University of Tübingen
Faculty of Science
Department of Computer Science
Theory of Machine Learning

Master thesis

Feature Attribution Methods: Shapley Values on Logical Formulas and Improving Explanations by Averaging

in

Cognitive Science

by

Frieder Göppert
Matrikelnummer 4001694
frieder.goeppert@student.uni-tuebingen.de

2.11.2021

First and second supervisor:
Prof. U. Luxburg
Prof. Z. Ataka

Abstract

This master thesis is concerned with methods explaining the output of a previously trained machine learning model by assigning importance values to its input features, so called (local post-hoc) feature attribution methods. We first investigate Shapley values, one specific method motivated by cooperative game theory that satisfies particularly desirable properties, on a simple class of non-linear models: logical formulas. Specifically, we ask what information about a global formula with dimension d is carried by a local explanation on one example input. We find that assuming independent, equally weighted features, an initial preservation of information for $d = 2$ diminishes with increasing d. Next to these theoretical results specific to Shapley values, this investigation led to an observation about feature attribution methods more general. We argue that in settings that allow for many accurate models relying on different features, a feature attribution based on a single model may not be representative: the learning algorithm could have easily picked different features resulting also in different attributions. We demonstrate that averaging feature attributions over multiple models can improve explanations for many widely used feature attribution methods on artificial geometric images, that are redundant by design, as well as on more realistic natural image data.

Contents

Chapter 1

Introduction

With increasing complexity and impact, one key challenge of modern machine learning is to provide next to accurate learned models also useful explanations for their behavior. Explanations can help to debug a model, detect its (perhaps undesirable) biases, satisfy transparency requirements and overall improve the understanding of a model. They are therefore at the core to make models more robust, fair, transparent and interpretable – traits that are highly desirable but still too often lacking.

The approaches of how a useful explanation can look like are manifold. Some aim at constructing ad-hoc interpretable models, others focus on explaining already trained black box models in a post-hoc manner. There are approaches that globally explain the model's behavior on all inputs, others restrict themselves to explain the model locally given one specific example input. Lastly, given an example input some try to give counterfactual explanations of the form: if the input x would have been (minimally) changed to x', the model's output would have changed; others try to attribute the model's output on this example to its input features.

In this work we focus on this last class of explanation methods, known as (local post-hoc) feature attribution methods. Feature attributions are the state of the art explanations in image recognition problems where ad-hoc interpretable models of comparable accuracy to black box deep neural networks are often unavailable. Despite their wide employment, they often seem to yield unreliable results and few theoretical guarantees have been proven for them. With this work we aim at increasing the theoretical understanding of these methods and at making them more reliable.

First, we investigate Shapley values, one specific feature attribution method satisfying particularly desirable properties, on a simple class of models: logical formulas. Specifically we ask, what information do local Shapley explanations carry about the global formula of dimension d. This investigation has led us next to some theoretical results specific for Shapley values to a more general idea of how to improve feature attributions.

We argue that in settings that allow for many accurate models relying on different features, a feature attribution of a single model may not be representative for the learning algorithm in general: a learned model could have easily relied on other features, hence yielding different attributions. Such a representative explanation, answering the question which features are on expectation important for an accurate prediction on a given example, can be desirable and is potentially more reliable. In the second part, we investigate whether averaging attributions of multiple trained models can improve widely used feature attribution methods on image classification problems, where feature attribution methods are highly relevant and redundancies in the data are commonly assumed.

Supplementary materials, including code for simulations and experiments, as well as additional results and figures, are publicly available and can be accessed at `github.com/fgoepp`.

Chapter 2

Uniqueness Properties of Shapley Values on Logical Formulas

2.1 Motivation

The goal of feature attribution methods is to attribute the prediction of a previously trained model on a specific example to its input features. The question of how to correctly distribute these importance values among features has many answers and often depends on the question an explanation should answer and importantly which axioms it should obey. In this part, we exclusively focus on Shapley values, a feature attribution method motivated by cooperative game theory, that satisfies many particularly desirable theoretical properties. Despite their theoretical groundedness, little is known about their behavior on non-linear function classes. Here, we investigate Shapley values on simple non-linear models: logical formulas.

Consider the xor-problem: Two binary variables X_1 and X_2 are used to predict a label Y which is determined by their exclusive disjunction. This problem is simple and can be learned by many machine learning algorithms, for example a neural net with one hidden layer. As the prediction problem is solved, one may ask for its explanation, for instance: Given the example $x' = (1, 0)$ how important are both features for a trained model's prediction $f(x') = 1$?
Shapley values approach this question by framing features as players in a cooperative game, each joining a coalition of players, that is sharing its value, to predict the output. A feature's Shapley value is then determined as the average effect on the prediction of this feature joining a coalition. In our example

there are two possible coalitions a feature can join: the empty one and the one including only the other feature. Assuming independent, equally weighted features, in the first case the prediction would not change on expectation – still $\frac{1}{2}$ (to know the value of one variable gives us no information of the value of a exclusive disjunction with another variable); in the second case with both features in the coalition the model could accurately predict the label, raising the average prediction from $\frac{1}{2}$ to 1. Thus, the average effect of a feature joining a coalition – its Shapley value – is $\frac{1}{4}$.

Given the example x', its prediction $f(x')$ and the Shapley attribution $\phi(f, x') = (\frac{1}{4}, \frac{1}{4})$, we now ask what this local explanation can tell us about the global formula: Is this explanation unique to the xor-formula, or if not how many such local explanations are needed to determine the global formula. We investigate this question in a general setting of logical formulas with dimension d assuming independent, equally weighted features. In Section 2.4 we prove

- that for $d = 2$ one local explanation fully determines a formula (preservation of information).
- that for $d > 2$ this preservation of information does not in general hold anymore.
- sufficient conditions for the explanation-equivalence of formulas on one example.

Additionally, in Section 2.5 we demonstrate with simulations that with increasing dimension d the number of formulas with unique explanations on one example decreases; and the average number of examples needed to identify a formula increases.

2.2 Background: Shapley values

In this section, we present prior work relevant as background for our theoretical investigation of Shapley values on logical formulas. We also introduce core theoretical concepts.

Shapley values were first introduced as a solution for coalitional games in cooperative game theory (Shapley, 1953). Specifically, given players $U = \{1, ..., d\}$ and a value set-function $v : 2^U \to \mathbb{R}$ with $v(U) \neq 0, v(\emptyset) = 0$, they quantify the contribution of each player $i \in U$ to $v(U)$. Since the marginal contribution $(v(T \cup \{i\}) - v(T))$ of player i to a coalition $T \subseteq U$ may depend on T, its Shapley value ϕ_i is given as weighted average over all possible coalitions:

$$\phi_i(v) = \sum_{T \subseteq U \setminus \{i\}} \gamma_d(T) \; (v(T \cup \{i\}) - v(T))$$

with $\gamma_d(T) = \frac{|T|!(d-|T|-1)!}{d!}$. These weights can be interpreted as the number of orderings fixing the position of player i over the number of all possible orderings of players. More importantly they uniquely insure $v(U) = \sum_{i=1}^{d} \phi_i(v)$.

The concept of contribution of players to a coalition can be easily transferred to the attribution of features in prediction problems (e.g. Lundberg and Lee, 2017; Janzing et al., 2020). Given a model $f : (\mathcal{X}_1, ..., \mathcal{X}_d) \to \mathcal{Y}$ and an example point x, we can define $v(T) := f_T(x) - f_\emptyset(x)$ where f_T is some simplified 'baseline' of f using only features in coalition $T \subseteq U$, with $f_\emptyset = \mathbb{E}[f(x)]$. With that, Shapley values $\phi_i(f, x)$ are the only feature attributions that satisfy all of the following four properties (Lundberg and Lee, 2017; Aas et al., 2019):

(P1) Completeness: $f(x) - \mathbb{E}[f(x)] = \sum_{i=1}^{d} \phi_i(f, x)$.

(P2) Sensitivity: If $f_T = f_{T \cup \{i\}}, \forall T$, then $\phi_i(f, x) = 0$.

(P3) Linearity: For $a, b \in \mathbb{R}$ holds $\phi(af_1 + bf_2, x) = a\phi(f_1, x) + b\phi(f_2, x)$.

