Stability of Feature Attributions Under Data Augmentation

How stable are SHAP and LIME explanations when models are trained with varying data augmentation strategies?

Lucca Barcelos, Paul Steinbrink
https://github.com/AkkiBC/iML_SFAuDA

1 Motivation

Modern machine learning models are routinely trained using data augmentation to improve generalization and robustness, yet the effect of these augmentations on the reliability of post-hoc explanations remains largely unexplored. As explanation methods such as SHAP and LIME are increasingly used in high-stakes settings, unstable feature attributions can undermine trust and decision-making. This project addresses a critical gap by systematically studying whether common augmentation strategies stabilize or destabilize explanations across training runs. Understanding this interaction is essential for deploying explainable models that are both performant and trustworthy.

2 Related Topics

This project relates to prior work on the stability and robustness of post-hoc explanation methods such as saliency maps, LRP, LIME, and SHAP. Research has shown that explanations can vary significantly when inputs are slightly perturbed, indicating that explanations may be less stable than the underlying model's predictions and that robustness is an important evaluation dimension in explainable AI (XAI). (Vascotto et al., 2025)

Several studies highlight the lack of consistency in surrogate-based explainers: for example, LIME's reliance on random perturbations can lead to different outputs on repeated runs, while SHAP's theoretical grounding tends to provide more stable, but still imperfect, feature attributions. (Bodria et al., 2023)

Finally, empirical evaluations in multiple domains (vision, tabular, and text) emphasize that explanation methods can be sensitive to noise, model variations, and data distribution shifts, motivating evaluation protocols that compare explanations across noise levels, augmentation regimes, or multiple runs instead of treating them as fixed outputs. (Repetto et al., 2025)

3 Idea

The goal of this project is to investigate how common data augmentation strategies affect the stability of feature attributions produced by post-hoc explanation methods such as SHAP and LIME. We will train machine learning models under different augmentation regimes and compare the consistency of their explanations across multiple training runs on a fixed test set. By quantifying attribution stability using simple correlation and variance-based metrics, the project aims to determine whether augmentations—intended to improve generalization—also lead to more reliable explanations or instead introduce additional variability. This analysis will help clarify the relationship between model training practices and explanation trustworthiness.

Algorithm 1 Explanation Stability under Data Augmentation

Require: Dataset D, learning algorithm A, augmentation strategy $\mathcal{T}$, explanation method E, number of runs R
Ensure: Explanations $\{\alpha_1, \ldots, \alpha_R\}$, stability score S
  for $r = 1$ to R **do**
    Sample augmented dataset $D_r \sim \mathcal{T}(D)$
    Train model $f_r \leftarrow A(D_r)$
    Compute explanation $\alpha_r \leftarrow E(f_r, D_{\text{test}})$
  end for
  Compute stability score $S \leftarrow \text{Stability}(\{\alpha_1, \ldots, \alpha_R\})$

4 Experiments

Dataset & Metrics We will focus primarily on small, well-understood benchmark datasets to ensure fast and reliable experimentation. For tabular data, we will use datasets such as Iris or Wine from scikit-learn, and for images, we will use the MNIST dataset. Explanations will be computed using SHAP and LIME, and stability will be quantified using simple metrics such as the variance of attribution values across repeated runs.

Experimental Scope Experiments will compare a baseline model trained without data augmentation to models trained with one or two simple augmentation strategies (e.g., random noise or basic geometric transformations). Each configuration will be repeated with a small number of random fixed seeds to capture variability while keeping reproducibility. Hyperparameters will largely be kept fixed across experiments, with only minimal ablations on regularization to avoid extensive tuning.

Estimated Computational Load All experiments are designed to run on a standard CPU-based system, without requiring specialized hardware. Based on prior experience with comparable setups, we anticipate that training on the tabular datasets will require only a few seconds per run, while training lightweight models on the image datasets should take a few minutes per run. Given the number of combinations and configurations to evaluate, the complete set of experiments is expected to complete within a few hours overall.

5 Timeline

For our 14-day project timeline, we expect the following distribution with possible overlaps:

- Research (1-2 days): reviewing related work and lecture material
- Implementation (3-4 days): setting up models, augmentations, and explanation methods
- Experiments (3-4 days): running controlled experiments across seeds and settings
- Analysis (2-3 days): computing stability metrics and visualizing results
- Reporting (2-3 days): writing the final report and preparing figures