

# Evaluation of Interpretable Machine Learning

**Manuel Heurich**

Matriculation no.: 5176607

Freie Universität Berlin

Department of Mathematics and Computer Science

Dahlem Center for Machine Learning and Robotics - Biorobotics Lab

This thesis is submitted for the degree of

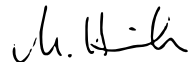
*Master of Science*

Supervisor:	Prof. Dr. Tim Landgraf Leon Sixt
Reviewer:	Prof. Dr. Dr. (h.c.) habil. Raúl Rojas
Started:	September 14, 2020
Finished:	February 22, 2021



## **Declaration**

Hiermit versichere ich, dass die vorliegende Arbeit von mir selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt wurde. Die vorliegende Arbeit ist frei von Plagiaten. Alle Ausführungen, die wörtlich oder inhaltlich aus anderen Schriften entnommen sind, habe ich als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen und wurde bisher nicht veröffentlicht.



Manuel Heurich  
February 22, 2021





## Abstract

Deep neural networks are often considered *Black Boxes* due to their complex structure and their high number on non-linear computations. Even though they might perform better as humans for a specific task, their lack of transparency causes distrust and prevents them from being used in a much broader field of applications. Interpretability increasing algorithms aim to provide insight into a model and its decision-making. Various methods appeared in recent years, but do they really work or do they misinterpret parameters of the model? The difficulty of evaluating interpretability is the lack of ground truth. This work provides two evaluation settings in which we test different attribution methods. Both define a reasonable ground truth due to the appropriate choice of model and data. First, we propose to evaluate the model on the simplest non-linear problem - XOR. This allows us to understand both model and data, deriving a ground truth based on this knowledge. Additionally, we employ an invertible neural network which allows us to inspect internals more easily. The results confirm a promising way of evaluating attribution methods and establish a reliable evaluation framework based on a derived ground truth.



# Table of contents

<b>List of figures</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Contribution . . . . .	3
<b>2 Related Work</b>	<b>5</b>
2.1 Interpretability . . . . .	5
2.2 Discussion of Attribution Methods . . . . .	6
2.2.1 Saliency Maps / Gradients . . . . .	7
2.2.2 Input $\times$ Gradients . . . . .	8
2.2.3 Smoothgrad . . . . .	8
2.2.4 Integrated Gradients . . . . .	9
2.2.5 Information Bottleneck . . . . .	10
2.3 Evaluating Attribution . . . . .	12
<b>3 Two-Dimensional Evaluation</b>	<b>15</b>
3.1 Visual Evaluation . . . . .	17
3.2 Quantification . . . . .	20
3.2.1 Dimensional Difference Quantification . . . . .	20
3.2.2 Correlation-based Quantification . . . . .	21
3.3 Summary . . . . .	24
<b>4 Inversion Evaluation</b>	<b>25</b>
4.1 Setup . . . . .	25
4.2 Evaluation of Integrated Gradients . . . . .	28
4.2.1 Gradient Independency . . . . .	29
4.2.2 Baseline Impact . . . . .	29
4.3 IBA Evaluation . . . . .	33

4.3.1	Attribution Results . . . . .	34
4.3.2	Evaluation through Inversion . . . . .	35
4.4	Summary . . . . .	39
<b>5</b>	<b>Conclusion</b>	<b>41</b>
	<b>References</b>	<b>43</b>
	<b>Appendix A Theoretical Background</b>	<b>45</b>
A.1	Backpropagation Algorithm . . . . .	45
A.2	Activation Functions . . . . .	45
A.3	Gaussian Probability Distribution . . . . .	46
A.4	Loss Functions . . . . .	46
A.5	Network Optimization . . . . .	46
	<b>Appendix B Additional Plots</b>	<b>49</b>
B.1	Ring of Gaussian . . . . .	49
B.2	Inversion Evaluation . . . . .	53

# List of figures

1.1	Goldfinch Attribution Example . . . . .	2
2.1	IBA Per-Sample Bottleneck . . . . .	12
3.1	XOR_net Dataset Visualization . . . . .	16
3.2	Attribution Results of XOR_net . . . . .	18
3.3	Attribution Value Differences . . . . .	23
4.1	Inversion Evaluation Ground Truth Facial Region . . . . .	27
4.2	Inversion Evaluation Ground Truth Background . . . . .	28
4.3	Gradient-Pixelvalue Plot . . . . .	30
4.4	Integrated Gradients: Separated Attribution Outputs . . . . .	32
4.5	Integrated Gradients: Ideal Baseline . . . . .	33
4.6	IBA Results for different $\beta$ (Smiling) . . . . .	34
4.7	IBA Results for fixed $\beta$ and different Labels . . . . .	36
4.8	Inverted IBA-noised Samples . . . . .	37
4.9	Pixel-wise Standard Deviation of Inverted IBA batch . . . . .	38
4.10	Pixel-wise Mean of Inverted IBA batch . . . . .	39
B.1	ROG Visualization . . . . .	50
B.2	Attribution Results of ROG_net . . . . .	51
B.3	Attribution Value Differences (ROG_net) . . . . .	52
B.4	Ground Truth Moustache . . . . .	53
B.5	Ground Truth Smile . . . . .	54
B.6	IBA Results for different $\beta$ (Attractiveness) . . . . .	55
B.7	IBA Results for different $\beta$ (Glasses) . . . . .	55
B.8	IBA Results for fixed $\beta$ and different Labels (male) . . . . .	56



# Chapter 1

## Introduction

### 1.1 Motivation

Machine Learning (ML), especially Deep Learning, became increasingly sophisticated in recent years. Human tasks and decision-making can be reinforced with or even replaced by intelligent systems due to powerful hardware and algorithms. Such tasks may include comfort enhancing applications, like autonomous mobility or smart home products, but also system-critical tasks, such as medical procedures. For instance, various implementations of detection algorithms can outperform oncologists on cancer detection in early stages, increasing the room for manoeuvre for medical treatment significantly. That said, for most humans, trust is a significant factor for embracing crucial self-affecting decisions made or influenced by others. For an algorithm to receive the required credibility can be challenging, and to fully exploit its assets, the user's confidence has to be aligned with the model's capabilities.

Modern machine learning algorithms may be famous for their accurate decision-making in real-time but less for how they express their reasoning for a particular prediction. With artificial neural networks being the dominant technique in state-of-the-art machine learning applications, interpretable machine learning as a research area aims at examining the *Black Box*, as the network is often called. Settling on the definition for interpretability in the context of ML as "the degree to which an observer can understand the cause of a decision" [Miller, 2018], its benefits are manifold, such as revealing biases, determining reliability and robustness or examining a model's generalization ability. In the context of regulation and ethics, the asset of interpretability serving as a measuring instrument of algorithmic decision-making becomes essential. For instance, the first paragraph of article 22 of the General Data Protection Regulation (GDPR) [European Parliament and Council of the European Union, 2016] prohibits solely automated decisions without human interference. The arguably main reason for passing this article are liability issues, especially for crucial use cases. This

universal ban can ultimately be reduced to the lack of trust in intelligent systems. Neither taking responsibility for obscured decision-making nor a change in the legal state will help integrate sophisticated ML algorithms in the foreseeable future. Interpretable models might.

Several attribution methods have been introduced to tackle the lack of interpretability. Each method provides different results, which may lead to different reasoning depending on the individual. How can the quality of interpretability increasing methods be comparably measured?

The visual illustration of feature importance is commonly accomplished via heatmaps, highlighting each input’s importance for a model. They are the most useful tool for indicating feature importance, especially in domains working with image data, as it is a descriptive format for humans. Figure 1.1 displays an example of two heatmaps for a classification task. Nonetheless, determining the quality of heatmaps remains challenging, as there is no ground truth for which areas are essential to the network. Existing work, for example, showed the validity of attribution heatmaps by zeroing out individual patches based on the received attribution scores [Samek et al., 2015]. However, this approach has its limitations caused by choice of baseline. For instance, if high attributed areas are replaced by zero, the image may drop out-of-distribution and affect a decreased classifier score, or a zero baseline has no impact on black parts of the image.

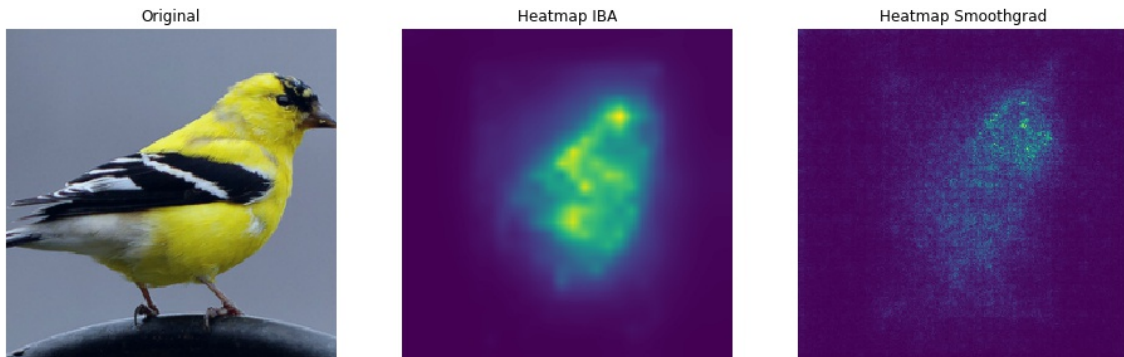


Fig. 1.1 From left to right: The original goldfinch sample taken from the ImageNet dataset; Heatmap computed with IBA [Schulz et al., 2020]; Heatmap computed with Smoothgrad [Smilkov et al., 2017].

This work proposes to control both model and datasets to derive a ground truth for evaluating attribution scores. The first setting reduces the problem’s complexity to the simplest non-linear problem, the XOR gate. Prior knowledge about the importance of both dimensions serves as the basis for evaluation. Furthermore, we utilize an invertible network architecture to establish a ground truth through targeted manipulations in the latent space. By inverting, those manipulations become measurable high-level alterations in the input domain.



## 1.2 Contribution

We tackle the issue of evaluating attribution methods from two fronts:

- **Two-dimensional evaluation:** By reducing the model and dataset complexity significantly, we create a ground truth based on prior knowledge, thus, any initial information about the data or model. This approach utilizes a simple feedforward network to learn the XOR operation, a simple non-linear problem. Prior knowledge is given by assuming that both input dimensions should contribute equally to solve this task successfully. Visual assessment and elementary statistical tools quantify the attribution results of the discussed methods. The Ring of Gaussians problem is introduced, extending the binary classification task to a multi-class problem. While maintaining the two-dimensional setup, the previous assumption still holds. Therefore, their outcome is comparable and conclusions about each method's results are drawn based on both settings.
- **Inversion evaluation:** Manipulations in latent space can be investigated in the input space using an invertible network architecture. We induce specific changes to alter high-level features, which help to approximate a ground truth. By applying a flow-based GLOW model [Kingma and Dhariwal, 2018], parameter manipulations in the latent space can be visualized in the input space through inversion. This property ensures measurable differences in the input domain for targeted changes in any latent space. The setup has shown itself incredibly convenient for evaluating the Information Bottleneck for Attribution [Schulz et al., 2020], as the enforced information loss can be quantified and visualized.

The second chapter covers related work, specifying the interpretability term, discussing the relevant attribution methods for this work and assessing the evaluation difficulties. Chapter three examines the two-dimensional evaluation approach, chapter four the inversion evaluation. A concluding discussion assesses the results before giving an outlook for potential future work.



# Chapter 2

## Related Work

This chapter covers this thesis's related work, concretizing the concept of interpretability and presenting contributing state-of-the-art attribution methods. The chapter further examines the evaluation issue for these methods.

### 2.1 Interpretability

With defining interpretability as "the degree to which an observer can understand the cause of a decision" [Miller, 2018], causality is directly linked to the goal of interpretable models. Miller [2018] presents a fitting definition based on the regularity theory of Hume and Millikan [2007]. Causality between two events exists, if "events of the first type are always followed by events of the second" [Miller, 2018]. Lewis [1973] extends this definition with the concept of counterfactuals, arguing that the simultaneous occasion of two events alone does not suffice for causality. "(E)vent (A) is said to have caused event (B) if, under some hypothetical counterfactual case the event (A) did not occur, (B) would not have occurred" [Miller, 2018]. When switching to the context of ML, these events can easily be translated to parameters of artificial networks.

If a model seems insufficient for the problem, we seek for interpretability [Lipton, 2017]. The desire is even more vital for crucial applications, i.e. affecting human health. Common measures determining a model's performance may not be sufficient, even though it might outperform humans, as we pursue transparent decision-making. Examples for performance measures can be found in the confusion matrix for supervised models, stating relations between positives and negatives. Disregarding cases or architectures, where it is even harder to produce such metrics (i.e. unsupervised learning, probability-based generative models), modelling all aspects of real-world problems may be infeasible. Hence, the demand for

model interpretability is inevitable, especially if increasingly sophisticated models try to solve complex real-world problems.

When concretizing research goals regarding this topic, literature often refers to desiderata of interpretability [Lipton, 2017] [Doshi-Velez and Kim, 2017]. Besides the benefit of building trust, interpretable models support insight by indicating data correlations. The model’s objective might differ from the real-world objective, with the latter likely being data and not necessarily output related. In the context of ethical decision-making, interpretable models aid in bias detection, thus assuring decisions within ethical standards. Nonetheless, to benefit from these properties, attributions need to be sincere and not falsely induced. Misleading interpretations create at least as much harm to a model’s image as genuine transparency would improve it. Therefore, the following methods are evaluated.

## 2.2 Discussion of Attribution Methods

This section aims to introduce recent years’ popular attribution methods to evaluate their results in later stages of this work successfully. The algorithms themselves and their intentions will be laid out, establishing the basis for further evaluation.

The goal of each method can be broken down to assigning an *Attribution* score, sometimes also referred to as *Importance* or *Relevance* score. Suppose a model learning a function  $F$  based on the input  $x = [x_1, \dots, x_n] \in \mathbb{R}^N$  returning the output  $F(x) = [f(x)_1, \dots, f(x)_C]$ , the attribution method assigns an attribution value  $A$  to the input  $x$  given a specific target class  $c$ , capital  $C$  denoting the number of classes. Some architectures, especially used for tasks aside from classification, may not need the specific target class restriction and therefore benefit from multi-class feature attribution.

The majority of the methods can be classified as perturbation-based and gradient-based.

**Perturbation-based** methods alter a specific part of the input, pass it through the network, and compare the output to the original sample to assign an attribution value to specific input dimensions [Ancona et al., 2018]. This alteration of input dimensions is usually done by masking to maintain the input structure.

A popular algorithm is proposed by Zeiler and Fergus [2013]. Their method, called *Occlusion*, uses this approach on an image classification task working on a relatively shallow convolutional model, utilizing a sliding window for masking different image areas. The comparison between the output of each forward pass and the original image results in an attribution score. Additionally, Zeiler and Fergus [2013] implemented each layer’s

approximate reverse operation, performing a deconvolution. By inverting each filter of the model back to the input space, the learned parameters can be visualized and interpreted.

This method’s alterations modify the occlusion algorithm’s hyperparameters, such as mask dimension, masking values, batch-wise permutation, and those related to the model’s architecture. More sophisticated approaches train a model producing generalizable masks [Dabkowski and Gal, 2017], following the same interpretability goal. A disadvantage all perturbation-based approaches share is the long computation time, as a forward pass is required for every sample [Ancona et al., 2018].

**Gradient-based** methods attribute importance by taking partial derivatives of the output with respect to the input. The computation of the derivatives is usually achieved by performing one forward pass of the input to attribute through the network [Ancona et al., 2018]. The alteration of the backpropagation mechanism offers an intuitive implementation approach, calculating partial derivatives with respect to the input instead of the previous layer.

Whether being as fundamental as described above [Simonyan et al., 2013] or designed explicitly around the backpropagation algorithm itself [Bach et al., 2015], they are also described as modified backpropagation methods. Due to the single forward pass by most of them, gradient-based attribution methods are generally computational efficient.

Before visiting each relevant gradient-based method for this work more closely, another criterium for differentiation can be made. Most algorithms or other interpretability increasing approaches function post-hoc, therefore not restricting the model. On the other hand, intrinsic methods produce interpretable parameters during the model’s training. The latter approach is mostly embodied via simple linear models or a specific weight design, hence restricting interpretable parameters. Even though post-hoc attribution methods are more convenient and model-transferable, their result may benefit a subjective purpose [Lipton, 2017]. The additional angle of biased interpretation does not seem to severely affect recent studies, as post-hoc methods receive significantly more attention.

