Benchmarking Explainability of ImageNet Classification Against Data Suboptimality

Alex Fleury, Brandon Kim, Gary Wu, Jamin Liu

December 2024

We have made all of our code available here:

https://github.com/jamin9902/cs2822r-final.

1 Introduction

Interpretable machine learning is an area of research that aims to help stakeholders understand ML models and their underlying decision-making processes. Studies in this field strive to capture the underlying reasoning of black box methods, and have explored the application of these methods in a diverse collection of real-world settings, ranging from healthcare to cybersecurity to law. Many such studies operate implicitly under the assumption of perfect data and circumstance; however, it is both unrealistic and impractical to assume that all data fed to decision-making machine learning models will be clean, unperturbed, and optimal. In this paper, we introduce a novel benchmarking of machine learning explanation methods that incorporate varying levels and types of data suboptimality, specifically for the task of **image classification**.

The formal problem specification for this paper is: How does data suboptimality through noise, occlusion, and resolution reduction impact the explanation quality (robustness & stability) of SmoothGrad and LIME on image classification?

1.1 Motivation

Our principal motivation for this project stems from the lack of pre-existing literature regarding the effects of data suboptimality on explanation methods for image classification.

Previous papers have benchmarked the effects of data suboptimality/perturbation on the performance of image classifiers themselves, but not on the explanations for classifications [3]. Other studies have explored how explanations can be manipulated through perturbations to images that produce significantly different

explanations; however, these studies only focus on a specific single perturbation method and do not explore suboptimality more broadly [2].

Beyond the lack of literature, our motivation additionally stems from a practical need for this benchmarking and the importance of transparency in the face of suboptimal or unclear data. Data suboptimality can occur frequently in image classification tasks. In real-world settings, there exist a myriad of factors that can lead to suboptimal image quality. However, image classification models must classify on these images nonetheless, and it becomes even more important in these cases that the explanations for their classifications maintain transparency.

2 Methods

To explore the effects of data suboptimality on image classification explanations, we first selected an image dataset for classification, as well as a black box image classification model. For our image classification setup, we use the following:

- **ImageNet-1k Dataset**
- **ResNet Image Classification Model**

To inject suboptimality into our image dataset, we selected image perturbation methods which we deemed commonly occurring in real-world suboptimal images. Additionally, we selected perturbation methods that could be used to increasing degrees to relate the performance of explanations to the severity of suboptimality. Specifically, we selected the following suboptimality methods:

- **Gaussian Noise**
- **Occlusion**
- **Resolution Reduction**

To explain image classification on the suboptimal data, we selected two common explanation methods which we explored during the course:

- **SmoothGrad**
- **LIME**

Finally, to evaluate the explanations produced by these methods on suboptimal images, we utilized the following evaluation methods to measure the quality of the generated classification explanations:

- **Robustness**
- **Stability**
- **Feature Attribution Maps**

2.1 ImageNet-1k Dataset

For our image dataset, we utilized ImageNet-1k, a subset of the larger ImageNet dataset. ImageNet-1k is a comprehensive image collection meticulously organized around the WordNet semantic hierarchy. WordNet, a sophisticated linguistic database, categorizes concepts through "synsets" - nuanced groupings that capture the semantic richness of language. With approximately 100,000 such synsets in the database, ImageNet-1k's goal is to provide around 1,000 carefully curated and precisely annotated images for each conceptual category, ensuring both breadth and high-quality representation. We accessed these data through HuggingFace.

2.2 ResNet-50 Model

For the image classification model, we employed ResNet-50, a widely used deep learning architecture used for image recognition tasks. ResNet-50 comprises 50 layers and uses a residual block architecture, which reduces the risk of vanishing gradients and allows for smooth gradient propagation in deep networks. The model is pre-trained on the ImageNet dataset, and Microsoft provides the specific version we accessed through the HuggingFace library.

2.3 Suboptimality

In theoretical settings, it is easy to assume that one can work with perfect data and use this presupposition to generate idealized results. In practice, this assumption almost never holds, meaning real-world implementations of these methods must always be robust to suboptimality. In this report, we investigate the following three types of image suboptimality.

