ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
UNIVERSITY OF PIRAEUS

DEMOKRITOS

Ensembling to leverage the interpretability of medical image analysis systems

by

Argyrios Zafeiriou

Submitted
in partial fulfilment of the requirements for the degree of

Master of Artificial Intelligence

at the

UNIVERSITY OF PIRAEUS

June 2022

Author ....

II-MSc “Artificial Intelligence”

June 17, 2022

Certified by..................................

Ilias Maglogiannis
Professor
Thesis Supervisor

Certified by..................................

Theodoros Gianniakopoulos
Principal Researcher
Member of Examination Committee

Certified by..................................

Michael Philippakis
Professor
Member of Examination Committee

Ensembling to leverage the interpretability of medical image analysis systems

By

Argyrios Zafeiriou

Submitted to the II-MSc "Artificial Intelligence" on June 17, 2022, in
partial fulfillment of the
requirements for the MSc degree

Abstract

Modern Artificial Intelligence (AI) systems have been achieving human-level and, in some cases, even higher predictive capabilities, solving numerous and various tasks. Two primary reasons behind this accomplishment are the rapid technological evolution, and the rising volume of available data, both of which allowed the development of multimillion parameter models. Inevitably, along with accuracy, complexity has also risen. But no matter how high the accuracy may be, some tasks, including any medical-related task, require explanations about the model's decision. When dealing with image data, the explanations of the model's decision usually take the form of salient and non-salient areas over the image that highlight the important and non-important areas respectively. Whichever the importance attribution method though, the saliency of an area represents the view of the model towards the stimuli that influenced mostly the outcome and can be as accurate as the quality of the features the model has learned. Thus, a plausible assumption would be that the better predictions the model makes, the more accurate explanations it produces. In this work, the efficacy of ensembling models as a means of leveraging explanations is examined, under the concept that ensemble models are combinatory informed. Apart from ensembling, a novel approach is herein presented for the aggregation of the importance attribution maps, in an attempt to examine an alternative way of combining the different views that several competent models offer. The purpose of aggregating is to lower computation costs, while allowing for the combinations of maps of various origin. Following a saliency map evaluation scheme, four tests are performed over three datasets, two of which are medical image datasets, and one is generic. The results indicate that explainability can, indeed, benefit from the combination of information, either by ensembling or aggregating. Discussion follows, in an attempt to provide insight over the mechanics that led to the provided results, as well as to give guidelines for potential future work.

Thesis Supervisor: Ilias Maglogiannis
Title: Professor

Acknowledgments

I would like to express my gratitude towards the supervisor of my thesis, Professor Ilias Maglogiannis, the members of the examinations Committee, Principal Researcher Theodoros Giannakopoulos and Professor Michael Philippakis, and the PhD candidate Athanasios Kallipolitis for their support towards the completion of my thesis. Special thanks to my family and friends also for their support, as well as copying with me due to spending less time with them during my studies.

This material is based upon work supported by the «Funding Body» Contract No.....

Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the «funding body» or the view of University of Piraeus and Inst. of Informatics and Telecom. of NCSR “Demokritos”.

Contents

List of Figures

List of Tables

1 Introduction

Modern Deep Learning systems have been achieving incredible results in solving tasks in fields like Computer Vision (CV), Natural Language Processing (NLP), Bioinformatics and many more. The complexity of these systems does not allow for direct explanation of their decision-making process and, therefore, the evaluation of its soundness. Oftentimes, the explainability of a competent model is not required nor worth to be concerned of, especially in cases of low-risk decisions, like, for instance, those of a recommendation system. When, on the other hand, the stakes are high, explainability is crucial. Medical AI applications fall into this category, since a medical diagnosis and, consequently, the health of an individual may be affected by the application's decision. Apart from evaluating the soundness of a decision, explainable models can also be used to highlight interesting data patterns and guide either the diagnosis procedure, or even medical research. In other words, for a model to be able to assist a doctor's diagnosis, it must be as transparent, interpretable and explainable as possible. But explainability is not just practically useful and a requirement of the user from the system. Latest conversations extend the importance of explainability as a lawful right of each stakeholder, while the European Commission (EC) steadily the past few years publishes guidelines, such as [1, 2], about the correct development and use of trustworthy AI. Every human that is subject to the decisions of an automated decision-making program, either directly or indirectly, have the right of explanation. This is an evident statement when considering that AI systems have found purpose in matters of the highest significance, such as Health [3] and Law [4], that outrightly affect the subject's life.

In the context of medical image analysis, model explainability has most commonly been expressed through the computation of saliency maps; pixelwise heatmaps of the original image in process that map the model's output to the significance of the image's areas. The challenge of assigning a relevance score to each pixel led to the development of several attribution algorithms, the ability of which to effectively represent the input feature importance has increasingly been examined. The comparison of various proposals of attribution methods indicates that there has been plenty of improvement on the field and most attempts rely on some common, well-established ideas. Occlusion, [5], [6],and gradient back-propagation, [7], [8], [9], [10], [11], based methods are some of the most popular relevance attribution techniques. The strong relations among these algorithms are cross-examined, demonstrated in [12], showing that, while the attribution method itself serves a critical role in the quality of the model's interpretation, the field seems to be moving on the well-examined side of the Explainable Artificial Intelligence (XAI). That being said, and till the next big breakthrough, XAI research can seek advancements on interpretability of AI systems by recruiting alternative methods such as ensembling.

An instinctive and prevalent hypothesis in the area of machine learning is that stronger predictive models have better learnt the underlying important data features. A common way to build such models is the ensembling of several weaker models trained to solve the same problem. Here lies the question, and topic of this thesis as follows: *Does an ensemble model hold more useful knowledge concerning the effect between stimuli and the corresponding predictions than the weaker backbone models*? A second question of equal importance is the following: *Can we rely on the same consensus of ensembling for better interpretability as we did for predictive strength*? While the models that form an ensemble are all trained towards the same goal, the structural differences and the training procedure followed by each of them lead to diverse models that predict also diversely. This diversity is much wanted when aiming for ensembles, since the resulted supplementary effect of combining different points of view for a given task is an appealing tactic for getting more accurate predictions. If the diversity in predictions originates from the learning of different yet informative data patterns, then the ensemble must also hold more information about which data features are the most important.

Apart from using ensembles as a means to combine the interpretability information for a given task, an aggregation formula of the saliency maps is also proposed. For the popular gradient back-propagation based attribution methods to be applied on an ensemble, the ensemble needs to be loaded on the computer’s memory at once. This is not feasible on common computers, especially for large architectures, such as those of modern Convolutional Neural Networks (CNNs), and high image resolutions, such as those captured by modern hardware. On the contrary, aggregation of images is, at least for the proposed aggregation formula, calculated with minimal computational sources. At the same time, aggregation allows for the combinations of saliency maps that do not solely originate from the application of an attribution method to a group of models, which is what ensembling is, but also of any saliency map that constitutes an informative interpretation. Other options to obtain meaningful and diverse saliency maps are the use of different attribution methods and of different parameters for the same attribution method, such as the choice of reference or of the target layer.

An attempt to answer the above questions is presented herein. By developing and applying a saliency map evaluation scheme similar to that used in [13], the quality of the saliency maps that a group of base models and their ensemble produce, as well as their aggregations are compared. In total, four tests are performed using groups of models trained on the ISIC-2019 dataset , for which the models are open published, and the NCT-CRC-HE, and Imagenette, datasets, for which the models are trained for the needs of this work. The results indicate that further examining the use of ensembles for richer feature visualizations is beneficial. Aggregating is equally promising too, suggesting that the already trained models and their respective saliency maps can directly supplement each other without the need for defining computationally demanding ensembles. Finally, discussion over the results to draw conclusions and make further observations is presented. Interesting future work capabilities are also being discussed.

2 Background

In this section, the basic past and current advances in the fields that this work is concerned of are briefly reviewed. Starting from the more general and moving on to the specifics, definitions, methods, and any aspect that is relevant to my thesis are introduced.

2.1 Interpretability and explainability

Defining *interpretability* and *explainability*, contrary to what one may expect to when dealing with Science, Technology, Engineering and Math (STEM)-related definitions, is not a straightforward and strict procedure. Authors, across and within disciplines, define and use these two aspects of modern AI with diversity. Furthermore, some differentiate between the two terms, [14, 15], while others use them interchangeably. A widely adopted but instinctive definition for interpretability is the one given by Miller in [16], *"interpretability is the degree to which an observer can understand the cause of a decision"*.

A possible reason for this phenomenon could be the subjectiveness of the matter, or what do we expect when asking for an interpretable, explainable and transparent system. In [17], Lipton recognizes some of the "*desiderata of interpretability research*". *Trust*, meaning confidence for letting the model decide without further supervision, and *causality*, meaning that the decision and its cause can be clearly associated, are two of the common requests from an interpretable model. For instance, when referring to a healthcare-related system, *trust* is useful, not in the sense of giving control of someone's health to a program, but in that of confidently rule out false negatives while maintaining high predictive power. *Causality* on the other hand could point promising directions to medical researchers by unveiling links between probable causes and target conditions. Suppose a linear model that through its coefficients positively links low carbohydrate diets to lower intensity outbreaks of an inflammatory disease. This model is a widely accepted interpretable model, that proposes a cause, the low carbohydrate consumption, for a characteristic of interest, the intensity of a disease's symptoms. Note here that causality does not coincide with statistical correlation, not even for this seemingly simple example. Models of such low complexity are also often referred as transparent and the inner mechanics can be observed. *Transparency* is yet another term closely related to explaining an AI system and the most “notorious” non-transparent systems are, of course, neural networks. Since it is not the purpose of this thesis to try and define interpretability, the above instinctive definition of interpretability suffices and thereby the terms interpretation and explanation are used interchangeably, as long as the context allows for clear meanings.

2.2 Interpretability methods - Attribution maps

Having clarified what exactly we need from an explanation method, it is time to choose the right explanation approach. Due to the rapid and parallel advances of XAI and ML, the right choice is not always that obvious. Extended review works, such as [18, 19, 20, 21, 15, 22] , tried to create a taxonomy around explainability methods and, by rightly approaching interpretability from different angles, several taxonomies have been described.

Based on the characteristics of the system, it is a senseful first step to classify a machine learning algorithm as either transparent or black-box. The interpretability methods that aim to explain either of these classes of models, are differentiated respectively. When developing a system whose interpretability is based on algorithm transparency, the available choices are limited and with simplicity as a key characteristic. That is, model transparency trades competency for interpretability. Even ensembling the much instinctive and explainable decision trees can lead to complex enough to be considered black-box models, a problem recently attempted to be addressed in [23]. On the other hand, state-of-the-art performances are achieved by Deep Neural Networks (DNNs), while modern image analysis systems are vastly dictated by CNNs. In contrast with transparent models, black-box models are developed with competency instead of interpretability as the main goal, and the search for interpretations becomes a post model training procedure. Thus, the class of black-box models rely on the so-called post hoc explanation techniques.

Local versus global interpretation methods is another duality among interpretability methods. Methods of a global scope aim to explain the overall behavior of a model. Which input feature patterns are determinant for the model's decisions and in what way? The previous linear model example offers global-scope interpretability, which is often achieved by coefficient analysis and for which a guide of correct use is presented by [24]. Local-scoped methods on the other hand seek explanations for a single decision datapoint. Into this category fall the much popular importance attribution algorithms like [10, 25, 6, 7], which aim to attribute an importance score for each input feature of a single data instance.

The explanation of a DNN often takes the form of an attribution map. Attribution maps or saliency maps refer to the mapping of the input features to importance scores. The more important an input feature is for the decision of the model, the higher its attributed importance score should be. As a solution to this problem, several algorithms have been proposed the past few years and can be separated into two major categories. Occlusion-based algorithms, such as [5, 6], operate by manipulating the input to a model and then observe any changes to its output. These types of algorithms, even though they act directly to the features and thus are model-agnostic and flexible, suffer from impracticability due to computational bottleneck. Each occlusion is followed by a full forward pass through the DNN and, considering the high resolution of many modern medical image data, the total computation time is significant. What is more, they are not completely reliable when capturing the nonlinear effects of multiple features occlusion, while occlusion itself is suspect of introducing out-of-distribution objects to the input. As an alternative to feature occlusion, gradient-based methods rely on the learned gradients of the model and compute the saliency map generally effectively with a single back-propagation. On the downsides of this category of algorithms, we have traded off both model-agnosticism and method's outcome to model's output variation relation.

2.2.1 Gradient-based attribution methods

Integrated Gradient

In [11], Sundararajan et al. proposed Integrated Gradients (IG) along with two axioms, namely *sensitivity* and *implementation invariance*, on which their algorithm is based on. *Sensitivity* is satisfied if for any data feature that leads to variations of the outputs of a ML model when everything else is held constant, the attribution method recognizes this feature as important. *Implementation invariance* is satisfied if for any two ML models that generate the same outputs for every input, the attributions are identical irrespectively of the specifics of the models. To achieve *sensitivity*, IG uses a, most of the times chosen to be zero, baseline reference, relatively to which the attribution of the input is calculated. Formally, for a DNN $f\colon I \rightarrow \mathbb{R}^n$, an input and baseline $x, x^o \in I$, and outputs in $\mathbb{R}^n$, the space of real-valued n-vectors, the $IG_i(x)$ along the i-th dimension of the input is defined as:

Equation 1. Intergrated Gradients calculation formula.

$$IG_i(x) = (x - x^0) \int_{a=0}^{1} \frac{\partial f(x^0 + a(x - x^0)}{\partial x_i} \partial a$$

When computing IG, the integral is substituted by a sum. What Eq. 1 describes is the average gradients when x varies over x^o along a linear path created by a moving on the unit interval. A notable property of IG is that the sum of the total attribution equals to the difference of the target output and the baseline output. This property, named as completeness by IG's authors, is considered desirable by other algorithms too, and is defined soon after in the context of the DeepLIFT algorithm referred as summation-to-delta.