### 2.2.1 Saliency Maps / Gradients

Simonyan et al. [2013] propose an attribution method that relates gradients to each input dimension and computing their absolute value (Eq. 2.1). The computation of the output’s partial derivatives is implemented with regard to the input instead of the previous layer’s output, using a redirected backpropagation algorithm [A.1].

$$a_i^c = \left| \frac{\partial f_c(x)}{\partial x_i} \right| \quad (2.1)$$

$a_i^c$  denotes the attribution value for the input  $x_i$  for a specific class  $c$ ,  $f_c(x)$  the output of the model and  $\partial$  represents the derivative. Taking the absolute value of the gradients restricts this method to attributing positive values only. Therefore, a negative or positive contribution cannot be differentiated; it only captures the magnitude of each gradient.

### 2.2.2 Input $\times$ Gradients

Shrikumar et al. [2017] improve the absolute gradient approach by multiplying the partial derivative with the input and omitting the absolute value of the gradients (Eq. 2.2).

$$a_i^c = \frac{\partial f_c(x)}{\partial x_i} \times x_i \quad (2.2)$$

The product may sharpen the attribution result but induces imbalanced attribution values due to the input magnitude itself. Larger input values result in higher contributions but are not necessarily more significant for the model's decision than lower input values. Nonetheless, allowing the impact of negative gradients enables the distinction between negative and positive attributions.

### 2.2.3 Smoothgrad

Smilkov et al. [2017] present an adaption of the saliency map [Simonyan et al., 2013] algorithm to reduce the noise of the attribution map. The improvement is achieved by adding noise to the input and computing the mean (Eq. 2.3). The averaging over artificially noised images compensates gradient volatility, improving saliency maps to qualitatively more appealing results at least for the human eye.

$$a_i^c = \frac{1}{N} \sum_{i=1}^N \frac{\partial f_c(x + g_i)}{\partial x_i} \quad (2.3)$$

The additional  $g$  denotes an *i.i.d* vector from a normal distribution  $g_i \sim N(0, \sigma^2)$ , with a zero mean and the standard deviation  $\sigma$ .

VarGrad [Adebayo et al., 2018a], as a variation of SmoothGrad, replaces the mean by the variance. (Eq. 2.4).

$$a_i^c = \sigma^2 \left( \frac{\partial f_c(x + g_i)}{\partial x_i} \right) \quad (2.4)$$

### 2.2.4 Integrated Gradients

Sundararajan et al. [2017] propose Integrated Gradients, which introduces a baseline  $x'$  as an additional parameter.

$$a_i^c = (x_i - x'_i) \times \int_{\alpha}^1 \frac{\partial f_c(x' + \alpha \times (x - x'))}{\partial x_i} \delta \alpha \quad (2.5)$$

The first part of equation 2.5 computes the difference between the input  $x$  and the baseline  $x'$  for dimension  $i$ . The integral accumulates every partial gradient  $\frac{\partial f_c(x)}{\partial x_i}$  along dimension  $i$  for a specific class  $c$ . Their element-wise multiplication completes the attribution map  $a$ .

The intention behind the algorithm is driven by multiple axioms, which served as guidance designing this method. The twofold sensitivity axiom states, if input and baseline deviate in one feature and are classified differently, the feature shall be assigned a non-zero attribution value. An example violating this axiom occurs assigning the pure gradient value for creating saliency maps, proposed by Simonyan et al. [2013]; a basic one ReLU layer model with one variable  $f(x) = 1 - \text{ReLU}(1 - x)$ . For input  $x = 1$ , the gradient is flat in every case, even though an input change, i.e. from  $x = 0$  to  $x = 1$ , may have caused a different classification and therefore should have received a non-zero attribution value. The second part of the sensitivity axiom states that if a function does not depend on the input, this input's attribution value is always zero indicating insensitivity.

Implementation invariance refers to the paradigm that attribution values shall be identical for functionally equivalent models. Two differently implemented models  $m_1$  and  $m_2$  are functionally equivalent, if  $f_{m_1}(x) = f_{m_2}(x)$  holds. This axiom becomes central if motivated with the desirable property of attribution methods being implementation independent. By utilizing the backpropagation algorithm for training the network weights, the chain rule  $\frac{\partial f}{\partial g} = \frac{\partial f}{\partial h} \times \frac{\partial h}{\partial g}$  implies implementation invariance, with  $g$  denoting the input,  $f$  denoting the output and  $h$  representing an implementation detail of the network. Gradient-based attribution methods, such as Integrated Gradients, therefore satisfy this axiom by default. Two criticised attribution methods failing Implementation Invariance are Layer-wise Relevance Propagation (Bach et al. [2015]) and DeepLift (Shrikumar et al. [2017]). Their usage of a modified backpropagation algorithm creates attribution results with discretized gradients. The formula of the chain rule does not hold in this case:  $\frac{f(x_1) - f(x_0)}{g(x_1) - g(x_0)} \neq \frac{f(x_1) - f(x_0)}{h(x_1) - h(x_0)} \times \frac{h(x_1) - h(x_0)}{g(x_1) - g(x_0)}$ . In essence, LRP and DeepLift may be sensitive to model specific characteristics, while gradient based, non-modified backpropagation methods are not. At least not for violating the elimination possibility of layer specific parts within the chain rule formula, which are interpreted as the implementation detail [Sundararajan et al., 2017].

Completeness refers to the attribution values adding up to the difference between the output values  $f(x)$  and  $f(x')$ ,  $\sum_{i=1}^n a_i(x) = f(x) - f(x')$ . Thus, the attribution should be equal to the difference between input and baseline. This property provides a reasonable scaling of the network output, which is even more beneficial if its value has numeric sensitive meaning other than the confidence score for labels. Ancona et al. [2018] and Shrikumar et al. [2017] incorporate this axiom in their work as *Sensitivity-n* and *Summation to Delta*.

The axiom linearity refers to the preservation of linear model structures. The following example covers three models  $f_1$ ,  $f_2$  and  $f_3$ , while the third one models the function  $a \times f_1 + b \times f_2$ , i.e., a linear combination of  $f_1$  and  $f_2$ . A conclusive attribution method shall be a weighed sum of  $f_1$  and  $f_2$  with the weights  $a$  and  $b$ .

### 2.2.5 Information Bottleneck

The Information Bottleneck for Attribution (IBA) proposed by Schulz et al. [2020] combines gradient-based approaches with feature perturbation. In short, the idea is the restriction of the information flow through the model while optimizing the model's objective. Each feature's contribution through a restricted model is measured as an attribution value. The restriction is realized by adding noise onto a specific layer's activations, thus perturbation-based due to feature occlusion but still exploiting the model's backpropagation mechanism. The noised bottleneck layer may not be the input and becomes a hyperparameter, especially with the different abstraction levels of deeper convolutional models in mind. Equivalently to the previous gradient-based methods, IBA operates post-hoc.

An informational bottleneck [Tishby et al., 2000] can be realized by introducing noise to a system replacing relevant information. Therefore, a new variable  $Z$  is introduced:

$$Z = \lambda(X)R + (1 - \lambda(X))\epsilon, \quad (2.6)$$

$R$  denotes the output of the bottleneck layer,  $\lambda(X)$  regulates the amount of noise  $\epsilon \sim N(\mu_R, \sigma_R)$  to be added to  $R$ .  $\lambda(X)$  is equally shaped as  $R$  with  $\lambda(X)_i \in [0, 1]$ . Equation 2.7 describes the following step for computing the mutual information  $I[R, Z]$  to estimate the information  $Z$  still contains of  $R$ .

$$I[R, Z] = \mathbb{E}_R[D_{KL}[P(Z|R)||P(Z)]] \quad (2.7)$$

$\mathbb{E}_R$  denotes the expectation,  $D_{KL}$  the Kullback-Leibler divergence,  $P(Z|R)$  and  $P(Z)$  the probability distributions. At this point, the authors assume that  $P(Z)$  is independent and normally distributed  $Q(Z) \sim N(\mu_R, \sigma_R)$ . The motivation for this assumption is, on the one hand, the infeasible calculation of  $P(Z)$  due to the integration over the feature map  $R$ . On the



other hand, the reasonable guess that latent layers tend to be Gaussian distributed [Klambauer et al., 2017].

$$a_I^c \triangleq I[R, Z] = \mathbb{E}_R[D_{KL}[P(Z|R)||Q(Z)]] - D_{KL}[Q(Z)||P(Z)] \quad (2.8)$$

Equation 2.8 includes the applied and derived assumption. With the known notation throughout this work, the mutual information corresponds to the attribution map in latent space  $a_I^c$ .

One forward pass achieves the computation of  $a_c$  with the IBA attribution method. Therefore, the loss function is enhanced with  $\mathbb{L}_I$ .

$$\mathbb{L}_I = \mathbb{E}_R[D_{KL}[P(Z|R)||Q(Z)]] \quad (2.9)$$

The mutual information's first summand (Eq. 2.8) can be translated directly to the information loss function  $\mathbb{L}_I$  to minimize (Eq. 2.9). Even though the independence assumption may not hold, the second summand being mandatory positive leads to an upper bound for the mutual information, if disregarded.

$$\mathbb{L}_A = \mathbb{L}_M + \beta \mathbb{L}_I \quad (2.10)$$

Equation 2.10 completes the loss function for the attributing forward pass. Therefore, the information loss is parametrized and added to the model's loss function  $\mathbb{L}_M$ . The objective of minimizing the attribution loss  $\mathbb{L}_A$  is now dependent on the model's performance, optimized with  $\mathbb{L}_M$ , but also on the  $\mathbb{L}_I$ . The enhanced loss function incentivizes the model to be as accurate as possible while holding less information due to the induced noise. This mechanism enforces the relevant information of a feature map to pass the narrowed information bottleneck.

To receive the attribution result in the input space, the desired  $a^c$ , the authors suggest bilinear interpolation for the spatial dimension to preserve local information as good as possible. The aggregation of the channel dimension is done by summation.

Schulz et al. [2020] propose two procedures for the determination of  $\lambda(X)$  (Eq. 2.6), and therefore each dimension's noise level: the Per-Sample Bottleneck and the Readout Bottleneck.

As the name suggests, the first approach optimizes  $\lambda(X)$  for one sample only. The processing steps include squashing  $\lambda(X)$  with the *sigmoid* function and induce blurring with the magnitude of the standard deviation to minimize architectural specific inaccuracies, see

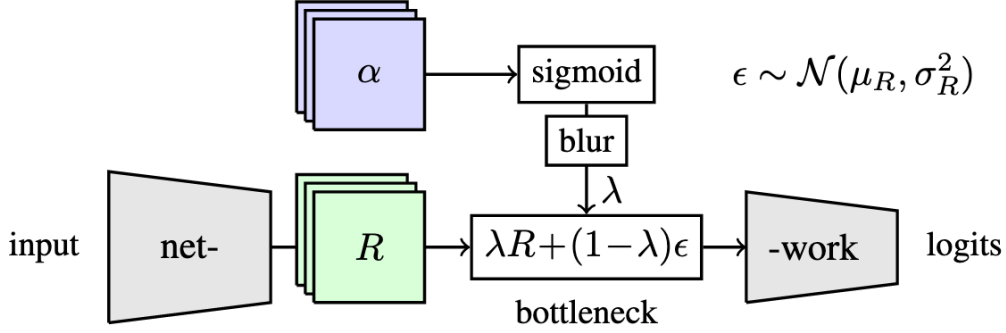


Fig. 2.1 Illustration of the IBA Per-Sample Bottleneck; The blue mask has an  $\alpha_i$  for each dimension  $R_i$  in the green feature map  $R$ . The Per-Sample Bottleneck optimizes the  $\alpha$  mask for one sample, which determines how much information passes for each  $R_i$ . ([Schulz et al., 2020]; Reprinted with the authors permission)

Figure 2.2. An exemplary inaccuracy is pooling layers ignoring parts of the input. This *local smoothness* prevents resulting attribution artefacts.

The Readout Bottleneck introduces a separate model to predict the  $\alpha$  mask in a previous forward pass. First, feature maps of different depths are collected and bilinear interpolated to enforce equal spatial dimensions. Without adding noise,  $\alpha$  is predicted based on these feature maps. A second forward pass then inserts the bottleneck.

## 2.3 Evaluating Attribution

Quantification of attribution quality is no straightforward task. The lack of ground truth needs to be compensated through appropriate metrics based on the attribution method’s technique. Newly introduced methods most likely outperform existing ones to establish themselves as useful. Thus, each published algorithm achieves more outstanding benchmarks, higher metric scores or has visually more appealing results on cherry-picked samples than others. How can attribution results be objectively measured?

Yang and Kim [2019] distinguish evaluation techniques based on the following characteristics:

1. Sensitivity measures of explanations to model and input perturbations
2. Accuracy drop due to highly attributed feature removal
3. Prior knowledge of feature importance
4. Human visual assessment

**Perturbation sensitivity measures** mimic the idea of perturbation-based attribution methods, reducing partial information and observe the output. In this case, the result to evaluate is the attribution map instead of the model output. The perturbed information flow is either realized via a change in model parameters or data manipulation. Adebayo et al. [2018b] introduce a test framework based on perturbation sensitivity, which is also referred to as sanity checks. These checks are not method-specific to support comparability.

**Accuracy drop due to highly attributed feature removal** describes a framework proposed by Samek et al. [2015]. Their locally applied feature removal, called regional perturbation, is a generalization attempt of Bach et al. [2015]. If the model accuracy suffers from highly attributed feature removal, the attribution can be validated. Although, without another training after removal, the manipulated input may be outside of the input distribution (OOD). Therefore, the accuracy drop may be due to OOD input data, not necessarily due to correctly attributed features. Hence, Yang and Kim [2019] propose input substitution within the input data distribution for a sound attribution evaluation.

**Prior knowledge of feature importance** approaches the evaluation with a (partially) known ground truth [Kim et al., 2017] [Yang and Kim, 2019]. This knowledge is based on data features that are supposedly relevant for a model. Prior knowledge may also derive from the network’s parameter structure if the problem at hand is sufficiently simple. The evaluation becomes plausible with appropriate metrics comparing results to the presumed ground truth.

**Human visual assessment** gains insights on human perception of attribution methods. Systematic tests shall obtain how useful attribution maps are for humans, i.e. understanding a model’s decision or predicting the decision solely based on the attribution.

The evaluations done in this work mostly rely on prior knowledge and a visual assessment to quantify the quality of state-of-the-art attribution methods. As the image domain and the two-dimensional results are visually appealing, a visual assessment seems plausible. Ultimately, for transparency communication, the visual illustration of transparency remains essential.



# Chapter 3

## Two-Dimensional Evaluation

This chapter describes the setup, defines the objective, quantifies the outcome and summarizes the results of the two-dimensional evaluation.

**XOR** The problem with evaluating attribution is that we neither understand the model nor the data. This approach aims to tackle the problem with a drastic reduction in complexity, both for model and data. Thus, we look at the simplest non-linear problem: the binary logical *XOR* gate.

Table 3.1 Value table of the exclusive disjunction (XOR)

$x_1$	$x_2$	$y$
0	0	0
0	1	1
1	0	1
1	1	0

The adaption from a binary to a continuous input space  $D_{XOR} = [0; 1]$  increases the problem's difficulty. As we also want to introduce cases of ambiguity, we induce noise. Each data point of Table 3.1 is sampled from a normal distribution with a standard deviation of  $\sigma = 0.2$ , its original input value serving as the mean to create a dataset with the size of 10.000 [Figure 3.1]. Due to the ambiguity for the threshold 0.5, the XOR operation is not clearly solvable for this input. This property translates into the following assumptions for the ground truth.

**Ground Truth** Suppose we would have a model that learned the optimal solution for the continuous XOR problem. What would be the ground truth attribution? It underlies two assumptions:

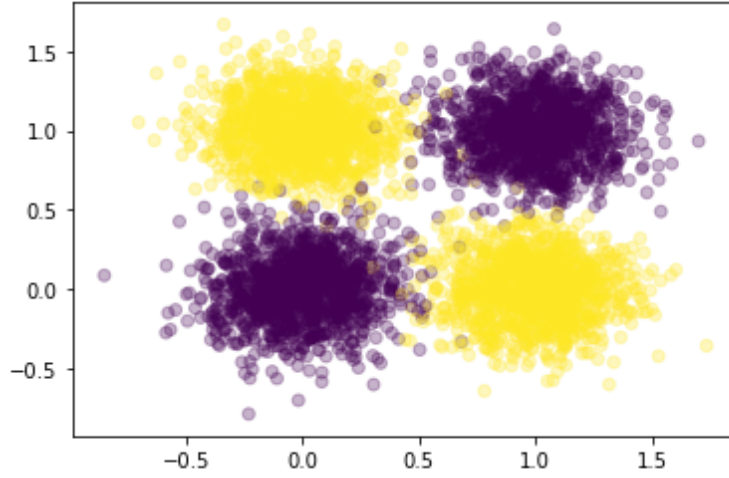


Fig. 3.1 Dataset for the continuous XOR problem. Each axis represents one input dimension  $x_i$ ; each color represents one output class.