(P4) Symmetry: If for feature i and j holds $f_{T \cup \{i\}}(x) = f_{T \cup \{j\}}(x)$ for all coalitions T neither containing i nor j, then $\phi_i(f, x) = \phi_j(f, x)$.

These properties are particularly desirable for attributions and a violation often leads to inconsistent explanations. For instance, violating sensitivity implies an irrelevant feature is attributed by a non-zero importance. In addition, Shapley values satisfy (P5) Implementation-Invariance: If f_1 and f_2 are equal on all inputs, then $\phi(f_1, x) = \phi(f_2, x)$.

Recent work on Shapley values mostly focuses on efficiently estimating them, for example Shapley Value Sampling (Strumbelj and Kononenko, 2010), Kernel SHAP (Lundberg and Lee, 2017), or evaluating different choices for the baseline function f_T. Namely, Lundberg and Lee (2017) proposed to use the conditional expectation $f_T := \mathbb{E}[f(x_T, X_{\bar{T}})|x_T)]$. However, Janzing et al. (2020) showed that this choice can lead to undesirable attributions in the setting of dependent features. They propose instead to use the marginal expectation $f_T := \mathbb{E}[f(x_T, X_{\bar{T}})]$. Lastly, for the model class of linear functions, Shapley values have shown to be given explicitly by $\phi_i(f, x) = w_i(x_i - \mathbb{E}[x_i])$, where

w_i are the model coefficients (Štrumbelj and Kononenko, 2014).

Our approach is novel, as we investigate Shapley values with logical formulas on simple non-linear functions. We rely on explicit calculations of Shapley values, as our main computational bottleneck is the number of models and not the number of coalitions, and aim at theoretical bounds on the quality of Shapley values. Lastly, we preliminary circumvent the baseline choice problem by assuming independent features.

2.3 Framework: Definitions and assumptions

In this section we formally define our theoretical framework, that is the definitions and assumptions we use.

For completeness, we summarize again the definition of Shapley values as feature attributions.

Definition 1 (Shapley values)**.** *Given a model $f : (\mathcal{X}_1, ..., \mathcal{X}_d) \rightarrow \mathcal{Y}$ and an example x, the Shapley value of feature i is given by*

$$\phi_i(f, x) = \sum_{T \subseteq U \setminus \{i\}} \frac{|T|!(d - |T| - 1)!}{d!} (f_{T \cup \{i\}}(x) - f_T(x))$$

where $U = \{1, ..., d\}$ is the set of all feature indices and f_T a baseline function using only features from the coalition T.

As we are interested in uniqueness of Shapley values for one prediction on one example, it will be useful to define a mapping, we call Shapley explanation, that includes all three concepts.

Definition 2 (Shapley explanation)**.** *Given a model $f : \mathcal{X} \rightarrow \mathcal{Y}$ and an example $x \in \mathcal{X}$, the Shapley explanation ψ is given by*

$$\psi(x, f) = (x, f(x), \phi(f, x)).$$

Feature independence and equal weights and are common simplifying assumptions. We discuss their limitations later.

Assumption 1 (Independent features)**.** *The features $X_1, ..., X_d$ are pairwise independent:*

$$\forall i, j : i \neq j \implies p(X_i = x_i) = p(X_i = x_i | X_j = x_j)$$

Assumption 2 (Equally weighted features)**.** *The features $X_1, ..., X_d$ are equally weighted:*

$$\forall i : p(X_i = 0) = p(X_i = 1)$$

Assuming independent features, it is unproblematic to define the Shapley baseline function as follows.

Assumption 3 (Shapley baseline function)**.** *The baseline function for Shapley values is*

$$f_T(x) := \mathbb{E}[f(x_T, X_{\bar{T}})],$$

that is the expected value of f with realisations of features in the coalition (x_T) over the distribution of features not in the coalition ($X_{\bar{T}}$).

In our setting, formulas are the ground truth, however Shapley attributions are based on a learned model. Therefore we make the following assumption, and – for brevity – treat model and formula from now on as equivalent.

Assumption 4 (Formulas are accurately learned by models)**.** *Any logical formula $f^* : \{0,1\}^d \to \{0,1\}$ can be accurately learned by some (sufficiently complex) model $\hat{f}$, that is*

$$\forall x \in \{0,1\}^d : f^*(x) = \hat{f}(x).$$

Note that there can be multiple accurate models of f^*, however since Shapley values are implementation-invariant (P5), this does not matter.

2.4 Theoretical results: When are Shapley values unique?

In this section we present our theoretical results and their proofs on the uniqueness of Shapley explanations for logical formulas. We prove that they are unique only for two dimensional formulas. Additionally, we find sufficient conditions for the explanation-equivalency of two formulas.

Proposition 1 (Preservation of information for $d = 2$)**.** *Given Assumptions 1-4, the Shapley explanation ψ is injective on the domain of two-dimensional examples and formulas $D_2 = \{0,1\}^2 \times \{f | f : \{0,1\}^2 \to \{0,1\}\}$.*

Proof. For injectivity of ψ, we have to show

$$\forall (x_1, f_1), (x_2, f_2) \in D_2 : \psi(x_1, f_1) = \psi(x_2, f_2) \implies f_1 = f_2.$$

Since $\psi(x, f) = (x, f(x), \psi(f, x))$ includes both the example x and the prediction $f(x)$, it suffices to show that $\forall x, f_1, f_2 : f_1(x) = f_2(x) \wedge \phi(f_1, x) = \phi(f_2, x) \implies f_1(x) = f_2(x)$, that is on any example, formulas with the same prediction have unique Shapley attributions. We show this simply by enumeration. For a visual summary see Figure 2.1, a full calculation is given in the Supplementary Code.

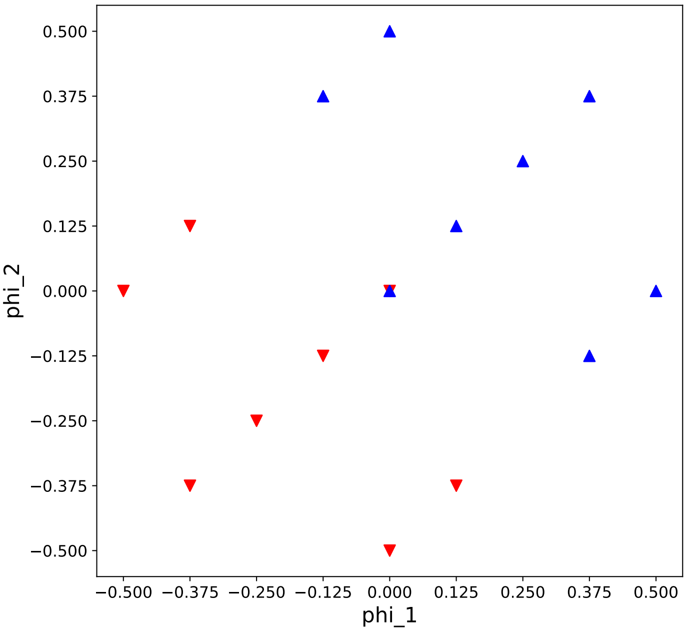

Figure 2.1: Shapley attributions ϕ of all 16 2-dimensional logical formulas (markers) given any example x. Predictions/outputs of formulas are encoded via blue upward triangles (1) and red downward triangles (0). Note that only at $\phi = (0, 0)$ there are two candidate formulas, that however disagree in their predictions. As this pattern holds for any of the 4 examples, for brevity, we show only one here.

Thus, the Shapley explanation ψ is injective for two-dimensional formulas and examples.