Layer-wise Relevance Propagation

Layer-wise Relevance Propagation (LRP) proceeds, as the name suggests, in a layer-by-layer manner, and with a backward pass distributes attributions through the network. The attribution is redistributed from each to layer to the previous and is equal to the activation of the target node *t*, while the relevance of all other target layer's nodes is set to be zero. The flow of relevance is dictated by a recursive rule, several of which are proposed by the author of LRP in [10]. ϵ-LRP is based on the rule defined in Eq. 2.

Equation 2. ϵ LRP's redistribution formula.

$$R_i^{(l)} = \sum_j \frac{z_{ij}}{\sum_k \left(z_{jk} + b_j\right) + \epsilon \cdot sign(\sum \ \left(z_{jk} + b_j\right))} r_j^{(l+1)}$$

Here $R_i^{(l)}$ is the relevance of the i-th node of layer l, $z_{ij} = w_{ji}^{(l+1,l)} x_i^{(l)}$ is the weighted activation of a neuron x_i onto the j-th neuron in the next layer and b_j

the additive bias of unit *j*. Figure 1 taken from [26] depicts the flow of relevance in between layers along with some common distribution rules.

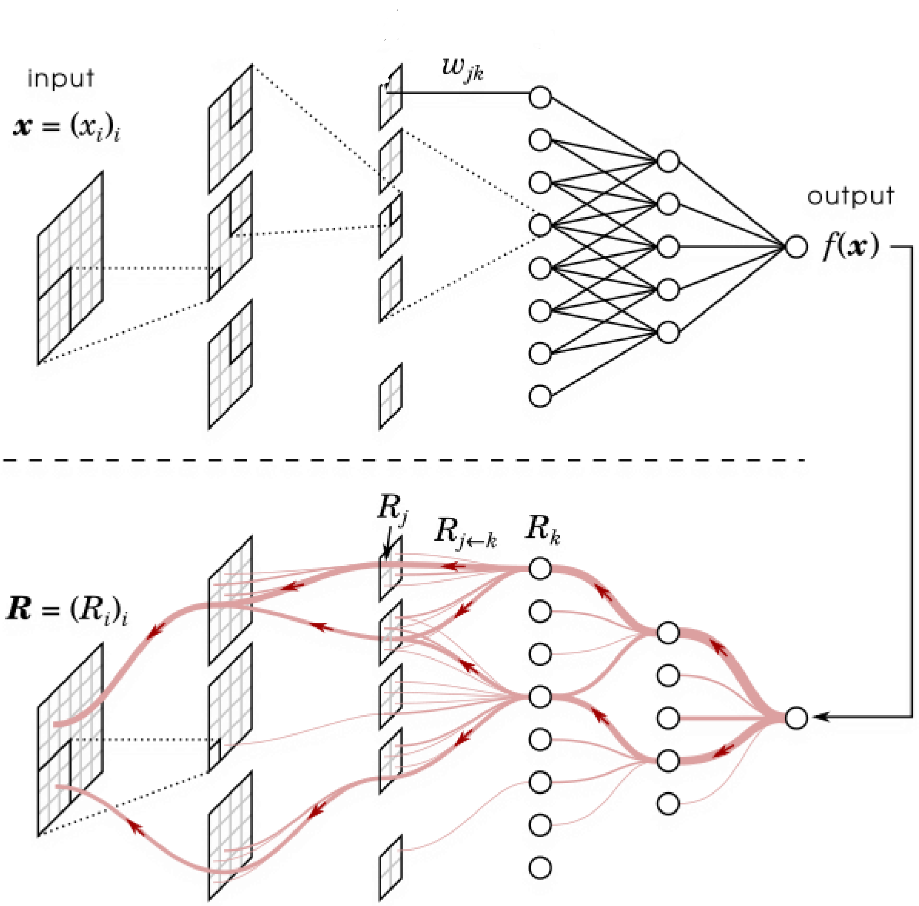

Figure 1. The flow of relevance inbetween layers using LRP.

DeepLIFT

Since DeepLIFT is the method used throughout this work, the description that follows is more detailed.

Difference-from-reference and summation-to-delta

Given an input *x*, for instance an image, and a DNN $f\colon I \rightarrow \mathbb{R}^n$, where *I* is the space of the input data and $\mathbb{R}^n$ the space of real-valued *n*-vectors, DeepLIFT aims to assign contribution scores Cx_i to each DNN's intermediate neuron $x_i \in \{x_1, \ldots, x_m\}$, .Neurons $\{x_1, \ldots, x_m\}$ are necessary and sufficient to compute the target neuron's output $t \in f(x)$. The computed contributions scores are relevant to the activation of the same neurons when the input is a reference input x_0 resulting to the output t_0. We are interested in the quantity $\Delta t = t - t_0$, or difference-from-reference, while $C_{x_i \Delta t}$ indicates that the contribution C_{x_i} refers to Δt. The difference Δt is also set equal to the total amount of contribution attributed to the difference $\Delta x = x\text{-}x_o$, formally the summation-to-delta property:

Equation 3. Summation-to-delta property.

$$\sum_{i=1}^{m} C_{x_i \Delta t} = \Delta t$$

By assigning contributions relative to a baseline, DeepLIFT handles two common limitations of many saliency methods, both of whom are caused by the nature of the DNN's gradients. First, zero gradient $\partial x_i / \partial t$ for a neuron x_i does not imply that its contribution for the difference Δt, $C_{x_i \Delta t}$, will also be zero, as opposed to reference absent algorithms. This is an important feature since a zero gradient neuron might still signal important information. Second, the difference-from-reference eliminates biases from the importance attribution, avoiding the creation of artifacts due to gradient discontinuities. The authors of the algorithm elaborate and give indicative examples of these two problems.

Propagating contributions

Multipliers

The flow of information from the output neuron back to the input is dictated by a layer-by layer step algorithm aspired by the chain-rule. First, authors of DeepLIFT define the multipliers $m_{\Delta x_i \Delta t} = C_{x_i \Delta t} / \Delta t$ for any input neuron x_i. It is an analogy of the partial derivative $\partial x_i / \partial t$, over of course larger differences than those the derivative describes. Given the values for multipliers of layer $X = \{x_1, \ldots, x_k\}$ and its immediate successor layer $Y = \{y_1, \ldots, y_l\}$, the multipliers can be propagated using the equation:

Equation 4. Multiplier's chain rule.

$$m_{\Delta x_i \Delta t} = \sum_{j} m_{\Delta x_i \Delta y_j} m_{\Delta y_j \Delta t}$$

Note how Eq.4 satisfies the summation-to-delta property of Eq 3.

The Linear rule

Multipliers are used to transfer contributions from layer to layer, but the actual contributions are calculated depending on the function each layer performs. For linear functions, such as convolutions and dense layers, the Linear rule applies.

An instance of a linear function is the mean averaging of the inputs, just like when voting individual model's outputs. Voting is used throughout this work when referring to ensembling, so it is useful to describe the flow of contribution when applying DeepLIFT and a mean averaging layer is the final classifier of the model. Assuming $|M|$ models, their outputs $X = \{x_1,\dots, x_{|M|}\}$, where $x_i \in \mathbb{R}^n$, then $y = \frac{1}{|M|}\sum_{i=1}^{|M|} x_i$ is the voting layer. Starting from the target output node $t \in y$, the Linear rule states that $\Delta t = \Delta y = \frac{1}{|M|}\sum_{i=1}^{|M|} \Delta_{x_i}$ and, consequently, $C_{\Delta_{x_i}\Delta_y} = \frac{1}{|M|}\Delta_{x_i}$ and $m_{\Delta_{x_i}\Delta_y} = \frac{1}{|M|}$. What is interesting at this contribution propagation step, is the fact that each model assigns contribution relative to its difference-from-reference Δ_{x_i}, which is actually the difference-from-reference of the jointly predicted output node. So, models that classified the input as something different from the ensemble's prediction, will also experience little to zero difference-from-reference and, therefore, attribute little to no contribution amount when back-propagating. This observation will come at hand when discussing the results of this work's tests on 5.1 paragraph.

The Rescale and Reveal-Cancel rules

For nonlinear functions that accept a single input, authors present two attribution rules, the Rescale and the Reveal-Cancel rule. The main difference among these rules is how they address positive and negative attributions. The first does not discriminate between the two, while the latter does. There are pros and cons about choosing either of them, which the authors describe. At the time of writing, *Captum* only supports the Rescale rule, and it is how DeepLIFT will handle nonlinear layers for this work. Nevertheless, there are no indications that the Reveal-Cancel rule would be more appropriate for any of the models involved.

GradCAM

Ancona et al described in [12] the links between several of the most popular gradient-based attribution algorithms, and how some of them coincide under certain parameter choices. For instance, ϵ-LRP is equivalent to DeepLIFT [7] with a zero reference if no nonlinear function maps zero to zero. Here reference is a neutral baseline for the input and its use is one more discriminative feature of gradient-based attribution methods. But algorithms whose mechanics vary significantly do exist. GradCAM, proposed in [8], does not rely on redistributing fixed sized attributions through back-propagating, the difference-from-reference concept, nor hard-coded axioms. By hypothesizing that the last convolutional layers of a CNN have large perceptive field and having focused on semantically important for the classification image features, GradCAM aims to detect what these layers have learnt for each class. Most commonly the last convolutional layer is chosen. Formally, if *L* is the last convolutional layer of the model *f*, then the output of *L* are *k* feature maps of a common dimensionality $A^l \in \mathbb{R}^{HxW}$, where *l*∈*{1, ..., k}* and $\mathbb{R}^{HxW}$ is the space of *HxK* matrices. The importance of the contents of each feature map is calculated relatively to the target class's node *t*, but before the use of a SoftMax function. For each map, a weight is calculated as in Eq 5.

Equation 5. GradCAM's weighting function of the feature maps.

$$w_t^k = \frac{1}{HxW} \sum_i \sum_j \frac{\partial t}{\partial A_{ij}^k}$$

Here the double summation represents a global average pooling operation. The averaging of the feature maps using the weights w_t^k results in an attribution representation of dimension *HxK,* which is finally subjected to an element-wise ReLU operation, a function that zeros out all negative attributions. The definition of GradCAM, apart from significantly distinguishing the algorithm from the previously described algorithms, imposes an applicability constraint. The demand for the CNN's last part to be comprised of a convolutional layer followed by a linear classifier limits the available choices of models, especially when opting for an ensemble.

2.3 Attribution maps evaluation

The evaluation of the predictive strength of a model is straightforward. Given the ground truth, the predictions of the model and the purpose for which a model is developed, we can define and compute one or more appropriate metrics. For instance, for a model developed towards detecting cancerous tissue from image data, eliminating false negatives is of greater significance compared to false positives. Assuming that the outputs are either positive or negative predictions, for presence or absence of cancer respectively, the model must demonstrate high sensitivity. These metrics are universal and objective.

For image data, mapping every input feature *x_i* to a saliency score *$S(x_i)$* is the same as attributing a value to each image pixel. Thus *$S(x_i)$* is injective, and the output result has the same dimension as the input image allowing the explanation to be visualized. This property allows for direct observation and qualitative evaluation of the attribution method's output. Domain experts examine and rate the quality of the output attributions maps or set the ground truth for relevant datasets, mostly in the context of salient object detection [27]. This type of evaluation suffers from important limitations. Firstly, having a subjective ground truth and, even more, a subjective metric is problematic. Though, by consulting a well sampled set of experts could soothe the problem of subjectiveness, the evaluation remains unquantified, and a second problem is exaggerated, that of data volume. For domains like medical imaging, which are both delicate and in need of high expertise, recruiting sets of experts for labeling thousands of datapoints is not feasible.

Driven by these limitations, a handful of quantitative metrics have been proposed. Manipulating the input data to observe the output score variation is of the most popular among evaluation schemes and are designed to quantify the *faithfulness* of the method, that is, whether the highlighted by the saliency map features are relevant to the model's reasoning. Chattopadhyay et al [25], apart from advising 'human subjects', proposed Average Drop (AD) and Increase in Confidence (IIC). These two metrics are supposed to act in a complementary fashion. AD is the percentage of drop of the model's prediction score after gradually removing the important areas that are designated by the class-discrimination map. This map is computed by the algorithm multiplied by the original image, essentially the important area pointed out by the method. IIC, on the contrary, results by removing the unimportant areas of the image. A limitation of these two metrics is the granularity upon they are measured. Removing the highlighted as important areas all at once, especially when using masks to do so, introduces artifacts in the image whose effect on the inference of the model is not tracked. Similarly, Petsiuk et al [28] introduced Deletion Area Under Curve (DAUC) and Integration Area Under Curve (IAUC). The main difference of AD, IIC and DAUC, IAUC is that the latter are computed by progressively masking relevant or unveiling irrelevant areas of the input image instead of directly doing so. The order in which the masking/unveiling takes place is that of the saliency values. For example, for images, the most important pixels will be masked or the least important will be unveiled first. This procedure leads to a drop curve and the area below that curve is the respective score. Although the progressive feature removal is still suspect of introducing artifacts, it offers an evaluation of higher resolution, since DAUC and IAUC scores are calculated for various levels of perturbation. The aggressive perturbation nature of masking is not tackled in this work though. A more general progressive procedure of manipulating the input image is described in [13]. Instead of masking, i.e., zeroing out pixels and areas of the image, a general perturbation function can be defined, such as blurring or replacing with uniformly or normally generated noise. Once again, a drop curve is formulated, and the authors choose to quantify the drop as the Area Over the Perturbation Curve (AOPC). In contrast with DAUC, for which a lower score is desirable, a higher AOPC score is better.

Just like occlusion-based attribution techniques, any change to the original data might introduce out-of-distribution data and, consequently, unreliable measurements. Thus, the method of data perturbation plays a key role for the reliability of the evaluation procedure and special precautions should be taken when considering the perturbation function. Simply masking image areas does not allow for such considerations. In this work, the attribution-evaluation scheme of [13] is used with special consideration in avoiding these problems. AOPC score is also adopted to quantify the evaluation results. More on section 4.3.