1. Both input dimensions  $x$  contribute equally to predict  $y$  for the continuous XOR gate if the problem is solvable.
2. The closer an input  $x$  is to the threshold of 0.5, the less solvable is the problem with 0.5 causing unsolvability.

The first assumption is made based on the characteristics of the binary XOR gate. Both input dimensions are needed equally to determine the output with certainty. If either of the two inputs were removed, a prediction would equal a coinflip. Due to the continuous input space, this scenario becomes possible. Therefore, we introduce a second assumption. If at least one input is 0.5, the problem is not solvable, as it can be interpreted either as 0 or 1. As a result, the other dimension's value becomes irrelevant.

Translated into desirable attribution values, an accurate attribution result reflects an equal attribution score if the distance between the input and the decision boundary is equal. In this case, both input dimensions provide the same amount of information. Then again, the closer  $x_1$  is to the threshold of 0.5, the less relevant is  $x_2$ , as the prediction turns into a guessing game. Thus,  $x_1$  should receive a higher attribution cancelling out  $x_2$ . Ideally, the magnitude of the attribution difference represents their distance to the decision boundary. For instance, the importance for  $x = (0, 0)$  should be equal and maximally high, whereas for  $x = (0, 0.5)$ ,  $x_1$  should be attributed with zero and  $x_2$  positive, as it enforces a coinflip scenario.

**Model** With simplicity in mind, we created a model inspired by the universal approximation theorem. It states that a feedforward network with at least one unrestricted hidden layer can

represent any arbitrary function [Leshno et al., 1993]. The fully connected network XOR\_net consists of the two-dimensional input layer, one hidden layer with  $n = 10$  nodes and a single output node for the binary classification. For the optimization, it uses the *mean squared error* loss function, *stochastic gradient descent* with a *learning rate* of  $10^{-3}$  and *momentum* of 0.9.

**Setup Attribution Methods** We use the Captum<sup>1</sup> library for the attribution method’s implementation, except for the Information Bottleneck for Attribution [Schulz et al., 2020]. The mandatory adjustments are twofold to run the published IBA implementation<sup>2</sup> on the low dimensional input. The noise inducing Gaussian kernel is downsized and the information bottleneck is applied directly to the model’s input. This approach’s chosen noise level is  $\beta = 10^{-9}$ , preserving the model’s accuracy almost entirely. We use a Binary Cross-Entropy loss function for the logit score and the label, which increases as the predicted score differs from the label. The BCE loss is added to the IBA loss. The Integrated Gradient method runs with a baseline of 0.5 to avoid the imbalance of the zero initialization. The Occlusion baseline is also set to 0.5 to avoid a direct suggestion of information through feature replacement. If one input is 0.5 and the second input is occluded with 0.5, both inputs provide zero information for a prediction.

The following sections visually examine the results of each method and quantify these observations. A summary concludes this chapter.

## 3.1 Visual Evaluation

This section aims to identify the strengths and weaknesses of the attribution methods visually. The attribution results are outlined for every input combination of the XOR operation, enabling a visual assessment. The XOR problem creates a two-dimensional space for each input dimension, showing the attribution value for a specific input, see Figure 3.2. The axes of each subplot represent the input dimension  $x_1$  and  $x_2$  over the entire input domain  $D_{XOR}$ . The first subplot of Figure 3.2 visualizes the model’s output, whereas the other subplots illustrate each dimension’s attribution individually. Hereafter, the result of each subplot is evaluated.

**Model Output** The model output accurately illustrates an XOR gate’s decision boundaries, complying with the measured test accuracy of  $\sim 99\%$ . Around the decision boundaries, 0.5 for

<sup>1</sup><https://captum.ai>

<sup>2</sup><https://github.com/BioroboticsLab/IBA>

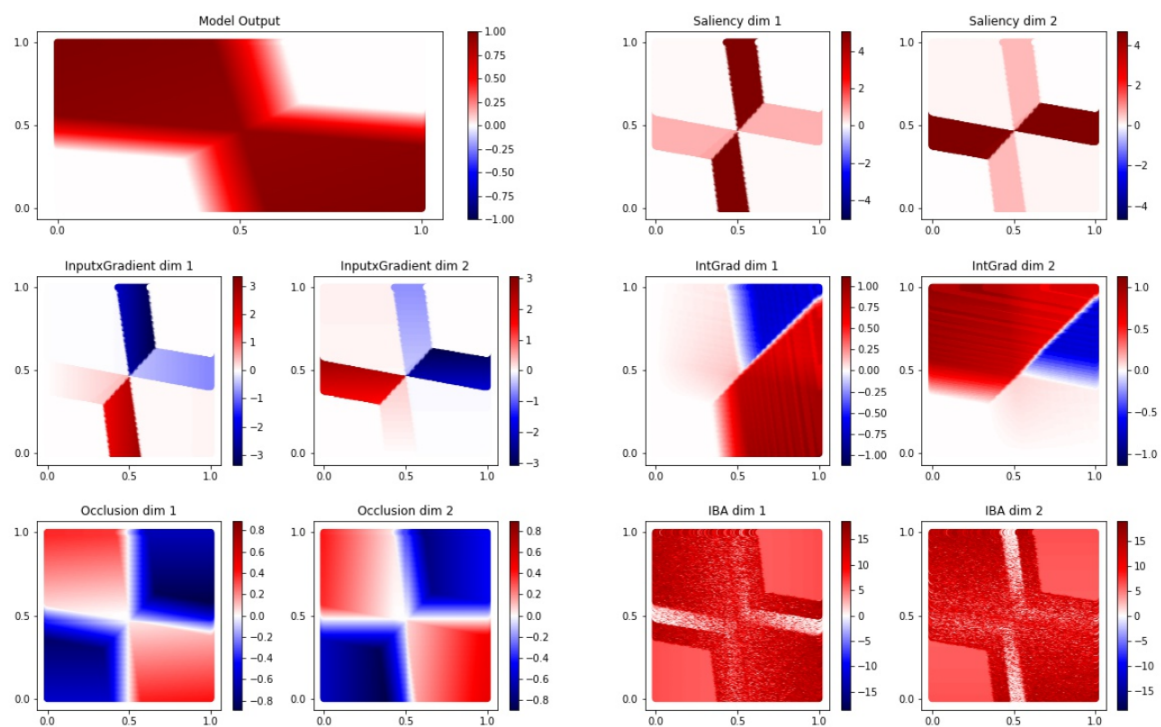


Fig. 3.2 Axes represent the two input dimensions  $x_1$  and  $x_2$ . From top left to bottom right: Model output of XOR\_net; Each input dimension's attribution results: Saliency Maps; Input  $\times$  Gradient; Integrated Gradients; Occlusion; Information Bottleneck for Attribution.



the respective input dimension, the model outputs values around 0.5, signalling insecurity as the input provides not enough information. Additionally, the gradual output changes passing the threshold area indicates the model's indecisiveness when facing those decision boundaries. In the centre area, when both inputs are  $\sim 0.5$ , the model outputs 1 with certainty. This speaks for an imperfect model, as the decision should be more unclear. The false certainty can be reduced to the lack of samples in this area, as we sample with a small standard deviation around 0 and 1.

The relevant literature sometimes distinguishes between attribution and sensitivity. Throughout this evaluation, we interpret a positive outcome of an attribution method as the attribution.

**Saliency Maps** Saliency Maps and  $\text{Input} \times \text{Gradient}$  appear similar in their behaviour, which is not surprising given their relation. Both methods attribute high values for inputs around 0.5 for their respective dimension. Consequently, the outputs appear opposed for each dimension. The result for saliency maps is aligned with the ground truth. The attribution is high if the input is around 0.5, disregarding the other dimension. The respective area around the decision boundary for input values  $> 0.5$  is attributed negatively by the  $\text{Input} \times \text{Gradient}$  algorithm. This result does not reflect the ground truth, as the second dimension becomes irrelevant for inputs around the threshold. According to this method, the model only values the bottom and left area near the decision boundary and disregards the threshold area for inputs  $> 0.5$ . Both methods attribute zero for the large regions of zero gradients. Even though this attribution affects both dimensions equally, some information should be derived.

**Smoothgrad** The Smoothgrad method is not relevant for this evaluation, as its original approach de-noises the attribution output of absolute gradients, hence Saliency Maps. The characteristics remain the same since it computes the mean of the input's noise-induced samples.

**Integrated Gradients** Integrated Gradients show a broader attribution result, valuing not only the respective decision boundaries but the opposite quadrant. Thus, the attribution results for e.g.  $x_1 = 0, x_2 = 1$  differ by a large margin, contradicting the ground truth. At non-threshold areas, the attribution results between the dimensions should look more aligned, as they both provide information for the prediction. This algorithm's results seem far worse compared to other methods and regarding the assumptions.

**Occlusion** The perturbation-based Occlusion algorithm attributes equal attribution values between the two input dimension. Therefore, the results are aligned with the ground truth assumptions. We receive zero attribution value for 0.5, which is amplified by the baseline choice of 0.5. Suppose an input is 0.5 and the method occludes the second input with 0.5. In that case, there is no change in output score measurable for the unsolvable problem, as the model is already maximally insecure. While we could modify the baseline choice for  $x = 0.5$ , this would not be practical in a more complex case.

**IBA** The IBA values both input dimensions more equally compared to the previous gradient-based methods. Assigned importance seems visually similar for both dimensions but noisy. Attribution values along the decision boundaries tend to be smaller for the respective dimension, whereas the attribution values spatially distanced from these boundaries resemble the output structure. The low attributed areas around the threshold of 0.5 align with the ground truth. However, the imbalance between both dimensions around the decision boundaries is visible.

## 3.2 Quantification

We introduce two metrics to quantify these observations. The first focuses on the difference in attribution values between the two input dimensions, thus supporting the assumption that both dimensions should contribute equally to the model’s outcome. The second approach measures Pearson’s correlation coefficient between the dimensions. Each dimension’s result is masked, separating threshold and non-threshold areas. Therefore, the correlation measure aligns with the second assumption of the ground truth.

### 3.2.1 Dimensional Difference Quantification

The difference in attribution score for the input space  $D$  is computed utilizing Equation 3.1. Figure 3.2 illustrates the corresponding plot.  $A_x$  denotes the attribution output of one dimension over the input space.

$$D = |(A_{x1} - A_{x2})| \quad (3.1)$$

As we are interested in the relation between attribution values rather than their absolutes, Equation 3.2 applies a min-max normalization to squash the values of  $D$  between 0 and 1. The  $\varepsilon$  denotes a small value, added to the maximum preventing a division with zero.

$$|D| = D - \min(D) / ((\max(D + \varepsilon) - \min(D))) \quad (3.2)$$

Lastly, we take the sum of the normalized data  $|D|$  divided by the number of steps created to represent the input space. The comparable scalar value  $d$  represents the attribution difference between both dimensions (Eq. 3.3).

$$d = \sum \frac{|D|}{\#steps} \quad (3.3)$$

Table 3.2 Sum of dimensional difference between both input dimensions (rounded to two decimal places); The lower the value, the more equal the attributions between both dimensions.

Method	Saliency Maps	InputxGradient	Integrated Gradients	Occlusion	IBA
$D$	61.41	47.59	99.81	58.26	36.28

Integrated Gradients show the most different attribution results between the dimensions, which seems reasonable regarding its large area of different values, see Figure 3.3. saliency Maps, Input  $\times$  Gradient and Occlusion, show similar results with this metric. These results do not reflect the visual evaluation. Occlusion and Saliency Maps show structurally correct results based on the ground truth, which is not showcased with this metric. IBA shows the lowest Sum of Difference between both dimensions, achieving the best alignment with assumption one.

The difference metric seems unfit, as it measures the sheer difference neglecting any output structure. The following correlation-based metric is better suited to take structural characteristics into account.

### 3.2.2 Correlation-based Quantification

This metric’s motivation is also induced by the second assumption defining the ground truth. If one input is close to the threshold, it should be attributed high, as the second input becomes irrelevant for the prediction. Therefore, we differentiate between two areas via masking: the decision boundary area  $mask_{DB}$  and the corner area  $mask_C$ .

$mask_C$  is defined for the input space’s model output  $< 0.2$  and  $> 0.8$ . This extracts the areas towards the corners of the space, with a large distance to the decision boundary.  $mask_{DB}$  is defined as the negative of  $mask_C$ . After normalizing (Eq. 3.2) the individual dimension outputs for each attribution method, both masks are applied. The filtered entries can be compared between the dimensions via the Pearson correlation coefficients, see Equation 3.4.  $R$  denotes the correlation coefficient,  $C$  denotes the covariance between its indices. As

we compute the coefficients between each dimensions'  $mask_{DB}$  and  $mask_C$ , two comparable values are obtained, describing the high and low informational areas. For  $mask_{DB}$ , one dimension is rotated by  $90^\circ$ . We only receive meaningful results if the input dimensions are compared to their respective attribution results at the threshold.

$$R_{ij} = \frac{C_{ij}}{\sqrt{C_{ii} + C_{jj}}} \quad (3.4)$$

This quantification approach separates the high information area from the less informational dense area towards the decision boundaries. The resulting values measure for linear correlation, from -1 negative to +1 positive linear. The following results are rounded to three decimal places.

Table 3.3 Pearson's correlation coefficient measuring linear correlation between the area distant from the threshold  $mask_C$  and decision boundary area  $mask_{DB}$ . High positive correlations between both masks indicate a better performance regarding the defined ground truth.

Method	Saliency Maps	InputxGradient	Integrated Gradients	Occlusion	IBA
$mask_{DB}$	0.809	-0.363	-0.300	0.559	0.014
$mask_C$	0.199	0.291	-0.582	0.949	0.520

Evaluating  $mask_C$ , Integrated Gradients show a negative correlation. The remaining methods all show a positive correlation indicating equality for this area. Confirming the visual impression, Occlusion's values show a strong correlation, and IBA performs second-best in this metric. The slightly positive Pearson coefficient for Saliency Maps and Input  $\times$  Gradient should be treated with care, as almost the entire masked area has attribution values of zero. This metric does not consider zero values attribution, it only measures the dimension's correlation. With Integrated Gradients sharing the same gradient-based properties, one would assume the result somewhat embellishes their performance.

Regarding the coefficients for  $mask_{DB}$ , Saliency Maps and Occlusion are strongly correlated. Both results align with the visual evaluation. Still, the Occlusion result of this area loses significance since it has very little information due to the baseline choice. Input  $\times$  Gradient and Integrated Gradients shows a negative correlation for this high informational area. IBA shows no correlation between both dimensions.

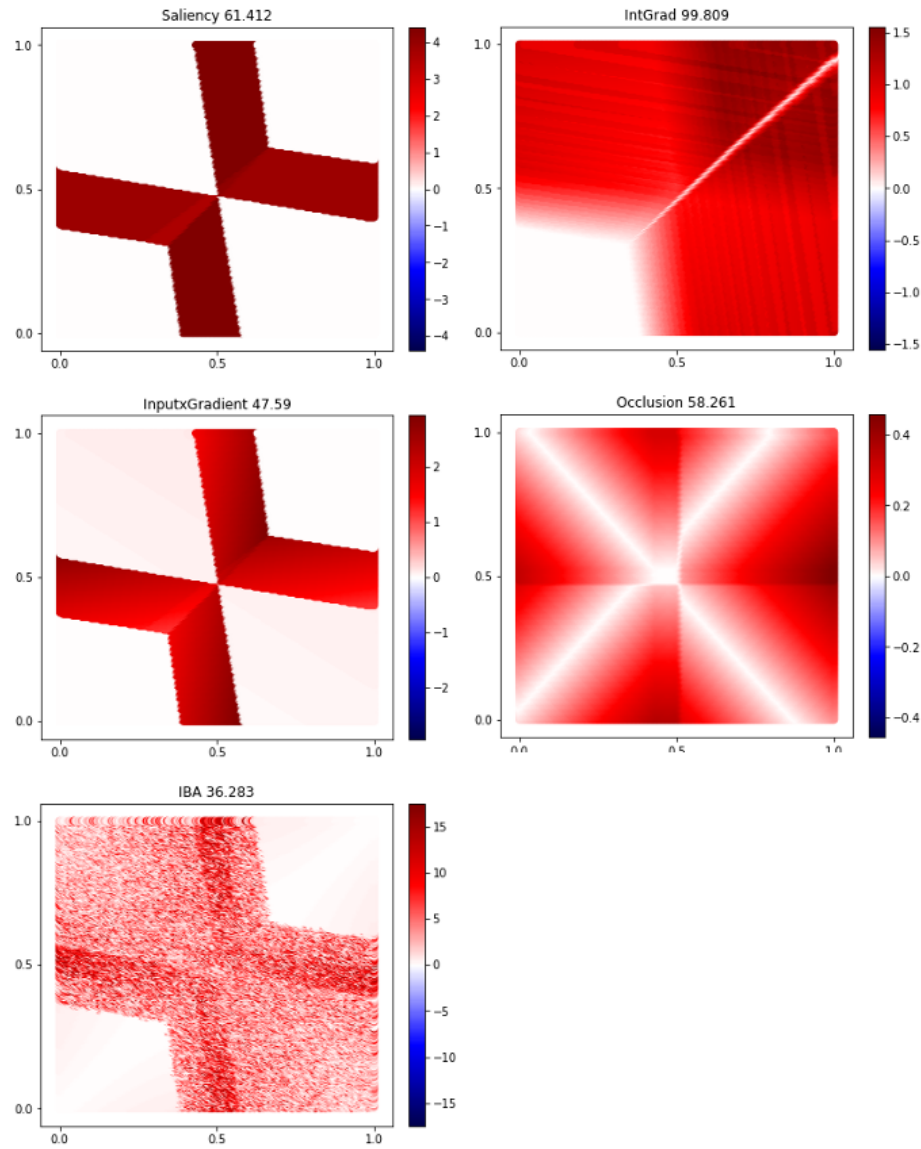


Fig. 3.3 Visualization of the difference in attribution values  $D$  between both dimensions for the entire input space through XOR\_net. From top left to bottom right: Saliency Maps; Integrated Gradients; Input  $\times$  Gradient; Occlusion; IBA.