□

Proposition 2 (No preservation of information for $d > 2$ in general)**.** *Given assumptions 1-4, the Shapley explanation ψ is not injective on the domain $D_d = \{0, 1\}^d \times \{f | f : \{0, 1\}^d \to \{0, 1\}\}$ for $d > 2$.*

Proof. We first show that the given proposition is true for $d = 3$ and then use the sensitivity property of Shapley values (P2) to prove that it also holds in

higher dimensions.
Let $d = 3$. For non-injectivity, we have to show that

$$\exists (x, f_1), (x, f_2) \in D_3 : \psi(x, f_1) = \psi(x, f_2) \wedge f_1 \neq f_2.$$

Consider $f_1(x) = (x_1 \text{ xor } x_2 \text{ xor } x_3)$ and $f_2(x) = (x_1 \wedge x_2) \vee (x_1 \wedge x_3) \vee (x_2 \wedge x_3)$ on the example $x = (0, 0, 0)$. Obviously, their predictions coincide: $f_1(x) = 0 = f_2(x)$. To calculate their Shapley attributions, we first derive the values of the baseline function f_T on each coalition T:

T	f_T	${f_1}_T(x)$	${f_2}_T(x)$
$\emptyset$	$\mathbb{E}[f(X)]$	$\frac{1}{2}$	$\frac{1}{2}$
$\{1\}$	$\mathbb{E}[f(0, X_2, X_3)]$	$\frac{1}{2}$	$\frac{1}{4}$
$\{2\}$	$\mathbb{E}[f(X_1, 0, X_3)]$	$\frac{1}{2}$	$\frac{1}{4}$
$\{3\}$	$\mathbb{E}[f(X_1, X_2, 0)]$	$\frac{1}{2}$	$\frac{1}{4}$
$\{1, 2\}$	$\mathbb{E}[f(0, 0, X_3)]$	$\frac{1}{2}$	0
$\{1, 3\}$	$\mathbb{E}[f(0, X_2, 0)]$	$\frac{1}{2}$	0
$\{2, 3\}$	$\mathbb{E}[f(X_1, 0, 0)]$	$\frac{1}{2}$	0
$\{1, 2, 3\}$	$\mathbb{E}[f(0, 0, 0)]$	0	0

With that, calculating Shapley values gives for any i (due to the symmetry of features):

$$\begin{aligned}
\phi_i(f_1, x) &= \sum_{T \subseteq U \setminus \{i\}} \frac{|T|!(|U| - |T| - 1)!}{|U|!} (f_{T \cup \{i\}}(x) - f_T(x)) \\
&= \frac{1}{3}(\frac{1}{2} - \frac{1}{2}) + \frac{1}{6}(\frac{1}{2} - \frac{1}{2}) + \frac{1}{6}(\frac{1}{2} - \frac{1}{2}) + \frac{1}{3}(0 - \frac{1}{2}) = -\frac{1}{6}, \\
\phi_i(f_2, x) &= \frac{1}{3}(\frac{1}{4} - \frac{1}{2}) + \frac{1}{6}(0 - \frac{1}{4}) + \frac{1}{6}(0 - \frac{1}{2}) + \frac{1}{3}(0 - 0) = -\frac{1}{6}
\end{aligned}$$

Thus, as there exist distinct formulas with the same prediction and Shapley explanation on the same example, the mapping is not injective for $d = 3$.
Since Shapley values satisfy sensitivity (P2), we can add irrelevant features to f_1, f_2 preserving their equivalent predictions and Shapley attributions. Therefore, the Shapley explanation ψ is not invective for $d > 3$. □

Proposition 3 (Sufficient conditions for explanation-equivalence of formulas). *Given Assumptions 1-4 and an example x, if two formulas f_1, f_2 have*

i) only relevant features U_r that are equally relevant: $\forall T, T' \subset U_r : |T| = |T'| \implies f_T(x) = f_{T'}(x)$,

ii) equivalent predictions: $f_1(x) = f_2(x)$, *iii) equivalent expectations:* $\mathbb{E}[f_1(X)] = \mathbb{E}[f_2(X)]$,

then their Shapley attributions are equivalent and explicitly given by

$$\forall i : \phi_i(f_1, x) = \phi_i(f_2, x) = \begin{cases} \frac{f_1(x) - \mathbb{E}[f_1(X)]}{|U_r|} & , i \in U_r \\ 0 & , \textit{else} \end{cases}$$

.

And by that, also their Shapley explanations are equivalent: $\psi(f_1, x) = \psi(f_2, x)$.

Proof. We prove the given proposition by first assuming only relevant features ($U = U_r$), and then show that $U \setminus U_r \neq \emptyset$ does not change the result.
Let $U = U_r$. *i*) and *ii*) and the completeness property (P1) of Shapley values imply:

$$\sum_{i=1}^{d} \phi_i(f_1, x) = f_1(x) - \mathbb{E}[f_1(X)] = f_2(x) - \mathbb{E}[f_2(X)] = \sum_{i=1}^{d} \phi_i(f_2, x)$$

i) implies that all features have the same Shapley values, since any f_T may be dependent on the number of features but not on which features are included. Thus the Shapley values are:

$$\forall i : \phi_i(f_1, x) = \phi_i(f_2, x) = \frac{f_1(x) - \mathbb{E}[f_1(X)]}{|U_r|}$$

If $U \setminus U_r \neq \emptyset$, irrelevant features have zero attributions due to the sensitivity (P2) of Shapley values. □

Note, that these sufficient conditions for explanation-equivalence of formulas are not necessary. We find that there can be more intricate combinatorial relations yielding equivalency. However, until now we did not formalize or proof any of them.

Lastly, we want to remark that Proposition 1 does not hold if we drop the assumptions of independent, equally weighted features. Consider for instance $f_1(x) = x_1$ xor x_2, $f_2(x) = x_1 \wedge x_2$ on the example $x = (0, 0)$ and the distribution

$p(X_1 = x_1, X_2 = x_2)$	X_1, X_2
$1 - 4q$	$(0, 0)$
q	$(0, 1)$
q	$(1, 0)$
$2q$	$(1, 1)$

for some $q \in (0, \frac{1}{4})$. Then $\forall i$ holds:

$$\phi_i(f_1, x) = \frac{1}{2}(0 - \frac{q}{1-3q} + \frac{q}{1-3q} - 2q) = -\frac{1}{2}q,$$
$$\phi_i(f_2, x) = \frac{1}{2}(0 - 0 + 0 - 2q) = -\frac{1}{2}q,$$

which means that f_1, f_2 have equivalent explanations.

2.5 Simulations: Trends in higher dimensions

To gain additional quantitative insights into Shapley values for logical formulas in higher dimensions, we conduct several exploratory simulations. For these, we examine Shapley explanations for formulas in dimensions $d \in \{2, 3, 4, 5\}$ and investigate the number of unique explanations as well as the average number of explanations needed to identify a formula.

2.5.1 Data generation

For $d \in \{2, 3, 4\}$ an exhaustive generation of all formula (2^{2^d}) and their Shapley values on all examples (2^d) is computational tractable and straight-forward. However, with $d = 5$ and $2^{2^5} = 2^{32}$ formulas we reach our computational limits. In this setting, we exploit the following (conjectured) symmetries of Shapley values on logical formulas:

- prediction-symmetry: Since Shapley values are linear (P3), we have $\forall f, x : \phi(f, x) = -\phi(1 - f, x)$ and thus:

  $$\phi(f_1, x) = \phi(f_2, x) \implies \phi(1 - f_1, x) = \phi(1 - f_2, x)$$

  This means, it is sufficient to consider only formulas with the same prediction, for instance $f(x) = 0$, since results are symmetric.

- example-symmetry: Based on simulations for smaller d we strongly conjecture, that the explanations on different examples are permutations of each other (however a full proof is currently missing):

  $$\forall f, x, x' \exists f' : \phi(f, x) = \phi(f', x')$$

  This means, it is likely sufficient to consider only explanations on one example, for instance $x = (0, 0, 0, 0, 0)$.

This reduces the computational load to 2^{31} formulas and their explanations on one example, which is tractable.

2.5.2 Proportion of unique explanations

In the main evaluation, we compute the proportion of unique Shapley explanations p_{u} of logical formulas with dimension $d \in \{2, 3, 4, 5\}$. This computation requires $\mathcal{O}(n^2), n = 2^{2^d}$ comparisons in the worst case, which is tractable only for $d \in \{2, 3, 4\}$. For $d = 5$, we use an non-exhaustive approach based on a subset of $\frac{1}{16}$ of all formulas. This gives us only a rough but nevertheless valuable estimate (based on the comparison with a non-exhaustive evaluation for $d = 4$ we expect the true number for $d = 5$ to be likely smaller than this estimate).