2.4 Combining models

Combining a number of models trained to solve a common problem is a popular and highly effective ML technique to build a stronger predictive system, also known as ensemble. Ensembling can be performed in many different ways, but the core condition for the ensemble in order to leverage accuracy, is for its backbone models to be also accurate and diverse. Diversity when ensembling is of crucial significance, since two classifiers or regressors that have learned the exact same patterns will also jointly perform the same. In other words, the models need to make mistakes on different datapoints for their conjuction to be better informed than they are as a unit.

Diversity can be introduced in many ways, especially for the much complex DNNs, and can originate from the data, the models themselves or the training procedure. Ways of introducing model diversity have been studied in the literature. Lakshminarayanan et al [29] propose training the same model using different random initializations, which suffices for the models to converge to different solutions. This method of differentiation is temperate considering that modern succesful models are being proposed constantly, and whose architectural differences can be extensive even when focusing on a particular category. For image data, CNNs have rightfully gained popularity the past years. Depth, width, activation functions, all affect greatly on which features does the CNN focus and are present to every architecture, from AlexNet [30] to the most "exotic" ones. Between two identical model, diversity can be introduced through the training hyperparameters, such the loss function, which dictates what the model values the most during its training and the learning rate, which guides every next step of the learning.

Modern software has made it easy to recruit various model architectures and also pretrained on vast datasets, a method known as Transfer Learning (TL). TL, for which a comprehensive review is presented in [31], aims to transfer knowledge gained while learning about another vision task to the problem of interest, ensuring lower training times. With tools like Pytorch [32] and TL, training models that are structurally different provides diversity with certainty. EfficientNets [33] is a family of CNNs that have monopolized the last ISIC challenges, achieving state-of-the-art results. The key contribution behind these modern nets is the development of AutoML [34], which allows for automated size regulation of the model so as to achieve efficiency along with accuracy. EfficientNets, just as every other modern model, have built upon earlier popular architectures. ResNets [35] family, which exploited the benefits of higher depths for accuracy, is some of them. The increase in depth is achieved by adding identity layers in-between layers. The idea behind ResNets was so successful that influenced numerous other architectures. ResNexts [36] introduced *cardinality* into the ResNets by adding multiple bottleneck blocks in place of a single block. One step further, ResNests [37] also added squeeze-and-excitation (SE) [38] blocks into the previous advancement. SE is a gating mechanism that is lightweight enough to be incorporated in large models and results in an *attention* effect when learning. While ResNets aimed for depth, InceptionNet [39] focused on wider layers. On the other hand, DenseNets [40] are inspired, as the name suggest, by dense connections. Every layer of the model is input for every other subsequent layer, not just the next one. Variations of these architectures are utilized in this work and in 4.2 the exact configuration are presented.

2.5 Combining saliency maps

In the context of XAI, ensembling has not been examined extensively, but works that indicate that ensembling is a useful methodology for leveraging interpretability do exist. Authors in [41] conclude that different model architectures, as well as different attribution methods, focus on different important data patterns of the image. In [42], the work most closely related to the topic of my thesis, Kallipolitis et al by consulting seasoned physicians for comparing the explanations the ensemble and its backbone models produce, concludes that indeed ensembling offers explainability benefits, just like it does for accuracy. Most of the work is oriented towards saliency detection and semantic segmentation tasks, [27, 43]. Even though saliency detection is an aspect closely related to interpretability, the differences are more than significant. Saliency detection models are trained and evaluated using as targets masks that separate the image in salient and non-salient pixels and, so, their outputs are monochrome, black and white, images and their evaluation is straightforward, expressed by metrics such as F1 and MAE scores. For these reasons, ensembling saliency detection models is based on leveraging accuracy, not interpretability, and is not related with the anti-causal reasoning of the classification process on which interpretations aim for.

What is most interesting about the work done on saliency detection, is the efforts of aggregating the output of the models as a means of improving the acquired saliency representation. Lots of aggregation methods have been proposed and tested. For instance, [44] proposes a standard pixel-level aggregation of the saliency maps, while [45], steps on that standard aggregation and tries to capture and formulate into the aggregation the neighboring relationships among the pixels. All of these efforts indicate that saliency map aggregations can be fruitful.

3 Methods

In this section, any methods and algorithms used throughout this work are discussed. Their formal description is complemented by examples and justification about their selection. First, a high-level description of the experimentation pipeline is given and right after the specifics of each step.

3.1 Experiment pipeline

The compared objects of interest are sets of saliency maps. The saliency maps produced by a set of models trained on a common dataset comprise the baseline saliency maps. The term baseline here arises from the fact that these maps are produced by directly applying an established attribution algorithm on the models and data at hand. The saliency maps of their ensemble, which are referred as the ensembled saliency maps, are obtained by defining the ensemble of the original models and then applying the attribution method. The ensembling strategy followed is the mean averaging of the outputs of the backbone models, also known as voting. A third set of saliency maps is created by aggregating the baseline saliency maps using a novel aggregation formula described in 3.3.2. Finally, as a baseline aggregation, the mean average of the baseline saliency maps comprise the mean averaged aggregated saliency maps set. Each and every saliency map is calculated by DeepLIFT, and the parameters described in 3.2.

The assessment of importance for each saliency map is based on the scheme used in [13]. The values of the saliency map define an order over the regions of the image based on their saliency, from the more to the least important areas. Following that order, the image is gradually perturbed by replacing the region with gaussian noise. The model that produced the saliency map inferences using the perturbed image as input. The more accurately the saliency map highlights the important regions for the model, the larger and more abrupt drop for the originally predicted class score is expected. For each saliency map set, the evaluation results are mean averaged across models and dataset, resulting in a single drop curve for the baseline, ensembled, aggregated and mean average aggregated saliency maps. Figure 2 represents a high-level description of the experimentation pipeline. The specifics of this pipeline are described in detail in the rest of this section, while experiment related details are given in section 4.

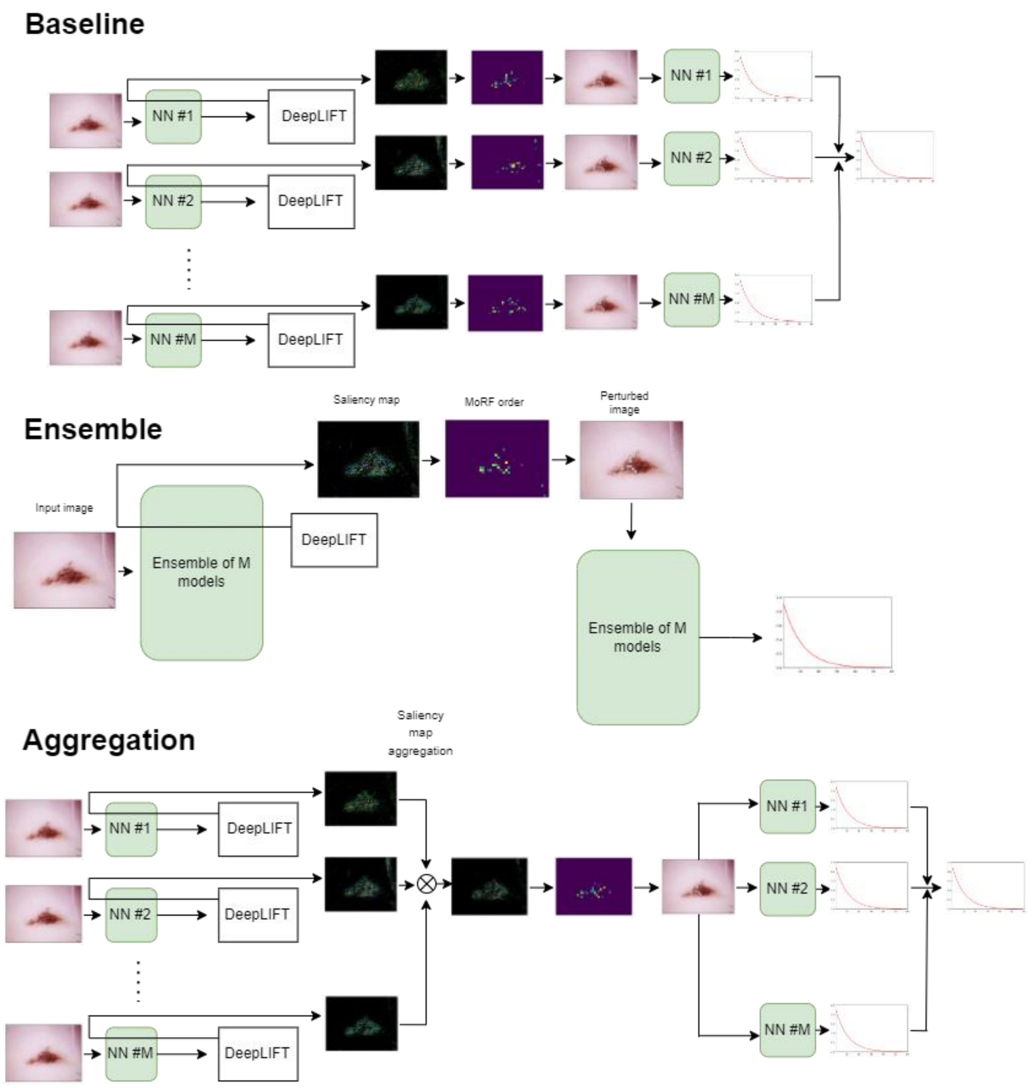

Figure 2. The experimentation pipeline followed to compare the quality of ensembled and aggregated saliency maps relative to their backbone maps.

3.2 Use of DeepLIFT in this work

Why DeepLIFT

Practicability and performance are two things to ask from an attribution method. DeepLIFT is a backpropagation-based algorithm, meaning that it assigns relevance scores by contributing importance from an output neuron all the way back to the input image. The computation of the algorithm’s output needs a single backward pass and is efficient enough for the purposes of this thesis, since DeepLIFT is called thousands of times. Apart from computation efficiency, architectural compatibility is also a strong prerequisite for easiness of use, especially when opting for diverse ensembles. Although not a model-agnostic algorithm, DeepLIFT is well implemented for a variety of CNN models. The *Captum* [46] library is used and for the models described in 4.2 only some minor modifications are needed. More specifically, *ReLU* instances must be uniquely used throughout the network.

As per the performance requirement, evaluating saliency maps and comparing attribution methods is not a straightforward procedure as discussed in 2.3. Nevertheless, there does exist a reason for preferring DeepLIFT over other methods. Adebayo et al [47] proposed a sanity check for attribution methods, which is based on the senseful expectation that if the model parameters are randomized, the saliency map must also change. In [48], authors expanded Adebayo’s sanity experiment and performed sanity check for several attribution algorithms. To support our choice, only DeepLIFT passed the check.

Choice of parameters

The choice of the reference image and the target layer of the saliency visualization are important parameters for the effectiveness of the attribution method. For choosing the target layer, in the case of applying DeepLIFT on a SoftMax’s preceding layer, DeepLIFT’s proposers suggest using an extra normalization step. Furthermore, the effect of regularizing the output scores to a mass of one for all models and all instances that SoftMax has, is also useful for the aggregation of the saliency maps as discussed in 3.3.2. For every saliency map computed in this thesis, the target layer is the post SoftMax output of the model.

The choice of the reference image when computing DeepLIFT is not obvious or irrelevant to the problem at hand. The authors of the algorithm point out the importance of the reference for DeepLIFT to capture meaningful information. Completely black images as references are a baseline approach, which is also the default for other reference-based algorithms. Other options are to use either a noisy or a blurred version of the input image. A reason for using a reference in the first place, is to simulate the absence of the image's features. While a totally black reference may be suitable for images like those of MNIST, analogy used in [7], where the background is actually black and the features white, for real world datasets a blurred image is considered to be a safer reference choice. Another reason for choosing a generally senseful reference and not trying to achieve better results by searching for more special options, is to keep the same reference while testing across several datasets. That way, any observed replicability is more confidently attributed to the hypothesis under examination rather to other parameters. As the reference, a blurred version of the image is used. Specifically, the image is convoluted with a large (53, 53) gaussian kernel, with a standard deviation of 30 for both axes. This filtering results in a highly blurred version of the input image, which must have preserved none of the discriminative features originally present. Figure 3 showcases an example of the heatmap that DeepLIFT produces. The third image depicts which regions of the image the heatmap highlights the most and it is a concept discussed later on in detail.

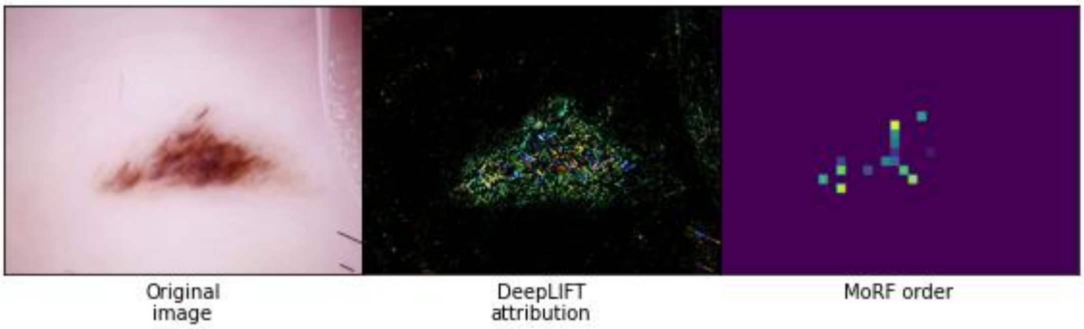

Figure 3. Example of DeepLIFT attributions map and the first 15 elements of the corresponding *MoRF* order.

3.3 Combining several models for interpretability

The main purpose of this thesis is to examine whether an ensemble model that is stronger than its backbone models when predicting, has also learnt better the important features of the images. Furthermore, an aggregation formula is proposed showing that the saliency maps of the backbone models hold more information when combined.

3.3.1 Ensembling

Given a set of models M trained on data of the same distribution, an ensemble is any decision-making combination of the models of M. A simple and common way to ensemble the models is the mean averaging of their decisions. This method, due to its simplicity, adds minimal bias to the results of the explanation method performed on the ensemble. By bias, I mean any unknown influence the ensembling specifics could introduce to the attribution assignment. Optimal ways of ensembling as per the performance of the explanation methods could be further studied. For all tests performed on this work, ensembling refers to the mean averaging of the backbone model's outputs.