### 3.3 Summary

This evaluation demonstrates the strengths of perturbation-based attributions and weaknesses of gradient-based attributions for low dimensional problems. This section presents a summary for each method addressed throughout this sanity check.

Occlusion, as a representative for perturbation attribution methods, performs exceptionally well in this setup. The approach’s main weakness does not affect the low dimensional application, local dependencies of inputs. With the sliding window approach and the lack of smart and dynamic window sizes and value replacement techniques, which are not involved here, Occlusion is limited in its performance. Nonetheless, in its pure form and for independent inputs, the method outperforms its gradient-based counterparts.

Saliency Maps also show great alignment with the defined ground truth. The biggest issue is the missing gradients for large parts of the input space which are then attributed with zero. Even though this shows consistency between both dimensions, missing attributions are not necessarily evaluable.

Input  $\times$  Gradient shares the problem of the missing gradients. It also misses the second assumption, as threshold inputs should be attributed highly. Overall it shares the same characteristics as Saliency Maps and the multiplication with the input has no considerable effect in this setup. The measurable difference is the negative attribution, which misleads in this case.

Integrated Gradients is outperformed by the other methods, as it could not match the ground truth with its results. The additional baseline hyperparameter has no significant influence in the two-dimensional setting.

IBA reflects outperforms the pure gradient-based methods in this low dimensional sanity check for inputs distant from the decision boundary. The attributions are equal between both dimensions. For inputs on the decision boundaries, IBA assigns low attribution values contradicting the ground truth. A weakness of the method is the random noise perturbing the attribution result.

# Chapter 4

## Inversion Evaluation

This chapter introduces the Inversion Evaluation, for which we lift the restrictions on the dataset and model. We utilize the invertible property as a transparency increasing tool to derive a ground truth. Based on this ground truth, Integrated Gradients [Sundararajan et al., 2017] and IBA [Schulz et al., 2020] are evaluated in a high dimensional environment using image data and a GLOW-based model architecture. The first method is examined in-depth, especially for its baseline concept. Furthermore, we optimize IBA to invert relevant high-level features and visualize them in the input space.

### 4.1 Setup

The approach behind this evaluation is comparable to the two-dimensional. In order to assess the attribution quality of any method, we need to define a ground truth. This section describes the setup for establishing this ground truth.

**Dataset** The CelebA dataset provides 202.599 facial images of 10.177 individuals. Each data point is manually annotated with 40 binary labels describing its characteristics. Those labels range from *having a beard* or *gender* all the way to more abstract properties like *attractiveness*. The latter induces an even more subjective bias since the labelling is only binary. Nonetheless, CelebA is a real and comprehensive dataset providing high-quality images. The image content varies in its pose, expression and background, but is similar regarding the proportion between background and facial region. With images being a prime example for locally dependent data, the model and attribution methods should identify them as high-level features.

**Model** Approaching this evaluation, we introduce the flow-based invertible model GLOW [Kingma and Dhariwal, 2018]. This model learns the input data distribution using a sequence of the same invertible function, the *normalizing flow*. (Eq. 4.1)

$$x \xleftrightarrow[f_1]{f_1^{-1}} h_1 \xleftrightarrow[f_2]{f_2^{-1}} h_2 \xleftrightarrow[f_3]{f_3^{-1}} h_n \xleftrightarrow[f_n]{f_n^{-1}} z \quad (4.1)$$

$x$  represents the input variable,  $z$  the output variable,  $h$  the latent variables and  $f$  the invertible function. Due to the choice of  $f$ , latent states can be inverted backwards. (Eq. 4.2)  $\theta$  denotes the parameters.

$$f(x) = g_\theta(z), \quad \text{with } f_\theta = g_\theta^{-1} \quad (4.2)$$

With  $p_\theta$  denoting the probability density function, Equation 4.3 shows the invertible function.

$$\log p_\theta(x) = \log p_\theta(z) + \log \left| \det \left( \frac{\delta z}{\delta x} \right) \right| \quad (4.3)$$

A well-trained model has learned the data distribution in an unsupervised fashion and can pass a specific sample forwards and backwards through the network, reconstructing the sample. Sixt et al. [2021] introduce their GLOW model to precisely manipulate learned high-level features in latent space. To enable these manipulations, such as gender-swaps or the smile's intensity, the authors append a linear classifier on the GLOW model's latent variable. Therefore, each sample can be classified via 40 labels and more importantly, these labels can be adapted before inverting the sample. We are not restricted to a binary flip in label values, as we work in a continuous space.

Sixt et al. [2021] refer to the manipulated inverted samples as counterfactuals. The creation of a counterfactual  $\hat{x}$  is formalized as follows.  $f(x)$  denotes the forward pass of the invertible model computing the logits  $z = f(x)$ . The linear classifier obtaining the labels  $y$  is defined as  $c(x) = \omega^T z + b$ , with  $\omega$  representing the weights and  $b$  the bias. The computation for the counterfactual is therefore  $\hat{x} = f^{-1}(z + \alpha \omega)$ . By setting  $\alpha$  accordingly, we can create specific manipulations for high-level features and are able to choose the magnitude of these changes freely.

As we work with the setup of Sixt et al. [2021], we acquire the following two design choices of the authors. The first one is the utilization of downsampling steps. Using the CelebA dataset reduces the input dimensionality from (3, 128, 128) down to 786, ignoring half of the information for each downsampling step. These steps are applied every 50 layers, hence seven times in total. For the inversion, the lost values are appended again, primarily to keep the latent dimension's shape as they only contain a fraction of the information. The second choice is the adapted loss function. The unsupervised loss is combined with the



classifier's supervised loss to train the model on the dataset's labels successfully. On the trained model, the classifier is applied after layer 321, as the supervised loss is disabled after this layer.

**Ground Truth** How can we define the ground truth, and how can we determine a successful attribution?

The ground truth can be derived from the invertible model's property, combined with counterfactuals. For instance, two counterfactuals  $\hat{x}_1$  and  $\hat{x}_2$  respectively with a negative and positive logit score for label 22 ("*moustache*") are inverted (Fig 4.1). With a sufficiently trained model, different "moustache" intensities are visible. Therefore, both images differ primarily in the region between the nose and mouth. This visible difference is measurable and approximates the ground truth for this specific model and label.

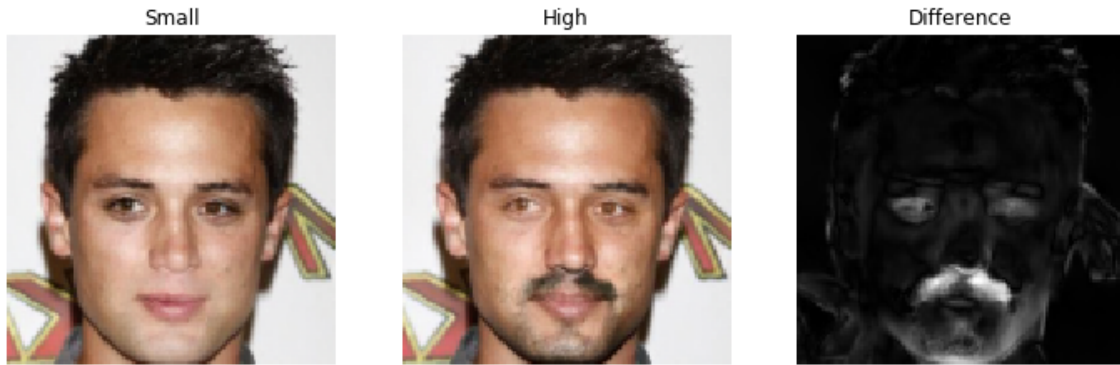


Fig. 4.1 The left picture shows a sample image with a small score for the label "*moustache*"; The centred picture shows a high score for this label; The right picture visualizes the difference between left and centre.

The attribution scores for  $\hat{x}_1$  and  $\hat{x}_2$  and target label 22 need to be particularly high in this region, to comply with the ground truth. The model associates the moustache label closely with this area.

A second part of the ground truth can be derived from the observations of Sixt et al. [2021]. The background changes for certain labels if its target value is manipulated. Figure 4.2 illustrates examples for the "*Attractiveness*" label. The tendency towards a brighter background for more *attractive* samples is visible. We expand the ground truth towards background importance. Though the background does not seem important for face related features, it could be a spurious correlation learned by the model.



Fig. 4.2 Generated CelebA images solely manipulated for the target label "*Attractiveness*" ([Sixt et al., 2021]; Reprinted with the authors permission)

**Setup Attribution Methods** For the high dimensional evaluation, we use our own Integrated Gradient implementation. Hence no attribution library is used. The variation of the baseline parameter into different extremes inspects its importance to the overall attribution. For the IBA evaluation, we use the published implementation <sup>1</sup>. Specific evaluations adapt the loss function to generate different results for the same sample to optimize the attribution quality.

The following section examines the Integrated Gradient algorithm and presents its attribution result for high dimensional image data. Section 4.3 evaluates the attribution result of IBA and inverts them back to the input space for visualization. A summary concludes this chapter.

## 4.2 Evaluation of Integrated Gradients

Sundararajan et al. [2017] motivated their attribution method with multiple axioms, which equip Integrated Gradients with useful properties. These properties separate this method from others which use a more straightforward gradient approach or modify the backpropagation algorithm for attribution. This section commits itself to an empirically driven evaluation of the algorithm and concludes how these properties apply in practice.

<sup>1</sup><https://github.com/BioroboticsLab/IBA>

The evaluation is twofold. The first segment tackles the gradients and determines their suitability for the attribution task. Therefore, the result also impacts other gradient-driven methods. The second focus is on the baseline. The subsection examines the importance and selection criteria for this hyperparameter.

### 4.2.1 Gradient Independency

Gradients are the model's tools to learn highly complex functions based on observations. More specifically, in which direction to optimize the parameters. As loss functions are usually formulated to be minimized to improve the objective, the first derivative, the gradient, indicates this function's slope and its direction. The adjustment of the model's parameters in the negative partial derivative direction increases the model's performance. With these gradients as a central element of artificial neural networks, it seems reasonable to repurpose them for attribution like Simonyan et al. [2013] work.

With gradients being the crucial part of the Integrated Gradients algorithm, are they fit to indicate attributional value? In order to provide unbiased information, gradients need to be independent of the input value. Translated to image data, the gradient of a pixel value cannot be influenced by that value to assign a fair attribution for every possible value. If that is not the case, an unbiased attribution based on gradients would not be possible, and the input value itself would influence its attribution score. To assure that this is the case, we calculate the gradients for a specific sample with regard to the input and compare their value pixel-wise (Fig 4.3). Red indicates the pixel values of the image, and by visual assessment, local dependencies are recognizable as the small plateaus with equal values. The blue dots appear more distributed.

We measure the correlation between the input sample and its gradients to support the results. The Pearson coefficient measuring linear correlation is 0.013, the Spearman correlation coefficient 0.019, indicating no rank correlation between both values. These results support the use of gradients for attribution. Locally dependent data structures can be reflected independently using the gradients of the model.

### 4.2.2 Baseline Impact

In this section, the influence of the baseline hyperparameter is examined. Furthermore, we determine the optimal baseline within this setup, by measuring its relevance with regard to the ground truth.

Recalling the equation from chapter two, Integrated Gradients requires a baseline for both parts of the term (Eq. 2.5 & 4.4). The first one computes the difference between input

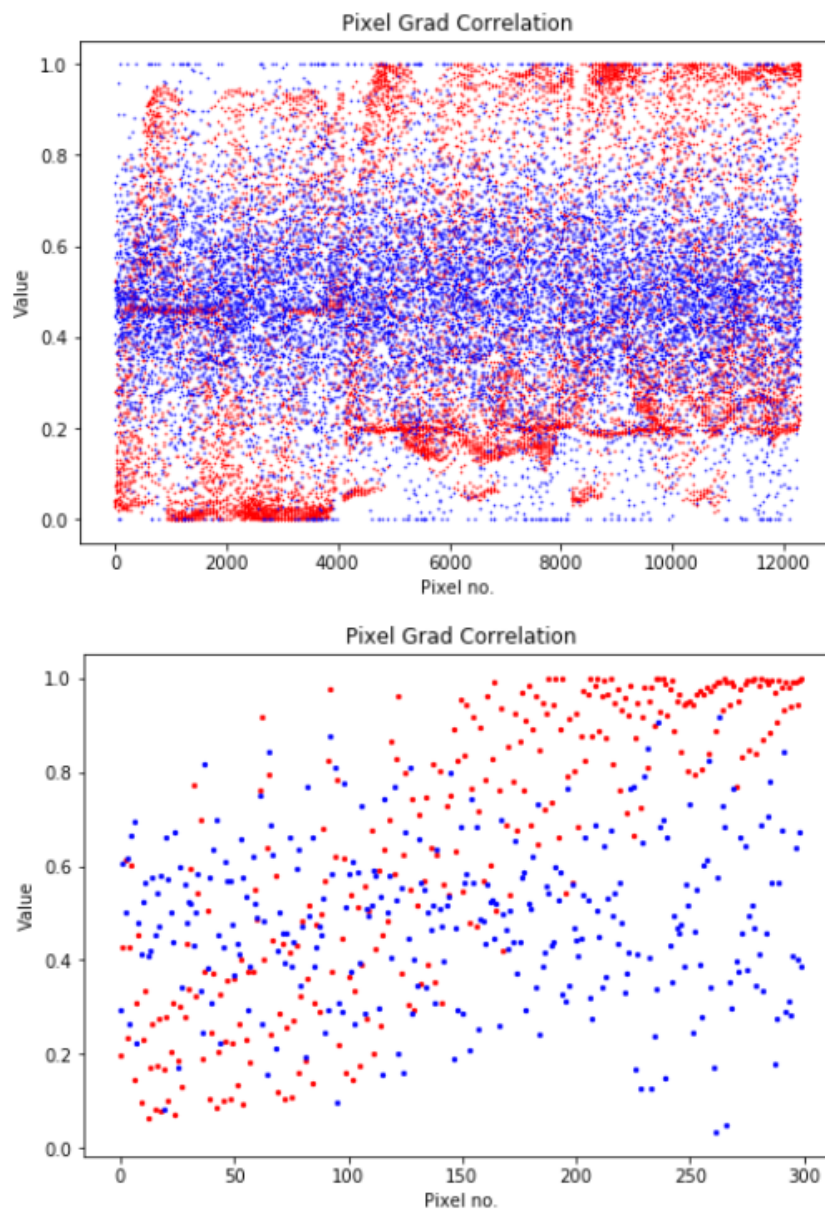


Fig. 4.3 Red dots indicate pixel values of an input sample; Blue dots indicate the gradient value of the model's output with regard to the input pixel. The top subplot shows the result for one entire image including the three color channels. The bottom subplot depicts a sequence of 300 data points each.

and baseline, multiplied element-wise with the integral. Moreover, the baseline is also the integration's key element, as it influences the partial derivatives severely. Translated toward application, the baseline likely has a significant impact on the attribution result.

$$a_i^c = (x_i - x'_i) \times \int_{\alpha}^1 \frac{\partial f_c(x' + \alpha \times (x - x'))}{\partial x_i} \quad (4.4)$$

**Impact Evaluation** The baseline assessment is not apparent since it is also a crucial element for the integral calculation. Still, we disentangle both parts of the computation to examine them separately. The assumption is that the pure difference between baseline and input provides essential information for the attribution result. In the following, we separate between baseline and gradient computation to evaluate them individually. The fact that the baseline also significantly affects the gradient calculation only leads to underestimating its actual impact.

