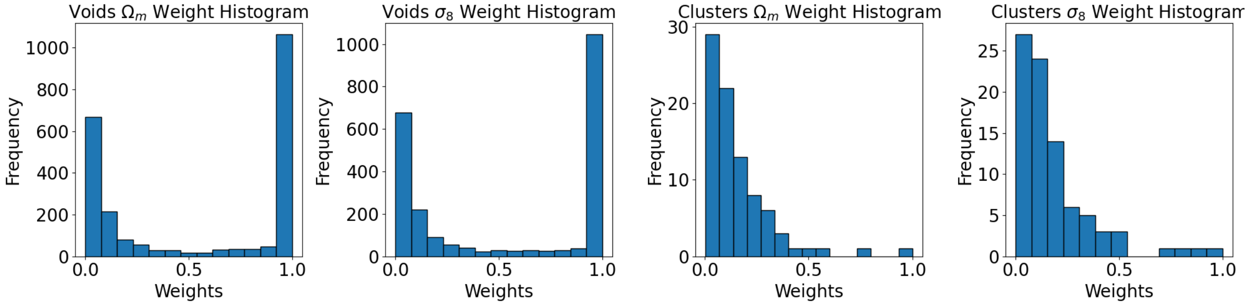


Figure 5: Voids (left two) are used more (have higher weights) in prediction, weighing 100% in about half the cases. Clusters (right two) are used less (have lower weights) in general, but more in predicting σ_8 than Ω_m.

3. In general, the voids have higher weights for prediction than the clusters in the noiseless maps we have used. This is consistent with previous work [Mat+20] that voids are the most important feature in prediction. This finding was intriguing to cosmologists. Given that the previous work relied on gradient-based saliency maps, it is important that we find consistent results with our attention-based wrapper.

It will be interesting to explore how these results change as we mimic realistic data by adding noise and measurement artifacts. Other aspects worth exploring are the role of “super-clusters” that contain multiple clusters, and how to account for the fact that voids occupy much larger areas on the sky than clusters (i.e., should we be surprised that they perform better?).

6 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 [Sel+16; Bae+09; SVZ14b; BF20], gradient × inputs [STY17a; DDF14; Smi+17] and through propagation methods [RSG18; Spr+14; Bac+15; SGK17; Mon+17].

Another type of attribution includes creating a surrogate model to approximate the original model [RSG16; LL17; Lau+18]. Other works use input perturbation including erasing partial inputs [PDS18; VL20a; KHL20; LMJ17; KCA17; RSG18; De +20] and counterfactual perturbation that can be manual [KHL20] or automatic [Cal+22; Zmi+19; Ami+22; Wu+21]. While the above methods focus on individual features, Tsang, Rambhatla, and Liu [TRL20] investigates feature interactions. Multiple works have shown the failures of feature attributions [Bil+22; STY17b; Ade+18; Kin+19]

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 [Jai+20] and [GHG20]. FRESH

[Jai+20] 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. The difference is that FRESH only selects one group of features while we select multiple and allow different attribution scores for each group. Another line of work learns different modules when using different input features, including CAM [LCG12], GA^{2}M [Lou+13], and NAM [Aga+21]. The key difference of our work from these works is that we use grouped attributions to model complex groups of features with different sizes, while previous works attribute to input features individually or in pairs.

7 Conclusion

In this paper, we identify a fundamental barrier for feature attributions in satisfying faithfulness tests. These limitations can be overcome when using grouped attributions which assign scores to groups of features instead of individual features. To generate faithful grouped attributions, we develop the SOP model, which uses a group generator to create groups of features, and a group selector to score groups and make a faithful prediction. The group attributions from SOP improve upon standard feature attributions on the majority of insertion and deletion-based interpretability metrics.

We applied the faithful grouped attributions from SOP to discover cosmological knowledge about the growth of structure in an expanding universe. Our groups are semantically meaningful to cosmologists and revealed new properties in cosmological structures such as voids and clusters that merit further investigation. We hope that this work paves the way for further scientific discoveries from faithful explanations of deep learning models that capture complex and unknown patterns.