For obvious efficiency reasons, the inference of an ensemble model is done in two steps, the inference of each one of the backbone models and then the averaging of those outputs. However, when computing the saliency maps produced by the ensemble, a structured network definition is necessary for the DeepLIFT algorithm to be performed. In that case, an extra combining layer, as the one showcased in Figure 4, is added after each model, whose only purpose is to average its inputs. Loading several models on memory and performing a back-propagation pass for DeepLIFT's needs, or any other gradient-based explanation method, is the main computational bottleneck of ensembling for explainability. In Figure 4, all classifiers output a vector of a common dimension, which is equal to number of target classes. The unifying layer performs a mean averaging operation that constitutes the ensemble's decision.

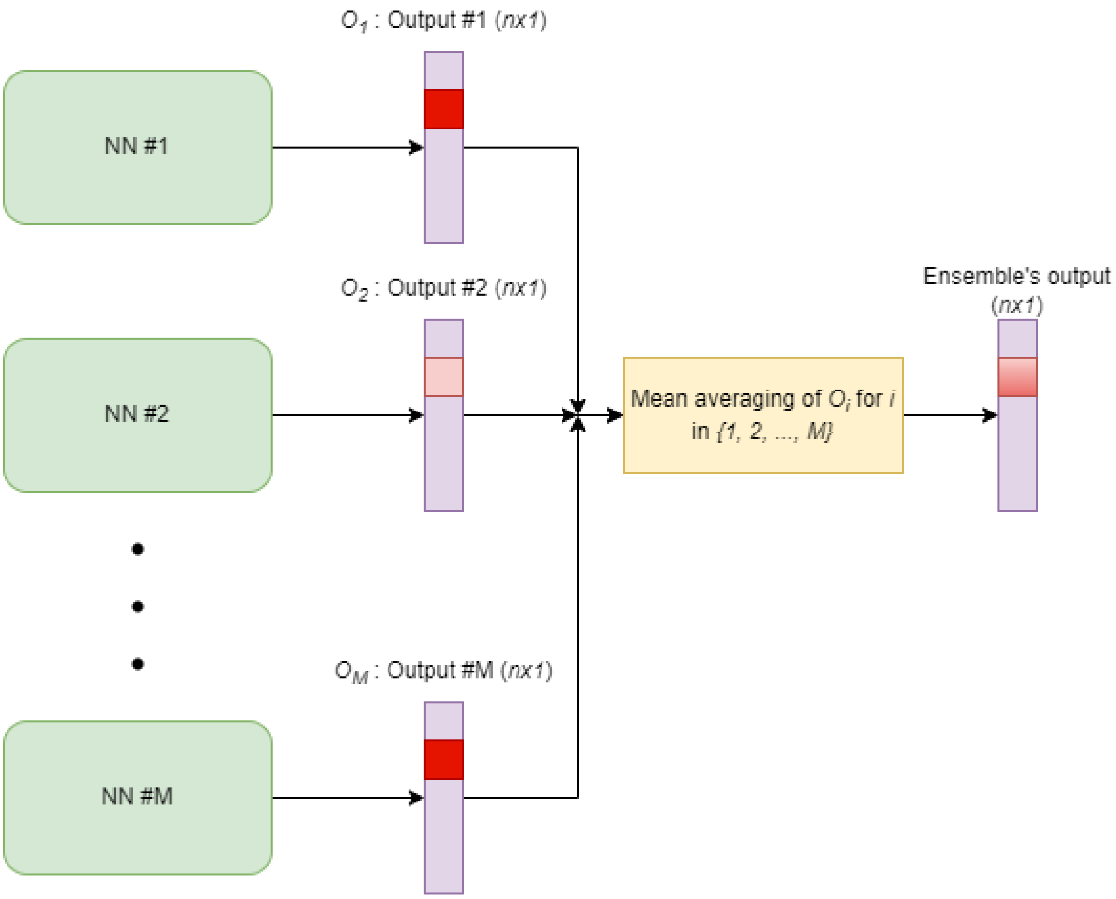
O_1 : Output #1 (*nx1*)
NN #1
O_2 : Output #2 (*nx1*)
NN #2
O_M : Output #M (*nx1*)
NN #M
Mean averaging of O_i for *i* in *{1, 2, ..., M}*
Ensemble's output (*nx1*)

Figure 4. The voting classifier layer of the ensembles. For *n* classes, the *n*-vector outputs O_i, where *i*∈*{1,...,M}*, of the *M* backbone models are mean averaged and the ensemble's output results.

3.3.2 Aggregation

The aggregation is performed on the saliency maps obtained by applying DeepLIFT on the models of *M*. The |*M*| saliency maps are initially filtered by the target output node based on which they had been computed, and only the maps whose generator model belongs to the majority group when grouped by prediction are taken under consideration when aggregating. In other words, the saliency map of a model is used in the aggregation if and only if the model predicted what most of the other model predicted too. The whole aggregation process is an unsupervised procedure since it is not expected for the ground truth to be always known. Verified ground truth assumes that the user, for instance the doctor, has already made a confident diagnosis. Nevertheless, this filtering directs the interpretation procedure to the class most likely to be true, but this is not the main motivation behind filtering. The saliency maps of different target nodes highlight features with totally different interpretation, that of disparate classes, and aggregating these saliency maps would produce controversial explanations.

For every model $f \in M^*$, where $M^* \subseteq M$, the majority of models that agree on their prediction, and for every region $g(k, l) \in x$, where x the image of interest and $g(k, l)$ the region around the (k, l) pixel of x, a weight w is computed by the following formula:

Equation 6. The proposed weighting of each area and model.

$$w(g(k,l))_f = \sum_f \left[t(g(k,l))_f \ \times f(x) - \frac{EMD_f(x)}{\sum_{g \neq f} EMD_g(x)} \right]$$

Where, t is the uniform ordering of the image's areas in the unit interval as ordered by $MoRF$,

Equation 7. The importance of the area for a given saliency map.

$$t(g(k,l))_f = 1 - \arg\left(MoRF(x)\right)_f / \left|MoRF(x)_f\right|$$

and $EMD_f(x)$ is the mean average of the Wasserstein metric [49] scores between the outputs of f and every other model of M^*. The valid use of the Wasserstein metric assumes a common metric space and mass among the two distributions. This is always true if every model's last layer is a SoftMax layer. Finally, if the weight is calculated to be negative for some model and image area, is set equal to zero.

Generally, an informative saliency map should be dense as per its attributions, meaning that the highlighted as important areas should explicitly stand out. With that in mind, the above aggregation aims to cancel as much noise as possible. On an image level, $f(x)$ and $EMD_f(x)$ reward prediction confidence and punish output distribution divergence, respectively. Both parameters should have noise reduction effects. Assigning higher weights to more confident models, straightforwardly reduces the share of the more divided and, therefore, more likely to focus on false pixels models. The divergence from distribution penalty has little to no effect for a large number of aggregated saliency maps, as it quantifies divergence of a model divided by the sum of divergencies of the rest of the models. The latter is a large number for every model. This behavior is depicted in Figure 5.

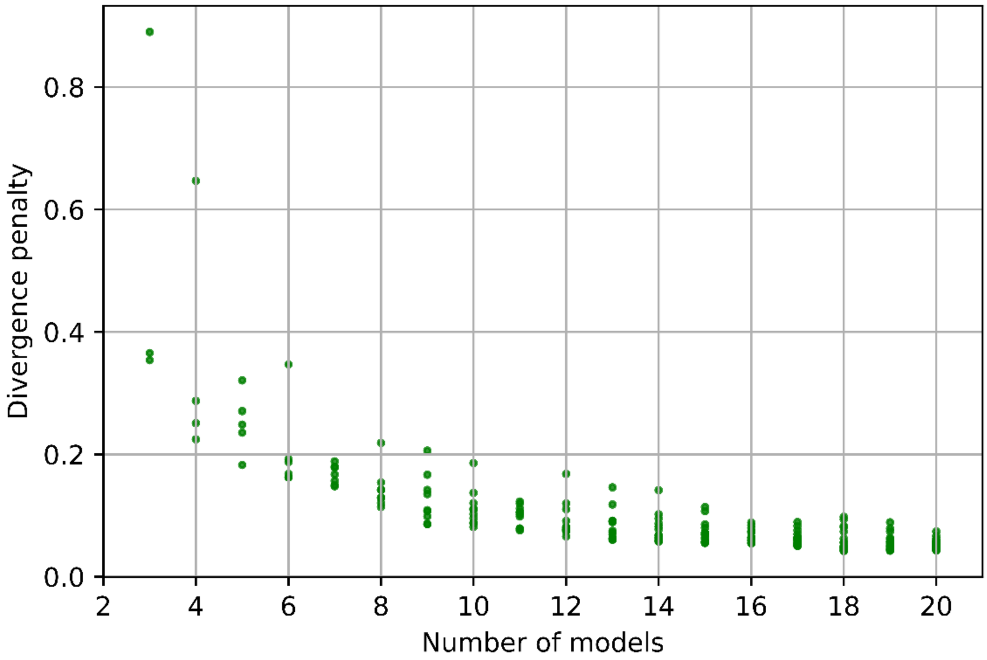

Figure 5. The behavior of the divergence penalty in comparison with the number of involved models.

For a smaller number of models though, which is what is usually true in most cases, the penalty becomes significant for highly divergent distributions. despite its symmetricity and its linear position in the sum of the aggregation. Figure 5 showcases how the divergence penalty behaves with respect to the number of involved models. The data for Figure 5 are artificially created. More precisely, 5-vectors are randomly created in groups of 3 and up to 20 elements, corresponding to ensembles with the respective number of backbone models. Each vector represents the prediction of a 'model' that outputs 5 class probabilities. As per the randomness, the only constraint is that the position of the maximum element of all vectors is common, as if the hypothetical models predict the same hypothetical class but with different probability distribution. For each number of models, any outlier datapoint corresponds to the divergence penalty of a divergent model. Finally, note that as a divergence metric the Kullback-Leibler (KL) divergence [50] has also been considered, but the Wasserstein metric was preferred due to its behavior when comparing distributions of separate support. For datasets like ImageNet [51], where the number of classes is large, it is possible for the outputs of two models to not share a common support. In that case, the KL divergence of these outputs is not real valued but infinity, in contrast with the Wasserstein metric.

On an area level, $t\big(g(k,l)\big)_f$ is the leading importance attributor among the areas of the image. If someone were to aggregate saliency maps originating from the same model but, for instance, computed by different attribution methods, then $f(x)$ and $EMD_f(x)$ would serve no purpose when assigning weights for the aggregation. In that case, $t\big(g(k,l)\big)_f$ would still produce meaningful results, as by definition leverages what the saliency maps themselves suggest as meaningful. Note how the ordering of the importance values in the unit interval allows for saliency maps of different scale to be aggregated also.

4 Experiments

In this section, the data and models used for the performed experiments are presented, along with the setup of each experiment. The obtained results follow, which are comprised by the drop curves and the corresponding AOPC scores, as described in 3.1 and 3.3.

4.1 Datasets

ISIC-2019

The International Skin Imaging Collaboration (ISIC) organizes in a yearly base open, award-giving challenges, where the contesters are called to solve one or more skin lesion-related computer vision tasks. The actual test data on which the submitters' work is evaluated is not published for several years after the launch and end of the competition. In the year 2019, one of the two tasks of the challenge, [52, 53, 54], was the classification of skin lesion images in one out of nine classes. The ninth class corresponded to the class 'other' for which no training data were available. Figure 6 showcases the target classes of the competition.

Next year's challenge [55] was a binary classification dataset, with classes being benign and malignant. Due to the nature of the task, the train dataset was highly unbalanced with 1.76% malignant labeled images. As a solution, the submitters of the winning solution for the 2020 challenge, [56], took advantage of the fact that the 2020 malignant class was melanoma diagnosed images. They trained their models on the nine classes of the 2019 dataset using both years data and for the prediction phase binned the predictions of non-melanoma as benign. Thankfully, the weights of the trained models are open released on Kaggle [57] and the corresponding code on GitHub [58], allowing for experiments without the extra cost of training the models. The only drawback of testing on these models is their indirect evaluation on the binary rather on the multiclass classification task. Of course, achieving high scores on the binary task is encouraging but up to some degree. Since the compared explanation maps are calculated for the same output nodes, the sensitivity comparison test is valid even for misclassifications.

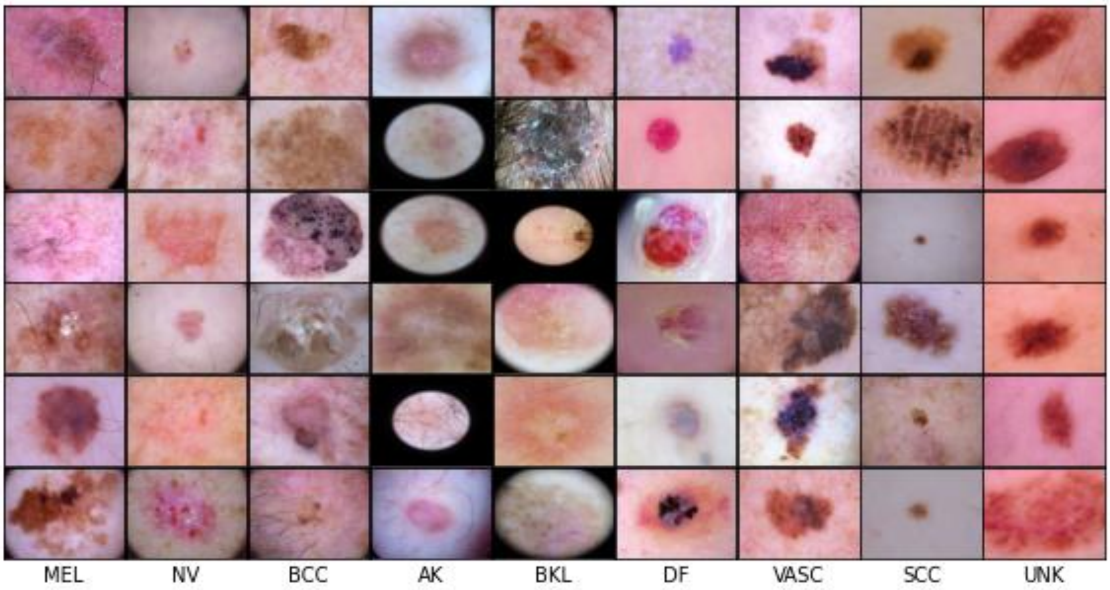

Figure 6. The 9 classes of the ISIC-2019 [52, 53, 54] dataset. The images are sampled from the corresponding test dataset.