(a) Original Image

(a) Gaussian Noise (b) Resolution Reduction (c) Occlusion

Figure 2: Three different types of image suboptimality

2.3.1 Gaussian Noise

Our first method of injecting suboptimality into images is through Gaussian noise. Under this suboptimality method, we alter each pxiel in an image with a noise value sampled from a Gaussian distribution. Specifically, for this distribution, we use a mean of 0 and increase the standard deviation of the distribution based on the desired level of suboptimality. In our experiments, we used standard deviations of 10, 25, and 50 for low, medium, and high levels of suboptimality, respectively. This type of suboptimality could present itself in real-life applications through low-quality image-capturing devices or precision loss in file compression.

2.3.2 Occlusion

The second data suboptimality method is occlusion, which covers a portion of the image with 0-valued pixels, effectively "blacking out" a part of the image. The level of suboptimality for this method determines the size of the occlusion and is expressed as a percentage of the overall image size. The location of occlusion is randomly determined to mimic real-world conditions. We can imagine scenarios where occlusion might be present in image classification through file corruption or edited images used for adversarial attacks.

2.3.3 Resolution Reduction

The final suboptimality method is resolution reduction, which blurs together pixels based on a specified block size. Within a specific block, all pixels are set to the average pixel value of the block, effectively reducing the number of unique pixels representing the image. For this method, the degree of suboptimality determines the size of the block used for the reduction. This type of suboptimality is likely most common in real-world settings and can simply be caused by the compression of image files.

2.4 Explanation Methods

Next, to conduct our benchmarking, we draw upon common machine learning explanation methods seen in class. These explanation methods aim to reveal

the underlying decision-making compass of our underlying ResNet black box model. We utilize the following explanation methods to explain our image classification task.

2.4.1 SmoothGrad

SmoothGrad is an explanation method designed to improve the interpretability of deep neural network predictions by reducing noise in gradient-based sensitivity maps. The key idea behind SmoothGrad is to reduce the appearance of spurious, noisy artifacts that often plague gradient-based explanation techniques by adding small random noise to the input and averaging the gradients across multiple noisy samples. [5]

2.4.2 LIME

LIME (Local Interpretable Model-agnostic Explanations) is a method for explaining the predictions of machine learning models, particularly black box models. It provides local interpretability by approximating the behavior of complex models around specific instances. To do so, it operates on data samples perturbed around an instance of the input, then fits an interpretable model (typically a linear regression model) to the black-box model's predictions on the perturbed samples. [4]

2.5 Evaluation Methods

Using the aforementioned explanation methods to explain image classification on perturbed data, we must then evaluate the quality of the generated explanations to complete our benchmarking. We implement the following three evaluation methods to examine the generated classification explanations quantitatively and visually.

2.5.1 Robustness

Robustness is the ability of a model to maintain accurate and consistent performance across different contexts and datasets. Robust models can handle noise, outliers, and distributional data shifts and remain accurate or functional even when faced with intentional obfuscations. In the context of explainability methods, robustness also involves the method's ability to consistently identify relevant features in the presence of perturbations.

For our robustness calculations, our explanation method generates a mask M, and we define $M = f(I)$, where f is the explanation method and I is a given image. We then do the same thing for a perturbed image I': generate a perturbed mask $M' = f(I')$. We then flatten the masks M and M' into a 2D array that represents the importance of each pixel or region in the image. We

then calculate the MSE for each between the masks:

$$MSE(M, M') = \frac{1}{n}\sum_{i=1}^{n}(M_i - M_i)^2$$

n is the total number of pixels in the flattened mask, and M_i is the ith pixel in the image.

2.5.2 Stability

Stability refers to the property of a model where its predictions don't change significantly when small perturbations are made to the training data. Even with slight modifications to the input, the model's output remains relatively consistent, indicating a robust and reliable algorithm; essentially, a stable model is not overly sensitive to small changes in the data it's trained on.