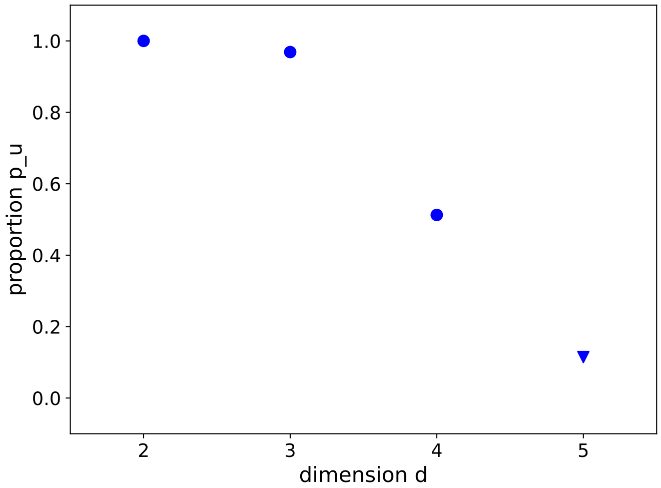

Figure 2.2: Proportion of unique Shapley explanations p_{u} on logical formulas in dimension d. Note that points for $d \in \{2, 3, 4\}$ (circles) are based on exhaustive calculations, whereas the point for $d = 5$ (triangle) is estimated via a subset of proportion $\frac{1}{16}$.

In Figure 2.2 we clearly see a trend of an decreasing proportion of unique explanations with increasing d, which leads to the following conjecture.

Conjecture 1 (Convergence of p_u to zero.)**.** *The proportion of formulas in dimension d with unique Shapley explanations $p_u(d)$ converges to 0 as $d \to \infty$.*

2.5.3 Average number of explanations to identify a formula

An additional interesting question, is how many explanations k_{id} are on average needed to uniquely identify a formula (among all others). Note, that

this number is only loosely related to the number k_{eq} of distinct formulas a formula has equivalent explanations to on average, as this neglects on *which* examples formulas are identical.
We estimate $\bar{k}_{\text{id}}$ and for comparison also $\bar{k}_{\text{eq}}$ for $d \in \{2, 3, 4\}$ via a sample of $N = 1000$ formulas each evaluated on a random permutation of examples:

d	$\bar{k}_{\text{id}}$	$\bar{k}_{\text{eq}}$
3	1	1
3	1.023	1.021
4	1.563	1.814

For both, we see a trend of increasing average numbers for increasing dimension, indicating a diminishing preservation of information about the global formula. Note that each explanation contains 2^d evaluations of f_T which balances with the number of possible formulas in d.

2.6 Conclusions

The main goal of the first part was to investigate what information about a global logical formula is preserved by a local Shapley explanation.

Assuming independent, equally weighted features we prove that for $d = 2$ one local explanation uniquely determines the global formula, all information is preserved. This implies that an explanation on one example also lets us derive the global behavior of the model on all other inputs, thus providing a global explanation of the model. Such a local-global translation is obvious for linear functions for many attribution methods, however for non-linear functions like logical formulas it is a new result specific to Shapley values. The preservation of information however does not hold for $d > 2$ or if we drop the assumption of independent, equally weighted features. In addition, we found some sufficient conditions for the explanation-equivalence of formulas on one example.

Our simulations demonstrate that with increasing dimension the proportion of formulas identifiable by one local explanation decreases and seems to converge to 0; and the average number of explanations needed to identify a formula increases. A formal proof for this first convergence is missing until now, and would be a desirable goal of further theoretical investigations. With these result, we give first theoretical insights into Shapley values on the non-linear model class of logical formulas under nice assumptions. Since they are based on explicit calculations of Shapley values, these also bound the quality of their estimations like Kernel SHAP (Lundberg and Lee, 2017).

Loosening the assumption of independent features opens up the question of how to treat features missing in a coalition, that is which baseline function to use. Especially, the case of strongly correlated features seems interesting (cf. Janzing et al., 2020): Given two perfectly correlated features, and a model f only relying on the first one, using the conditional expectation as baseline function would also give the second feature non-zero attribution; whereas the marginal expectation would not. In the first case we lose sensitivity for f, which is problematic from a model debugging perspective. However, also the second choice has an arguably unsatisfying consequence: Even though the second feature could predict the output, its attribution given this single model indicate that it is not important – this single attribution is not representative.

We recognized that this holds not only for Shapley values but feature attribution methods in general. The idea emerged, that averaging over the attributions of many trained models could be a simple way to confront this problem. Due to its potentially more general application and constrained resources, we explored this idea thoroughly and exclusively in the second part of this master thesis.

Chapter 3

Improving Feature Attribution Methods by Averaging over Models

3.1 Motivation

Consider the very simple learning problem of two perfectly correlated binary features X_1, X_2 which also determine the binary label Y:

$$p(x_1, x_2, y) = \begin{cases} 0.5 & , \ x_1 = x_2 = y \\ 0 & , \ \text{else} \end{cases} .$$

Since both features are equally predictive of the label, one reasonable intuition would be that a local explanation of why the example $x' = (1, 1)$ has label $y = 1$ should attribute both features equally. However, this symmetry does not necessarily hold, since there is a multitude of accurate models and the applied learning algorithm might select between some of them, based on its initialisation, randomly. For example, a single neural net might select the model $f_1(x) = x_1$ and a feature attribution method faithful to this model could yield $\text{attr}(f_1, x') = (1, 0)$. From a model debugging perspective this explanation is useful: we get the information that our model ignores the second feature. However, from the perspective to understand the importance of features using a certain learning algorithm it is not representative: a trained accurate model could have been easily different, for instance $f_2(x) = x_2$ or $f_3(x) = 0.1\ x_1 + 0.9\ x_2$, and hence its feature attribution. The average of multiple models' attributions would provide a more representative explanation: if the algorithm has an input symmetry it would converge towards an equal attribution of both features, or otherwise reflect its asymmetry accurately.

Of course, the above example is quite artificial. However, with growing dimensionality of data, more nuanced but qualitative similar scenarios may naturally arise. Especially for natural image data it is a common assumption that there are many redundant features available, both with respect to pixels as well as on higher levels, to predict its category. For instance, a model may recognize a duck in an image by its beak *or* by its feet *or* by the texture of its plumage; additionally in these cases neighboring pixels may be highly correlated and not all of them are needed for an accurate prediction. Moreover, image recognition is one of the sub-fields of machine learning, where black box models such as deep neural networks are still the state of the art, comparable ad-hoc interpretable models are often missing and which therefore strongly relies on post-hoc feature attribution methods.
Based on this motivation, we investigate whether averaging attributions over multiple accurate models can improve widely used feature attribution methods on images in the second part of this master thesis. For this purpose, we conduct experiments on

- a simple geometric data set, that includes redundant features by design,
- more realistic natural image data sets

and evaluate them visually and quantitatively via two localization metrics.

3.2 Related work

In this section we cover prior work relevant for our approach and formally introduce feature attribution methods we investigate.

On image recognition problems, where adequate ad-hoc interpretable models are often unavailable, feature attribution methods are the most relevant class of explanation methods. Formally, they map a trained model, usually some deep neural network, $f : \mathbb{R}^d \to \mathbb{R}^c$, where c is the number of classes, and an example image $x \in \mathbb{R}^d$, to an equally sized attribution vector (sometimes also called saliency map) of the prediction $f(x)$:

$$\mathrm{attr}(f, x) : \mathcal{F} \times \mathbb{R}^d \to \mathbb{R}^d.$$

In recent years, a vast amount of approaches how to attribute input features have been explored. We do not aim at covering them exhaustively here, but introduce some widely used, that we later apply:

- **Gradient** (e.g. Baehrens et al., 2010), that is $\mathrm{g}(f, x) = \frac{\partial f(x)}{\partial x}$ quantifies how much a small change in each input dimension of x would change the prediction $f(x)$.
- **Gradient x Input** (Shrikumar et al., 2016) takes the dot-wise product of gradient and input: $\mathrm{gi}(f, x) = \frac{\partial f(x)}{\partial x} \odot x$.
- **Guided BackPropagation** (Springenberg et al., 2014) propagates the gradient back to input features with negative gradient entries set to zero at ReLU units.
- **Integrated Gradients** (Sundararajan et al., 2017) sums gradients of scaled versions of the input image with respect to some baseline image, and is given by: $\mathrm{ig}(f, x) = (x - x') \int_0^1 \frac{\partial f(x_0 + \alpha(x - x'))}{\partial x} d\alpha$.
- **Gradient SHAP** (Lundberg and Lee, 2017; Erion et al., 2021) computes the expectation of integrated gradients over a distribution of baselines D to approximate Shapley values: $\mathrm{gs}(f, x) = \mathbb{E}_{x' \sim D}\ \mathrm{ig}_{x'}(f, x)$.