We define the baseline part  $P_i^B = x_i - x'_i$  and the integral part  $P_i^I = \int_{\alpha}^1 \frac{\partial f_c(x' + \alpha \times (x - x'))}{\partial x_i}$ . Figure 4.4 displays the resulting heatmaps for the Integrated Gradients method (column two) and each part (column three and four) for two samples with two different baselines, respectively. The first and third row received a zero baseline, whereas the second and fourth row got the inverted original picture with a reduced smiling target. The latter provides much information for the attribution, as it resembles the ground truth. Therefore, the results of column two show a much more targeted attribution result for the high informational baseline. The zero baseline outcome remains noisy, as the  $P_i^I$  generates relatively noisy heatmaps, independent of the chosen baseline.

We calculated the difference between the Integrated Gradients heatmap and each part and averaged this difference for multiple batches to substantiate these observations. For a zero baseline, with the visually much weaker result, their contribution is almost equal. That is reasonable, as the induced baseline provides no additional information. The *ideal* baseline provides a lot more information and has less difference with the resulting Integrated Gradients heatmap. In our case, the averaged difference is 77 times smaller.

**Ideal Baseline** Based on the experiments, the ideal baseline resembles the ground truth. It provides the maximum amount of information, which is reflected in the method's attribution result. For this setup, the ground truth is generated via targeted changes in latent space, altering high-level features humans, and more importantly, the model would define as relevant. Thus, the invertible architecture facilitates the development of a high-level feature ground truth that the model confirms.



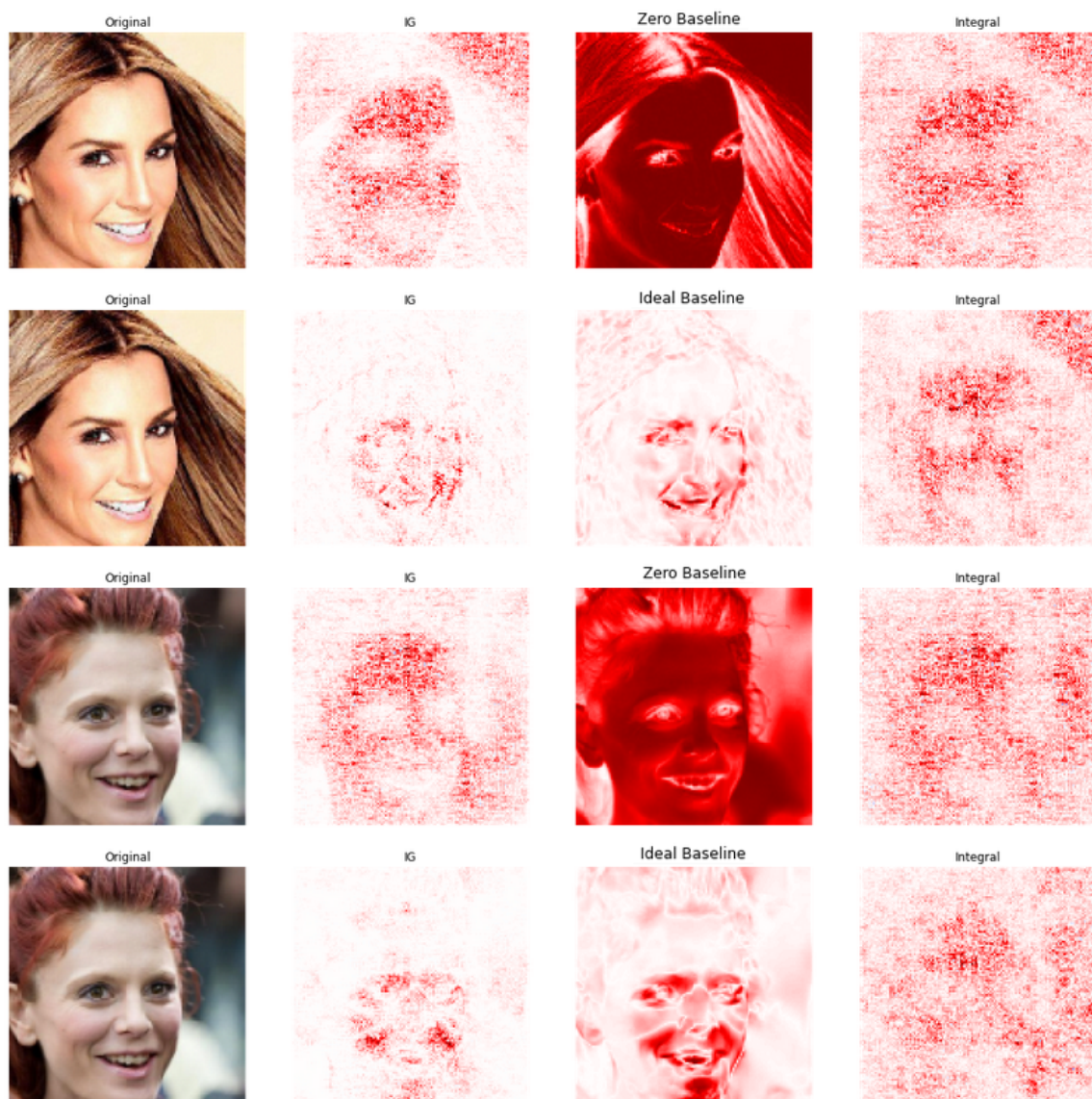


Fig. 4.4 The first column shows the original sample; the second column the complete heatmap; the third column the baseline part  $P_i^B$ ; the fourth column the integral part  $P_i^I$ . The first and the third row received a zero baseline; the second and fourth row the less intensive smiling image.

For most applications, the ideal baseline is not realistic, so a zero baseline is used. The results provide little detail if the baseline provides no information. The integral part features a much broader area, stretching over the entire facial region in our setup. If the background is bright and textured, the integral part of the algorithm assigns positive attribution values for this region. These correlations resemble the defined ground regarding background importance.

Reversely, if the ideal baseline exists, it increases the performance of the method severely. Nonetheless, based on a utopian baseline, it should be possible to create a more powerful attribution method than Integrated Gradients. In this scenario, the integral does not benefit the attribution outcome. However, if some prior information is available, the baseline can improve the performance significantly.

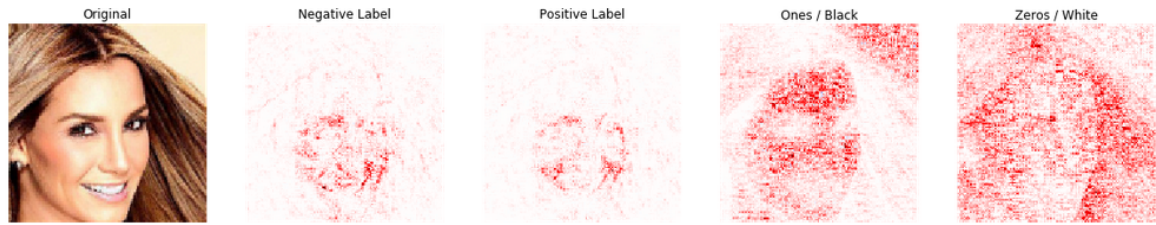


Fig. 4.5 Integrated Gradients Attribution results for target label *"Smiling"*. From left to right: Original; Negative label baseline; Positive label baseline; Black baseline; White baseline

### 4.3 IBA Evaluation

Schulz et al. [2020] motivate their attribution method with a block of the information flow. The idea is realized by introducing random noise to a specific layer, where the noise-level determines the amount of information replaced. Dimensions, which provide more information, are subject to less noise due to optimizing the models' objective. Ideally, irrelevant features are entirely replaced by noise. IBA introduces two hyperparameters to optimize for each application individually, the noise level per feature and the information bottleneck position within the model. The authors present two ways of obtaining the individual bottleneck parameters, i.e. the noise levels. The per-sample bottleneck optimizes the parameters for one sample with defined loss (Eq. 2.10), and the Readout bottleneck trains a separate model to learn the parameters for the entire dataset. Throughout this evaluation, we focus on the optimization for one sample. The second hyperparameter is the information bottleneck's position in the model. For convolutional networks, the authors propose a positioning within the first third of the network to still maintain local information of the input.

In this work, the bottleneck is placed at layer 40 for the 351 layers deep Glow model. As the attribution result is computed at a latent layer, the attribution acquires this layer’s shape. Via bilinear interpolation, the attribution result is projected towards the input size. For images, the channel dimension’s maximum value is selected to represent the heatmap’s brightness level.

This section presents the attribution results of the Per-Sample Bottleneck approach within the invertible setting. We analyze multiple noise levels for various samples and assess them regarding the ground truth. Additionally, we invert these suppressed samples to visualize and evaluate the attribution results in the input space.

### 4.3.1 Attribution Results

The Per-Sample bottleneck optimizes the attribution results for precisely one sample. Therefore, the noise intensity needs to be adjusted for every sample individually and potentially finetuned for different target labels. The noise level refers to the  $\beta$  of equation 2.10, as it regulates the proportion of randomness induced to the model. Figure 4.6 illustrates results for different  $\beta$  levels.

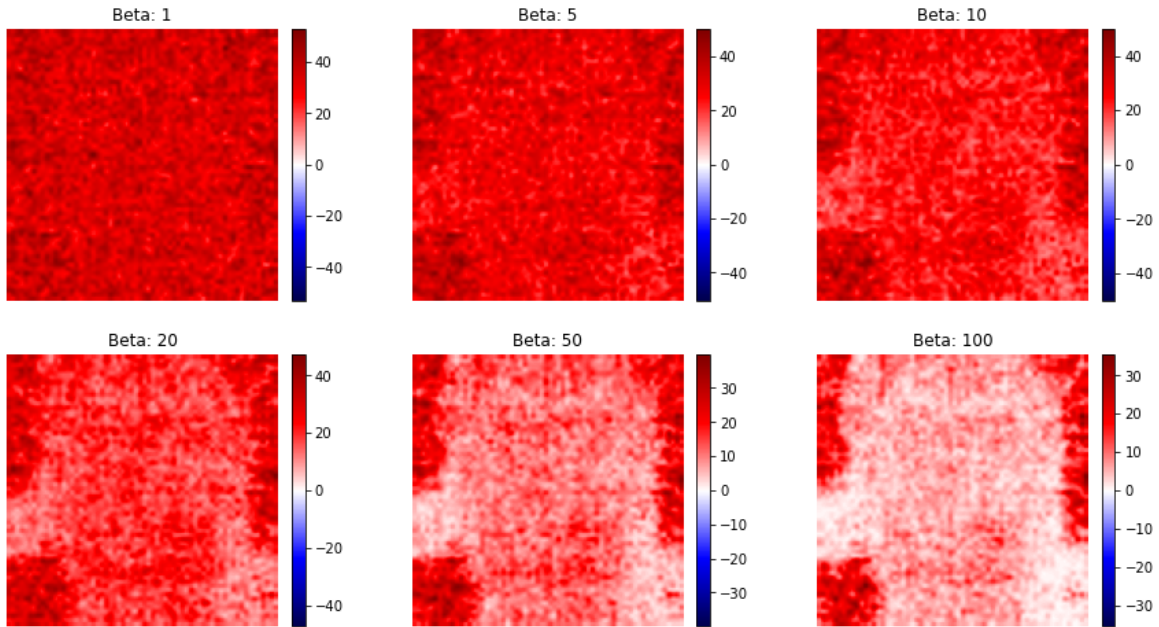


Fig. 4.6 Attribution results for the sample picture of Fig. 4.1 for different  $\beta$  and an increased target for label 31 "Smiling".



The first observation is the attribution result’s noisiness, reflecting the induced noise of the method itself. The blurriness due to the interpolation to the input space enhances the visual effect. Compared to the results of the original paper, our heatmaps appear more fragile. There may be two reasons for this difference. The first is the highly attributed background. This result is no outlier, as the background, especially for lighter pixel values, receives exceptionally high scores for most samples. The second reason is the marginal and overlapping differences for our specific classification task. Nonetheless, the result for  $\beta = 50$ , apart from the background, indicate attribution scores that align with the ground truth. The mouth and nose region are attributed higher and appear less noisy compared to other facial regions. For comparison see [B.6, B.7].

To further examine the sensitivity between particular labels, we set a fixed beta and compare the relevant labels results. By choosing the labels *"Attractiveness"*, *"Smiling"* and *"Glasses"* we aim for visible changes of the attribution results at the specific facial region that corresponds to the ground truth for high-level features. Figure 4.7 illustrates these changes for the fixed  $\beta = 50$ . Neglecting the background, we see low attributions for the woman’s hairstyle, whereas the facial region’s focus shifts per label. Therefore, according to IBA, *"Attractiveness"* corresponds primarily to the background and upper facial region, *"Smiling"* to the background and the lower facial region and *"Glasses"* to the background and the central region around the eyes.

The attribution result for the facial region aligns with the ground truth we defined, albeit its noisiness. This ground truth does not take the background changes into account since, for humans, background and label are unrelated. However, the experiments of Sixt et al. [2021] show a significant change of background for specific label manipulations. These results align with these attributions, indicating that the model relies strongly on the background for its classification. With these experiments, IBA appears accurate with its attributions.

### 4.3.2 Evaluation through Inversion

This section measures the performance of IBA based on the twofold ground truth for the facial and the background region of each image. As Sixt et al. [2021] confirm a correlation between label and background, the ground truth we derived for the facial area is enhanced.

By utilizing the inverse operation of our model, we can visualize induced noise in the input space. Figure 4.8 illustrates inverted samples for different noise intensities. More relevant areas should be less noisy. One might notice that the skinned regions of the face and the background are slightly more identifiable.

To measure attribution consistency for each pixel, we create a batch of the same image and compute its attribution. With IBA inducing random noise, each attribution is unique. We

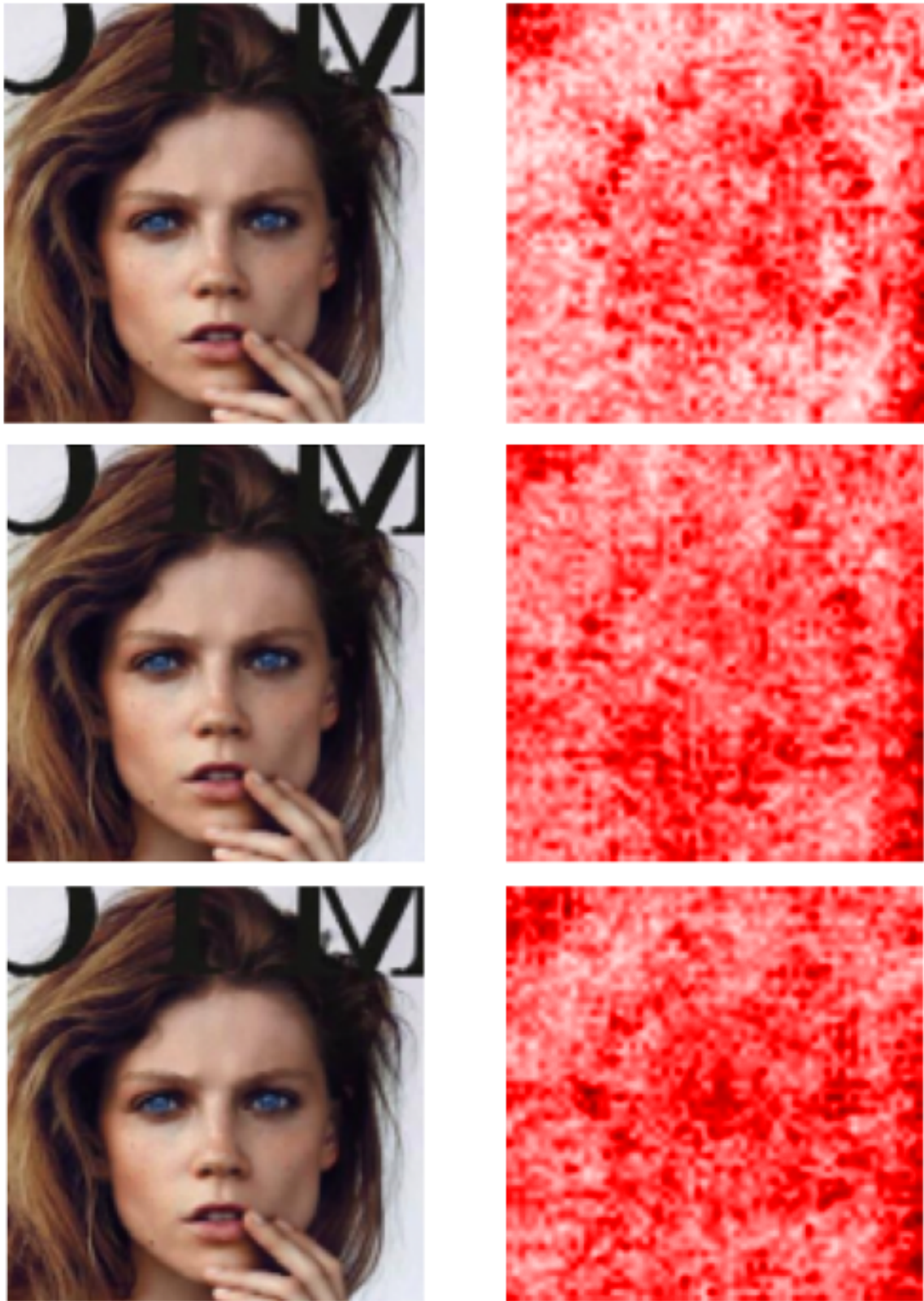


Fig. 4.7 Attribution results for one sample and a fixed  $\beta = 50$ . The right column shows the original, the left column attributes from top to bottom: *"Attractiveness"*, *"Smiling"*, *"Glasses"*.