We engineered a stability calculation method that aligns with our project goals and setup. We define, given an explanation map M, we generate $M'^{(1)}$, $M'^{(2)}$, ..., $M'^{(k)}$, explanation masks for perturbed versions of the same image after applying k different independent perturbations. $M'^{(i)} \perp\!\!\!\perp M'^{(j)} \ \forall \ i, j$. We proceed to flatten the masks per usual and then measure the variability across masks:

$$Variance(M') = \frac{1}{k}\sum_{i=1}^{k}(M'^{(i)} - \mu_i)^2$$

where $\mu_j = \frac{1}{k}\sum_{i=1}^{k} M_j'^{(i)}$. We are measuring how much the explanation fluctuates across perturbations. Then, we define stability:

$$Stability = \frac{1}{n}\sum_{i=1}^{n} Var(M')_j$$

this represents how consistent explanation highlights important regions for the same input under repeated, differing perturbations. [1]

2.5.3 Feature Attribution Maps

Feature attribution maps provide insights into how input features contribute to a model's predictions. These maps visually represent the importance of each feature in influencing the model's output for a specific instance. We can use both SmoothGrad and LIME to generate feature attribution maps. Again, we vectorized the masks and then a) converted them to images for analysis and b) computed calculations based on the perturbed and non-perturbed versions.

3 Experimental Setup

We conducted a series of experiments comparing LIME and SmoothGrad. Our experiments focused the three data perturbation methods: noise, occlusion, and resolution reduction. We applied each type of suboptimality at three intensity levels:

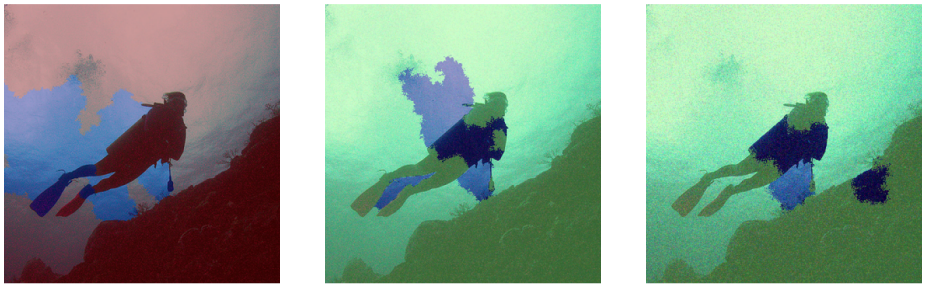
1. **Noise:** We introduced increasing levels of Gaussian noise with standard deviation $\sigma^2 \in \{10, 25, 50\}$, where $0 \leq \sigma^2 \leq 255$.
2. **Occlusion:** We used 0-valued pixel blocks to cover 10%, 30%, and 50% of the image area based on image height and width.
3. **Reduction:** We downsampled using larger and larger block sizes (5, 10, and 20 pixels) to mimic coarse/low-quality inputs with low resolution.

For each image, we applied each of the three types of data perturbation methods using the three degrees of severity as outlined above. For each perturbed image, we generated explanations using both LIME and SmoothGrad. We also generated an explanation on the original, unperturbed image. We evaluated these explanations using the evaluation methods described above, as well as the false positive rate and total highlighted area.

4 Results

Executing the experimental design outlined above, we achieved the following results in our benchmarking.

4.1 Noise



(a) $\sigma = 10$ (b) $\sigma = 25$ (c) $\sigma = 50$

Figure 3: LIME explanations on image recognition of a scuba diver at varying noise injection levels σ

(a) $\sigma = 10$ (b) $\sigma = 25$ (c) $\sigma = 50$

Figure 4: LIME explanations on image recognition of a plant at varying noise injection levels σ

The results for noise perturbations revealed stark differences in robustness between the two explanation methods. SmoothGrad maintained consistent explanation quality across all noise levels, as illustrated in Figure 2, with only minimal increases in false positive rates and relatively stable highlighted areas.

For LIME: for $\sigma^2 \in [0, 75]$, the results are more interesting: LIME explanations improved samples from normal distributions with standard deviations in this range. Beyond that, ($\sigma^2 \geq 80$) LIME became increasingly unstable. Its explanations were more fragmented and often highlighted irrelevant image regions. Figure 2 shows LIME's saliency maps drifting away from critical features. Might remove: These discrepancies underscore LIME's vulnerability to pixel-level perturbations.

4.2 Occlusion

When small portions of the image were occluded (10%), LIME and SmoothGrad retained coherent explanations and stable highlighted areas. However, as the masked region grew to 30% or even 50%, the methods began to diverge. Figure 3 illustrates that LIME's explanations became disjointed and sometimes failed to emphasize meaningful features when occlusions hid large parts of the image. In contrast, SmoothGrad showed greater resilience, adjusting its focus to the remaining visible structures and still managing to highlight the most relevant portions of the images.