A very general improvement of feature attribution methods was introduced with the SmoothGrad method (Smilkov et al., 2017). By averaging explanations based on images sampled with noise around a target image, it de-noises and visually improves the explanations of attribution methods relying on gradients:

$$\mathrm{sg}(f, x) = \frac{1}{N} \sum_{i=1}^{N} \mathrm{attr}(x + n_i), n_i \sim \mathcal{N}(0, \sigma^2).$$

Importantly, it reduces noise within the evaluation of a single model, whereas we aim at reducing randomness external to models.

With the development of feature attribution methods on images also began their critical assessment, including some theoretical investigations (Nie et al., 2018; Sixt et al., 2020) but mostly empirical work (Samek et al., 2016; Adebayo et al., 2018; Kindermans et al., 2019) often concluding an unreliability of these methods with respect to certain desirable properties. We do not argue against these assessments in this work, rather, we aim at making explanations of attribution methods more stable and representative, providing a better basis for such assessments.

Lastly, the observation that many high-dimensional learning problems allow for many good models at least goes back to Breiman (2001) and is since then known as Rashomon effect. In prediction, this observation is addressed for example by ensemble methods like bagging (Breiman, 1996), where the outputs of multiple models are averaged. However, in case of explanations this phenomenon has – to our knowledge – been neglected until now.

3.3 Methods

In this section, we describe the methods we use to investigate averaging feature attributions, namely a formal definition of the averaging procedure and localization metrics.

3.3.1 Averaging

Given some learning algorithm $\mathcal{A}$ that selects models $f_i, i \in \{1, ...N\}$ based on a fixed training set $(X_j, Y_j)_{j=1,...,m}$ (or alternatively i.i.d. sets) and an attribution method attr, we define the average attribution on example x as

$$\text{avg}(x) := \frac{1}{N}\sum_{i=1}^{N} \text{attr}(f_i, x), f_i \sim \mathcal{A}.$$

Note, that the learning algorithm can be defined very flexibly. For example it can include the training scheme of a single model-architecture given a set of hyper-parameters, but is also able to cover different model-architectures applied on the same problem. The main idea is that it includes some source of randomness in the selection of models, usually through random initialization or random batching of training data.

3.3.2 Localization metrics

For the assessment of feature attributions with a given ground truth, we build on well-known localization metrics. Given a ground truth set of pixel-indices $L \subseteq \{1, ..., d\}$ indicating the position of an object, feature attributions $a = \text{attr}(f, x) \in \mathbb{R}^d$ and an inferred localization from attributions $\hat{L}_\alpha = \{i \in \{1, ..., d\} | a_i > \alpha\}$, with some threshold α, then

- **Intersection over Union** (e.g. Rezatofighi et al., 2019) is defined as

$$\text{IoU}_\alpha(L, \hat{L}_\alpha) = \frac{|L \cap \hat{L}_\alpha|}{|L \cup \hat{L}_\alpha|} \in [0, 1].$$

- **Area under the Precision Recall Curve** (e.g. Davis and Goadrich, 2006) first generates a Precision-Recall-Curve by varying α and then computes its integral:

$$\mathrm{AUPRC}(L, a) = \int_0^1 p_{\alpha|r=r_\alpha} dr \ \in [\frac{|L|}{d}, 1]$$

with precision $p_\alpha(L, \hat{L}_\alpha) = \frac{|L \cap \hat{L}_\alpha|}{|\hat{L}_\alpha|}$ and recall $r_\alpha(L, \hat{L}_\alpha) = \frac{|L \cap \hat{L}_\alpha|}{|L|}$.

Both of these metrics measure some notion of detection accuracy, and are especially appropriate when labels are not balanced, that is $\frac{|L|}{d} \neq 0.5$.

3.4 Experiments

In this section we present our empirical experiments to evaluate averaging feature attributions over multiple trained models for different methods. In the first experiment, we use a simple artificial data set, that includes redundant features by design, as a proof of concept. In subsequent experiments, we use more realistic, natural image data to further investigate the effects of averaging over models in realistic settings.

For all these experiments there is a multitude of reasonable and interesting (parameter) choices. Here, we only present in detail the explored choices, we found most insightful. All experiments are implemented using `pytorch` (Paszke et al., 2019) and the `captum` interpretability library (Kokhlikyan et al., 2020).

3.4.1 Artificial geometric data

Experiment 1 (Rectangles). The goal of the first experiment is to test whether averaging feature attributions over many models improves explanations, in a simple geometric setting which contains by design many redundant features, and hence allows for many different accurate models.

Data set. Our data set consists of 16x16-pixels black and white images, displaying 0 to 4 4x4-pixels dark rectangles at 4 fixed positions overlayed with smoothed uniform noise. Any of the $2^4 = 16$ rectangle combinations has, in the standard setting, $n_{\text{train}} = 100$ repetitions in the training set and $n_{\text{test}} = 10$ repetitions in the test set. The image label is, in general, determined by a two-dimensional logical formula of the presence (1) or absence (0) of the 2 upper rectangles. Here, we consider specifically the *or*-formula. Therefore a single dark pixel of any of the 2 upper rectangles would theoretically allow to accurately predict the label.

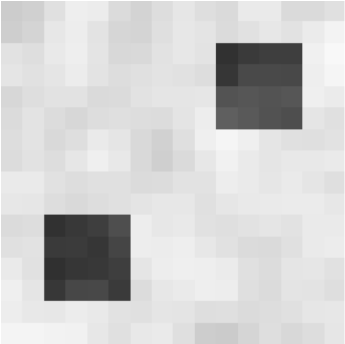

Figure 3.1: Example input for Experiment 1. The depicted example encodes $x = (0, 1, 1, 0)$ and therefore has label $y = x_1 \vee x_2 = 1$.

Model architecture. On this relatively simple task, we choose a fully connected neural network with three hidden layers of size 100 and ReLU activation functions.

Training and explaining models. We train $N = 25$ models with default random initialisation and randomly batched data sets until generalization (zero test error) using the Adam-optimizer (Kingma and Ba, 2014). We explain models on all 16 rectangle-combinations with the feature attribution methods: Gradient, Gradient x Input, Guided Backpropagation, Integrated Gradients and Gradient SHAP. For Integrated Gradient we use the zero-image as baseline and for Gradient SHAP the average image of the training sample. The motivation of the latter choice is to reduce all internal sources of noise of attribution methods to see the effect of averaging over models more clearly, and still use a representative data point for the distribution. The rationale to use identical training sets is similar, guaranteeing that variability of models is not due to variability between the data sets.