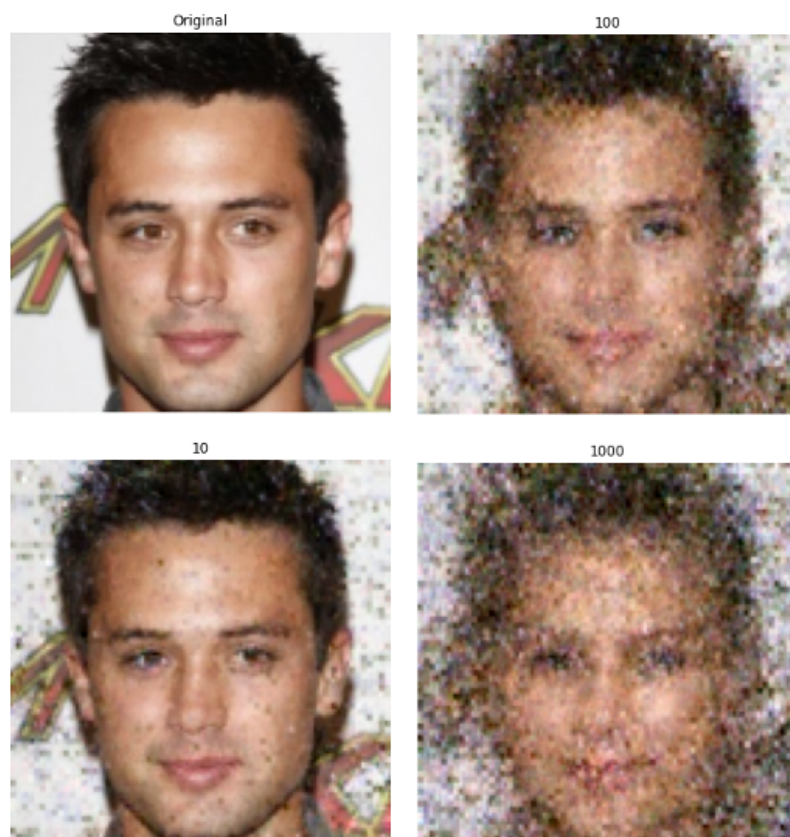


Fig. 4.8 Inverted results after the applied IBA for different  $\beta$  levels. The optimized label is "Attractiveness".

then invert the sample to receive the inverted input with less information. As a sanity check for IBA, we measure the pixel-wise standard deviation over the batch’s unique samples. If a pixel is assigned with a high standard deviation, this pixel varies significantly between each attribution, indicating low confidence. A low fluctuation between within the batch indicates consistency and, therefore, certainty for the attribution. Figure 4.9 demonstrates the result.

**Pixel-wise standard deviation** The deviation is the smallest for highly attributed areas, applying to large parts of the facial region and the background. The texture in the background shows a high variance compared to the monotone background. This seems reasonable, as this specific texture may be interpreted as irrelevant parts of the person’s head. Nonetheless, the standard deviation over multiple samples in the input space provides additional confidence in the attribution and arguably improves the attribution result for invertible models. The bilinear interpolation to reshape the attribution result from the latent layer can be circumvented with the inversion. This step ensures an accurate attribution, as the flow between the input and the latent layer is not approximated through interpolation. The information bottleneck can still be set to any latent layer to address different features.

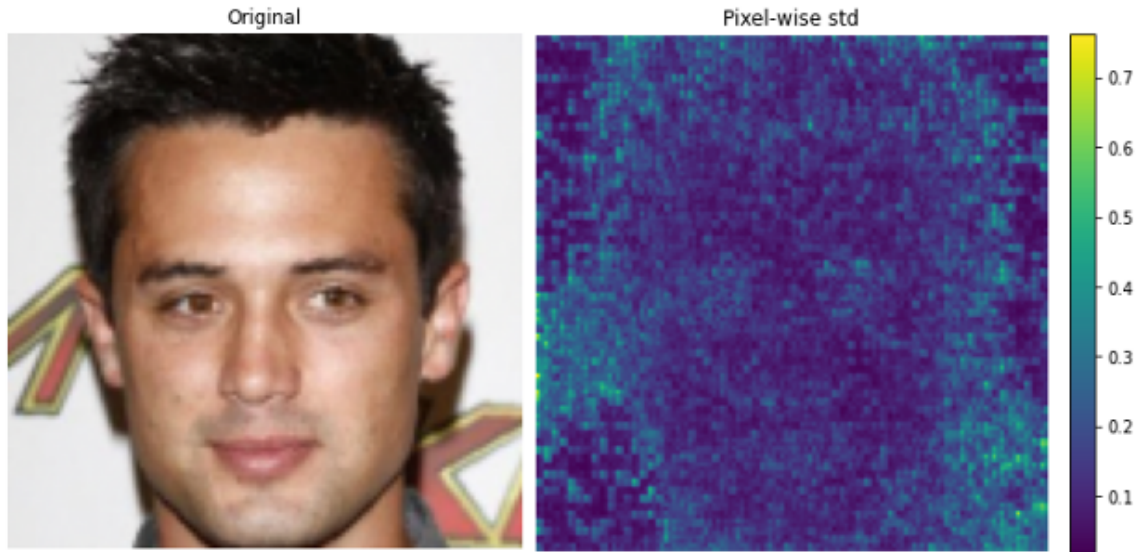


Fig. 4.9 Original and visualization of the pixel-wise standard deviation. The induced noise level before inversion is  $\beta = 50$ .

**Pixel-wise mean** For the last visualization, we compute a Smoothgrad-like mean over one batch. This improves the visibility of relevant features in the input space more clearly. Figure 4.10 presents the result of the pixel-wise mean up to extremely perturbed images. The



averaged images are optimized for the label "*Attractiveness*". For  $\beta = 10^4$ , one observes less perturbed facial regions compared to the background.

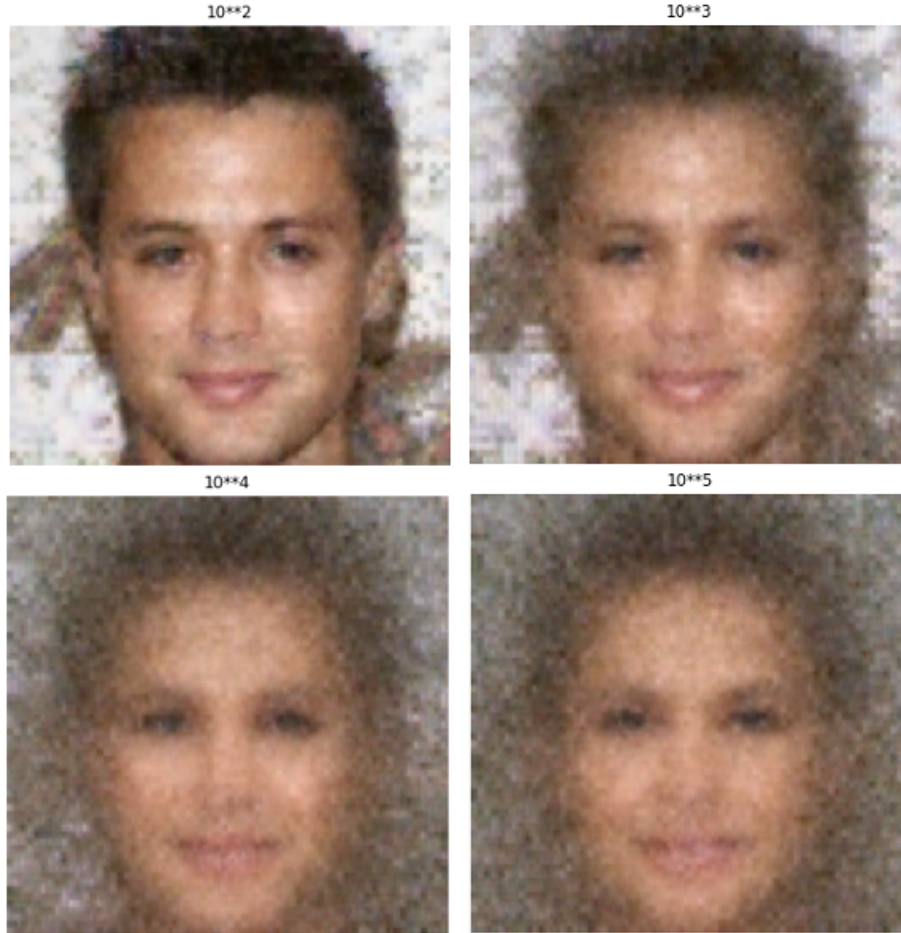


Fig. 4.10 Visualization of the pixel-wise mean for different  $\beta$  levels for the label "*Attractiveness*".

The results of the IBA method align with all sanity checks throughout this line of evaluation. It also measures a high relevance of the background for the CelebA dataset, which extends the ground truth we defined for this setting. This background importance for the same setup is also observed by Sixt et al. [2021].

## 4.4 Summary

We create ground truth for the evaluation setup through invertible models. This inversion property enables targeted manipulation in the latent space, which can be observed in the

input domain, i.e., for image data. Using a simple binary classifier, we flip the logit tensor for specific labels to receive a visible and measurable change in high-level features. Thus, we define a ground truth based on the visual effect a change in latent space has.

Based on this ground truth, we evaluate the attribution results of Integrated Gradients [Sundararajan et al., 2017] and the Information Bottleneck for Attribution [Schulz et al., 2020].

**Integrated Gradients** The evaluation of the Integrated Gradients algorithm is twofold. Firstly, the independence between the input and the respective gradient is examined. Their independence is established with Pearson’s and Spearman’s correlation coefficient, showing neither linear nor rank correlation. This observation enables gradient-based methods to attribute independently from their received input, a prerequisite for an unbiased attribution. The second part inspects the baseline’s impact on the attribution result. We split the algorithm into two parts, obtaining a part influenced by the baseline and the integral part cumulating gradients. By setting the ground truth as a baseline, we show a substantial improvement in the attribution result. When the ideal baseline is unknown, the algorithm still attributes useful features but broader and noisier. Nonetheless, if prior knowledge is available, the Integrated Gradients algorithm benefits significantly from it.

**Information Bottleneck for Attribution** Using the Per-Sample Bottleneck, we optimize the noise intensity manually for specific samples with regard to the ground truth. We also observe large attribution values for the background for an appropriate display of the relevant facial region. This observation is label independent. By inverting the noise-induced image, we receive perturbed images in the input space. The pixel-wise standard deviation for multiple noise inductions of the same sample confirms the high attributions for the image’s background and facial part. A drastic increase of noise and the averaging visualize the central importance of the facial regions for the label *"Attractiveness"*. In contrast, the background loses its information value for the classification. As the ground truth supports the attribution results of IBA, the performance in this setting confirms its quality. Additionally, it reveals the significance of the background for this model.

Generally, the invertible model offers the transparency of assuming a reasonable ground truth for evaluating attribution methods. The following chapter concludes the results of this work.

# Chapter 5

## Conclusion

Attribution methods strive for interpretability, as they try to provide more in-depth insight into both data and model. Still, attribution techniques emphasize varying solutions for the same problem. The discrepancy between these methods raises the question of how to evaluate the results. The lack of ground truth arising from unknown data and model properties makes a sound evaluation of the attribution result difficult. Related work tries to circumvent the missing truth by relating the attribution result to performance of the network [Samek et al., 2015]. While this approach introduces additional assumptions about the heatmap’s properties and the occluding baseline, we derive the ground truth for evaluation.

This work develops a ground truth within two different settings. The first reduces the problem’s complexity significantly to the simplest non-linear problem, the XOR gate. With the introduction of noise, we expand the binary input space to a continuous one, deriving a twofold ground truth. The continuous input space effects ambiguity in the threshold area, initiating an unsolvable problem for input values of 0.5. This event should reflect a high attribution result for the threshold input, as it cancels out the relevance of the second input dimension.

We paired the prior knowledge of the two-dimensional problem with a simple, fully connected model with one hidden layer. By monitoring the attribution map over the entire input space, we could see a strong alignment of the Occlusion method with the ground truth. With the gradient-based algorithm showing the same characteristics, Saliency Maps show the most promising results. However, they remain restricted due to missing gradients for inputs distant from the decision boundary. Input  $\times$  Gradient and Integrated Gradients contradict the defined ground truth, whereas the IBA results resemble the characteristics of the ground truth, albeit its noisiness. The defined ground truth enables examining the attribution and detects weaknesses of gradient-based methods for the two-dimensional problem.

The second setting derives a ground truth based on the chosen network architecture. The utilization of an invertible model enables high-level feature alterations of high-dimensional data by manipulating variables in the latent space. Working on image data, we can flip a specific feature and measure its impact on a sample in input space. The model’s alteration for a specific label is defined as the ground truth for this label.

With this setup, we assign significant relevance to the chosen baseline for the Integrated Gradients method. Without any prior knowledge of the problem, the attribution method still provides a trustworthy result but appears less specific and noisier. Furthermore, we interpret the attribution maps of IBA and extract relevant areas according to the ground truth in the input space.

Both evaluation frameworks demonstrate the weaknesses of the employed attribution methods, which puts their performance into perspective. The two-dimensional setup shows the limitations of pure gradient-based methods, while the tested perturbation-based methods, especially Occlusion, perform better. For the high-dimensional problem, the Integrated Gradients algorithm shows a considerable sensitivity for the choice of baseline. It produces much better attribution maps if prior knowledge is induced via the baseline hyperparameter. Through invertible models, we can extract relevant high-level features and evaluate the attribution methods’ performance based on this ground truth. The noisy results of the IBA attribution resemble this ground truth vaguely.

The evaluation of interpretability increasing methods based on ground truth shows promising results. By opening the Block Box and utilizing known properties both of the model and data, conclusions for the attribution method’s performance can be drawn. Hopefully, this work encourages further evaluation approaches based on known properties of the ground truth. Further work for these frameworks includes examining additional methods, such as LRP [Bach et al., 2015] or DeepLift [Shrikumar et al., 2017]. A synthetic dataset may improve the invertible setup, as so far only the model architecture reveals insight for the ground truth. Well-known data properties might help the specification of the ground truth, and thus, improve the accurate quantification. Additional insight through truth-based evaluations may improve individual methods and nurture new ideas that further align with our information.



# References

- Adebayo, J., Gilmer, J., Goodfellow, I., and Kim, B. (2018a). Local Explanation Methods for Deep Neural Networks Lack Sensitivity to Parameter Values. *arXiv e-prints*, page arXiv:1810.03307.
- Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. (2018b). Sanity Checks for Saliency Maps. *arXiv e-prints*, page arXiv:1810.03292.
- Ancona, M., Ceolini, E., Öztireli, C., and Gross, M. (2018). Towards better understanding of gradient-based attribution methods for Deep Neural Networks. *arXiv e-prints*, page arXiv:1711.06104.
- Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7).
- Dabkowski, P. and Gal, Y. (2017). Real Time Image Saliency for Black Box Classifiers. *arXiv e-prints*, page arXiv:1705.07857.
- Doshi-Velez, F. and Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning. *arXiv e-prints*, page arXiv:1702.08608.
- European Parliament and Council of the European Union (2016). Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 (general data protection regulation).
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- Hume, D. and Millican, P. (2007). *An enquiry concerning human understanding*. Oxford University Press.
- Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., and Sayres, R. (2017). Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). *arXiv e-prints*, page arXiv:1711.11279.
- Kingma, D. P. and Dhariwal, P. (2018). Glow: Generative Flow with Invertible 1x1 Convolutions. *arXiv e-prints*, page arXiv:1807.03039.
- Klambauer, G., Unterthiner, T., Mayr, A., and Hochreiter, S. (2017). Self-Normalizing Neural Networks. *arXiv e-prints*, page arXiv:1706.02515.