References

[Abb+22] T. M. C. Abbott et al. “Dark Energy Survey Year 3 results: Cosmological constraints from galaxy clustering and weak lensing”. In: *Physical Review D* 105.2 (Jan. 2022). DOI: 10.1103/physrevd.105.023520. URL: https://doi.org/10.1103%2Fphysrevd.105.023520.

[Ade+18] Julius Adebayo et al. “Sanity checks for saliency maps”. In: *Advances in neural information processing systems* 31 (2018).

[Aga+21] Rishabh Agarwal et al. “Neural Additive Models: Interpretable Machine Learning with Neural Nets”. In: *Advances in Neural Information Processing Systems*. Ed. by A. Beygelzimer et al. 2021. URL: https://openreview.net/forum?id=wHkKTW2wrmm.

[Ami+22] Afra Amini et al. “Naturalistic Causal Probing for Morpho-Syntax”. In: *Transactions of the Association for Computational Linguistics* 11 (2022), pp. 384–403. URL: https://api.semanticscholar.org/CorpusID:248811730.

[Bac+15] Sebastian Bach et al. “On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation”. In: *PLoS ONE* 10 (2015). URL: https://api.semanticscholar.org/CorpusID:9327892.

[Bae+09] David Baehrens et al. *How to Explain Individual Classification Decisions*. 2009. arXiv: 0912.1128 [stat.ML].

[Beu23] Serge Beucher. *Image Segmentation and Mathematical Morphology*. Accessed: September 29, 2023. 2023. URL: https://people.cmm.minesparis.psl.eu/users/beucher/wtshed.html.

[BF20] Jasmijn Bastings and Katja Filippova. “The elephant in the interpretability room: Why use attention as explanation when we have saliency methods?” In: *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.

[Bil+22] Blair Bilodeau et al. “Impossibility Theorems for Feature Attribution”. In: *ArXiv* abs/2212.11870 (2022). URL: https://api.semanticscholar.org/CorpusID:254974246.

[Cal+22] Nitay Calderon et al. “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)*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 7727–7746. DOI: 10.18653/v1/2022.acl-long.533. URL: https://aclanthology.org/2022.acl-long.533.

[CBD15] 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.

[Cha+22] Pierre Chambon et al. “RoentGen: vision-language foundation model for chest x-ray generation”. In: *arXiv preprint arXiv:2211.12737* (2022).

[DDF14] Misha Denil, Alban Demiraj, and Nando de Freitas. “Extraction of Salient Sentences from Labelled Documents”. In: *ArXiv* abs/1412.6815 (2014). URL: https://api.semanticscholar.org/CorpusID:9121062.

[De +20] Nicola De Cao et al. “How do Decisions Emerge across Layers in Neural Models? Interpretation with Differentiable Masking”. In: *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.

[DeY+20] Jay DeYoung et al. “ERASER: A Benchmark to Evaluate Rationalized NLP Models”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 4443–4458. DOI: 10.18653/v1/2020.acl-main.408. URL: https://aclanthology.org/2020.acl-main.408.

[Dos+21] Alexey Dosovitskiy et al. “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.

[Eve+10] Mark Everingham et al. "The pascal visual object classes (voc) challenge". In: *International journal of computer vision* 88 (2010), pp. 303–338.

[Flu+22] Janis Fluri et al. "Full wCDM analysis of KiDS-1000 weak lensing maps using deep learning". In: *Physical Review D* 105.8 (Apr. 2022). DOI: 10.1103/physrevd.105.083518. URL: https://doi.org/10.1103%2Fphysrevd.105.083518.

[Gat+21] M. Gatti et al. "Dark energy survey year 3 results: weak lensing shape catalogue". In: *MNRAS* 504.3 (July 2021), pp. 4312–4336. DOI: 10.1093/mnras/stab918.