NCT-CRC-HE

The train part of the NCT-CRC-HE dataset [59] is a set of 100.000 histological image-patches taken from 86 microscope histopathology images of normal and colorectal cancer tissue. The patches are non-overlapping, with a 224×224 resolution and belong to one out of nine classes, namely, adipose (ADI), background (BACK), debris (DEB), lymphocytes (LYM), mucus (MUC), smooth muscle (MUS), normal colon mucosa (NORM), cancer-associated stroma (STR) and colorectal adenocarcinoma epithelium (TUM), showcased in Figure 7. The validation part of the dataset is comprised of 7180 image-patches from 50 images taken with the exact same resolution and micron per pixel characteristics as the train set. No patient overlap exists among the train and validation sets, while both are hematoxylin and eosin (HE) stained. HE staining causes the various tissue structures to be highlighted differently, either by color, shade or hue. More precisely, hematoxylin is used to illustrate cell nucleus' detail, while eosin is the most commonly used counterstain that distinguishes between the cytoplasm and nuclei of cells. It is typically purple, with different shades of purple for different types of connective tissue fibers. The process of HE staining can introduce inconsistencies to the coloring of each slide but modern procedures highly eliminate such occurrences. In bottom line, this technique aids the pattern recognition task significantly.

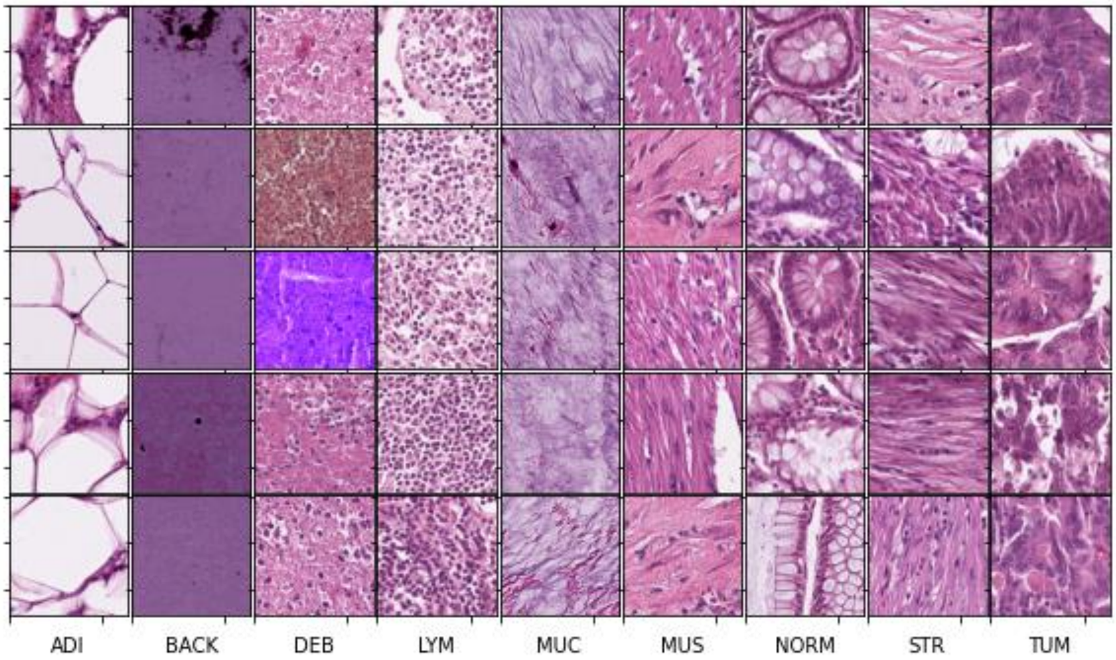

Figure 7. The 9 classes of the NCT-CRC-HE [60] dataset. The images are sampled from the corresponding test dataset.

Imagenette

Imagenette [61], as the name suggests, is a subset of ImageNet, [51], the popular benchmark that most pretrained models are trained on. From the 1000 classes of the full dataset, 10 easily distinguishable classes, showcased in Figure 8, have been chosen to form a basic benchmark set. The concept behind Imagenette is similar to that of CIFAR10, but this time the resolution of the images has been kept at their original shape, allowing for modern CNN architectures to show off their feature learning capabilities and for saliency methods to produce interpretable outputs.

Testing on a generic and well curated dataset is useful for minimizing the possibility of some special data characteristics, i.e., some unknown bias in the ISIC or NCT-CRC-HE datasets, interfering with the outcome of the tests. For instance, for an output score to drop due to feature perturbations, some other classes need to be assigned proportionally higher scores. The class 'Other' for the ISIC-2019 dataset introduces a dataset specific bias, where perturbing an image that belongs to the class ‘Other’ should result to an image still belonging to that same class, and maybe even with a higher probability. The vanilla dataset Imagenette aims to provide a saliency map comparison that is no affected by such characteristics. Furthermore, the simplicity of Imagenette helps for multiple and accurate enough models to be trained quickly so the available computational and time resources can be devoted on the evaluation of the saliency maps, as well as on experimenting with the aggregation formula formulation.

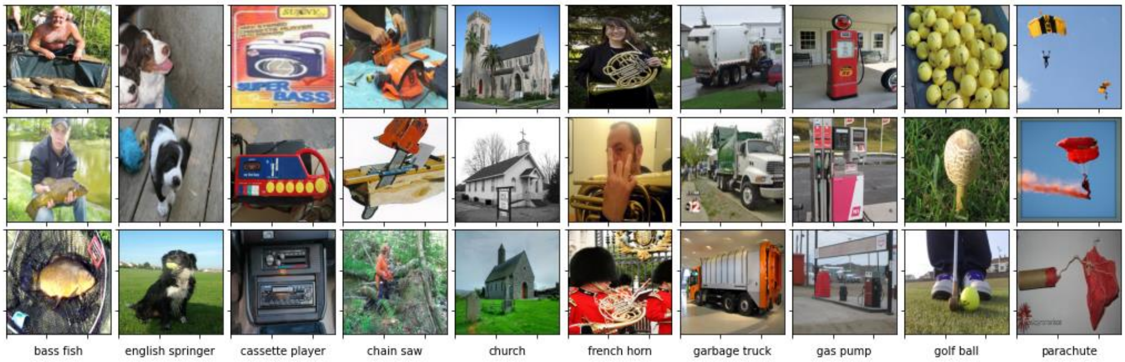

Figure 8. The 10 classes of the Imagenette [61] dataset. The images are sampled from the corresponding test dataset.

4.2 Models

ISIC-2019

In [56] an ensemble of 18 models was submitted as a solution to the SIIM-ISIC 2020 challenge. Some of the models are trained by using metadata of the patients as well, making them unsuitable for producing comparable saliency maps due to the difference in the expected input. Table 1 shows the architectures used in the testing along with an id for easiness of reference later. The AUC scores refer to the binary 2020 challenge and not the multiclass 2019 one, so they are only indirectly representative of the models’ strength. Most of the models trained belong to the EfficientNet’s family. Two variations of the ResNet family are also trained, a ResNext model with squeeze-and-excitations blocks and one is a ResNest model.

Table 1. ISIC-2019 trained models

Model Id	Architecture	AUC (2020)
ISIC-M1	EfficientNet-B4	0.9002
ISIC-M2	EfficientNet-B5	0.9216
ISIC-M3	EfficientNet-B6	0.9154
ISIC-M4	EfficientNet-B7	0.9271
ISIC-M5	SE-ResNext-101	0.9337
ISIC-M6	ResNest-101	0.9267

NCT-CRC-HE and Imagenette

Except for the ISIC dataset, where trained weights are published by their authors, a selection of diverse pretrained models are trained in an automated fashion for both the NCT-CRC-HE and Imagenette datasets. For every combination of these models, the corresponding ensemble is evaluated, and the highest scoring ensemble, as shown in Tables 2 and 3, is chosen for testing. The main goal is to obtain architecturally diverse and adequately accurate models in a time efficient manner, and this is the reason for choosing an automated training process instead of a more targeted one. The models are not tuned in perfection and the achieved scores are not the highest possible. Kallipolitis et al [42] achieved very high scoring models on the NCT-CRC-HE, while the Imagenette can be considered as a toy dataset. Nevertheless, the models do serve their purpose for this work, which is to offer predictions confident enough to obtain meaningful saliency maps, as well as to compose a higher scoring than themselves ensemble.

Tables 2 and 3, apart from the scores and ids of each model, contains metadata about the training procedure also. NCT-E and Imnet-E refer to the ensemble of the corresponding dataset’s models. The final layers of the models depicted in the classifier column of the tables replace the classifier of the original architectures. For dropout layers, the corresponding parameter concerns the percentage of zeroed out neurons when inferencing during the training procedure, while for linear layers, the corresponding parameters concern the input and output dimensions of the layer. The learning rate is the initial learning rate of each training, as on plateaus of more than 10 epochs, the learning rate is decreased by a factor of 0.1. Note that all architectures are loaded with pretrained weights on the ImageNet dataset. The goal here is to exploit the effects of TL so as for the model to converge faster and optimally.

Table 2. NCT-CRC-HE trained models.

Model Id	**Architecture**	**Classifier**	**Epochs**	**F1-macro**
NCT-M1	EfficientNet-B5	Dropout(0.6), Linear(2048,9), SoftMax()	100	0.8454
NCT-M2	Resnext50_32x4d	Dropout(0.6), Linear(2048,512), ReLU(), Linear(512,9), SoftMax()	200	0.8916
NCT-M3	Densenet-169 [40]	Linear(1664, 9), SoftMax()	65	0.9156
NCT-E				0.9272

Table 3. Imagenette trained models.

Model Id	**Architecture**	**Classifier**	**Epochs**	**F1-macro**
Imnet-M1	Inception-v3 [39]	Linear(2048, 10), SoftMax()	50	0.9532

Imnet-M2	Resnet-101	Dropout(0.6), Linear(2048,512), ReLU(), Linear(512,10), SoftMax()	25	0.9447
Imnet-M3	Densenet-169	Linear(1664,10), SoftMax()	50	0.9520
Imnet-E				0.9646

4.3 Saliency map evaluation

To evaluate how precise or not a saliency map is, a quantitative evaluation scheme is necessary. A repetitive perturbation-inference procedure, similar to that of [13], is implemented and used throughout this work. After describing every step of the evaluation pipeline, I also discuss some of their pitfalls. There exist two strong arguments why these pitfalls do not hinder the validity of the evaluation results. First, the evaluation pipeline, along with its drawbacks, is constant along every aspect that is under comparison. For instance, AOPC has received some criticism by [62], but if used for two saliency maps under the exact same regime, all of image, perturbation process and model inferencing being common, it is a sufficient comparison metric. Second, the size of the test datasets in each test performed is more than enough to produce reliable statistics.

4.3.1 MoRF ordered perturbation

Given an image *x* and its corresponding importance heatmap *h(x)*, the Most Relevant First order of *x*, formally *MoRF(x)*, is the descending order of the image's regions, as these are scored for importance by the heatmap itself. Note that DeepLIFT's score assignments are considered important if they are far from zero. That means that negative values do not mean negative importance, but rather negative difference from the reference baseline. *MoRF(x)* orders the pixel elements by their absolute value and dictates the order in which the image's areas are perturbed. Perturbing the most important areas, and since those are highlighted by the saliency map that corresponds to the initially predicted class, should lead to large drop of that class's score. The higher and steeper the score drop is, the more accurate the importance ordering of the image's area and, consequently, the heatmap representation is.

As discussed earlier, the perturbation of a feature aims to hide that information from the model in order to measure its significance. But, since the model is trained to identify instances coming from the same distribution as that of the training data, and the perturbed image may not satisfy this condition, the resulting evaluation suffers from some unreliability. Apart from the data distribution shift problem, perturbation can also introduce objects, especially when an aggressive perturbation function, such as masking, is chosen. In this work, the perturbation function is a noise replacement rule. Specifically, for perturbing the original image x for the i-th time, the i-th element on the *MoRF(x)* order is replaced by gaussian noise and the perturbation x_i results. The generated gaussian noise and the original image share a common mean and standard deviation, in an attempt to avoid as much as possible out-of-distribution images.

The regions that each perturbation step concerns could be singular pixels or areas of the image constituted by groups of pixels. For the latter, a grid over the image can be defined. There are two reasons to prefer larger areas over pixelwise segmentation. The first one is efficiency, a 1024×1024 image consists of over 10^6 pixels, but only 4096 16×16 areas. The second reason is that the outright perturbation of a 16×16, even though it makes for a coarser evaluation step, allows for observable changes on the image. In other words, a single pixel can hardly be considered an image feature for most tasks, while visualizing only a few of the first *MoRF* elements can be informative. In 4.4, the size of the tiles the image is segmented on is listed as an hyperparameter of the testing.

For this 1024×1024 example, 4096 is still a large number of areas. For this reason, the number of perturbations for an inference to take place, as well as the total percentage of perturbed image areas, are two more hyperparameter listed in 4.4. Not perturbing the whole image further helps tackle the out-of-distribution generation problem.

Figure 9 is an example of the perturbation-inference procedure for the 15 first elements of the MoRF order, which are depicted in Figure 3. The model predicts every 3 perturbations and the output score is decreased over 22% for the total perturbation of 1.25% of the image.