- Leshno, M., Lin, V. Y., Pinkus, A., and Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867.
- Lewis, D. (1973). The Journal of Philosophy, Vol. 70, No. 17, Seventieth Annual Meeting of the American Philosophical Association Eastern Division (Oct. 11, 1973), pp. 556-567. *Journal of Philosophy, Inc.* url accessed: 01/02/2021, primary source originally accessed: 23/01/2012.
- Lipton, Z. C. (2017). The Mythos of Model Interpretability. *arXiv e-prints*, page arXiv:1606.03490.
- Miller, T. (2018). Explanation in Artificial Intelligence: Insights from the Social Sciences. *arXiv e-prints*, page arXiv:1706.07269.
- Samek, W., Binder, A., Montavon, G., Bach, S., and Müller, K.-R. (2015). Evaluating the visualization of what a Deep Neural Network has learned. *arXiv e-prints*, page arXiv:1509.06321.
- Schulz, K., Sixt, L., Tombari, F., and Landgraf, T. (2020). Restricting the Flow: Information Bottlenecks for Attribution. *arXiv e-prints*, page arXiv:2001.00396.
- Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A. (2017). Not Just a Black Box: Learning Important Features Through Propagating Activation Differences. *arXiv e-prints*, page arXiv:1605.01713.
- Simonyan, K., Vedaldi, A., and Zisserman, A. (2013). Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *arXiv e-prints*, page arXiv:1312.6034.
- Sixt, L., Schuessler, M., Weiß, P., and Landgraf, T. (2021). Interpretability through invertibility: A deep convolutional network with ideal counterfactuals and isosurfaces.
- Smilkov, D., Thorat, N., Kim, B., Viégas, F., and Wattenberg, M. (2017). SmoothGrad: removing noise by adding noise. *arXiv e-prints*, page arXiv:1706.03825.
- Sundararajan, M., Taly, A., and Yan, Q. (2017). Axiomatic Attribution for Deep Networks. *arXiv e-prints*, page arXiv:1703.01365.
- Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. *arXiv e-prints*, page physics/0004057.
- Yang, M. and Kim, B. (2019). Benchmarking Attribution Methods with Relative Feature Importance. *arXiv e-prints*, page arXiv:1907.09701.
- Zeiler, M. D. and Fergus, R. (2013). Visualizing and Understanding Convolutional Networks. *arXiv e-prints*, page arXiv:1311.2901.

# Appendix A

## Theoretical Background

### A.1 Backpropagation Algorithm

The backpropagation algorithm computes the gradients of the loss function with regard to the individual weights of a model. Therefore, it is applied for supervised learning problems using gradient descent. The partial derivatives are a prerequisite for optimizing the network weights. Using the chain rule of calculus, the algorithm computes each derivative starting from the last layer along the computational graph of the model to the first (Eq. A.1).

$$\frac{\partial z}{\partial x_i} = \sum_j \frac{\partial z}{\partial y_j} \frac{\partial y_j}{\partial x_i} \quad (\text{A.1})$$

### A.2 Activation Functions

**ReLU** The Rectified Linear Unit activation function is defined as the positive part of its argument. See equation A.2.

$$\text{ReLU}(x) = \max(0, x) \quad (\text{A.2})$$

Compared to its non-linear counterparts, it provides several advantages. The main reasons for its popularity in modern neural networks are its efficient computation and the prevention of a vanishing gradient, enabling the training of deep neural networks.

### A.3 Gaussian Probability Distribution

The Gaussian Probability Distribution is a continuous probability distribution for a real-valued random variable  $x$ . Its probability density function can be described with equation A.3.  $\mu$  denotes the mean;  $\sigma$  denotes the standard deviation.

$$f(x|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \times \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \quad (\text{A.3})$$

### A.4 Loss Functions

**Mean Squared Error (MSE)** The MSE quantifies the difference between a label vector  $y$  and the prediction vector  $\hat{y}$  of a modelled function for  $N$  logit values according to equation A.4.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y - \hat{y})^2 \quad (\text{A.4})$$

**Binary Cross Entropy (BCE)** The BCE computes the difference between two binary values according to equation A.5. It is convenient to use as a loss function for multiple classification problems if their labels can be binary.  $y$  denotes the label vector,  $\hat{y}$  the prediction vector, and  $N$  denotes the number of logit values of the output vector.

$$BCE = -\frac{1}{N} \sum_{i=1}^N y \times \log \hat{y} + (1 - y) \times \log(1 - \hat{y}) \quad (\text{A.5})$$

### A.5 Network Optimization

**Gradient Descent** Gradient Descent is an algorithm to optimize a function  $f(x)$  by adapting its parameters  $x$ . In the context of machine learning, the function optimization refers to the minimization of the formulated loss function in order to improve the model's performance. This is achieved by iteratively adjusting the parameters along the steepest descent, indicated by the negative partial derivative for each parameter. The update of each parameter  $x_i$  to the optimal  $x_i^{opt}$  that minimizes the loss function can therefore be computed with equation A.6.  $\epsilon$  denotes the learning rate, a scalar value that defines the step size for each iteration [Goodfellow et al., 2016].

$$x_i^{opt} = x_i - \epsilon \frac{\partial f}{\partial x_i} \quad (\text{A.6})$$

**Stochastic Gradient Descent** The stochastic gradients descent algorithm is mostly applied for training machine learning models. It mostly is infeasible to compute the gradient descent for every data point in the data set at once. Therefore, the stochastic approximation is used by sampling batches (or mini-batches) of data points used to optimize one iteration. If the entire dataset is used for the one optimization step, we refer to them as deterministic algorithms [Goodfellow et al., 2016].



# Appendix B

## Additional Plots

### B.1 Ring of Gaussian

The Ring of Gaussians (ROG) problem expands the XOR gate by multiple classes while still relying on the two-dimensional input space. The setting is motivated by the same prior knowledge approach for evaluating attribution methods and relies on the same ground truth. The motivation for creating the dataset is solely driven by increasing the classification difficulty. For the two-dimensional setup, each class’s mean is located on a circle. (Fig. B.1). This arrangement supports the assumption of equal importance for both input dimensions, as the information of one input value does not suffice for a confident prediction.

The ROG\_net model solving this task is equally structured as XOR\_net to maintaining comparability. Due to multiple classes, the output dimension is enhanced to  $C = 5$ . With the input data’s value range increases to  $D_{ROG} = [-4; 4]$ , the data points are Gaussian sampled with an increased standard deviation  $\sigma = 0.45$ ;  $N \sim (\mu_c, \sigma)$ .

The Ring of Gaussian dataset supplements the XOR problem’s visual evaluation. The enhancement towards a multi-class problem should display the same characteristics for each attribution method, as the input remains two-dimensional. Figure B.2 displays the attribution results of ROG\_net. The output is not binary, and the decision boundaries are not orthogonal to each other. Therefore, no input affects its counterpart as irrelevant, complicating the evaluation compared to the XOR problem. Nonetheless, the results are worth looking into but are not essential for further evaluation.

While the top left plot represents the model’s output, the results for the attribution for each dimension become visually more complex. For the gradient-based algorithms, the attribution results for values on decision boundaries remain high. The difference between both dimensions for this area is visible. Still, for areas distant from these boundaries, the

results also partly differ, contradicting the ground truth (Fig. B.3). Contrary to the XOR case, the perturbation-based Occlusion suggests varying information between both dimensions, especially for inputs closer to the original binary values. The IBA method presents exciting results. For both dimensions, the decision boundaries receive the noisiest attributions, whereas the remaining regions are attributed opposite. For the input value 0.5, each dimension receives high attributions. This characteristic is comparable to the XOR case.

The ground truth is insufficient for this case and needs further assumptions to enable a fair evaluation.

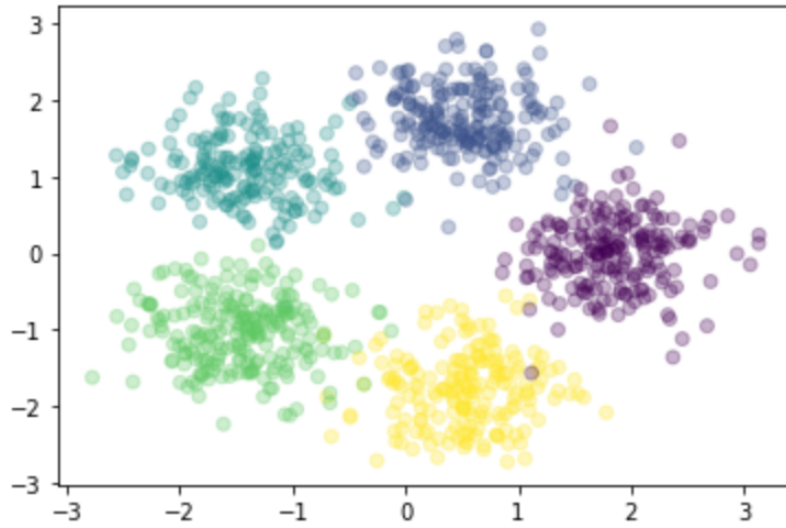


Fig. B.1 Visualization of the Ring of Gaussian dataset with  $n = 1000$  data points; #classes  $c = 5$ ; radius  $r = 1.8$ ; standard deviation  $\sigma = 0.45$ .



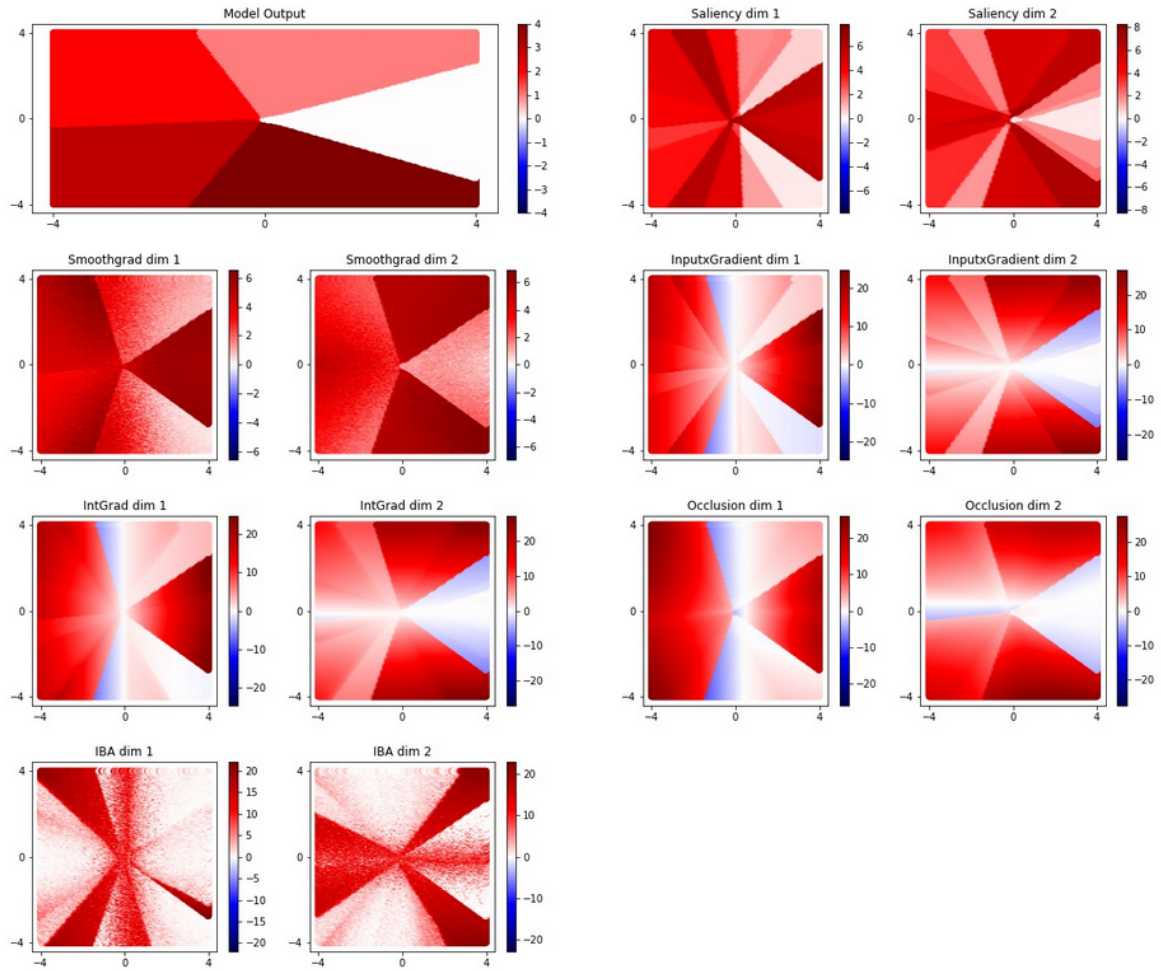


Fig. B.2 Axes represent the two input dimensions  $x_1$  and  $x_2$ . From top left to bottom right: Model output of ROG\_net; Each input dimension's attribution results: Saliency Maps; Smoothgrad; Input  $\times$  Gradient; Integrated Gradients; Occlusion; IBA.

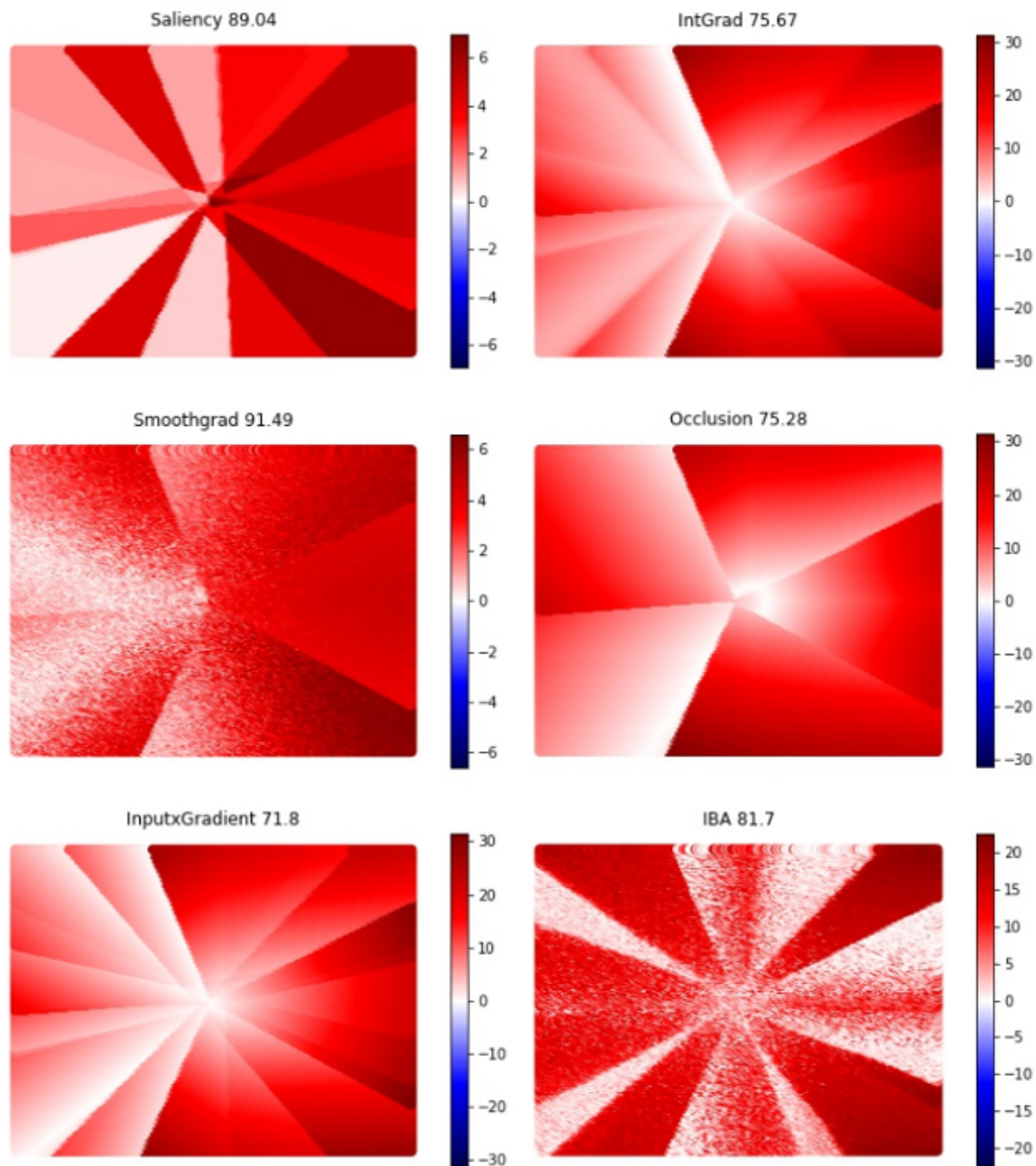


Fig. B.3 Visualization of the difference in attribution values  $A_{diff}$  between both dimensions for the entire input space through ROG\_net. From top left to bottom right: Saliency Maps; Integrated Gradients; Smoothgrad; Occlusion; Input  $\times$  Gradient; IBA. The scalar value above each subplot indicate the summed difference.