Results. For all methods, we observe that averaging attributions over models visually decreases noise (Figure 3.2). This is especially pronounced for Gradient and Guided Backpropagation. For Gradient x Input, Integrated Gradients, and Gradient SHAP, attributions within the rectangles are less de-noised and more strongly influenced by the original input (whether this is desirable is debatable), however the rest of the image is clearly less noisy. Importantly, all methods attribute the only the upper rectangles strongly and the rest of the image weakly. The lower (irrelevant) rectangles are very slightly accentuated with increasing N, which might be due to their structural difference relative to their surrounding and how this is processed in training. Although only a

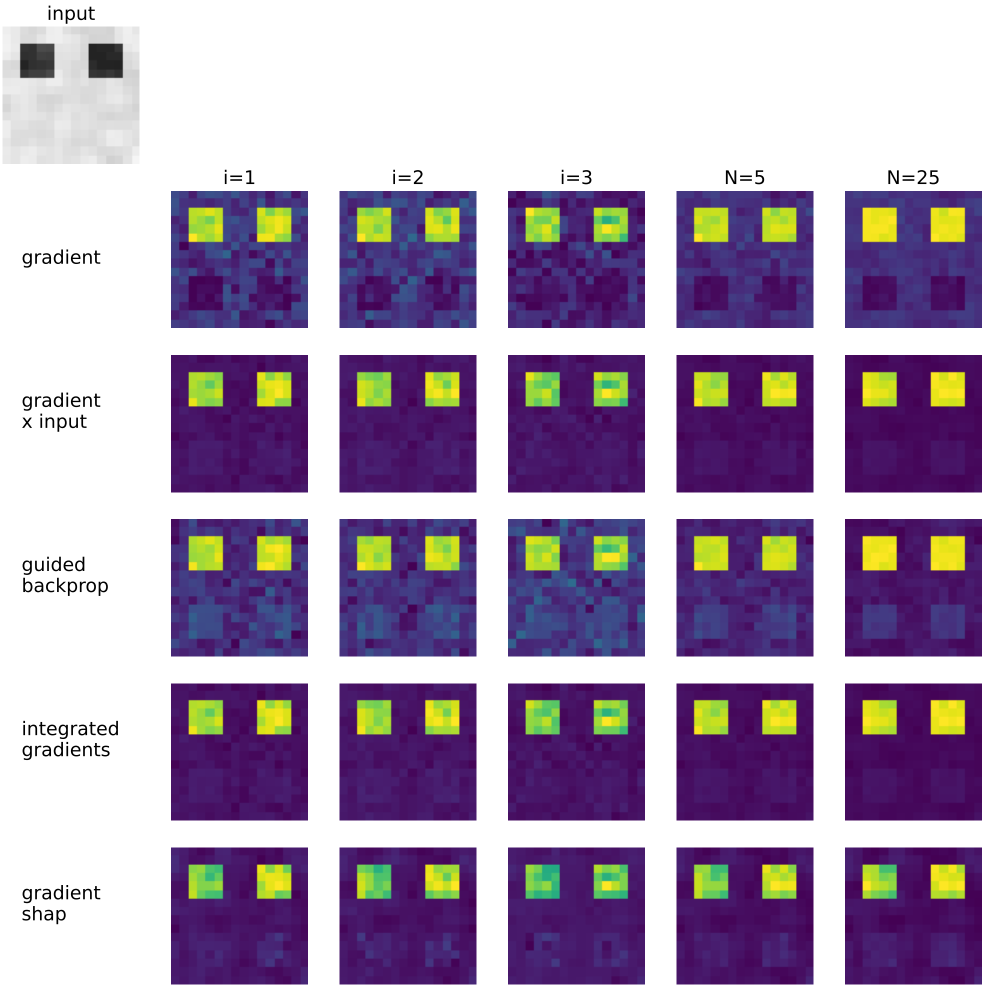

Figure 3.2: Results of Experiment 1. Attributions on one example input using different attribution methods (rows) of single models i or averaged over N trained models (columns). Attributions are normalized per image. Bright, yellow shading indicates high values.

minor observation, it hints at the potential to discover biases of a learning algorithm in general by providing more representative attributions. Note, that methods do not necessarily agree on their attribution given an input, this is however not the main focus of this current evaluation.

Alternative parameter choices. In addition to the standard setting above, we explore alternative choices: train models 3 times longer, use a larger training set of tenfold size, or generate i.i.d. training sets for single models. Across these choices the general pattern remains the same as in the standard setting. For details see the Supplementary.

In conclusion, with Experiment 1 we deliver a first proof of concept for the benefits of averaging attributions over models.

3.4.2 Natural image data

In the second part of experiments, we test the effects of averaging attributions over models in more realistic image scenarios.

Experiment 2 (Ducks in ImageNet).
Data set. Our data set consists of colour images of 10 ImageNet2012 (Russakovsky et al., 2015) categories either containing or not containing a pasted fixed image of a duck at random position and with random rotation and size. The image label is determined by the presence (1) or absence (0) of the duck. In total, the training and test set consist of 12.852 and 500 images with balanced labels. Although these 'collage' images lack certain realistic properties, such as coherent shading and proportions, they are a first step towards full-realistic images and comparable to other approaches in the literature (Yang and Kim, 2019). Importantly, by assuming that attributions should only lie within the area covered by the duck, this data set enables us to compare attributions to a ground truth.

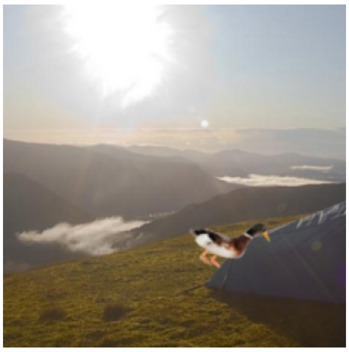

Figure 3.3: Example data point for Experiment 2.

Model architecture. We find a ResNet18 architecture (He et al., 2016) appropriate for this learning problem, with the last layer adapted to a binary classification problem.

Training and explaining models. We train 100 models using random initialisation on randomly batched versions of this data set until generalization (test error < 0.005) with Adam. We generate feature attributions for all models on 25 validation images with the methods: Gradient, Gradient x Input, Guided Backprop, Integrated Gradients and Gradient Shap. For Integrated Gradients we used the zero/black-image as baseline and SmoothGrad with

$n = 5$ samples and variance $\sigma = 1$. For SHAP we use a set of 5 images with uniformly drawn pixel-values as baselines, and SmoothGrad with $n = 50$ samples and variance $\sigma = 0.1$. The use of SmoothGrad in this experiment is motivated to distinguish between noise removed within models in contrast to variability removed between models. By relying on identical training sets we exclude another source of randomness, and provide thereby a stronger test. Additionally, in many real world cases i.i.d. data is costly or not available.

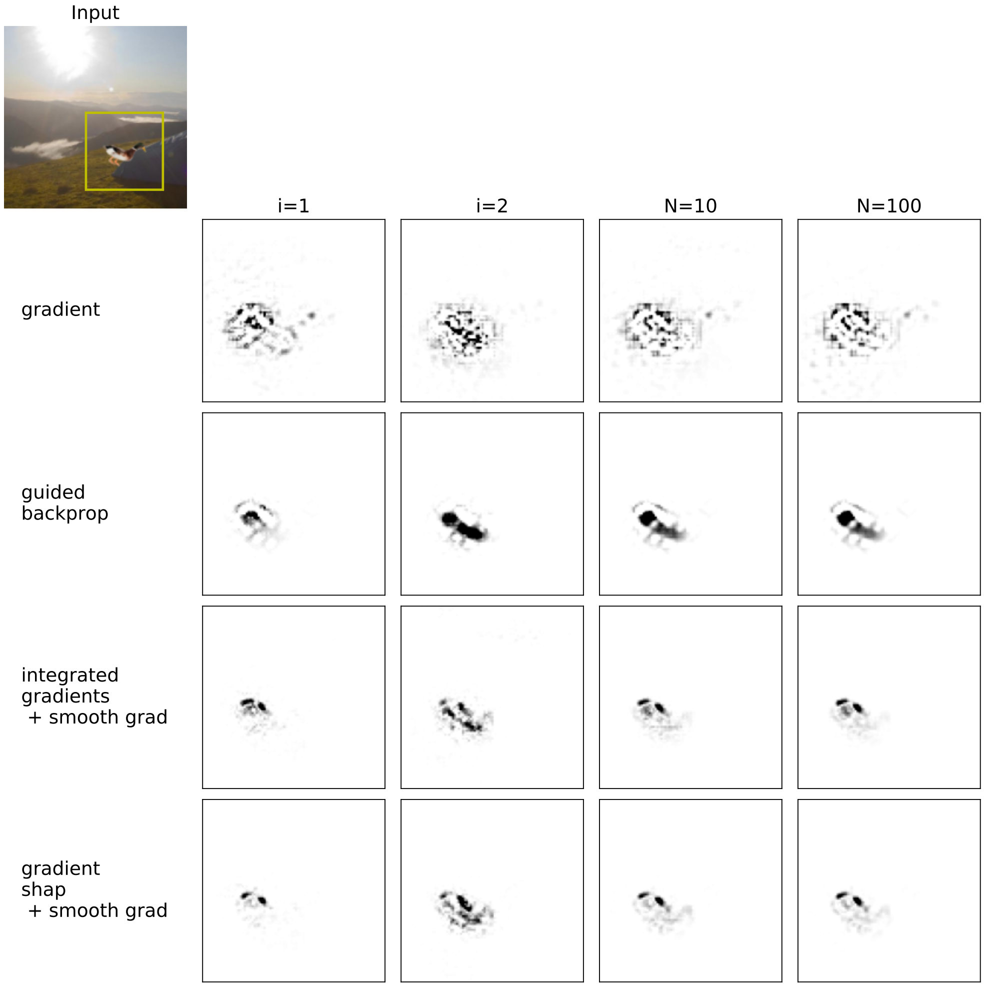

Figure 3.4: Results of Experiment 1: Positive attributions on on example input using different attribution methods (rows) for single trained models i or averaged over N models (columns). For better visualization, only attributions within the yellow square are depicted, attributions are clipped at 50% of the maximal attribution value and are normalized per image. Dark shading indicates high attribution.