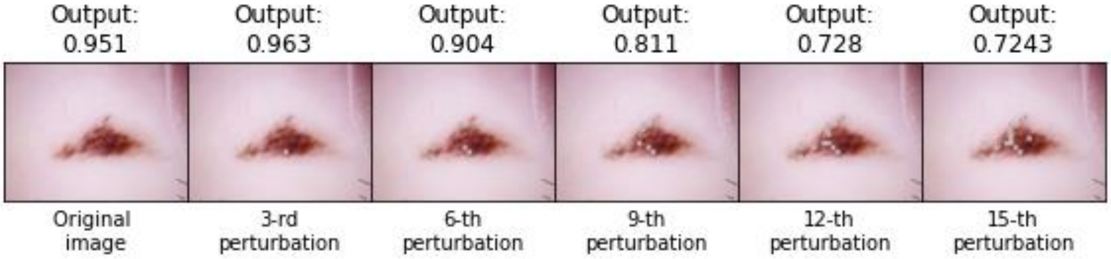

Figure 9. First steps of the perturbation-inference process. The *MoRF* order of the perturbation is depicted in Figure 3 as a heatmap.

4.3.2 AOPC score

The score per perturbation step resulted by the evaluation procedure above can be easily visualized, but a quantified metric offers a more direct comparison for the evaluation results. A popular metric is the AOPC score, given by:

Equation 8. AOPC score.

$$AOPC = \frac{1}{L+1}\left\langle \sum_{k=0}^{L} \left(f(x_0) - f(x_k) \right) \right\rangle_{p(x)}$$

Here, *L* is the number of perturbations and *p(x)* denotes the average over the entire dataset.

It is important to note that AOPC is both dataset and model dependent. Two AOPC scores acquired on a dataset by inferencing with two different models, *f_1* and *f_2*, are not comparable. Not only because different models may have learnt different features, which is also what ensembling tries to take advantage of, but also because different model architectures may not be equally robust to noise. The dataset dependency comes from the fact that the value of an output node is directly dependent on the scores of the rest of the nodes. This means that, in order for a class score to drop while testing by perturbing the input, some other classes need to become more prominent by the same perturbation. This is not obvious, especially for the first number of perturbations, since for the later perturbation steps the image is closer to randomness than ever before. For that said, AOPC is mostly useful for comparisons among perturbation curves of the same model for attribution maps of the same images.

4.4 Experiments setup

As of now, I have described an evaluation and comparison pipeline. Having a dataset and a set of models competent enough to recognize some discriminative patterns in the data, four sets of saliency maps are computed and later compared among each other. The baseline saliency maps (BM) result by applying DeepLIFT on each model, the ensemble saliency maps (EM) by applying DeepLIFT on the ensemble of the models, the aggregated saliency maps (AM) by aggregating the baseline maps as described in 3.3.2, the mean average aggregated saliency maps (MAM) by mean aggregating the baseline saliency maps.

The baseline saliency maps are tested as described in 4.3, where the model that makes the inferences is the model that is used to produce the maps themselves. In this manner, a prediction drop curve is calculated for each available model and the results are mean averaged to get the baseline drop curve (BDC) and the corresponding AOPC score. Since the rest of the saliency sets come from combining information from all the models, they are evaluated once using as inference model every involved model, then the results are mean averaged to get the ensemble (EDC), the aggregated (ADC) and the mean average aggregated (MADC) drop curves along with their respective AOPC scores.

In total, four tests are being performed. Since the ISIC models are mostly EfficientNets, the *ISIC-A* test aims to examine whether different depths of similar architectures learn different features that an ensemble can take advantage of. *ISIC-B* utilizes more architecturally different models to put focus on diversity rather than depth. *NCT* and *Imagenette* tests' purpose it to examine the replicability and generality of any observed results. Table 4 summarizes the performed tests, the participating models and the hyperparameters chosen. ”Area size perturbed” concerns the size of the area perturbed in each perturbation step. “Perturbations per inference“ is the number of perturbation steps between every model inference. “Percentage of perturbed image“ is the total percentage of perturbed image at the end of the evaluation.

All of the size of areas perturbed on each step, the number of perturbation steps between two model inferences and the percentage of the image finally perturbated, are mostly dictated by the original resolution of the images and the size of the datasets. For evaluating a single saliency map of the ISIC-2019 dataset using the perturbation-inference configuration shown in Table 4, and while using an Nvidia RTX3070 GPU, an average of 2.8 seconds is required for the models of ISIC-A test. For the whole test dataset and for all saliency maps sets, this translates to a total time of 102 hours. Considering that computational resources are also needed for the development of the presented work, the configurations of Table 4 are chosen with computation time at mind.

Table 4. Tests performed and involved models.

Test ID	**Models involved**	**Area size perturbed**	**Perturbations per inference**	**Percentage of perturbed image**
ISIC-A	ISIC-M1, ISIC-M2, ISIC-M3, ISIC-M4	(15, 15)	3	15 %
ISIC-B	ISIC-M4, ISIC-M5, ISIC-M6	(15, 15)	3	15 %
NCT	NCT-M1, NCT-M2, NCT-M3	(8, 8)	3	50 %
Imagenette	Imnet-M1, Imnet-M2, Imnet-M3	(8, 8)	3	50 %

4.5 Results

Figures 10 to 13 and Tables 5 to 8 depict the output scores of the models throughout the perturbation process and the corresponding AOPC scores.

ISIC-A

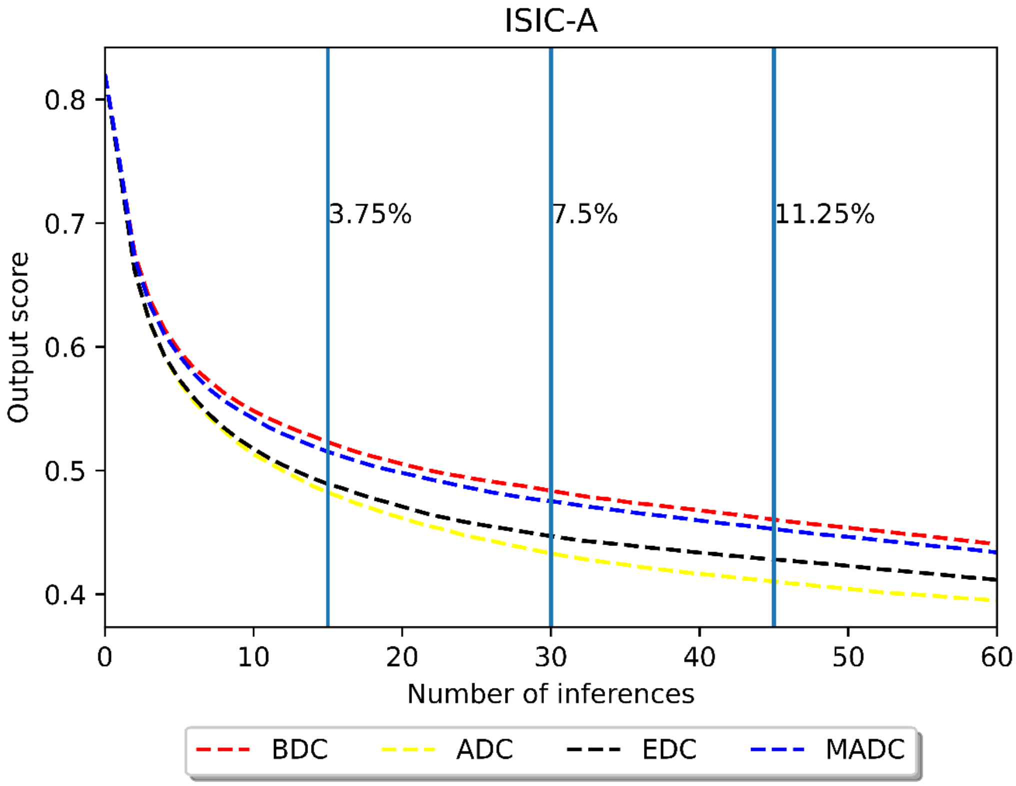

Figure 10. The drop curves of the ISIC-A test.

Table 5. ISIC-A AOPC scores per total perturbation percentage.

Noise level	BDC	ADC	EDC	MADC	EDC over BDC	ADC over BDC
3.75 %	0.2220	**0.2490**	0.2462	0.2272	10.90 %	12.16 %
7.5 %	0.2703	**0.3064**	0.2999	0.2768	10.95 %	13.35 %
11.25 %	0.2926	**0.3372**	0.3275	0.3032	11.92 %	15.24 %
15 %	0.3146	**0.3574**	0.3456	0.3216	9.85 %	13.46 %

ISIC-B

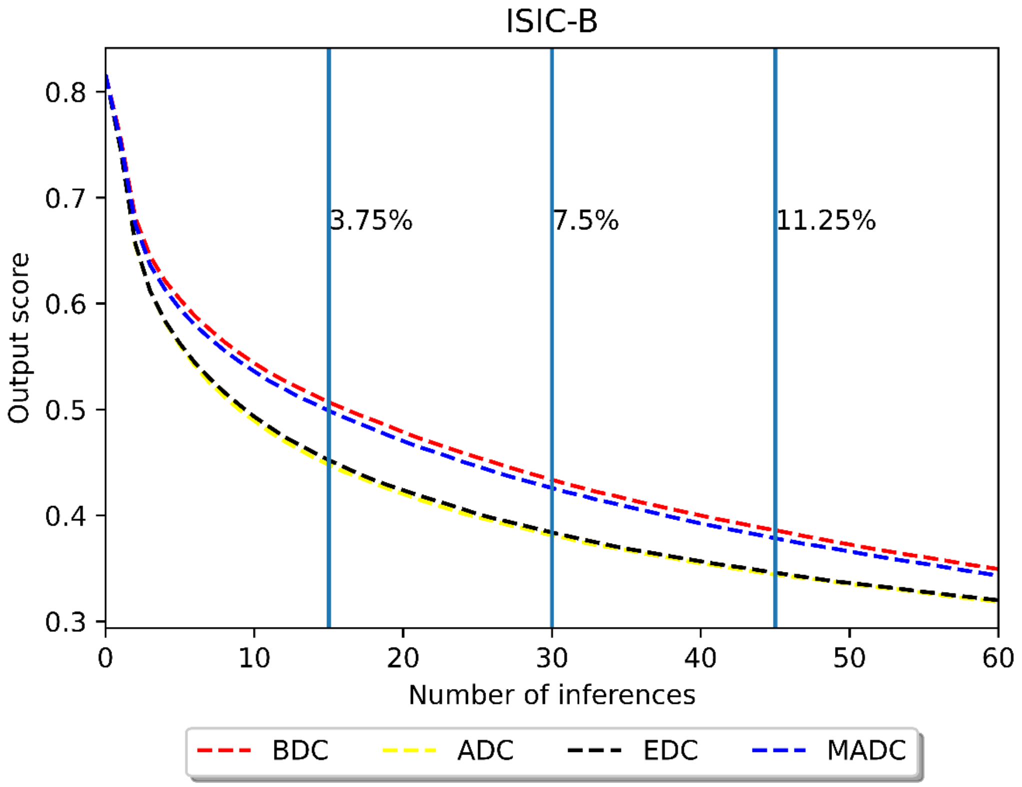

Figure 11. The drop curves of the ISIC-B test.

Table 6. ISIC-B AOPC scores per total perturbation percentage.

Noise level	BDC	ADC	EDC	MADC	EDC over BDC	ADC over BDC
3.75 %	0.2193	**0.2628**	0.2600	0.2266	18.55 %	20.00 %
7.5 %	0.2831	**0.3330**	0.3300	0.2909	16.56 %	17.62 %
11.25 %	0.3244	**0.3732**	0.3705	0.3320	14.21 %	15.04 %
15 %	0.3554	**0.4009**	0.3986	0.3628	12.15 %	12.80 %

NCT

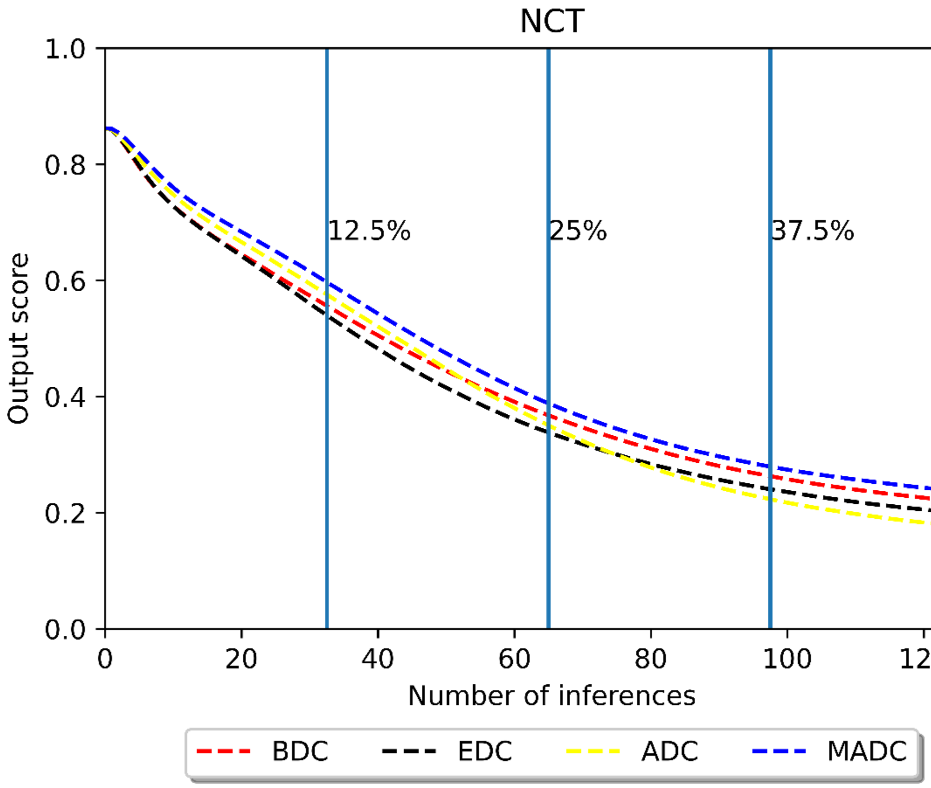

Figure 12. The drop curves of the NCT test.

Table 7. NCT AOPC scores per total perturbation percentage.