[GHG20] 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*. Online: Association for Computational Linguistics, Nov. 2020, pp. 1080–1095. DOI: 10.18653/v1/2020.findings-emnlp.97. URL: https://aclanthology.org/2020.findings-emnlp.97.

[Glo+22] Ben Glocker et al. "Risk of bias in chest X-ray foundation models". In: *arXiv preprint arXiv:2209.02965* (2022).

[Gra06] L. Grady. "Random Walks for Image Segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.11 (2006), pp. 1768–1783. DOI: 10.1109/TPAMI.2006.233.

[Jai+20] Sarthak Jain et al. "Learning to Faithfully Rationalize by Construction". In: *Annual Meeting of the Association for Computational Linguistics*. 2020.

[Jai+22] Saachi Jain et al. "Missingness bias in model debugging". In: *arXiv preprint arXiv:2204.08945* (2022).

[Jef+21a] N Jeffrey et al. "Dark Energy Survey Year 3 results: Curved-sky weak lensing mass map reconstruction". In: *Monthly Notices of the Royal Astronomical Society* 505.3 (May 2021), pp. 4626–4645. DOI: 10.1093/mnras/stab1495. URL: https://doi.org/10.1093%2Fmnras%2Fstab1495.

[Jef+21b] N. Jeffrey et al. "Dark Energy Survey Year 3 results: Curved-sky weak lensing mass map reconstruction". In: *MNRAS* 505.3 (Aug. 2021), pp. 4626–4645. DOI: 10.1093/mnras/stab1495.

[Kac+23] Tomasz Kacprzak et al. "CosmoGridV1: a simulated LambdaCDM theory prediction for map-level cosmological inference". In: *JCAP* 2023.2, 050 (Feb. 2023), p. 050. DOI: 10.1088/1475-7516/2023/02/050.

[KCA17] Ákos Kádár, Grzegorz Chrupała, and Afra Alishahi. "Representation of Linguistic Form and Function in Recurrent Neural Networks". In: *Computational Linguistics* 43.4 (Dec. 2017), pp. 761–780. DOI: 10.1162/COLI_a_00300. URL: https://aclanthology.org/J17-4003.

[KHL20] Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. *Learning the Difference that Makes a Difference with Counterfactually-Augmented Data*. 2020. arXiv: 1909.12434 [cs.CL].

[Kin+19] Pieter-Jan Kindermans et al. "The (Un)reliability of Saliency Methods". In: *Lecture Notes in Computer Science* (2019), pp. 267–280. 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.

[LAC22] Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. *Towards Faithful Model Explanation in NLP: A Survey*. 2022. arXiv: 2209.11326 [cs.CL].

[Lau+18] Thibault Laugel et al. *Defining Locality for Surrogates in Post-hoc Interpretablity*. 2018. arXiv: 1806.07498 [cs.LG].

[LCG12] 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. Beijing, China: Association for Computing Machinery, 2012, pp. 150–158. ISBN: 9781450314626. DOI: 10.1145/2339530.2339556. URL: https://doi.org/10.1145/2339530.2339556.

[LL17] 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. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 4768–4777. ISBN: 9781510860964.

[LMJ17] Jiwei Li, Will Monroe, and Dan Jurafsky. *Understanding Neural Networks through Representation Erasure*. 2017. arXiv: 1612.08220 [cs.CL].

[Lou+13] Yin Lou et al. "Accurate Intelligible Models with Pairwise Interactions". In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '13. Chicago, Illinois, USA: Association for Computing Machinery, 2013, pp. 623–631. ISBN: 9781450321747. DOI: 10.1145/2487575.2487579. URL: https://doi.org/10.1145/2487575.2487579.

[MA16] André F. T. Martins and Ramón Fernández Astudillo. "From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification". In: *ArXiv* abs/1602.02068 (2016).

[Mat+20] José Manuel Zorrilla Matilla et al. "Interpreting deep learning models for weak lensing". In: *Physical Review D* 102.12 (Dec. 2020). ISSN: 2470-0029. DOI: 10.1103/physrevd.102.123506. URL: http://dx.doi.org/10.1103/physrevd.102.123506.