Quantitative evaluation. To evaluate the attributions quantitatively, we compute localization metrics Intersection over Union (IoU) and Area under the Precision-Recall-Curve (AUPRC) averaged over 25 images containing ducks.

For the Intersection over Union, we interpolated over thresholds dependent on the α-percentile of each attribution distribution per image. To estimate metrics for different numbers of averaged models $N \in \{1, 10, 100\}$, we find that the average of $n_1 = 10$ performances of single models and $n_{10} = 3$ performances of 10 models are adequate.

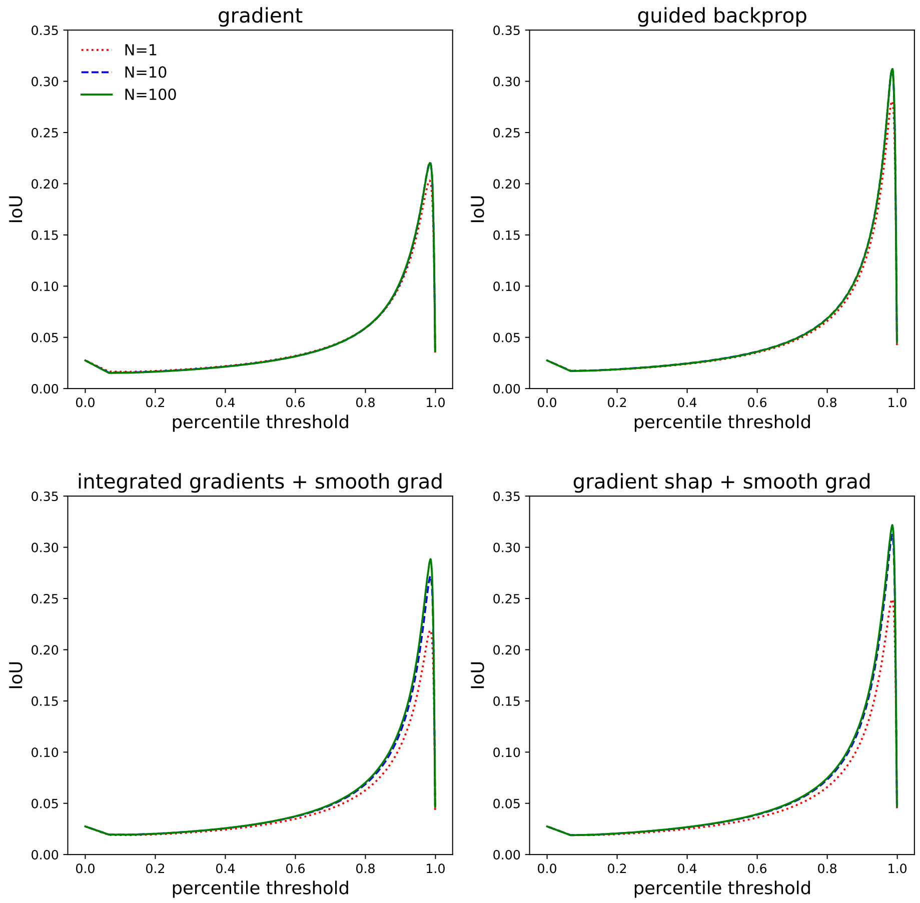

Figure 3.5: Quantitative Results (A) of Experiment 2. Depicted are the Intersection over Union (IoU) curves of varying percentile threshold for different attribution methods and number of averaged models. The threshold determines above which percentile of the attribution distribution, an pixel-attribution indicates presence of a duck. Note that lines for $N = 10$ and $N = 100$ largely overlap.

Results. Visually, we observe that averages over mutliple models are more uniform within ducks and less noisy around it for all methods except Gradients. (Figure 3.4). Quantitatively, averages over multiple models have higher localization metrics IoU (Figure 3.5), AUPRC and maxIoU than single models (Figure 3.6). These improvements are relatively small for Gradients and Guided Backprop, and larger for Integrated Gradients and Gradient Shap. Morevover, these improvement seems to saturate after averaging more than N

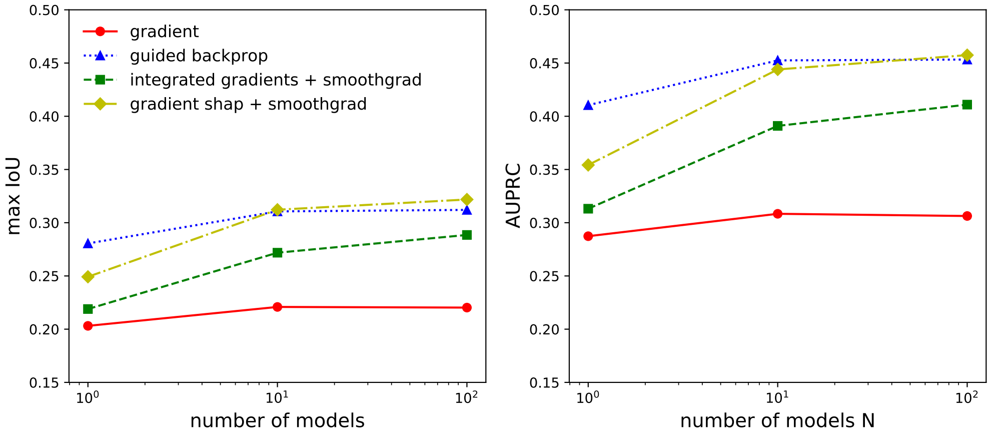

Figure 3.6: Quantitative Results (B) of Experiment 2. Depicted are the maximal average Intersection over Union (left) and average AUPRC values (right) over 25 test images using different attribution methods and number of averaged models N. The max IoU can be interpreted as the IoU with optimized choice of percentile threshold α over all images.

models, in most of the cases. Importantly, the methods Integrated Gradients and Gradient SHAP, which relied on SmoothGrad, were also improved, indicating, that in fact averaging over models decreases another source of noise, one between models. Interestingly, all improved methods attribute a high importance to the duck's torso but zero or low importance to its feet. This could be a matter of size, but also hint at a bias for textures over shapes of the network.

Alternatively, we also explored training individual models on i.i.d. data, which led to slightly larger improvements (for details see the Supplementary).

Experiment 3 (Averaging ImageNet-trained models).

At last, since the data set from Experiment 2 still contains a lot of unrealistic properties, for example incoherent shading and proportions, we try averaging attributions of 10 prominent trained models on ImageNet (provided via pytorch) on some examples from images. Here we used the methods Gradient, Gradient x Input, Guided Backprop and Gradient SHAP with 5 random uniform baselines.

For all methods we see differences between single model attributions and a more uniform de-noised average (Figure 3.7). Despite the lack of quantitative results, this hints at the potential of also averaging models with different architectures, where there is arguably a larger variance between models.