Noise level	BDC	ADC	EDC	MADC	EDC over BDC	ADC over BDC
12.5 %	0.1768	0.1592	**0.1809**	0.1444	2.32 %	-10.00 %
25 %	0.2938	0.2837	**0.3092**	0.2623	5.24 %	-3.43 %
37.5 %	0.3812	0.3851	**0.3999**	0.3545	4.90 %	1.02 %
50 %	0.4379	0.4511	**0.4570**	0.4138	4.36 %	3.01 %

Imagenette

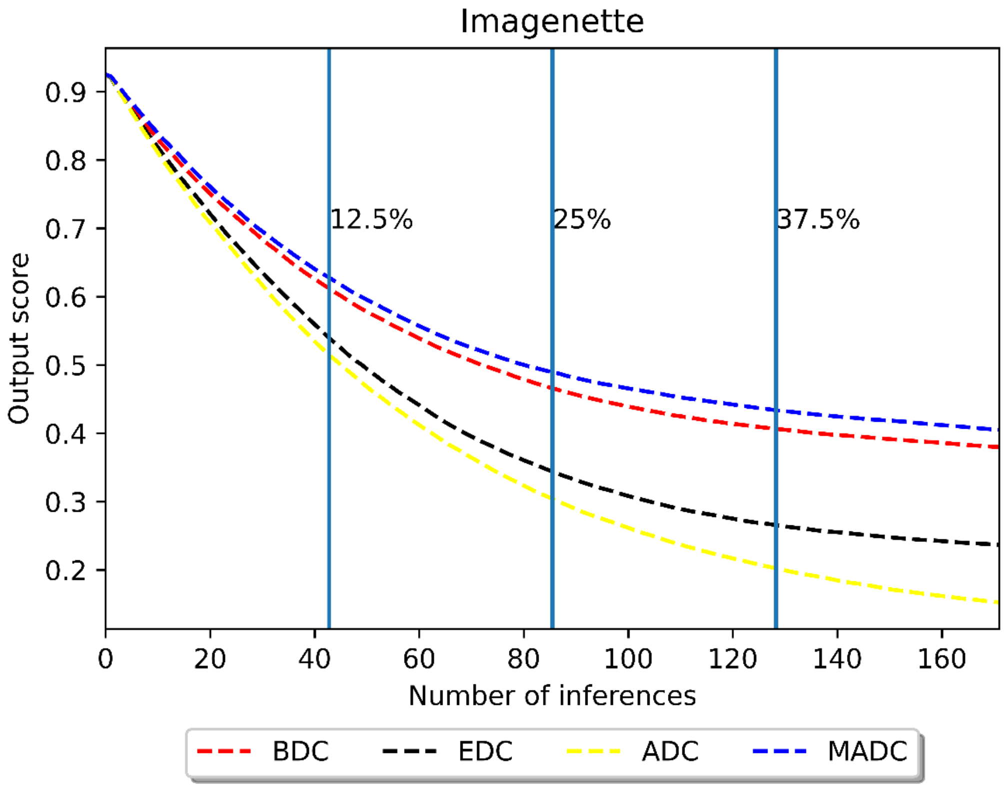

Figure 13. The drop curves of the Imagenette test.

Table 8. Imagenette AOPC scores per perturbation percentage.

Noise level	**BDC**	**ADC**	**EDC**	**MADC**	**EDC over BDC**	**ADC over BDC**
12.5 %	0.1749	**0.2219**	0.2085	0.1647	19.21 %	26.87 %
25 %	0.2855	**0.3752**	0.3532	0.2710	23.71 %	31.41 %
37.5 %	0.3549	**0.4762**	0.4445	0.3363	25.24 %	34.17 %
50 %	0.3963	**0.5404**	0.4981	0.3758	25.68 %	36.36 %

4.6 Qualitative evaluation

The score drops to perturbation level figures do show some differences between the sets of saliency maps. Under the used evaluation scheme, the ensemble and aggregated saliency maps seem to be more accurate about their highlighted areas. But since the AOPC score is relevant to the problem at hand and not a global metric, how much important are these improvements on the attribution visualizations and are they noticeable at all? The most accurate qualitative evaluation requires the eye of an expert, especially when dealing with medical images. Figure 14 showcases a few instances of the ISIC-2019 dataset, where the baseline DeepLIFT saliency maps can be visually compared to the corresponding ensemble and aggregated saliency maps. The images are randomly chosen, except for the fact that the lesion is required to be easily separated from healthy skin, in order for a high-level qualitative comparison to be performed even by non-experts.

Only a few valid qualitative observations can be made by non-experts. The first concerns the coherence of the maps, that is whether the salient areas are those that clearly differ from healthy skin, or are they scattered with a noisy manner. The most consistent about its coherence set seems to be the AM. The ISIC-M6 model also displays some focused saliency maps, while being the lowest scoring model of the three, in contrast what one may have expected. Interestingly enough, EM, although performed well on the quantitative evaluation, seems to include noisier maps. A second observation that can be made is the saliency or not of clearly irrelevant areas, such as ink marks, hair, or the perimeter of the camera lens. The aggregation offers a way of potentially suppressing such artifacts. If a subset of the base models has mostly focused on other regions apart from the artifact itself, its salience on the aggregated map will appear to be lower. Of course, that could be true for relevant regions too. For instance, for the example on Figure 14 where the number one is written on the skin, both the ink and the second lesion at the bottom of the image, which seems to differ from the main lesion at the center though, are suppressed on the AM.

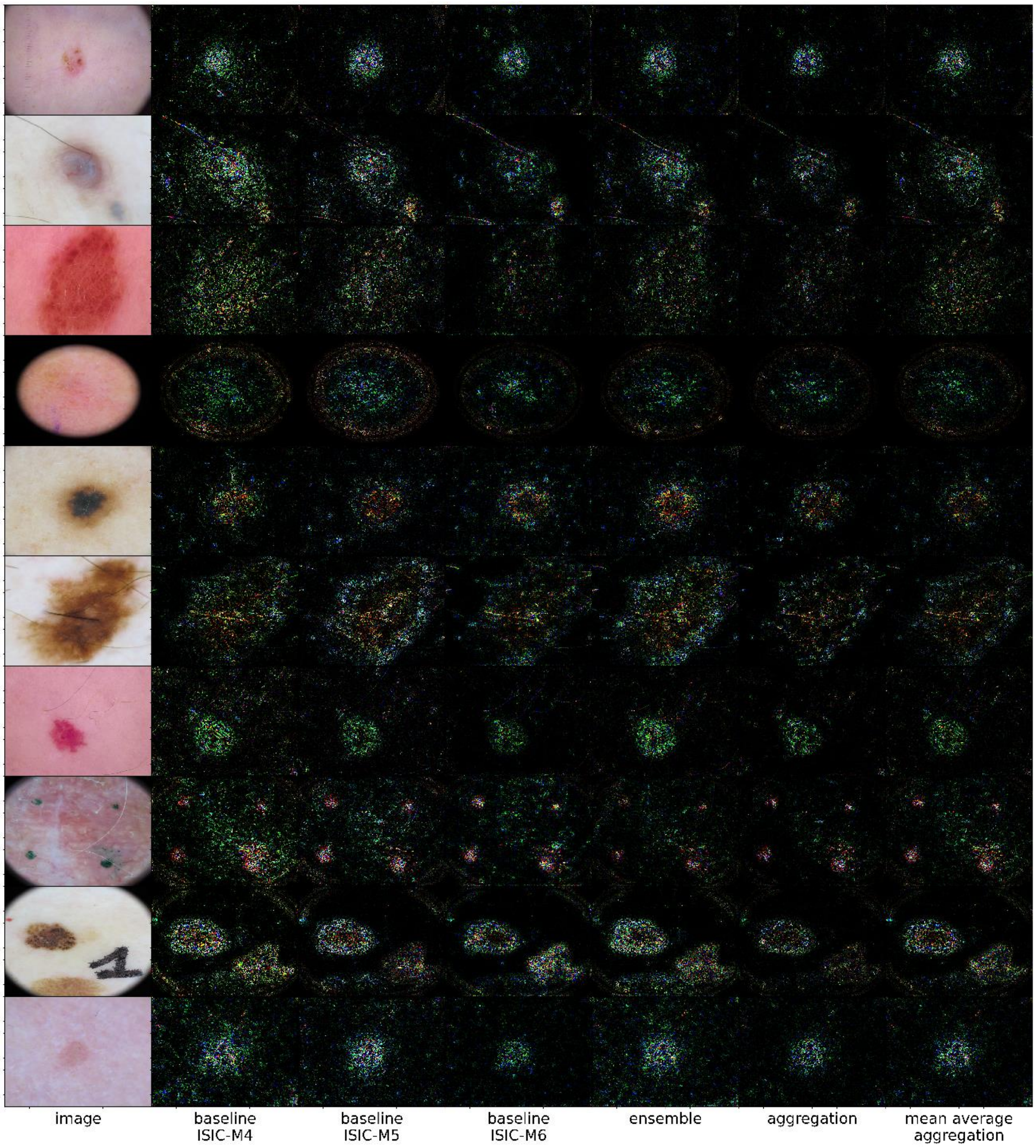

Figure 14. Examples of saliency maps from the ISIC-2019 dataset.

5 Discussion

In this paragraph, an attempt to interpret the experimentation results takes place. Examples of saliency maps along with a high-level qualitative evaluation are also presented. Finally, the hypothesis that stronger models should also offer better interpretations is discussed, always based on the obtained results.

5.1 Conclusions

A first thing to notice is that the output scores drop more abruptly for the tests performed on the ISIC-2019 dataset, as opposed to the more linear drops of the other two figures. Two contradicting hypotheses are that either the ISIC models have learnt more important features about their target distribution and, thus, produce better saliency maps, or the models trained on the NCT-CRC-HE and Imagenette datasets are more robust to the perturbations. The fact that the originally ImageNet-trained models, when retrained on the Imagenette dataset, lose confidence about their predictions linearly to the noise addition too, is in favor of the latter hypothesis. The robustness to noise can originate from the dataset itself and does not necessarily require better trained models.

The percentage of total perturbed image along with the quality of the saliency maps affect the total drop on the output of the models. For the ISIC-A and ISIC-B tests, perturbation of 15% of the input image led to 0.4 and 0.49 average drop correspondingly. Note that the models used for the ISIC-B test are stronger, under the evaluation described in section 4.2. When perturbing half of the image, for the NCT test all the saliency map datasets result to the model assigning scores close to the baseline output score, or in other words predicting randomly. For the Imagenette test, BDC and MADC halt their drop well before reaching the 0.1 randomness level, while EDC and ADC achieve much lower outputs, being the clearest indication that combining the saliency maps is beneficial for interpretability.

Across tests and set ups, the replicable results are the more interesting. EDC and ADC are steadily below BDC and MADC. The only exception is the NCT test, where ADC, even though finally reaches the lowest output scores, for the more important first number of perturbations is above BDC. What is interesting about that test, is that all curves are tangled and close to each other, indicating that the choice of the models, the characteristics of the dataset or both combined do not allow for further improving the BM. Nevertheless, either ensembling models as a means to create a better informed model, capable of assigning more accurate explanations, or directly combining the interpretability information provided by the available models, both seem to be useful for the interpretability of the system.

Did the combination of several models provide more information?

The fact that in all tested scenarios MAM and BM show similar behavior indicates that most, if not all, of the models focus on the same image features. If this is true, then the successful acquisition of better attribution maps through the combinations of several models might actually lie not on the combined information being richer, but rather on indirect enhancement of the attribution assignment. Theoretically this is a plausible explanation. When comparing AM and MAM, it is clear that the noise reduction orientation of the proposed formula offers substantial improvement over the simple mean averaging of the maps. The proposed aggregation deliberately focuses on features that all models highlighted and silences areas of the maps that seem to be noisy, where a noisy area is any salient area that is highlighted only by a few models and not the majority of them. Of course, in cases where all models pay attention on an unimportant feature, for example a hair on an ISIC image is a usual suspect for being highlighted on the attribution map due to its sharp edges, the aggregation of their maps, either by mean averaging or else, will most likely maintain that feature as important. As for the EM, the computation mechanism of the DeepLIFT algorithm when applied on the ensemble architecture could benefit the importance attribution. First off, starting from the predicted output node, the contributions are back-propagated to the corresponding output node of the backbone models. This means that whichever the initial backbone model's prediction was, the attribution expresses the class most likely to be true, since it is the one the ensemble decided. Most importantly though, when applying DeepLIFT on the ensemble's backbones, as well as on the ensemble itself, the exact same quantity Δx and, due to the summation-to-delta property, the exact same difference-from-reference Δt is attributed on the intermediate neurons starting from the target output node. When the back-propagation of the attribution step reaches the layer that combines the backbones models, the back-propagated importance is broken down into proportionate chunks that will continue back-propagating but in parallel. As discussed in 2.2.1, the contribution assigned from each model is equal to the difference-from-reference of the output node that corresponds to the ensemble's predicted class, and this quantity is expected, most of the times, to be close to zero for models that disagree with the voting decision. In other words, as part of an ensemble, each backbone model f_i assigns only a fraction of the importance amount that assigned as a stand-alone model, and what is more, this fraction is proportionate to the difference $f_i(x) - f_i(x_o)$, where x and x_o are the input and its reference respectively. That being said, while several models may provide richer feature learning and, therefore, more interpretability information, either through their saliency maps when aggregating or their architectures and weights when ensembling, the combined saliency maps can possibly be more informative by indirectly enhancing the contribution attribution. A mixture of both of these mechanisms is most likely to be responsible for the seen improvements.

