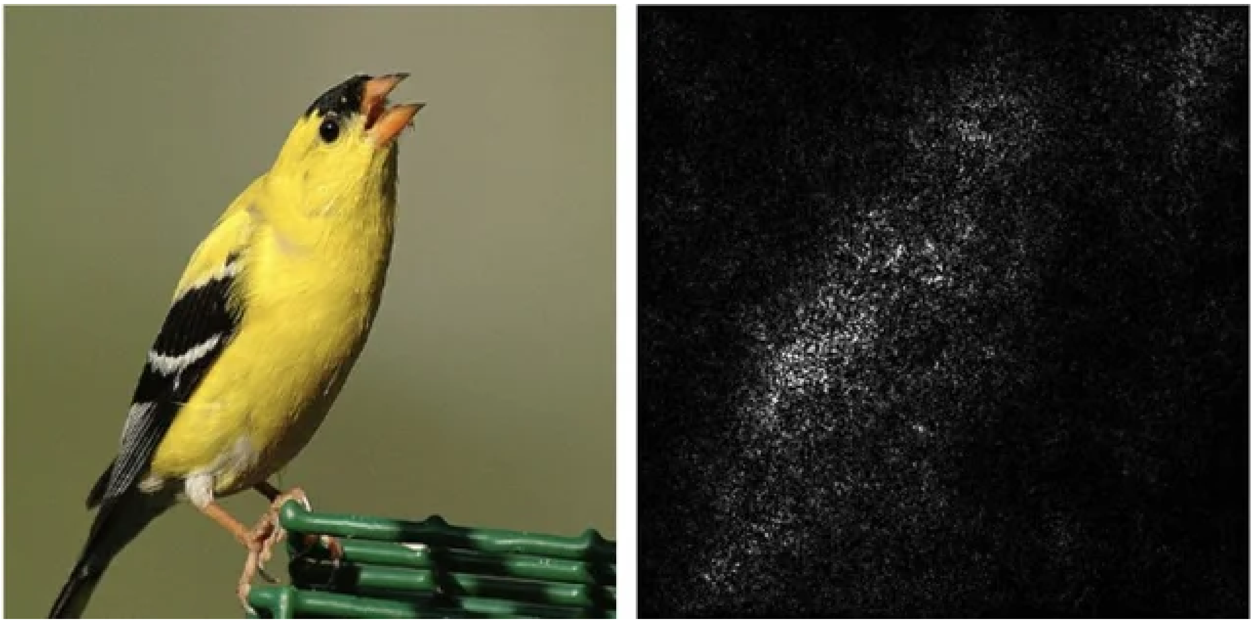
Advanced Machine Learning Project



Feature Attribution using Integrated Gradients

By Shivarjun Sarkar, Abhinav Sharma, Nir Raunch, Scott Fields & Harsh Mehta

Introduction

Over the past decade, predictive models are increasingly being considered for deployment in high-stakes domains. However, the successful adoption of predictive models in these settings depends heavily on how well decision makers can understand and consequently trust their functionality. Only if decision makers have a clear picture of the behavior of these models can they assess when and how much to rely on these models, detect potential biases in them, and develop strategies for improving them.

The increasing complexity as well as the proprietary nature of predictive models is making it challenging to understand these complex black boxes, thus motivating the need for tools and techniques that can explain them in a faithful and human interpretable manner. This is one of the biggest issues faced by Deep Learning is its lack of interpretability when compared to simpler models. Especially within business settings, it helps to be able to explain what factors are having the greatest impact on the predictions. Additionally, it may help to know which features have insignificant contributions.

While models like linear regression have simple weights and tree-based models have feature importance, the complexity and layers of deep learning hinders its interpretability. Several techniques have been recently proposed to construct post hoc explanations of complex predictive models. While these techniques differ in a variety of ways, they can be broadly categorized into **perturbation vs. gradient based techniques,** based on the approaches they employ to generate explanations.

One method attempting to solve this issue is **Integrated Gradients**. Integrated gradients work by considering the straight line path from the baseline to the input, and compute the gradients at all points along the path. Integrated gradients are obtained by accumulating these gradients. Specifically, integrated gradients are defined as the path integral of the gradients along the straight line path from the baseline x0 to the input x. This method was suggested by Mukund Sundararajan, Ankur Taly & Qiqi Yan. We will show why this method is a superior approach than vanilla gradients and why it satisfies axioms like **Sensitivity and Implementation Invariance** Additionally, our team will evaluate the Integrated Gradients method compared to other perturbation(SHAP, LIME) and gradient based(Deep Lift and LRP) feature attribution methods. We will also attempt to demonstrate that both methods converge to the same output (explanation) in expectation, i.e., when the number of perturbed samples used by these methods is large.

Datasets

https://www.image-net.org/download.php [Images]
https://www.kaggle.com/hmavrodiev/london-bike-sharing-dataset [Tabular]

Approach

As mentioned above, deep learning methods are often deprioritized due to lack of interpretability. We start building up on current explainable AI methods such as **LIME and Shapley(SHAP)** and present the case for why faster and better is needed for interpreting deep learning models. Similar to Shapley the idea is to obtain marginal contributions to the dependent variable from our feature space. However, we leverage gradients being computed in a neutral net for our use-case. To gauge in the sensitivity part of our feature space, we define a baseline (say a black image in case of image data) and create inputs that are consequently more similar to our feature space (images with increasing intensity) We avail the gradients from all these input settings and integrate the gradients to be able to explain exactly which part of the input (pixel in terms of images) is contributing to the explainability of the predictions. We plan to perform an **axiomatic validation** of the same to prove integrated gradients being a viable and robust solution for the neutral net interpretability problem.

Most attribution methods also rely on comparing the input at hand to one or more counterfactual inputs (which are often referred to as “reference points” or “baselines”). However, the role of counterfactuals here is to tease apart the relative importance of features rather than to identify new instances with favorable predictions. SHAP, which is based on the concept of Shapley Values from game theory, operates by considering counterfactuals that “turn off” features and noting the marginal effect on the prediction. In other words, we note the change in prediction when a feature is made absent. This is done for all combinations of features, and a certain weighted average of the marginal effect is computed. Integrated Gradients examines the gradients at all counterfactual points that interpolate between the input at hand, and a certain “all off” input (i.e., one where features are turned off). LIME operates by examining counterfactuals that randomly perturb features in the vicinity of the input.

We will try to show how the **feature attributions done by the various methods** differ from each other and analyze how each method performs on **images and tabular data**. We will attempt to show that **in expectation the perturbation and gradient based methods converge to the same output**. We will also try to understand **how robust the models are**, in particular the concept of Integrated Gradients.

References

[1] Axiomatic Attribution for Deep Networks

[2] Attributing a deep network's prediction to its input features

[3] Integrated gradients

[4] https://arxiv.org/pdf/2102.10618.pdf

[5] https://arxiv.org/pdf/2102.10618.pdf

[6] https://dl.acm.org/doi/pdf/10.1145/3351095.3375624