[Mol22] Christoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2nd ed. Independently published, 2022. URL: https://christophm.github.io/interpretable-ml-book.

[Mon+17] Grégoire Montavon et al. "Explaining nonlinear classification decisions with deep Taylor decomposition". In: *Pattern Recognition* 65 (2017), pp. 211–222. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2016.11.008. URL: https://www.sciencedirect.com/science/article/pii/S0031320316303582.

[PDS18] Vitali Petsiuk, Abir Das, and Kate Saenko. "RISE: Randomized Input Sampling for Explanation of Black-box Models". In: *British Machine Vision Conference*. 2018.

[Raj+17] Pranav Rajpurkar et al. "Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning". In: *arXiv preprint arXiv:1711.05225* (2017).

[Rey+20] Mauricio Reyes et al. "On the Interpretability of Artificial Intelligence in Radiology: Challenges and Opportunities". In: *Radiology: Artificial Intelligence* 2.3 (2020), e190043. DOI: 10.1148/ryai.2020190043.

[Rib+19] Dezső Ribli et al. "Weak lensing cosmology with convolutional neural networks on noisy data". In: *Monthly Notices of the Royal Astronomical Society* 490.2 (Sept. 2019), pp. 1843–1860. ISSN: 0035-8711. DOI: 10.1093/mnras/stz2610. eprint: https://academic.oup.com/mnras/article-pdf/490/2/1843/30194757/stz2610.pdf. URL: https://doi.org/10.1093/mnras/stz2610.

[RSG16] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016).

[RSG18] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Anchors: High-Precision Model-Agnostic Explanations". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1 (Apr. 2018). DOI: 10.1609/aaai.v32i1.11491. URL: https://ojs.aaai.org/index.php/AAAI/article/view/11491.

[Rus+15] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

[Sel+16] Ramprasaath R. Selvaraju et al. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization". In: *International Journal of Computer Vision* 128 (2016), pp. 336–359. URL: https://api.semanticscholar.org/CorpusID:15019293.

[SFH17] 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. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 3859–3869. ISBN: 9781510860964.

[SGK17] 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. Sydney, NSW, Australia: JMLR.org, 2017, pp. 3145–3153.

[Smi+17] Daniel Smilkov et al. *SmoothGrad: removing noise by adding noise*. 2017. arXiv: 1706.03825 [cs.LG].

[Spr+14] Jost Tobias Springenberg et al. “Striving for Simplicity: The All Convolutional Net”. In: *CoRR* abs/1412.6806 (2014). URL: `https://api.semanticscholar.org/CorpusID:12998557`.

[STY17a] 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`.

[STY17b] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. *Axiomatic Attribution for Deep Networks*. 2017. arXiv: `1703.01365 [cs.LG]`.

[SVZ14a] 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.

[SVZ14b] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*. 2014. arXiv: `1312.6034 [cs.CV]`.

[Ter+23] Chakkrit Termritthikun et al. “Explainable Knowledge Distillation for On-Device Chest X-Ray Classification”. In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2023).

[TRL20] Michael Tsang, Sirisha Rambhatla, and Yan Liu. “How does This Interaction Affect Me? Interpretable Attribution for Feature Interactions”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 6147–6159. URL: `https://proceedings.neurips.cc/paper_files/paper/2020/file/443dec3062d0286986e21dc0631734c9-Paper.pdf`.

[Vas+17] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: `https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

[VL20a] Bhavan Vasu and Chengjiang Long. “Iterative and Adaptive Sampling with Spatial Attention for Black-Box Model Explanations”. In: *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)* (Mar. 2020). DOI: `10.1109/wacv45572.2020.9093576`. URL: `http://dx.doi.org/10.1109/WACV45572.2020.9093576`.

[VL20b] 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)*. Mar. 2020.

[WP19] 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)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 11–20. DOI: `10.18653/v1/D19-1002`. URL: `https://aclanthology.org/D19-1002`.