5.2 Observations

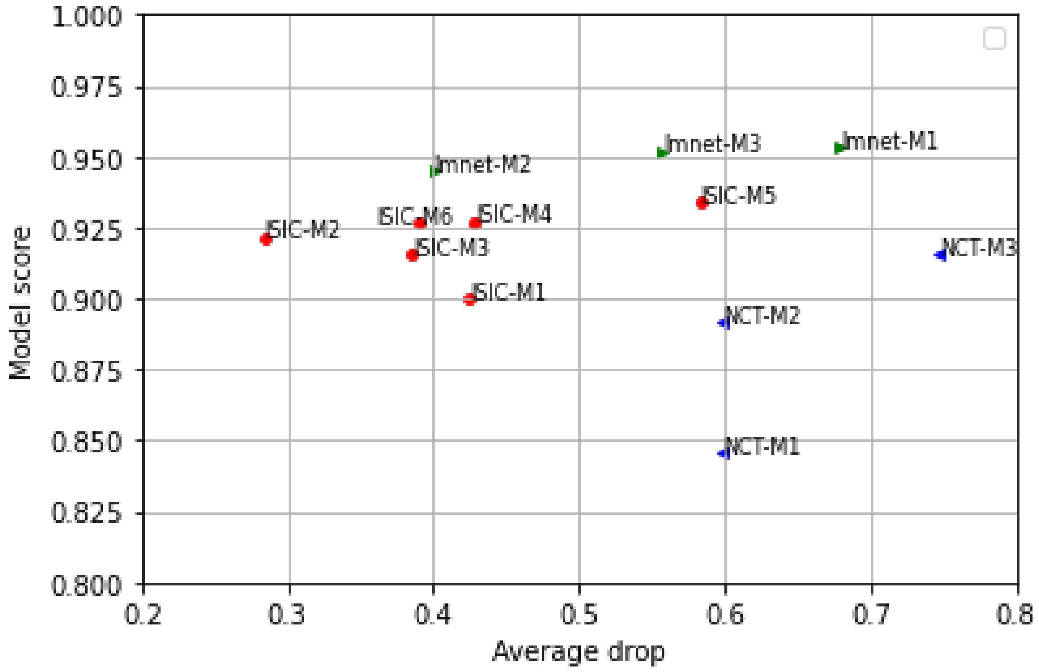

Figure 15. Model's average output drop to model's score scatterplot.

A hypothesis, which also supports the idea that an ensemble might produce better attribution maps than those of its backbones, is that the strongest, as per their predictions, models also produce the best attribution maps. Figure 15 suggests that this is partially true. While the data points do not monotonically ascend, there seems to exist a positive relationship between the score of a model and the total average drop of its output during the perturbation-evaluation procedure. The main concern about accepting this conclusion is of course the size of the sample, since a bigger dataset will add more confidence to the results. What is more, for the NCT models, where the variance of the scores is the largest, the average drop remains steady for models NCT- M1 and NCT-M2 that vary significantly on their accuracy. This contradicts the largest sample of the ISIC models, where great variance on the average drop is observed for closely performing models. In other words, both the sample size and the fact that the data points lie close to each other on the y-axis, could easily create the illusion of higher scoring models being prone to higher drops when testing their attribution maps. That being said, with the data at hand, the hypothesis cannot be accepted, nor rejected.

6 Future work

Since every result of this work suggests that ensembling is promising for the interpretability of image recognition systems, it would be interesting to research the different aspects of this idea. What methods of ensembling better combines and expresses the learnt image features of the models? Is the architectural or the training diversity more important when ensembling for leveraging interpretability? A very interesting approach would be to focus the training on the features of the image themselves, rather on the accuracy of the predictions. Given appropriate labeling, a loss function that prioritizes the importance of the learnt features could place such a focus. Another way to emphasize the importance of the distinct features, and since the final goal is to utilize ensembling of the models for an ML task, would be to train models on recognizing specific features of each class, instead of demanding a generic feature learning.

Apart from ensembling, aggregating provided interesting improvements too, and with minimal computational resources compared to applying DeepLIFT, or any other attribution method, to an ensemble. The proposed aggregation approach allows for the combination of attribution maps of several sources and not only from different models. For instance, different attribution methods, different parameter set ups for the same attribution method, instances of the same model but from different stages of its training procedure, are all possible choices for producing a number of meaningful and potentially complementary attribution maps, using only a single model.

The simplicity of the proposed aggregation formula, which mainly targets the salient pixels while suppressing noise, leaves plenty of room for improvement. For example, the intensity of the attribution is not considered, only the ordering of the values. While for the MoRF perturbation order the actual attribution value of the pixel is unimportant, in reality it makes a big difference for the human interpretation, and it should be taken under consideration.

In most cases, the important features appear coherent in the image data. Apart from being an edge, the neighboring pixels of an important pixel have no obvious reason for not being also important. Aiming for coherence, either as a post processing step of the attribution map aggregation or during the weighting of the aggregation, could bring improvements in the final visualization.

The difficulty of evaluating a saliency map has already been discussed. Despite the effort of applying an as valid as possible evaluation scheme, the maps are only compared as per their faithfulness. Further evaluation of the saliency maps would most certainly add to the credibility of the presented results, especially by focusing on more aspects of the quality of the maps evaluation. Since the interpretations are meant to assist human experts, human-based evaluation is the most important 'metric', which although costly, is utterly necessary.

7 References

[1] "Ethics Guidelines for Trustworthy AI," High-Level Expert Group on Artificial Intelligenc, 2019.

[2] "Assessment List for Trustworthy Artificial Intelligence for self-assessment," High-Level Expert Group on Artificial Intelligence, 2020.

[3] "www.europarl.europa.eu/EPRS_STU(2022)729512_EN.pdf".

[4] H. Surden, "Artificial Intelligence and Law: An Overview," 2019.

[5] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," ECCV, 2013.

[6] R. C. Fong and A. Vedaldi, "Interpretable Explanations of Black Boxes by Meaningful Perturbation," IEEE International Conference on Computer Vision (ICCV), 2017.

[7] A. Shrikumar, P. Greenside and A. Kundaje, "Learning Important Features Through Propagating Activation Differences," ICML, 2019.

[8] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization," International Journal of Computer Vision, 2016.

[9] A. Shrikumar, P. Greenside and A. Kundaje, "Not Just a Black Box: Learning Important Features Through Propagating Activation Differences," ArXiv, 2016.

[10] S. Bach, A. Binder, G. Montavon, F. Klauschen, K. Müller and W. Samek, "On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation," PloS one, 2015.

[11] M. Sundararajan, A. Taly and Q. Yan, "Axiomatic attribution for deep networks," ICML, 2017.

[12] M. Ancona, E. Ceolini, C. Öztireli and A. Gross, "Towards better understanding of gradient-based attribution methods for Deep Neural Networks," ICLR, 2017.

[13] W. Samek, A. Binder, G. Montavon, S. Lapuschkin and K. Müller, "Evaluating the visualization of what a Deep Neural Network Has Learned," IEEE Transactions on Neural Networks and Learning Systems, 2015.

[14] S. Montavon, W. Samek and K. Müller, "Methods for Interpreting and Understanding Deep Neural Networks," Digit. Signal Process., 2017.

[15] P. Linardatos, V. Papastefanopoulos and S. Kotsiantis, "Explainable AI: A Review of Machine Learning," Entropy, 2020.

[16] T. Miller, "Explanation in Artificial Intelligence: Insights from the Social Sciences," Artif. Intell., 2017.

[17] Z. C. Lipton, "The Mythos of Model Interpretability," ACM Queue, 2016.

[18] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders and K. Müller, "Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications," Proceedings of the IEEE, 2021.

[19] A. Arrieta, N. D. Rodríguez, J. Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Bengamins, R. Chatila and H. Fransisco, "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies,, Opportunities and Challenges toward Responsible AI," Inf. Fusion, 2019.

[20] F. Doshi-Velez and K. Been, "Towards A Rigorous Science of Interpretable Machine Learning," ArXiv, 2017.

[21] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. A. Specter and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning," IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA), 2018.

[22] E. Tjoa, "A Survey on Explainable Artificial Intelligence (XAI): towards Medical XAI," IEEE Transactions on Neural Networks and Learning Systems, 2019.

[23] A. N. Borual, S. K. Biswas and S. Bandyopadhyay, Transparent rule generator random forest (TRG-RF): an interpretable random forest, Evolving Systems, 2022.

[24] M. Mukaka, “Statistics corner: A guide to appropriate use of correlation coefficient in medical research.,” Malawi medical journal : the journal of Medical Association of Malawi, 2012.

[25] A. Chattopadhyay, A. Sarkar,, P. Howlader and V. Balasubramanian, “Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks,” IEEE Winter Conference on Applications of Computer Vision (WACV), 2017, 2018.

[26] http://www.heatmapping.org/.

[27] A. Borji, C. Ming-Ming, H. Jiang and J. Li, “Salient Object Detection: A Survey,” Computational Visual Media, 2014.

[28] V. Petsiuk, A. Das and K. Saenko, “RISE: Randomized Input Sampling for Explanation of Black-box Models,” BMVC, 2018.

[29] B. Lakshminarayanan, A. Pritzel and C. Blundell, “Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles,” NIPS, 2016.

[30] A. Krizhevsky, I. Sutskever and G. E. Hinton, “ImageNet Classification with Deep Convolutional,” Commun. ACM, 2012.

[31] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong and Q. He, “A Comprehensive Survey on Transfer Learning,” Proceedings of the IEEE, 2019.

[32] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, F. Fang, J. Bai and S. Chintala, “PyTorch: An Imperative Style, High-Performance Deep Learning Library,” NeurIPS, 2019.

[33] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” ICML, 2019.

[34] X. He, K. Zhao and X. Chu, “AutoML: A Survey of the State-of-the-Art,”

Knowl. Based Syst., 2019.

[35] K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, 2016.

[36] S. Xie, R. B. Girshick, P. Dollár, Z. Tu and K. He, “Aggregated Residual Transformations for Deep Neural Networks,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 2017.

[37] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, Z.-L. Zhang, H. Lin, Y. Sun, T. He, J. Mueller, R. Manmatha, M. Li and A. Smola, “ResNeSt: Split-Attention Networks,” ArXiv, 2020.

[38] J. Hu, L. Shen, S. Albanie, G. Sun and E. Wu, “Squeeze-and-Excitation Networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017.

[39] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, 2016.

[40] G. Huang, Z. Liu and K. Q. Weinberger, “Densely Connected Convolutional Networks,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 2017.

[41] M. Ayhan, L. B. Kuemmerle, L. Kuehlewein, W. Inhoffen, G. Aliyeva, F. Ziemmsen and P. Berens, “Clinical validation of saliency maps for understanding deep neural networks in ophthalmology,” medRxiv, 2021.

[42] A. Kallipolitis, K. Revelos and I. Maglogiannis, “Ensembling EfficientNets for the Classification and Interpretation of Histopathology Images,” Algorithms, 2021.

[43] V. Singh and N. Kumar, “Saliency bagging: a novel framework for robust salient object,” The Visual Computer, 2019.

[44] A. Borji, M.-M. Cheng, H. Jiang and J. Li, “Salient Object Detection: A Benchmark,” IEEE Transactions on Image Processing, 2015.

[45] L. Mai, Y. Niu and F. Liu, “Saliency Aggregation: A Data-Driven Approach,”

IEEE Conference on Computer Vision and Pattern Recognition, 2013.

[46] N. Kokhlikyan, . V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan and O. Reblitz-Richardson, “Captum: A unified and generic model interpretability library for PyTorch,” ArXiv, 2020.

[47] J. Adebayo, J. Gilmer, M. Muelly, I. J. Goodfellow, M. Hardt and B. Kim, “Sanity Checks for Saliency Maps,” NeurIPS, 2018.

[48] L. Sixt, M. Granz and T. Landgraf, “When Explanations Lie: Why Many Modified BP Attributions Fail,” ICML, 2019.

[49] L. V. Kantorovich, “Mathematical Methods of Organizing and Planning Production,” Management Science, 1939.

[50] S. Kullback and R. A. Leibler, “On information and sufficiency,” Ann Math Stat, 1951.

[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” CVPR, 2009.

[52] P. Tschandl, C. Rosendahl and H. Kittler, “The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions,” Sci. Data 5, 2018.

[53] N. C. F. Codella, D. Gutman, M. E. Celebi, B. Helba, M. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler and A. Halpern, “"Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC)",” ArXiv, 2017.

[54] M. Combalia, N. C. F. Codella, V. Rotemberg, B. Helba, O. Reiter, A. C. Halpern, S. Puig and J. Malvehy, “BCN20000: Dermoscopic Lesions in the Wild,” ArXiv, 2019.

[55] V. Rotemberg, N. Kurtansky, B. Betz-Stablein, L. Caffery, E. Chousakos, N. Codella, M. Dusza, S. Dusza, P. Guitera, D. Gutman, A. Halpern, B. Helba, H. Kittler, K. Kose, S. Langer, S. Lioprys, J. Malvehy, S. Musthaq, J. Nanda, O. Reiter, G. Shih, A. Stratigos, P. Tschandl, J. Weber and P. Soyer, “A patient-centric dataset of images and metadata for identifying melanomas

using clinical context," Sci Data, 2021.

[56] Q. Ha, B. Liu and F. Liu, "Identifying Melanoma Images using EfficientNet Ensemble: Winning Solution to the SIIM-ISIC Melanoma Classification Challenge," ArXiv, 2020.

[57] "https://www.kaggle.com/datasets/boliu0/melanoma-winning-models".

[58] "https://github.com/haqishen/SIIM-ISIC-Melanoma-Classification".

[59] Kather, J. Nikolas, Halama, M. Nielsand and Alexander, "100,000 histological images of human colorectal cancer and healthy tissue," Zenodo, Apr. 07, 2018.

[60] J. N. Kather, N. Halama and A. Marx, "100,000 Histological Images Of Human Colorectal Cancer And Healthy Tissue.," 2018.

[61] "https://github.com/fastai/imagenette".

[62] R. J. Tomsett, D. Harborne, S. Chakraborty, P. K. Gurram and A. Preece, "Sanity Checks for Saliency Metrics," AAAI, 2019.

[63] L. Sixt, M. Granz and T. Landgraf, "When Explanations Lie: Why Many Modified BP Attributions Fail," Tim, 2019.

[64] J. Ker, L. Wang, J. Rao and C. T. C. Lim, "Deep Learning Applications in Medical Image Analysis," IEEE Access, 2018.

[65] J. Howard, "imagenette".