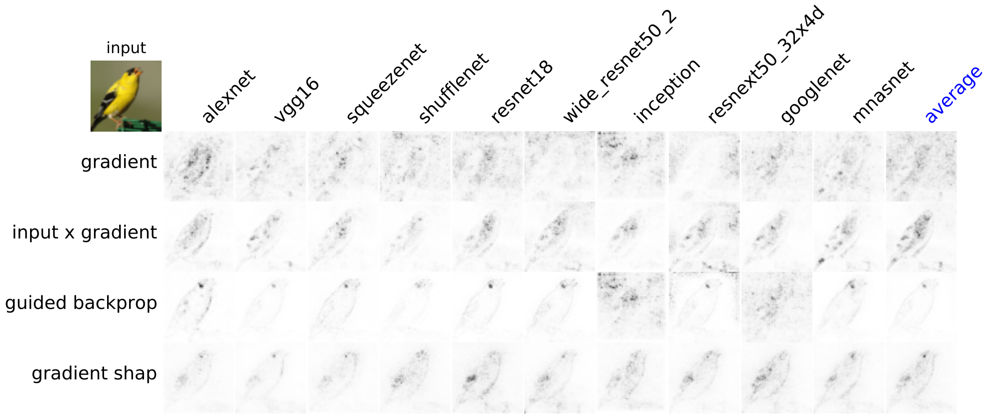

Figure 3.7: Results of Experiment 3. Given the example input, absolute attributions of different models (columns) generated by different methods (rows) are depicted. The last columns shows the average attribution over models. Dark shading indicates high attribution. For visualization, values are clipped at 50 % of the maximal attribution per image.

3.5 Conclusions

The goal of the second part of the master thesis was to investigate the effects of averaging attributions over multiple trained models for different widely used feature attribution methods on images.

In the first experiment, we demonstrated on an artificially redundant data set that averaging can visually de-noise and yield more uniform attributions for the used feature attribution methods. Moreover, in all average explanations some irrelevant features were slightly accentuated, which hints at the potential of averaging to detect inductive biases of the learning algorithm by providing a more representative explanation. Using a fully connected neural network yielded already for single models quite balanced attributions of redundant features. Testing other architectures on this problem would be an interesting follow-up. Especially considering networks designed to compress features could provide single attributions with higher variance that are even more 'unrepresentative' and highlight the benefit of averaging more strongly. For instance Baumgartner et al. (2018) used a larger but similar toy problem, resulting in more extreme single attributions, where averaging over multiple models could also be applied. Nevertheless, with this experiment we gave a proof of concept for the benefits of averaging attributions of multiple models.

In a subsequent experiment, we tested whether averaging can also improve explanations on more realistic image data containing ducks. We find that for many attribution methods average attributions are visually less noisy and more uniform. In addition, we provide numerical evidence that localization metrics based on averaged attributions are increased. Importantly, both visually and numerically the effects of averaging seem to saturate for more then $N = 10$ models. This seems to be a positive result, since (re-)training multiple models can often be costly. However, a more thorough investigation of this cost-benefit trade-off is certainly needed. Again, average attribution provide a more representative explanation of models, showing the bias towards the duck's torso more clearly. Lastly, averaging 10 prominent models trained on ImageNet also yield some visual improvements of attributions.

With these experiments, we provide first empirical evidence that averaging attributions over multiple models can improve explanations for some widely used feature attribution methods on images. By highlighting which features are on average important for an accurate prediction, these average explanations provide more representative information about the behavior of the learning algorithm, and are in themselves more stable and reliable.

Chapter 4

Discussion

In this work we aimed at increasing the theoretical understanding of the explanation class of feature attributions and making them more reliable. We investigated theoretical properties specifically for Shapley values on logical formulas and an averaging method to improve feature attribution methods in general.

In the first part, we asked what information about a global formula in dimension d is carried by a local Shapley explanation. Assuming independent, equally weighted features we find that for $d = 2$ one local Shapley explanations uniquely determines the global formula: all information is preserved. This means that by a local explanation on one example we also get a global explanation of the model. This is a new and interesting result for a non-linear model class. However, with $d > 2$ or loosening the assumption of independence, this does not hold anymore. With additional simulation we show that for increasing d the information about the global formula carried by local explanations decreases. With that we provide first theoretical results for Shapley values, a method based on particularly desirable axioms, on a simple non-linear class of models, which also bound the quality of their empirical estimates. A further theoretical investigation into the functional relationship of information preservation and dimensionality, a more thorough examination under loosened assumptions and comparing these results to image data could be interesting next steps.

The examination of dependent features has led to an observation relevant for feature attribution methods in general: In settings with redundant features that allow for a multiplicity of accurate models relying on different features, a single feature attribution may not be representative (a property that can be meaningful and desirable): A learning algorithm that includes some randomness could have easily picked different features resulting in different attributions. To counter this we propose to average attributions over multiple trained models and test its effects for widely used feature attribution methods first on an artificial geometric data set, which contains redundant features by design, and then on more realistic image data, were redundancies are commonly assumed.

In both cases, we see that averaging can make attribution less noisy and more uniform. Moreover, providing an explanation of the behavior of a learning algorithm on average provides more stable attributions enhances the detection of inductive biases. Importantly, averaging also improves methods using SmoothGrad, which reduces noise within the evaluation of models. This suggests that averaging reduces another source of randomness external to models.

Although the effects of averaging are clearly noticeable with our modeling choices, we think fine-tuning (hyper-)parameters and especially exploring other network architectures, that compress features more heavily and thus provide single models with higher variance, could increase these effects and are fruitful next steps. Additionally, it would be a self-evident next step, to investigate the effects of averaging systematically on real world data sets and problems. Another potential use of average attributions is to view them as estimations of explanations in the limit for a certain method, which provides a more useful basis to compare and critically assess feature attribution methods between each other but also relate tabular to methods on images. For example it would be interesting to see whether Gradient SHAP on images containing logical formulas (via rectangles or real world objects additional to ducks) in the limit can recover the newly found uniqueness properties of Shapley values on tabular logical formulas. Lastly, the average may not be the only reasonable aggregation function. In our settings we ensured that models are equally accurate. However, in scenarios where this is not the case it might be useful to include the model's confidence into the aggregation as is often done in prediction ensembles.

In conclusion, in this work we extend the theoretical understanding of explanation methods by providing interesting first theoretical results for Shapley values on logical formulas, and help making explanation more reliable by giving first empirical evidence that averaging attributions of multiple accurately trained models can improve widely used feature attribution methods on images.

References

Aas, K., Jullum, M., and Løland, A. (2019). Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *arXiv preprint arXiv:1903.10464*.

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. (2018). Sanity checks for saliency maps. *arXiv preprint arXiv:1810.03292*.

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., and Müller, K.-R. (2010). How to explain individual classification decisions. *The Journal of Machine Learning Research*, 11:1803–1831.

Baumgartner, C. F., Koch, L. M., Tezcan, K. C., Ang, J. X., and Konukoglu, E. (2018). Visual feature attribution using wasserstein gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8309–8319.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231.

Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240.

Erion, G., Janizek, J. D., Sturmfels, P., Lundberg, S. M., and Lee, S.-I. (2021). Improving performance of deep learning models with axiomatic attribution priors and expected gradients. *Nature Machine Intelligence*, pages 1–12.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Janzing, D., Minorics, L., and Blöbaum, P. (2020). Feature relevance quantification in explainable ai: A causal problem. In *International Conference on Artificial Intelligence and Statistics*, pages 2907–2916. PMLR.

Kindermans, P.-J., Hooker, S., Adebayo, J., Alber, M., Schütt, K. T., Dähne, S., Erhan, D., and Kim, B. (2019). The (un) reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 267–280. Springer.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., et al. (2020). Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pages 4768–4777.

Nie, W., Zhang, Y., and Patel, A. (2018). A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. In *International Conference on Machine Learning*, pages 3809–3818. PMLR.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 658–666.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., and Müller, K.-R. (2016). Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673.

Shapley, L. S. (1953). *17. A value for n-person games.* Princeton University Press.

Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. (2016). Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713.*

Sixt, L., Granz, M., and Landgraf, T. (2020). When explanations lie: Why many modified bp attributions fail. In *International Conference on Machine Learning*, pages 9046–9057. PMLR.

Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. (2017). Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825.*

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806.*

Strumbelj, E. and Kononenko, I. (2010). An efficient explanation of individual classifications using game theory. *The Journal of Machine Learning Research*, 11:1–18.

Štrumbelj, E. and Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665.

Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR.

Yang, M. and Kim, B. (2019). Benchmarking attribution methods with relative feature importance. *arXiv preprint arXiv:1907.09701.*