[Wu+21] Tongshuang Wu et al. “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)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 6707–6723. DOI: `10.18653/v1/2021.acl-long.523`. URL: `https://aclanthology.org/2021.acl-long.523`.

[Zha+18] Jianming Zhang et al. “Top-down neural attention by excitation backprop”. In: *International Journal of Computer Vision* 126.10 (2018), pp. 1084–1102.

[Zha+22] Ran Zhang et al. “A Generalizable Artificial Intelligence Model for COVID-19 Classification Task Using Chest X-ray Radiographs: Evaluated Over Four Clinical Datasets with 15,097 Patients”. In: *arXiv preprint arXiv:2210.02189* (2022).

[Zmi+19] Ran Zmigrod et al. “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*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 1651–1661. DOI: `10.18653/v1/P19-1161`. URL: `https://aclanthology.org/P19-1161`.

A Theorem Proofs for Section 2

Theorem 2.3 (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 α_i is faithful if it reflects a contribution of α_i to the prediction. Then, the feature attribution α^* 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

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

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 α^* 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 2a. □

Lemma A.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 $S \neq \{1, \ldots, d\}$, $p(x_S) = 0 = \sum_{i \in S} \alpha_i$ so α incurs no insertion error for all but one subset. For the last subset $S = \{1, \ldots, d\}$, the insertion error is 1. Therefore, the total insertion error is at most 1 for $\alpha = 0_d$. □

Theorem 2.5 (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 an approximate lower bound of $\exp(\lambda_2 d + \lambda_1) + \lambda_0$ error in insertion-based faithfulness tests, where $(\lambda_2, \lambda_1, \lambda_0) = (0.198, 1.332, 4.778)$ and $d \leq 20$.*

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 α^* using ECOS in the `cvxpy` library for $d \in \{2, \dots, 20\}$. 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 2b. □

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 A.2. (Grouped deletion error) The *grouped deletion error* of a 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 A.3. (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 A.4. *Consider p_1 and p_2, the polynomials from Theorem 2.3 and Theorem 2.5. Then, there exists a grouped attribution with zero deletion and insertion error for both polynomials.*

Proof. Let $[d]$ denote $\{1, \dots, d\}$. 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

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

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. □

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
```

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

C.1 Training

ImageNet ImageNet [Rus+15] 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 [Eve+10] is an object detection dataset with 20 classes. We train with multilabel classification training, but predict with single binary labels for target classes [Zha+18].

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

C.2 Evaluation Details

C.2.1 Accuracy

We evaluate on accuracy to measure if the wrapped model has comparable performance with the original model, following Jain et al. [Jai+20]. For post-hoc explanations, the performance shown will be the performance of the original model, since they are not modifying the model.

C.2.2 Deletion and Insertion

Petsiuk, Das, and Saenko [PDS18] 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 reflect the most attributed features, then the prediction consistency should drop down from the stsart and result in a lower deletion score.

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

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

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.

C.2.3 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.

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 [Mat+20] defined a cluster as a region with a mean intensity of greater than $+3\sigma$, where σ is the standard deviation of the intensity for each weak lensing map. This provides a natural threshold for our groups: we can identify 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 $\text{Intensity}(x, S) \geq 3\sigma$. Similarly, Matilla et al. [Mat+20] define a void as a region with mean intensity less than 0. Then, a group S is labeled as a cluster if $\text{Intensity}(x, S) < 0$.

D.1 Cosmogrid Dataset

CosmoGridV1 is a suite of cosmological N-body simulations, spanning different cosmological parameters (including the parameters Ω_m and σ_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., [Jef+21b]).

D.2 Preprocessing

For input features used in CosmoGridV1, we segment the weak lensing maps using a contour-based segmentation method watershed [Beu23] implemented in scikit-image. We use watershed instead of a patch segmenter because watershed is able to segment out potential input features that can constitute voids and clusters. In our preliminary experiments, we also experimented with patch, quickshift [Gra06] for segmentation. Only the model finetuned on watershed segments is able to obtain comparable MSE loss as the original model.