Considering images 1 and 2, SmoothGrad's explanations deteriorated more gracefully than LIME's. Relative to LIME, SmoothGrad minimizes the worst-case scenario as its averaging technique keeps explanations in the right area; it is just less precise in best-case scenarios.
However, when the occluded sections were optimally placed, LIME improved greatly across all metrics. LIME's local explanation methods showed resilience when unimportant features were suppressed.

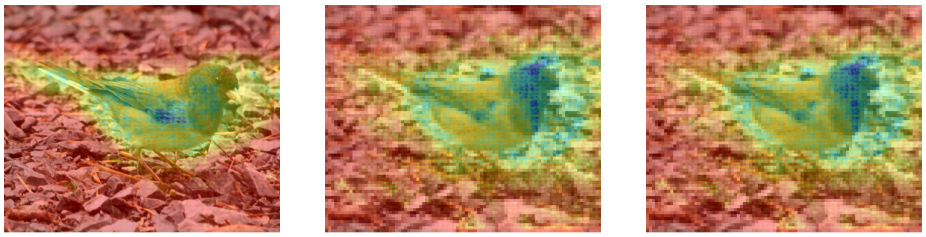
Figure 5: Cropped images displayed side by side. Left: Original image. Right: Perturbed image.

4.3 Resolution Reduction

Coarse and low-resolution images pose a unique challenge by removing fine-grained detail. LIME struggled to identify meaningful features; its explanations drifted toward irrelevant regions as block size increased. Figure 4 demonstrates this trend: LIME's highlighted areas under severe resolution reduction bear little resemblance to the original relevant features. SmoothGrad showed stronger resilience for reasons similar to those described for Occlusion. Resolution reduction perturbed the image to a point where humans would have difficulty classifying, yet SmoothGrad kept its explanations generally correct.

(a) Original Image (b) Low Res. Reduction (c) High Res. Reduction

Figure 6: LIME explanations on image recognition at varying levels of resolution reduction



(a) Original Image (b) Low Res. Reduction (c) High Res. Reduction

Figure 7: SmoothGrad explanations on image recognition at varying levels of resolution reduction

4.4 Quantitative Data

Quantitative data supports the above visual claims. We have below the robustness loss for both SmoothGrad and LIME according to each degree of perturbation we tested and defined above. Now, we consider false positive

Table 1: SmoothGrad Robustness Loss

	Low	Medium	High
Noise	0.00004	0.00003	0.00003
Occlusion	0.00006	0.00005	0.00007
Resolution	0.00013	0.00013	0.00014

Table 2: LIME Robustness Loss

	Low	Medium	High
Noise	0.087	**0.077**	0.087
Occlusion	0.079	0.086	0.090
Resolution	0.105	0.106	0.109

rate data. It supports the above regarding LIME improvement in general, noise, and (sometimes) occlusion at reasonable levels. For SmoothGrad, however, the results are not as indicative of improvement. This data adds the non-visual takeaway that medium levels (as described above) are the lowest regarding *each* perturbation type.
For LIME, our stability analysis led to relatively unclear results, with variation resulting from a) smaller sample sizes than required for a rigorous average over all images in ImageNet—1k.

Table 3: LIME False Positive Rate

	Low	Medium	High
Noise	0.95	**0.92**	0.97
Occlusion	0.86	**0.88**	0.97
Resolution	0.99	**0.94**	0.97

Table 4: SmoothGrad False Positive Rate

	Low	Medium	High
Noise	0.225	0.223	0.226
Occlusion	0.410	0.380	0.315
Resolution	0.320	0.325	0.327

5 Discussion

Throughout our research and experimentation, we have noticed a few key discussion areas.

5.1 Explanation-Specific Differences

The results are indicative of underlying mechanisms surrounding LIME and SmoothGrad. We noticed that between the two, LIME tended to show greater instability under intense levels of perturbations. This could result from its reliance on local interpretability, amplifying the effect of noise in highly perturbed images.

LIME's ability to improve with proper occlusions leads us to think that its local approximation method improves when naturally less-relevant features are removed. Spurious gradients surrounding non-relevant regions of the image are shrunk to 0 and allows the local model to focus on areas that truly reflect the model's rationale in prediction.

