|  |
| --- |
|  |
| SHapley Additive exPlanations (SHAP) |
| CS677 Deep learning – fall 2020 |
| Fernando Rios | Hassan Ouanir | Maha Faruque |

**Introduction**

Most Machine Learning algorithms are complicated and not easy to understand. In fact, most of the time, these complicated deep learning models are referred to as a ‘black box’ because they may be good for making predictions but not for interpreting. That is where different methods for explainability like SHAP is needed so that users can interpret the predictions from the models. Machine Learning and Deep Learning have changed tremendously as time goes on and have expanded into many more fields of industry such as retail, healthcare, etc. Machine Learning algorithms may have started simply and with a goal of studying patterns and relationships but along the way, they have become more complex and the more sophisticated they become the more difficult it can be to explain the model to a business. A good example would be banking and how it is still common for them to use more simple models like linear models because it is crucial for them to be able to interpret the model in order to make decisions. The disadvantage of using simpler models is that the more complex models commonly have better performance, but they face the issue of interpretability. So, on one hand, you may decide to use a simpler model with lesser performance and have no issues with interpretability but on the other hand, you may decide to use a complex model with better performance but then you face the challenge of interpretability. In order to use more complex models and deal with the gaps of interpretability, we use methods like SHAP.

SHAP is a game theoretic method and is used for explaining predictions. The SHAP method assigns every feature a value for a prediction. This approach uses the Shapley values from cooperative game theory. The Shapley values inform us of how distributed the predictions are among the features. However, there are still other methods for interpreting model predictions and there is a struggle that exists for when one method is preferable than another. Explainable AI has become a field of interest to many individuals and many new methods and improvements are continuously occurring and that adds to the challenge for when one method may be more preferred than another. In this report, we will discuss LIME which is another method.

**Shapley Values**

SHAP is a combination of LIME and Shapley values. We will get to LIME in a later section of this report but SHAP is a special case of Shapley values. SHAP values are utilized when given a deep learning model which contains features as inputs and has predictions as outputs and you need to interpret what decisions the model is doing. SHAP values cut down a prediction piece by piece to show the real impact of each feature. By doing so, SHAP values explain the effect of a certain value for any given feature instead of the prediction we would have if that feature had a baseline value. The SHAP explanation method creates Shapley values from coalitional game theory.

The Shapley value calculation is the average of the marginal contributions of a feature value in all permutations. Shapley values are used to measure the contributions of the predictors. Shapley regression values are feature importance for linear models with multicollinearity. Shapley sampling values are used to interpret models by using sampling approximations and the impact of removing a variable from the model by integrating samples from the training set. Quantitative input influence is the broad framework that is used for more than feature attributions. It is an additive feature attribution method. All three of these methods use classic equations from game theory to interpret model predictions. A simple way to understand Shapley values is by thinking of players and a game since this is an approach from game theory. So, consider a prediction being interpreted by assuming each feature value is a player in a game and the prediction is the payout. So, Shapley value informs us of how fairly the payout is distributed among all the feature values. Shapley value also satisfies the following properties: efficiency, symmetry, dummy, and additivity. An important note to mention when implementing Shapley is that there is no easy way of removing a feature for a given instance’s prediction which is required in the calculation of the Shapley value.

**Baselines**

When computing the Shapley values for a model, a baseline needs to be chosen, a baseline from the domain of the model that would be appropriate to compare against. To go from the baseline to the wanted target instance, multiple trials, known as counterfactuals, needs to occur in the calculation process. The model then predicts outcomes for these counterfactuals which are needed to calculate the Shapley values. It’s important to pick a good baseline since the baseline can heavily influence the attributions.

In order to choose a good baseline, you need to know the kinds of baseline there are. So, there are uninformative and informative baselines. Uninformative baseline is where no additional information is presented and is like an average instance from the training data or it can be like an instance that has low information which produces a high entropy model prediction. Informative baseline has high information and specifies the most crucial feature attributions that were produced in the observed outcome difference. An informative baseline is good to use when high information is needed and there is a need to understand the feature differences that may have resulted from the question being asked.

**Attribution Methods & Aggregate Attributions**

Due to the nature of Shapley value, it isn’t practical to calculate the exact value since it scales exponentially to the number of input features. However, there are three very useful attribution methods that can help with the model across several data modalities. The three attribution methods for TensorFlow models are: integrated gradients, samples Shapley, and XRAI. Integrated gradient is helpful for neural networks and differentiable models since it provides computational leads for large input feature spaces. Sampled Shapley is more useful for non-differentiable models such as models that contain meta-ensembles of trees and neural networks. XRAI is most useful for image models when needed to localize attributions at the pixel vs region level. The catch with attribution methods is that they rely heavily on the efficient approximations of the Shapley value. Due to that, it is important to quantify the error and better the approximation by altering some parameters of the attribution method.

Shapley-based attributions contain unique properties which allow for comparisons between models if the baseline is the same. In fact, it is possible to aggregate these attributions along several instances for the purpose of achieving a more comprehensive look of the model’s behavior and better the confidence by looking at a single instance attribution. There are different ways to conduct an analysis, such as using the same model and multiple slices in the data or using different models on the same dataset.

**LIME vs SHAP**

LIME and SHAP are both input attribution methods in the field of explainable AI and have contributed to the rapidly advancing research progress in this field. SHAP uses Shapley values to consider every possible prediction for an instance using all possible iterations of inputs. Due to this extensive and exhaustive approach, that is why many of the properties like efficiency are satisfied. LIME on the other hand, creates linear models for every prediction to interpret how the model works in that space. LIME may be considered a subset of SHAP but does not contain the same properties. SHAP is an exhaustive approach and therefore takes longer than LIME. The field of explainable AI is still growing and constantly improving, but many of the methods are not optimized for all types of models yet including SHAP. LIME does not offer the same guarantee that Shapley value does of having a fair distribution of contribution for each variable. However, SHAP depends on the datasets to infer a baseline value. Both methods are great for explainability and both have their advantages and limitations.

**Conclusion**

Explainability methods like SHAP, LIME, etc. have been growing and improving recently as there is more demand for these methods to help users interpret predictions for complex models. Although SHAP is gaining a lot of attention, it is important to remember that it has its limitations like any other approach. SHAP should be used as a sort of support tool and not a replacement for other model analysis processes. This report contains many details and key points for the SHAP approach and contains a comparison to the LIME approach. Each approach has its own usefulness and should be used when appropriate. No explainability method is perfect and they all have their pitfalls but the purpose is the same for all explainability methods, which is, to help individuals be well-versed and to ultimately be able to interpret the predictions from the model.

**References**

*AI Explanations Whitepaper*. Google, storage.googleapis.com/cloud-ai-whitepapers/AI%20Explainability%20Whitepaper.pdf.

*A Unified Approach to Interpreting Model Predictions*. Lundberg, Scott, and Su-In Lee. *ArXiv.org*, 25 Nov. 2017, arxiv.org/abs/1705.07874.