## B.2 Inversion Evaluation

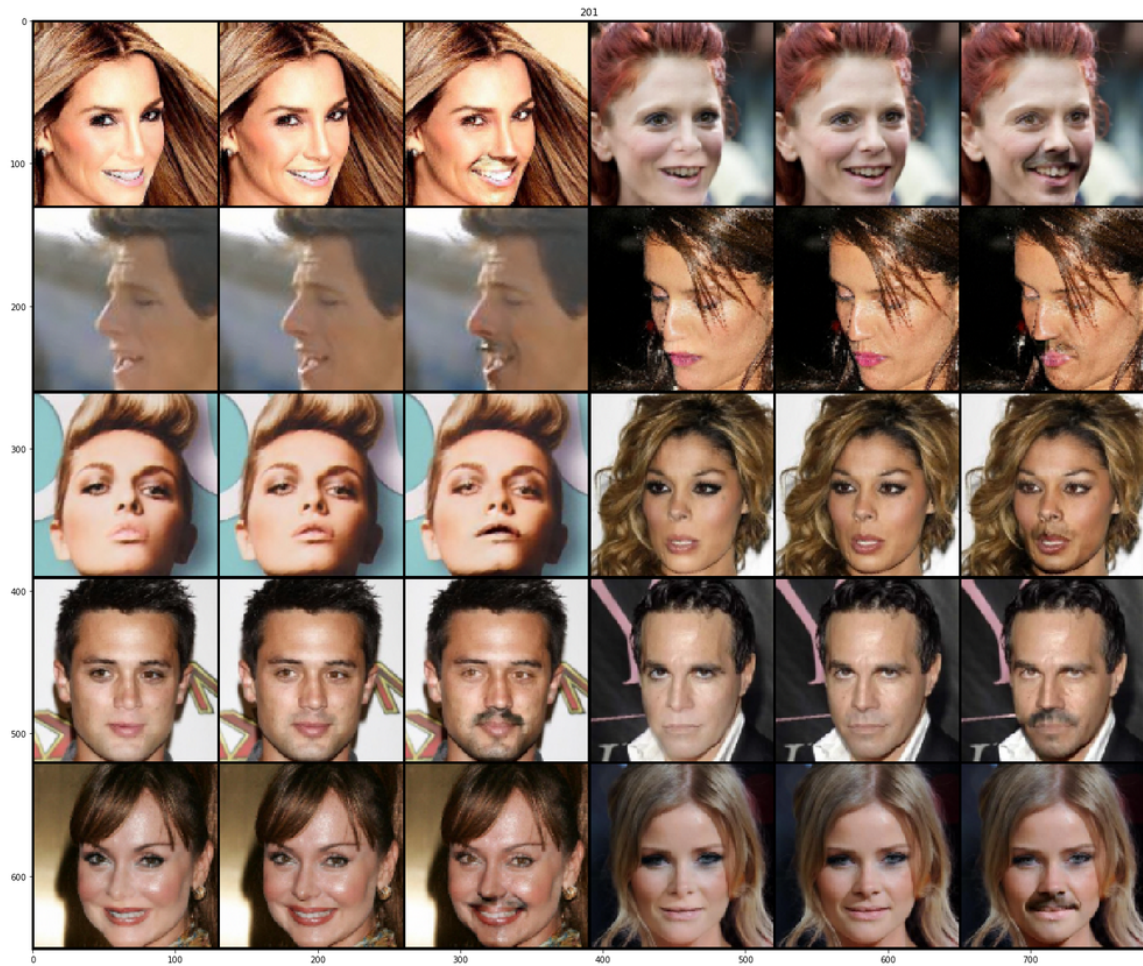


Fig. B.4 Targeted manipulation for the Label *moustashe*. The left picture of each triplet has a small label score. The middle picture is the original. The left picture shows has a small label score.



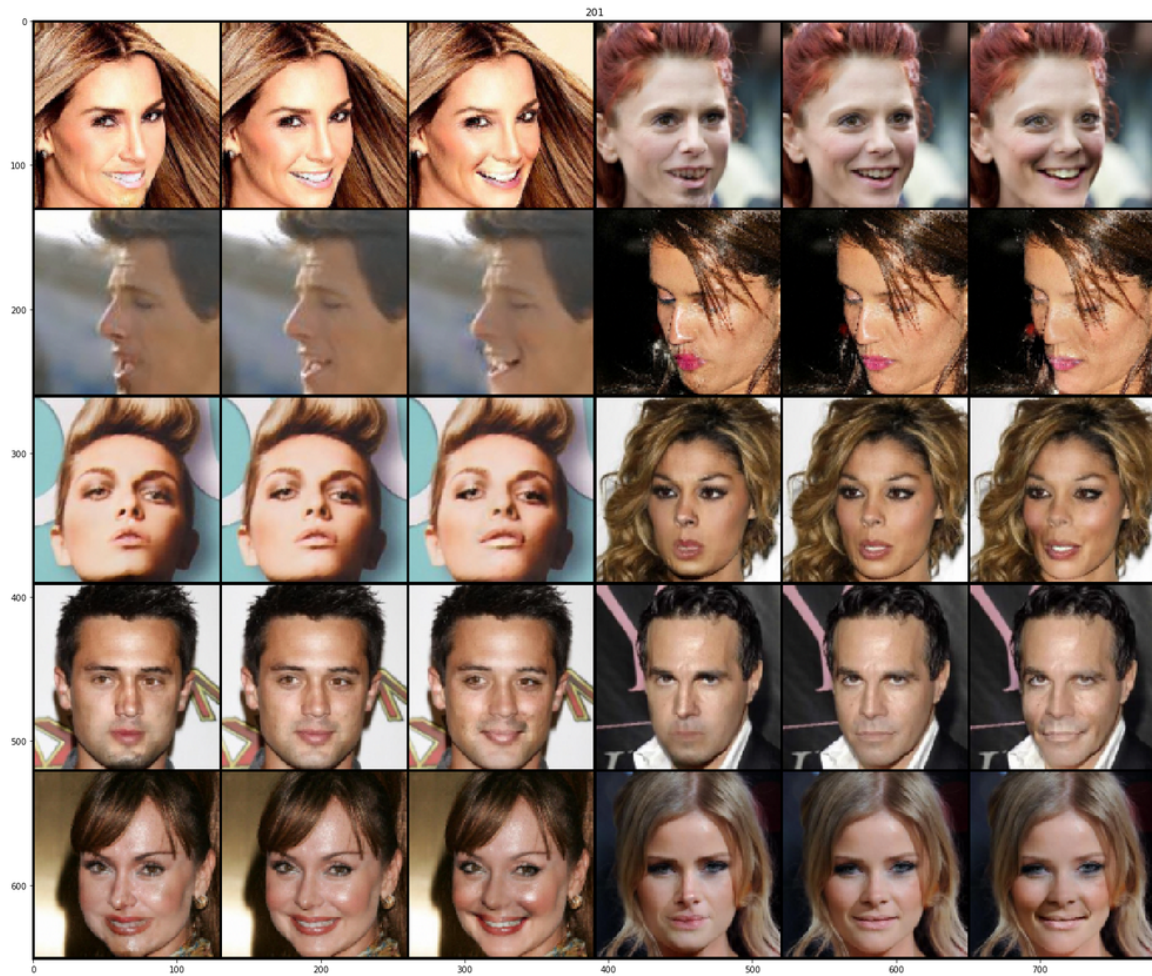


Fig. B.5 Targeted manipulation for the Label *Smiling*. The left picture of each triplet has a small label score. The middle picture is the original. The left picture shows has a small label score.

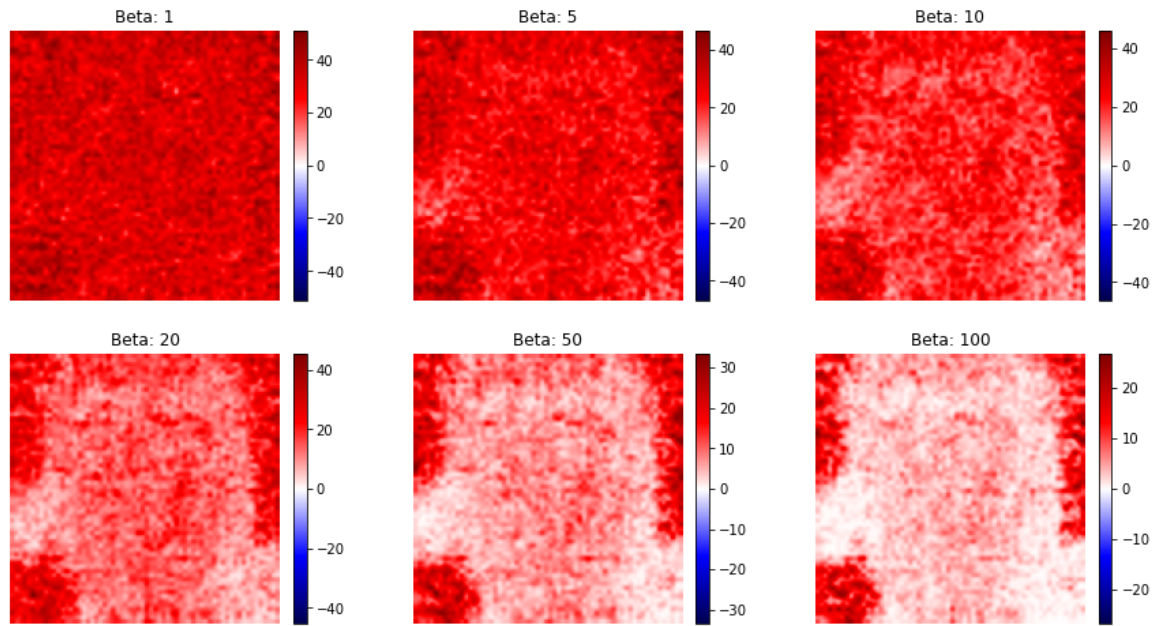


Fig. B.6 Attribution results for the sample picture of Fig. 4.1 for different  $\beta$  and an increased target for label 2 "Attractiveness".

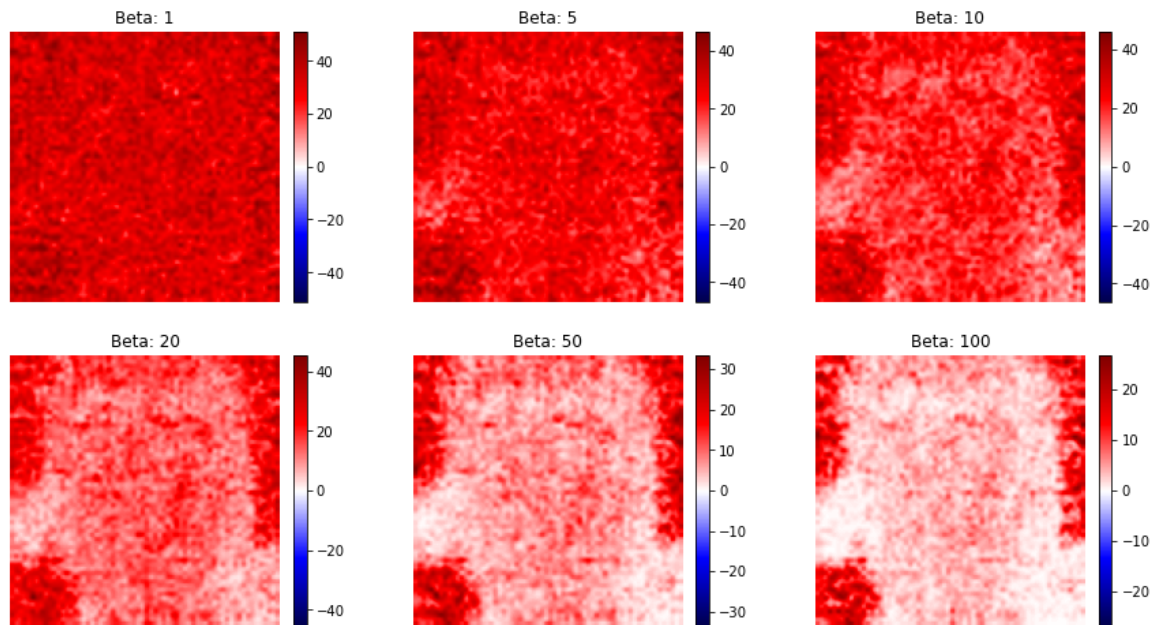


Fig. B.7 Attribution results for the sample picture of Fig. 4.1 for different  $\beta$  and an increased target for label 15 "Glasses".

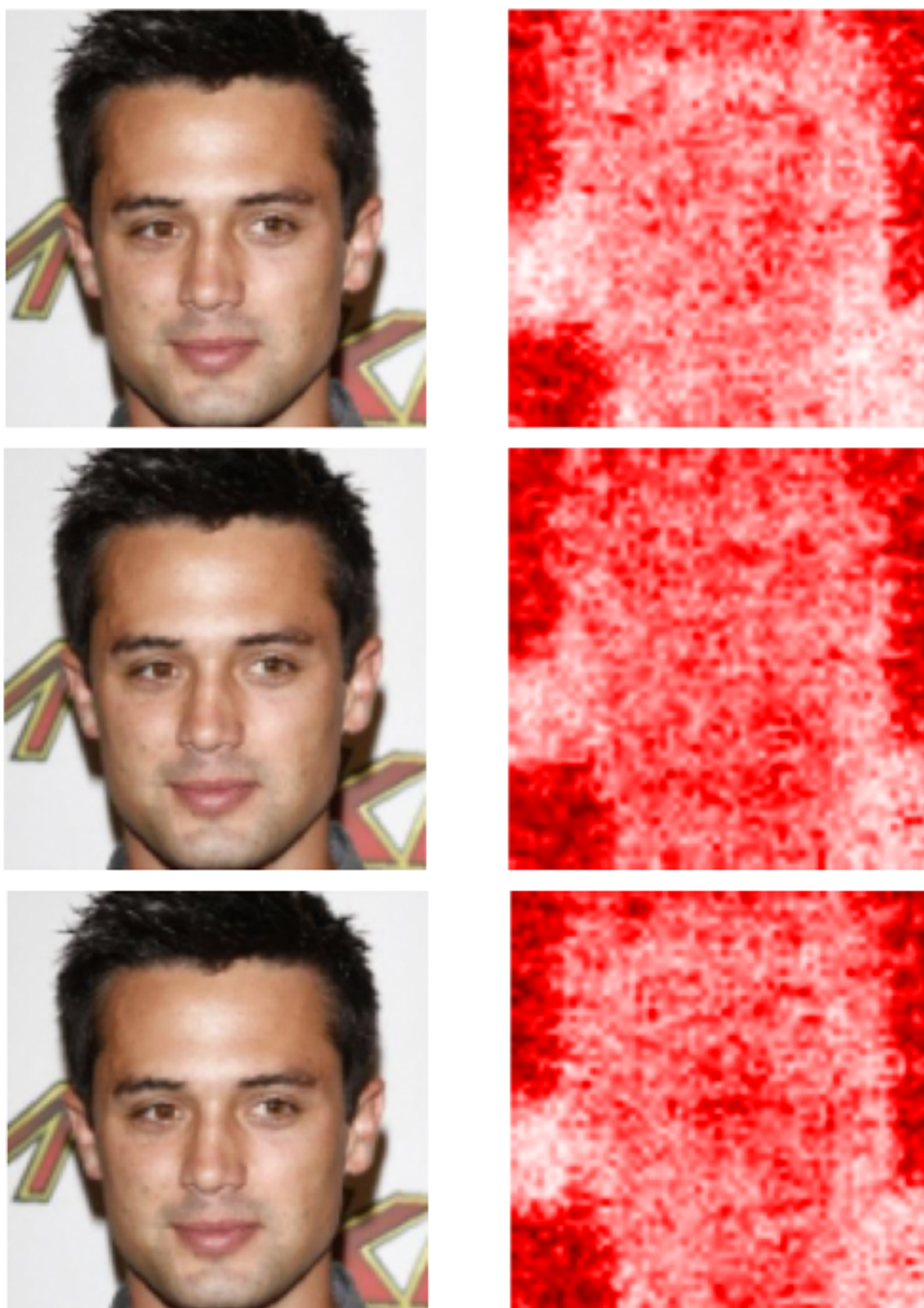


Fig. B.8 Attribution results for one sample and a fixed  $\beta = 50$ . The right column shows the original, the left column attributes from top to bottom: "*Attractiveness*", "*Smiling*", "*Glasses*".