In contrast, SmoothGrad's use of gradient averaging allowed it to remain relatively consistent in capturing important regions even under very sub-optimal conditions.

5.2 Perturbation-Specific Insights

Gaussian Noise: The addition of low levels of gaussian noise unexpectedly improved LIME's performance. As previously noted, this suggests that slight perturbations may help mitigate overfitting. As noise increased, both explanation methods began to struggle, with LIME's saliency maps diverging from important features.

Occlusion: Occlusion revealed stark differences in method resilience. SmoothGrad adjusted its focus to remaining visible features. On the other hand, LIME's explanations often became disjointed, particularly under larger occlusions. Our results tentatively suggest that SmoothGrad's averaging mechanism is more robust to data loss.

Resolution Reduction: Resolution reduction seemed to be the the most problematic form of perturbation. LIME's explanations became increasingly fragmented as block size increased, often highlighting irrelevant regions. Meanwhile, SmoothGrad's explanations somewhat retained alignment with the original image's features, but as block size increases the interpretations become less tightly concentrated around relevant features.

5.3 Feature Correlation Trade-off in LIME

As mentioned in the results, noise levels in a lower range $25 \leq \sigma^2 \leq 75$ can increase explanation quality by reducing over-fitting to spurious details, but noise levels beyond this range can suppress sensitivity to subtle, critical features, increasing the risk of less precise explanations. Specifically, LIME's local condensed its explanations to important regions in the image under additional noise levels for the majority of images.

However, in some cases, this better representation of the model's prediction introduced unrelated 'important' features in random places in the map (which had little to no correlation to the true relevant region), i.e., the noise introduced randomness in other parts of the image. In practice, this makes interpretability dubious when *it is not immediately clear* what the ground truth is. In these cases, humans would be misled into thinking that features are correlated when they aren't. In practice, it is also hard to know when low and high noise levels exist or can be beneficial.

5.4 Ground Truths

It is important to note that we are limited in our ability to establish a ground truth against which to benchmark our explanations. In our implementation, our approach to establishing a ground truth was to leverage our local devices' compute and classify a non-perturbed image with each explanation method $\sim$ 30 times.

We then averaged over the explanations provided by LIME and SmoothGrad and benchmarked these results with the one-trial explanations on perturbed images. We recognize that this is not a fool-proof route to actual ground truths for a Resnet on these images. However, this does not prevent us from reaching useful conclusions about the performance of specific explanation methods against specific data suboptimality.

5.5 Real-world Applicability

We think practically about what types of data suboptimality are common in the real world.

Specifically with **occlusion**, we can imagine scenarios in which occlusion can actually help explanation methods, such as where areas causing confusion can be blocked out. Additionally, some scenarios are particularly destructive for explanation methods, such as blocking out key features within the data. In the scenario that occlusion is deliberate (think blocking of sensitive information in medical patient images), we might see no difference in classification accuracy given that the data isn't supposed to be read anyway and likely won't disrupt important features in the data.

6 Conclusion

Deploying image classification models in the real world presents a meaningful challenge in the face of suboptimal data. While previous research has benchmarked explanation methods on the evaluation methods outlined in our project, they have not considered the data suboptimality challenge. Through our experiments, we unsurprisingly find that higher levels of data perturbation negatively affect explanation performance. However, in the process we also uncover interesting findings, most notably that the introduction of small amounts of noise actually can help the performance of LIME explanations as can occlusion in the right contexts.

Overall, we hope that our project encourages meaningful consideration on the effect of data suboptimality in image classification explanation. We hope that our discussion highlights the need for rigorously exploring this topic to understand how the consequences of real-world suboptimal data may impact interpretable machine learning methods for image classification - especially in high-stakes situations.

Future studies could expand upon this work by incorporating additional perturbation types, such as lighting variations or non-random occlusion patterns, to further benchmark explanation performance under sub-optimal real-world conditions.

References

[1] Umang Bhatt, Adrian Weller, and José M. F. Moura. Evaluating and aggregating feature-based model explanations. *CoRR*, abs/2005.00631, 2020.

[2] Ann-Kathrin Dombrowski, Maximilian Alber, Christopher J. Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame, 2019.

[3] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *CoRR*, abs/1903.12261, 2019.

[4] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should I trust you?: Explaining the predictions of any classifier. *CoRR*, abs/1602.04938, 2016.

[5] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017.